Chapter 1
Concepts of Symbiotic Robot Organisms

1.1 From Robot Swarm to Artificial Organisms: Self-organization of Structures, Adaptivity and Self-development

Serge Kernbach

Collective systems possess very interesting properties. They are flexible, reliable, have extended capabilities for adaptation, self-organization and self-development. Many natural systems, such as atomic or molecular phenomena (Balzani *et al.*, 2003), social insects or animals (Camazine *et al.*, 2003) are collective on the level of their aggregation or population. Since these survived millions of years in the course of multiple evolutionary processes, we can learn from them how to achieve long-term stability, diverse functionality and reliability for artificial collective systems.

In technical progress, in particular in robotics (Siciliano & Khatib, 2008), artificial collective systems are also a focus of research and technological development. However, due to a specific structural and functional organization, collective systems represent several essential challenges for researchers. We can emphasize three most important challenges whose solutions may contribute not only to technological advancements but also to theoretical understanding of underlying processes in collective systems. These challenges are structural adaptability, evolvability and self-development and, finally, a long-term independency of these systems.

Structures in natural systems, for example protein structures of bio-molecular systems (Alberts *et al.*, 2008) or social structures in groups of animals, are a major factor in defining a collective functionality and finally a collective behavior. Features of individuals are still important, however a collective is capable of demonstrating a functionality not available to individuals. Designers of artificial systems are able to create not only functions and behavior, but also to define new structures (Kernbach, 2008). The interplay between structures, functions and behavior allows multiple self-organizing and self-developmental processes. Many holistic and reduction approaches have been suggested to deal with this problem, such as classical and modern control theory (Kolesnikov, 1994), methods

P. Levi and S. Kernbach (Eds.): Symbiotic Multi-Robot Organisms, COSMOS 7, pp. 5–77.
springerlink.com © Springer-Verlag Berlin Heidelberg 2010

from distributed AI and multi-agent systems (Weiss, 1999), bio-inspired solutions (Floreano & Mattiussi, 2008) and synergetics (Haken, 1988). However, due to an emergence on functional and behavioral levels, the complexity of such structural systems is very high. This, and a lack of understanding of structural phenomena, often hinders researchers in realizing desired properties of artificial systems by using structure-function dependencies.

Adaptive and self-developmental processes in collective systems happen on different levels and are defined in a wide range from adaptation, self-defence and self-healing to unbounded self-evolving. These are very attractive and desired properties allowing systems to develop themselves from simple to complex ones, to increase their own functional diversity and improve their own control. Working with these processes in artificial systems, we encounter several problems. First, technical collective systems are driven by two different forces: design goal and adaptive fitness. Such issues as long-term controllability, predictability and validation are the focus of research here. Secondly, biological concepts of adaptation and self-development are valid for populations, involving such processes as death, birth, reproduction and others, which are not very natural for robotics. Therefore we are looking for such approaches, which can be applied even for a single robot, do not require very powerful computational resources and may be utilized in short-term operational situations. We call them *on-line* and *on-board* self-developmental processes. It is obvious, that an exploration of new techniques, for example a combination of evolution and self-organization (so called "evolutionary self-organization") or traditional AI-based decision making, planning or learning with bio-inspired approaches, is required.

Long-term independency and is an integrating property of many other factors. It can be understood as a capability to work independently a long time period without the need for continuous human maintenance and supervision. This property depends largely on reliability and good engineering of the system, on capabilities of regulatory autonomy to deal with unbounded issues in self-development, and can be considered as the most challenging task in autonomous robotics.

Research in collective robotics is intensively addressing these problems. One methodological approach is represented by networked robotics (Kumar *et al.*, 2008). Networked robotics assumes that essential communication resources are necessary for problem solving. Another alternative is given by swarm robotics (Parker, 2008), which involves interactions instead of communication. Both approaches utilize individual functionality of robots in creating *behavioral emergence*, i.e. *individual functionality* $\rightarrow$ *emergent collective behavior*. This scheme can address all three challenges from the viewpoint of functions and behavior, such as exploring social hierarchies, ecological dynamics or population-based evolution. However, an essential problem still remains untouched by these systems; they cannot build structures, i.e. no structural level is available.

The structural level is important for investigating the generating dependencies such as *collective structures* $\rightarrow$ *emergent collective functionality* $\rightarrow$ *emergent collective behavior*. This reflects the very important role of *functional emergence* which appears in such processes as embryogenetic, morphogenetic and ontogenetic development, cell differentiation, intrinsic evolution in robot systems and others.

To investigate dependencies between structures and functions, we need a new class of robot systems having a cellular-like structure and capable of autonomous self-assembling into organisms. Since these new systems possess self-developmental features, it is expected that several system-relevant (i.e. not relevant for individual robots) mechanisms and functions will emerge. A capability of artificial organisms to modify their own morphology and size means that such mechanisms and functions should be very flexible, scalable and be implemented in a specific cooperative way, i.e. without essential centralization. Such cooperation between different regulative, structural and behavioral aspects is the central issue of artificial organisms. For a similar co-dependent functionality of natural systems, the notion of symbiosis is used. To emphasize the cooperative aspect of structural self-developmental phenomena in these systems, we call them *symbiotic multi-robot organisms*.

Symbiotic multi-robot organisms allow us to address the challenges of collective robotics from the viewpoint of structures and developmental plasticity. This increases the number of available degrees of freedom for emerging control, where the system can change its own structures to adapt to its environment. This enables new structural self-organization and can involve bounded and unbounded self-developmental processes. In general, this can result in extended reliability, adaptability and long-term independence of such artificial systems. In addition to new technology, this may lead to deeper understanding the phenomena of collective intelligence and artificial evolution.

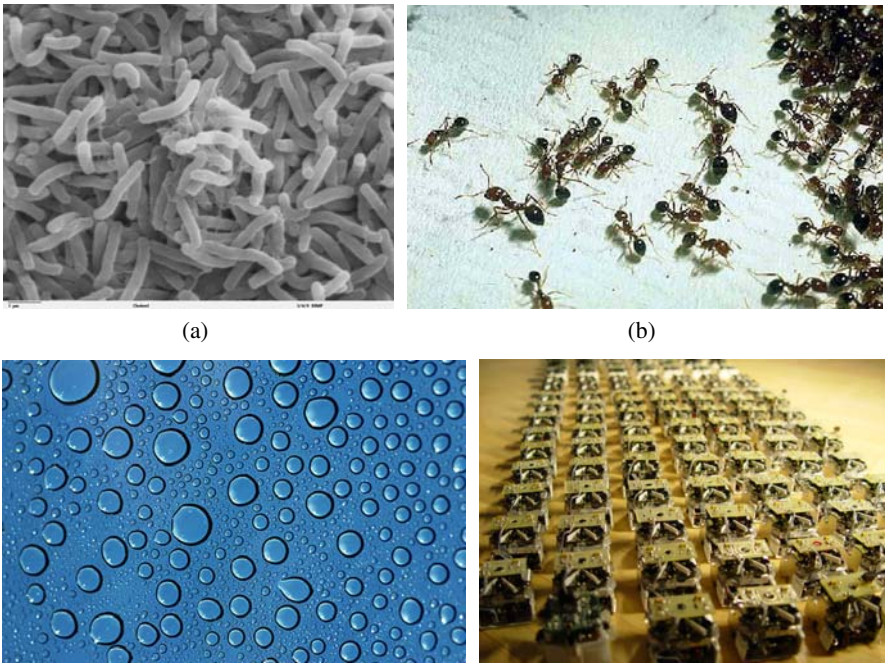
1.1.1 Mono- and Multi- functional Artificial Self-organization

Collective systems consist of many independent interacting elements, we can find them in living and nonliving nature, see Fig. 1.1(b,c); at all scales: from nano- and micro-scales, such as bacteria in Fig. 1.1(a) to large-scales such as galaxies. Currently, we stet a growth of different artificial collective systems, see Fig. 1.1(d).

Collective systems possess many amazing properties and phenomena, which fascinate researchers. These systems scale well and are very reliable, they possess different self-regulating mechanisms, they are capable of self-organization and emergent phenomena – when ordered macroscopic behavior emerges from interactions among microscopic elements. Macroscopic behavior is often visible as different dynamic or static patterns, as shown in Fig. 1.2. When representing the structure of collective systems, we have to draw two representation levels:

microscopic level of consideration is the level of interacting elements (Fig. 1.1), where individual behavior is the focus and collective properties of the system are not observable; **macroscopic level of consideration** is the level where the whole collective behavior (Fig. 1.2) is visible to an external observer and properties of individuals are neglected.

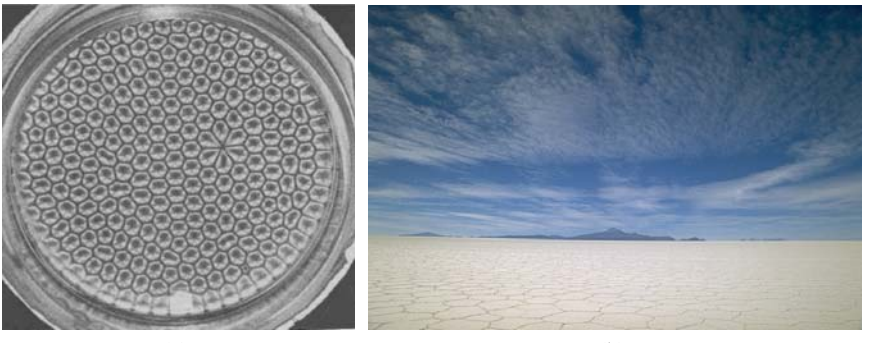
One of the key problems in collective systems is that *"we cannot proceed directly from microscopic to macroscopic level, i.e. from individual models to emergent*



(a) (b)

(c) (d)

Fig. 1.1 Examples of collective systems – microscopic level of consideration; **(a)** Vibrio cholera bacteria in SEM micrograph (courtesy of Patrick Hunt and Andreea Seicean phunt@stanford.edu); **(b)** Swarm of ants; **(c)** Water droplets on glass; **(d)** Artificial swarm of micro-robots Jasmine.



(a) (b)

Fig. 1.2 Macroscopic patterns emerge as a result of different self-organizing processes – macroscopic level of consideration; **(a)** Benard-cells, (from presentation of E. Laurien, Rayleigh-Benard-Konvektion, University of Stuttgart,Germany); **(b)** Patterns in nature (with permission of Bernhard Mühr).

collective behavior. If we need some specific collective behavior, we do not know which individual models can produce it" (Arnold, 1988, p.212). The main reason is the enormous complexity generated by interactions among components. Each interaction step creates a new correlation cascade and this dramatically increases the total complexity (Prigogine, 1962). However, complexity in collective systems is distributed in different way. To describe it, the notion of operational principle is introduced (Kernbach, 2008).

Collective systems with **vertical operational principle** have strong hierarchies in their organization: elements on the lowest-level are ruled by a few elements on higher levels. The organization and distribution of complexity looks like a pyramid: high complexity below and a low complexity on the top. Collective systems with **horizontal operational principle** do not have hierarchical organization, their complexity is similar on all levels of abstraction.

Operational principles are directly related to the problem of structures and control (Turchin, 1977). Collective systems with the vertical operational principle utilize hierarchical control (Vinter, 2000). Systems with horizontal operational principle use self-organizing control mechanisms (usually away from instabilities) without growing hierarchies. This makes the development of such mechanisms more difficult, however horizontal systems are very scalable and reliable.

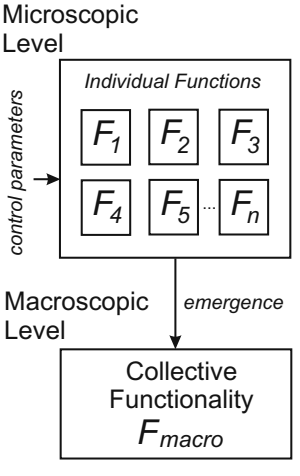


Fig. 1.3 Two levels of consideration – relation between functions and behavior.

Elements on the microscopic level behave according some local interaction rules, we denote them as functions F_i, see Fig. 1.3. Moreover, all of F_i are on the same level of hierarchy, in other words there is no explicit control in this system. F_i have some control parameters, which can depend on environmental conditions as well as on internal parameters. On the macroscopic level we observe a collective functionality F_{macro}. There are two types of F_{macro}. Let us consider the Fig. 1.4(a). This is a typical heterogeneous collective traffic system, performing a rescue operation on the highway. Combining microscopic functionality F_i of each vehicle in this system, we can observe a multitude of F_{macro}, such as collective transportation, different rescue missions and others. In Fig. 1.4(b) we show another collective system – artificial salamander, developed in biologically Inspired Robotics Group, EPFL (Chevallier *et al.*, 2008). Each segment of this system is autonomous in terms of behavioral activities and is synchronized with other segments

(a) (b)

Fig. 1.4 Examples of collective systems with mono- and multi- functional F_{macro}; **(a)** Heterogeneous multi-functional collective system; **(b)** Artificial salamander (biologically inspired robotics group, EPFL), demonstrating mono-functional behavior (Chevallier *et al.*, 2008).

through bio-inspired signal transmission. All segments are connected to each other and the whole system can demonstrate a functional pattern of specific anguine-like locomotion.

F_{macro} is **multi-functional**, when it can demonstrate many different functional patterns. Moreover, diversity degree of F_{macro} depends on heterogeneity and a common number of elements. F_{macro} is **mono-functional**, when it can demonstrate only one homogeneous functional pattern, parameterized by one or several parameters.

Both, mono- and multi- functional systems possess several advantages and disadvantages. Most mono-functional systems (both collective and not collective) are cheaper and simpler from a control perspective. When we consider modern manufacturing such as flexible (Qiao *et al.*, 2006) or reconfigurable manufacturing systems (Galan *et al.*, 2007), we encounter mostly only mono-functional systems. In general, mono-functional systems are more attractive for creating technically useful behavior. Disadvantages of mono-functional systems are low reliability and scalability compared to multi-functional systems.

Collective functionality F_{macro} can be random, chaotic and, in several cases, can represent an ordered (or synchronized) pattern. This ordered collective behavior may have forms of symmetrical patterns, as shown in Fig. 1.2 or, in general, it may be synchronized in spatial, temporal and functional ways. The process, leading to ordered macroscopic behavior F_{macro} through interactions between F_i, is denoted as self-organization. "*The self-organization is a process by which global external influence stimulate the start of internal for the system mechanisms, which bring forth the origin of specific structures in it*" (Bushev, 1994, p. 24). We will denote this process as **functional self-organization**. In artificial collective systems, the designer can change interaction among elements and thereby modify their collective properties.

All F_i can be created in such a specific way, that collective behavior has an ordered character – this can be denoted as **artificial functional self-organization**. As already mentioned, the problem of creating purposeful self-organization, as well as a general problem of emergent phenomena, is one of complexity. There is no way to predict analytically such rules F, which will lead to the desired collective behavior. However, self-organization possesses several advantages making this phenomenon attractive in practical applications:

- *Flexibility of self-organized collective behavior.* Collective behavior in artificial systems can be easily changed by modifying interactions. Even a small change, e.g. around critical points, can qualitatively change the whole collective behavior. The mechanisms of "adaptive self-organization" can provide a high degree of flexibility.

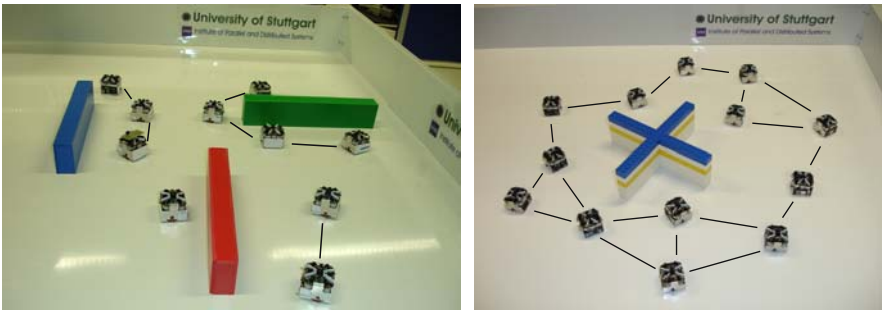
- *Reliability and Scalability.* Since all F_i are on the same level of hierarchy, some elements can be removed (destroyed) without changing collective functionality. Scalability of collective systems is based on the same principle. When self-organized mechanisms provide scalability for load parameters, like number of elements or diversity (Constantinescu *et al.*, 2004), collective systems may be scalable or even super-scalable, see Sect. 1.3.

In the following section we extend the notion of self-organization for the structural case.

1.1.2 Collective Robotics: Problem of Structures

Consider now self-organization in technical collective systems, in particular in collective robotics, we remark that these systems possess additional degree of freedom. They are able to change their own macroscopic structures. These structures are topologies of information networks, neighborhood connectivity or even 3D structures. We consider an example of such structures based on collective perception in a robot swarm, as shown in Fig. 1.5. To recognize large objects, small swarm robots should create a network around the object of interest. Important are not only topologies (open, closed, chain-like, star-like), but also connectivity of robots because it results in different object recognition algorithms running in each robot (Pradier, 2005). In Fig. 1.5 we can observe two different networks, open-chain in Fig. 1.5(a) in experiments with color perception in swarm (Zetterström, 2006) and surrounding the object in Fig. 1.5(b) in collective perception of large objects (Kornienko *et al.*, 2005). In this way, robots have several structural rules of how to create different networks. In turn, the emerging structure influences the local functional rules in each robot and, finally, the whole swarm demonstrates different collective perception behavior.

Normally, structures and functionality are closely related to each other. By changing macroscopic structure, the system also changes its own functionality and correspondingly behavior (Kernbach, 2008). The relation between structures, functions and behavior can be represented as shown in Fig. 1.6. We denote this relationship as "generating" because the upper level generates the lower level, i.e. structures generate functions and functions generate behavior. However, the relation between structures and functions is non-trivial and several types of this relation are observed.



(a) (b)

Fig. 1.5 Examples of different network structures in collective perception, lines show communication between robots; **(a)** Experiments in color perception in swarm, robots create open-chain network and recognize an object by a feature matching (Zetterström, 2006); **(b)** Collective perception of large objects, robots create close-chain network and recognize an object by a probabilistic approach (Kornienko *et al.*, 2005).

In the first case, such as collective perception in Fig. 1.5, structural rules generate functional rules. For example, the structure of the network is defined by structural rules, by the number of locally achievable robots and by geometry/size of the object. In turn, the amount of information flowing from robot to robot depends on the topology. Each robot adapts its own processing rules (i.e. functionality) to this information flow. Collective perception is finally defined by combination of different processing rules.

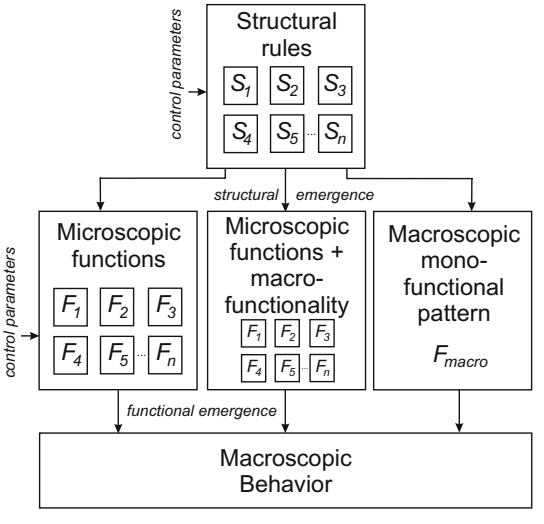
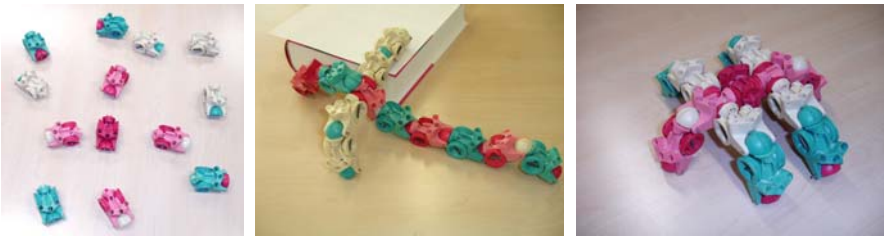


Fig. 1.6 Relation between structures, functions and behavior.

In this way, collective behavior is defined by interacting individual functionalities, in turn, defined by the structure of the network. More generally, we can observe here two emergent processes at functional and structural levels with multi-functional collective activity.

In the second case, elements create different structures with mono-functional macroscopic functionality. This functionality ultimately demonstrates a behavior. A



(a) (b) (c)

Fig. 1.7 Different structures and functionality; **(a)** Initial not aggregated modules; **(b)** Aggregated snake-legged structure and resulting crawling locomotive functionality; **(b)** Aggregated 6-leg structure and resulting legged functionality.

famous example are L-systems, proposed by Aristid Lindenmayer in 1968 (see e.g., (Prusinkiewicz & Hanan, 1980)). The central notion of L-systems is the concept of rewriting, i.e. successive replacing a simple initial object using a set of rewriting rules. These replacing rules can be viewed as structural rules, producing different tree-like fixed patterns (they can be viewed as a mono-functional activity), see more in Sect. 2.4. Robots, which are able to dock to each other and create 3D functional organisms, are another example of mono-functional macroscopic functionality. Topological models of such robots are shown in Fig. 1.7(a). These models have rotational and bending degrees of freedom (Kornienko *et al.*, 2007). Following specific connection (structural) rules, they can create different macroscopic structures, such as snake-like or 6-legged systems, see Fig. 1.7(b-c). As we observe, these 3D structures possess only mono-functional locomotion, defined by the corresponding snake-like or legged motion principle.

Considering Fig. 1.7(b-c), we should take into account not only spatial functionality, but also diverse sensing, homeostatic, energetic and other processes, which will run in real robots. Depending on a spatial position, modules can specialize in performing different tasks. For example, in the topological model of a 6-leg organism shown in Fig. 1.7(c), we can imagine that only a few elements will perform actuation (e.g. they specialize as active joints), elements in the middle take a role of information processing, there are sole-, front- or back- sensor elements. We observe in this case a combination of emergent and macro-functional approaches, as shown in Fig. 1.6.

By analogy with functional self-organization, we define **structural self - organization** as a process leading to emergence of different microscopic and macroscopic functional patterns, which, in turn, emerge as collective phenomena on behavioral level. Since in artificial systems corresponding structural and functional rules can be changed, we denote self-organization in such systems as **artificial structural self-organization**. In Table 1.1 we collect several characteristics of collective systems capable of structural phenomena. The main difference between functionally and structurally self-organizing systems consists of a higher developmental

Table 1.1 Several characteristics of collective systems capable of structural phenomena.

Level	Advantages	Problems
Regulatory	Self-regulation	Long-term stability
	Internal homeostasis and self-healing	Possible communication overhead
	High developmental plasticity	Long-term controllability
Structural	High reliability and scalability	
	Dynamical change of structures	Predictability of functional emergence
Functional	Mono-functional behavior	
	Dynamical change of functionality	Predictability of behavioral emergence
	Emergence of functionality	Difficulties with analytical prediction
Behavioral	New class of adaptive behavior	Double emergence

plasticity in the last case. In the next section we consider using this plasticity for adaptation and self-development.

1.1.3 Adaptability and Self-development

In previous sections we considered collective systems capable of structural self-modification and briefly introduced the advantages of this approach in relation to extended adaptability. This section is devoted to a deeper treatment of adaptability and principles of self-modification.

Adaptability is often considered in biological terms of natural evolution (Williams, 1996) or environmental uncertainty (Conrad, 1999) as well as in management and business processes (Gurvis & Calarco, 2007). There have been several attempts to create a common theory of adaptability, such as the approach suggested by Michael Conrad (Conrad, 1999). Overviewing the vast literature on the field of adaptation, we can recognize three main streams driving further development and representing different methodologies and different approaches to adaptation. The first and oldest stream is related to the theory of adaptive control. Several early works in adaptive control date from the late 50s - early 60s (Whitaker, 1959), (Osbourne *et al.*, 1961). In the mid-late 70s there appeared several issues related to temporary stabilities (Egardt, 1979), which in turn led to iterative control re-design and identification, and contributed in the mid-80s to robust adaptive control (Anderson *et al.*, 1986), (Rohrs *et al.*, 1985). Overviews of adaptive architectures can be found in textbooks (Narendra & Annaswamy, 1989), (Sastry & Bodson, 1989), which can be generalized as a high-level architecture, shown in Fig. 1.8(a) (Anderson, 2005).

Adaptive control consists of two parts, a conventional feedback-based control loop and adaptive part, depicted by the dashed line in Fig. 1.8(a). The environment is not explicitly integrated into this model, it is implicitly reflected by introducing disturbances and by uncertainties in the plant. The goal of the adaptive part is to estimate the behavior of a plant (by the identifier) and to calculate dynamically

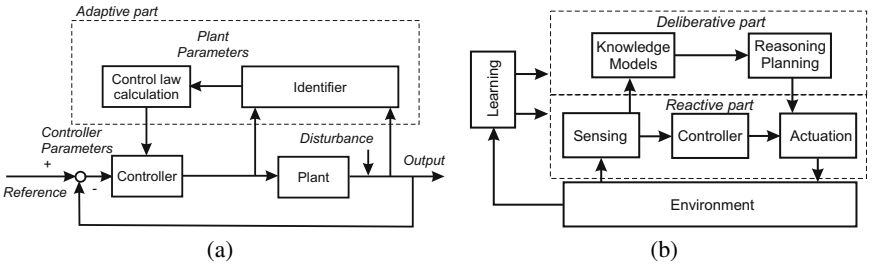


Fig. 1.8 A high-level architecture for **(a)** adaptive control **(b)** adaptive behavioral systems.

the control law (by the control law calculator). When in optimal control, a control law is designed off-line by a designer, an adaptive controller does it on-line. Most challenges in adaptive control theory are concentrated around adaptation of control to parameters of a plant when these parameters are unknown or changing.

The second mainstream of adaptation is located around adaptive behavior, which first arises within the AI community, e.g. (Beer, 1990), and involves cognitive aspects of adaption (Keijzer, 2003). There appear a few new components in the scheme from Fig. 1.8(a): explicit environment, sensing and actuation, as well as the deliberative cycle, shown in Fig. 1.8(b). When the reactive part of this scheme is in fact the optimal controller from Fig. 1.8(a), the deliberative part represents a new AI component. The adaptive system is now embedded into the unpredictable/dynamically changing environment; these systems are often referred to as situated systems (Mataric, 2002). Sensing and actuation represent a "body" of the system, intelligence (and so adaptation) is treated in term of embodiment (Pfeifer *et al.*, 2006). Achieving adaptivity in this context is spread into several approaches: different learning techniques in reactive and deliberative parts (Bull *et al.*, 2007), (Butz, 2002), (Puterman, 1994), behavior-based approaches (Kernbach *et al.*, 2009c), adaptive planning and reasoning (Weiss, 1999), biological inspiration in cognition (Cliff, 2003), evolutionary approaches (Alba & Tomassini, 2002) and many others. The goal of adaption can be formulated as achieving desired environmental responses according to some selected fitness/reward criteria.

The third mainstream towards adaptation is related to the community around distributed and software-intensive systems, computational, communication and sensor networks. With some degree of generalization, the business applications can be also related to this mainstream (SAP, 2005). The environment involves explicit users; the system itself is separated into different levels (applications), which run in parallel (Ledeczi *et al.*, 2000). The goal of adaptation here is related to scalability, self-optimization and self-protection, recognition of context, as well as to the software-engineering issues addressing reliability (Cheng *et al.*, 2008).

Based on this overview, we can say that adaptability represents a key point of systems working in real environments. Different uncertainties, variation of parameters

or even the appearance of unknown situations requires adaptive mechanisms, which allow the system to fit to these changed conditions. However, technical systems possess a goal-oriented behavior, they should be adaptive but also still be capable of achieving their design goals. To some extent, these systems are driven by two different forces: by fitness and by goal. In some cases, the goal of the system can also be focused on its own development. Here, the goal is transformed to the so-called self-concept and the system undergoes self-developmental processes. We now consider adaptive mechanisms and self-development in more detail.

Adaptation, Environment and Control. Since environmental changes require an adaptive reaction from a system, which in turn requires specific control mechanisms, we can divide changes and reactions into those forecast in advance and correspondingly those not forecast in advance. This division is relative, because in practical situations each change has forecasted and not forecasted components.

Adaptability is closely related to environmental changes and the ability of a system to react to these changes and the capability of the designer to forecast reaction of the environment to the system's response. Therefore adaptability is defined in term of the triple-relation: *environmental changes→ system's response → environmental reaction*. In general, adaptability is the ability of a collective system to achieve desired environmental reactions in accordance with a priori defined criteria by changing its own structure, functionality or behavior initiated by changed environment.

In Table 1.2 we roughly specify four different categories of environmental changes. According to environmental changes from this table, we can identify five

Table 1.2 Four types of environmental changes in robotic applications and examples of cases both forecast and not forecast in advance.

Environmental changes leading to:	Examples: Forecast in Advance	Examples: Not Forecast in Advance
Appearance of new situations	Installation of industrial robots in a new workshop	Work in previously unexplored environment (e.g. landing on Mars)
Changed functionality	Changing a type of locomotion (e.g. from wheeled to legged), when changing a terrain type	Search and rescue scenario when robots encounter unknown obstacles
Modified behavioral response	Gravitational perturbance of flying object in space and finding new control laws for engines	Distributed control of legged locomotion for obstacles of random geometry
Optimization of parameters	Changing of day/night light and adapting intensity of additional light	Adapting locomotive parameters for randomly moving obstacles

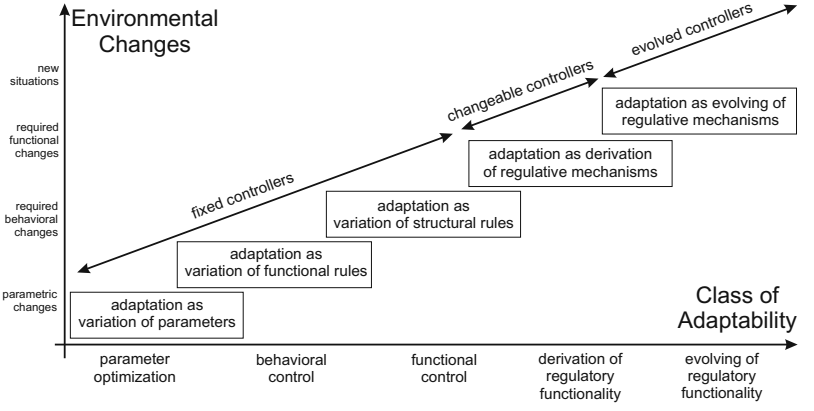


Fig. 1.9 Different adaptivity mechanisms in collective systems.

different classes of adaptability in collective systems, capable of structural phenomena: optimization mechanisms; behavioral control; functional control; derivation of new regulatory functionality and, finally, evolving of new regulatory functionality. These mechanisms are graphically represented in Fig. 1.9. To implement these adaptivity mechanisms we need to introduce two additional levels into the collective system from Fig. 1.6. The first level is related to control, we call it the *regulative level*, see Fig. 1.10. We find on this level different controllers, such as explicit and implicit rule-based (artificial neural networks), different bio-inspired, self-referred or learning systems.

These controllers influence structural or functional rules as well as change parameters of a corresponding level. All controllers work on the scheme *change of input parameters* $\rightarrow$ *changes of output parameters/rules*. The main goal of the regulative level is to maintain internal homeostasis of the system, to execute different tasks or, more generally, to demonstrate purposeful behavior depending on external conditions. Controllers at the regulative level allow some degree of adaptability for the system.

In detail, it depends to which extend a designer of these controllers was able to foresee possible changes of an environment and to integrate a reaction on these changes into controllers. The controllers allow different degrees of reaction on changes. However, the system at the regulative level is able to react only to changes whose parameter range was predicted in advance during the development of controllers or learning mechanisms. To react to such changes, which are not predictable at the design stage, we need to introduce a second level, which can modify regulative controllers – we denote this as the *generating level*. Following the scheme of adaptivity from Fig. 1.9, the generating level contains different deriving and evolving mechanisms.

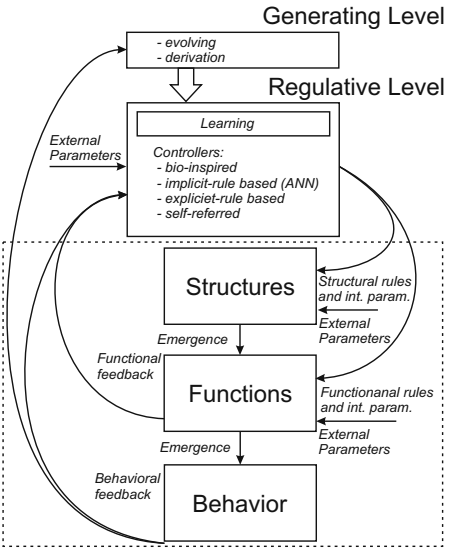


Fig. 1.10 Functional scheme of regulative and generating levels in structural systems.

Deriving is primarily related to distributed problem solving and planning approaches, known in the multi-agent community (Durfee, 1999), symbolic tasks decomposition (Kornienko *et al.*, 2004b), structural decomposition (Scassellati, 1998), self-referred dynamics (Kataoka & Kaneko, 2000) and others. These approaches are fast, deliver a predictable behavior and can be applied when a new situation is at least structurally known. Evolving is basically related to evolutionary approaches, see e.g. (Koza, 1992), and can be applied when the situation is completely unknown and a large search space of possible solutions should be explored. Recently, evolutionary approaches have been applied to a wide class of robotic problems (Floreano *et al.*, 2008a).

Invariant Goals, Self-Concept and Unbound Self-development. The mechanisms mentioned above allow adaptive behavior on different levels. However to avoid conflicts between achieving a goal and adapting to the environment, the goals at the generating level should be independent of adaptation processes, in other words, they should be formulated invariant to possible adaptations. There are several mechanisms expressing such an invariant property of the generating level: symmetries, conservation laws or e.g. "templates". Templates are well-known in cognitive science (Gobet & Simon, 1996) (also as "schemas" or "prototypes"), in topological research (in knot and braid theory) (Birman, 2004), as well as known as "frames" in the AI community (Minsky, 1977). The idea of a template is to describe most general "stereotypical" properties or features of some common class of situations/processes/objects.

As mentioned, goals can be focused on the system itself, i.e. they involve a self-concept. To explain the idea of the self-concept, we consider the case when a system should have a specific form, such as for the symmetric movement of legs, segmented (as in insects) construction of body, or there are imposed constraints or a priori desired properties. The self-concept contains in a compressed form a description of these "own" constraints or properties. The notion of self-concept originated in human psychological research, e.g. (McLean *et al.*, 2007), and is basically related to self-developmental processes. Self-development, is mostly known in human psychology, e.g. (Maslow, 1998), (McLean *et al.*, 2007), (Avstreih, 1981); in robotics the self-development usually refers to learning (Oudeyer & Kaplan, 2004) and

especially to ANN-based applications. Recently, there appear several works which apply psychological ideas to robotics, e.g. (Bonarini *et al.*, 2006), as well as the appearance of developmental robotics (Lungarella *et al.*, 2003) focused on ontogenetic processes related to cognitive science and the concept of embodiment.

Self-Development is bounded or unbounded process of functional, structural and regulatory changes undertaken by the system itself, related to its self-concept. A prerequisite is developmental plasticity on all levels. The self-concept can be expressed by symmetries, conservation laws, be planning- or fitness-driven or even have a character of unbounded metrics for open-ended evolution. Normally, self-development is initiated by differences between the self-concept and endogenous or environmental factors and may be unlimited in time and complexity.

In self-development we have to point out one principal element, related to the bounded and unbounded character of evolutionary changes. When in adaptive processes, these driving forces are mostly bounded, expressed by reward or fitness, the self-concept may include driving forces which are of unbounded character. In this way, self-development does not necessary imply any evolutionary progress, but a progress driven by the unbounded force of a self-concept. More generally, unbounded self-development (also denoted as open-ended evolution) *is characterized by a continued ability to invent new properties – so far only the evolution of life on Earth (data partly from the fossil record) and human technology (data from patents) have been shown to generate adaptive novelty in an open-ended manner* (Rasmussen *et al.*, 2004). We find some first ideas about open-ended evolution in (von Neumann, 1966) and (Waddington, 1969). Open-ended evolution is also related to indefinite growth of complexity (Ruiz-Mirazo *et al.*, 2008) and unbounded diversity (Maley, 1999). Ruiz-Mirazo and co-authors expressed the interesting idea that *"the combination of both self-assembly and self-organization processes within the same dynamic phenomenon can give rise to systems with increasing levels of molecular as well as organizational complexity"*. They also proposed to decouple genotype and phenotype from each other. A similar idea of increase homeostatic autonomy in macroevolution was proposed by (Rosslenbroich, 2009), which leads to us to not-fitness driven self-developmental processes. Several implementations of open-ended evolutionary scenarios, e.g. (Spector *et al.*, 2007), do not use any explicit behavioral fitness, moreover, there is no complexity growth in such "classical" artificial life simulator as Tierra and Avida (Standish, 2003). In this work Russell Standish proposed to improve these systems: "*a key step in doing this is to generate a process that adaptively recognises complexity, since it will be impossible to include humans in the loop, even when run on conventional computing platforms*".

These works lead us to an interesting question about the unbounded self-concept: which process can generate complexity? One of the first remarks is from von Neumann: "*synthesis of automata can proceed in such a manner that each automaton will produce other automata which are more complex and of*

Table 1.3 Several characteristics of self-developmental processes in collective systems.

Process	Developmental plasticity	Self-Concept
Regulatory	Structural and functional plasticity of the system, controllers can change their own transfer functions.	*(bound)* Achieving a targeted goal in changing environment. *(unbound)* Increasing performance characteristics.
Homeostatic	Like in the regulative case, but related to maintaining steady internal states in changing environment.	*(bounded)* Endogenous steady state. *(unbounded)* Achieving best possible homeostasis for diverse scalability metrics.
Learning	Changeable structure of regulative system.	*(bounded)* e.g. positive or negative rewards. *(unbounded)* Fitting very large (infinite) parameter space, e.g. by exploring structural-functional relations.
Planning-driven	Structural, functional and regulative plasticity.	*(bounded)* Minimizing deviations from a plan. *(unbounded)* Self-referred planning.
Fitness-driven	Structural, functional and regulative plasticity.	*(bounded)* Explicit fitness. *(unbounded)* Implicit fitness (optimizing energy balance, maximizing offsprings).
Open-ended	Capability for unbounded evolutionary activity.	*(unbounded)* Unbounded metrics.

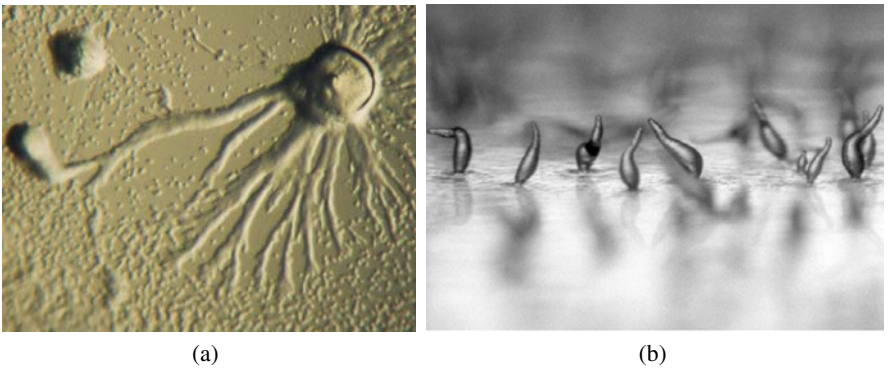
higher potentialities than itself" (von Neumann, 1966). A similar approach is observed in L-Systems (McCormack, 1993) (authors used evolutionary process but human operator in the selective loop) as well as in self-referred dynamics (Kataoka & Kaneko, 2000). It seems that structural production can lead to growth of complexity and diversity. However, considering the Kolmogorov complexity of fractal structures, which is equal to the shortest production set of rules (Kouptsov, 2008), we note the complexity of the whole fractal is independent of its size – the self-similar structural production does not increase complexity. Thus, we require that production systems include parameters which perturb generating structures. In this way, structural production rules parameterized by a random (environmental) value may lead to infinite growth of complexity and diversity, and are candidates for the unbounded self-concept. In Table 1.3 we collected several possible self-developmental processes in structural collective systems with bounded and unbounded self-concepts.

The final point which should be mentioned in this section is related to conflicts between achieving a goal and adaptive behavior. When a degree of adaptation is low, there are no essential conflicts between them. However, when plasticity is high, and the system can be hindered by adaptive processes from reaching the main goal, we are facing a new conceptual problem of a long-term controllability of adaptive and self-developmental processes. Obviously, either the goal should be formulated

in such an invariant way as allows multiple approaches for its achievements, or adaptive processes should basically be limited.

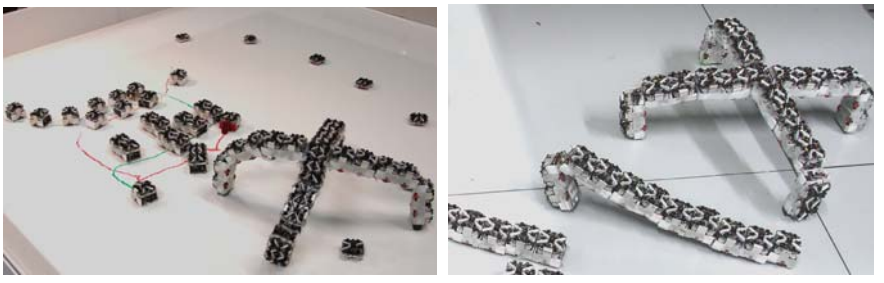
1.1.4 Artificial Symbiotic Systems: Perspectives and Challenges

As demonstrated in previous sections, collective systems capable of structural phenomena possess essential developmental plasticity, allow applications of horizontally and vertically self-organizing, planning- and fitness- based approaches and combine advantages provided by mono-functional and swarm-like systems. We can find in nature several examples of such systems, one of the most famous – Dictyostelium discoideum – social amoebae, known also as cellular slime molds (Kessin, 2001), see Fig. 1.11. These soil-living unicellular amoebae feed on bacteria. When the food resources run out, the amoebae produce and send signal molecules cAMP. This chemotaxis mechanism creates a gradient field towards an aggregation point and the collection of up to 100,000 cells first into a slug, see Fig. 1.11(a) and then into a fruiting body – a multi-cellular organism, Fig. 1.11(b). During this process, amoebae undergo different developmental processes such as cell differentiation, morphogenetic growth, self-protection, sexual and asexual reproduction and other. The principles of self-movement, aggregation and emergence of macroscopic functionality can be also demonstrated by artificial systems, in particular by swarm robots (Kornienko *et al.*, 2007). Like amoebae, swarm robots can send aggregation signals and aggregate into artificial organisms, see Fig. 1.12, and develop different macro-functionality through morphogenetic processes. In this way robots can combine collective mono-functional actuation with multi-functional collective phenomena, enabling advances in scalability and reliability. The research



(a) (b)

Fig. 1.11 Dictyostelium discoideum, commonly referred to as slime mold, capable of a transition from a collection of unicellular amoebae into a multicellular organism; **(a)** Dictyostelium Aggregation, (image source wikipedia); **(b)** Differentiation of unicellular amoeba Dictyostelium discoideum into multicellular "slugs" (100x) (Taken by Matthew Springer, University of California, San Francisco, using stereomicroscopy).



(a) (b)

Fig. 1.12 Swarm robots and a model of multi-robot organism; **(a)** Aggregation of swarm robot into several organisms; **(b)** Models of multi-robot organisms with different macro-actuation.

area of artificial organisms combines approaches and techniques from swarm, reconfigurable and evolutionary robotics.

Primarily, artificial organisms consist of heterogeneous modules. We distinguish between:

Active Module	Autonomous modules, capable of locomotion and actuation with different DoF. These modules possess independent energy source, computational and communication capabilities, see Sect. 2.1.
Passive Module	Non-autonomous modules, which do not have individual locomotion and actuation capabilities, however they are able to carry additional energy sources, structural load and other specialized passive functionality.
Tools	Autonomous modules, such as active wheels and grippers, which are specialized in some functionality. These modules also possess independent energy, computation and communication, see Sect. 2.1.6.

Following the idea of developmental plasticity, organisms are able to self-assemble and self-disassemble, see Sect. 4.5. In particular, this means that artificial organisms have two principally different states: swarm-mode and organism-mode, and progress through several phases: swarm, self-assembling, homeostatic regulation, macroscopic regulation, and so on, see Table 1.4. Organisms possess a homeostatic system, see Sect. 4.4, which performs self-regulation and energy management, common memory, common control, common event system, sensor-fusion and others, see Sects. 2.3.1.1 and 3.2. By changing scalability conditions, e.g. increasing the number of elements or diversity of elements, the self-regulatory systems try to establish an endogenous steady state. Finally, artificial organisms represent a combination of totally distributed and totally centralized regulatory systems. This is for the following reasons.

Table 1.4 Phases of the organism life cycle with respect to behavioral, functional, structural and self-developmental activities. Several phases can be executed simultaneously.

Phase	Behavioral, functional and structural activities	Self-developmental activities
Swarm	Massive-parallel execution of tasks typical for swarm: search, covering, resources localization and so on (**swarm-mode**).	Creating behavioral diversity by using population-based approaches, such as e.g. artificial sexuality and using functional self-organizing phenomena for behavioral emergence.
Self-assembling	Aggregation into planar structures, defining future topology and macroscopic functionality of an organism (**transition from swarm-mode to organism-mode**).	Diverse self-organizing, planning- and fitness-driven morphogenetic processes leading to functional emergence.
Homeostatic stabilization	Low- and intermediate-level processes, taking place immediately after mechanical self-assembly, and intended to regulate energetic, sensor, topological, information, memory and communication subsystems for the current topology of the organism (**organism-mode**).	Endogenous parametric self-regulation by using such approaches as artificial immune network, planning- or fitness-driven developmental mechanisms.
Cognitive regulation	Intermediate- and high-level processes of self-recognition (self-awareness) of own cognitive capabilities such as a number and functionality of available sensors, creating own- and world-models, creating sensomotor couplings and so on (**organism-mode**).	Creating and parameterizing internal cognitive structures and models basically by using planning- and fitness-driven mechanisms.
Macroscopic regulation	High-level processes, creating macroscopic regulative structure of the whole organism, related to macro-locomotion and macro-actuation, learning and evolving (**organism-mode**).	Main regulative self-developmental process creating control structures.
Self-repairing	Process leading to re-configuration or even removing of damaged modules from the organisms in case of any malfunctions. This is a relatively complex process based on self-diagnostic functionality of homeostatic subsystems and includes topology change phase and following homeostatic and regulative phases (**organism-mode**).	Self-developmental mechanisms are similar to homeostatic phase, however more strongly focused on self-diagnostics.
Macro-actuation	Working in the organism-mode to achieve targeted goals (**organism-mode**).	In this phase, organism performs learning and evolving activities related to behavioral and functional self-development.
Topology change	Taking a planar form for particular dissembling. This is a complex process, where all modules from old organism should not lose information collected in the organism mode. This leads to new homeostatic and regulative phases in a new organism (**organism-mode**).	This is the main structure-changing process, caused by learning, evolving or planning mechanisms decided to create new functionality and so a new structure.
Self-dis-assembling	Taking a planar form for a total dissembling (**transition from organism-mode to swarm-mode**).	Creating behavioral diversity as in the swarm phase.

By self-assembling and self-disassembling, organisms are working with different self-concepts. They may be small or large organisms with only active or passive elements, or they may have a combination of active, passive elements and tools. Thus, organisms should be able to work over a wide range of load and diverse scalability parameters. Obviously, strongly centralized regulatory systems have advantages for controlling complex macro-locomotion, however they have several difficulties with scalability and reconfigurability. Distributed swarm-like regulatory mechanisms have advantages in providing flexibility and scalability, however they have a relatively slow dynamics and can lead to communications overhead. We need a combination of central and distributed regulatory system which is fast enough for macro-locomotion but also flexible enough to deal with reconfigurability. This looks like the cooperative work of different sub-systems without building strongly centralized instances. Following bio-inspired ideas of cooperation between different species, we label these systems as symbiotic multi-robot organisms or artificial organisms.

Artificial Symbiotic Systems utilize structural developmental plasticity and a two layers regulatory architecture for creating a control system with horizontal operational principles. In particular, *symbiotic* means specific cooperation between different multi-functional regulative approaches, allowing a growth of hierarchies without building strongly centralized regulation for mono-functional collective activities.

Challenges and Perspectives. Issues of challenges in evolutionary, reconfigurable and swarm robotics have been described several times since the early 90s. We can refer to works (Mataric & Cliff, 1996), (Ficici *et al.*, 1999), (Lipson, 2000), (Sofge *et al.*, 2003), (Teo, 2004) related to challenges with fitness estimation, "reality gap" and others, whereas more recent works give overviews of challenges in robotics area (Siciliano & Khatib, 2008) such as over-motorization of reconfigurable systems or communication in swarm robotics. A combination of evolutionary techniques with other approaches creates new open questions about e.g. evolution and learning or "evolutionary self-organization" (where local rules are developed evolutionary but the self-organization remains outside of the evolutionary loop) in different scales. However, since artificial organisms combine all three areas, this results not only in a combination of problems and advantages, but also in the appearance of qualitatively new challenges. We believe that these new challenges are related to *developmental plasticity*, a *long-term independency* and to the *self-∗ issues*.

Developmental plasticity, as already discussed, in general means the structural and regulatory flexibility of basic elements, such as biological cells, hardware modules or software agents. The more plastic are basic elements, the more diverse and manifold are the resulting structures. Cells or bacteria provide great plasticity, whereas technological solutions are still far away from these biological solutions. Achieving a similar plasticity for artificial systems, by e.g. following inspirations from "natural chemistry", represents a serious challenge for the next few years.

Current robotic systems depend on maintenance, repair, specific energy sources and other infrastructure. These are not available in human-free environments or during long autonomous missions in e.g. space or ocean. A *long-term independence* means that robots can achieve their design goals without infrastructure and services provided by humans and in a variety of environments. This raises many different issues not only for the robot design, energy harvesting, reliability, adaptivity and regulatory autonomy, but also for predictability and controllability of long-term autonomy and self-development.

As already mentioned, swarm-like systems possess a high degree of redundancy and scalability. When some cell-modules in the organisms malfunction or are destroyed, they can be autonomously replaced by other cells, provided that some reserve of such cells exists. Thus, a combination of monofunctional actuation and swarm-like reliability may result in a new generation of self-monitoring and self-repairing systems. More generally, the *self-∗ issues*, i.e. self-awareness, self-reflection, self-regulation, self-reproductions, self-concept and others, are a sequence of different "self"-related problems. One of the main problems here consists not only in the lack of understanding of processes leading to e.g. self-reflection, self-awareness and consciousness, but also in the collective and emergent character of these phenomena. Thus, issues of plasticity, long-term independency and self-∗ problems are general challenges and, from this point of view, can be considered as the main benchmarks for artificial organisms.

Concluding this section, we would like to point out one important issue: artificial organisms can be viewed as extremely simplified analogues of living organisms. Both living and artificial organisms face similar problems – getting energy, surviving in the environment, different forms of self-protection and self-awareness, organization of long-term and short-term developmental processes and others. On the basis of artificial organisms we can gain deeper insights into a long-term evolution and its controllability, phenomena of individual and collective intelligence, mechanisms of multi-cellular regulation and other issues, which are of a great relevance in our understanding of the complexity of life.

1.2 Towards a Synergetic Quantum Field Theory for Evolutionary, Symbiotic Multi-Robotics

Paul Levi, Hermann Haken

We use the profound theoretical framework of the general quantum field theory (QFT), (Bjorken & Drell, 1965), (Penrose, 2006) to describe the interactions and dynamics of many-body systems which are in our case robot cells (or genes) developing an organism. Here it is important to state that we just use the formal analogy with QFT that is of purely formal nature, meaning that we operate here on a macroscopic level and no quantum effects appear or must be considered (Levi, 2009), (on the opposite to (Penrose, 1994)). In near future this assumption might be revised since NEMS (Nano-Electro-Mechanical-Systems) or even functional modules

that combine Nano-technology with molecules (NEBS, Nano-Electro-Biological-Systems) appear where quantum effects can not be excluded. The QFT-formalism is based on creation and annihilation operators that generate quantized field operators that in our contribution obey the non-relativistic QFT (Schrödinger field operators). On the opposite to the relativistic QFT these field operators can obey two different statistics (either Fermi-Dirac statistics or Bose-Einstein statistics), where these statistics are represented by two different types of commutator rules.

The operator $\mathbf{a}_j^\dagger(\underline{x}, t, s_j)$ creates a fermionic unit j (e.g. a robot-cell) at the position $\underline{x}$, at time t and the internal state s_j. In this contribution all operators will be written in bold letters. The operator $\mathbf{a}_j(\underline{x}, t, s_j)$ annihilates such a unit. It represents in mathematical terms the Hermitian conjugate operator. These two operators obey the exclusion principle of Pauli (e.g. no two robot-cells can take the same position and all other indices must also be different); therefore they anti-commutate (see appendix, Sect. 1.2.5.5): $\{\mathbf{a}, \mathbf{a}^\dagger\} = \mathbf{a}\mathbf{a}^\dagger + \mathbf{a}^\dagger\mathbf{a} = 1$. More generally spoken we consider every fermionic unit as an agent (Levi, 1989) that can directly interact with other agents or interact via exchanges of different messages with other agents. The notion of agent should clearly demonstrate our intention to include already on the level of robot-cells (or general to a fermionic units) cognitive abilities (Floreano & Mattiussi, 2008), (Trianni, 2008) that allow these units to aquire information and to perform also inferences that usually cannot be performed by units that are created by the classical QFT-theory.

In this formalism the direct interaction between different agents (on operator level) can be described by the following expression, where $\mathbf{O}(\underline{x}, \underline{x}')$ defines a position dependent operator (e.g. a transition operator):

$$\mathbf{a}_j^\dagger(\underline{x}, t, s_j)\, \mathbf{a}_k^\dagger(\underline{x}', t, s_k)\, \mathbf{O}(\underline{x}, \underline{x}')\, \mathbf{a}_k(\underline{x}', t, s_k)\, \mathbf{a}_j(\underline{x}, t, s_j). \tag{1.1}$$

The other class of operators are bosonic creation operators $\mathbf{b}^\dagger$ and annihilation operators $\mathbf{b}$. They obey the commutation relation $[\mathbf{b}, \mathbf{b}^\dagger] = \mathbf{b}\mathbf{b}^\dagger - \mathbf{b}^\dagger\mathbf{b} = 1$. These two operators obey the Bose-Einstein statistics and represent the fact that bosonic units can generate several fields that can be -in contrast to fermionic fields- in the same state (e.g. identical messages). In physics these operators describe all types of forces between elementary particles that are mediated by field operators $\Psi(x)$ and that are generated by the sum of corresponding particle operators. We use these operators to generate fields that can be seen as different types of message fields (e.g modes in the laser paradigm, (Haken, 1970)). In analogy to QFT we consider the generating units as different virtual particles that we call infermons.

An interaction of an agent (fermionic unit) with such a bosonic field is e.g. given by the following formula for the self-adjoint interaction Hamilton operator $\mathbf{H}_I$:

$$\mathbf{a}_j^\dagger(\underline{x}, t, s_l)\, \mathbf{a}_j(\underline{x}, t, s_m)\, \mathbf{b}^\dagger(\underline{x}, t) + \mathbf{a}_j^\dagger(\underline{x}, t, s_m)\, \mathbf{a}_j(\underline{x}, t, s_l)\, \mathbf{b}(\underline{x}, t). \tag{1.2}$$

The term $\mathbf{a}_j^\dagger(\underline{x}, t, s_l)\mathbf{a}_j(\underline{x}, t, s_m)$ describes the state transfer of a fermionic unit j from the energetic higher state s_m to a lower energetic state s_l by creation of the state s_l and the annihilation of the state s_m at the position $\underline{x}$ and the time t. As a result of

this operation the bosonic operator $\mathbf{b}^\dagger$ generates a field of a dedicated type (sending of a message) at the same position $\underline{x}$ and time t. The second term in this equation describes the process that a bosonic field is annihilated (unit j receives a message) and therefore annihilates the lower state s_l and transfers to the upper state s_m.

In the formalism of synergetics the generated bosonic "information fields" can effect as an order field (order "parameter" field) that synchronises the exchanges of different message types between robot-cells in order to generate coherent collective activities of these cells (Levi *et al.*, 1999). This coherence can also exist if the force mediating bosons are different. But it can also occur that the synchronisation of messages is not achievable (e.g. evident by a strong lumping of messages) resulting in a decoherence of the message flow. Such a field decoherence prevent the cooperation of cells. In term of synergetics this means that there is a threshold for order parameters. Beneath this threshold there is no coherency, at the threshold a bifurcation occurs, and above the threshold the heterogeneity can be controlled.

The QFT-formalisms in the deployment of an extended operator method delivers in addition equation of motions for each kind of creation operator and annihilation operator by the following commutator e.g. for $\mathbf{a}^\dagger$:

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{a}^\dagger = \frac{\mathrm{i}}{\hbar}[\mathbf{H}_\mathrm{I}, \mathbf{a}^\dagger]. \tag{1.3}$$

A fundamental, undisturbed form of such an equation of motion for the creation operator $\mathbf{b}^\dagger$ is given by use of (1.5)

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{b}^\dagger = \mathrm{c}\mathbf{b}^\dagger - \mathrm{d}\mathbf{b}^\dagger(\mathbf{b}^\dagger\mathbf{b}). \tag{1.4}$$

This operational equation of motion, where c is a control parameter, d is proportional to a coupling parameter (often also incorrect named constant), and $\mathbf{b}^\dagger$ plays the role of an order operator field (order parameter field) is very characteristic for self-organising processes, since it represents the so-called Eigenanteile of such processes. This equation will build the basis of our approach and will be stepwise extended by perturbation and solved.

There is an additional rationale to us to involve QFT-formalism. This is the postulation that information is represented by a quantizing field operator whose components obey an equation of motion that is dictated by the interactions of both classes of operators (Haken, 2004).

As mentioned above we use fermionic operators $\mathbf{a}$ and $\mathbf{a}^\dagger$ for the construction and deconstruction of robot-cells (agents). In favour of genetic operations we use the following naming for bosonic operators: $\mathbf{cr}^\dagger$ (create cross-over) and $\mathbf{cr}$ (annihilate cross-over); $\mathbf{mt}^\dagger$ (create mutation) and $\mathbf{mt}$ (annihilate mutation). In addition we employ a more general bosonic message operator $\mathbf{ms}^\dagger$ (create a message), $\mathbf{ms}$ (annihilate a message) that is used to describe the temporal starting points and the duration of actions and can be applied to characterise conditions like temperature, pressure and concentrations under which e.g. the operations of cross-over and mutation have to take place. Here we just focus on the duration of an interaction that controls the on and off switching of individual operators. This process portrays the

selection of operations. We involve the operators $\mathbf{ms}^{\dagger}$ and **ms** to handle extrinsic and intrinsic sensor data to initiate and to terminate bosonic activities. But at last the sensor based selections result all time in start and stop instructions. Therefore we can describe with our "switch-model" all types of selection operations.

The activation respectively the deactivation of bosonic operators describe the effect of the agilities of regulation networks (Dayan & Abbott, 2001), (Alberts *et al.*, 2008). This means that we have separated and combined equations of motion for fermionic and bosonic operators, where the combined solutions can be considered as self-organised processes. Fermionic units generate bosonic fields and these exert again a feed-back "force" to the fermions. We abstain here to introduce additional genetic operators like $\mathbf{rp}^{\dagger}$ (creation of reproduction) and **rp** (destruction of reproduction), since the structure of the interactions of fermionic and bosonic units is in our case - comparable with the three body problem - already visible if we use three different types of mediating virtual bosonic particles (infermons).

In the construction of this chapter we are guided by the interaction of light and material (Haken, 1985). The model pattern is a laser that operates above the threshold. The quantum field approach approves in addition the beneficial separation of field equations and matter equations. This means that we can separate the equations of motion for the bosonic operators from the ones that are defined by e.g. for a flip operator $\mathbf{a}_{\mathrm{j}}^{\dagger}(\mathrm{m})\ \mathbf{a}_{\mathrm{j}}(\mathrm{l})$ for fermions with different internal states.

We summarise the introduction in order to give a road map for this contribution. Messages are represented by bosonic field quanta, they are generated and handled by agents (represented by fermionic operators that generate cognitive units). The message based interactions between agents generate the information that an agent can acquire. The complete approach is ambitious: QFT theory is fused with synergetics, and not enough, from this combination an unusual definition of information is deduced and distributed in a multi-agent system. It is intended that this information can be used for the description of the development of an artificial organism generated out from individual robot-cells. On this way we started our description of such a development by the development of water from H_2O molecules, since it is provable in reality. The final step do describe an organism as a many-body system where the component obey the Fermi-Dirac statistics (fermionic agents) and their interactions are mediated by different time dependent bosonic fields that represent e.g. external signals (**ms**) or different communications (**mt**, **cr**) between the components is still an ongoing work and is not finished in this contribution.

1.2.1 Cooperative (Coherent) Operations between Fermionic Units

1.2.1.1 Interaction Hamilton Operator

As it is usually done in physics we start with a Hamiltonian that describes the interaction between the fermionic operators and bosonic operators. The goal of this sub-paragraph is to present equations of motion for the three types of bosonic operators that result from their interactions with the fermionic units. This is important

since by synergetic principles the bosonic creation operators are considered as order "parameters" fields of the self-organisation process between agents and message fields (like the interaction of mater and light). Further we are looking for the control parameters that start such a self-organised process. To get these results we involve an adiabatic approach. Unfortunately these calculations are complex and long. Readers that are not accustomed to such operator calculus could consider this formalism as the way to deduce the elementary QFT-based self-organised equation that we mentioned already in the introduction (1.4).

Concerning the "agentifying" of the fermionic units this means that we have an overlay of multi-parameter fields where we have later on to analyse which parameter values primarily of the bosonic fields can guarantee the synchronisation (cooperation) of the agents by these fields.

We model the interactions between fermionic units in analogy to the resonant actions of a multimode laser in the representation of photonic statistics. This means in greater detail that if the energy E_m of state m is higher than the energy E_l of state l ($E_m - E_l > 0$) and this energy is equal e.g. to the energy $\hbar\omega_{jk}^{ms}$ of the bosonic field $\hbar\omega_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(n)\mathbf{ms}_{jk}(n)$ then cooperative (coherent) interactions between the fermionic agents can proceed. In this paper we work in the interaction representation and assume exact resonance, otherwise the Hamiltonians of the free fields must be taken into account. The interaction part of the Hamilton operator is defined by:

$$\begin{aligned}\mathbf{H}_I = {} & i\hbar \sum_{j,k,l,m,n} g_{jk}^{ms}\left(\mathbf{a}_j^{\dagger}(l)\mathbf{a}_j(m)\mathbf{ms}_{jk}^{\dagger}(n) - \mathbf{a}_j^{\dagger}(m)\mathbf{a}_j(l)\mathbf{ms}_{jk}(n)\right) + \\ & i\hbar \sum_{j,k,l,m,n} g_{jk}^{cr}\left(\mathbf{a}_j^{\dagger}(l)\mathbf{a}_j(m)\mathbf{cr}_{jk}^{\dagger}(n) - \mathbf{a}_j^{\dagger}(m)\mathbf{a}_j(l)\mathbf{cr}_{jk}(n)\right) + \\ & i\hbar \sum_{j,k,l,m,n} g_{jk}^{mt}\left(\mathbf{a}_j^{\dagger}(l)\mathbf{a}_j(m)\mathbf{mt}_{jk}^{\dagger}(n) - \mathbf{a}_j^{\dagger}(m)\mathbf{a}_j(l)\mathbf{mt}_{jk}(n)\right).\end{aligned} \tag{1.5}$$

The bosonic creation and annihilation operators generate or destroy a bosonic field by the interaction with fermionic fields. The indices have the following meaning: j is detached to fermionic units $\mathbf{a}_j^{\dagger}, \mathbf{a}_j$. The index k describes the message type (mode) of a bosonic field. The bosonic operators mediate the "forces" between the fermionic units and stand for the message exchange between them, where the more specific message type (bosonic states) are notated by n. The letters l and m describe internal states (e.g. exited state) of fermionic units. Bosonic operators that represent the interaction e.g. with the environment, or more general sensor data, are defined by $\mathbf{ms}_{jk}^{\dagger}(n)$ and $\mathbf{ms}_{jk}(n)$. The operators $\mathbf{cr}_{jk}$ and $\mathbf{cr}_{jk}^{\dagger}$; respectively $\mathbf{mt}_{jk}$ and $\mathbf{mt}_{jk}^{\dagger}$ define the direct internal interactions, e.g. the command for to genes to mutate.

The creation and annihilation operators are Hermetian adjoint, therefore complex coefficients should also be complex conjugated, only if they are real the coefficients for both conjugated operators are equal. Here we make the assumption that all coefficients are real.

The set of equations of motion for all three creation operators is given by (let n be fixed, and $\hbar = \frac{h}{2\pi}$ is Planck's quantum of action):

$$\frac{d}{dt}\mathbf{ms}^{\dagger}_{jk} = \frac{i}{\hbar}[\mathbf{H}_I, \mathbf{ms}^{\dagger}_{jk}] = -\frac{i}{\hbar}\frac{\partial \mathbf{H}_I}{\partial \mathbf{ms}_{jk}} = \sum_{l,m} g^{ms}_{jk}(\mathbf{a}^{\dagger}_j(m)\mathbf{a}_j(l)) = \sum_{l,m} g^{ms}_{jk}\boldsymbol{\alpha}^{\dagger}_j(m,l) \quad (1.6)$$

$$\frac{d}{dt}\mathbf{cr}^{\dagger}_{jk} = \frac{i}{\hbar}[\mathbf{H}_I, \mathbf{cr}^{\dagger}_{jk}] = -\frac{i}{\hbar}\frac{\partial \mathbf{H}_I}{\partial \mathbf{cr}_{jk}} = \sum_{l,m} g^{cr}_{jk}(\mathbf{a}^{\dagger}_j(m)\mathbf{a}_j(l)) = \sum_{l,m} g^{cr}_{jk}\boldsymbol{\alpha}^{\dagger}_j(m,l) \quad (1.7)$$

$$\frac{d}{dt}\mathbf{mt}^{\dagger}_{jk} = \frac{i}{\hbar}[\mathbf{H}_I, \mathbf{mt}^{\dagger}_{jk}] = -\frac{i}{\hbar}\frac{\partial \mathbf{H}_I}{\partial \mathbf{mt}_{jk}} = \sum_{l,m} g^{mt}_{jk}(\mathbf{a}^{\dagger}_j(m)\mathbf{a}_j(l)) = \sum_{l,m} g^{mt}_{jk}\boldsymbol{\alpha}^{\dagger}_j(m,l) \quad (1.8)$$

Here we abbreviated the formulas by the use of the state flip operator $\boldsymbol{\alpha}^{\dagger}_j(m,l) = \mathbf{a}^{\dagger}_j(m)\ \mathbf{a}_j(l)$. For further use we also introduce here in addition the Hermitean conjugated flip operator $\boldsymbol{\alpha}_j(m,l) = \mathbf{a}^{\dagger}_j(l)\ \mathbf{a}_j(m)$.

The second derivative of $\mathbf{ms}^{\dagger}_{jk}$ has the form (n is again fixed):

$$\frac{d^2}{dt^2}\mathbf{ms}^{\dagger}_{jk} = \sum_{l,m} g^{ms}_{jk}\frac{d}{dt}\boldsymbol{\alpha}^{\dagger}_j(m,l). \quad (1.9)$$

The derivation of the term $\boldsymbol{\alpha}^{\dagger}_j(m,l)$ is calculated by

$$\begin{aligned}\frac{d}{dt}\boldsymbol{\alpha}^{\dagger}_j(m,l) &= \frac{i}{\hbar}\left[\mathbf{H}_I, \boldsymbol{\alpha}_j(m,l)\right] \\ &= \sum_k g^{ms}_{jk}\mathbf{ms}^{\dagger}_{jk}\boldsymbol{\sigma}_j(m,l) + \sum_{k,l,m} g^{cr}_{jk}\mathbf{cr}^{\dagger}_j\boldsymbol{\sigma}_j(m,l) + \sum_{k,l,m} g^{mt}_{jk}\mathbf{mt}^{\dagger}_{jk}\boldsymbol{\sigma}_j(m,l).\end{aligned} \quad (1.10)$$

We used the following abbreviation for the self-conjugated operator

$$\boldsymbol{\sigma}_j(m,l) = \mathbf{a}^{\dagger}_j(m)\mathbf{a}_j(m) - \mathbf{a}^{\dagger}_j(l)\mathbf{a}_j(l) = \boldsymbol{\sigma}^{\dagger}_j(m,l). \quad (1.11)$$

In laser terminology $\boldsymbol{\sigma}_j$ describes the activity above the laser threshold (saturated inversion). In our case this denotes a time dependent activity of agent j where it produces messages (bosonic fields). This message generation can be quantified by the number of messages that are are transmitted (corresponds to the number of photons). We use the notation $\boldsymbol{\sigma}^0_j$ if agent j is in an equilibrium state where it generates no messages (typically activities below the laser threshold prescribing incoherent interactions with the surroundings).

The resulting formulas of the second derivations of the three creation operators are:

$$\frac{d^2}{dt^2}\mathbf{ms}^{\dagger}_{jk} = \sum_{l,m} g^{ms2}_{jk}\mathbf{ms}^{\dagger}_{jk}\boldsymbol{\sigma}(m,l). \quad (1.12)$$

$$\frac{d^2}{dt^2}\mathbf{cr}^{\dagger}_{jk} = \sum_{l,m} g^{cr2}_{jk}\mathbf{cr}^{\dagger}_{jk}\boldsymbol{\sigma}(m,l). \quad (1.13)$$

$$\frac{d^2}{dt^2}\mathbf{mt}_{jk}^{\dagger} = \sum_{l,m} g_{jk}^{mt2}\mathbf{mt}_{jk}^{\dagger}\boldsymbol{\sigma}(m,l). \tag{1.14}$$

The total sum of the "acceleration" of the creation operator $\mathbf{ms}_{jk}^{\dagger}$ is given by

$$\sum_{j,k}\frac{d^2}{dt^2}\mathbf{ms}_{jk}^{\dagger} = \sum_{j,k,l,m} g_{jk}^{ms2}\mathbf{ms}_{jk}^{\dagger}\boldsymbol{\sigma}(m,l). \tag{1.15}$$

Analogue formulas also hold for the two remaining creation operators. The influences of the three creation operators $\mathbf{ms}_{jk}^{\dagger}, \mathbf{cr}_{jk}^{\dagger}, \mathbf{mt}_{jk}^{\dagger}$ on the total interaction Hamiltonian are determined, beside the squared coupling constants $(g_{jk}^{ms})^2$, $(g_{jk}^{cr})^2$, $(g_{jk}^{mt})^2$, (all $\ll 1$), primarily by the rate of $\boldsymbol{\sigma}_j(m,l)$. Thus we calculate

$$\begin{aligned}\frac{d}{dt}\boldsymbol{\sigma}_j(m,l) = \frac{i}{\hbar}\left[\mathbf{H}_I, \boldsymbol{\sigma}_j^{\dagger}(m,l)\right] = -\Big(\sum_k g_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} + \sum_k g_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} + \\ \sum_k g_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}\Big)\boldsymbol{\alpha}_j(m,l) - \Big(\sum_k g_{jk}^{ms}\mathbf{ms}_{jk} + \sum_k g_{jk}^{cr}\mathbf{cr}_{jk} + \sum_k g_{jk}^{mt}\mathbf{mt}_{jk}\Big)\boldsymbol{\alpha}_j^{\dagger}(m,l).\end{aligned} \tag{1.16}$$

The commutator rules and the anti-commutator formulas must be valid for all time. This will be violated if we introduce only damping constants κ_{jk} (dissipation) for the creation operators, damping constants γ_{jk} for the flip operators, and a relaxation time T_j that defines the time span where $\boldsymbol{\sigma}_j$ recovers to the stationary value $\boldsymbol{\sigma}_j^0$. The validity of the commutator and anti-commutator formulas will be restored if we introduce external fluctuating forces F_{fluc}, which are mandatory that the commutators respectively the anti-commutators are exactly fulfilled all time (Haken, 1985). They represent the interaction with the environment. We will model these forces not explicitly in the interaction Hamiltonian (the formalism is simpler since we have not to calculate the expectation values of the stochastic forces) but implicitly with the aid of different statistics representing additional, stochastic interactions between the fields and the environmental restrictions (e.g. Poisson distribution for the mean number of messages).

If we include damping constants κ_{jk} into the formulas (1.6), (1.7) and (1.8) then the final set of equations is transferred to:

$$\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = \sum_{l,m} g_{jk}^{ms}\boldsymbol{\alpha}_j^{\dagger}(m,l) - \kappa_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}. \tag{1.17}$$

$$\frac{d}{dt}\mathbf{cr}_{jk}^{\dagger} = \sum_{l,m} g_{jk}^{cr}\boldsymbol{\alpha}_j^{\dagger}(m,l) - \kappa_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}. \tag{1.18}$$

$$\frac{d}{dt}\mathbf{mt}_{jk}^{\dagger} = \sum_{l,m} g_{jk}^{mt}\boldsymbol{\alpha}_j^{\dagger}(m,l) - \kappa_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}. \tag{1.19}$$

The temporal derivation of the flip operator is transferred to the following form if the damping constants γ_{jk} are included into formula (1.10):

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\alpha}_{\mathrm{j}}^{\dagger}(\mathrm{m},\mathrm{l}) = &\sum_{k}\left(\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) - \gamma_{\mathrm{jk}}^{\mathrm{ms}}\boldsymbol{\alpha}_{\mathrm{jk}}^{\dagger}(\mathrm{m},\mathrm{l})\right) + \\ &\sum_{k}\left(\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) - \gamma_{\mathrm{jk}}^{\mathrm{cr}}\boldsymbol{\alpha}_{\mathrm{jk}}^{\dagger}(\mathrm{m},\mathrm{l})\right) + \\ &\sum_{k}\left(\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) - \gamma_{\mathrm{jk}}^{\mathrm{mt}}\boldsymbol{\alpha}_{\mathrm{jk}}^{\dagger}(\mathrm{m},\mathrm{l})\right).\end{aligned} \tag{1.20}$$

In the last step we include the relaxation time $\mathrm{T_j(m,l)}$ in the formula (1.16). This step changes this equation into the following formula:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) = &-\left(\sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} + \sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} + \sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\right)\boldsymbol{\alpha}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) - \\ &\left(\sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} + \sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} + \sum_{k}\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\right)\boldsymbol{\alpha}_{\mathrm{j}}^{\dagger}(\mathrm{m},\mathrm{l}) + \\ &\frac{1}{\mathrm{T_j(m,l)}}\left(\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l}) - \boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\right).\end{aligned} \tag{1.21}$$

All in all we get a set of coupled non-linear differential equations. They can be solved if we assume that the damping constants and the relaxation time are different in the orders of magnitudes in the sense of adiabatic elimination (Haken, 1970). For example, for the "**ms**-constants" this estimation looks like

$$\gamma_{\mathrm{jk}}^{\mathrm{ms}} > \frac{1}{\mathrm{T_j(m,l)}} > \kappa_{\mathrm{jk}}^{\mathrm{ms}}. \tag{1.22}$$

Under the above mentioned assumption, that the damping constant γ_{jk} dominates the other parameters, we can set (if these inequalities are not true then a synchronised, self-organising process cannot start) $\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\alpha}_{\mathrm{j}}^{\dagger}(\mathrm{m},\mathrm{l}) = 0$ in (1.20) and solve this equation:

$$\boldsymbol{\alpha}_{\mathrm{j}}^{\dagger}(\mathrm{m},\mathrm{l}) = \sum_{\mathrm{k}}\left(\frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}}{\gamma_{\mathrm{jk}}^{\mathrm{ms}}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} + \sum_{\mathrm{k}}\frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}}{\gamma_{\mathrm{jk}}^{\mathrm{cr}}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} + \sum_{\mathrm{k}}\frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}}{\gamma_{\mathrm{jk}}^{\mathrm{mt}}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\right)\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}). \tag{1.23}$$

This result will then be inserted into (1.21) yielding the result:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) = -2\Big(&\sum_{\mathrm{k}}\frac{(\mathrm{g}_{\mathrm{jk'}}^{\mathrm{ms}})^2}{\gamma_{\mathrm{jk'}}^{\mathrm{ms}}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}} + \sum_{\mathrm{k}}\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}})^2}{\gamma_{\mathrm{jk}}^{\mathrm{cr}}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\mathbf{cr}_{\mathrm{jk}} + \\ &\sum_{\mathrm{k}}\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}})^2}{\gamma_{\mathrm{jk}}^{\mathrm{mt}}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\mathbf{mt}_{\mathrm{jk}}\Big)\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) + \frac{1}{\mathrm{T_j(m,l)}}\left(\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l}) - \boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\right).\end{aligned} \tag{1.24}$$

In the next step we set $\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l}) = 0$, solve the resulting equation with respect to $\boldsymbol{\sigma}_{\mathrm{j}}(\mathrm{m},\mathrm{l})$ and insert the result into (1.21). These steps generate the following approximate solution:

$$\begin{aligned}\boldsymbol{\alpha}_j^{\dagger}(m,l) = \boldsymbol{\sigma}_j^0(m,l)\sum_k \frac{g_{jk}^{ms}}{\gamma_{jk}^{ms}}\mathbf{ms}_{jk}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jk}^{ms})^2}{\gamma_{jk}^{ms}}\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}\right)+\\ \boldsymbol{\sigma}_j^0(m,l)\sum_k \frac{g_{jk}^{cr}}{\gamma_{jk}^{cr}}\mathbf{cr}_{jk}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jk}^{cr})^2}{\gamma_{jk}^{cr}}\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}\right)+\\ \boldsymbol{\sigma}_j^0(m,l)\sum_k \frac{g_{jk}^{mt}}{\gamma_{jk}^{mt}}\mathbf{mt}_{jk}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jk}^{mt})^2}{\gamma_{jk}^{mt}}\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}\right).\end{aligned} \tag{1.25}$$

In the last step we insert (1.24) e.g. into (1.16). The result for $\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger}$ reads then:

$$\begin{aligned}\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} &= sum_{l,m} g_{jk}^{ms}\boldsymbol{\alpha}_j^{\dagger}(m,l) - \kappa_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}\\ &= \sum_{l,m} g_{jk}^{ms}\boldsymbol{\sigma}_j^0(m,l)\Bigg(\sum_q \frac{g_{jq}^{ms}}{\gamma_{jq}^{ms}}\mathbf{ms}_{jq}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jq}^{ms})^2}{\gamma_{jq}^{ms}}\mathbf{ms}_{jq}^{\dagger}\mathbf{ms}_{jq}\right)+\\ &\quad \sum_q \frac{g_{jq}^{cr}}{\gamma_{jq}^{cr}}\mathbf{cr}_{jq}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jq}^{cr})^2}{\gamma_{jq}^{cr}}\mathbf{cr}_{jq}^{\dagger}\mathbf{cr}_{jq}\right)+\\ &\quad \sum_q \frac{g_{jq}^{mt}}{\gamma_{jq}^{mt}}\mathbf{mt}_{jq}^{\dagger}\left(1-2T_j(m,l)\frac{(g_{jq}^{mt})^2}{\gamma_{jq}^{mt}}\mathbf{mt}_{jq}^{\dagger}\mathbf{mt}_{jq}\right)\Bigg) - \kappa_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}.\end{aligned} \tag{1.26}$$

Finally, we rearrange this equation:

$$\begin{aligned}\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = &\sum_{l,m}\Bigg(g_{jk}^{ms}\boldsymbol{\sigma}_j^0(m,l)\Bigg(\sum_q \frac{g_{jq}^{ms}}{\gamma_{jq}^{ms}}\mathbf{ms}_{jq}^{\dagger}+\sum_q \frac{g_{jq}^{cr}}{\gamma_{jq}^{cr}}\mathbf{cr}_{jq}^{\dagger}+\sum_q \frac{g_{jq}^{mt}}{\gamma_{jq}^{mt}}\mathbf{mt}_{jq}^{\dagger}\Bigg)-\\ &\kappa_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\Bigg(\sum_q \frac{(g_{jq}^{ms})^3}{(\gamma_{jq}^{ms})^2}\mathbf{ms}_{jq}^{\dagger}\left(\mathbf{ms}_{jq}^{\dagger}\mathbf{ms}_{jq}\right)-\\ &\sum_q \frac{(g_{jq}^{cr})^3}{(\gamma_{jq}^{cr})^2}\mathbf{cr}_{jq}^{\dagger}\left(\mathbf{cr}_{jq}^{\dagger}\mathbf{cr}_{jq}\right) - \sum_q \frac{(g_{jq}^{mt})^3}{(\gamma_{jq}^{mt})^2}\mathbf{mt}_{jq}^{\dagger}\left(\mathbf{mt}_{jq}^{\dagger}\mathbf{mt}_{jq}\right)\Bigg)\Bigg).\end{aligned} \tag{1.27}$$

To generate the first derivative of the total bosonic field $\mathbf{ms}^{\dagger}$ (it summaries all created **ms**-messages) we must sum up over all indices j and k:

$$\begin{aligned}\frac{d}{dt}\mathbf{ms}^{\dagger} = &\sum_{l,m,j,k}\Bigg(g_{jk}^{ms}\boldsymbol{\sigma}_j^0(m,l)\Bigg(\sum_q \frac{g_{jq}^{ms}}{\gamma_{jq}^{ms}}\mathbf{ms}_{jq}^{\dagger}+\sum_q \frac{g_{jq}^{cr}}{\gamma_{jq}^{cr}}\mathbf{cr}_{jq}^{\dagger}+\sum_q \frac{g_{jq}^{mt}}{\gamma_{jq}^{mt}}\mathbf{mt}_{jq}^{\dagger}\Bigg)-\\ &\kappa_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\Bigg(\sum_q \frac{(g_{jq}^{ms})^3}{(\gamma_{jq}^{ms})^2}\mathbf{ms}_{jq}^{\dagger}\left(\mathbf{ms}_{jq}^{\dagger}\mathbf{ms}_{jq}\right)-\\ &\sum_q \frac{(g_{jq}^{cr})^3}{(\gamma_{jq}^{cr})^2}\mathbf{cr}_{jq}^{\dagger}\left(\mathbf{cr}_{jq}^{\dagger}\mathbf{cr}_{jq}\right) - \sum_q \frac{(g_{jq}^{mt})^3}{(\gamma_{jq}^{mt})^2}\mathbf{mt}_{jq}^{\dagger}\left(\mathbf{mt}_{jq}^{\dagger}\mathbf{mt}_{jq}\right)\Bigg)\Bigg).\end{aligned} \tag{1.28}$$

Analogous equations are true for $\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{cr}^{\dagger}$ and $\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{mt}^{\dagger}$. We have now defined all formulas and can begin to solve them. The first step in this direction will be the calculation of the solution of (1.26). Afterwards the analogous equations for the temporal derivations of $\mathbf{cr}_{\mathrm{jk}}^{\dagger}$ and $\mathbf{mt}_{\mathrm{jk}}^{\dagger}$ must be solved.

We perform this procedure step by step, and we start with the simplification of Eq. (1.26). This yields the formula:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} = \boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\left(\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}})^{2}}{\gamma_{\mathrm{jk}}^{\mathrm{ms}}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} - \kappa_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} + \frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}}{\gamma_{\mathrm{jk}}^{\mathrm{cr}}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} + \frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}}{\gamma_{\mathrm{jk}}^{\mathrm{mt}}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\right) - \\ 2\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\mathrm{T}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\left(\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{ms}})^{2}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\left(\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jq}}\right) + \frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{cr}})^{2}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\left(\mathbf{cr}_{\mathrm{jk}}^{\dagger}\mathbf{cr}_{\mathrm{jk}}\right) + \right. \\ \left.\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{mt}})^{2}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\left(\mathbf{mt}_{\mathrm{jk}}^{\dagger}\mathbf{mt}_{\mathrm{jk}}\right)\right).\end{aligned} \tag{1.29}$$

In short:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} = \mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} - \mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\left(\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}\right) + \mathrm{c}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} - \mathrm{d}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\left(\mathbf{cr}_{\mathrm{jk}}^{\dagger}\mathbf{cr}_{\mathrm{jk}}\right) + \\ \mathrm{c}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger} - \mathrm{d}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\left(\mathbf{mt}_{\mathrm{jk}}^{\dagger}\mathbf{mt}_{\mathrm{jk}}\right).\end{aligned} \tag{1.30}$$

We introduced the following abbreviations:

$$\begin{aligned}&\mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}} = \boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}})^{2}}{\gamma_{\mathrm{jk}}^{\mathrm{ms}}} - \kappa_{\mathrm{jk}}^{\mathrm{ms}}\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l}), \quad \mathrm{c}_{\mathrm{jk}}^{\mathrm{cr}} = \boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}}}{\gamma_{\mathrm{jk}}^{\mathrm{cr}}}, \\ &\mathrm{c}_{\mathrm{jk}}^{\mathrm{mt}} = \boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\frac{\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}}\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}}}{\gamma_{\mathrm{jk}}^{\mathrm{mt}}}; \\ &\mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}} = 2\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\mathrm{T}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{ms}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{ms}})^{2}}, \quad \mathrm{d}_{\mathrm{jk}}^{\mathrm{cr}} = 2\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\mathrm{T}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{cr}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{cr}})^{2}}, \\ &\mathrm{d}_{\mathrm{jk}}^{\mathrm{mt}} = 2\boldsymbol{\sigma}_{\mathrm{j}}^{0}(\mathrm{m},\mathrm{l})\mathrm{T}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\frac{(\mathrm{g}_{\mathrm{jk}}^{\mathrm{mt}})^{3}}{(\gamma_{\mathrm{jk}}^{\mathrm{mt}})^{2}}.\end{aligned}$$

In analogy to (1.30) we rewrite the equation for $\mathbf{cr}_{\mathrm{jk}}^{\dagger}$:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} = \mathrm{e}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger} - \mathrm{f}_{\mathrm{jk}}^{\mathrm{cr}}\mathbf{cr}_{\mathrm{jk}}^{\dagger}\left(\mathbf{cr}_{\mathrm{jk}}^{\dagger}\mathbf{cr}_{\mathrm{jk}}\right) + \mathrm{e}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} - \mathrm{f}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\left(\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}\right) + \\ \mathrm{e}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger} - \mathrm{f}_{\mathrm{jk}}^{\mathrm{mt}}\mathbf{mt}_{\mathrm{jk}}^{\dagger}\left(\mathbf{mt}_{\mathrm{jk}}^{\dagger}\mathbf{mt}_{\mathrm{jk}}\right).\end{aligned} \tag{1.31}$$

Similar abbreviations are:

$$e_{jk}^{ms} = \boldsymbol{\sigma}_j^0(m,l)\frac{(g_{jk}^{cr})^2}{\gamma_{jk}^{cr}} - \kappa_{jk}^{cr}\boldsymbol{\sigma}_j^0(m,l), \quad e_{jk}^{ms} = \boldsymbol{\sigma}_j^0(m,l)\frac{g_{jk}^{cr}g_{jk}^{ms}}{\gamma_{jk}^{ms}},$$

$$e_{jk}^{mt} = \boldsymbol{\sigma}_j^0(m,l)\frac{g_{jk}^{cr}g_{jk}^{mt}}{\gamma_{jk}^{mt}};$$

$$f_{jk}^{cr} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{cr})^3}{(\gamma_{jk}^{cr})^2}, \quad f_{jk}^{ms} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{ms})^3}{(\gamma_{jk}^{ms})^2},$$

$$f_{jk}^{mt} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{mt})^3}{(\gamma_{jk}^{mt})^2}.$$

In the end the analogous equation for $\mathbf{mt}_{jk}^{\dagger}$ looks like:

$$\frac{d}{dt}\mathbf{mt}_{jk}^{\dagger} = s_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger} - t_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}) + s_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - t_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + s_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} - t_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}). \tag{1.32}$$

The abbreviations are again conforming to the previous one:

$$s_{jk}^{mt} = \boldsymbol{\sigma}_j^0(m,l)\frac{(g_{jk}^{mt})^2}{\gamma_{jk}^{mt}} - \kappa_{jk}^{mt}\boldsymbol{\sigma}_j^0(m,l), \quad s_{jk}^{ms} = \boldsymbol{\sigma}_j^0(m,l)\frac{g_{jk}^{mt}g_{jk}^{ms}}{\gamma_{jk}^{mt}},$$

$$s_{jk}^{cr} = \boldsymbol{\sigma}_j^0(m,l)\frac{g_{jk}^{cr}g_{jk}^{mt}}{\gamma_{jk}^{mt}};$$

$$t_{jk}^{mt} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{mt})^3}{(\gamma_{jk}^{mt})^2}, \quad t_{jk}^{ms} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{ms})^3}{(\gamma_{jk}^{ms})^2},$$

$$t_{jk}^{cr} = 2\boldsymbol{\sigma}_j^0(m,l)T_j(m,l)\frac{(g_{jk}^{cr})^3}{(\gamma_{jk}^{cr})^2}.$$

Eqs. (1.30) - (1.32) and their three Hermitean conjugates are the differential equations we have to solve. All these three differential equations are composed of three symmetrical parts. Nevertheless we cannot solve each symmetrical term by itself and then add all three together because the full equations are non-linear. In addition the actions of these three parts e.g. of (1.30) are different. The first term models the own contribution (Eigenanteil) $\mathbf{ms}_{jk}^{\dagger}$, the remaining two parts describe the coupling of $\mathbf{ms}_{jk}^{\dagger}$ to the creation operators $\mathbf{cr}_{jk}^{\dagger}$ and $\mathbf{mt}_{jk}^{\dagger}$. The mutual interactions of all three bosonic fields denotes a strong dependency between them, meaning e.g. that the **ms**-field tries to control the **mt**-field, or by interference effects new combinations of frequencies occur, or the frequency of two fields is changed. In technical terms this means that there exist frequency lockings and phase lockings that are generated by the interactions of these three modes.

Transferred to symbiotic organisms these effects can stabilise e.g. the connections of different cells since there is a strong regulative, dynamic regime that controls the internal communications by **mt**-fields and **cr**-fields by the **ms**-field. But it is also possible (depending from the settings of the various parameters) that the **mt**-field (mutation) dominates the other two fields and the organism destabilises. The kind of interactions of the three bosonic fields define whether the robot-cells can cooperate together. In a synergetic view all this depends on the values of the control parameters e.g. c_{jk}^{ms} of the **ms**-field and the dynamics of the order field $\mathbf{ms}_{jk}^{\dagger}$, (see next sub-paragraph).

The additional principal tools beside the above performed synergetic based calculations (e.g. control parameter, order parameter field (Haken, 1977) are the methods of dynamical systems (Ghrist *et al.*, 1997) and the techniques of differential geometry (Kobayashi & Nomizu, 1996), (Guckenheimer & Holmes, 1983). We start the descriptions of the solutions by the view of dynamical systems. Hereby, as a standard procedure, the equilibrium of a dynamical system will be studied by the behavior of an invariant set Λ of the vector field $f(\Lambda) = \Lambda$. Such a set Λ can be one-point set (fixed-point) or a manifold (e.g. circle of fixed points, see Fig. 1.13(b)). In any case, we analyse the behavior of a dynamical system primarily in equilibrium states in order to describe the stability of the system invariant sets.

1.2.2 Individual Contributions of the Eigenanteile

In the first step the equations of the bosonic creation operators $\mathbf{ms}_{jk}^{\dagger}$ and $\mathbf{ms}_{jk}$ are solved. Since we are interested in the results of the measurements of these two operators we calculate the expectation values of them. It is also usual to analyse the flow of the field. We do this in the second sub-chapter. In the third sub-chapter an external, periodic force affects the expectation values of the two **ms**-operators, where such a field e.g. come from a periodic electro-magnetic wave. If we go deeper into the physics such a wave can polarise the robot cells. The calculated phase portraits demonstrate the resulting periodicity in this representation space.

1.2.2.1 Uncoupled and Unforced Contributions of the Operators and Their Expectation Values

In order to get an impression of the first incomplete results of (1.30) we study the behavior of the Eigenanteile of $\mathbf{ms}_{jk}^{\dagger}$ (Eigenanteil-results for $\mathbf{cr}_{jk}^{\dagger}$ and $\mathbf{mt}_{jk}^{\dagger}$ are similar). Such Eigenanteile represent in a self consistent way how the "**ms**-field" interact with itself via the interaction with a fermionic agent. In more detail; we separate from (1.30) the partial formula

$$\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = c_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}). \tag{1.33}$$

Supplementary we write down the conjugate equation

$$\frac{d}{dt}\mathbf{ms}_{jk} = c_{jk}^{ms}\mathbf{ms}_{jk} - d_{jk}^{ms}\mathbf{ms}_{jk}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}). \tag{1.34}$$

These two formulas are the first elementary building blocks of our approach. The operator $\mathbf{ms}_{jk}^{\dagger}$ plays the role of an order parameter, c_{jk}^{ms} defines the control parameter, and formula (1.25) expresses the slaving principle.

The creation respectively annihilation operators are complex operators. The appropriate method of solutions of the two adjoint operator equations can be done in the space of eigenfunctions Φ_α of the annihilation operator $\mathbf{ms}_{jk}$, where

$$\mathbf{ms}_{jk}\Phi_\alpha = \alpha\Phi_\alpha, \alpha \in \mathbb{C}. \tag{1.35}$$

We take the expectation value

$$\left\langle \Phi_\alpha \left| \mathbf{ms}_{jk} \right| \Phi_\alpha \right\rangle = \alpha, \tag{1.36}$$

with the normalisation $\langle \Phi_\alpha \mid \Phi_\alpha \rangle = 1$. For the adjoint operator holds $\langle \phi_\alpha \mid \mathbf{ms}_{jk}^{\dagger} \mid \phi_\alpha \rangle = \alpha^*$. Below we will use the short notations $\langle \mathbf{ms}_{jk} \rangle$ and $\langle \mathbf{ms}_{jk}^{\dagger} \rangle$ for these two expectation values.

For clarity we drop the indices in the two operator equations (1.33) and (1.34) and get the equivalent equations for the expectation values

$$\frac{d\alpha}{dt} = c\alpha - d|\alpha|^2\alpha, \ \frac{d\alpha^*}{dt} = c\alpha^* - d|\alpha|^2\alpha^*. \tag{1.37}$$

If we split the expectation values of the **ms** operator into the real part $u = \mathrm{Re}\left\langle \mathbf{ms}_{jk} \right\rangle$ and imaginary part $v = \mathrm{Im}\left\langle \mathbf{ms}_{jk} \right\rangle$ then we obtain from both equations the following differential equations for u and v:

$$\frac{d}{dt}u = c_{jk}^{ms}u - d_{jk}^{ms}u(u^2+v^2) = c_{jk}^{ms}u - d_{jk}^{ms}ur^2. \tag{1.38}$$

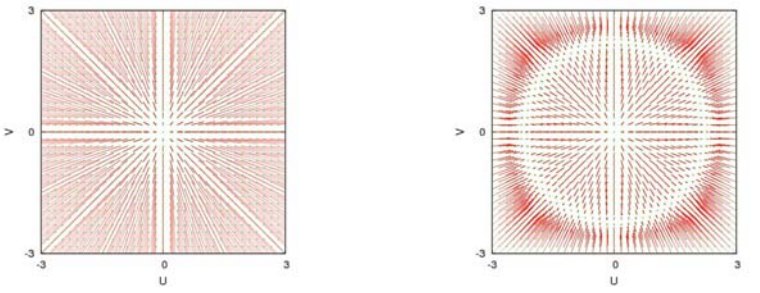
$$\frac{d}{dt}v = c_{jk}^{ms}v - d_{jk}^{ms}v(u^2+v^2) = c_{jk}^{ms}v - d_{jk}^{ms}vr^2. \tag{1.39}$$

These equations are symmetric in (u,v). The single unstable fixed point is the origin $O = (0,0)$, if $c_{jk}^{ms} > 0$ and $d_{jk}^{ms} < 0$. It represents a saddle point that is physically not relevant. Fig. 1.13(a) demonstrates this fact by a phase portrait of u and v, and it shows the influence of the unstable fixed point in the origin O[1].

The alternative calculations with $\langle \mathbf{ms}_{jk} \rangle = \alpha = r(t)e^{i\varphi(t)}$ and $\langle \mathbf{ms}_{jk}^{\dagger} \rangle = \alpha^*$ deliver in a more elegant way the two equivalent differential equations if both coefficients are real:

$$\frac{d}{dt}r = c_{jk}^{ms}r - d_{jk}^{ms}r^3, \text{and} \frac{d}{dt}\varphi = 0. \tag{1.40}$$

[1] We express our gratitude to Dr. Victor Avrutin for his uncomplaining and substantial calculation support of this work.



(a) (b)

Fig. 1.13 Phase portraits of u and v. **(a)** There is an unstable fixed point in the origin origin O = (0,0). The values are $c_{jk}^{ms} = 1.5$ and $d_{jk}^{ms} = -0.3$; **(b)** The values are $c_{jk}^{ms} = 1.5$ and $d_{jk}^{ms} = 0.3$.

Fig. 1.13(b) demonstrates the effect if d_{jk}^{ms} gets positive; there arises a circle of attractive fixed points with radius $r_0 = \sqrt{\frac{c_{jk}^{ms}}{d_{jk}^{ms}}}$. For physical relevant "laser actions" we have to use a positive d_{jk}^{ms} in order to get a stable amplitude.

1.2.2.2 Flow of the Continuous and Uncoupled Eigenanteile

Here we specify the explicit solutions of the two Eqs. 1.33 and 1.34, and calculate the accordant flow in terms of dynamical systems. A dynamical system $\frac{dx}{dt} = f(x)$, is a system of differential equations, where x = x(t)$\in \mathrm{R}^n$ and f is a vector field that generates a continuous flow $\Phi_t(x) = \Phi(x,t)$, that also can be considered as a one parameter group under the operation of composition $(\Phi_s \circ \Phi_t(x)) = \Phi_{s+t}(x)$ that satisfies the equation

$$\frac{d}{dt}\Phi(x,t)|_{t=\tau} = f(\Phi(x,\tau)), \forall\ x \text{ and } \tau \in I = (a,b) \subseteq \mathbb{R}^n. \tag{1.41}$$

We repeat again the continuous Eigenanteil

$$\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = c_{j,k}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{j,k}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}). \tag{1.42}$$

If we drop all indices of (1.37) then we get the solutions:

$$r(t) = \sqrt{c}\Big((\frac{c}{r_0^2} - d)e^{-2ct} + d\Big)^{-1/2},\ \varphi = \varphi_0. \tag{1.43}$$

The fixed point is, as we already know, defined by $r_0 = \sqrt{\frac{c}{d}}$; its stability is calculated by

$$\frac{dr}{dr_0}|_{\sqrt{\frac{c}{d}}} = de^{-2ct}. \tag{1.44}$$

This fixed point r_0 is stable, if c and d are both positive, reflecting the already known circular orbit of radius r_0. In the limit $t \to \infty$, $r(t)$ converges to r_0. We get an equivalent result if we consider the derivative D of the Poincaré map

$$DP(r_0) = \frac{dP}{dr_0}|_{\sqrt{\frac{c}{d}}} = d^{3/2}e^{-4\pi c} < 1. \tag{1.45}$$

The continuous flow operator is given by:

$$\Phi_t(r_0, \varphi_0) = \left(\sqrt{c}\left(\left(\frac{c}{r_0^2} - d\right)e^{-2ct} + d\right)^{-1/2}, \varphi_0\right). \tag{1.46}$$

Fig. 1.14 demonstrates the flow (one-parameter group) for $r(t)$ and different constant φ values.

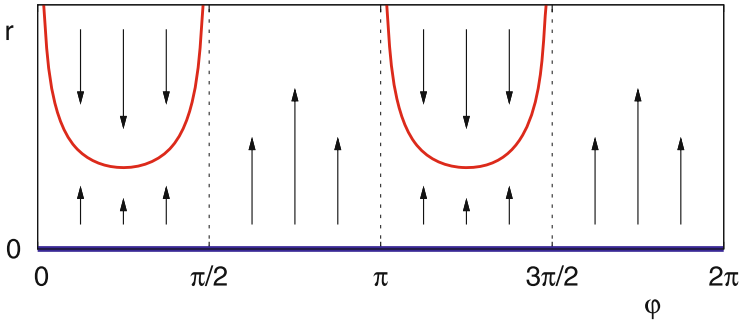


Fig. 1.14 Schematic representation of flow (phase curves) in the (r, φ) plane with $r(\varphi) = \sqrt{\frac{c}{d}}$. There is a pattern repetition for $0 < \varphi < \frac{\pi}{2}$, $\pi < \varphi < \frac{3\pi}{2}$, etc. Red curves are attractive; the blue line is repulsive (unstable fixed point in the origin).

1.2.2.3 Uncoupled and External Forced Oscillations of the Operators and Their Expectation Values

In the next step we apply an external periodic force $A\sin(\omega t)$ with real amplitude A:

$$\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = c_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + A\sin(\omega t). \tag{1.47}$$

$$\frac{d}{dt}\mathbf{ms}_{jk} = c_{jk}^{ms}\mathbf{ms}_{jk} - d_{jk}^{ms}\mathbf{ms}_{jk}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + A\sin(\omega t). \tag{1.48}$$

In the short notation – with coefficient dropping – these two equations read as $(\alpha = r(t)e^{i\varphi(t)})$

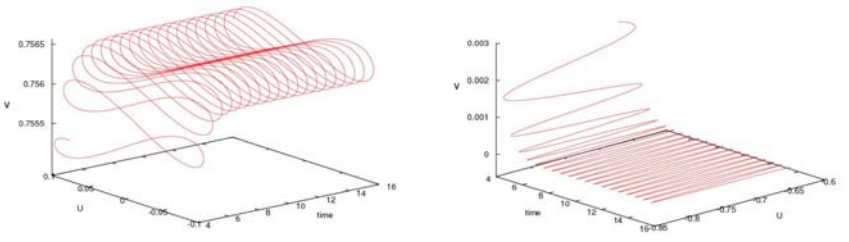
$$\frac{\mathrm{d}}{\mathrm{dt}}\alpha = \mathrm{c}\alpha - \mathrm{d}|\alpha|^2\alpha + \mathrm{A}\sin(\omega \mathrm{t}),\ \frac{\mathrm{d}}{\mathrm{dt}}\alpha^* = \mathrm{c}\alpha^* - \mathrm{d}|\alpha|^2\alpha^* + \mathrm{A}\sin(\omega \mathrm{t}). \tag{1.49}$$

If we take again for granted that c and d are real then the resulting differential equations are $(\alpha = \mathrm{r}(\mathrm{t})\mathrm{e}^{\mathrm{i}\varphi(\mathrm{t})})$:

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r} = \mathrm{cr} - \mathrm{dr}^3 + \mathrm{A}\sin(\omega \mathrm{t})\cos(\varphi). \tag{1.50}$$

$$\mathrm{r}\frac{\mathrm{d}}{\mathrm{dt}}\varphi = \mathrm{A}\sin(\omega \mathrm{t})\sin(\varphi). \tag{1.51}$$

Represented in the Cartesian coordinates (u, v) two different solutions are demonstrated in Fig. 1.15. In Fig. 1.15(a) the solution has a form of an eight that twist up; in Fig. 1.15(b) the solution twists down to the plane $\mathrm{v} = 0$, where it performs sinus oscillations.



(a) (b)

Fig. 1.15 Phase portrait of $\alpha = \mathrm{u} + \mathrm{iv}$.The parameter values are: **(a)** $\mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}} = -0.75$, $\mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}} = 1.3$, A = 1, $\omega = 10$; **(b)** $\mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}} = 0.75$, $\mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}} = 1.3$, A = 1, $\omega = 10$.

1.2.3 Separate Perturbations of the Eigenanteile

In the next step we disturb the Eigenanteile e.g. for $\mathbf{ms}_{\mathrm{jk}}^{\dagger}$(similar results are obtained also for the other Eigenanteil-equations) by a mixed term $\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}$:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} &= \mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger} - \mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}(\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}} \\
\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{\mathrm{jk}} &= \mathrm{c}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}} - \mathrm{d}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}(\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}_{\mathrm{jk}}^{\mathrm{ms}}\mathbf{ms}_{\mathrm{jk}}^{\dagger}\mathbf{ms}_{\mathrm{jk}}.
\end{aligned} \tag{1.52}$$

A similar procedure as before, $\mathbf{ms}_{\mathrm{jk}} = \mathrm{r}(\mathrm{t})\mathrm{e}^{\mathrm{i}\varphi(\mathrm{t})}$, $\mathbf{ms}_{\mathrm{jk}}^{\dagger} = \mathrm{r}(\mathrm{t})\mathrm{e}^{-\mathrm{i}\varphi(\mathrm{t})}$, transfers Eq. (1.50) with dropped indices into

$$\mathrm{i}(\frac{\mathrm{d}}{\mathrm{dt}}\varphi)\mathrm{r} + \frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r} = \mathrm{cr} - \mathrm{dr}^3 + \hat{\mathrm{g}}\mathrm{r}^2\mathrm{e}^{-\mathrm{i}\varphi}. \tag{1.53}$$

The splitting of this formula into real part and imaginary part generates the result

$$\frac{d}{dt}r = cr - dr^3 + \hat{g}r^2\cos(\varphi). \tag{1.54}$$

$$\frac{d}{dt}\varphi = -\hat{g}r\sin(\varphi),\ (r \neq 0). \tag{1.55}$$

There is an unstable fixed point for $r_0 = 0$ and φ arbitrary; and a stable fixed point in

$$r_0 = \frac{1}{2d}(\hat{g} + \sqrt{\hat{g}^2 + 4cd}) \text{ and } \sin(\varphi_0) = 0,\ \hat{g} \neq 0. \tag{1.56}$$

In Cartesian coordinates Eqs. (1.52) read as:

$$\frac{d}{dt}u = c_{jk}^{ms}u - d_{jk}^{ms}u(u^2 + v^2) + \hat{g}_{jk}^{ms}u(u^2 + v^2) = c_{jk}^{ms}u - d_{jk}^{ms}ur^2 + \hat{g}_{jk}^{ms}ur^2. \tag{1.57}$$

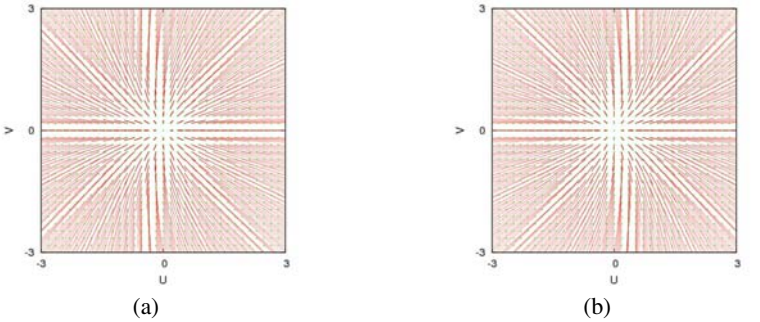
$$\frac{d}{dt}v = c_{jk}^{ms}v - d_{jk}^{ms}v(u^2 + v^2) = c_{jk}^{ms}v - d_{jk}^{ms}vr^2. \tag{1.58}$$

The nontrivial fixed points in the (u, v) space are:

$$u_0 = \pm r_0, v_0 = 0. \tag{1.59}$$

Fig. 1.16 shows the phase flow of these two equations with respect to the unstable fixed point in the origin. By comparison with Fig. 1.13(a) we see that the flow outside the u axis turns to the right side (a) or left side (b).

In the next parameter fixation c_{jk}^{ms} stays unchanged and d_{jk}^{ms} gets positive. Fig. 1.17 demonstrates the definite change of the flow behavior. In Fig. 1.17(a) the fixed point in the origin stays unstable, in addition two new fixed points $u_0 = \pm r_0$ are created. The right fixed point $u_0 = r_0$ is stable and the left fixed point $u_0 = -r_0$ gets a saddle point. In Fig. 1.17(b) the two fixed points $u_0 = \pm r_0$ change their roles.



(a) (b)

Fig. 1.16 Phase portrait. The values are: **(a)** $\hat{g}_{jk}^{ms}$ = 0.25 , $\hat{g}_{jk}^{ms}$ = 0.25; **(b)** $\hat{g}_{jk}^{ms}$ = - 0.25. In both cases c_{jk}^{ms} = 1.5 and d_{jk}^{ms}=- 0.3.

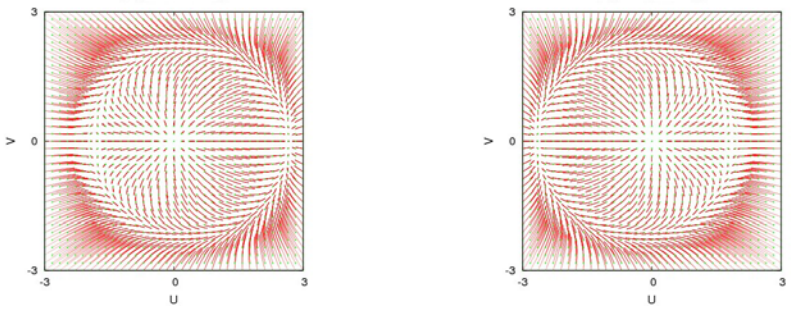


Fig. 1.17 Phase portrait. The values are: **(a)** $\hat{g}_{jk}^{ms}$= 0.25 , $\hat{g}_{jk}^{ms}$ = 0.25; **(b)** $\hat{g}_{jk}^{ms}$ = - 0.25. In both cases c_{jk}^{ms} = 1.5 and d_{jk}^{ms} = 0.3.

The interpretation of e.g. Fig. 1.17(a) in agent view can be given as follows. There are three agents ,we call them organisators, the organisator in the origin (repulsive fixed point) sends all messages (bosonic fields) to its neighbour to the left (saddle point) and to the right (attractive fixed point). The organisator to the right collects all messages that flow in asymptotically to the circle to this agent. Only the messages that start in direction of the circle from the organisator to the left stay on the circle until they reach the right organisator. All messages that are generated by other agents that are outside of the circle also reach the right organisator. All messages that are generated by the organisator in the origin and that are coming from outside of the circle build together an asymptotic flow that can be considered as a fibration (foliation) where the circle is considered as an unstable manifold W^u that defines a base space of a bundle.

1.2.4 Coupling of the Disturbed Eigenanteil Equations

Here our approach is oriented on equations for multimode laser (Haken, 1985). We do this step by step and start with a small perturbation of the coupled Eigenanteile. This means that we just add particle number operators for each type of quantized bosonic fields e.g. $\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}$ that counts the number of "**ms**-messages" (n_{ms}). This approach reads as:

$$\begin{aligned}\frac{d}{dt}\mathbf{ms}_{jk}^{\dagger} = {} & c_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + c_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} - d_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \\ & (c_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger} - d_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk})) + \hat{g}_{jk}^{ms}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + \hat{g}_{jk}^{cr}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \\ & \hat{g}_{jk}^{mt}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}). \end{aligned} \tag{1.60}$$

We set (with indices dropped) $\mathbf{ms} = r_1 e^{i\varphi_1}$, $\mathbf{cr} = r_2 e^{i\varphi_2}$, $\mathbf{mt}^{\dagger} = r_3 e^{i\varphi_3}$, and get the transformed equation:

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r}_1 - \mathrm{i}\mathrm{r}_1\frac{\mathrm{d}}{\mathrm{dt}}\varphi_1 = \mathrm{c}^{\mathrm{ms}}\mathrm{r}_1 - \mathrm{d}^{\mathrm{ms}}(\mathrm{r}_1)^3 + \mathrm{c}^{\mathrm{cr}}\mathrm{r}_2\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_2)} - \mathrm{d}^{\mathrm{cr}}(\mathrm{r}_2)^3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_2)} + \\ \mathrm{c}^{\mathrm{mt}}\mathrm{r}_3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_3)} - \mathrm{d}^{\mathrm{mt}}(\mathrm{r}_3)^3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_3)} + (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\mathrm{e}^{\mathrm{i}\varphi_1}. \tag{1.61}$$

Here we can set $(\mathrm{r}_1)^2 = \mathrm{n}_{\mathrm{ms}}$, etc. After the splitting of this formula into a real part and imaginary part, the resulting equations are:

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r}_1 = \mathrm{c}^{\mathrm{ms}}\mathrm{r}_1 - \mathrm{d}^{\mathrm{ms}}(\mathrm{r}_1)^3 + (\mathrm{c}^{\mathrm{cr}}\mathrm{r}_2 - \mathrm{d}^{\mathrm{cr}}(\mathrm{r}_2)^3)\cos(\varphi_1-\varphi_2) + \\ (\mathrm{c}^{\mathrm{mt}}\mathrm{r}_3 - \mathrm{d}^{\mathrm{mt}}(\mathrm{r}_3)^3)\cos(\varphi_1-\varphi_3) + (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\cos(\varphi_1). \tag{1.62}$$

$$\mathrm{r}_1\frac{\mathrm{d}}{\mathrm{dt}}\varphi_1 = (\mathrm{c}^{\mathrm{cr}}\mathrm{r}_2 - \mathrm{d}^{\mathrm{cr}}(\mathrm{r}_2)^3)\sin(\varphi_1-\varphi_2) + (\mathrm{c}^{\mathrm{mt}}\mathrm{r}_3 - \mathrm{d}^{\mathrm{mt}}(\mathrm{r}_3)^3) \\ \sin(\varphi_1-\varphi_3) + (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\sin(\varphi_1). \tag{1.63}$$

We get equivalent formulas for the two other creation operators (the annihilation operators deliver identical equations as the creation operators).

In the next step we multiply the summed and weighted particle number expression $\hat{\mathrm{g}}^{\mathrm{ms}}_{\mathrm{jk}}(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{cr}}_{\mathrm{jk}}(\mathbf{cr}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{mt}}_{\mathrm{jk}}(\mathbf{mt}^{\dagger}_{\mathrm{jk}}\mathbf{mt}_{\mathrm{jk}})$ with the different creation operators e.g. with $\mathbf{ms}^{\dagger}_{\mathrm{jk}}$. This addition models the fact that the inversion (more fermionic message receiver are in a higher state (excited) than in a lower basic states; named saturation in laser technology) will be reduced by all three fields (modes):

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}^{\dagger}_{\mathrm{jk}} = \mathrm{c}^{\mathrm{ms}}_{\mathrm{jk}}\mathbf{ms}^{\dagger}_{\mathrm{jk}} - \mathrm{d}^{\mathrm{ms}}_{\mathrm{jk}}\mathbf{ms}^{\dagger}_{\mathrm{jk}}(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}) + \mathrm{c}^{\mathrm{cr}}_{\mathrm{jk}}\mathbf{cr}^{\dagger}_{\mathrm{jk}} - \mathrm{d}^{\mathrm{cr}}_{\mathrm{jk}}\mathbf{cr}^{\dagger}_{\mathrm{jk}}(\mathbf{cr}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}) + \\ \mathrm{c}^{\mathrm{mt}}_{\mathrm{jk}}\mathbf{mt}^{\dagger}_{\mathrm{jk}} - \mathrm{d}^{\mathrm{mt}}_{\mathrm{jk}}\mathbf{mt}^{\dagger}_{\mathrm{jk}}(\mathbf{mt}^{\dagger}_{\mathrm{jk}}\mathbf{mt}_{\mathrm{jk}}) + \\ \Big(\hat{\mathrm{g}}^{\mathrm{ms}}_{\mathrm{jk}}(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{cr}}_{\mathrm{jk}}(\mathbf{cr}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{mt}}_{\mathrm{jk}}(\mathbf{mt}^{\dagger}_{\mathrm{jk}}\mathbf{mt}_{\mathrm{jk}})\Big)\mathbf{ms}^{\dagger}_{\mathrm{jk}} + \\ \Big(\hat{\mathrm{g}}^{\mathrm{ms}}_{\mathrm{jk}}(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{cr}}_{\mathrm{jk}}(\mathbf{cr}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{mt}}_{\mathrm{jk}}(\mathbf{mt}^{\dagger}_{\mathrm{jk}}\mathbf{mt}_{\mathrm{jk}})\Big)\mathbf{cr}^{\dagger}_{\mathrm{jk}} + \\ \Big(\hat{\mathrm{g}}^{\mathrm{ms}}_{\mathrm{jk}}(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{cr}}_{\mathrm{jk}}(\mathbf{cr}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}) + \hat{\mathrm{g}}^{\mathrm{mt}}_{\mathrm{jk}}(\mathbf{mt}^{\dagger}_{\mathrm{jk}}\mathbf{mt}_{\mathrm{jk}})\Big)\mathbf{mt}^{\dagger}_{\mathrm{jk}}. \tag{1.64}$$

The solution of this equation reads now:

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{r}_1 - \mathrm{i}\mathrm{r}_1\frac{\mathrm{d}}{\mathrm{dt}}\varphi_1 = \mathrm{c}^{\mathrm{ms}}\mathrm{r}_1 - \mathrm{d}^{\mathrm{ms}}(\mathrm{r}_1)^3 + \mathrm{c}^{\mathrm{cr}}\mathrm{r}_2\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_2)} - \mathrm{d}^{\mathrm{cr}}(\mathrm{r}_2)^3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_2)} + \\ \mathrm{c}^{\mathrm{mt}}\mathrm{r}_3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_3)} - \mathrm{d}^{\mathrm{mt}}(\mathrm{r}_3)^3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_3)} + \\ (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\mathrm{r}_1\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_1)} + \\ (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\mathrm{r}_2\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_2)} + \\ (\hat{\mathrm{g}}^{\mathrm{ms}}(\mathrm{r}_1)^2 + \hat{\mathrm{g}}^{\mathrm{cr}}(\mathrm{r}_2)^2 + \hat{\mathrm{g}}^{\mathrm{mt}}(\mathrm{r}_3)^2)\mathrm{r}_3\mathrm{e}^{\mathrm{i}(\varphi_1-\varphi_3)}. \tag{1.65}$$

Finally we include cubic operator terms like $\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{cr}_{\mathrm{jk}}$, they describe the indirect interactions between the bosonic fields (field modes) that are initiated by their interactions with the fermionic agents $\mathbf{a}^{\dagger}_{\mathrm{j}}$ and $\mathbf{a}_{\mathrm{j}}$. For reason of simplicity we do not

write down new coupling coefficients (for precise calculations they must be adapted and changed)

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{jk}^{\dagger} =& c_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + c_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} - d_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \\ & c_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger} - d_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}) + \\ & \left(\hat{g}_{jk}^{ms}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + \hat{g}_{jk}^{cr}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \hat{g}_{jk}^{mt}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk})\right)\mathbf{ms}_{jk}^{\dagger} + \\ & \left(\hat{g}_{jk}^{ms}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + \hat{g}_{jk}^{cr}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \hat{g}_{jk}^{mt}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk})\right)\mathbf{cr}_{jk}^{\dagger} + \\ & \left(\hat{g}_{jk}^{ms}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + \hat{g}_{jk}^{cr}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \hat{g}_{jk}^{mt}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk})\right)\mathbf{mt}_{jk}^{\dagger} + \\ & \hat{g}_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}^{\dagger}\mathbf{cr}_{jk} + \hat{g}_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}^{\dagger}\mathbf{mt}_{jk} + \hat{g}_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}^{\dagger}\mathbf{ms}_{jk} + \\ & \hat{g}_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}^{\dagger}\mathbf{mt}_{jk} + \hat{g}_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}^{\dagger}\mathbf{ms}_{jk} + \hat{g}_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}^{\dagger}\mathbf{cr}_{jk}.\end{aligned} \tag{1.66}$$

The result is:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}r_1 - ir_1\frac{\mathrm{d}}{\mathrm{dt}}\varphi_1 =& c^{ms}r_1 - d^{ms}(r_1)^3 + c^{cr}r_2e^{i(\varphi_1-\varphi_2)} - d^{cr}(r_2)^3e^{i(\varphi_1-\varphi_2)} + \\ & c^{mt}r_3e^{i(\varphi_1-\varphi_3)} - d^{mt}(r_3)^3e^{i(\varphi_1-\varphi_3)} + (\hat{g}^{ms}(r_1)^2 + \\ & \hat{g}^{cr}(r_2)^2 + \hat{g}^{mt}(r_3)^2)r_1e^{i(\varphi_1-\varphi_1)} + \\ & (\hat{g}^{ms}(r_1)^2 + \hat{g}^{cr}(r_2)^2 + \hat{g}^{mt}(r_3)^2)r_2e^{i(\varphi_1-\varphi_2)} + \\ & (\hat{g}^{ms}(r_1)^2 + \hat{g}^{cr}(r_2)^2 + \hat{g}^{mt}(r_3)^2)r_3e^{i(\varphi_1-\varphi_3)} + \\ & \hat{g}_{jk}^{ms}(r_1)^2r_2e^{i(\varphi_2-\varphi_1)} + \hat{g}_{jk}^{ms}(r_1)^2r_3e^{i(\varphi_3-\varphi_1)} + \\ & \hat{g}_{jk}^{cr}(r_2)^2r_1e^{i2(\varphi_1-\varphi_2)} + \hat{g}_{jk}^{cr}(r_2)^2r_3e^{i(\varphi_1+\varphi_3-2\varphi_2)} + \\ & \hat{g}_{jk}^{mt}(r_3)^2r_1e^{i2(\varphi_1-\varphi_3)} + \hat{g}_{jk}^{mt}(r_3)^2r_2e^{i(\varphi_1+\varphi_2-2\varphi_3)}.\end{aligned} \tag{1.67}$$

One additional possible quadratic direct coupling of the bosonic operators that is initiated by similar atomic interactions (Zaslavsky, 2007) is given by the following equations:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{jk}^{\dagger} =& c_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger} - d_{jk}^{ms}\mathbf{ms}_{jk}^{\dagger}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + c_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} - d_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \\ & c_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger} - d_{jk}^{mt}\mathbf{mt}_{jk}^{\dagger}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}) + \\ & g_{jk}\Big((\mathbf{ms}_{jk}^{\dagger})^2 - (\mathbf{ms}_{jk})^2 + (\mathbf{cr}_{jk}^{\dagger})^2 - (\mathbf{cr}_{jk})^2 + (\mathbf{mt}_{jk}^{\dagger})^2 - (\mathbf{mt}_{jk})^2\Big).\end{aligned} \tag{1.68}$$

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}\mathbf{ms}_{jk} =& c_{jk}^{ms}\mathbf{ms}_{jk} - d_{jk}^{ms}\mathbf{ms}_{jk}(\mathbf{ms}_{jk}^{\dagger}\mathbf{ms}_{jk}) + c_{jk}^{cr}\mathbf{cr}_{jk}^{\dagger} - d_{jk}^{cr}\mathbf{cr}_{jk}(\mathbf{cr}_{jk}^{\dagger}\mathbf{cr}_{jk}) + \\ & c_{jk}^{mt}\mathbf{mt}_{jk} - d_{jk}^{mt}\mathbf{mt}_{jk}(\mathbf{mt}_{jk}^{\dagger}\mathbf{mt}_{jk}) + \\ & g_{jk}\Big((\mathbf{ms}_{jk}^{\dagger})^2 - (\mathbf{ms}_{jk})^2 - (\mathbf{cr}_{jk}^{\dagger})^2 + (\mathbf{cr}_{jk})^2 + (\mathbf{mt}_{jk})^2 - (\mathbf{mt}_{jk}^{\dagger})^2\Big).\end{aligned} \tag{1.69}$$

There is not enough space in this subchapter to present all the details of the graphical solutions of all before mentioned distorted equations, but we have observed in parts that a tendency to asymptotic stability of the QFT-approach can be observed, if the coupling constants, the damping constants κ, and γ, and the relaxation time T are not space-time or even field dependent. The coupled equations "prefer" to decouple in direction to their Eigenanteile. A proof of this proposition can not be given at this time since this a problem in a very high parameter space and we are still on the way to perform in a systematic manner the necessary calculations, and we know that also chaotic solutions exist.

1.2.5 Information Model and Interactions of Structured Components

1.2.5.1 Interaction Revisited

The interaction of fermionic agents and bosonic message ("force") fields that represent different signal quanta like photons, intracellular signaling proteins or extracellular signaling proteins (e.g. synaptic, endocrine (hormone based), etc.) will be modeled in the first approach again by the interaction Hamilton operator (1.5):

$$\begin{aligned}\mathbf{H}_{\mathrm{I}} = \mathrm{i}\hbar \sum_{\mathrm{j,k,l,m,n}} \mathrm{g}^{\mathrm{ms}}_{\mathrm{jk}} \left(\boldsymbol{\alpha}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\mathbf{ms}^{\dagger}_{\mathrm{jk}}(\mathrm{n}) - \boldsymbol{\alpha}^{\dagger}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\mathbf{ms}_{\mathrm{jk}}(\mathrm{n}) \right) + \\ \mathrm{i}\hbar \sum_{\mathrm{j,k,l,m,n}} \mathrm{g}^{\mathrm{cr}}_{\mathrm{jk}} \left(\boldsymbol{\alpha}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\mathbf{cr}^{\dagger}_{\mathrm{jk}}(\mathrm{n}) - \boldsymbol{\alpha}^{\dagger}_{\mathrm{j}}(\mathrm{m},\ \mathrm{l})\mathbf{cr}_{\mathrm{jk}}(\mathrm{n}) \right) + \\ \mathrm{i}\hbar \sum_{\mathrm{j,k,l,m,n}} \mathrm{g}^{\mathrm{mt}}_{\mathrm{jk}} \left(\boldsymbol{\alpha}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\mathbf{mt}^{\dagger}_{\mathrm{jk}}(\mathrm{n}) - \boldsymbol{\alpha}^{\dagger}_{\mathrm{j}}(\mathrm{m},\mathrm{l})\mathbf{mt}_{\mathrm{jk}}(\mathrm{n}) \right).\end{aligned} \tag{1.70}$$

We use the state flip (transition) operators $\boldsymbol{\alpha}^{\dagger}_{\mathrm{j}}$ and $\boldsymbol{\alpha}_{\mathrm{j}}$, that have been introduced in Sect. 1.2.1; further we consider the state l (e.g. $l = 0$, here we use the cursive l in order to avoid confusion with the number 1) as resting state, and we use the earlier mode index k as an identification of fermionic agent. The absorption of a message brings the receiver into an excited state $\mathrm{m} = 1$. We repeated above the previous defined interaction Hamiltonian since the application of the new definition of information (calculated by operators that create or annihilate quantized fields) implicates a modified version of $\mathbf{H}_{\mathrm{I}}$ that is primarily based on mutual message exchange. Being on the way to present the QFT-based description of information as an operator whose expectation values depend on the involved states (can be considered a context awareness) we have to consider shortly two possible probability distributions of the message exchange since the kind of message exchange influences the definition of information.

In addition we assume the coherence of all message exchanges (above the laser threshold that is considered as a control parameter that defines a positive net gain) so all expectation values of these messages obey the Poisson distribution. Hereby all the different messages (of the fixed type n) that are exchanged in the activity

phase of the operator pair can be considered as a "message-field" that responds to a certain distribution function. In our case of coherency this is the Poisson distribution for the number N of transmitted messages of type n (photonic statistics). The kind of statistics can be expressed by the number operator e.g. for $\mathbf{N_{ms}} = \mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}$. The variance (dropping of the indices **ms**) of this message number is defined by

$$\langle(\mathbf{N} - \langle\mathbf{N}\rangle)^2\rangle = \langle\mathbf{N}^2\rangle - \langle\mathbf{N}\rangle^2 = \langle(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}})^2\rangle - \langle\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}\rangle^2.$$

For the Poisson distribution function the following approximation is true $\langle\mathbf{N}^2\rangle - \langle\mathbf{N}\rangle^2 \approx \langle\mathbf{N}\rangle$. This means that the messages are send in a fixed averaged time distance. If the Bose-Einstein statistics is fulfilled then the value of the variance is different $\langle\mathbf{N}^2\rangle - \langle\mathbf{N}\rangle^2 = \langle(\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}})^2\rangle - \langle\mathbf{ms}^{\dagger}_{\mathrm{jk}}\mathbf{ms}_{\mathrm{jk}}\rangle^2 = \langle\mathbf{N}\rangle(\langle\mathbf{N}\rangle + 1)$. The relevance of this distribution function is the lumping of messages, the ramification is the decoherence of the field and therefore no cooperation between fermionic agents come about.

1.2.5.2 Postulation of Information Rules

We define the concept of information by four characteristic rules:

1. *Synchronisation*. There is a mechanism that synchronises the communication between two or more agents (partners) meaning the semantic compatibility of the agents that initiate a communication session. In practice such a communication channel is handcrafted by a developer.
2. *Compatibility*. The receiver of a message has in the sense of Pulitzer not only to acknowledge the received message; but even more important is the demand that the receiver "understands" the matter of the sender and behaves (reacts) in a manner as it is assumed by the sender (Haken, 1988). In more detail this request denotes that both communication partners are in a configuration where they are compatible in their internal states $\mathrm{S_j}$, in their knowledge $\mathrm{W_{jk}}$ and even more important their distance of information $\mathrm{dist}(\mathrm{Inf_j}, \mathrm{Inf_k})$ is below a given threshold. If all conditions are fulfilled then a semantic equivalence is available.
3. *Component Building*. Two components (agents) j and k continue the "negotiation" concerning their combination as long as their common knowledge $\mathrm{W_{jk}}$ is minimised (similar status of knowledge for both agents defined by a symmetric Kullback measure), the expectation values of the individuals state operators $\mathrm{S_j} = \langle\mathbf{S_j}\rangle = \mathrm{tr}(\boldsymbol{\rho}\mathbf{S_j})$ and $\mathrm{S_k} = \langle\mathbf{S_j}\rangle = \mathrm{tr}(\boldsymbol{\rho}\mathbf{S_k})$ are maximised, and the common information $\mathrm{Inf_j} = \langle\mathbf{Inf_j}\rangle = \mathrm{tr}\langle\boldsymbol{\rho}\mathbf{Inf_j}\rangle$ of agent j is minimised (for more detailed calculations see Sect. 3.3.2).
4. *Open ended evolution*. The process of information collection is iterative and describes three basic algorithms. These are the information that describe the genotype (defining the evolution and fitness), the phenotype (defining the behaviour and higher cognitive abilities) and the controllers coordinating the gathering of information (learning) concerning evolution and fitness and the interactions of these two different learning procedures. The whole process ends if a minimum (infimum) of total information in a given environment has been achieved

(open systems). Here one open question is how the fitness of a given individual influences its genome in that different species emerge that are genetically incompatible.

Here $\boldsymbol{\rho}$ is the density operator (density matrix) and tr is the trace of a matrix: e.g. $\mathrm{tr}(\boldsymbol{\rho}\mathbf{S}_\mathrm{j}) = \sum_\mathrm{m} \langle \mathrm{m} \left| \boldsymbol{\rho}\mathbf{S}_\mathrm{j} \right| \mathrm{m} \rangle$. The total individual state operator $\mathbf{S}_\mathrm{j}$ of an agent j with several internal states (state sum) is defined by:

$$\mathbf{S}_\mathrm{j} = \sum_\mathrm{k} \mathrm{e}^{\left(\boldsymbol{\mu}_\mathrm{j}\left(\langle \mathbf{N}_\mathrm{jk}^2 \rangle - \langle \mathbf{N}_\mathrm{jk} \rangle^2\right) - \mathrm{W}_\mathrm{jk}\right)/\langle \mathbf{N}_\mathrm{j}^2 \rangle}, \tag{1.71}$$

where $\langle \mathbf{N}_\mathrm{jk} \rangle = \langle \mathbf{ms}_\mathrm{jk}^\dagger \mathbf{ms}_\mathrm{jk} \rangle$ is the mean value of messages that agents j and k exchange, and $\langle \mathbf{N}_\mathrm{jk}^2 \rangle = \langle \mathbf{ms}_\mathrm{jk}^\dagger \mathbf{ms}_\mathrm{jk} \mathbf{ms}_\mathrm{jk}^\dagger \mathbf{ms}_\mathrm{jk} \rangle$ is the expectation value of $\mathbf{N}_\mathrm{jk}^2$. The expression $(\langle \mathbf{N}_\mathrm{jk}^2 \rangle - \langle \mathbf{N}_\mathrm{jk} \rangle^2)$ defines the expectation value of the quadratic fluctuations. The knowledge W_jk is defined as the symmetric Kullback measure of the two probability distributions of agents j and k, (see Sect. 3.3, Eq. 3.8).

The activities of an agent can be compared to that one of a chemical potential that characterise the possibilities of a substance to interact with other substrates, to transfer into other states and to distribute all over the space. A reaction, conversion, and redistribution can only occur without enforcement if the potential in the initial state is greater as in the final state. These features are also relevant to statistical physics where we have borrowed our definition (here we neglect for brevity other contributions e.g. of $\mathbf{cr}_\mathrm{jk}, \mathbf{mt}_\mathrm{jk}$)

$$\boldsymbol{\mu}_\mathrm{j} = \sum_{\mathrm{k}=0}^{\mathrm{K}} \sum_{\mathrm{m_j}=0}^{\mathrm{M}} \ln\left(\mathrm{g}_\mathrm{jk}^\mathrm{ms} \mathbf{a}_\mathrm{j}^\dagger(\mathrm{m_j}) \mathbf{ms}_\mathrm{jk} + \mathrm{g}_\mathrm{jk}^{\mathrm{ms}*}\left(\mathbf{a}_\mathrm{j}(\mathrm{m_j}) \mathbf{ms}_\mathrm{jk}^\dagger\right)\right). \tag{1.72}$$

The quantisation of $\boldsymbol{\mu}_\mathrm{j}$ is performed by the inclusion of quantum field operators and hereby finally we get the quantisation of the information. Further benefits of the approach with the "chemical potential" are the possibility to combine different agents to a bigger unit, to model diffusion and the possibility to describe the adaptation of an agent performed by different phase transitions (e.g. solid state, fluid state, gaseous state). This can be initiated by special messages and appropriate coupling constants that generate an appropriate state transition. Thus it is possible to consider e.g. environmental restrictions as dedicated messages from outside.

The operator of the "statistical potential" of an agent j, after it evaluates all messages it has received, is defined by:

$$\boldsymbol{\Omega}_\mathrm{j} = -\langle \mathbf{N}_\mathrm{j}^2 \rangle \ln\left(\sum_\mathrm{k} e^{\left(\mu_\mathrm{j}\left(\langle \mathbf{N}_\mathrm{jk}^2 \rangle - \langle \mathbf{N}_\mathrm{jk} \rangle^2\right) - \mathrm{W}_\mathrm{jk}\right)/\langle \mathbf{N}_\mathrm{j}^2 \rangle}\right) = -\langle \mathbf{N}_\mathrm{j}^2 \rangle \ln \mathbf{S}_\mathrm{j}. \tag{1.73}$$

Since we consider $\boldsymbol{\mu}_\mathrm{j}$ as a dynamic variable the state operator $\mathbf{S}_\mathrm{j}$ can be characterised beside physical or chemical terms also by states of nonlinear dynamics like equilibrium states, periodic states or even chaotic states. Finally we define the operator of information as the derivative of the operational "statistical potential" with

respect to $\left\langle \mathbf{N}_j^2 \right\rangle$ that correspondsd to the temperature of the state S_j in statistical physics:

$$\mathbf{Inf}_j = -\frac{\partial \boldsymbol{\Omega}_j}{\partial \left\langle \mathbf{N}_j^2 \right\rangle}. \tag{1.74}$$

Transferred to our approach this means that the synchronisation of two agents (e.g. robot-cells) is guaranteed, since the communicative coherence and message exchange symmetry between all agents is established. Expressed in a new interaction Hamiltonian this might look like:

$$\begin{aligned}\mathbf{H}_I' =& i\hbar \sum_{j,k} g_{jk}^{ms} \mathbf{ms}_{jk}^{\dagger} \mathbf{ms}_{jk} \left(\boldsymbol{\alpha}_j(m_j,0)\boldsymbol{\alpha}_k^{\dagger}(m_k,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0)\boldsymbol{\alpha}_k(m_k,0) \right) + \\ & i\hbar \sum_{j,k} g_{jk}^{cr} \mathbf{cr}_{jk}^{\dagger} \mathbf{cr}_{jk} \left(\boldsymbol{\alpha}_j(m_j,0)\boldsymbol{\alpha}_k^{\dagger}(m_k,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0)\boldsymbol{\alpha}_k(m_k,0) \right) + \\ & i\hbar \sum_{j,k} g_{jk}^{mt} \mathbf{mt}_{jk}^{\dagger} \mathbf{mt}_{jk} \left(\boldsymbol{\alpha}_j(m_j,0)\boldsymbol{\alpha}_k^{\dagger}(m_k,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0)\boldsymbol{\alpha}_k(m_k,0) \right).\end{aligned} \tag{1.75}$$

This interaction Hamiltonian fulfills the rule of synchronisation mentioned before (rule 1). The equation of motion for $\boldsymbol{\alpha}_j^{\dagger}(m_j,0)$ are:

$$\begin{aligned}\frac{d}{dt}\boldsymbol{\alpha}_j^{\dagger}(m_j,0) =& \frac{i}{\hbar}[\mathbf{H}_I', \boldsymbol{\alpha}_j^{\dagger}(m_j,0)] \\ =& 2\sum_k g_{jk}^{ms} \mathbf{ms}_{jk}^{\dagger} \mathbf{ms}_{jk} \left(\boldsymbol{\alpha}_j^{\dagger}(0,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0) \right) + \\ & 2\sum_k g_{jk}^{cr} \mathbf{cr}_{jk}^{\dagger} \mathbf{cr}_{jk} \left(\boldsymbol{\alpha}_j^{\dagger}(0,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0) \right) + \\ & 2\sum_k g_{jk}^{mt} \mathbf{mt}_{jk}^{\dagger} \mathbf{mt}_{jk} \left(\boldsymbol{\alpha}_j^{\dagger}(0,0) - \boldsymbol{\alpha}_j^{\dagger}(m_j,0) \right).\end{aligned} \tag{1.76}$$

Consequently we can solve the equations of motion for all operators that are part of the definition of the chemical potential $\boldsymbol{\mu}_j$, then with the aid of the knowledge W_{jk} and the aid of the "statistical potential" $\boldsymbol{\Omega}_j$ a value for Inf_j can be calculated. If the knowledge of two robot-cells j and k are compatible (e.g. the difference between Inf_j and Inf_k is below a threshold) then not only the dynamic of the information flow between every two agents but also the total information flow can be calculated and analysed (e.g. by phase portraits) whether the combination of two robot-cells or even of an organisms is a stable equilibrium state or this state is unstable.

1.2.5.3 Structured Objects

The activities of different genes/cells are usually clearly different and they might have strong coupling constants. This fact implies that such units can operate as "seeds" for agent-based (morphological) networks that are constructed by strong

message exchange and can be considered as the predefinition of a final structure. The result is a topological framework that describes the structure of an organism.

We try to demonstrate our approach by the example of water. A single water molecule is build up by two hydrogen atoms H_1, H_2 and one molecule O that are bounded together by two covalent bonds. This means that there is a communication only between H_1 and O, respectively H_2 and O. A possible interaction Hamiltonian is, where we only mention **ms** and $\mathbf{ms}^\dagger$ that represent primarily external effects (dominant internal processes are manifested in a Hamiltonian that owns the same structure):

$$\begin{aligned}
\mathbf{H}_{\mathrm{I}}^{\mathrm{H_2O}} = \mathrm{i}\hbar \mathrm{g}_{\mathrm{HO}}^{\mathrm{ms}} \Big(& \left(\mathbf{ms}_{\mathrm{H_1O}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{H_1}} (\mathrm{m}_{\mathrm{H_1}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_1O}} \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{O1}}, 0)\right) \\
& \left(\mathbf{ms}_{\mathrm{H_2O}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{H_2}} (\mathrm{m}_{\mathrm{H_2}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_2O}} \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) - \\
& \left(\mathbf{ms}_{\mathrm{H_2O}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_2O}} \boldsymbol{\alpha}_{\mathrm{H_2}}^{\dagger} (\mathrm{m}_{\mathrm{H_{2,0}}})\right) \\
& \left(\mathbf{ms}_{\mathrm{H_1O}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_1}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_1O}} \boldsymbol{\alpha}_{\mathrm{H_1}}^{\dagger} (\mathrm{m}_{\mathrm{H_1}}, 0)\right) \Big) \\
= \mathrm{i}\hbar \mathrm{g}_{\mathrm{HO}}^{\mathrm{ms}} \Big(& \left(\mathbf{ms}_{\mathrm{H_1O}}^{\dagger} \mathbf{ms}_{\mathrm{H_1O}} \mathbf{ms}_{\mathrm{H_2O}}^{\dagger} \mathbf{ms}_{\mathrm{H_2O}}\right) \\
& \left(\boldsymbol{\alpha}_{\mathrm{H_1}} (\mathrm{m}_{\mathrm{H_1}}, 0) \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{O1}}, 0) \boldsymbol{\alpha} \mathrm{H}_2 (\mathrm{m}_{\mathrm{H_{2,0}}}) \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) - \\
& \left(\mathbf{ms}_{\mathrm{H_2O}}^{\dagger} \mathbf{ms}_{\mathrm{H_2O}} \mathbf{ms}_{\mathrm{H_1O}}^{\dagger} \mathbf{ms}_{\mathrm{H_1O}}\right) \\
& \left(\boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_2}}, 0) \boldsymbol{\alpha}_{\mathrm{H_2}}^{\dagger} (\mathrm{m}_{\mathrm{H_2}}, 0) \boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_1}}^{\dagger} (\mathrm{m}_{1}, 0)\right) \Big) \\
= \mathrm{i}\hbar \mathrm{g}_{\mathrm{HO}}^{\mathrm{ms}} \Big(& \mathbf{N}_{\mathrm{H_1O}}^{\mathrm{ms}} \mathbf{N}_{\mathrm{H_2O}}^{\mathrm{ms}} \left(\boldsymbol{\alpha}_{\mathrm{H_1}} (\mathrm{m}_{\mathrm{H_1}}, 0) \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{0_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_2}} (\mathrm{m}_{\mathrm{H_2}}, 0) \boldsymbol{\alpha}_{\mathrm{O}}^{\dagger} (\mathrm{m}_{\mathrm{0_2}}, 0)\right) - \\
& \mathbf{N}_{\mathrm{H_2O}}^{\mathrm{ms}} \mathbf{N}_{\mathrm{H_1O}}^{\mathrm{ms}} \left(\boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_2}}, 0) \boldsymbol{\alpha}_{\mathrm{H_2}}^{\dagger} (\mathrm{m}_{\mathrm{H_2}}, 0) \boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_1}}^{\dagger} (\mathrm{m}_{\mathrm{H_1}}, 0)\right) \Big)
\end{aligned} \tag{1.77}$$

Here $\mathbf{N}_{\mathrm{H_1O}}^{\mathrm{ms}}$ and $\mathbf{N}_{\mathrm{H_2O}}^{\mathrm{ms}}$ are the "message counter operators" and the index H_2O does not mean the water molecule but the message exchange between the second hydrogen atom and the oxygen. This approach can be considered as generalised information based diffusion approach in the sense that between all participants there is a two-way synchronised message exchange. But till now the connection of this Hamiltonian with the standard theory e.g. (Haken & Wolf, 1998) is not obvious, and indeed, the consensus with the reality is still open and still has to be proofed.

The next question that arises is how we can model the structure of a H_2O molecule. This can only be done if we not only consider the message exchange but also the activities that the receivers perform in consequence of exchange of "news". In our case we must consider the subelements of the three molecules. These are the electrons (more precise: the interactions of the orbitals of the participating

electrons) that make the covalent bond (overlapping of adjacent atomic orbitals) polar. In more detail we must consider the interaction of the two $1s^1$ orbitals of the hydrogen atoms with the $2p^4$ orbit of the oxygen under the strong observation of the Pauli Principe. Here we do not delve into the details of quantum physic (respectively physical chemistry) but argue with the relevant symmetry groups for the H_2O molecule that consider all orbitals together. The drawback of this procedural message is that this kind of the group theory cannot explain why the water molecule is stable but e.g. He_2 is unstable. Such stability declaration can usually only be given by direct calculations of the interaction patterns.

The oxygen nucleus draws electron orbits away and as a result an electrical tetraeder structure is generated (Alberts *et al.*, 2008); the symmetry group is therefore T_d (full notation: $\overline{4}3m$). The spatial structure of it is defined by C_{2v}.

The stabilities of individual parts can again be calculated e.g. by

$$\frac{\mathrm{d}}{\mathrm{dt}}(\mathbf{ms}^{\dagger}_{\mathrm{H_1O}}\boldsymbol{\alpha}_{\mathrm{H_1}}(\mathrm{m_{H_1}},0)) = \frac{\mathrm{i}}{\hbar}\left[\mathbf{H}_{\mathrm{I}}^{\mathrm{H_2O}}, (\mathbf{ms}^{\dagger}_{\mathrm{H_1O}}\boldsymbol{\alpha}_{\mathrm{H_1}}(\mathrm{m_{H_1}},0))\right] \tag{1.78}$$

and analysed by the methods of nonlinear dynamics as we have done before.

The energy calculation can be done by the standard model of harmonic oscillator. Here we couple two oscillators together. The non interacting Hamiltonian is just the well known sum of the two separate oscillators

$$\mathbf{H} = \hbar\sum_{\underline{\mathrm{i}}=1}^{2}\omega_{\underline{\mathrm{p}}_{\underline{\mathrm{i}}}}\left(\mathbf{a}^{\dagger}_{\underline{\mathrm{p}}_{\underline{\mathrm{i}}}}\mathbf{a}_{\underline{\mathrm{p}}_{\underline{\mathrm{i}}}} + \frac{1}{2}\right), \tag{1.79}$$

where $\underline{\mathrm{p}}_{\underline{\mathrm{i}}}$ is a 3-d momentum, and the eigenfunctions are also well known (Hermite polynomials).

The reader recognised certainly that the symmetry groups of the H_2O molecule are not really relevant to the connection of some few robot-cells to a structured component of a symbiotic organism, but we operate here again in strong analogy to this algebraic method. The interaction between the participating active cells is modeled by the contents of the exchanged messages. Thus we can describe whether two cells own similar genomes and therefore are compatible to aggregate to a major component. A message can also characterise e.g. pure physical effects that two cells merge together since they are by chance (probability) sticked together close enough and mutual attractive forces merge them to a new unit. By this approach defects of individual cells can be modeled and detected. The remedy of such defects can be performed by the calculation and comparison of the pairwise exchanged information and thus even modify or exchange malfunctioning robot cells. The Hamiltonian does not deliver these information but it will be used to describe and to analyse the dynamics of the flow of the dedicated messages, and the symmetry of such a flow can be defined by a relevant group. Here we can read in more detail e.g. whether cells are attractive, repulsive, or neutral. Further such a group could deliver a hint which symmetry group governs the information space. In sequence, this knowledge delivers the definition of a metric in such a space and therefore provide us with the correct calculation of the distance of two information values.

Now we turn to the coupling of two water molecules. The two water molecules are connected by a so called hydrogen bond between the oxygen atoms via one mediating hydrogen molecule. We model this by the following Hamiltonian:

$$\begin{aligned}
\mathbf{H}_{\mathrm{I}}^{\mathrm{water}} = \mathrm{i}\hbar \mathrm{g}_{\mathrm{OO}}^{\mathrm{ms}} \Big(& \left(\mathbf{ms}_{\mathrm{O_1H_3}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{O_1}} (\mathrm{m}_{\mathrm{O_1}}, 0)\right) \left(\mathbf{ms}_{\mathrm{O_1H_3}} \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_3}}, 0)\right) \\
& \left(\mathbf{ms}_{\mathrm{H_3O_2}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{H_3}} (\mathrm{m}_{\mathrm{H_3}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_3O_2}} \boldsymbol{\alpha}_{\mathrm{O_2}}^{\dagger} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) - \\
& \left(\mathbf{ms}_{\mathrm{H_3O_2}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{O_2}} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) \left(\mathbf{ms}_{\mathrm{H_3O_2}} \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_{3,0}}})\right) \\
& \left(\mathbf{ms}_{\mathrm{O_1H_3}}^{\dagger} \boldsymbol{\alpha}_{\mathrm{H_3}} (\mathrm{m}_{\mathrm{H_3}}, 0)\right) \left(\mathbf{ms}_{\mathrm{O_1H_3}} \boldsymbol{\alpha}_{\mathrm{O_1}}^{\dagger} (\mathrm{m}_{\mathrm{O_1}}, 0)\right) \Big) \\
= \mathrm{i}\hbar \mathrm{g}_{\mathrm{HO}}^{\mathrm{ms}} \Big(& \left(\mathbf{ms}_{\mathrm{O_1H_3}}^{\dagger} \mathbf{ms}_{\mathrm{O_1H_3}} \mathbf{ms}_{\mathrm{H_3O_2}}^{\dagger} \mathbf{ms}_{\mathrm{H_3O_2}}\right) \\
& \left(\boldsymbol{\alpha}_{\mathrm{O}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_3}}, 0) \boldsymbol{\alpha} \mathrm{H3} \left(\mathrm{m}_{\mathrm{H_{3,0}}}\right) \boldsymbol{\alpha}_{\mathrm{O_2}}^{\dagger} (\mathrm{m}_{\mathrm{O_2}}, 0)\right) - \\
& \left(\mathbf{ms}_{\mathrm{H_3O_2}}^{\dagger} \mathbf{ms}_{\mathrm{H_3O_2}} \mathbf{ms}_{\mathrm{O_1H_3}}^{\dagger} \mathbf{ms}_{\mathrm{O_1H_3}}\right) \\
& \left(\boldsymbol{\alpha}_{\mathrm{O_2}} (\mathrm{m}_{\mathrm{O_2}}, 0) \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_3}}, 0) \boldsymbol{\alpha}_{\mathrm{O_1}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_1}}^{\dagger} (\mathrm{m}_{\mathrm{H_1}}, 0)\right) \Big) \\
= \mathrm{i}\hbar \mathrm{g}_{\mathrm{HO}}^{\mathrm{ms}} \Big(& \mathbf{N}_{O_1H_3}^{ms} \mathbf{N}_{H_3O_2}^{ms} \left(\boldsymbol{\alpha}_{\mathrm{O_1}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_3}}, 0) \boldsymbol{\alpha}_{\mathrm{H_2}} (\mathrm{m}_{\mathrm{H_2}}, 0) \boldsymbol{\alpha}_{\mathrm{O_2}}^{\dagger} (\mathrm{m}_{\mathrm{0_2}}, 0)\right) - \\
& \mathbf{N}_{\mathrm{H_3O_2}}^{\mathrm{ms}} \mathbf{N}_{\mathrm{O_1H_3}}^{\mathrm{ms}} \left(\boldsymbol{\alpha}_{\mathrm{O_2}} (\mathrm{m}_{\mathrm{O_2}}, 0) \boldsymbol{\alpha}_{\mathrm{H_3}}^{\dagger} (\mathrm{m}_{\mathrm{H_3}}, 0) \boldsymbol{\alpha}_{\mathrm{O_1}} (\mathrm{m}_{\mathrm{O_1}}, 0) \boldsymbol{\alpha}_{\mathrm{H_1}}^{\dagger} (\mathrm{m}_{\mathrm{H_1}}, 0)\right) \Big).
\end{aligned} \tag{1.80}$$

Analogous to a symmetry group of water we have to search for a symmetry group of a developed organism. This can usually be done on geometrical level (e.g spherical or cylindrical symmetry) or, as we propose, on the level of the information flows of different organising agents that represent different major components (compartments) of an organism. Here we can study the flow of different message types that are send and received among these agents hereby defining the dynamic type of agent (e.g. attractive) and search for the symmetry groups that describe these flows.

1.2.5.4 Action Integral as Fitness Measure

We employ the action integral

$$\mathrm{W}_{12} = \int_{\mathrm{t_1}}^{\mathrm{t_2}} \mathscr{L}\left(\Phi(\mathrm{x}), \partial_{\mu}\Phi(\mathrm{x}), \Psi(\mathrm{x}), \partial_{\mu}\Phi(\mathrm{x})\right) \mathrm{dt} \tag{1.81}$$

to describe and evaluate the propagation of an action. We will use it to define a measure how fit (efficient) a combined structured object in comparison to its individual components is. Here $\mathscr{L}$ is the Lagrangian density operator, Φ is a bosonic

quantized field, and Ψ is a fermionic field that solves adequate Schrödinger equations. The Lagrangian density operator without interaction is:

$$\begin{aligned}\mathscr{L}_S \;=\; i\hbar\boldsymbol{\Phi}^{\dagger}(x)\,\dot{\boldsymbol{\Phi}}(x) \;+\; i\hbar\boldsymbol{\Psi}^{\dagger}(x)\,\dot{\boldsymbol{\Psi}}(x) \;-\frac{\hbar^2}{2m_{\boldsymbol{\Phi}}}\nabla\boldsymbol{\Phi}^{\dagger}(x)\,\nabla\boldsymbol{\Phi}(x) - \\ \Phi^{\dagger}(\mathrm{x})\,\mathrm{V}_{\Phi}(\mathrm{x})\,\Phi(\mathrm{x}) \;-\frac{\hbar^2}{2\mathrm{m}_{\Psi}}\nabla\Psi^{\dagger}(\mathrm{x})\,\nabla\Psi(\mathrm{x}) \;-\; \Psi^{\dagger}(\mathrm{x})\,\mathrm{V}_{\Psi}(\mathrm{x})\,\Psi(\mathrm{x}).\end{aligned} \tag{1.82}$$

In this formula ∇ is the 3-d Nabla operator and the point defines the temporal derivation. The general correlation between the Schrödinger-densities (Index S) densities and the interaction densities $\mathscr{L}_{\text{total}} \;=\; \mathscr{L}_{\text{S}} \;+\; \mathscr{L}_{\text{I}}$ and $\mathscr{H}_{\text{total}} \;=\; \mathscr{H}_{\text{S}} \;+\; \mathscr{H}_{\text{I}}$ is

$$\mathscr{H}_{\text{total}} \;=\; \pi_{\Phi}\dot{\Phi} \;+\; \pi_{\Phi^{\dagger}}\dot{\Phi}^{\dagger} \;+\; \pi_{\Psi}\dot{\Psi} \;+\; \pi_{\Psi^{\dagger}}\dot{\Psi}^{\dagger} \;-\; \mathscr{L}_{\text{total}}, \tag{1.83}$$

where $\pi_{\Phi} \;=\; \partial\mathscr{L}_{\text{total}}/\partial\dot{\Phi}$, etc. are the conjugated fields. For example we got for $\mathscr{H}_{\text{S}}$ the result $\left(\pi_{\Phi} \;=\; \mathrm{i}\hbar\Phi^{\dagger}, \pi_{\Psi} \;=\; \mathrm{i}\hbar\Psi^{\dagger}\right)$, all remaining conjugate fields are zero:

$$\begin{aligned}\mathscr{H}_{\text{S}} \;=& \frac{\hbar^2}{2\mathrm{m}_{\Phi}}\nabla\Phi^{\dagger}(\mathrm{x})\nabla\Phi(\mathrm{x}) + \Phi^{\dagger}(\mathrm{x})\mathrm{V}_{\Phi}(\mathrm{x})\Phi(\mathrm{x}) + \\ & \frac{\hbar^2}{2\mathrm{m}_{\Psi}}\nabla\Psi^{\dagger}(\mathrm{x})\nabla\Psi(\mathrm{x}) + \Psi^{\dagger}(\mathrm{x})\mathrm{V}_{\Psi}(\mathrm{x})\Psi(\mathrm{x}).\end{aligned} \tag{1.84}$$

The field operator Φ obeys an equal time commutation rule, whereas Ψ fulfills an equal time anti-commutation rule. The stability of a system is usually investigated by the form of potential that defines equilibrium states. In our case we have to fix the two potentials $\mathrm{V}_{\Psi}(\mathrm{x})$ and $\mathrm{V}_{\Phi}(\mathrm{x})$.

Typical potentials might be (Zaslavsky, 2007): the kicked oscillator, where a is the distance of the two positions

$$\begin{aligned}\mathrm{V}_{\Psi}(\mathrm{x}) = \frac{1}{2}\alpha\sum_{\mathrm{j}}\left(\underline{\mathrm{x}}_{\mathrm{j}+1} - \underline{\mathrm{x}}_{\mathrm{j}} - \mathrm{a}\right)^2 - \frac{1}{2}\beta\sum_{\mathrm{j}}\left(\underline{\mathrm{x}}_{\mathrm{j}+2} - \underline{\mathrm{x}}_{\mathrm{j}} - 2\mathrm{a}\right)^2 \\ +\gamma\sum_{\mathrm{j}}\mathrm{m}_{\mathrm{j}}\omega_{\mathrm{j}}^2\cos\left(2\pi\mathrm{j}\frac{\mathrm{t}}{\mathrm{T}}\right),\end{aligned} \tag{1.85}$$

and the perturbed oscillator

$$\mathrm{V}_{\Psi}(\mathrm{x}) \;=\; \frac{1}{2}\sum_{\mathrm{j}}\left(\frac{\mathrm{p}^2}{\mathrm{m}_{\mathrm{j}}} + \mathrm{m}_{\mathrm{j}}\omega_{\mathrm{j}}^2\mathrm{x}_{\mathrm{j}}^2\right) + \sum_{\mathrm{j}}\mathrm{m}_{\mathrm{j}}\omega_{\mathrm{j}}^2\cos\left(\mathrm{x}_{\mathrm{j}} - \mathrm{vt}\right). \tag{1.86}$$

The separate insertion of each of this two potentials into the Schrödinger equation delivers us with a standard solution. But we are more interested in the interaction representation rather than in the Schrödinger representation. A typical part of our information based interaction Hamiltonian H_{I} for a water molecule is:

$$\begin{aligned}\mathbf{H}_{\mathrm{I}} &= \mathbf{N}^{ms}_{H_1O}\boldsymbol{\alpha}\left(m_{H_1},0\right)\boldsymbol{\alpha}^{\dagger}_{O}\left(m_{O_1},0\right)\\ &= i\hbar\int \boldsymbol{\Phi}^{\dagger}_{H_1O}\left(x\right)\boldsymbol{\Phi}_{H_1O}\left(x\right)\Psi^{\dagger}_{H_1}\left(x',0\right)\Psi_{H_1}\left(x',m_{H_1}\right)\\ &\qquad \Psi^{\dagger}_{O}\left(x'',m_O\right)\Psi_{O}\left(x'',0\right)d^3x\,d^3x'\,d^3x''\\ &= i\hbar\int \mathscr{H}_I(x,x',x'')\,d^3x\,d^3x'\,d^3x''.\end{aligned} \tag{1.87}$$

To get the equation of motion e.g. for $\boldsymbol{\Phi}^{\dagger}_{H_1O}$ in the interaction representation the calculation follows the same pattern as before

$$\frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\Phi}^{\dagger}_{H_1O} = \frac{\mathrm{i}}{\hbar}\left[\mathbf{H}_{\mathrm{I}},\boldsymbol{\Phi}^{\dagger}_{H_1O}\right]. \tag{1.88}$$

Such equations have to be solved for all field operators that are part of the interaction Hamiltonian and afterwards inserted in the action integral

$$\mathrm{W}_{12} \;=\; \int_{\mathrm{t_1}}^{\mathrm{t_2}}\mathscr{L}_{\mathrm{I}}\left(\boldsymbol{\Phi}\left(\mathrm{x}\right),\boldsymbol{\Psi}\left(\mathrm{x}\right)\right)\mathrm{dt} = -\int_{\mathrm{t_1}}^{\mathrm{t_2}}\mathscr{H}_{\mathrm{I}}\left(\boldsymbol{\Phi}\left(\mathrm{x}\right),\boldsymbol{\Psi}\left(\mathrm{x}\right)\right)\mathrm{dt}. \tag{1.89}$$

Hereby we pay attention to the fact that the interaction Hamiltonian includes no derivatives therefore we can set $\mathscr{L}_{\mathrm{I}} = -\mathscr{H}_{\mathrm{I}}$. By the calculation of this integral (with even more elaborated formulas for $\mathbf{H}_{\mathrm{I}}$ like (1.77) we can achieve the result that a H_2O molecule is stable or not. In addition we can go on this way and calculate whether the combination of two H_2O molecules is stable and so on. Finally we can calculate whether water can originate or not. The maximal fitness is achieved if not only some stable liquid evolves but this liquid also holds all additional, positive features of water. This maximal fitness can not only be accomplished by the calculations of the action integral but also by explicit calculation of the relevant information as defined in 1.2.5.2.

The symmetry group of the Schrödinger equation and therefore for the field operators $\boldsymbol{\Phi}\left(x\right)$ and $\boldsymbol{\Psi}\left(x\right)$ is the $\mathrm{U}(1)$. This implies that the local gauge transformation for these operators is defined by the phase transformation $e^{\varphi(x)}$ (entirely analogous to classical electromagnetic fields) and $\mathscr{L}_{\mathrm{I}}$ must also be invariant under such arbitrary phase changes (analogous to classical electromagnetic fields). The Lagrangian density mentioned above fulfills this invariance requirement.[2]

1.2.5.5 Appendix

Fermionic anticommutators:

$$\left\{\mathbf{a}_{\mathrm{i}}\left(l\right),\mathbf{a}^{\dagger}_{\mathrm{j}}\left(m\right)\right\} = \delta_{\mathrm{ij}}\delta_{lm},\left\{\mathbf{a}_{\mathrm{i}}\left(l\right),\mathbf{a}_{\mathrm{j}}\left(m\right)\right\} = 0,\left\{\mathbf{a}^{\dagger}_{\mathrm{i}}\left(l\right),\mathbf{a}^{\dagger}_{\mathrm{j}}\left(m\right)\right\} = 0. \tag{1.90}$$

[2] Please remind that we mentioned in this contribution three different symmetries: symmetry of the interaction Hamiltionian, symmetry of the information flow, and symmetry of an organism (body compartments).

Bosonic commutators:

$$\left[\mathbf{b}_{\mathrm{i}}(l), \mathbf{b}_{\mathrm{j}}^{\dagger}(m)\right] = \delta_{\mathrm{ij}}\delta_{lm}, \left[\mathbf{b}_{\mathrm{i}}(l), \mathbf{b}_{\mathrm{j}}(m)\right] = 0, \left[\mathbf{b}_{\mathrm{i}}^{\dagger}(l), \mathbf{b}_{\mathrm{j}}^{\dagger}(m)\right] = 0. \tag{1.91}$$

Here l and m stands for the set of all internal states ("quantum") numbers including the time (commutators respectively anti-commutators are calculated at equal time), the indices i and j are labels of fermionic units (e.g. robot cells); in the case of bosonic operators they characterise the message type (bosonic field). If the commutator of two operators is zero then both equivalent fields (physical terms) can be measured simultaneously.

1.3 Functional and Reliability Modelling of Swarm Robotic Systems

Alan Winfield, Wenguo Liu, Jan Dyre Bjerknes

A robotic swarm is an example of a stochastic, dynamical and often non-linear system. Developing models that allow overall swarm properties to be predicted from the low-level parameters of the individual robots that comprise the swarm is challenging. For this reason many swarm robotics algorithms are validated with reference to simulation studies or limited real-robot experiments only, with no underpinning mathematical model or proof. This approach is inherently limited since simulation or real-robot experiments can only explore small parts of a system's parameter space, and hence provide only weak "inductive" proof of an system's correctness, or reliability. Yet if swarm robotic systems are to find real-world application, especially in safety- or mission-critical applications (Rouff *et al.*, 2003; Truszkowski *et al.*, 2004; Winfield *et al.*, 2006b), we need the strong validation provided by mathematical models of both swarm function and swarm reliability.

This chapter is presented in two parts. In the first section we review approaches for mathematical modelling of collective robotic systems, and outline a macroscopic modelling approach based upon developing a probabilistic finite state machine (PFSM) description of the overall swarm, then expressing the PFSM as a system of differential equations that model the change in the average number of robots in each state, with time. We then illustrate this approach with a case study example of a mathematical model of a wireless connected swarm of mobile robots operated in unbounded space. In the second section we use a modified version of the same case study swarm system to develop failure modes and effects analysis (FMEA), and a reliability model for the swarm. In particular we address the common assumption that swarm systems are automatically scalable and robust and show, for our case study swarm system, that this assumption is incorrect.

1.3.1 Macroscopic Probabilistic Modelling in Swarm Robotics

In recent years probabilistic approaches to modelling swarm robotic systems have been developed and successfully applied. One way to classify these is based on their representation of the swarm and its units.

Microscopic models reproduce each real robot in the targeted system separately, with dedicated — more or less detailed — representations. The *Macroscopic* approach instead models the target swarm robotic system with a single representation, for instance summarising fractions or total numbers of robots in the swarm engaged in specific tasks.

One of the first examples of probabilistic modelling of a swarm of robots at the microscopic level is that proposed by (Martinoli *et al.*, 1999) to study object aggregation; robot's interactions with other robots and the environment are modelled as a series of stochastic events, with probabilities determined by simple geometric considerations and systematic experiments with one or two real robots. The very same microscopic method was applied to the analysis of collaborative stick pulling (Ijspeert *et al.*, 2001).

In general, macroscopic models are more computationally efficient than their microscopic counterparts. One of the fundamental elements of the macroscopic probabilistic model are the Rate Equations, which have been successfully applied to a wide variety of problems in physics, chemistry, biology and the social sciences. For instance, Sumpter and Pratt (Sumpter & Pratt, 2003) developed a general framework for modelling social insect foraging systems with generalised rate functions (differential equations). Sugawara and coworkers (Sugawara & Sano, 1997; Sugawara *et al.*, 1999) first presented a simple macroscopic model for foraging in a group of communicating and non-communicating robots, with analysis under different conditions; for further work see (Sugawara & Watanabe, 2002). (Lerman & Galstyan, 2001; Lerman & Galstyan, 2002b) proposed a more generalised and fundamental contribution to macroscopic modelling in multi-agent systems. In (Lerman & Galstyan, 2002a), they presented a mathematical model of foraging in a homogeneous multi-robot system to understand quantitatively the effects of interference on the performance of the group. In (Lerman *et al.*, 2004), they developed a macroscopic model of collaborative stick-pulling, and the results of the macroscopic model quantitatively agree with both embodied and microscopic simulations. Agassounon and Martinoli used the same approach to capture the dynamics of a robot swarm engaged in collective clustering experiments (Agassounon & Martinoli, 2002).

Rather than using a time-continuous model, Martinoli and coworkers (Martinoli, 2003; Martinoli & Easton, 2003; Martinoli *et al.*, 2004) considered a more fine-grained macroscopic model of collaborative stick-pulling which takes into account more of the individual robot behaviours in the discrete time domain using difference equations. They suggested that time-discrete models are the most appropriate solution for the level of description characterised by logical operators and behavioural states. Similarly, Correll et al. (Correll & Martinoli, 2005) used a macroscopic probabilistic model for analysis of beaconless and beacon-based strategies for a swarm turbine inspection system, and furthermore to find an optimal

collaboration policy minimising the time to completion and the overall energy consumption of the swarm in (Correll & Martinoli, 2006b; Correll & Martinoli, 2006a). In (Correll & Martinoli, 2007), a macroscopic probabilistic model is proposed to analyse self-organised robot aggregation inspired by a study on aggregation in gregarious arthropods.

Both microscopic and macroscopic probabilistic modelling approaches rely on two main common assumptions (Martinoli & Easton, 2003; Martinoli *et al.*, 2004): Firstly, the fulfilment of Markov properties (or semi Markov properties), i.e. the robot's future state depends only on its present state and on how much time it has spent in that state. This assumption is true when robots use reactive control: robots decide on future actions based solely on input from sensors and the action they are currently executing. Therefore the robots can be represented as stochastic Markov processes and the system can be modelled as a probabilistic finite state machine. Secondly, the assumption that the coverage of the arena by the groups of robots is spatially uniform, and the low-level strategies of the robot do not play a critical role on the metric of the system of interest. Indeed, finding an appropriate mathematical description for the transition probabilities is the main challenge in applying both microscopic and macroscopic probabilistic modelling approaches. The second assumption becomes particularly useful for computing the transition probabilities for the robots. In this case, the probabilities of basic events, detecting an object for instance, only depend on geometrical considerations and are given by the ratio of the total extended area of the object related to the total area of the arena where the robots could appear. However, uniform coverage might not always be the case and depends on the environment and the robots' controllers. For example, (Hayes *et al.*, 2000) considered a more complex situation where the distribution of the robot cannot be assumed to be uniform in the arena for an odour plume localisation task. The configuration of the environment and the robots' controllers must be taken into account for probabilistic models.

Despite the success of the above examples, there is little existing work on mathematical analysis of adaptive multi-robot systems in dynamic environments. Lerman and Galstyan (Lerman & Galstyan, 2003; Galstyan & Lerman, 2005; Lerman *et al.*, 2006) extended the macroscopic probabilistic model to study distributed systems composed of adaptive robots that can change their behaviour based on their estimates of the global state of the system. In their study, a group of robots engaged in a puck collecting task need to decide whether to pick up red or green pucks based on observed local information. They claim that the model can be easily extended to other systems in which robots use a history of local observations of the environment as a basis for making decisions about future actions. Liu and Winfield (Liu *et al.*, 2007; Liu, 2008; Liu *et al.*, 2009) developed a macroscopic model for a swarm of foraging robots with adaptation, where the different priorities in behaviour selection and the heterogeneities in individual parameters pose great challenges to modelling. Their model has been successfully used to analyse the performance of the adaptive foraging swarm and further to design optimal parameters for individual controllers.

1.3.1.1 Methodology

For most behaviour-based robotic systems, although the behaviour of a particular robot at a given time is fully determined, the transitions from one state (behaviour) to another could exhibit some probabilistic properties over time within the population of the swarm. The central idea of probabilistic modelling, either micro- or macroscopically, is to describe the experiment as a series of stochastic events and use rate equations to capture the dynamics of these events. A general approach to developing a macroscopic probabilistic model for swarm robotic tasks can be summarised as follows:

step 1. describe the behaviour of the individual robots of the swarm as a finite state machine (FSM);
step 2. transform the FSM into a probabilistic finite state machine (PFSM), describing the swarm at a macroscopic level;
step 3. develop a system of rate equations for each state in the PFSM, to describe the changing average number of robots among states at a macroscopic level;
step 4. measure the state transition probabilities using experiments with one or two real robots, or estimate them using analytical approaches, and then
step 5. solve the system of rate equations.

The controller design for individual robots may take the behaviour based approach introduced by (Brooks, 1986), in which a robot's behaviour is normally determined by its current sensor inputs. A state in the FSM may include one or more of these low level behaviours, and the transition from one state to another will only depend on its current state, rather than historical states; or sometimes on how long the robot has spent in that state. In step 2, the FSM can be transformed into a PFSM by replacing the conditions associated with the transition edges in the FSM with the probabilities that the corresponding conditions are true. In the PFSM, each state represents the average number of robots in that state. The changes of average number of robots in each state of the PFSM over time can then be described using a set of rate equations, either in continuous time or discrete time. Since in a robotics system both analog and digital sensors or actuators are used, we use difference equations (DE) in discrete time to capture the dynamics of the system rather than ordinary differential equations (ODE) in continuous time.

Estimating the transition probabilities for the PFSM (and the DEs) can be a significant challenge, although in some cases these probabilities can be measured by running experiments with one or two real robots; however such an approach is not ideal since the ultimate goal of a mathematical model is analysis and prediction of the effect of individual robot parameters on collective behaviour, rather than trying to match the model to experiments. The transition probabilities are properties of the swarm — they are not design parameters but byproducts of interactions among individuals or between the individual robots and their environment. Using measured values as part of a mathematical model renders the model somewhat less convincing. Moreover, changing environmental conditions or individual parameters

might well then invalidate the measured transition probabilities. In some cases it is not practical to measure all the required transition probabilities because they vary dynamically (time dependent). We advocate geometrical approaches to estimating these probabilities, based on reasonable assumptions. One such assumption is that the distribution of robots within their operational area is, over time, uniform, so the probability that one event is triggered, for example collision with other robots or a bounding wall, could be calculated as the ratio of two areas that the robot covers, on average.

The DEs can be solved analytically or numerically, depending on the complexity of the model. For some tasks, obtaining a direct solution would be possible, for example certain difference equations, in particular linear constant coefficient difference equations, can be solved using z-transforms, see (Martinoli *et al.*, 2004; Agassounon, 2003) for details. For some DEs it is impossible to find a solution using a direct approach such as the z-transform. A numerical approach is then used. In step 5, to obtain numerical solutions for the DEs, certain initial conditions are normally applied, for example the initial number of robots in each state. Another widely used condition is that the total number of robots in each iteration must remain constant.

We now take a simple task as an example to clarify the above approach. Fig. 1.18(a) shows a finite state machine, with only two states, for the robots engaged in a task. Clearly, at each time step, a robot could be in either state A or state B. The robot will transfer from state A to state B whenever the condition a is true. However, it will stay in state B for τ_b seconds and then will move to state A after this time is up. Correspondingly, a probabilistic finite state machine is presented in Fig. 1.18(b). Now the transition between two states in the PFSM is described as a probabilistic event rather than a deterministic event triggered by certain conditions. For example, p in this PFSM represents the probability that condition a is true, which means that a robot will transfer from state A to B with probability p each time step. The probability of transferring from B to state A is 1 but the transfer will be delayed by τ_b seconds, which is the same as in the counterpart FSM. Although the PFSM is derived from the FSM at the individual robot level, it will be used to investigate the changes of number of robots in each state in the swarm by introducing another variable into each state. Here $N_A(k)$ and $N_B(k)$ are used to denote the average number of robots in states A and B respectively. A set of difference equations are then developed to capture the changes of this specific system. Let us consider state A first: the number of robots in state A decreases because of some robots moving to

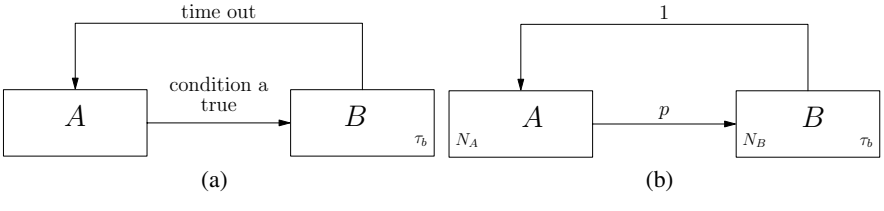


Fig. 1.18 A finite state machine (left) with two states, and its corresponding probabilistic finite state machine (right).

state B, but it increases because of other robots transferring back from state B, thus we have:

$$N_A(k+1) = N_A(k) - pN_A(k) + \Delta_B(k - T_b) \tag{1.92}$$

where $pN_A(k)$ denotes the number of robots moving to state B at time step k, and $\Delta_B(k)$ represents the number of robot transferring to state B at time step k, which is

$$\Delta_B(k+1) = pN_A(k) \tag{1.93}$$

since there is a delay in the transition from state B to A, the number of robots moving to state A from B, at time step k, is equivalent to $\Delta_B(k - T_b)$, where T_b is discretised from τ_b.

Similarly, for state B, we have

$$N_B(k+1) = N_B(k) + pN_A(k) - \Delta_B(k - T_b) \tag{1.94}$$

In fact, since the population of the swarm is constant, say N_0, Eq. (1.94) can be simplified as

$$N_B(k+1) = N_0 - N_A(k+1) \tag{1.95}$$

Assume that the transition probability p has been obtained as outlined above, then given the initial conditions, for example $N_A(0) = N_0$ and $N_B(0) = 0$, the number of robots in each state at any time step, i.e. $N_A(k)$ and $N_B(k)$, can be obtained using a numerical approach. We can therefore use these results to quantitatively analyse the steady-state or dynamic group performance of the system.

1.3.1.2 Case Study: Functionally Modelling a Wireless Connected Swarm of Mobile Robots

Although the simple example above covers the most common type of states and transitions in the macroscopic modelling approach, modelling real systems differs from case to case. The complexity of the model depends very much on the specific task and the metrics of greatest interest. Generally, a number of simplifying assumptions and techniques must be used to model a real system. To illustrate these techniques, a case study is presented in this section. The case study focusses on how to construct a PFSM based on the metrics of interest and how to estimate state transition probabilities in an unbounded environment.

A class of algorithms which make use of local wireless connectivity information alone to achieve swarm aggregation have been developed in (Nembrini *et al.*, 2002; Nembrini, 2005). The basic algorithm, which we refer to as the α-*algorithm*, is very simple. The default behaviour of a robot is forward motion. While moving each robot periodically broadcasts an "I am here" message. The message will of course be received only by those robots that are within wireless range: its neighbours. If the number of a robot's neighbours should fall below the threshold α then it assumes it is moving out of the swarm and will execute a 180° turn. When the number of neighbours rises above α (i.e. when the swarm is regained) the robot then executes

a random turn. This is to avoid the swarm simply collapsing on itself. We say that the swarm is *coherent* if any break in its overall connectivity lasts less than a given time constant. Coherence gives rise to both swarm aggregation and a (coherent) connected *ad hoc* wireless network. In the interests of simplicity we can consider each robot as having three basic behaviours, or states: move *forward* (default); *coherence*, triggered by the number of neighbours falling below α, and *avoidance*, triggered by the robot's collision (proximity) sensor.

The robot updates its connectivity information less frequently than its proximity sensor data. The ratio of connectivity sampling rate to the sampling rate of proximity sensors, which we refer to as *cadence*, is introduced into the basic α-algorithm to prevent the robot from updating its connectivity state too frequently (we need to give the robot time to complete its turn in response to a connection loss, for example, before re-checking its connectivity). By default, the robot will move forward at a fixed velocity. It will update its connectivity state after a certain duration, say T_C (steps), and if it finds the number of connected neighbours has dropped below the threshold α, then it will move into the *coherence* state and execute the U-turn behaviour to try to recover the lost connections; if and when the number of connected neighbours then increases, the robot will execute a random turn. Providing the number of connected neighbours remains at or above α, the robot can lose or gain connections but remain in the *forward* state. Thus, depending upon its connectivity, a robot will either remain in the *forward* state or switch between *forward* and *coherence* states unless it collides with other robots (triggered by the proximity sensor). Such an event will cause the robot to move into the *avoidance* state and execute a collision avoiding turn for time T_A (steps), after which the robot will return to its previous *forward* or *coherence* state. Note that changes in connectivity take precedence over collision avoidance, thus if a change of connectivity is detected while the robot is in the *avoidance* state (i.e. taking avoiding action), the robot will - if required - immediately transition into the appropriate coherence or forward behaviours.

Fig. 1.19 shows the basic robot Finite State Machine (FSM). We reflect the fact that the avoidance behaviours are subsumed within the two top-level states *coherence* and *forward* by showing sub-states $avoidance^C$ and $avoidance^F$. Note that although changes in connectivity take precedence, because the proximity sensor is sampled more frequently than the connectivity (defined above as cadence) collision avoidance is still assured.

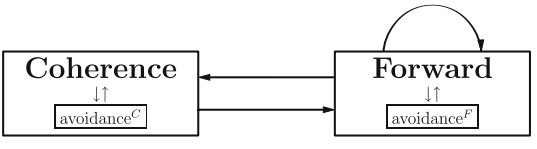


Fig. 1.19 Robot Finite State Machine.

1.3.1.3 A Probabilistic Model of Connectivity

In the α-algorithm $robot_i$ has a number of connected neighbours d_i. Clearly, the range of values for d_i is bounded. The maximum value d_{max} is determined geometrically by the ratio of the areas covered by the wireless sensor range and the avoidance

sensor range; for the robot to remain in the default *forward* state, the lower bound on d_i is α. Now in the α-algorithm when $d_i < \alpha$ the robot moves into the *coherence* state in which it turns back to try and recover the swarm and hence bring d_i back to a value greater than or equal to α. However, the coherence behaviour is not always successful and it is possible for a robot to have fewer than $(\alpha - 1)$ connections. In fact, the robot will continue to try and recover the swarm for values of $0 < d_i < \alpha$. Based on these observations we can now propose a PFSM which completely models the swarm connectivity, as shown in Figs. 1.20 and 1.21.

Fig. 1.20 is, in effect, the simple FSM of Fig. 1.19 expanded to show every possible number of network connections in each of the two states *coherence* and *forward*, together with every possible transition between the states and their probabilities.

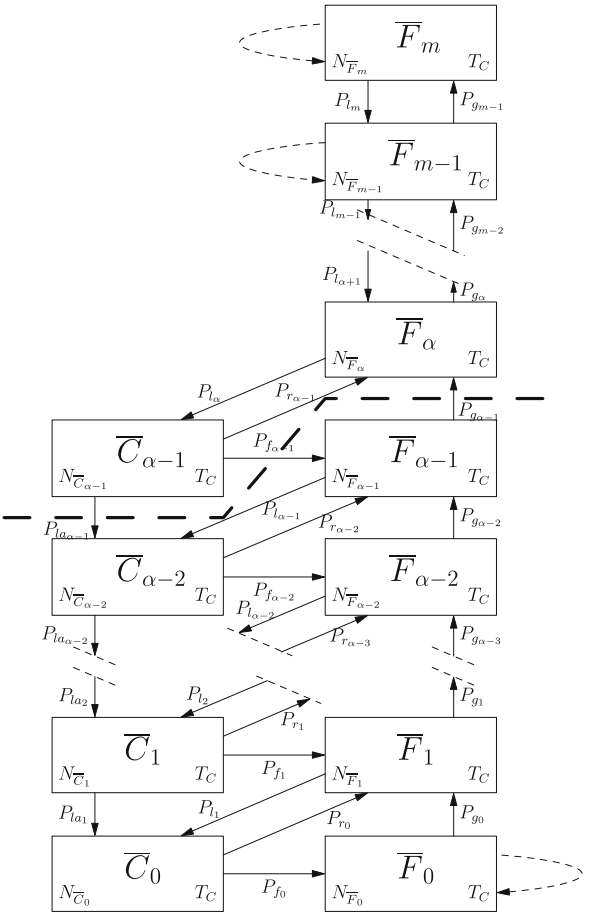


Fig. 1.20 PFSM of the robot controller. $\overline{F}_i$ represents the *forward* state with i connections; $\overline{C}_i$ represents the *coherence* state with i connections. $N_{\overline{C}_i}$ and $N_{\overline{F}_i}$ indicate the average number of robots in corresponding states and T_C indicates the number of time steps spent in each state.

Fig. 1.21 Left: *coherence* state $\overline{C}_i$ expanded to show sub-states A_i^C and C_i. **Right:** *forward* state $\overline{F}_i$ expanded to show sub-states A_i^F and F_i. The average number of robots in each state is shown as $N_\cdot$; T_A is the number of time steps spent in the *avoidance* states A_i^C and A_i^F.

Each of the discrete *forward* states represents a different value of d_i; the $\overline{F}_m$ state is the state with the maximum number of connections d_{max}, the $\overline{F}_{m-1}$ state is the state with $d_{max}-1$ connections, counting down until we reach the $\overline{F}_0$ state with 0 connections; there are a total of $d_{max}+1$ *forward* states, including $\overline{F}_0$. Note that $\overline{F}_0$ is the "lost robot" state representing the failure of the algorithm to maintain the coherence of the swarm. Consider the *forward* state $\overline{F}_\alpha$. The loss of a connection with probability P_{l_α} will cause a transition into the *coherence* state $\overline{C}_{\alpha-1}$. If the action of that state is successful then the robot will transition, after T_C steps and with recovery probability $P_{r_{\alpha-1}}$ back into the $\overline{F}_\alpha$ state. If, on the other hand, the coherence behaviour fails, the robot will move into the $\overline{F}_{\alpha-1}$ state. The likelihood of this is the coherence failure probability $P_{f_{\alpha-1}}$. A loss of connection in each of the forward states $\overline{F}_1 \ldots \overline{F}_\alpha$ will trigger a transition into *coherence* states $\overline{C}_0 \ldots \overline{C}_{\alpha-1}$ respectively.

Fig. 1.21 completes the PFSM by expanding the two states $\overline{C}_i$ and $\overline{F}_i$ into their respective sub-states, again reflecting the structure of the FSM of Fig. 1.19. Fig. 1.21(right) shows that a robot in each of the *forward* states F_i might collide with another robot, with probability P_{a_i}, triggering a transition into its corresponding *avoidance* state A_i^F, returning to the initial *forward* state after T_A steps. Similarly, Fig. 1.21(left) shows that a robot in each of the *coherence* states might also collide with another robot triggering its transition into corresponding *avoidance* states A_i^C, also returning after T_A steps.

1.3.1.4 Geometrical Estimation of Transition Probabilities

Given the PFSM for the $\alpha-$algorithm, a set of difference equations can be derived for the state transitions following the approach outlined in Sect. 1.3.1.1. The full model has a number of transition probabilities, summarised in Table 1.5. Each of these probabilities is conditional on the connectivity status for those robots and therefore impossible to measure in experiments with one or a few real robots.

Here a novel geometry-based approach is developed to estimate these probabilities in our wirelessly connected swarm. Let V denote the normal forward speed of each robot. It follows that the relative speed between two robots varies from 0 to $2V$ and the relative heading varies from 0° to 360°. Consider one of the robots in the swarm, say $robot_i$, with d_i neighbours at time step k. Fig. 1.22 illustrates some of its neighbours, shown as $robot_A$, $robot_B$, $robot_C$ and $robot_D$. Let us assume that $robot_i$ is in either *forward* or *coherence* states, then after one time step (of duration

Table 1.5 State transition probabilities, d_i represents the number of connections for $robot_i$.

probabilities	comments
$P_{a_{d_i}}$	collision with another robot
$P_{l_{d_i}}$	loss of a connection in *forward* state
$P_{g_{d_i}}$	gain of a connection
$P_{r_{d_i}}$	recovery of a connection
$P_{f_{d_i}}$	failure to recover a connection
$P_{la_{d_i}}$	loss of a connection in *coherence* state

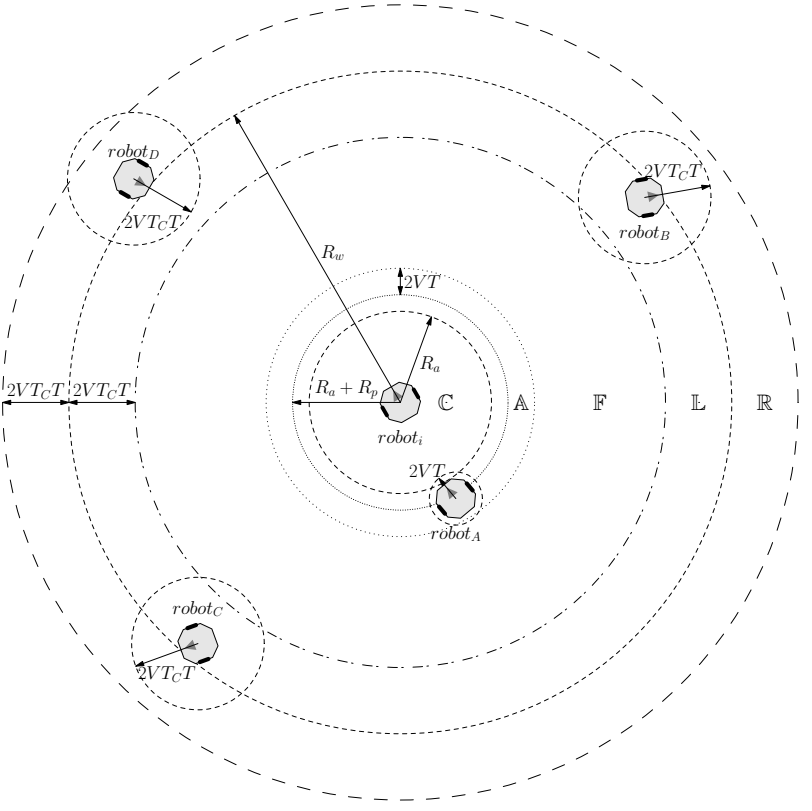


Fig. 1.22 *Robot*$_i$ and its neighbours. Robots are marked with filled circles. Each robot has a communication range R_w and avoidance radius R_a, R_p denotes the physical size (radius) of the robot. $\mathbb{C}$, $\mathbb{A}$, $\mathbb{F}$, $\mathbb{L}$ and $\mathbb{R}$ in the figure represent the collision, avoidance, forward, connection loss and connection recovery areas respectively; each is an annular region bounded within two circles with the same origin.

T), each of its neighbours will move a distance from 0 to $2VT$. It is clear that only the robots close enough will have a chance of moving into $robot_i$'s collision area (within radius R_a, marked $\mathbb{C}$ in Fig. 1.22), and thus drive $robot_i$ to change to state *avoidance*. For instance, as shown in Fig. 1.22, $robot_A$ may possibly trigger $robot_i$'s avoidance sensor next time step while $robot_B$, $robot_C$ and $robot_D$ cannot. Similarly, after T_C steps, $robot_C$ located in area $\mathbb{L}$ will possibly move out of $robot_i$'s communication range resulting in $robot_i$ losing one connection, and $robot_D$ located in area $\mathbb{R}$ might move into $robot_i$'s communication range, with some probability, in which case $robot_i$ will gain one connection at step $k+T_C$. However, $robot_B$ located in area $\mathbb{F}$ can neither trigger $robot_i$'s avoidance sensor nor cause a change in the number of its connected neighbours. Thus, in order to estimate state transition probabilities, we need only consider situations where neighbouring robots fall within the annular regions in Fig. 1.22: $\mathbb{A}$, in which a collision might occur; $\mathbb{L}$, in which a connection loss might occur; or $\mathbb{R}$, in which a connection recovery might occur. A detailed derivation of each of these probabilities is given in (Winfield *et al.*, 2008).

1.3.1.5 Running the Macroscopic Model

We now run the macroscopic model with state transition probabilities estimated using the geometrical approach. Fig. 1.23 shows the average number of robots in states *forward*, *coherence* and *avoidance*, in which we merge states A^C and A^F from the sub-PFSMs in Fig. 1.21, plotted against connectivity. The left-hand plots show the results collected from a sensor-based simulation using Player/Stage, while the right-hand side plots show the results from the PFSM model run with the estimated state transition probabilities. The total average number of robots, summing all states, is also plotted as the topmost curve in each graph.

First we note that the PFSM model generates the same "bell" shaped curves as the simulation, and for all three values of α the peak occurs at or very close to the same connectivity value. The PFSM model for $\alpha = 5$ somewhat underestimates the number of robots in all three states and also shows a longer "tail" of robots with high connectivity values than is measured from simulation; however, the model shows reasonable agreement at very low connectivity values, especially in predicting 'lost' robots (with connectivity of zero). At $\alpha = 10$ the macroscopic model again shows a longer tail of high connectivity robots than the simulation; also evident is the same overestimate in the number of robots in the *forward* state at connectivity values below α. The overestimate in *forward* robots is even more pronounced at $\alpha = 15$. We also see that the 'lost' and very low connectivity robots are not seen in the model for $\alpha = 10$ and $\alpha = 15$. In all three pairs of results the greatest discrepancy between the macroscopic model and the simulation is in robots in the *forward* state. In contrast, the macroscopic model shows much stronger agreement with simulation for the number of robots in *coherence* and *avoidance* states. Given the simplifying assumptions for constructing of the PFSM and estimating of the transition probabilities, it is perhaps surprising that the macroscopic model does generate such plausible results for the swarm connectivity structure. For details of low-level robot parameters and full discussion of the modelling assumptions refer to (Winfield *et al.*, 2008).

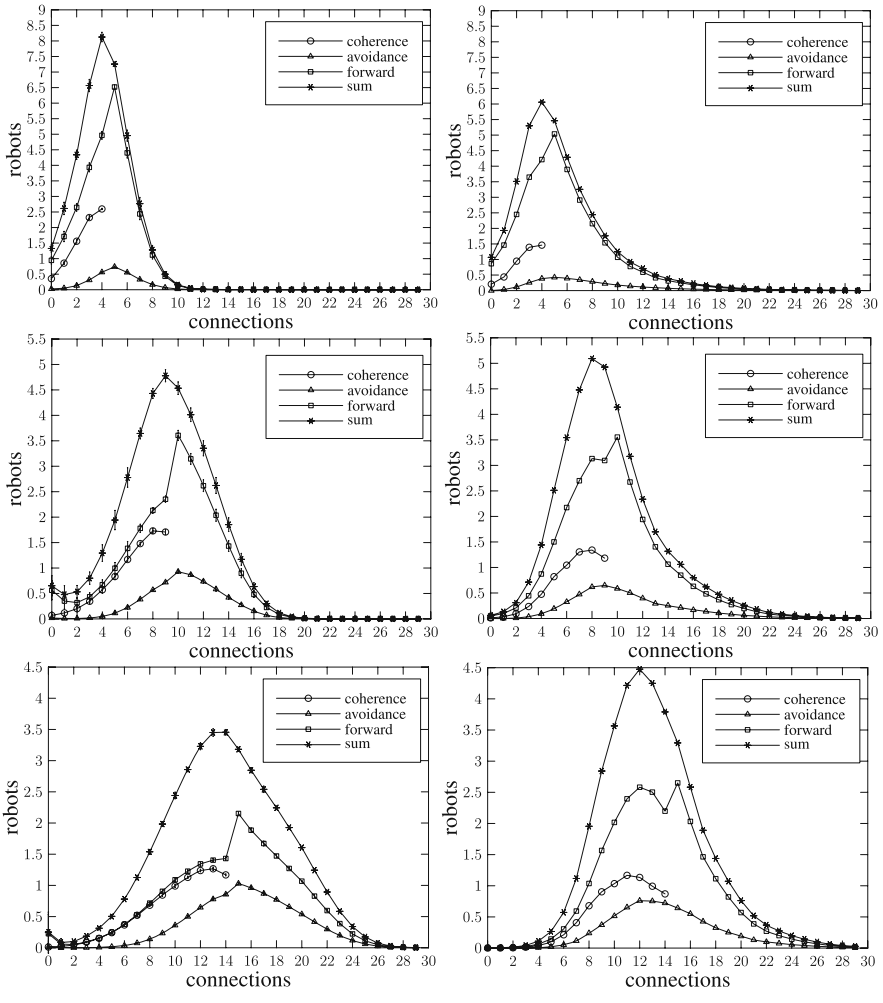


Fig. 1.23 Number of robots in state *coherence*, *forward* and *avoidance* plotted against number of neighbours (connectivity). From top to bottom, $\alpha = 5,\ 10$ and 15. **Left:** Player/Stage simulation, average of 10 runs, each simulation lasts for 10000 seconds. **Right:** Macroscopic model using geometrically estimated probabilities.

1.3.2 Reliability Modelling of Swarm Robotic Systems

Research papers in Swarm Robotics frequently assert that swarm robotic systems are both scalable and robust. We accept the defining criteria for swarm robotics set out in (Dorigo & Şahin, 2004), which include importantly "local and limited sensing and communication", and we note that these criteria also state that swarm robotics studies should "aim for scalability". The fact that individual robots in the swarm make decisions based only on local sensing and communication is assumed to lead

naturally to swarms that will scale to very large numbers of robots. The high degree of parallelism in robot swarms, which typically consist of homogeneous robots, is assumed to lead to a high level of robustness and dependability. It is indeed true that robot swarms can exhibit an unusual level of tolerance to failure of individual robots, or external threats, when compared with conventionally engineered distributed systems, but it is not safe to assume that scalability and robustness are automatically properties of all (or any) swarm systems. It is surprising therefore that, in the field of swarm robotics, there has been very little systematic study of dependability and fault tolerance; but see (Winfield *et al.*, 2005; Winfield & Nembrini, 2006; Christensen *et al.*, 2009).

In this section we develop a reliability model for a case study swarm of robots that exhibit emergent, or self-organised, swarm taxis. We first undertake a failure modes and effects analysis (FMEA) for the case study swarm, then show that we need to model this swarm — from a reliability perspective — as a k-out-of-N system. We then extend the k-out-of-N reliability model to take account of worst-case partial robot failures and swarm scaling properties, introducing the new concept of swarm self-repair. We conclude with a model of reliability as a function of swarm size.

1.3.2.1 Case Study: Reliability Modelling Emergent Swarm Taxis

We modify the α-algorithm described above in Sect. 1.3.1.2 in two ways. Firstly, the *coherence* behaviour is achieved not by making a 180° turn when the number of a robot's connected neighbours falls below α, but instead by timing the duration since the robot last made an avoidance manoeuvre and if that value exceeds a given threshold ω, the robot turns towards its estimate of the centre of the swarm; an estimate based on readings from the ring of infrared proximity sensors around the robot's body. To increase the distance at which robots can sense each other, and also to enable robots to distinguish between robots and ambient infra-red, each of the robots is equipped with infra-red emitters that flash at 80 Hz. By sampling the sensors at 400 Hz and passing the data through a bandpass filter the 80 Hz flashing is reliably detected. Each robot can then estimate the direction of the local centre of the swarm based on which of its sensors detect a flashing signal from other robots. For the results obtained from hardware trials with real robots reported below we set $\omega = 2.5$ s; ω (like α) controls the overall swarm density.

Secondly, we add an additional "beacon" sensor to each robot. The beacon sensor is a very simple sensor, in that it is unable to detect the range and bearing of the remote beacon and has only a two-state output: on = illuminated or off = not-illuminated. An important feature of the physical placement of the beacon sensor is that it can be occluded by other robots, thus those robots that have a direct line-of-sight to the beacon will have beacon sensors illuminated, and those robots that are in the shadow of other robots will have beacon sensors not-illuminated. This means that for a typical swarm only the robots on or close to the leading edge of the swarm (with respect to the beacon) will have illuminated beacon sensors. In our experimental trials we use an IR beacon and make use of the same IR sensors that

are used for both short-range communication and collision avoidance, for beacon sensing.

We then introduce a simple symmetry breaking mechanism. Each robot has short-range avoidance sensors (for sensing collisions with obstacles or, more typically, other robots). We set the avoid sensor radius for those robots that are illuminated by the beacon to be slightly larger than the avoid sensor radius for those robots in the shadow of other robots. This simple mechanism results in a net swarm movement (taxis) toward the beacon. Note that the swarm taxis is an emergent property of the swarm: with a simple two-state beacon sensor a single robot cannot sense the direction of the beacon, and even with the symmetry breaking mechanism two or three robots are not enough to give rise to emergent swarm taxis; experimentally we find that swarm taxis requires at least five robots. This is important to our case study as we are interested in determining the reliability of a swarm with emergent swarm behaviours. For a detailed analysis of the swarm taxis behaviour see (Bjerknes *et al.*, 2007).

1.3.2.2 Failure Modes and Effects Analysis

In this section we undertake a Failure Mode and Effect Analysis (FMEA) for our case study robot swarm. The methodology is straightforward, see (Dailey, 2004). We attempt to identify all of the possible hazards, which could be faults in robots or robot sub-systems (internal hazards), or environmental disturbances (external hazards). Then, in each case, we analyse the effect of the hazard on each of the overall swarm behaviours. Thus, we build up a picture of the tolerance of the swarm to both types of hazard and begin to understand which hazards are the most serious in terms of compromising the overall desired swarm behaviours. FMEA is, at this stage, essentially qualitative. Here we consider only internal hazards; external hazards (i.e. communications noise) were investigated in (Nembrini, 2005).

First we identify the internal hazards. In keeping with the swarm intelligence paradigm our robot swarm contains no system-wide components or structures, thus the only internal hazards that can occur are faults in individual robots. Since, in our case, the robots of the swarm are all identical, then (internal) hazards analysis requires us to consider only the faults that could occur in one or more individual robots, and then consider their effect on the overall swarm behaviours. Table 1.6 identifies the fault conditions for an individual robot.

Table 1.6 Internal Hazards for a single robot

Hazard	Description
H_1	All systems failure
H_2	Coherence sensor(s) failure
H_3	Avoidance sensor(s) failure
H_4	Beacon sensor failure
H_5	Motor sub-system failure

Table 1.6 makes the assumption that failures of robot sub-systems can occur independently. This is a reasonable assumption, given that our mobile robots are in reality an assembly of complex but relatively self-contained sub-systems. Hazard H_1 represents a total failure of the robot; failure of the robot's power supply would, for instance, bring about this terminal condition. Hazards H_2, H_3 and H_4 represent a failure of the robot's communication, avoidance and beacon sensing functions respectively. Finally hazard H_5 motor failure, covers the possibility of mechanical or motion-controller failure in one or both of the motors in our differential drive mobile robot, such that the robot is either unable to move at all or can only turn on the spot (which from an overall swarm point of view amounts to the same thing).

Let us now consider the effects of each the hazards enumerated above on the overall swarm behaviours. We will consider here the effect on the overall swarm of the hazard occuring in one or *a small number* of the individuals in the swarm. Of course the question of how many is a small number in this context is important, and will return to the question of what proportion of robots need to fail in order to *seriously* compromise the desired overall swarm behaviours.

Hazard H_1: total systems failure. Complete failure of one or a small number of robots caused, for instance, by power failure will clearly render the robot(s) stationary and inactive. They will be wirelessly disconnected from the swarm and will be treated, by the swarm, as static obstacles to be avoided. Ironically, given that this is the most serious failure at the level of an individual robot, it is relatively harmless as far as the overall swarm is concerned. Apart from the loss of the failed robots from the swarm, none of the overall swarm behaviours are compromised by this hazard - however, we can expect the swarm to be temporarily slowed by the obstacle represented by the failed robot. We therefore label this effect E_1, with an upper-case E to denote that it is a potentially serious fault.

Hazard H_2: coherence sensor failure. Failure of the coherence sensors sub-system in one or a small number of mobile robots means that those robots cannot sense the centre of the swarm. Given that basic swarm aggregation depends upon the coherence mechanism, then robots with fault H_2 might become physically lost to the swarm and wander off at random. As far as the swarm is concerned these robots simply become (moving) obstacles to be avoided. The overall swarm behaviours are, however, essentially unaffected. This hazard has, therefore, a relatively benign effect, except of course that the failed robots remain mobile within the environment and - in some circumstances - this may be undesirable. We label this effect e_2, with a lower-case e to denote that it is a non-serious fault.

Hazard H_3: avoidance sensor failure. Failure of the avoidance sensor(s) in one or a small number of robots has little effect on overall swarm behaviour. A single robot with failed avoidance sensors will be avoided by the other robots in the swarm and hence have no overall effect. In the unlikely event that two or more robots with

Table 1.7 Summary of Failure Modes and Effects

Swarm Behaviour	H_1	H_2	H_3	H_4	H_5
Aggregation	-	e_2	-	-	-
Ad hoc network	-	e_2	-	-	-
Beacon taxis	E_1	e_2	-	-	E_5

failed avoidance sensors collide with each other then physical damage might result from such collisions, but the overall swarm behaviours remain unaffected.

Hazard H_4: beacon sensor failure. Failure of the beacon sensor in one or a small number of robots has a practically undetectable effect on the overall swarm behaviour. This is because the emergent beacon taxis behaviour results from the symmetry breaking mechanism outlined above. Since a differential is created between two substantive parts of the swarm, the effect of one or a small number of robots with failed beacon sensors is negligible.

Hazard H_5: motor failure. The effect of motor failure in a single robot, or small number of robots, is interesting. Robot(s) with fault H_5 become – in effect – stationary but, given that their signalling and other sensing systems continue to function, they remain within the swarm. These robots continue to fully contribute to the swarm aggregation and *ad hoc* network emergent behaviours. It is only when the swarm needs to physically translate its position. i.e. for the beacon taxis behaviour, that hazard H_5 becomes a serious problem. In this case robots with motor failure will have the effect of physically anchoring the swarm, either impeding or, at worst, actually preventing the swarm from moving toward its target. This is a potentially serious hazard since one or a small number of robots with motor failure could seriously compromise the desired swarm-taxis behaviour. We shall label this fault effect as E_5, with an upper-case E to denote that it is potentially serious.

To summarise, Table 1.7 shows the swarm fault effects, as defined above, generated by one or a small number of robots with hazards $H_1 ... H_5$, for each emergent swarm behaviour. Table 1.7 clearly shows that the serious swarm failure effects E_1 and E_5 occur in only 2 out of 15 possible combinations of robot hazard and swarm behaviour; 10 out of the 15 hazard scenarios have no effect at all on swarm behaviour, and the remaining 3 have only minor, non-serious, effects.

1.3.2.3 The k-out-of-N Reliability Model

The purpose of a reliability model is to enable the estimation of overall system reliability, given the (known) reliability of individual components of the system, see (Elsayed, 1996). Reliability R is defined as the probability that the system will operate without failure, thus the unreliability (probability of failure) of the system, $P_f = 1 - R$. In our case the overall system is the robot swarm and its components are the individual robots of the swarm.

From a reliability modelling perspective a swarm of robots is clearly a parallel system of N components (robots). If the robots are independent, with equal probability of failure p, then the system probability of failure is clearly the product of robot probabilities of failure. Thus, for identical robots, $R = 1 - p^N$. p can be estimated using a classical reliability block diagram approach on the individual sub-systems of the robot; since the individual robot does not internally employ parallelism or redundancy then its reliability will be modelled as a series system, giving p less than the worst sub-system in the robot, which is most likely to be its motor drive system.

However, this simplistic modelling approach makes a serious and incorrect assumption, which is that the overall system remains fully operational if as few as one of its components remains operational. This is certainly not true of our case study swarm. The desired emergent swarm behaviours require the interaction of multiple robots; our swarm beacon taxis behaviour is a dramatic example: with one robot only the behaviour simply cannot emerge. It is a general characteristic of swarm robotic systems that the desired overall swarm behaviours are not manifest with just one or a very small number of robots. However, the question of how many (or few) robots are needed in order to guarantee a required emergent behaviour in a particular swarm and for a particular behaviour is often not straightforward.

Thus, from a reliability perspective, we need to consider the swarm as a k-out-of-N:G system. That is, a system of N parallel elements which requires that at least k of these elements are operational ("good") for the overall system to function correctly. In a swarm of N robots, if more than $N - k$ fail, the self-organised functionality of the overall swarm will be compromised.

In a k-out-of-N:G system, the probability that *at least* k out of N robots are working at a given time t is given by:

$$P(k,N,t) = \sum_{i=k}^{N} \binom{N}{i} (e^{-t\lambda})^i (1 - e^{-t\lambda})^{N-i} \tag{1.96}$$

where $\lambda = \frac{1}{MTBF}$, (Kuo & Zuo, 2002). MTBF is the mean time before failure of an individual robot.

Based on Eq. (1.96) we can now plot swarm reliability against time for our case study swarm. Experimental trials indicate that at least five robots have to be working in order for the emergent swarm taxis behaviour to work properly. Thus, we can model our swarm as a 5-out-of-N system. Consider now the individual robots' MTBF. Carlson et al. tracked failure data for 13 robots by three different manufacturers over a period of two years. They found the MTBF to be eight hours (Carlson & Murphy, 2003). Experiments with the e-pucks used in our experimental trials might suggest that their failure rate might be higher (because of poor design of the e-puck battery connector). However, as no systematic data is available, the value reported by Carlson et al. will be used here. Fig. 1.24 (top) plots Eq. (1.96) for a swarm of ten robots, and shows that the swarm reliability starts to decline rapidly after 100 minutes of operation.

Fig. 1.24 (bottom) plots the reliability of the same swarm of ten robots, with the same values for k and MTBF, against the distance the swarm will travel

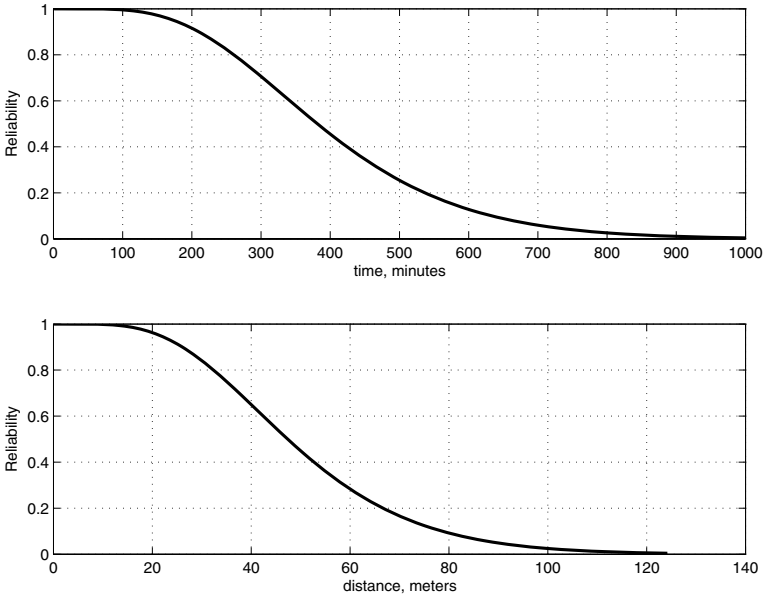


Fig. 1.24 Top: The reliability of a robot swarm modelled as a k-out-of-N system, with $k = 5$, swarm size $N = 10$ robots and MTBF = 480 m. Bottom: Reliability of the same swarm as a function of distance travelled, based on a measured mean swarm velocity of 12.4 cm. per min. for a swarm of 10 robots.

(the emergent swarm taxis behaviour) based on a measure mean swarm velocity of 12.4 cm per minute for a swarm of 10 robots. Although providing some insight, the reliability assessments based on the k-out-of-N model here fail to take into account two important factors. Firstly, each robot that fails is likely - depending on the exact nature of that failure - to slow down the swarm; if the failed robot(s) are immobile then the swarm will slow down until it "escapes" from the failed robots, leaving them behind. Secondly, the swarm velocity might then change after then failed robot(s) have been left behind, typically a smaller swarm (of at least 5 robots) will have a higher swarm taxis velocity. We now analyse these factors in more detail in order to improve the swarm reliability model.

1.3.2.4 Swarm Self-repair

We now introduce the concept of *swarm self-repair*. Consider the case-study swarm and its failure modes and effects analysis outlined above in Sect. 1.3.2.2. We have conducted a series of trials of the emergent beacon swarm-taxis algorithm, using 10 e-puck robots (Mondada *et al.*, 2009), in which we artificially introduce different failure modes into one or more robots of the swarm. Our trials broadly confirm the FMEA and demonstrate that, while all failure modes have the effect of slowing down swarm progress toward the beacon, the swarm is tolerant to the simultaneous

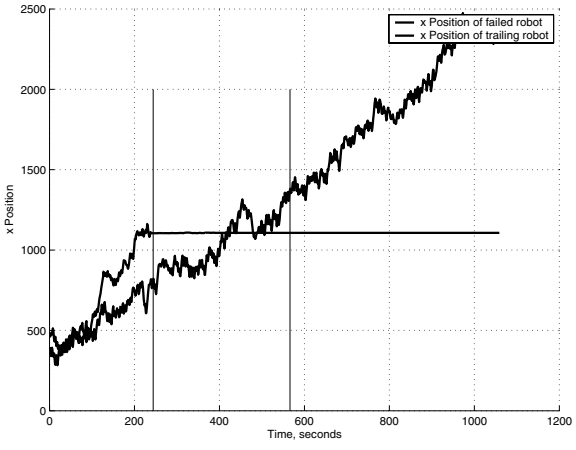


Fig. 1.25 Hardware trials using 10 e-puck robots: single robot complete failure H_1, swarm self-repair time. Two robots are tracked: the failed robot and the trailing robot from the rest of the swarm. At about 250 s. a single robot on the leading edge of the swarm experiences failure H_1; at about 580 s. the trailing robot escapes the failed robot.

(i.e. worst case) failure of more than one robot. Furthermore, we notice two different categories of effect on the overall swarm: (1) sensor failures $H_2...H_4$ which slow down progress of the swarm, but the whole swarm reaches the beacon and (2) motor failures H_1 and H_5 which hold back progress of the swarm until the swarm breaks free of the failed robots; for a detailed analysis of these results see (Bjerknes, in press). Consider the second, and more serious category, which gives rise to the notion of swarm self-repair.

Refer to Figs. 1.25 and 1.26. We define swarm self-repair time as the time between (simultaneous) motor failure of one (or more) robots and the point at which the trailing robot in the rest of the swarm escapes the influence of the failed robot(s). This is a useful metric because it varies with both the type of robot motor failure (H_1 or H_5) and the number of robots. Table 1.8 lists the measured swarm self-repair times for one and two simultaneous failures for failure modes H_1 (robot completely failed) and worst case H_5 (robot partially failed - motors failed but electronics still operational). For comparison the table also shows a baseline notional self-repair time: the time the swarm would take to leave behind a failed robot if that robot failure did not slow down the swarm.

1.3.2.5 Swarm Scaling and Reliability

We have argued that in the k-out-of-N reliability model above, the minimum value of $k = 5$ because the swarm taxis property is present even with as few as 5 robots. For $N = 10$ robots and an MTBF of 8 hours, this reliability model suggests that the swarm will become unreliable after approximately 100 minutes. While it is clear that we can increase the swarm reliability by increasing the individual robots' MTBF,

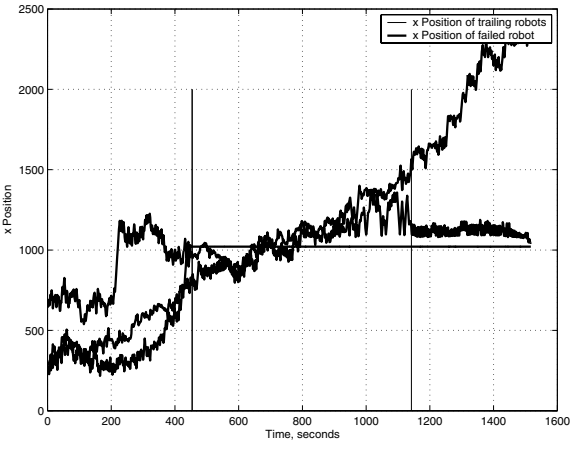


Fig. 1.26 Hardware trials using 10 e-puck robots: single robot partial failure H_5, swarm self-repair time. Three robots are tracked: the failed robot, the trailing robot from the rest of the swarm and a third healthy robot left behind with the failed robot. At about 450 s. a single robot on the leading edge experiences failure H_5; at about 1150 s. the trailing robot of escapes the failed robot. Observe that one healthy robot is also left behind by the swarm and only 8 robots proceed to the beacon.

Table 1.8 Mean swarm self-repair times for the case study swarm of $N = 10$ e-puck robots. Ten runs for each case. *In the final case of two partially failed robots, in only six runs did the swarm reach the beacon.

Case	Mean (s)	Std. Dev. (s)
Baseline (no penalty)	328	174
One failed robot H_1	387	132
Two failed robots H_1	453	172
One failed robot H_5	879	417
Two failed robots H_5	1279	see note*

can we also make the swarm more reliable by increasing swarm size? At first it might seem plausible to suggest that the increased redundancy in a larger swarm would maintain reliability for a longer period. One may even be led to believe that the swarm could be made reliable for an arbitrarily long time, given a sufficiently large number of robots. This is not correct, and we now combine a model of swarm self-repair with the k-out-of-N model to determine the maximum upper size for our case-study swarm.

Consider the argument informally. When a swarm is larger it will take longer to self-repair than a smaller swarm. There are two reasons for this. Firstly, it is a property of our case study swarm that the swarm taxis velocity reduces with increasing swarm size. Secondly, the swarm is physically larger and must move a longer distance before it is fully self-repaired. Thus the self-repair rate will *remain constant*

with increased swarm size. However, for a given robot MTBF, the swarm failure-rate will *increase* for larger swarms. It is unavoidable that at some point the failure rate will overtake the self-repair rate of the swarm, and the swarm will come to a complete halt - the desired emergent swarm-taxis property will fail. In fact a swarm of sufficient size would die under its own weight, so to speak, before it has even started to move.

We now estimate the values of k and self-repair time t_s as a function of N. Thus the k-out-of-N model written as Eq. (1.96) will be modified to take the time to self-repair into account.

The value of k

In experimental tests it is clear that, for complete failures H_1, two out of ten robots could fail without permanently damaging the swarm. The swarm would always self-repair. The cases with partial failure H_5 fared less well. When one out of ten robots failed, the swarm did always self repair, even though a functioning robot might occasionally become stuck with the failed robot. But when two out of ten robots failed, the swarm would suffer a complete breakdown in four out of ten cases, and in the remaining six cases, as many as three healthy robots stayed behind with the failed robots.

Based on this the value of k will be conservatively estimated as 90% of N for a k-out-of-N:G system. In other words, when the swarm has ten percent failed robots or less it will be assumed that it can self repair. Arguably, this may not hold true for larger swarms - the empirical evidence is limited to swarms with ten robots. But this is our best estimate from the evidence available.

The value of t_s

We know from an analysis of the scaling properties of our case study swarm (Bjerknes, in press), that swarm-taxis velocity v as a function of N follows this relationship:

$$v(N) = CN^{-\frac{1}{2}} \tag{1.97}$$

Where C is a scaling constant. Thus larger swarms move more slowly. Note, as stated already, that the minimum value of swarm size N for the swarm to exhibit swarm taxis is 5, thus Eqn. 1.97 is not valid for $N < 5$.

Clearly, the diameter d, of the swarm will increase with swarm size.

$$d(N) = D\sqrt{N} \tag{1.98}$$

Where D is the density constant for the swarm.

Since a robot can fail anywhere within the swarm: on the leading edge, in the middle of the swarm or at the trailing edge, the average distance that the swarm needs to move before it has moved away from the failed robot will be half the diameter, $\frac{d}{2}$. Thus the self-repair time becomes $t_s = \frac{d}{2v}$.

Thus,

$$t_s(N) = \frac{D\sqrt{N}}{2C\frac{1}{\sqrt{N}}} \tag{1.99}$$

Which simplifies to

$$t_s(N) = \frac{D}{2C}N \tag{1.100}$$

Eq. (1.100) is important as it demonstrates that the self-repair time increases linearly with N. Based on this equation it is now possible to introduce a new constant for a given swarm, namely the self-repair-time-constant. Let this constant have the symbol S for Self-repair, where $S = \frac{D}{2C}$. Now we have established that S is linear with N, we can determine its value experimentally. For a swarm with ten robots with one partially failed robot the mean self-repair time was found to be 879 s (see Table 1.8). This was for a case with ten robots, so the self-repair constant for our case study swarm, for the worst case partial failures H_5, then becomes $S = \frac{879}{10} = 87.9$.

Using the estimated values for k and t_s and the k-out-of-N reliability model we can now plot swarm reliability against swarm size N.

Fig. 1.27 shows that with an MTBF of 8 hours, a swarm with as few as 40 robots will have a reliability of only 0.5. This reliability model is based on a number of assumptions (including, for instance, a circular swarm morphology that remains constant with increasing swarm size), together with experimentally estimated constants. Notwithstanding these assumptions and estimates, the main idea that the self-repair-time increases with larger swarms is well argued based on the experiments presented here. Even though the actual reliability for a given swarm size may be a somewhat

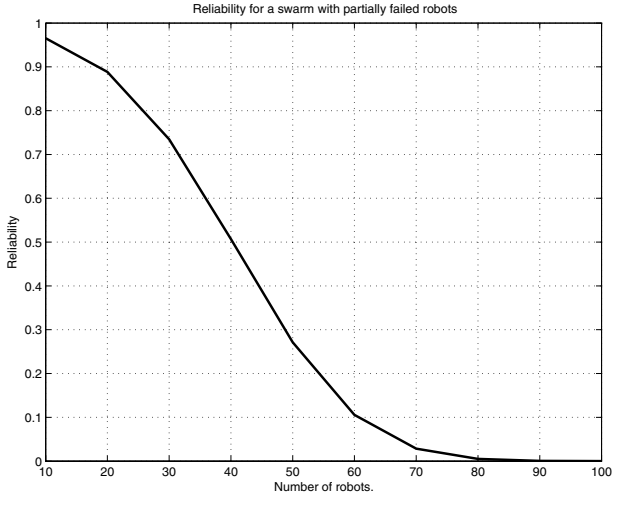


Fig. 1.27 Reliability of the case study swarm as a function of swarm size, based on a k-out-of-N reliability model and assuming worst case partially failed robots H_5; $k = 0.9N$, self-repair-time-constant $S = 87.9$ and robot MTBF 8 h.

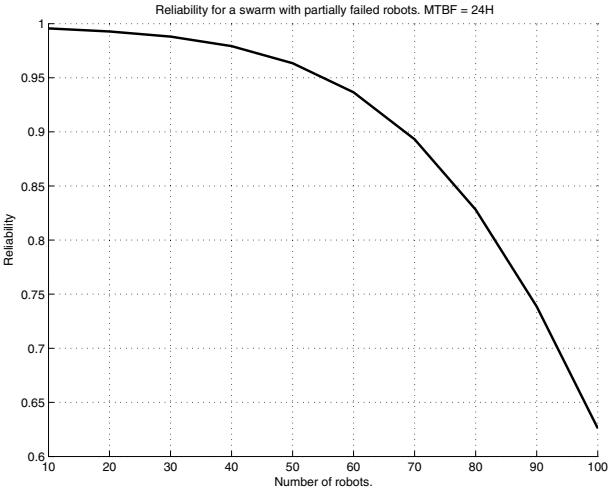


Fig. 1.28 Reliability of the case study swarm as a function of swarm size, based on a k-out-of-N reliability model and assuming worst case partially failed robots H_5; $k = 0.9N$, self-repair-time-constant $S = 87.9$ and robot MTBF 24 h.

higher or lower than the k-out-of-N model suggests, it is undoubtedly true that our case study swarm will eventually become non-functioning with increasing size, and that this occurs at a much lower swarm size than one might intuitively expect. Clearly we can significantly improve swarm reliability by increasing robot MTBF, as shown in Fig. 1.28, for a four-fold improvement of individual robot MTBF to 24 h.

1.3.3 Concluding Discussion

In Sect. 1.3.1 we have reviewed approaches for mathematical modelling of collective robotic systems, and outlined a macroscopic modelling approach based upon developing a probabilistic finite state machine (PFSM) description of the overall swarm, then expressing the PFSM as a system of differential equations that model the change in the average number of robots in each state, with time. We then illustrated this approach with a case study example of a mathematical model of a wireless connected swarm of mobile robots operated in unbounded space. The model demonstrates a novel robot-centric approach for estimating state transition probabilities. There is no doubt that proving the correctness of collective systems requires mathematical modelling, and we believe that the macroscopic probabilistic approach outlined here provides us with a powerful modelling technique.

Consider now the applicability of this functional modelling approach to symbiotic multi-robot organisms — the subject of this volume. Sect. 1.1, Table 1.4 identifies swarm-mode, organism-mode and the transitions between these modes.

The PFSM modelling approach can be applied in swarm-mode during searching/resources localisation (i.e. foraging) and during the transition from swarm to organism (morphogenesis). We have already developed a PFSM model for adaptive foraging, see (Liu *et al.*, 2007; Liu, 2008; Liu *et al.*, 2009). For morphogenesis the model will be used to predict the time robots take to self-assemble into a 2D planar structure, i.e. the energy cost of this phase. The aim here is to use the model to improve the behaviour of individual robots and hence optimise energy consumption and achieve faster transition from swarm-mode to organism-mode.

Sect. 1.3.2 has shown that the frequent assumption that collective systems based upon the swarm intelligence paradigm are automatically scalable and robust is unsafe. By undertaking a reliability analysis of a swarm system in which the desired swarm properties are truly emergent (self-organising), we have shown that, for the worst-case partial-failures of individual robots, overall system reliability falls very rapidly with increasing swarm size. When compared with conventionally designed distributed systems our case study swarm does exhibit an unusual level of tolerance to failure of individual robots (and indeed a self-repair mechanism) that — in a sense — comes for free with the swarm intelligence paradigm, but that fault tolerance does not scale well. Of course our case-study swarm has no mechanisms for actively identifying and compensating for partially-failed robots, which leads to the conclusion that scaling collective systems from tens to hundreds or thousands of robots might not be achievable without such mechanisms, i.e. distributed artificial immune systems (see Sect. 4.4).

Consider now the applicability of the reliability modelling approach to symbiotic multi-robot organisms. We would argue that the multi-state k-out-of-N approach can be applied to the searching/resource localisation (foraging) task during swarm-mode. It is likely that foraging will require swarm aggregation/taxis so that a subset of robots moves together to an object of interest thus, depending on the algorithm design, the reliability model given in this chapter may apply in some modified form. The k-out-of-N reliability model might also be applicable to organism-mode since, in principle, we might expect a multi-robot organism comprising N robots to continue to function if at least k of these robots are functioning. Clearly the failure modes and effects analysis (FMEA) will be more complex since the consequences of a failed robot, and the type of failure, will depend on the position and function of that robot within the organism.