13

Bio-Inspired Networking — Self-Organizing Networked Embedded Systems

Falko Dressler

Autonomic Networking Group, Dept. of Computer Science 7, University of Erlangen, Martensstr. 3, 91058 Erlangen, Germany.
dressler@informatik.uni-erlangen.de

Summary. The turn to nature has brought us many unforeseen great concepts and solutions. This course seems to hold on for many research domains. In this article, we study the applicability of biological mechanisms and techniques in the domain of communications. In particular, we study the behavior and the challenges in networked embedded systems that are meant to self-organize in large groups of nodes. Application examples include wireless sensor networks and sensor/actuator networks. Based on a review of the needs and requirements in such networks, we study selected bio-inspired networking approaches that claim to outperform other methods in specific domains. We study mechanisms in swarm intelligence, the artificial immune system, and approaches based on investigations on the cellular signaling pathways. As a major conclusion, we derive that bio-inspired networking techniques do have advantages compared to engineering methods. Nevertheless, selection and employment must be done carefully to achieve the desired performance gains.

Key words: bio-inspired networking, autonomic networking, self-organization, networked embedded systems, bio-inspired algorithms

13.1 Introduction

The proliferation of wireless sensor networks (WSN) and similar ad hoc networks based on huge amounts of spontaneously interacting nodes is changing the world of telecommunications. In addition to the increasing number of communicating nodes, node mobility is an issue as addressed, for example, in sensor/actuator networks (SANET). Previously, controllability and determinism were the keywords during protocol development and network research. Based on the primary objectives of WSN, nodes communicate using a radio interface, they are battery-driven, small, and cover only few resources. Therefore, new key factors have been identified for developing communication methods. Above all, scalability of the employed mechanisms is required.

Researchers anticipate self-organization methods as the general solution to the depicted communication issues in WSN and SANET. Centralized

R.P. Würtz (ed.), *Organic Computing*. Understanding Complex Systems,
doi: 10.1007/978-3-540-77657-4_13,

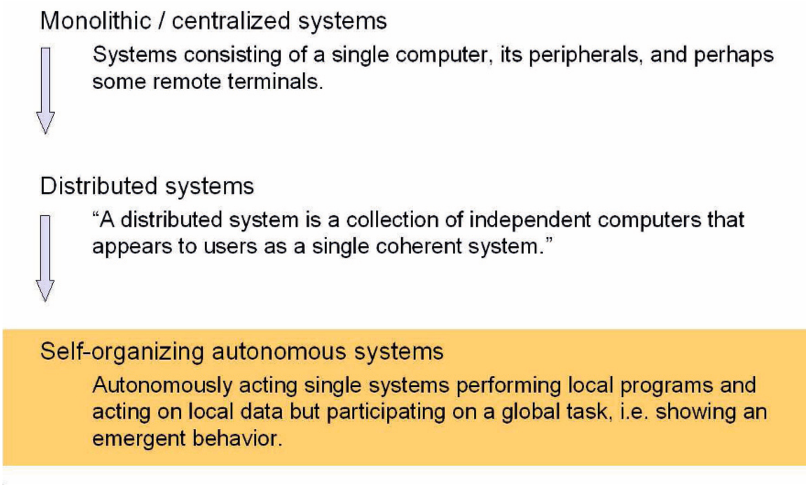

Fig. 13.1. The changing world: centralized systems, decentralized control, and self-organization [13].

management and optimized control will be replaced by methodologies that focus on local knowledge about the environment and adequate decision making processes. Similar problems are known and well-studied in nature. Therefore, such biological solutions should be analyzed for adaptation to the communication in ad hoc networks and WSN.

The goal of this article is to provide an overview of some bio-inspired networking mechanisms and to introduce the underlying biological functionality as well as the adaptation to technical processes. Even though it is not intended as a general review, it summarizes the best-known approaches and explains selected mechanisms in more detail.

13.2 Networked embedded systems

Networked embedded systems are used in many application scenarios. Above all, wireless sensor networks (WSN) are widely studied [3, 6]. Sensor networks consist of multiple, usually hundreds or even thousands of sensor nodes. Such networks do not have a predominant topology but are created dynamically, ad hoc on demand. The nodes themselves can be of any size. Nevertheless, most publications understand sensor nodes as small, battery-driven devices with limited processing power and memory, radio communication, and sensors to measure physical parameters such as the temperature.

Similarly, sensor/actuator networks (SANET) extend the idea of wireless sensor networks to mobile actuation systems, e.g. robot-like systems. In general, such SANET are built of cooperating mobile autonomous systems that allow some kind of actuation, e.g. handling, mobility [2].

With WSN and SANET, new issues appeared that are not covered by existing communication methods and protocols. Some of these issues are inherent in the idea of interconnecting thousands of networked embedded systems, others evolve based on particular application scenarios of WSN:

Node mobility: In general, sensor networks are believed to be stationary, i.e., to have a fixed topology – at least in terms of node location. Admittedly, node mobility is becoming a major concern of new application scenarios such as logistics. SANET, on the other hand, inherently include location dynamics and mobility.

Network size: In contrast to other networks, the number of nodes that are building a network on demand can be very high. Structured networks such as the Internet benefit from a hierarchical organization and a centralized management of subnetworks. WSN and SANET are infrastructureless networks facing scalability problems if too many nodes are concerned.

Deployment density: Depending on the application scenario, the node density in a WSN can be very high. This may break existing medium access control protocols and lead to energy exhaustion just for neighborhood detection.

Energy constraints: Instead of having unlimited energy for computation and communication, energy constraints are much more stringent than in fixed or cellular networks. Usually, sensor nodes are battery operated and in certain cases, recharging of the energy source is impossible. We distinguish replenishable power sources, e.g., for wearable sensors, non-replenishable power sources, e.g. for sensors deployed in remote, hazardous terrain, and regenerative power sources.

Data / information fusion: Limited bandwidth as well as the mentioned power constraints demand aggregation techniques. Each data packet that has to be transported through a WSN is expensive. Aggregated data reduce energy consumption and provide higher usefulness.

In summary, it can be said that self-organization mechanisms are needed for higher scalability in WSN/SANET communication [12]. The basic mechanisms available include neighborhood discovery, topology (re-)organization, and probabilistic approaches. Since optimization on a global level is no longer possible, there is always a discrepancy between multiple objectives. For example, the latency of path-finding with on-demand routing protocols may be too high and periodic routing overhead in a table-driven routing protocol may consume a significant amount of bandwidth [1]. On the other hand, the probability of successful transmission might be too low for stateless approaches. Therefore, hybrid architectures may improve the scalability and optimize the network behavior depending on the application scenario.

Figure 13.1 illustrates the control and management of systems consisting of multiple subsystems. Centralized control is primarily used to operate in an environment consisting of a few nodes. Using centralized information about all systems, optimized solutions for communication and task allocation can be

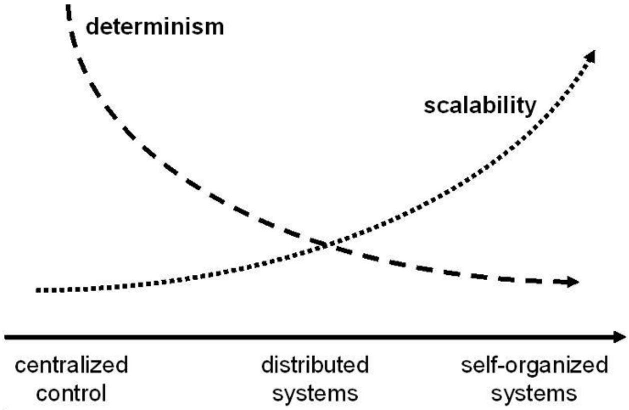

Fig. 13.2. Antagonism between determinism/controllability vs. scalability in system management and control [13].

derived. Examples are perfect schedules for medium access and real-time failure detection and repair. Distributed control allows to manage larger numbers of systems in a scalable way by preserving most systems characteristics such as controllability. Nevertheless, optimization becomes harder and predictability is reduced. Finally, self-organizing systems should help to overcome all scalability problems.

Unfortunately, determinism and controllability of the overall system are reduced. The relation between determinism and scalability is depicted in figure 13.2. Another issue is the challenge of programming such less predictable systems showing emergent behavior.

Referring to networked embedded systems and their management and control, self-organization mechanisms are needed in order to support a large amount of simultaneously intercommunicating nodes. In WSN and SANET, we need new methods to identify available communication paths, nodes, and their capabilities and resources. Additionally, data handling including storage, aggregation, and distribution must be changed and adapted to the new requirements. All mentioned operations should be possible without knowledge about the current network topology, available nodes, their addresses, their location, and others.

13.3 Self-organization: "from nature to engineering"

The turn to nature for solutions to technological problems has brought us many unforeseen great concepts. This encouraging course seems to hold on for many aspects in technology. First studies on biological self-organization and its possible adaptation to technical solutions date back to the 1960ies. Von Foerster [30] and Eigen and Schuster [16] proposed to employ self-organization

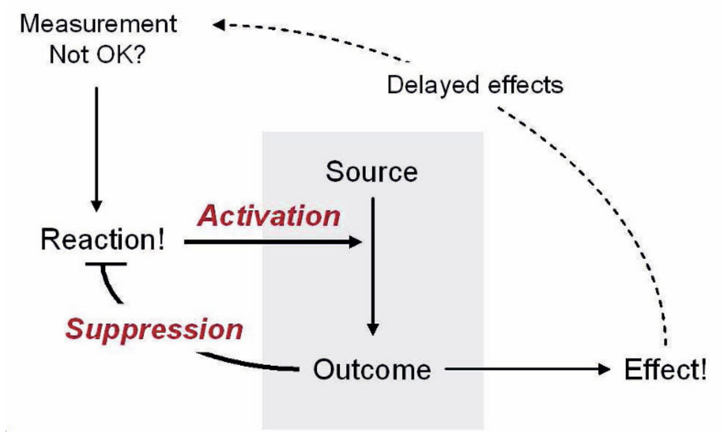

Fig. 13.3. System control using positive and negative feedback loops.

methods as known from many areas in biology. He saw the primary application in engineering in general. Nevertheless, it has been shown that communications can benefit from biologically inspired mechanisms as well.

13.3.1 Basic principles of self-organization

There are three major principles of self-organization mechanisms: feedback loops, local state evaluation, and interaction between individuals. Additionally, probabilistic methods that provide scalability and some degree of predictability can be found in nature and adapted to technology. This process needs careful consideration to prevent mistakes due to limited knowledge about the biological processes or due to the lack of correlation between the natural and the technical models [12].

Figure 13.3 depicts a system that employs all three principles. The main system is performing some action on a source to provide an outcome. Based on this system, the mentioned mechanisms for self-organization need to be discussed in more detail:

Feedback loops: One major component in understanding the interaction of components producing a complex pattern are positive and negative feedback loops. Positive feedback acts as an amplifier for a given effect. In order to prevent overreaction and misregulation, negative feedback is used to efficiently control the system behavior. An example for a positive feedback loop is depicted in figure 13.3, the activation of the processing step. Additionally, a negative feedback loop is included. The outcome directly suppresses an environmental reaction and, therefore, reduces the activation capabilities, i.e., the level of the system's inherent ability to become activated due to observed effects.

Local state: The second ingredient is the local state. This means that all subsystems are acquiring and acting upon locally stored information. Any global control or dependency is prevented in order to enable fully autonomous behavior embedded into a global context. The idea of using local state only is depicted in our example by missing external control processes.

Interactions: Information transfer between individuals is necessary to update the local state. There are two ways to conduct such interactions: direct interaction or communication between related subsystems and indirect information exchange by interacting with the environment. This process is also known as *stigmergic* [9]. The example in figure 13.3 includes stigmergic interactions. The system influences the environment (it produces some effect). This effect can be measured and directly increases or decreases the activation capabilities to the system behavior.

Probabilistic methods: In order to prevent synchronization problems and to increase the variety of application domains scalability is often achieved by random selection.

13.3.2 Bio-inspired techniques in technical systems

The development in the area of bio-inspired engineering is relying on various research fields including swarm intelligence, the artificial immune system, evolutionary and genetic algorithms, and cell and molecular biology based approaches. Some of the best known approaches should be summarized here whereas selected methods are depicted in more detail in the following section.

The behavior of large groups of interacting small insects such as ants and bees builds the basis for the field of ***swarm intelligence***. Simple and seemingly unrelated, autonomously working individuals are considered to compose complex cooperative tasks. Similar actions are required in various areas of engineering and computer science. Thus, swarm intelligence is forming a basis for building self-organizing systems [5, 19]. The focus lies on the formation of groups or clusters that allow efficient task allocation mechanisms. Successful application of swarm intelligence methods has been demonstrated in task allocation and control of multi-robot systems [24]. Recently, similar applicability has been shown in sensor networks [26].

The immune system of mammals builds the basis for research on the ***artificial immune system*** (AIS). The reaction of the immune system, even to unknown attacks, is a highly adaptive process. Therefore, it seems obvious to apply the same mechanisms for self-organization and self-healing operations in computer networks. In the last decade, several architectures for an AIS have been proposed [20, 17]. Application examples include autonomous communication [29] as well as ad hoc networking [27] Additionally, security scenarios including virus and intrusion detection already benefited from AIS approaches [22, 23].

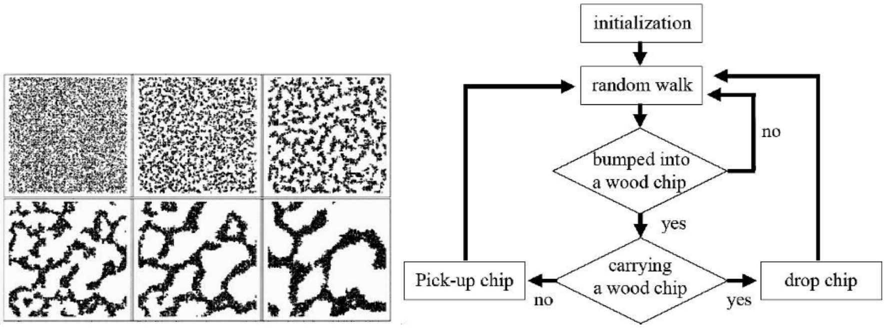

Fig. 13.4. The emergent collective intelligence of groups of simple agents [5].

Evolutionary algorithms (EA) are self-manipulating mechanisms. The evolution in nature is the basis for such methodologies. In particular, there are multiple ways for organisms to learn. A natural selection process (survival of the fittest) is going on letting only the optimal prepared organisms to survive and to reproduce. Changes appear for example by mutations. An overview to evolutionary algorithms is provided for example in [4, 7].

An emerging research area looks for ***cell and molecular biology*** based approaches. All organisms are built in the same way. They are composed of organs, which consist of tissues and finally of cells. This structure is very similar to computer networks, and so are the cellular signaling pathways. Therefore, research on methods in cell and molecular biology promises high potential for computer networking in general and adaptive sensor networks and network security in particular [14, 25].

While many advantages can be identified that make the use of bio-inspired techniques successful, we also need to comment the limitations of bio-inspired mechanisms. Biology always makes compromises between different goals and it is well known that biology sometimes fails. Additionally, some natural mechanisms are not well understood and well-defined problems may be solved by other means.

13.4 Bio-inspired networking

Primarily, the goal of this section is to demystify the concepts of bio-inspired networking. Based on selected approaches, the objectives and solution paths of biologically inspired methods are depicted in more detail.

13.4.1 Swarm intelligence

The collaborative work of a multitude of individual autonomous systems is necessary in many areas of engineering. Swarms of small insects such as bees

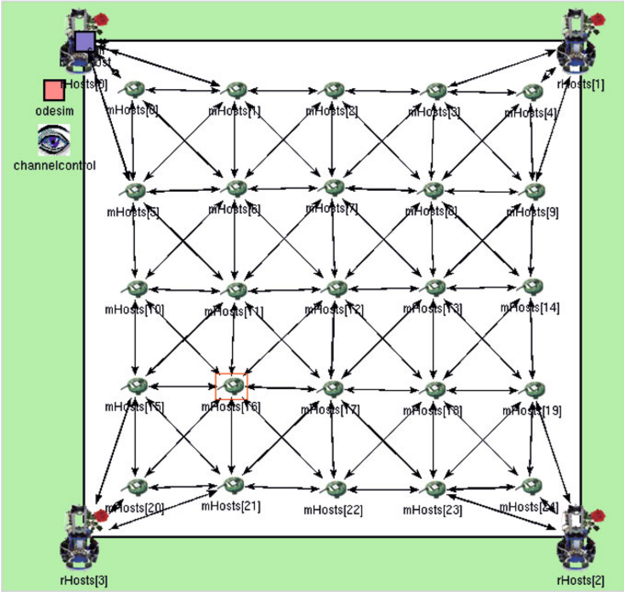

Fig. 13.5. Simulation setup for evaluation of the attractor based task allocation and routing.

or ants address similar issues. For example, ants solve complex tasks by simple local means. There is only indirect interaction between individuals through modification of the environment, e.g., pheromone trails are used for efficient foraging. Finally, the productivity of all involved ants is better than the sum of their single activities and ants are “grand masters” in search and exploration.

The basic principles are simple. All individuals – the systems that collaborate on an overall task – follow simple rules that lead to impressive global behavior, which emerges based on the simple rules and interactions between the systems, either directly or indirectly via the environment. An example is described in figure 13.4. The foraging algorithm used by termites to collect wood chips is shown on the left hand side. Using a simulation model, the overall visible behavior was studied [5]. Quickly, the chips are heaped together and structures emerge from the scene as shown on the right hand side.

Attractor-based routing and task allocation

As a specific example to demonstrate the capabilities of swarm intelligence methods in networking, we chose an attractor scheme for routing and task allocation [26]. In sensor networks supported by mobile robots, routing decisions usually need to be taken on demand because the network topology changes over time. Additionally, multiple tasks may be needed to be executed by different systems in the network. Usually, static programming or complex, auction-based task allocation strategies are used, whereas those approaches fail in large scale and highly dynamic scenarios. The algorithm described here is based on the AntHocNet approach that enables self-organized routing control in ad hoc networks [10]. The pheromone trail mechanism is exploited to search for optimal paths through ad hoc networks. After a short learning phase, the optimal solution can be derived from messages transmitted previously over suboptimal paths.

The new approach is based on a probabilistic scheme. Each node performs a local decision process that provides the basis for task allocation and routing decisions. The basic idea is as simple as powerful. If a node successfully performed a particular task (whether forwarding a packet or anything else), its probability to perform this task again is increased. Similarly, the probability is decreased if the node failed for a particular task. Additionally, each node observes the behavior of the surrounding nodes to update its local behavior accordingly.

More formally, this algorithm can be written as follows. Each node n associates to a task T_i to an attractor τ_i with $i \in T$. At the moment of selecting a task to perform, the node computes a probability for choosing task T_i as follows:

$$P(i) = \frac{\tau_i^{\beta}}{\sum\limits_{k \in T} \tau_k^{\beta}} \tag{13.1}$$

The parameter β was introduced to increase the exploitation of good paths. Each node initializes τ_i with τ_{init}. If the node successfully performed the given task i, τ_i is recalculated as follows: $\tau_i = \min\{\tau_{\max}, \tau_i + \Delta\tau\}$. Similarly, τ_i is reduced for unsuccessful operations: $\tau_i = \max\{\tau_{\min}, \tau_i - \Delta\tau\}$.

The complete algorithm, the corresponding calculations, and an in-depth evaluation can be found in [26]. In that paper, a set of experiments was performed to demonstrate the advantages of the attractor scheme. The simulation setup is shown in figure 13.5. 25 nodes were places in a grid on a playground of 500m$\times$500m. Four different tasks were defined to be performed by all these nodes. Disregarding task allocation, we focus on the associated route selection in this network.

Figure 13.6 shows selected simulation results. On the left hand side, we show a typical snapshot of the distribution of tasks in the network. The plot refers to task T_3. It can be seen that when a node had high probability of performing T_3, its neighbors were likely to have a low one. The routes that were used to send the data to the base host are depicted in the same figure on the right. The network was split in two halves: there were few links between the top right triangle and the bottom left triangle. This figure does not represent the steady state of the network. The network reached a dynamic equilibrium, where things continually changed. This is especially true for the depicted routes, since the routing table entries were removed after a while, and new discoveries took place.

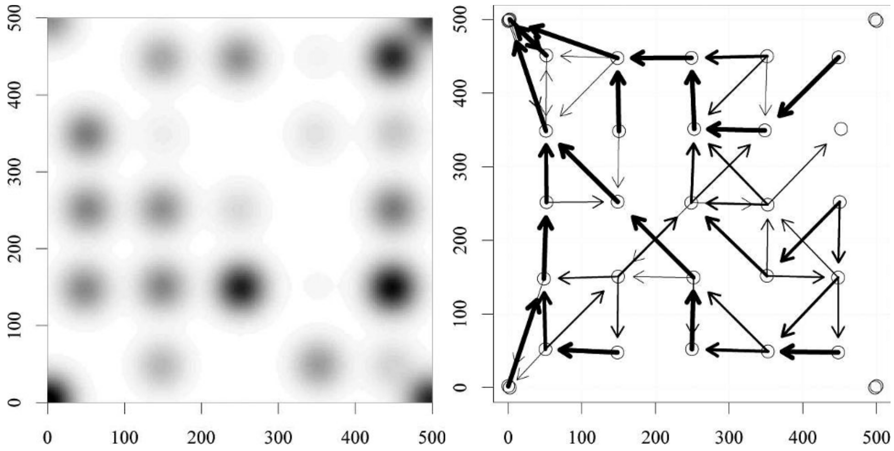

Fig. 13.6. Simulation results [26]. Left: Distribution of task T_3 among the nodes. The darker the circle, the higher is the probability that an agent performs T_3. Right: Routes to deliver the output of T_3 to the base host (in the upper left corner). The arrows show the known next hops for every node. Their thickness is proportional to the probability of choosing a node as next hop.

This example illustrates an architecture for attractor-based task allocation and routing. The nodes make use of solutions inspired by ants' behavior. The control architecture is based on strong interlayer and interagent interactions. The latter are local, meaning that they occur only between agents within a given range, smaller than the experimental area. The architecture is based on probabilistic decisions. During the lifetime of the network, the nodes adapt their probability to execute one task from a given set. The architecture exploits the interactions between agents, but only within a limited range. The local interactions are, however, sufficient to induce a global pattern, i.e., to provide a self-organizing behavior. No particular knowledge of the environment or of the other nodes' activity is required. Moreover, the architecture is based on a cross-layer design, in which application and network layers collaborate on a common objective.

13.4.2 Artificial immune system

Artificial immune systems are computational systems inspired by theoretical immunology and observed immune functions, principles and models, which are applied to complex problem domains [8]. The primary goal of an artificial immune system (AIS) is to efficiently detect changes in the environment or deviations from normal system behavior. The most impressing capabilities of the immune system are its recognition capabilities (anomaly detection, noise tolerance), its robustness, diversity, the capability of reinforcement learning, and the possibility to memorize observations. These features allow to build self-optimizing and self-learning processes.

The role of the mammalian immune system can be summarized as follows. It should protect the body from infections. For this, two immune responses were identified. The primary one is to launch a response to invading pathogens leading to an unspecific response (using leucoytes). In contrast, the secondary immune response remembers past encounters, i.e., it represents the immunologic memory. It allows a faster response the second time around showing a very specific response (using B-cells and T-cells).

The immune recognition is based on the complementarity between the binding region of a receptor and a portion of an antigen called epitope. Antibodies have a single type of receptor, while antigens might show several epitopes. This means that different antibodies can recognize a single antigen. The immune system needs to be able to differentiate between self and non-self cells. Antigenic encounters may result in cell death; therefore, the immune system establishes some kind of positive and negative selection.

The scope of AIS is widespread. There are applications for fault and anomaly detection, data mining (machine learning, pattern recognition), agent-based systems, control, and robotics. In the mammalian immune system, the shape of the molecules defines the degree of binding. In an AIS, a similar distance measure is needed. Typically, antigens and antibodies are described in form of vectors, i.e. $Ab = \langle Ab_1, Ab_2, \ldots, Ab_L \rangle$ and $Ag = \langle Ag_1, Ag_2, \ldots, Ag_L \rangle$. Different shape spaces can be used depending on the current environment:

Real-valued shape space: the attribute strings are real-valued vectors.
Integer shape space: the attribute strings are composed of integer values.
Hamming shape space: composed of attribute strings built out of a finite alphabet of length k.
Symbolic shape space: usually composed of different types of attribute strings, such as a 'name', a 'color', etc.

Based on this definition, the matching of antigens to antibodies can be described using their *affinity*. The affinity is related to the distance. For example, the Euclidean distance can be used:

$$D = \sqrt{\sum_{i=1}^{L} (Ab_i - Ag_i)^2} \tag{13.2}$$

Other distance measures such as Hamming or Manhattan can be used as well. The main application in computer science and engineering is anomaly detection. The normal behavior of a system is often characterized by a series of observations over time. The problem of detecting novelties, or anomalies, can be viewed as finding deviations of a characteristic property in the system. For computer scientists, the identification of computational viruses and network intrusions is considered one of the most important anomaly detection tasks.

One of the first AIS was presented in [20]. Based on this work, misbehavior detection and attack or intrusion detection systems were developed according to the working principles of the natural immune system [22, 23, 27]. Besides network security applications, the operation and control of multi-robot systems was addressed by AIS approaches. The collaborative behavior of robots collecting objects in an environment is difficult to optimize without central control. It was shown that an emerging collective behavior through communicating robots using an AIS overcomes some of the problems. The immune network theory was used to suppress or encourage robots behavior [28].

Misbehavior detection in mobile ad hoc networks

In ad hoc networks, each node serves as both an end system and a router. This allows to build dynamic on demand network topologies supporting mobile systems as well. Various routing protocols for mobile ad hoc networks have been proposed focusing on the efficiency in terms of route detection and maintenance (time, overhead, etc). This dynamic behavior allows – on the one hand – to enable sophisticated mobile applications. On the other hand, such dynamics also open ways to attack the network on the routing protocol layer. Such attacks might be initiated for denial of service reasons as well as for taking over the ad hoc network for private services. A third reason for misbehavior in ad hoc networks is the occurrence of faulty nodes. Either the system might be erroneous or the routing protocol might be incorrectly implemented. A misbehavior detection scheme using an artificial immune system has been developed [27], which works for DSR (dynamic source routing), a particular ad hoc routing protocol. The goal was to build a system that, like its natural counterpart, automatically learns and detects new misbehavior. It employs negative selection, an algorithm used by the natural immune system. In the original paper, the mapping of the natural immune system concepts such as self, antigen and antibody to a mobile ad hoc network is defined and the resulting algorithm for misbehavior detection is presented. The following elements have been defined:

Body: the entire mobile ad-hoc network.
Self-Cells: well behaving nodes.
Non-Self Cells: misbehaving nodes.
Antigen: Sequence of observed DSR protocol events recognized in sequence of packet headers. Examples of events are “data packet sent”, “data packet received”, “data packet received followed by data packet sent”, “route request packet received followed by route reply sent”.
Antibody: A pattern with the same format as the compact representation of antigen
Negative Selection: Antibodies are created during an offline learning phase. In a deployed system this would be done in a testbed with nodes deployed by an operator.

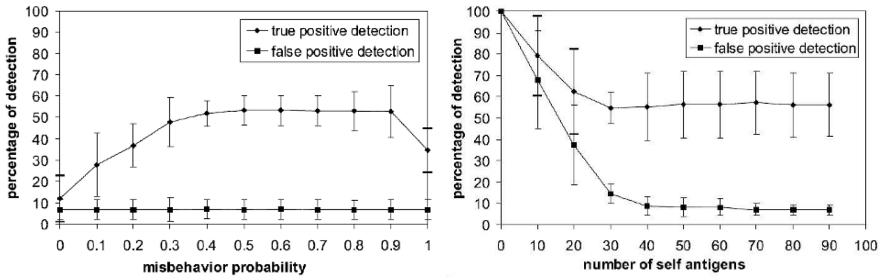

Fig. 13.7. Impact of misbehavior and parameter tuning [27]: Probability of correct detection of misbehaving nodes (true positive) and erroneous detection of well behaving nodes (false positive) vs. misbehavior probability for the misbehaving node (left) and number of self antigens collected for learning (right).

Since antigens represent traces of observed protocol events, such sequences would become very long in a short period of time. Therefore, all traces need to be limited by a time limit Δt for the observation interval. A typical sequence (the letters represent different protocol events) would look like this: $l_1 = (EAFBHHEDEBHDHDHDHD\ldots)$. Then, a number of "genes" are defined. A gene is an atomic pattern used for matching. Typical examples are g_1=#E in sequence or g_2=#(H*D) in sequence. With this information, l_1 can be mapped to an antigen like this: l_2=(3 2 7 6). Finally, the antigens are encoded in binary representation. The numeric range of antigens is split into several intervals and the bit in the representation is set to one if the antigen belongs to this particular interval: l_3=(0000000010 0000000010 0000001000 0000001000).

As previously described, a matching function must be defined to associate antigens to antibodies. Antibodies have the same format as antigens (such as l_3), except that they may have any number of nucleotides equal to 1. An antibody matches an antigen if the antibody has a 1 in every position where the antigen has a 1. This approach has already been successfully demonstrated in [21]. It is used in this paper as a method that allows a detection system to have good coverage of a large set of possible non-self antigens with a relatively small number of antibodies. Antibodies are created randomly, uniformly over the set of possible antibodies. During negative selection, antibodies that match any self antigen are discarded.

The primary evaluation criteria for such detection approaches are the true positive detection rate and the false positive detection rate, i.e., the number of successfully identified misbehaving nodes and the number of accidentally mis-identified nodes, respectively. As shown in figure 13.7, the approach yields quite encouraging results.

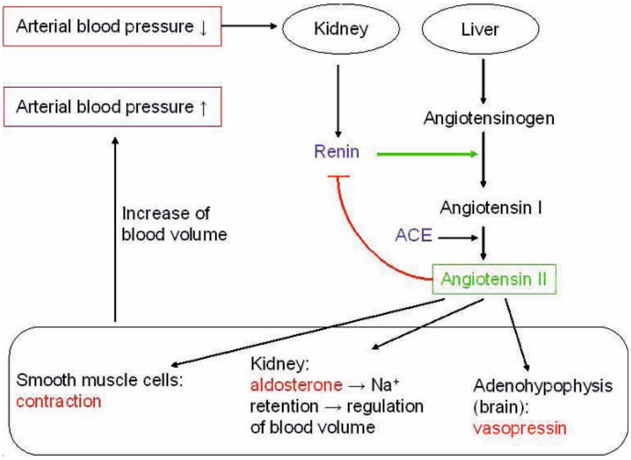

Fig. 13.8. Overview of the regulation of signaling cascades responsible for regulating the blood pressure [15].

13.4.3 Intercellular information exchange

Regarding efficient networking, investigations into the structure and organization of intercellular communication seem to be valuable. Molecular biology is the basis of all biological systems and features high specificity of information transfer. Interestingly, we find many similar structures in biology and in technology, especially in computer networking [25]. The primary concepts are intra- and intercellular signaling pathways and diffuse communication in large-scale structures. Considering the knowledge about molecular biology and its similarity to communication networks [14], it is possible to extract the following principles: efficient response to a request, shortening of information pathways, and directing of messages to an applicable destination.

The information pathways can be distinguished into local and remote. Local: a signal reaches only cells in the neighborhood. The signal induces a signaling cascade in each target cell resulting in a very specific response, which vice versa affects neighboring cells. Remote: a signal is released into the blood stream, which carries it to distant cells and induces a response in these cells, which then passes on the information or can activate helper cells (e.g. the immune system). Signals can appear in the form of particles, i.e., proteins and hormones, as well as of environmental conditions that can be observed and changed, e.g. the calcium concentration.

Inhibitors and promoters forming efficient feedback loops

An example for successful application of the described communication method in WSN is the feedback loop mechanism [15]. Here, the Angiotensin-based regulation process for the blood pressure was used to model the control loop for an efficient regulatory process in an organism. In the case of decreasing arterial blood pressure, the kidney starts to produce a specific protein, renin. This protein initiates a cascade of conversions and activations, respectively. So it promotes the conversion of another protein (angiotensinogen) to a shorter one (now called angiotensin I), which is finally translated to angiotensin II. This protein represents the final response, which now has many effects on different cells in different organs in order to increase the blood pressure to its normal level. At the same time, a molecular negative feedback mechanism finishes the whole cellular reaction. If all receptor are bound by angiotensin II, the reaction is blocked, which in turn also blocks the primary conversion of angiotensinogen to angiotensin II in the way that the initial renin secretion is blocked. This process is shown in figure 13.8.

This process was adapted to work in a sensor network by using the following two concepts:

1. The density of the sensor network allows for alternative feedback loops via the environment: directly via the physical phenomena to be controlled by the infrastructure.
2. Indirect communication allows for more flexible organization of autonomous infrastructures and reduces the number of control messages.

In a sensor network, the control of activities requires information exchange between multiple nodes in the network. Such communication is needed for at least two reasons. First, the control information must be transported to the appropriate destination and, second, the destination must respond to the request by confirming the instructions. All conventionally designed network protocols for such a function follow the same principles. Transmission of a data packet destined for the particular target is initiated. State information is accumulated at several points in the network until a response packet is received which confirms the transaction. The paradigms for data transport in sensor networks are already changing. Directed diffusion, which was introduced in [18], has some interesting features: data-centric dissemination, reinforcement-based adaptation to the empirically best path, and in-network data aggregation and caching. Similar changes are expected for the control information flow which we are focusing on.

As learned from biology, a diffuse communication principle has been proposed [11, 15]. Each message to be sent is given a priority, which reflects the importance of achieving the particular task. Based on this priority, the message is sent to a percentage of the direct neighbors and an even lower percentage of remotely accessible nodes. This process is repeated until the desired job is confirmed running or the job is canceled globally. Thereby, a random factor is applied to the dispersion of information or, in particular, to the distribution of tasks. The benefit lies in better system efficiency and reliability, especially in unreliable multihop ad hoc wireless sensor networks.

13.5 Conclusion

In conclusion, it can be said that many approaches for bio-inspired networking have been studied and we can already see first impressive solutions and applications. Basically, the following mechanisms have been adapted to solve open issues in networking: feedback loops, i.e. positive feedback to initiate actuation or data aggregation, and negative feedback for network congestion control and smooth regulation; local state information for efficient data fusion, energy control, and clustering; and weighted probabilistic approaches for task allocation, controlled communication and congestion control. Finally, we are facing a multi-objective optimization process that balances between overhead (latency vs. energy) vs. predictability.

Self-organization mechanisms for communication and coordination between networked embedded systems need further research and development. There are many objectives and many directions, but similar solutions can be derived. Bio-inspired networking is a powerful approach among several others. Ongoing research objectives include efficient data dissemination, handling and storage in WSN as well as task allocation schemes and distributed control in SANET.

13.6 Acknowledgments

This work is partially a result of a collaborative research project conducted together with Dr. Bettina Krüger, Dept. of Physiology, University of Erlangen-Nuremberg, Germany.

References

1. K. Akkaya and M. Younis. Energy-aware routing of time-constrained traffic in wireless sensor networks. *Journal of Communication Systems, Special Issue on Service Differentiation and QoS in Ad Hoc Networks*, 17(6):663–687, 2004.
2. I. F. Akyildiz and I. H. Kasimoglu. Wireless sensor and actor networks: Research challenges. *Elsevier Ad Hoc Network Journal*, 2:351–367, October 2004.
3. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–116, August 2002.
4. P. J. Bentley, T. Gordon, J. Kim, and S. Kumar. New trends in evolutionary computation. In *Congress on Evolutionary Computation (CEC-2001)*, pages 162–169, Seoul, Korea, May 2001.
5. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, 1999.
6. D. Culler, D. Estrin, and M. B. Srivastava. Overview of sensor networks. *Computer*, 37(8):41–49, August 2004.
7. S. K. Das, N. Banerjee, and A. Roy. Solving otimization problems in wireless networks using genetic algorithms. In *Handbook of Bio-inspired Algorithms*. 2004.

8. L. N. de Castro and J. Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach.* Springer, Berlin, 2002.
9. G. Di Caro and M. Dorgio. AntNet: Distributed stigmergetic control for communication networks. *Journal of Artificial Intelligence Research*, 9:317–365, December 1998.
10. G. Di Caro, F. Ducatelle, and L. M. Gambardella. AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications, Special Issue on Self-organization in Mobile Networking*, 16:443–455, 2005.
11. F. Dressler. Locality driven congestion control in self-organizing wireless sensor networks. In *3rd International Conference on Pervasive Computing (Pervasive 2005): International Workshop on Software Architectures for Self-Organization, and Software Techniques for Embedded and Pervasive Systems (SASO+STEPS 2005)*, Munich, Germany, May 2005.
12. F. Dressler. Self-organization in ad hoc networks: Overview and classification. Technical Report 02/06, University of Erlangen, Dept. of Computer Science 7, March 2006.
13. F. Dressler. *Self-Organization in Sensor and Actor Networks.* John Wiley & Sons Ltd., West Sussex, 2007.
14. F. Dressler and B. Krüger. Cell biology as a key to computer networking. In *German Conference on Bioinformatics 2004 (GCB'04), Poster Session*, Bielefeld, Germany, October 2004.
15. F. Dressler, B. Krüger, G. Fuchs, and R. German. Self-organization in sensor networks using bio-inspired mechanisms. In *18th ACM/GI/ITG International Conference on Architecture of Computing Systems - System Aspects in Organic and Pervasive Computing (ARCS'05): Workshop Self-Organization and Emergence*, pages 139–144, Innsbruck, Austria, March 2005.
16. M. Eigen and P. Schuster. *The Hypercycle: A Principle of Natural Self Organization.* Springer, Berlin, 1979.
17. S. Hofmeyer and S. Forrest. Architecture for an artificial immune system. *Evolutionary Computation*, 8(4):443–473, 2000.
18. C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCOM'00)*, pages 56–67, Boston, MA, USA, August 2000.
19. J. Kennedy and R. C. Eberhart. *Swarm Intelligence.* Morgan Kaufmann Publishers, San Francisco, 2001.
20. J. O. Kephart. A biologically inspired immune system for computers. In *4th International Workshop on Synthesis and Simulation of Living Systems*, pages 130–139, Cambridge, Massachusetts, USA, 1994. MIT Press.
21. J. Kim and P. J. Bentley. The artificial immune system for network intrusion detection: An investigation of clonal selection with negative selection operator. In *Congress on Evoloutionary Computation (CEC-2001)*, pages 1244–1252, Seoul, Korea, July 2001.
22. J. Kim and P. J. Bentley. An evaluation of negative selection in an artificial immune system for network intrusion detection. In *Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 1330–1337, San Francisco, CA, July 2001.

23. J. Kim and P. J. Bentley. Towards an artificial immune system for network intrusion detection. In *Congress on Evolutionary Computation (CEC-2002)*, pages 1015–1020, Honolulu, May 2002.
24. M. J. B. Krieger, J.-B. Billeter, and L. Keller. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406:992–995, August 2000.
25. B. Krüger and F. Dressler. Molecular processes as a basis for autonomous networking. *IPSI Transactions on Advances Research: Issues in Computer Science and Engineering*, 1(1):43–50, January 2005.
26. T. H. Labella and F. Dressler. A bio-inspired architecture for division of labour in sanets. In *1st IEEE/ACM International Conference on Bio-Inspired Models of Network, Information and Computing Systems (IEEE/ACM BIONETICS 2006)*, Cavalese, Italy, December 2006.
27. J.-Y. Le Boudec and S. Sarafijanovic. An artificial immune system approach to misbehavior detection in mobile ad-hoc networks. In *First International Workshop on Biologically Inspired Approaches to Advanced Information Technology (Bio-ADIT2004)*, pages 96–111, Lausanne, Switzerland, January 2004.
28. D. W. Lee, H. B. Jun, and K. B. Sim. Artificial immune system for realization of cooperative strategies and group behavior in collective autonomous mobile robots. In *4th International Symposium on Artificial Life and Robotics*, pages 232–235, 1999.
29. J. Suzuki and Y. Yamamoto. Biologically-inspired autonomous adaptability in a communication endsystem: An approach using an artificial immune network. *IEICE Transactions on Information and Systems*, E84-D(12):1782–1789, December 2001.
30. H. von Foerster. On self-organizing systems and their environments. In M. C. Yovitts and S. Cameron, editors, *Self-Organizing Systems*, pages 31–50. Pergamon Press, 1960.