Cyber Biosphere for Future Embedded Systems

Franz J. Rammig
Heinz Nixdorf Institute
Universität Paderborn
Paderborn
Germany
franz@upb.de

Abstract Future Embedded Systems are heading into a degree of complexity which is far beyond today`s level. As most technical artifacts will be interconnected in some sense ("*Internet of Things*") Embedded Systems of the future cannot be treated as isolated entities any longer. Two major tendencies to cope with this challenge can be observed. The first one takes its inspiration from the technical roots of Embedded Systems. They are looked at from their technical nature but the traditional boundaries of Embedded Systems, especially to consider them as isolated systems are overcome. This approach became well known under the name "*Cyber Physical Systems (CPS)*". The second approach observes the existence of highly successful and relatively stable systems in form of our *biosphere*. So it seems to be wise to take inspirations from the achievement of nature. This approach became rather popular under the term "*Biologically Inspired Systems*" or "*Organic Computing*"[1]. In this paper we will concentrate on the latter attempt to build the highly complex, highly sophisticated Embedded Systems of the future. Inspirations from ant colonies, from the hormone system, and from the immune system will shortly be discussed using specific examples. Some comparisons with the CPS approach will be made as well.

Keywords: Biologically Inspired Techniques, Ant Colony Algorithms, Artificial Hormone Systems, Artificial Immune Systems

---

[1] See http://www.organic-computing.de/ for the Organic Computing Initiative and http://www.aifb.uni-karlsruhe.de/Forschungsgruppen/EffAlg/projekte/oc/inhalte for the Organic Computing Priority Program funded by the German Science Foundation (DFG)

1 Introduction

Engineers are interested to build highly efficient, highly reliable, and highly deterministic systems; they are interested to keep their systems completely under control under all potential circumstances. For this purpose the embedded systems community, especially the real-time researchers have developed sophisticated solutions: deterministic real-time scheduling techniques, schedulability analysis, collision-free communication protocols, time-triggered architectures, formal proof techniques, just to mention some of them. Adapting inspirations from the biosphere, a world that seems to follow completely different approaches, appears to be strange idea at the first glance. On the other hand engineers are impressed by the robustness of extremely complex biological systems. A human, made of billions of cells, interacting in a highly sophisticated manner, is continuously exposed to billions of enemies (antigens) which change their attacking strategies rapidly and in a non predictable manner. By simple MTBF calculations one would conclude that a human's lifetime should not exceed some hours. However such a complex system survives in a hostile environment for a very long time. The same can be said for any kind of complex bio-conglomerates. So, biological systems have proven to be extremely robust even in dynamically changing hostile environments. Of course engineers also are able to design highly complex systems. A today's SoC comprises a billion of transistors as well and it runs reliably for a long time. Giant SW systems like telephone switching systems are very reliable as well. What can be questioned, however, is the stability and robustness in case of changing environmental conditions or in case of unforeseen hostile circumstances. Of course biological systems can handle unforeseen situations also only to a certain amount. In cases beyond this level of flexibility the respective species disappears. However, it seems that this limit of biological flexibility is much broader than in conventional technical artifacts. From this observation it does not surprise that one of the most stable, most robust and most adaptive complex technical artifact is the internet. In fact the internet follows a couple of basic principles of biological systems like distributive design, postponing decisions and actions into the operational phase, self-organization, emerging redundancy, just to mention some of them.

Common to the highly complex systems of the future are the following key characteristics:

- complex volatile networks in which components cooperate as well as possibly compete,
- decentralized control and components acting autonomously,
- an unobservable global system state and thus components with only local knowledge,
- optimization of own benefits being the driving force of a component's cooperation,
- adapting to and learning from environmental changes as a universal ability of components,
- limited availability of resources combined with security and safety requirements.

In each of these settings, the global system state is neither observable nor would a knowledge of it (due to its complexity) be of any help. New properties emerge while the network's components adapt to and learn from other components. These fundamental characteristics raise a number of new research questions that need to be addressed in order to achieve any progress in this area. All the mentioned properties are present in biological systems as well. Therefore it seems to be attractive looking for inspirations in this domain. Biological systems seem to follow optimal strategies (or at least near-optimal ones) in the presence of partial or even unreliable information. Biological components are able to "decide" which information is relevant and which need not be considered. They follow "algorithms" reaching stable, robust, and desirable behavior in a distributed network. Biological entities find out about their right option of interaction with cooperating or even competing other components. Nature "invented" clever, adaptive, and efficient communication principles. All this is done under restricted resources and even in case of failing parts. Nature transformed most of the decisions and actions into the operational phase of biological artifacts which results in highly adaptable systems. These systems reflect on both their own and their environment's behavior and consequently change themselves. Nature provides techniques that can ensure the correctness of emergent volatile systems.

To sum up: Highly complex systems behave like global economy. By their tradition engineers tend to organize their artifacts in the way of planned economy. Nature is an economy driven by free enterprise of selfish agents. Such economies may be far away from optimality, they tend to locally show nondeterministic behavior at certain points of time. But they seem to be extremely robust on the long term. In this paper we would like to provide some hints why it could be wise for engineers to accept a certain amount of free economy as well.

2 Ant Colony Algorithms

The total biomass of ants on earth is more or less the same as the biomass of mankind. Ants can be seen as one of the most advanced examples of social bio-systems. Ant colonies can be interpreted as a specific kind of an organism, forming an interesting compromise between simple swarms of single cell life and highly organized multi-cell systems (e.g. mammals) where most cells are fixed at a specific location and play a specific role. Differently from these two extremes in an ant colony the individual constituent (an ant) is a multi-cell object, mobile, intelligent to a certain degree, but closely embedded into a global collaborative scheme. *Ant Colony Optimization* (ACO) is a cooperative meta-heuristic being successfully applied to various combinatorial optimization problems. Ants tend to find the shortest path from their nests to a food source in a relatively short time. For doing so, they communicate in an indirect manner, called *stigmergy*. Moving ants deposit traces of pheromone on their trail. On the other hand, ants have the tendency to follow trails which are marked by pheromone. This establishes a positive feedback which makes a marked trail even more attractive. Evaporation of pheromone establishes a negative feedback. When alternative trails are chosen randomly in the beginning, the pheromone level of a path is inverse proportional to the path's length with high probability. Dorigo et al. [5] were among the first to apply ACO to graph-related optimization problems like the Traveling Salesman Problem (TSP). A more general theory has been developed in his book [6], proceedings of dedicated conferences have been published as well [7, 8].

In their papers [3, 4] the authors describe the application of Dorigo's basic approach to the scheduling problem of MPEG streams via the 802.11e EDCA. For this purpose the precedence-constrained MPEG scheduling has to be mapped onto a directed graph, expressing the precedence relationships of MPEG *Groups of Pictures* (GoP). This results in a cyclic graph consisting of the various I, P, and B frames contained in the GoP being represented as nodes and the precedences as directed edges. A feasible solution represents a schedule of MPEG frames where each frame is expected to be transmitted within its (previously defined) delay bounds. On such a graph a colony of π ants is deployed. An ant of such a colony sitting on a "border" node of a partially feasible schedule selects an edge from this node to an attainable node according to a probabilistic function as in Dorigo's original work. A tour is said to be completed if all π ants of a colony have returned to the initial I-frame. Then the best selected path is evaluated by counting the number of timely scheduled frames. On each edge of this path the pheromone values are updated. The updated value is proportional to the ratio of the achieved solution and the optimal one (all frames of the GoP scheduled timely). As a result, near optimal solutions that entail higher concentration of pheromone will have a higher impact on the edge selection process in subsequent tours. In experiments this algorithm turned out to be nearly as efficient (concerning needed computation time) as a dedicated scheduling algorithm designed at our institute by the same author. However it showed a much more robust behavior with respect to rapidly changing load and transmission distortions.

Large ad hoc networks can be clustered following an approach based on division of labor in colonies of social insects like *Pheidole Rea*. The basic idea in this case is to treat each node of an ad hoc network either as a "major" ant or a "minor" one. A major represents a cluster head which means a higher workload while the minors are member nodes of clusters. The main power of the approach is originating from the built-in elasticity. Both types of species have a certain threshold to become major or minor. On the other hand they are stimulated by received signals. Whenever the strength of such signals is above a certain threshold the role of a major may change to a minor or vice versa. Typical stimuli signals are signal strengths of received messages, frequency of received messages, etc. Thresholds are established e.g. by the power reserve of a node. A cluster head with flattening power resources has a tendency to become a minor (member node), an "isolated" member node to become a cluster head; see [10] for more details. This approach again shows enormous robustness against rapidly changing situation.

In our fine-granular distributed RTOS *NanoOS*, services are distributed over the nodes of a cluster; the clusters being created as described above. The optimization goal here is to migrate services dynamically to such nodes that the global communication costs between services and application tasks requesting these services are minimized. Note that the requesting application tasks may reside on any nodes of a cluster. This problem again can be mapped onto an ACO problem. In our approach services are the equivalent of food sources, service locations are the equivalent of shortest paths, calls made by the requesters are the ants, and requesters are the nests.

Wireless links form the paths which the ants can use for movement. While the requests are being routed to the destination service, they leave pheromone on the nodes. The pheromone, on the other hand, evaporates over time. This solution is further enhanced to also consider the specific workload on the destination nodes of potential migrations. In addition geographically related paths are handled in such a way that they bundle attracting force into their direction. Details can be found in [11].

2 Artificial Hormone Systems

All biological system can be seen just as a collection of individually operating cells which follow some collaborative principle of operation based on some communication means. Electrical signaling via the nerve system constitutes a means of directed communication in the sense of single-cast or multicast. Controlled and centrally coordinated actions like contraction of specific muscles to enable movement may serve as an example. In other situations when an extremely high number of potentially receiving cells have to be addressed and if those cells are widely spread across a body a multi-cast communication scheme is desirable. In bio-systems this is carried out by means of the hormone system which can be interpreted as a way of biological broadcasting. Specific chemicals are generated by the sending instance and cause reactions on the side of receiving cells. It is essential that the receiving cells can react in a specific manner. This specific reaction may depend on cell type or even on a specific cell instance and its current environmental setting. Even the set of hormones may be specific for the different cells. Hormones unknown to a certain receiver are just ignored. So the intended communication is established only between processing elements that share a joint reservoir of hormones. By this concept multi-cast can be implemented easily. This simple basic principle thus can be tailored in numerous ways to result in the desired behaviors.

In [1] the authors discuss an approach to apply concepts of artificial hormone systems to task allocation on heterogeneous processing elements. In their approach each of the processing elements and the tasks to be assigned may secret “hormones” or may react on receiving ones. This approach strictly follows a decentralized approach. Each processing element may have an individual rule set for the secretion of hormones or how to react on receiving certain ones. The only common rules are given by some agreement what hormones to be used. In their approach the authors implement a distributed feedback controller by means of two principle types of hormones, so called accelerators (positive feedback) and so called suppressors (negative feedback). The first ones are sent out to indicate the willingness of a processing element to attract additional tasks, the second one to indicate the inability to do so. The approach results in a couple of self-x properties: *self-configuration* as there is no central control, *self-optimization* as there may be included rules to re-open the assignment “market” periodically or stimulated by some events, *self-healing* as a failing task or processing element is no longer sending hormones and by this disturbs the equilibrium which causes some re-allocation. The authors have built a flexible simulation environment which allows them to experiment with a variety of parameter settings.

Stress response is a special version of a hormone system. The "*Fight-or-flight*"-theory by Walter Cannon [2] describes the reaction of humans and animals to threads. In such stress situations specific physiological actions are taking place by the sympathetic nervous system of the organism as an automatic regulation system without the intervention of conscious thought. For example, *epinephrine* a hormone is released which causes the organism to release energy to react on the threat (fight or flight). This concept is adopted to control the on-line reconfigurable real-time operating system DREAMS[2] (***D**istributed **R**eal-time **E**xtensible **A**pplication **M**anagement **S**ystem*) which has been developed by our group. This RTOS is able to manage system tasks and user tasks in the form of different "profiles" by means of a special resource manager [17] (***F**lexible **R**esource **M**anager - FRM*). DREAMS is tailored to the special demands of self-optimizing applications. The manager tries to optimize the resource utilization at run-time. The optimization includes a safe over-allocation of resources, by putting resources that are held back for worst-case scenarios by tasks at other tasks' disposal. The interface to the FRM is called *Profile Framework*. By means of the Profile Framework the developer can define a set of profiles per application. Profiles describe different service levels of the application, including different quality and different resource requirements. All states belonging to one profile build the state space that can be reached when the profile is active. The different profiles can be assigned to specific emergency categories using a generic monitoring concept for self-optimizing systems. The intent is to protect tasks systematically against hazards or faults. These hazards or faults might result from their self-optimizing behavior themselves, but self-optimizing behavior can also support the re-allocation of resources to handle threats. The concept distinguishes four different emergency categories:

1) The system operates regularly and uses its self-optimization for the major system objectives.
2) A possible threat has been detected and the self-optimization is not only used to optimize the behavior but also to reach system states, which are considered to be safer than the current one.
3) A hazard has been detected that endangers the system. Fast and robust countermeasures, like a reflex, are performed to reach a safer state (1 or 2).
4) The system is no longer under control; the system must be immediately stopped or a minimal safe-operational mode must be warranted, to minimize damage.

The artificial hormone system is applied to ensure that the system can provide more resources to enable more efficient countermeasures whenever it experiences entering emergency category 2. The idea is, when a task of the system detects a threat for the system it releases virtual epinephrine. This distributed epinephrine forces non-critical tasks into a profile with lower resource consumption. By this, resources are freed and this permits the critical task to handle the threat more appropriately by switching into a specific emergency handling profile which usually is more resource-hungry. The virtual epinephrine carries the information how much additional resources the epinephrine secreting task requires to activate its threat-handling profile. It is assumed

---

[2] Recently a new version of DREAMS has been created, called **O**rganic **Rec**onfigurable **O**perating **S**ystem (ORCOS). It can be downloaded from https://orcos.cs.uni-paderborn.de

that all tasks are sorted according their safety critical nature. Like the cardiovascular system of an organism the resource manager broadcasts the epinephrine to the tasks. Tasks with the lowest safety level have the shortest reaction time. When the epinephrine is injected into such a task it can react by switching into a special profile with lower resource requirements. The task then updates the information inside the epinephrine how much resources are still required. This updated epinephrine then is secreted again, by this over-writing the hormone already received by tasks at higher safety levels which react more slowly. By this technique finally every task has information about the threat and can react accordingly. The complexity of this process is linear with respect to the number of tasks. The reaction of the tasks to the epinephrine ("consuming" it by update) is done in a short, constant time. Details can be found in [9].

3 Artificial Immune Systems

Immunocomputing intends to establish another kind of computing. The main idea is to copy the immune system's ability to identify abnormal objects ("*antigens*") with high separation precision and to attack such antigens using adapted means ("*antibodies*") in an extremely efficient manner. All this is done in a distributed but interlinked manner and is quickly adapted to varying situations (occurrence of previously unknown antigens) by a sophisticated learning ability. As biological immune systems are based on chemical reactions of proteins, immunocomputing is based on the "*Formal Protein*" as its basic element. A protein is an essential component of organisms and participates in every process within cells. Proteins constitute *epitopes* present in antigens and antigen presenting cells. Proteins constitute also *paratopes* present in antibodies. An epitope is the minimum molecular structure that is able to be recognized by the immune system. One epitope matches with a paratope in molecular recognition. An epitope or a paratope are made of around 10 amino-acids. An antigen presenting cell is a cell that has digested an antigen and presents in its surface an epitope. A protein is composed of amino-acids arranged in a linear chain. The 3D shape or tertiary structure of the epitope is recognized by a paratope. It means an epitope is a kind of surface protein. That is why proteins will be seen as the basic element in immunocomputing.

Cytokines are introduced as an additional concept into immunocomputing [18] to establish collaboration. In biological systems cytokines are groups of proteins secreted by many types of cells. Each cytokine binds to a specific cell's surface receptor signaling a specific action i.e. differentiation into plasma cells, antibody secretion or cell death. They bind also through own receptors constituted from proteins, too.

The basic entities in a biological immune system and therefore also in immunocomputing are so-called *B-cells*. B-cells in the immune system secrete antibodies, i.e. the actuators of immune reaction. On the other hand they also secrete cytokines in order to signal something to another cell. This introduces a positive feedback into the immune system. Then, a B-cell will be taken as a generic cell V_i with two components expressed by $V_i = (c_i, P_i)$ where $c_i \in \mathbb{N}$ represents a cytokine (action to be carried out) and $P_i \in \mathbb{R}^q = ((p_1)_i, \ldots, (p_q)_i)$ is a point in a q-dimensional space. P lies within a cube $max\{| (p_1)_i |, \ldots, | (p_q)_i |\} \leq 1$. It represents a protein transformed into the so-called FIN (*Formal Immune Network*) space. In biological terms it represents an antigen binding site (antigen detection) of an antibody or, simplifying, an antibody.

We applied cFIN (*cytokine FIN*) to build self-repairing FPGAs, following a *Built-in Self-Test (BIST)* approach. The circuit under test receives a test pattern and the response is evaluated by means of cFINs. In this case, an antibody represents the expected output, transformed into the FIN space. An antigen is the response of the circuit under test. A cytokine represents the action to be taken for fault recovery purposes. It is important that the system has to be trained beforehand using a training matrix $V(c,A)$. $A = A_1,\ldots,A_n$ with $A_i = (Input_i, Output_i, Stimuli_i, State_i)$ is a matrix with information about expected responses under defined input patterns. Each expected or unexpected response then is linked to an action expressed by c with $c_i = (self_i, action_i)$. The first component indicates the differentiation between *self* and *not self*, the second one identifies the action to be taken. Using the cytokine communication system, on-line learning can take place during operation. Details can be found in [16]. For general readings on immunocomputing see [19, 20].

4 Discussion

The three approaches presented here are just examples of a broad potential when getting inspiration from nature. Of course these approaches include much more sophistication than the simple principles presented here just to initiate a discussion. In any case it is wise to collect more profound knowledge about biological systems before gaining real benefit out of them for engineering disciplines. Even the three sketches presented here, however, show some interesting similarities. The reason is that nature "invented" life by "inventing" cells. For billions of years life did exist solely in form of single cell entities. So whatever emerged as biological system remains a collection of individual cells, a collection of cells which may cooperate very closely, a collection of cells where the cells may be differentiated into highly specialized ones. However the cells never lost their property of autonomy. Biological systems are federated ones. Social insects may be seen as a copy of the same principle; now using more elaborate "macro cells". And this principle can be recursively extended. It may not be so surprising that the federation principle can be found using more and more complex "cells", a principle that reaches up to human societies. Federation seems to be a very useful principle to achieve robustness. Usually there is some dedication, some division of labor in federated systems. The degree of this division of labor increases by the complexity of the federal community. However it can be observed that in most cases there is more or less elasticity.

Components of a community dedicated to specific tasks can take over other tasks whenever they receive stimulations beyond their present threshold. This observation certainly is a valuable inspiration for future embedded systems. Our own experiments in the areas of service migration, clustering, or real-time scheduling of media streams did show very robust and fault tolerant behavior when following this principle. Division of labor together with elasticity provides a good compromise between efficiency and avoidance of single points of failure.

Federated systems following the basic principle of delegation (distribute globally only what to do, let the individual components decide how to do) rely on an adequate communication scheme. It can be observed that nature created the entire bandwidth from unicast/multicast (nerve system) to multicast/broadcast (secreting hormones/cytokines or pheromones) and from dedicated "cabling" (nerve system) via "*powerline communication*" (hormones/cytokines) to wireless (pheromone). Common to all these communication approaches is the fact that they are tailored for federated systems. All biological systems are made as a collection of cells and each single cell is equipped with sensors and actuators. All higher order constructions make use of this basic principle. By the same reason similarities can be observed between the different communication concepts. Nerve threads are made by sequences of nerve cells communicating via their synapses making use of the ability of any cell to cause and sense electrical potentials. Other capabilities of cells for sensing and acting are given by the ability to expose specific proteins on their surface and to sense the surface of proteins (necessary in any case as part of a cell's digestion system). This principle is used within the hormone system, in immune-networks via cytokines, and also when using pheromone for communication. Common to these techniques is again the principle of delegation. It is up to a cell how to react on a sensed signal. This reaction may depend of the specific cell type or even cell instance (thus enabling multicast) or on actual environmental or state conditions of a cell. An interesting aspect is the reuse of energy flows (cardiovascular system) to transmit messages. This is a kind of biological powerline communication. Stigmergy can be seen as transforming hormones or cytokines to a more general environment. An important principle in any case is a decay mechanism for messages, evaporation of pheromones in case of ant colony communication via stigmergy. Of course, the communication demands in technical systems differ. However it is worth to consider biological communication techniques as inspiration as well. Large, complex systems need a certain degree of self-organization or, even less tight, self-coordination. Under such circumstances pre-planned communication systems seem to be no longer adequate. By the principle of delegation the amount of information to be communicated can be reduced dramatically. We discussed in this paper techniques to make efficient use of stigmergy as part of ACO solutions for service migration and soft real-time scheduling. Hormone-based communication has been discussed in applications for task allocation and stress management while cytokine-based communication plays an important role in our work on self-healing FPGAs based on artificial immune systems. All these communication techniques turned out to be sufficiently efficient and extremely robust.

More recently the discussion about Cyber Physical Systems (CPS) emerged. One of the major arguments within this community is that the traditional separation into functional and non functional properties of computation seems to be no longer adequate when building the deeply embedded but widely distributed systems of the future. Especially abstracting away time which in most areas of computing is a common principle turns out to be a dangerous assumption. The solutions proposed include the usage of a strict and very precise global time source and then abstracting this source to a "*sparse time*" model [14, 15]. Based on such a model adequate OO architectures can be built, e.g. using the TMO approach of UC Irvine [12, 13]. This approach seems to be completely different from the techniques of handling time in biological systems. They tend to follow an approach to approximate and correct afterwards if the approximation turns out to be wrong or not precise enough. It definitely makes no sense to look for inspirations from biology in an ideological manner. Technology opens potentials that were not available within evolution up to now and these potentials have to be used. Establishing a precise global time base was made possible by GPS and comparable systems and as it is available it should be used. Other aspects addressed in CPS research, however, match relatively well with inspirations we can get from biological systems. As already mentioned several times in this paper, all biological systems are build bottom-up using a strict cell-based approach. These cells are more comparable to components than to objects in the OO sense. Communication is done by signaling values; it then is up to the components how to react. This basic principle of delegation constitutes much of the success of biological systems and should be considered as a basic principle for CPSs as well. Biological systems do not distinct between functional and non functional properties. Nature always is aware of resources, is making use of what is available (considers the available "*platform*"), provides solutions how to handle lacking resources to a certain amount. This is another principle to be considered as inspiration when building CPSs. If such systems are built in a bottom-up manner by creating cells based on and closely adapted to available platforms, being sensitive for certain sets of rules, and being highly adaptive, capable of learning, then many of the CPSs' challenges might be solvable. Building a generic framework, a *Cyber Biosphere (CBS)* may be an attempt worth to be worked on.

5 Conclusion

In this paper some arguments are presented for taking inspirations from biology when designing the complex technical artifacts of the future. Using some examples it has been shown, that such inspirations may be helpful especially when the systems have to behave in a robust manner in rapidly changing environments. However, one never should make the mistake just to copy nature into technical artifacts. Our artifacts have to work in a dependable manner for some years or decades. Nature “thinks” in terms of millions of years, short-term behavior is of minor interest. Nature optimizes the long-term global performance; the specific entity is of no interest. Engineers have to consider the single entity, they are liable for. So, taking inspiration from nature should always be an option but never more than an option among others.

6 References

1. A. von Renteln, U. Bringschulte, and M. Weiss: Examinating Task Distribution by an Artificial Hormone System Based Middleware. In: Proc. 11th Symposium on Object-Oriented Real-Time Distributed Computing (ISORC08), pp. 119 – 123, IEEE (2008)

2. W. B. Cannon. Bodily Changes in Pain, Hunger, Fear and Rage: An Account of Recent Research into the Function of Emotional Excitement. Appleton-Century-Crofts (1929)

3. M. Ditze: Evaluation of an Ant Colony Optimization Based Scheduler for the Transmission of Multimedia Traffic in the 802.11e EDCA. In: Proc. of 3rd ACM Interntl. Worksh. on Wireless Multimedia Networking and Performance Modeling (WMUNEP), Chania, Greece (2007)

4. M. Ditze, M. Becker: An Improved Adaptive ACO Meta Heuristic for Scheduling Multimedia Traffic Across the 802.11e EDCA. In: Proc. of 15th Annual Multimedia Computing and Networking (MMCN’08), San Jose, USA (2008)

5. M. Dorigo, V. Maniezzo, and A. Clorni: The Ant System: Optimization by a Colony of Cooperating Agents. IEEE Transactions on Systems, Man,and Cybernetics,Volume 26, pp. 944 – 955, (1996)

6. M. Dorigo and T.Stützle AntColony Optimization, MIT Press (2004), ISBN 0262042193

7. M. Dorigo, M. Birattari, C. Blum, L.M. Gambardella, F. Mondada, and Th. Stützle, (Eds.) Ant Colony Optimization and Swarm Intelligence, 4th International Workshop, ANTS 2004, Brussels, Belgium, September 5-8, 2004, LNCS , Vol. 3172 (2004)

8. M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, and Th. Stützle (Eds.): Ant Colony Optimization and Swarm Intelligence, 5th International Workshop, ANTS 2006, Brussels, Belgium, September 4-7, 2006, Springer, ISBN 3-540-38482-0 (2006)

9. H. Giese, N. Montealegre, T. Müller, and S. Oberthür : Acute stress response for self-optimizing mechatronic systems. In: IFIP Conference on Biologically Inspired Cooperative Computing (BICC06), Springer (2006)

10. T. Heimfarth, P. Janacik, and F. J. Rammig: Self-Organizing Resource-Aware Clustering for Ad Hoc Networks. In: Proc. of Software Technologies for Embedded and Ubiquious Systems (SEUS2007), LNCS vol. 4761, pp. 319 -328, Springer (2007)

11. T. Heimfarth, P. Janacik: Experiments with Biologically-InspiredMethods for Service Assignment in Wireless Sensor Networks. In: IFIP Conference on Biologically Inspired Cooperative Computing (BICC08), Springer (2008)

12. K.H. Kim: Object Structures for Real-Time Systems and Simulators. IEEE Computer, 30(8), pp. 325 – 333, IEEE (1997)

13. K.H. Kim, Y. Li, K.-W. Rim, and E. Shokri: A Hierarchical Resource Management Scheme Enabled by the TMO Programming Scheme. In: Proc. 11th Symposium on Object-Oriented Real-Time Distributed Computing (ISORC08), pp. 370 – 376, IEEE (2008)

14. H. Kopetz: Embedded System Complexity. In: Self-optimizing Mechatronic Systems: Design the Future. 7th International Heinz Nixdorf Symposium, pp. 469 – 486, HNI-Verlagsschriftenreihe (2008)

15. H. Kopetz: The Complexity Challenge in Embedded System Design. In: Proc. 11th Symp. on Object-Oriented Real-Time Distributed Computing (ISORC08), pp. 3 – 12, IEEE (2008)

16. N. Montealegre and F. Rammig: Immuno-repairing of FPGA designs. In: IFIP Conference on Biologically Inspired Cooperative Computing (BICC08), Springer (2008)

17. S. Oberthür and C. Böke: Flexible resource management - a framework for self-optimizing real-time systems. In B. Kleinjohann, G. R. Gao, H. Kopetz, L. Kleinjohann, and A. Rettberg, editors, Proceedings of IFIP Working Conference on Distributed and Parallel Embedded Systems (DIPES'04), pp. 177 - 186. Kluwer Academic Publishers (2004).

18. Tarakanov, A.O., Kvachev, S.V., Sukhorukov, A.V.: A formal immune network and its implementation for on-line intrusion detection. In: Computer Network Security, LNCS vol. 3685, pp. 394–405, Springer (2005)

19. L.N. de Castro and J. Timmis: Artificial Immune Systems: A New Computational Approach. Springer-Verlag, London. UK., (2002)

20. J. Timmis, P. Bentley, and E. Hart: Proceedings of the 2nd international conference on artificial immune systems. Number 2787 in LNCS 2787 (2003)