Complex Adaptive Systems (CAS) Approach to Production Systems and Organisations

László Monostori[1,2] and Balázs Csanád Csáji[1]

[1]Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest, Hungary
[2]Department of Manufacturing Science and Technology, Faculty of Mechanical Engineering,
Budapest University of Technology and Economics, Budapest, Hungary

Abstract

Theoretical study of complex systems receives more and more attention as most sciences broaden their perspectives. First, the paper overviews a few important complexity approaches, then it argues that complex adaptive systems (CASs) are especially important for production control research. As examples, firstly, a CAS based scheduling mechanism is described in which agents apply reinforcement learning to handle complex production control tasks, secondly, a semi-formal model of production networks is given combining stochastic processes, graph or network theory, and CASs. Further research issues are also highlighted.

Keywords:
Modelling, Production, Complexity, Adaptive Agents

1 INTRODUCTION

Growing complexity is one of the most significant characteristics of today's manufacturing, which is manifested not only in manufacturing systems, but also the products to be manufactured, in the processes, and the company structures [1]. The systems operate in a *changing environment* rife with *uncertainty*.

The need to be able to measure the complexity of a system, structure or problem and to obtain quantitative relations for complexity arises in more and more sciences: besides computer science and engineering, the traditional branches of mathematics, physics, chemistry, biology and social sciences are also confronted more and more frequently with this problem [2].

Complex Adaptive Systems (CASs) represent a relatively new theory, with the goal to study the structures and dynamics of systems and the question, how the adaptability of systems creates complexity [3], [4]. A CAS can be considered as a multi-agent system, where a major part of the environment of any given adaptive agent consists of other adaptive agents.

The main aim of the paper is to illustrate the appropriateness of the CAS approach for modelling different levels of production systems and organisations. Firstly, classical complexity measures are shortly surveyed, followed by enumerating some attempts to characterise the complexity in the manufacturing domain. The CAS approach is described shortly in Chapter 3, followed by chapters focusing on two different levels of the production hierarchy, namely the shop floor level and the level of the production networks. A CAS-based adaptive distributed production control is illustrated in Chapter 4, where the scheduling problem is formulated as a Markov decision process (MDP), and a three-level learning procedure is introduced. Chapter 5 gives a semi-formal model of production networks, combining stochastic processes, graph or network theory, and CASs. Further research issues conclude the paper.

2 COMPLEXITY AND ITS MEASURES

The meaning of the word "complexity" is vague, ambiguous, no universal, precise (e.g., formal) definition exists accordingly. Yet, there are approaches, especially in mathematics and computer science, which aim at defining special forms of complexity. In this section we provide a brief overview of some important classical complexity approaches.

2.1 Classical complexity measures

Since Alan Turing introduced his mathematical machines (viz., the Turing-machines) in the 1930s, they have become a fundamental tool for analyzing algorithms and problems. According to the theory of computational complexity, complexity is measured by the quantity of computational resources made use of by a particular task. Several complexity measures are known which are associated with algorithms [2], e.g., time-complexity, space-complexity and, for distributed systems, communication-complexity. Regarding the complexity of problems, the two most important classes of problems are P and NP. By definition, a P-problem is a decision problem that can be solved by a deterministic Turing machine in polynomial time. An NP-problem can be solved by a nondeterministic Turing machine in polynomial time. Roughly, problems in P are "easy" problems, while problems in NP are "hard" ones. Naturally, this classification is a simplification. We know that any P-problem is also an NP-problem. However, the question, whenever P = NP, is currently the most important problem of computational complexity theory.

Information complexity, viz. entropy, tries to measure the randomness or disorder of objects. This approach was suggested by Claude Shannon who in 1948 introduced entropy to communication theory [5]. Entropy provides a measure of the amount of information associated with the occurrence of given states. This is of key importance in information- and code-theory, however, can also be applied to measure other complex systems, e.g., graphs or networks.

Note that Shannon himself borrowed the concept of entropy from physics (viz. thermodynamics) by using the Boltzmann-Gibbs formulation of entropy.

In the 1960s Solomonoff, Kolmogorov and Chatin (independently) introduced a complexity concept which is often called algorithmic information complexity. Given a universal Turing-machine, the Kolmogorov complexity of a (bit)string (description) is the length of the shortest program that generates the description and halts. In other words, Kolmogorov defined the complexity of a structure as the length of its shortest description (namely, on a universal Turing machine). A structure is "simple", if it can be described by a short program, and "complex", if there is no such a short description. As an example, a random string is complex since its shortest description is specifying it bit-by-bit.

Also in the 1960s Krohn and Rhodes introduced a complexity definition aiming at measuring the complexity of abstract algebraic structures, e.g., groups and semigroups with the concepts of homomorphisms and wreath products. In computer science the Krohn-Rhodes theory gave new methods to build any finite state automaton using series-parallel emulation by simple components.

In the second half of the 20th century it became more and more important to measure the complexity of structures in natural sciences (e.g., in chemistry and biology). The theory of topological or network complexity addresses this problem and applies graph theory as its basis. There are several measures to define the complexity of a graph, some of which will be investigated later.

One of the latest complexity approaches is the theory of Complex Adaptive Systems (CASs). It has deep roots in the interdisciplinary field of multi-agent systems [6]. They will be investigated in Chapter 3.

2.2 Modelling of complexity in manufacturing

The complexity of manufacturing processes and systems, namely, how to deal with it, how to measure it, how to manage or control it, have come in the foreground of research in the past decades (e.g., [1], [6], [7], [8]). Decoupled models of product and process complexities were described in [6] where the quantity of information, diversity of information and the information content were considered as main influencing factors. As a continuation from the same authors, manufacturing operational complexity was also assessed addressing the human element of the production, too [7]. A complexity coding system for the main elements of a manufacturing system, i.e., machines, buffers and material handling equipment, was introduced in [8].

As regard to supply chains or production networks, even bigger complexity is to be faced. In [9] a focal buyer company with its supply base (the part of the supply network which is actively managed by the focal company), was treated. In the supply base complexity was conceptualized in three dimensions: the number of suppliers in the base, the degree of differentiation among them, and the level of inter-relationships among the suppliers. The authors concluded that though a reduction in complexity may lead to lower transaction costs and increased supplier responsiveness, in certain circumstances it may also increase supply risk and reduce supplier innovation. Consequently, reducing supply base complexity, in general, may be cost-effective, but blindly reducing it, may potentially decrease the buyer's overall competitiveness. The operational complexity of supplier-customer systems is modelled from an information-theoretic perspective in [10], capturing, in relative terms, the expected amount of information to describe the state of the system. The operational complexity was found to be associated with the operational costs of running a supply chain, as shown by queuing model- and simulation-based investigations [11]. However, dependency between the complexity index used and the costs was found in case of make-to-stock production, but not in the make-to-order case [11].

3 COMPLEX ADAPTIVE SYSTEMS

The theory of *Complex Adaptive Systems* (CASs) which was put forward by Holland [3], [4] is a new paradigm with the goal to study the structures and dynamics of systems and the question, how the adaptability of systems creates complexity. A CAS can be considered as a multi-agent system with seven basic elements in which "a major part of the environment of any given adaptive agent consists of other adaptive agents, so that a portion of any agent's efforts at adaptation is spent adapting to other adaptive agents". The first four concepts of Holland's seven basic elements, i.e., aggregation, nonlinearity, flow and diversity, represent certain characters of agents, are very important in the adaptation and evolution process, while the other three concepts, i.e., tagging, internal models and building blocks, are mechanisms of agents for communicating with the environment.

Environmental conditions change, due to the agents' interactions as they compete and cooperate for the same resources or for achieving a given goal. This, in turn, changes the behaviour of the agents themselves. The most remarkable phenomenon exhibited by a CAS is the emergence of highly structured collective behaviour over time from the interactions of simple subsystems [12]. The emergence of a complex adaptive behaviour from the local interactions of the agents is demonstrated in Figure 1.

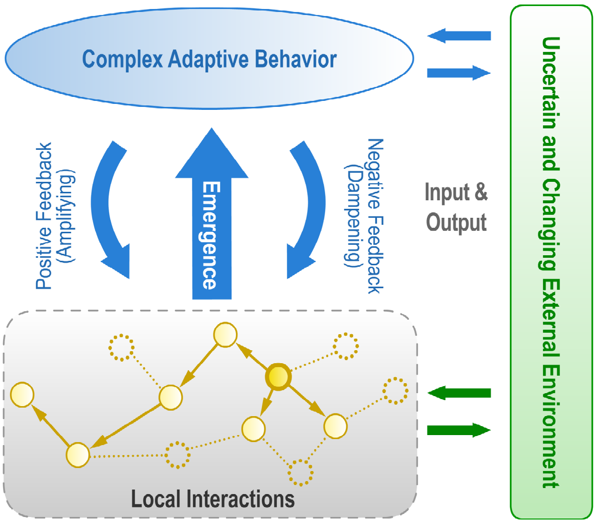

Figure 1: Emergence in Complex Adaptive Systems.

Both the CAS and its environment simultaneously co-evolve in order to maintain themselves in a state of quasi-equilibrium, i.e., on the edge of chaos [13].

In designing CAS, non-linear phenomena, incomplete data and knowledge, a combinatorial explosion of states, dynamic changes in environment and the frame problem are some notable examples of difficulties to be faced. The central question is realising an artifactual system that achieves its purpose in unpredictable conditions. It is difficult to approach problems like this by using only existing principles, such as analysis and determinism [14].

The Webster Dictionary describes *synthesis* as putting parts or elements together so as to form a whole, or the combination of separate elements of thought into a whole, as of simple into complex conceptions, species into genera, individual propositions into systems.

Synthesis is a necessary component of problem solving processes in almost all phases of the artifacts' lifecycle that starts with design, goes through the phases of planning, production, consuming and ends with the disposal of the product. Emergence plays a key role in solving difficult problems arising in synthesis.

The main concern here is whether and when, the completeness of information could be achieved in the description of the environment and in the specification of the purpose of the artifactual system. With respect to the incompleteness of information on the environment and/or the specification, the difficulties in synthesis can be categorised into three classes [15], [12]:

- *Class I*: Problem with complete description: if all the information concerning the environment and specification are given, then the problem is completely described. However, it is often difficult to find an optimal solution.
- *Class II*: Problem with incomplete environment description: the specification is complete, but the information on the environment is incomplete. Since the problem is not wholly described in this case, it is difficult to cope with the dynamic properties of the unknown environment.
- *Class III*: Problem with incomplete specification: not only the environment description but also the specification is incomplete. Problem solving, therefore, has to start with an ambiguous purpose, and the human interaction becomes significant.

3.1 Complex Adaptive Systems as Multi-Agent Systems

Complex Adaptive Systems have deep roots in the interdisciplinary field of *Multi-Agent Systems* (MASs) of artificial intelligence research, [4], [16], [17]. The formers are special cases of the latters which represent a general and flexible framework to describe and model (partially) autonomous systems including their interactions [16]. An agent is basically a self-directed entity, it is an object with its own value system and a means to interact (e.g., communicate) with other objects like this.

A MAS is formed by a network of computational agents that interact and typically communicate with each other. In *hierarchical* (guided) architectures, there are multiple levels of subordination relationships. In a *heterarchical* (self-organized, organic) architecture, agents communicate as peers, no fixed subordination relationships exist and, usually, the global information is eliminated, and, consequently, global optima cannot be guaranteed. On the other hand, advantages of these heterarchical multi-agent systems include: self-configuration, scalability, fault tolerance, emergent behaviour, massive parallelism, reduced complexity, increased flexibility, and reduced cost [18].

What distinguishes a CAS from a MAS is the focus on top-level properties and features like self-similarity, complexity, emergence and self-organization. Basically a CAS consists of large numbers of diverse entities (for example, agents) that are interconnected and have the capacity to change and learn from experience [4], [16]. Many of our most difficult problems centre on CAS.

4 ADAPTIVE AGENTS AT THE SHOP FLOOR LEVEL

4.1 Multi-agent systems in manufacturing

Engineering design; process planning; production planning and resource allocation; production scheduling and control; process control, monitoring and diagnostics; enterprise organisation and integration; production networks; assembly and life-cycle management; were enumerated in a very recent survey on agent-based systems in manufacturing [18]. The application fields cover all important aspects of manufacturing indicating the viability of the MAS-approach.

Agent technology is considered an especially important approach for developing distributed manufacturing systems. Holonic manufacturing systems (HMSs) consist of autonomous, intelligent, flexible, distributed, co-operative agents or holons [19]. One of the most promising features of the holonic approach is that it represents a transition between fully hierarchical and heterarchical systems [20]. HMSs with *adaptive agents* received a great deal of recent attention [21].

4.2 CAS-based adaptive distributed production control

In practice, the agents mostly have only incomplete and uncertain information on the environment (surrounding world) that they have to work in, additionally, this environment could be non-stationary. Moreover, they also have to face algorithmic-complexity problems, viz. even if they deal with static, highly simplified and abstract problems in which the solution is surely exists and can be attained in finitely many steps, they may not have enough computation power to achieve it in practice (as this is the case, for example, with many NP-hard problems).

A promising way to overcome these difficulties is the application of machine-learning techniques. It means designing systems which can adapt their behaviour to the current state of the environment, extrapolate their knowledge to the unknown cases and can learn how to optimize the achieved solutions.

Several complex systems can be effectively dealt with approximation. This means that we use a solution/model which is ”close” to the one that we have originally aimed at, however, it is much more simpler. For example, a lot of very-hard/complex (combinatorial) optimization problems can be efficiently handled, if we satisfy with approximate solutions, viz. solutions which are not optimal (suboptimal) but close to an optimal one. Naturally, the effectiveness of the approximation strongly depends on the used architecture and on the way we measure the distance between the objects.

Now, an adaptive iterative distributed scheduling algorithm is proposed that operates in a market-based production control system. The idea of negotiation based scheduling has emerged long before [22].

First, the basic frame of the approach is informally defined. In a multi-agent based manufacturing system, autonomous agents control different real world entities. In the presented system the two most important types of agents are the *resource agents* and *the order agents*. Resource agents control physical parts (such as machines, furnaces, conveyors, pipelines, material storages, etc.), while order agents control the production of a job. In the presented market-based production control system if a new job arrives at the system, a new order agent is created and associated with that job. An order agent or a group of cooperating order agents announces a sequence of operations and the resource agents can bid for that sequence. Only resource agents being able to do at least the first operation of that job are allowed to bid. Before an agent bids, it gathers information about the possible costs of making that sequence. If the sequence contains only one operation, the agent has all the information it needs, however, if the sequence contains other operations as well, which probably cannot be processed by the machine of the agent, it starts to search for subcontractors. It becomes a partial order agent and announces the remaining part of the sequence. The other resource agents which can do the next operation, may bid for the remaining operation sequence. Consequently, *a recursive announce-bid process* begins. At the end, when all the possible costs of that (partial) job are known, the agent bids. If the order agent which announced that job, is contented with it (it is the best bidder), the agent (and its subcontractors) get the job (award). Therefore, the schedule generation in the suggested agent-based system is a recursive, iterative process with announce-bid-award cycles based on market mechanisms.

The main problem with the mechanism described above is the *combinatorial explosion* of the possible schedules. More precisely, it makes a complete enumeration, in some sense, and thus, its time complexity makes it unusable in practice. The agents should not investigate every potential schedule, because this can be extremely time-consuming. If an agent wants to bid for an operation sequence and it needs information about the production costs of the part of the job, which it cannot do, it should not announce the part to every resource agent. It should make only a restricted tendering among the agents that will give a presumably good bid. They can apply *adaptive sampling* to learn the potentially good partners to cooperate with. These partner-value estimations can be learnt with *neurodynamic programming*, which is the combination of *reinforcement learning* and artificial neural networks, especially *kernel machines*.

Markovian Production Control

It can be shown that this scheduling approach can be formulated as a special *Markov decision process* (MDP) [17]. The aim of learning in an MDP is to find an optimal (or approximately optimal) *policy* that maps the *states* (possible situations) to the *control actions* available in that state. Formally, if we denote the state by x and the set of actions available in the state by $U(x)$ then an action a (form the $U(x)$ set) is executed by $\pi(x,a)$ probability, where π is called a control policy. In an MDP the state transitions are stochastic, however, the *Markov property* is assumed. There is a *reward* $r(x,a)$ associated with each state-action pair and the aim is to find such a policy that optimize the *expected cumulative rewards* over time [23]. The performance of a control policy in the long run is specified by its the *value function*. The value of a state with respect to a given policy is, roughly, the total amount of reward an agent can expect to gather starting from that state and following the policy thereafter. In case of *finite* MDPs, there always exists at least one optimal policy and each such policy shares the *same* optimal value function.

In theory, the optimal control policy of a (finite) MDP can be exactly computed by *dynamic programming* methods, such as value iteration, policy iteration or the Gauss-Seidel method [23]. However, due to the "*curse of dimensionality*" (namely, in practical situations both the required memory and the amount of computation is extremely large) calculating an exact optimal solution by these methods is practically infeasible, except for very small problems. Reinforcement learning techniques often try to use simulation as a sampling technique to overcome the computational demands by advanced statistical approaches, such as MCMC.

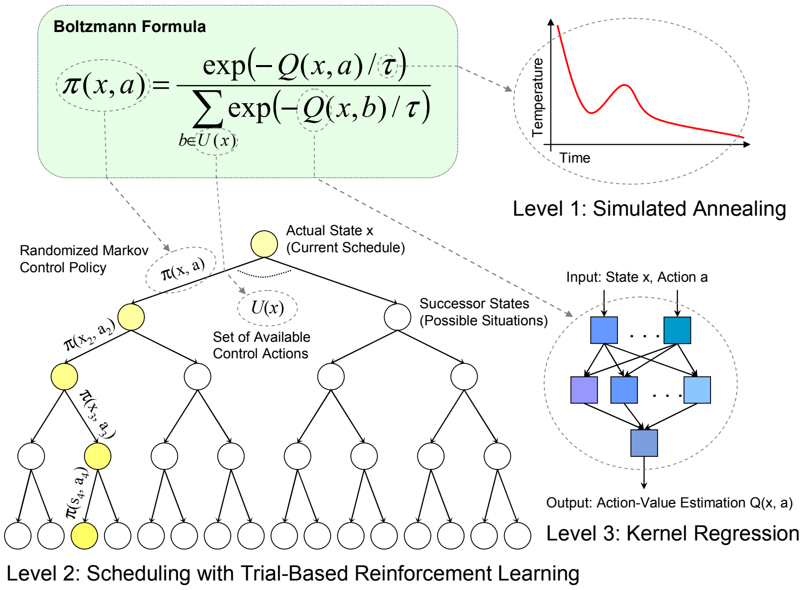

Figure 2: The three main levels of learning combined by the Boltzmann formula.

Three Levels of Learning

The paper suggests using Watkins' Q-learning algorithm [23] to calculate a near optimal policy. The aim of Q-learning is to learn the optimal *action-value function* Q^* rather than directly learning an optimal policy. The system can search in the space of feasible schedules by *simulating* the possible occurrences of the production process with the model. The trials can be described as state-action pair trajectories. After each sample (trial, iteration) the system makes updates asynchronously on the approximated values of the visited pairs according to the Q-learning rule. Only a subset of all pairs is updated in each trial. The paper suggests a triple-level learning mechanism to achieve effective production control, see Figure 2. The most important learning level is the level of reinforcement learning, since it computes an approximately optimal policy through adaptive sampling. The action-value function is represented as a kernel machine and the temperature of the system is controlled by a Metropolis algorithm. The progress of the system during learning can be described as follows (more details and examples are in [17]):

- Simulate a sate-action trajectory from the starting state using order and resource agents and a policy generated by the Boltzmann formula.
- After a terminal state is reached, back propagate the final performance and update the Q estimates of the visited states according to Q-learning.
- Fit a smooth approximating function to all of the available Q estimates, e.g. by SVR type Kernel Regression (Gaussian kernels).
- Decrease the temperature, except there were changes and disturbances in the system (in that case increase the temperature).
- Increase iteration counter and, unless some terminating conditions are met, go back to the first step.

5 A SEMI-FORMAL MODEL OF PRODUCTION NETWORKS

The conceptual overview of the proposed three-layer model can be found in Figure 3.

5.1 The environment as a stochastic process

We suggest a very abstract environment model: the uncertain behaviour of the environment should be described by a multivariate random variable and, since the environment may change over time, we consider a sequence of variables like this, one for each observable time step. These sequences are called *stochastic processes* [24].

Stochastic processes are standard models used in statistics, signal processing and machine learning. They consist of a sequence of random variables, $X_1, X_2, \ldots, X_{t-1}, X_t, X_{t+1}, \ldots$, where each X_t is a random variable, a measurable function from the sample space of a probability measure space to a measurable space of possible outcomes. They describe an event which is uncertain from the viewpoint of the observer. Multivariate random variables have vector output, namely, they render several values to an element of the sample space. They can be adequately described by their distributions. In the model proposed the environment is considered as a stochastic process, which may be assumed to be stationary or Markovian, i.e., we do not concern with the inner structure and the internal dependencies of the environment. It is treated as a black box, however, we still have a formal statistical model to work with.

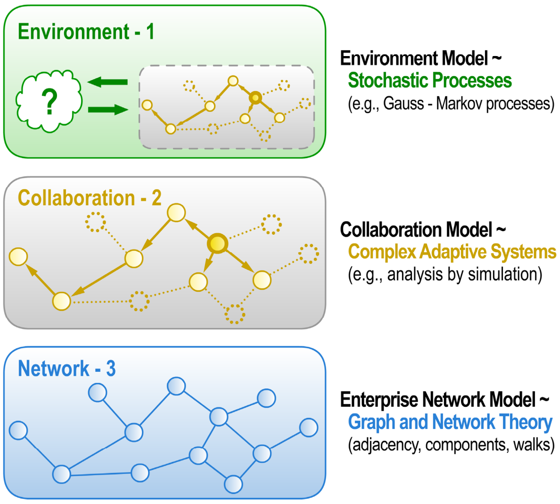

Figure 3: Conceptual overview of the proposed model.

In our model at each time *t* the state of the environment can be described with a multivariate random variable, X_t, and $X_{t,i}$ is a single valued random variable that describes a particular aspect of the environment we want to take into account. Such aspects are, e.g., the number of requests, the number of products that the customers have ordered, due dates, the trustiness of the customers, external costs, the economic situation (e.g., interest and currency rates, asset prices) or even the social, the cultural and the political situation.

As to measuring the complexity of random variables, the concept of (differential) entropy can be applied [5]. Note that the probability distribution of a random variable can be estimated by using accumulated historical data.

5.2 Modelling structures with network theory

It is assumed that the core topology of an enterprise network is quasi-static or slowly varying, hence it can be adequately modelled with network theory.

The basis of modern network theory [25], [26] and, hence, of network- or topological complexity is graph-theory, which is one of the fundamental theories in discrete mathematics. Its history goes back to Euler's celebrated solution of the Königsberg bridge problem in 1735.

The elements of graphs can be naturally associated with the elements of an enterprise network. An association, for example, could be as follows

- vertices ~ e.g., companies or functionalities,
- edges ~ e.g., connections between companies, functionality associations,
- vertex weights ~ e.g., prod. capabilities,
- vertex labels ~ e.g., competences,
- edge labels ~ e.g., connection type,
- edge weights ~ e.g., collaboration strength,
- colors, labels, weights ~ e.g., roles, types, strengths.

Network theory offers many "off-the-shelf" complexity measures that could also be applied to measure the complexity of the core-structure of enterprise networks [26], e.g., adjacency-related measures, symmetry-based measures, entropy-related measures, component-related measures, total walk count, the A/D Index. By applying results from network theory, measures for the static description of a network can be immediately used.

5.3 Collaborations as Complex Adaptive Systems

Probably the most important elements of an enterprise collaboration are dynamic and, therefore, hard to model and analyze. Here we suggest modelling the dynamic behaviour of an enterprise network as a CAS (Chapter 3).

6 CONCLUSIONS

The paper argued that the CAS approach is viable at different levels of manufacturing. Special emphasis was laid, on the one hand, on CAS-based production scheduling and control, by reinforcement learning, and, on the other hand, a semi-formal model of production networks, combining stochastic processes, graph or network theory, and CASs, illustrating the appropriateness of the CAS-based approach for handling complex, production-related tasks.

Complex adaptive systems, however, exhibit patterns of behaviour that can be considered archetypal or prototypical. One can benefit from the knowledge of these patterns [28]. Managing such systems an appropriate *balance between control and emergence* must be found [28]. The difficulty in understanding the effects of individual characteristics of the agents on their collective behaviour underlines the importance of using simulation as primary tool for designing and optimising such systems. In this respect, the proper *balance between simulation and theory* is to be aimed at [29].

Our further research activities will also go in this direction, i.e., to find the appropriate balances between simulation and theory, on the one hand, and between control and emergence, on the other.

ACKNOWLEDGEMENTS

The work was supported by the Grant “Production structures as Complex Adaptive Systems” of the Hungarian Scientific Research Fund (OTKA) and by the “Coll-Plexity” project belonging to the 6th Framework Programme of the EU.

REFERENCES

[1] Wiendahl, H.P., Scholtissek, P., 1994, Management and Control of Complexity in Manufacturing, Annals of the CIRP, 43/2: 533-540.

[2] Lovász, L., Gács, P., 1999, Complexity of Algorithms. Lecture Notes, Boston Univ., Yale University.

[3] Holland, J.H. (1992). Complex Adaptive Systems, Daedalus, Boston, 121/1: 17-30.

[4] Holland, J.H. (1995). Hidden Order: How Adaptation Builds Complexity, Helix Books, Addison-Wesley, NY.

[5] Shannon, C. E., 1948, A Mathematical Theory of Communication, Bell System Tech. J., 21: 623–656.

[6] ElMaraghy, W.H., Urbancic, R.J., 2003, Modeling of Manufacturing Systems Complexity, Annals of the CIRP, 52/1: 363-366.

[7] ElMaraghy, W.H., Urbancic, R.J., 2004, Assessment of Manufacturing Operational Complexity, Annals of the CIRP, 53/1: 401-406.

[8] ElMaraghy, H.A., Kuzgunkaya, O., Urbancic, R.J., 2005, Manufacturing Systems Configuration Complexity, Annals of the CIRP, 54/1: 445-450.

[9] Choi, T.Y., Krause, D.r., 2006, The Supply Base and its Complexity: Implications for Transaction Costs, Risks, Responsiveness, and Innovation, J. of Operations Management, 24:637-652.

[10] Sivasadan, S., Efstathiou, J., Calinescu, A., Huatuco, L.H., 2006, Advances on Measuring the Operational Complexity of Supplier-Customer Systems, European J. of Operational Research, 171:208-226.

[11] Wu, Y., Frizelle, G., Efstathiou, J., 2007, A Study on the Cost of Operational Complexity in Customer-Supplier Systems, Int. J. of Prod. Economics, 106:217-229.

[12] Ueda, K., Márkus, A., Monostori, L., Kals, H.J.J., Arai, T., 2001, Emergent Synthesis Methodologies for Manufacturing, Annals of the CIRP, 50/2: 535-551.

[13] Waldrop, M. (1992). Complexity, the Emerging Science at the Edge of Order and Chaos, VIKING, Penguin group, London.

[14] Monostori, L., Ueda, K., 2006, Design of Complex Adaptive Systems: Introduction, Advanced Engineering Informatics, Elsevier, 20/5: 223-225.

[15] Ueda, K., Vaario, J., 1998, The Biological Manufacturing System: Adaptation to Growing Complexity and Dynamics in Manufacturing Environment, CIRP J. of Manuf. Systems, 27/1: 41-46.

[16] Holland, J. H., 1998, Emergence: from Chaos to Order, Cambridge (MA), Perseus.

[17] Csáji, B. Cs., Monostori, L., Kádár, B., 2006, Reinforcement Learning in a Distributed Market-Based Production Control System, Advanced Engineering Informatics, Elsevier, 20/5: 279-288

[18] Monostori, L., Váncza, J., Kumara, S.R.T., 2006, Agent-based Systems for Manufacturing, Annals of the CIRP, 55/2: 697-720.

[19] Valckenaers, P., Van Brussel, H., 2005, Holonic Manufacturing Execution Systems, Annals of the CIRP, 54/1: 427-432.

[20] Bongaerts, L., Monostori, L., McFarlane, D., Kádár, B., 2000, Hierarchy in Distributed Shop Floor Control, Computers in Industry, 43/2: 123-137.

[21] Monostori, L., Kádár, B., Csáji, B.Cs., 2002, The Role of Adaptive Agents in Distributed Manufacturing, Proc. of the 4th International Workshop on Emergent Synthesis (IWES'02), Kobe, Japan, 135-142

[22] Márkus, A., Kis, T., Váncza, J., Monostori, L., 1996, A Market Approach to Holonic Manufacturing, Annals of the CIRP, 45/1: 433-436.

[23] Bertsekas, D.P., Tsitsiklis J.N., 1996, Neuro-Dynamic Programming, Athena Scientific

[24] Papoulis, A, Pillai, S.U., 2001, Probability, Random Variables and Stochastic Processes. McGraw-Hill.

[25] Barabási, A.-L.. 2002, Linked: The New Science of Networks. Perseus, Cambridge.

[26] Newman, M. E. J., 2003, The Structure and Function of Complex Networks. SIAM Review, 45/2: 167–256.

[27] Monostori, L., Csáji, B.Cs., 2007, Production Structures as Complex Adaptive systems. Proc. of the 40th CIRP Int. Sem. on Manufacturing Systems, Liverpool, UK

[28] Choi, T.Y., Dooley, K.J.,Rungtusanatham, M., 2001, Supply Networks and Complex Adaptive Systems: Control versus Emergence. J. of Op. Man., 19: 351-366.

[29] Surana, A., Kumara, S., Greaves, M., Raghavan, U.N., 2005, Supply-chain Networks: A Complex Adaptive Systems Perspective. International Journal of Production Research, 43/20: 4235–4265.