Proceedings of the World Congress on Engineering and Computer Science 2007
WCECS 2007, October 24-26, 2007, San Francisco, USA

A Basic Approach Towards Cognitive Production Systems

Christian Brecher, Tobias Kempf, Frank Possel-Dölken, Tilman Buchner
Laboratory for Machine Tools and Production Engineering (WZL) at RWTH Aachen University

Abstract — **Particularly countries with high labor costs always demand for higher productivity of manufacturing processes and increasing process flexibility at the same time. Therefore the dilemma between a planning-orientated manufacturing and a value-orientated manufacturing approach has to be resolved. Based on the insight that this request has remained unsatisfied so far, this paper presents a possible approach harnessing cognitive control architectures. The research questions that arise from this approach are presented and discussed in detail.**

Index Terms — **Cognitive architectures, flexible manufacturing systems, intelligent automation, self-optimizing production systems**

I. Introduction

The superior goal with regard to the design and operation of production systems is to find (nearly) optimal operating points which result in low unit costs over varying batch sizes. There have been many efforts in the past to shift the curve of unit costs further down. But the demand for more flexibility requires new methods which aim at flattening the curve in order to widen the sufficiently optimal "operating range" of the production system to a larger batch size range (see Fig. 1).

To reach the goal of a flat curve of unit costs, the two main dilemmas of production technology have to be solved. These are on the one hand the dilemma between *scale* and *scope*. On the other hand there is the dilemma between being *planning-oriented approaches* and *value-oriented approaches*.

A production system that aims at *economies of scope* realizes the one-piece-flow and is highly flexible. In contrast to that a production that is geared towards the *economies of scale* will gain cost advantages by concentrating on highly synchronized, mastered processes with high repetition frequencies. Increasing the product flexibility in this context is generally expensive and sometimes technically not feasible.

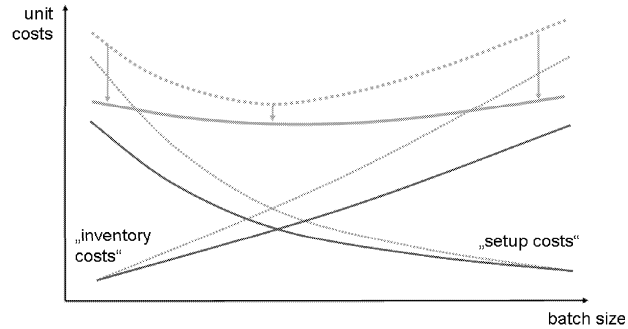

Fig. 1: Relationship between unit costs and batch size

The second dilemma can be localized between "lean" shop-floor-oriented production with a clear focus on value-adding processes and a highly optimized, planning-oriented production. The combination of both dilemmas makes up the so-called *polylemma of production* [1] (see **Error! Reference source not found.**). The research in production technology aims at the elimination of this polylemma which means to widen and minimize the curve of unit costs at the same time. Numerous methods to resolve the polylemma result in higher efforts within the virtual chain (which means: on the planning side). These are no value-adding activities. For this reason different methods are needed to address the dilemma between being planning-oriented and value-oriented approaches. The simplification or even elimination of planning tasks requires an increasing level of intelligence for the control of value-adding processes. This objective leads to the concept of *self-optimization* (see Fig. 3). According to [2] we can speak of self-optimization when the following steps are conducted continually:

- Analysis of the system's actual situation,
- Assigning of system objectives,
- Modification of system behavior.

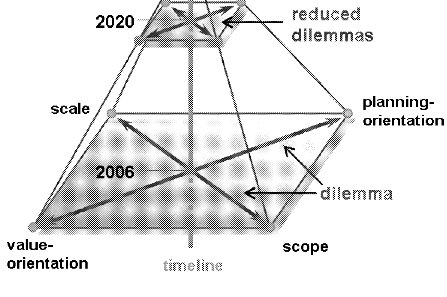

Fig. 2: The polylemma of production

Manuscript received July 20, 2007. This work is supported by the German National Science Foundation DFG within the framework of the Cluster of Excellence "Integrative Production Technology for High Wage Countries".

Prof. Dr.-Ing. Christian Brecher is director of the Chair of Machine Tools, which is part of the Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, 52056 Aachen, Germany, (e-mail: c.brecher@wzl.rwth-aachen.de).

Dipl.-Ing. Tobias Kempf is with the Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, 52056 Aachen, Germany, (e-mail: t.kempf@wzl.rwth-aachen.de).

Dr.-Ing. Frank Possel-Dölken is the managing director of the Cluster of Excellence "Integrative Production Technology for High Wage Countries", RWTH Aachen University, 52056 Aachen, Germany, (e-mail: f. possel-doelken@wzl.rwth-aachen.de).

Dipl.-Ing. Tilman Buchner is with the Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen University, 52056 Aachen, Germany, (e-mail: t.buchner@wzl.rwth-aachen.de).

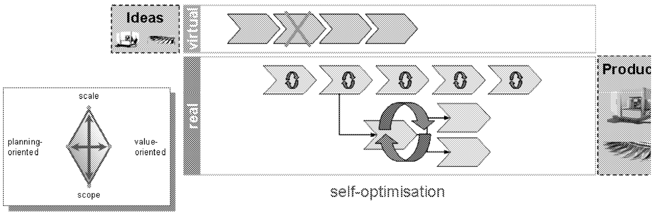

Fig. 3: Self-optimization as a means to reduce planning efforts

A system that is capable of self-optimization should have the ability to recognize interrelations between facts and to analyze the current situation in order to reach certain goals that might even be subject to change. The resulting question is: to what extent does today's automation meet this demand?

II. Survey of Today's Automation

Throughout the CIM-euphoria of the 1980s companies tried to massively support the flow of information within production processes by means of IT-technologies. The expectations to make processes far more flexible and at the same time keep them robust and easy to maintain have largely been disappointed. Various attempts to integrate knowledge-based systems for design and diagnosis, production scheduling systems or adaptive control technologies were undertaken and yet have proven to be mostly inadequate. Now – 20 years later – it has to be admitted, that automation technology is well-suited to master fixed processes at high rates. But constraints, objectives or task flows that are exposed to variations can hardly be handled and have to be incorporated in complex planning phases or even require the direct intervention of the human expert during runtime. This is certainly true for **enterprise level** systems like ERP or SCM that have to integrate a lot of heterogeneous knowledge from all kinds of sources that may even be located outside the system borders of a company. But on our way downwards the pyramid of automation we still find this statement to be true. Regarding **manufacturing execution** systems (MES) there is a trend of using IT-technologies and standards and also decentralization efforts can be noticed (e.g. multi-agent systems, [3]). But besides the fact, that there are no universal data models, today's solutions lack the intelligence for autonomous scheduling. At the **cell level** there is almost no permeability between single subsystems that are characterized by traditionally isolated applications (which is true for engineering tools as well as for runtime systems). An optimization with respect to a given task thus remains up to the expertise and integrative capabilities of the human operator who does the commissioning and programming. At the **machine level** one has to deal with the problem that information from the planning level (e.g. CAD/CAM) has been thinned out already. In addition, important state information of the plant is not available and – even if it was – interrelations between machines, tools, workpieces and the manufacturing processes are far from being understood. Only in robotics one might concede moderate efforts to implement "fuzzy" processes by the use of external sensors like vision systems (e.g. [4]). But a shift of extensive planning activities into the production process, as it is subject of research for assembly tasks, e.g. [5], can only rudimentarily be found.

III. Intelligent Automation: A Research Vision

Our research vision is a control technology that is capable of aggregating distributed knowledge for a given problem domain and of reasoning on this knowledge-base in a flexible and robust manner. This *principle of self-organization* contrasts the status quo, in which the human expert has to explicitly bring together all relevant resources and also has to anticipate all potential situations that the system will have to undergo. An intelligent control system however would be able to emulate human patterns of behavior to some extent, in order to autonomously find solution paths and match them with the human expert or other resources within the production environment. Observations of the own history and that of interaction with the human expert should enable the system to integrate fuzzy, incomplete or even contradictory information to a coherent overall picture. As a result the system would be able to reach an adequate and stable behavior (*principle of self-optimization*).

To gain a massive reduction of not-value-adding planning processes an automation solution should be as general as possible. That means that we will have to look for generic methods to implement the principles of self-organization and self-optimization in production technology which are valid for different problem descriptions (see Fig. 4). Such problem descriptions are:

- optimization of products according to objective functions and cost functions over the entire process chain
- optimization of the scheduling of manufacturing jobs to different work cells with respect to resources and time constraints
- optimization of the commissioning of a robot cell
- optimization of workflows in a particular machine
- optimization of the behavior of a particular mechatronic module

This leads to the following research question: Are there any general models or methods for self-optimization that can be introduced systematically into production systems in order to reduce planning efforts on the virtual side? And what might the practical implementation within the scope of automation technology look like?

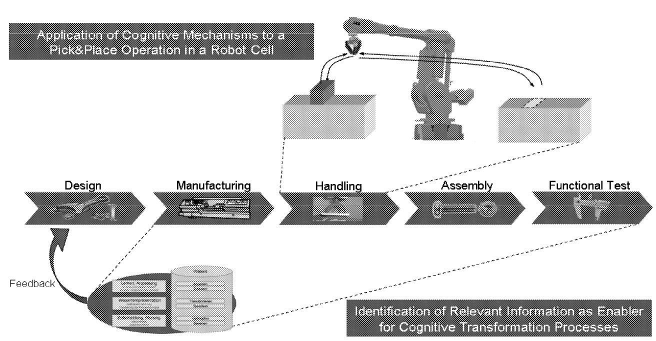

Fig. 4: Different problem domains in production systems

IV. TOWARDS COGNITIVE CONTROL: A RESEARCH PROGRAM

The basic approach is based on the notion that a system which follows the vision stated above can be considered as being intelligent. The idea is to search for models that are suitable to describe intelligence as comprehensive as possible. Then the question is posed, how these models can be transported to technical – particularly manufacturing – systems. This approach requires a viewpoint that comprises several scientific areas. Cognitive sciences have been dealing for some decades with the question of how to describe information processes in human beings. Some areas in computer sciences even proceed to another question, which is based on the first one: Can these models be exploited to inversely generate intelligence in technical systems?

The models hat have been elaborated are known as **cognitive architectures**. They all have in common a distinction between a general structure which comprises several mechanisms and the specific content that is responsible for the systems adequate behavior within a particular domain. Despite this commonality it is quite obvious that different architectures are based on completely different paradigms. The "cognitivist paradigm" goes back to the basic assumption formulated by [6]: Intelligent systems are formal symbol systems that follow fixed syntactical rules. The grounding of the symbols in the real world is done by the programmer which means that he – and only he – gives meaning to the symbols by assigning them to entities and their interrelations. In doing so he creates a representational framework to reflect his knowledge for a certain domain. On behalf of this symbolic representation the cognitive system can operate computationally by means of rules or algorithms in order to solve a particular problem. A system like this can even learn by modifying existing rules or creating new rules or facts. Architectures that are based on that kind of model are typically hierarchical with a clear distinction between several functional modules that are run through sequentially. This paradigm – which we might call the "classic" way – has borne some *Unified Theories of Cognition (UTC)*. These theories claim to combine several micro-theories that explain sub-phenomena within the scope of different research fields. Two of the most outstanding representatives for such architectures are SOAR [7] and ACT-R [8]. These architectures offer the advantage of maximum transparency towards the user and their capability for deliberative behavior. A major drawback lies in the fact that they are virtually unable to cope with unforeseen situations.

The "cognitivist paradigm" is opposed by the "emergent paradigm". This method puts the emphasis on the system's embeddedness into the real-world environment. An emergent system has to prove its efficiency by autonomously organizing itself in the real world that is subject to continual change – a quality which is called *dynamical situatedness*. The knowledge about this environment or even the knowledge about the system itself is not explicitly represented anywhere. It just emerges by the implicit coaction of microstructures. Artificial neural networks are a typical example. Another important representative

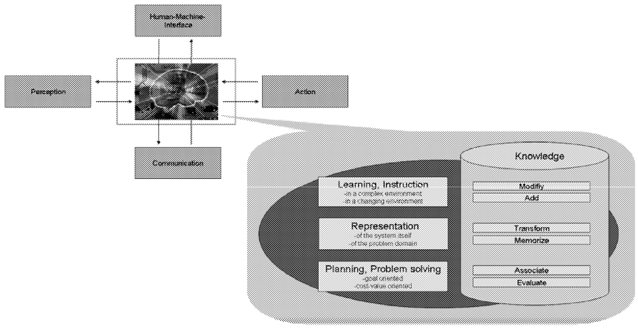

Fig. 5: Basic concept of a cognitive architecture

can be found in *behavior based robotics*. The original paradigm was formulated by [9] and has been picked up and more or less modified in numerous robotic applications since then (a quite recent example that owes significant features to this method is [10]) Unlike the knowledge-oriented cognitivist way the behavior based method stresses the development of skills that ultimately lead to adequate behavior. That means again, that behavior emerges by the implicit cooperation of parallel subprocesses and not – as in the cognitivist view – by functional decomposition which leads the system in a hierarchical manner from the detection of an inner state to any action towards a supposed goal. Emergent systems quite obviously are more reactive and robust than cognitive systems. These advantages are counter-balanced by the fact that we have little transparency which ultimately means that emerging behavior is non-determinate. Moreover action selection is hard to achieve since there is a lack of a central unit. Any interactions between parallel processes have to be established by a high amount of communication efforts.

For automation purposes hybrid methods seem to be the most promising alternative. They try to find combinations of the advantages of both sides while avoiding the drawbacks (a somehow pragmatic idea which – of course – is subject to philosophical debates within the cognitive scientific community). Examples for this can be found in quite a number of technical areas (e.g. [11], [12]).

For a knowledge-intensive field like production technology an explicit representation of knowledge is indispensable. For that reason the proper **modeling of domains** is of very high significance. The objective of a model is, to represent semantically the facts about the environment, the system`s capacities and the objectives to follow. To get there, all relevant sources of knowledge have to be exploited. By means of a thorough analysis all entities and their relations need to be structured and classified. Object descriptions, constraints, system states, task flows have to be merged into a coherent semantic representation with a granularity that corresponds as exactly as possible to the problem.

After that the representation of the knowledge may take place using adequate notations which support the knowledge base being operated on computationally. In order to get there, concepts of semantic modeling that have been used in bordering areas have to be taken into consideration for their possible ap-

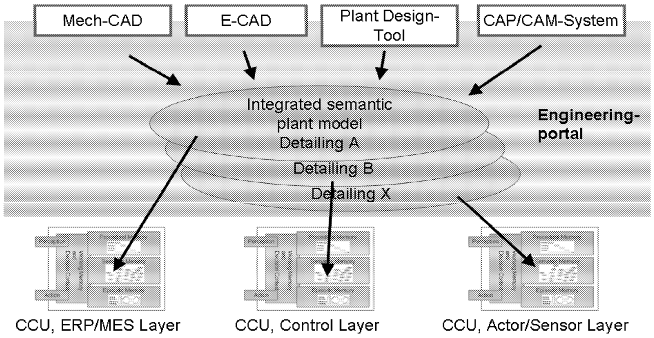

Fig. 6: Semantic modeling integrating multiple sources of knowledge

plicability (e.g. [13]). Besides, general methods offered by computer sciences need to be evaluated (e.g. first-order-logic, frames, etc.).

A major challenge is the integration of computational knowledge that is already there, but encrypted by some data interfaces of any legacy systems. So the long-term question will be what future engineering systems need to look like in order to offer the semantic knowledge produced in an engineering process in a transparent way, usable by a cognitive control unit (see Fig. 6).

Let us consider the example of an industrial handling task. Here we first need a functional description of the gripper and a geometrical description (may be more than that) of the work-pieces to be grasped. What actually needs to be described is driven by the process itself which means that one has to care about e.g. material, center of gravity, and the attributes of the surface but neither for the inner structural composition nor for the color.

The question is: How can CAD-models be semantically conditioned to generate any of the requested information? And which information has to be created from the scratch? In the context of the autonomous planning of the pick-and-place task we might think of integrating already known collision-free transfer moves whereas the fine movements for the gripper to grasp the object properly have to be planned cognitively by the system itself.

In order to generate such internal plans methods of **artificial intelligence** have to be looked at carefully. The research question is: starting from a more or less abstract description of a task or a goal, how can an adequate sequence of elementary operations be designed (semi-) autonomously (see Fig. 7)? The easiest way to think of might be an uninformed search within a known solution space. On top of that methods that use heuristic or statistical methods have proven to be a better choice for many problems. In many applications technical systems make use of rule-based methods to control a planning process. In addition to declarative knowledge about facts such methods also require some sort of production rules which then have to be part of the system's knowledge base.

The key will be to find a reasonable combination of representation formalisms and inference algorithms to represent and manipulate knowledge within manufacturing processes. In the

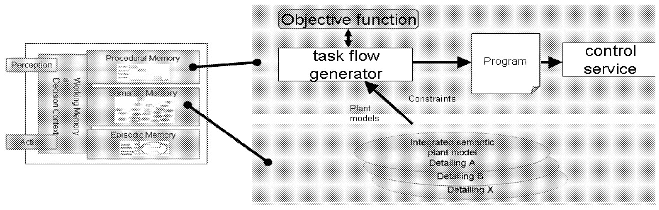

Fig. 7: Generating a task flow from semantic models

scope of our example the main question could be which strategy should be used to grasp a particular object and how the pick-operation can be transformed into a task flow that suits the constraints of the machine (and – of course – of the process).

But a cognitive control has to be more than just an expert system. The concept of self-optimization already implies that there is more to it than just finding the one and only solution to a given problem. Sometimes we have to deal with several possible solutions. At other times there isn't any. In other words: Objectives may be ambiguous or the solution space may be (at least partly) unknown. In such cases we might capitalize from probabilistic procedures or case based-reasoning methods. A major challenge is the question of how a system can learn. Learning means to enlarge or to prune the existing pool of knowledge by the use of instructive, deductive or – likely to be the most interesting – inductive techniques. But: to give a system any room for improvement, there is a need for feedback by the real world.

Fig. 8 shows that on any stage of manufacturing, **perception and action** play a key role in the cognitive concept. Perception means to aggregate, filter and integrate diverse and often fuzzy information from within or outside the system. At the lower levels this typically comes along with the use of sensors. The main challenge is to relate the information to a system state in a way that corresponds to the internal knowledge representation. This is the key for a system to learn the effects of its own behavior. In the scope of our example this could mean that a pick-operation which proves to be inefficient or even fails – by means of some act of machine learning – will lead to a modified action. For this case this may be another type of motor movement. On a higher level within the production system this could just as well be e.g. the switching of a manufacturing job between two resources.

Our example of the handling task demonstrates the reactive component of behavior based cognitive control. Partly sensory information is used to modify the knowledge base and to iden-

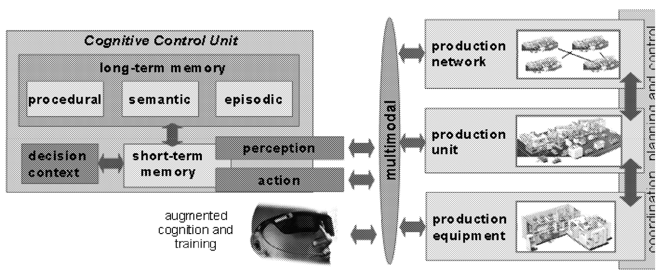

Fig. 8: Perception and action on different stages

tify the system's inner state. At the same time the gripper benefits from the sensor-information to perform direct sensor-actuator coupling for gripper control. This makes clear that autonomous planning is not the only challenge within self-optimizing production systems. An appropriate control architecture also has to take care of the coordination, control and surveillance of both – the explicit execution of designed plans and the subordinate continuous processes. For the pick-and-place example, the control is – below the planning level – in charge of the trajectory generation, the gripper control and the vision system for object recognition.

Another field that deserves our observation is the correlation between **human and technical systems**. To affirm that a cognitive control which is able to perform self-optimization will be a self-sustaining system would be some kind of misunderstanding. Technically speaking, this would hardly be possible. In most cases it wouldn't be desirable either. The human will always be – by virtue of his expert knowledge, his intuition and his extreme ability to learn – an essential part of the whole production system. He has to (by whatever means) express a task, communicate knowledge and experience, solve failure situations and even exhibit his fine motor skills in a "real" human-machine-cooperation. Therefore an important research question is – especially in the context of the design of cognitive control systems: How is the human mental model to be matched to the strictly rational model of a technical system. This goes far beyond the question of software ergonomics. How can a control that came to some sort of conclusion autonomously make plausible its decision to the human operator? By what channels and devices should the human try to generate and enrich knowledge in the technical system (for the pick-and-place task one might e.g. consider audio-visual or haptic systems).

The research question can be formulated as follows: What are the requirements to a model of information exchange in order to maximize the learning effect on both sides? Due to the view that tries to encompass all relevant levels of production technology this does not mean talking about a particular HMI but gives room to methodological consideration for human-machine cooperation models.

Last but not least we will have to talk extensively about appropriate **architectures for cognitive control**. First of all this addresses the control unit which has to implement all the rele-

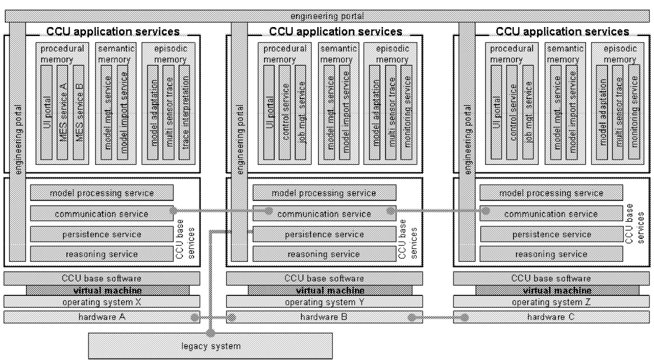

Fig. 10: Architecture of a cognitive control unit

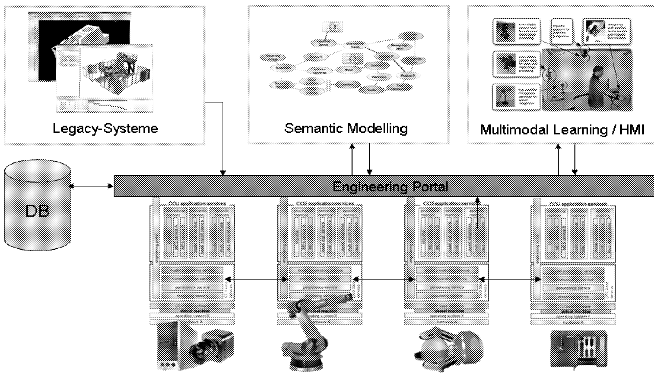

Fig. 9: Example of a cognitive robot cell

vant cognitive functional modules. Fig. 10 shows a possible design of such an architecture. A distinction is made between *base services* on the system level, which form the generic cognitive structure of the control unit, and *application services*, which contribute the domain specific knowledge. Several different *long term memory* structures are supplied as containers for this knowledge. The *short term memory* is used to allocate the currently relevant knowledge with regard to the environmental and system's state in order to reason on this knowledge. The *engineering portal* represents the interface to the human and to diverse legacy systems. Another important base service is responsible to establish communication between several units. This allows a decentralized cognitive structure in the sense of multi-agent systems.

The question of architecture goes beyond the control unit itself. In a broader sense it rises the question about an adequate infrastructure for automation technology in production systems. This is because the status quo shows that the lack of intelligence for decision is not the only obstacle on the road to self-optimizing systems. Transparency is more than just having a description of the static knowledge about the problem domain and the task. It is even more than collecting dynamic information about current states from somewhere in some data format. It also means that the essential information is available anywhere in the system and can be interpreted anywhere. This leads to more research questions. Some of them are:

What does hardware-independent software architectures look like that overcomes the numerous breakups between several software systems? Such breakups can be diagnosed not only with regard to traditional domains (like Logic Control, Numeric Control, Motion Control, Robot Control) but also along the life cycle (engineering, production, service) or between virtual and real process chains ("off line" and "on line").

How can a future programming paradigm be described which permits to formulate objectives in a comprehensive way? This would mean to keep a balance in order to neither be limited to the view of single subsystems nor to neglect essential characteristics of such subsystems.

How can a plug-and-participate functionality be achieved which allows self-organization to take place? This eventually leads to the concept of abolishing traditional software or hardware-interfaces (or at least make them invisible).

V. Conclusion

The necessity to resolve the dilemma between being planning-oriented and value-oriented manufacturing approaches leads to a request for production systems that are capable of self-organization and self-optimization. Whereas today's automation technology can basically cope only with fixed processes there is considerable potential in the use of cognitive concepts. This leads to a research approach that focuses on the transparency of knowledge, the capability of reasoning and planning as well as on a model of cooperation between human and the manufacturing system. This paper contains a first attempt of defining some methods to overcome traditional vertical and horizontal system break-ups. The resulting research questions are being discussed within the scope of the program *Cognitive Control Systems* at RWTH Aachen University.

VI. References

[1] Orilski, S; Schuh, G.: "Roadmapping for Competitiveness of High Wage Countries," in *Tagunsgsband "XVIII ISPIM Conference: Innovation for Growth – the Challenges for East and West. "*Warsaw, 2007.

[2] Frank, U. et al.: *Definition der Selbstoptimierung.* Arbeitspapier des Sonderforschungsbereichs 614: Selbstoptimierende Systeme des Maschinenbaus. Revision 1.105, 26. August 2004.

[3] Possel-Dölken, F.: *Projektierbares Multiagentensystem für die Ablaufsteuerung in der flexibel automatisierten Fertigung.* Berichte aus der Produktionstechnik 10/2006. Shaker Verlag. Aachen, 2006.

[4] Pauli, J.: *Learning-Based Robot Vision.* Principles and Applications. Springer-Verlag. Berlin, 2001.

[5] Hildebrand, B.; Knoll, A.; Scheering, C.; Zhang, J.: *Kognitive Aspekte bei der Realisierung eines robusten Robotersystems für Konstruktionsaufgaben.* Forschungsberichte der Technischen Fakultät, Department of Information Technology. Report 98-01, University of Bielefeld.

[6] Newell, A.: *Unified Theories of Cognition.* 3. Aufl. Cambridge: Harvard University Press, 1994.

[7] Soar. Available: http://sitemaker.umich.edu/soar/home

[8] ACT-R. Available: http://act-r.psy.cmu.edu/

[9] Brooks, R.A.: "Intelligence without Representation," in *Artificial Intelligence*, vol. 47, 1991, pp. 139-159.

[10] Bischoff, R. ; Graefe, V.: "Design Principles for Dependable Robotic Assistants," in *International Journal of Humanoid Robotics*, vol. 1, 2004, no.1, pp. 95-125.

[11] Albus, J.S; Anthony J.B.: RCS: "A cognitive architecture for intelligent multi-agent systems," in *Annual Reviews in Control*, vol.29 , 2005, pp. 87-99.

[12] Burghart, C. et al.: "A Cognitive Architecture for a Humanoid Robot: A First Approach," in *IEEE-RAS Int. Conference on Humanoid Robots*, Humanoid, 2005.

[13] Lescher, M.: *Eine Methode zur automatisierten Generierung von Arbeitsabläufen für den Service an Produktionssystemen*. Doctoral Thesis, RWTH Aachen Universty, submitted for publication.