Towards Agent-Based Multi-Site Scheduling

Jürgen Sauer[1], Tammo Freese[2], Thorsten Teschke[2]

Abstract. Scheduling problems are usually treated within single plant environments or within companies with several production locations. Due to the globalization of markets companies can no longer be regarded isolated from each other. They are in fact elements of spatially distributed production and logistics networks, where apart from the actual production process transport and stock keeping gain importance. This contribution analyses the organizational structures found in distributed production networks and proposes an approach for their mapping to multiagent systems for integrated production planning and scheduling. Moreover, a platform for multiagent systems deployment is outlined, which is apt to satisfy the major requirements to an agent platform for dynamic, distributed production and logistics networks.

1 INTRODUCTION

In the past companies have often been regarded as self-contained units with well-defined business relationships to consumers and suppliers. Measures of optimization were confined by a company's boundaries. Today, competencies for fast and economical development and manufacturing of complex products are distributed to different companies. This trend as well as the markets' evolution to buyer's markets and shortened product life cycles necessitate optimizations beyond a company's boundaries. Concepts like *supply management*, *supply chain management* and eventually *virtual enterprises* have been devised in order to open up new potentials of optimization by regarding a company's interweavement with its suppliers and consumers ([1; 2]).

2 MOTIVATION AND PROJECT OUTLINE

The globalization of markets leads to the formation of spatially distributed production and logistics networks. In addition to the actual production process transport and stock keeping are of increased importance, since they strongly affect a company's ability to meet delivery deadlines. For this reason centralized approaches to production planning and scheduling for companies with a single production site cannot be transferred to distributed enterprises directly. Moreover, schedule stability decreases in centralized approaches, since even locally resolvable perturbances like machine breakdowns change the central schedule. *Multi-site scheduling* ([3]), instead, represents a promising approach where planning and scheduling encompasses two levels of hierarchy. On the upper, global level a rough schedule for the entire enterprise is generated based on imprecise, cumulated information on available resource capacities. This rough schedule defines targets for the enterprise's individual sites, which refine it into site-specific schedules. The distribution of the planning and scheduling process effects an increased schedule stability, since locally resolvable deviations do not have to be regarded within global planning and scheduling.

The AMPA project (Agent-Based Multi-Site Planning and Scheduling Application Framework) is concerned with distributed planning and scheduling in dynamically changing logistics networks ([4]). Starting from the notion of multi-site scheduling, the dynamics of business relationships and the individuality of companies within a logistics network has to be considered by pursuing a multiagent approach. According to [5] multiagent systems represent a suitable abstraction for modelling scheduling problems. In order achieve the goals stated above, the multi-site scheduling approach will be enhanced in many respects. An integrated consideration of production and transport planning and scheduling is striven for. Moreover, the restriction to two levels of planning and scheduling hierarchy abolished, yielding a more detailed decomposition of the planning and scheduling problem. Finally, the strict hierarchical decomposition of planning and scheduling problems regarded in multi-site scheduling is complemented by a network dimension which is especially suitable for representing inter-company relationships. This imposes negotiation tasks in vertical (along the hierarchy) as well as in horizontal (between departments of different firms) direction.

Agents to be employed in the production, transport and stock keeping domain differ regarding the knowledge as well the heuristics and strategies used for performing their planning, scheduling and coordination tasks. The development of software agents and a respective agent platform within the AMPA project will therefore pursue a component-based approach ([6]). Specific types of agents can then be created by exchanging and configuring software components (cf. [7] and [8]).

In order to realize such an approach three major steps have to be performed. First, an adequate model of the organizational structures with respect to multi agents has to be found. Second, the planning and scheduling requirements and capabilities of the agents on the different levels of the organizational model have to be defined. And third, a communication model between the agents along the hierarchy as well as between agents of different company substructures has to be developed. The following chapters focus on the first step and describe the basic notions of the problem area as well as the organizational model for the design of an agent-based multi-site scheduling system.

[1,2,3] Fachbereich Informatitk, Universität Oldenburg, Escherweg 2, D26121 Oldenburg, Germany, email: sauer@informatik.uni-oldenburg.de, http://www-is.informatik.uni-oldenburg.de/~sauer

3 BASICS

Within the scope of the preceding outline of the AMPA project's contents the major subjects "supply chain management", "virtual enterprises" and "software agents" and "multi-site scheduling" have emerged. These shall be illustrated more detailed in the subsequent sections.

3.1 Multi-Site Scheduling

Scheduling problems are usually treated in a single plant environment where a set of orders for products has to be scheduled on a set of machines [9-12]. However, within many industrial enterprises the production processes are distributed over several manufacturing sites, which are responsible for the production of various parts of a set of final products. Usually, there is no immediate feedback from the local plants to the logistics department and communication between the local schedulers takes place without any computer-based support.

Due to the distribution of production processes to different plants some specific problems arise in addition to the problems of the dynamic complex scheduling environment:

- Interdependencies between production processes that are performed in different plants have to be regarded.
- In global scheduling generalized and inprecise data are used instead of precise data.
- Existing (local) scheduling systems for individual plants that accomplish the local realization of global requirements should be integrated.
- The coordination of decentralized scheduling activities for all plants within one enterprise is necessary since several levels of scheduling with their specific scheduling systems have to work cooperatively in a dynamic distributed manufacturing environment.
- The uncertainty about the actual "situation" in individual plants has to be regarded.
- Different goals have to be regarded on the different levels.

The multi-site scheduling approach [3] presents a hierarchical two-level structure reflecting the organizational structure often found in business.

On the upper global level requirements are generated for intermediate products manufactured in individual locations. Local scheduling (at individual locations) deals with the transformation into concrete production schedules which represent the assignment of operations to machines. On both levels predictive, reactive as well as interactive problems are addressed, not only to generate schedules but also to adapt them to the actual situation in the production process. Additionally, communication between the systems is needed to support the consistent exchange of data and to coordinate the local scheduling systems.

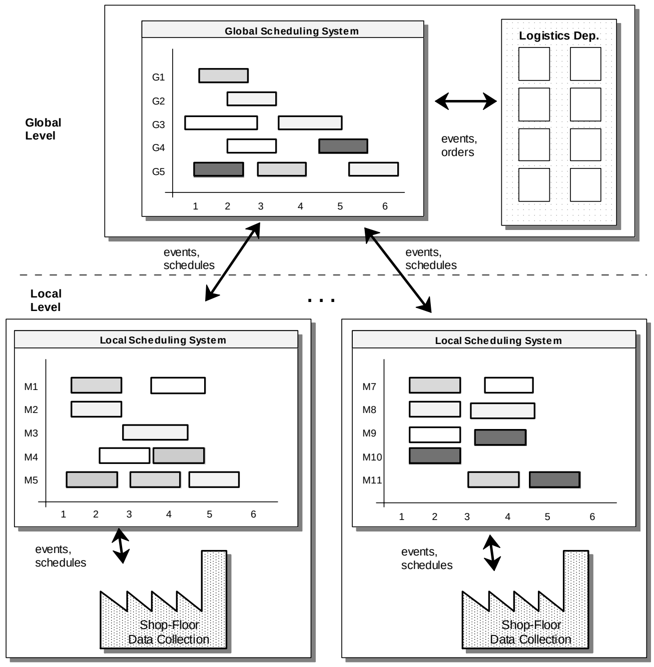

Figure 1. Multi-Site Scheduling

The multi-site scheduling tasks can be characterized as follows:

- *Global predictive scheduling*: A global-level schedule with an initial distribution of internal orders to local production sites is generated.
- *Global reactive scheduling:* If problems cannot be solved on the local level or the modified local schedule influences other local schedules (inter-plant dependencies), global reactive scheduling can then cause a redistribution of internal orders to local plants and adapt the global schedule.
- *Local predictive scheduling:* Based on the global schedule, the local plants draw up their detailed local production schedules.
- *Local reactive scheduling:* In case of local disturbances, the local reactive scheduler first tries to remedy them locally by interactive repair.
- *Communication and coordination:* Both levels have to be provided with data as actual and consistent as possible. Therefore information has to be sent between the lecels, e.g. the global schedule consisting of information on internal orders, affiliated intermediate products, machine groups to use, time windows that should (possibly) be met, and required quantities of intermediate products, unexpected events that effect the local resp. the global level (e.g. the cancellation of an order or breakdowns of machine groups).

For the solution of the predictive and reactive scheduling tasks several problem solving approaches are useful. Some of them have been checked for the MUST (Multi-Site Scheduling System) approach [3]. Table 1 shows the tasks and some of the appropriate methods from which several are investigated in the MUST project.

Table 1: Multi-Site Scheduling Tasks and Methods

Problem Area	Techniques
Global Predictive Scheduling	Heuristics, Constraints, Genetic Algorithms, Fuzzy-Logic
Global Reactive Scheduling	Interaction, Heuristics, Constraints
Local Predictive Scheduling	Constraints, Heuristics, Genetic Algorithms, Neural Networks, OR-Systems
Local Reactive Scheduling	Interaction, Heuristics, Constraints, Multi-Agents
Communication	Blackboard, Contract Net

The approach is implemented in the distributed knowledge-based scheduling system MUST. The system architecture reflects the two level approach and consists of one global scheduling subsystem and several local subsystems, one for each individual production site. All systems include the knowledge-based techniques described for the predictive and reactive scheduling tasks to be performed. Communication is realized using the blackboard paradigm. The MUST subsystems are implemented as decision support systems thus lacking one of the major characteristics of multi-agent systems, which is the proactivity (see 3.4).

3.2 Supply Chain Management

Decreasing transaction costs, advanced control of processes and thinking in profit centers increasingly lead to companies outsourcing those parts of the creation of value without core competencies. The growing force to shorten delivery periods and product innovation cycles while at the same time increasing the rates of return induced by globalized markets requires an intensified cooperation of all companies along the inter-company supply chain ([1]).

The concept of a supply chain is insofar misleading as the companies involved in the development and manufacturing as well as transport, distribution and selling of a product usually do not constitute a chain, but rather a network. Supply chain management coordinates the activities within this logistics network under the overall goal of inter-company and intersite optimization of a product's development and manufacturing process as well as the innovation of processes.

Characteristic for supply chain management is the strategic, long-term cooperation of companies as well as the small number of suppliers for a particular product. Cooperations according to the supply chain management approach rely on massive exchange of information, which presupposes trust_between the partners within the supply chain and the long-term abolishment of information barriers between the individual companies.

Among the risks of supply chain management is the development of unilateral dependencies and the potential abuse of information on co-producers. Additionally, due to its long-term orientation the concept of supply chain management is not suitable for short-term cooperations ([2]).

3.3 Virtual Enterprises

A virtual enterprise represents a network of companies affiliated in order to perform a particular, temporally limited task, thereby appearing as an entity. Opposed to the concept of supply chain management, which regards strategic relationships of rather long-term nature, virtual enterprises excel especially by their ability to flexibly reorganize themselves. A virtual enterprise can be viewed as a temporarily existing supply chain.

Virtual enterprises are established in order to be able to flexibly react to the opportunities of highly dynamic markets. The selection of partners is based on cost effectiveness and uniqueness of their products instead of more traditional factors like organizational size, geographic location, IT infrastructure, employed technologies and implemented processes. The companies engaged in a virtual enterprise share their knowledge, their competencies and business relationships in order to perform the virtual enterprise's task. This combination of forces is to enable the companies to reach global markets with products and solutions that each of them could not have accomplished on its own ([2]).

3.4 Software Agents

Software agents represent a software development paradigm which is appropriate for distributed problem solving. They are employed in numerous systems of distributed artificial intelligence (DAI). In common linguistic usage, an agent is everyone who acts on behalf of another. In computer science the concept of a software agent is not uniformly defined. In [13] Franklin and Graesser present a comparison of numerous definitions.

Wooldridge defines an agent as "a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives" ([14]). Therefore, a fundamental property of an agent is *autonomy*: an agent operates without direct interference by humans or other systems, and has control over its behaviour and its internal state. The concept of an intelligent agent extends this definition by the capability of acting flexibly, whereby the notion of flexibility comprises three characteristics:

- *reactivity*: agents perceive their environment and react timely and appropriately to changes within this environment;
- *pro-activeness*: agents do not only react to observed changes within their environment, but are capable of taking the initiative in a goal-directed fashion;
- *social ability*: agents interact with other agents (and possibly humans) by exchanging information formulated in a mutually agreed communication language. Moreover, the notion of social abilities comprises complex patterns of behaviour based on communication protocols, e.g. for the purpose of negotiation.

This concept of intelligent agents is perfectly suitable for the domain of production planning and scheduling. First, an agent has some kind of knowledge of the problem to be solved (the scheduling problem) and its environment (e.g., other agents or the shop-floor), and is capable of negotiation. Second, it is able to quickly react to changes within its environment, e.g. a machine breakdown. And third, agents are pro-active, allowing them, e.g., to improve their schedules while no other service request are issued [15]. Therefore, this definition is adopted for the agents to be developed within AMPA: Every agent shall be able to schedule its activities (autonomy), to change its schedule in case of disturbances (reactivity), and to optimize its schedule (pro-activeness). Messages concerning changes, disturbances etc. are exchanged using a communication language commonly agreed upon (social abilities).

Additional features of agents that are studied in different approaches of DAI are ([16]):

- *mobility*: mobile agents are able to move within electronic networks;
- *veracity*: a truthful agent does not knowingly provide other agents with false information, e.g., on its environment or its internal state;
- *benevolence*: benevolent agents do not have conflicting goals, and they try to achieve what they are asked to;
- *rationality*: rational agents try to achieve their goals. They do not knowingly act in a way conflicting with their goals.

The central problem of multiagent systems is how to achieve coordinated action among agents in a way yielding problem solving capabilities that exceed those of any individual agent.

4 MAPPING ORGANIZATIONAL STRUCTURES TO SOFTWARE AGENTS

The basic elements of business organizations are *posts* and *relationships* between them. Important dimensions of business organization systems are specialization, coordination and the directional system. The aspect of specialization is concerned with the division of work, which is about different organizational units performing partial tasks of different kinds. The division of work attained by specialization requires the coordination of its entailed activities. This task can be simplified by means of hierarchies. The directional system specifies authorities to instruct, responsibilities, and powers of decision which a superordinated post has regarding to a subordinated post. Concerning the structure of posts within the directional system two typical basic forms can be distinguished. The single-line system rests on the principle of unity of command and organizes posts in a tree structure. The multiple-line system aims at realizing the principle of shortest paths in interdepartmental coordination problems and organizes posts in a graph structure. In the latter form, a post can be subordinated to several posts within the hierarchy. This, however, may entail questions of authority and the risk of unclear responsibilities ([17]).

4.1 Organization Model

The organization model proposed by AMPA enhances the multi-site scheduling approach delineated in section 3.1 in two respects: first, the hierarchy considered by multi-site scheduling is extended by additional levels, and second, the hierarchical, intra-company perspective is complemented by a network-like, inter-company dimension.

Hierarchical structures are a common representation for intra-company directional systems for they are suitable for defining powers of decision, authorities to instruct, duties of supervision and tasks of inspection. This approach is also pursued within the scope of AMPA. Therefore, posts are defined according to resource-oriented aspects and arranged in a hierarchy. Potential posts are, e.g., an entire company, production sites, job shops, warehouses, transport vehicles, resource groups, or machines. A post is represented by a planning agent.

Complex organization structures, however, are not exclusively organised hierarchically. This applies especially to legally and economically independent enterprises, for relationships between them do neither define powers of decision nor authorities to instruct; they only represent the aspect of coordination between their directional systems. In addition to the hierarchical, static dimension of the intra-company directional system in AMPA the network-like, dynamic dimension of the coordinating, logistical relationships on all levels of hierarchy is considered by a special type of relationship. Within AMPA organizations are accordingly represented by an overlay of hierarchical and network-like structures, thus achieving both vertical and horizontal integration. The resulting organization model is illustrated by Figure 2, where the enterprise under consideration is represented by dark nodes. These nodes also represent the problem area of multi-site scheduling. External organizational units are depicted using pale nodes.

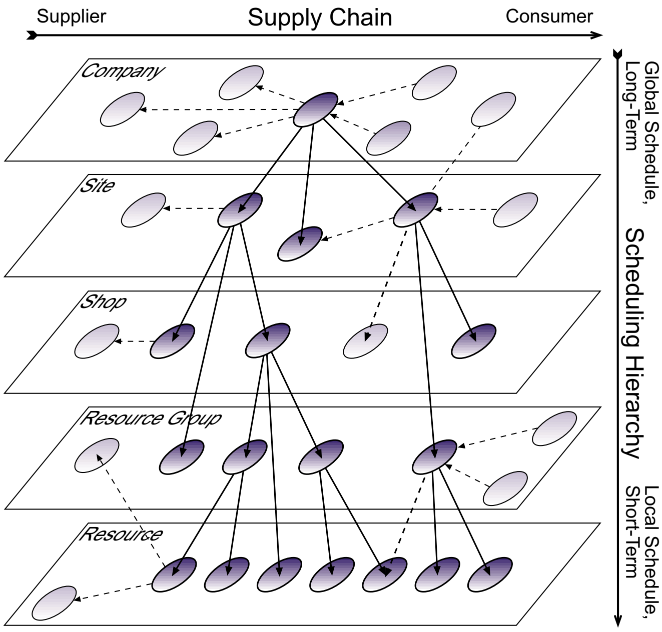

Figure 2. Organization Model

On a more formal level the approach pursued within AMPA can be regarded as a combination of the single-line system and the multiple-line system. This combination aims at clear responsibilities on the one hand and improved ways of coordination by shorter communication paths on the other. In order to achieve these goals while avoiding the disadvantages pointed out before two different types of relationships are considered. Both are directed and define a potential usage of a post by another, whereby subsequently the former post will be referred to as *supplier* and the latter as *consumer*. Usage as indicated by such a relationship is thus regarded as the supplier providing services for the consumer.

Disciplinary subordination relationships are part of the directional system and serve the representation of the organization model's hierarchical intra-company dimension. In addition to potential usage of a post by another they specify the sole responsibility a superordinated post (the consumer) has for its disciplinarily subordinated posts (the suppliers). In order to achieve the therefore required unambiguousness a post may maximally be subordinated to one other post, i.e. the structure of the disciplinary subordination relationships correspond to the single-line system. The responsibility of a superordinated post for its disciplinarily subordinated posts is expressed by the latter posts' opportunity to report order requests to the former post, if these requests cannot be accomplished by a service provided by a post subordinated to the latter. Moreover, a disciplinary subordination relationship requires a subordinated post to grant access to information on its state (e.g., its workload) to its superordinated post. This information may affect the superordinated post's decision making process. In Figure 2 disciplinary relationships are depicted by solid arrows.

Functional relationships represent the network-like dimension of the organization model and serve the coordination between organizational units which are not connected by the directional system. A functional relationship solely defines a usage relationship between two posts, i.e. the supplier may neither forward requirements to the consumer, nor does the consumer have any kind of responsibilities for the supplier. Moreover, the supplier is not bound to give away information on its state to the consumer. In addition to one disciplinary subordination relationships a post may engage in arbitrary functional relationships. Hence, the system of functional relationships corresponds to the multiple-line system. In Figure 2 functional relationships are represented by broken arrows.

4.2 Mapping to Software Agents

In the AMPA project an enterprise is represented by a system of agents. This allows a direct mapping of the enterprises' structures and their communication as well as the integration of existing scheduling applications. The following steps describe how an enterprise is mapped onto a multiagent system.

1) *Identify agents:* Every entity in an enterprise for which AMPA should do the scheduling is represented by an agent. However, if a scheduling system for one or more of the entities exists, it need not be replaced, but it is represented by a single agent which acts as a wrapper. Therefore existing scheduling systems like the subsystems of MUST may be integrated easily which leads to an open scheduling environment.
2) *Define scheduling tasks*: Depending on the position in the network different planning and scheduling tasks have to be performed by the agents. These tasks have to be identified and the appropriate scheduling knowledge, e.g. one of the algorithmic solutions presented in table 1, has to be added to the agent.
3) *Add disciplinary relations:* Between the Units which are represented by the agents identified in step 1, there are disciplinary relations. These organize the agents in a tree structure. If the network of disciplinary relationships between the agents has no tree structure, exactly one of the relationships must be chosen for having a unity of command.
4) *Add functional relationships:* Aside disciplinary relations, there are also functional relations between the units in an enterprise. These should be adopted in the agent system.

The first three steps yield an agent structure that represents the internal structures of an enterprise. The following step is to embed this enterprise into its environment.

5) *Integrate suppliers and consumers:* To achieve an integration of an enterprise into its logistics network, the suppliers and consumers of the enterprise have to be represented in the agent system. If they also use AMPA agents, the correct agents must be identified. Otherwise, wrapper agents encapsulate the communication with these business associates. If a large group of similar suppliers or consumers has to be integrated, it can alternatively be represented by a single agent which wraps the communication with all the members of the group. The agents are integrated in the agent structure by adding functional relationships to the agents for which they are suppliers or consumers. For both communication possibilities an appropriate communication model on the basis of the contract net protocol has to be defined.

4.3 A platform supporting the agents

For deployment of the agent system, an agent platform is needed that acts as an environment for the agents. It aims in supporting the flexible and easy configuration of new agents to be incorporated into the whole system. Based on a Java virtual machine the platform supports amongst others:

- *Distribution (agent context):* The agents of an AMPA are normally distributed in two different kinds. A group of agents which runs on the same server is *locally distributed*. In a *global distribution,* the agents run on different servers. As an example, the agents of an enterprise run on one server, whereas the enterprises which build a virtual enterprise or a supply chain will normally run servers on their own.
- *Communication (communication layer):* To support different kinds of distribution, there must be different kinds of communication between local and global distributed agents. An agent platform should have a communication interface that encapsulates that hides this difference from the agents. So an agent has no information whether its communication partners reside on the same or on a different server.
- *Platform independence:* For deploying an agent system in an heterogenous environment of computer systems, it is either required to develop specific agents for every platform, or to use a platform that offers an equal interface on all systems. The first possibility is not suitable especially in virtual Enterprises, where a huge number of different systems is to expect. Therefore it is more convenient to develop an agent platform in Java which would be independent of the underlying hardware structure.
- *Security (security policy):* An agent represents a real system and acts at least in parts autonomously. In this context, security must have a high priority. An agent platform has to implement suitable security mechanisms that protect the agents from unauthorized access. In contrast, authorized users must have full control over their agents. To achieve a combination of these requirements, the platform must have an authorization concept that offers roles with different access grants.
- *Persistency:* An agents needs access to persistent information like the production schedule or the product ontology. To achieve this in combination with platform independence, the agent platform has to decouple the application layer (the agent) and the persistency layer.

- *User interfaces:* In most cases, the possibility to observe and agent and to intervene in its actions is very important for the acceptance of an agent system. Therefore an agent system should support connections of agents and user interfaces.
- *Transactions:* Complex scheduling processes involve many subsequent negotiations with agents along the supply chain. Sometimes they can only be executed partly what leads to an inconsistent schedule. The agent platform should provide a transaction concept which allows the rollback of failed transactions.
- *Configuration (configuration information, resources):* The persistency and security mechanisms require a possibility to replace underlying database management systems and ERP systems. Moreover, configuring the deployed agents is necessary.

Figure 3 shows an architecture outline that targets to fulfil the requirements above.

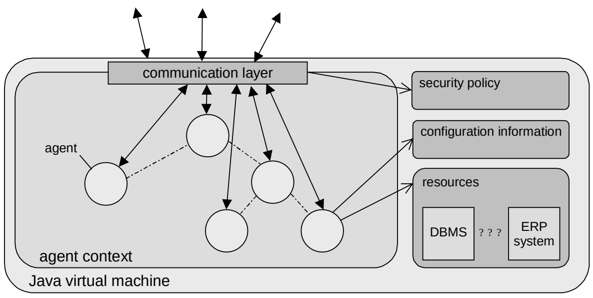

Figure 3. Architecture of agent platform

The component structure of the platform and the agents used within the platform allow for the easy configuration of different systems. On the basis of a generic agent it is possible to generate specialized agents for production, transport or stock. This is realized by the exchange and configuration of components of the agents [7].

4.4 Related work

A comparable approach is pursued by Swaminathan et al. in [7], who represent the structural elements of a supply chain like, e.g., production and transport units by agents. In contrast to the organization model proposed in this section hierarchical relationships are not considered, i.e. a supply chain is modelled as a flat network.

Other agent-based approaches for enterprise modeling often focus on the definition of ontologies supporting the description of the workflows within the enterprises [18]. Within the "Enterprise"-project agents are used to represent tools that perform activities. These agents are integrated in a system for workflow management. The TOVE project [19] uses agents to represent the "classical" functions of production planning and -control. In addition these agents are connected via information agents providing the necessary information for the "functional" agents involved.

5 CONCLUSION AND FURTHER WORK

On the basis of the multi-site scheduling approach an extension in several directions is proposed. Not only one enterprise with distributed production has to be considered but also the suppliers of the several units shall be integrated in the scheduling task. The systems supporting the scheduling tasks are organized as agent-based systems thus offering all the advantages of multi-agent systems. The presented approach for an agent-based system performing scheduling in production networks is still under development. First steps have been the modeling of the organizational structure for a multi-agent system. The next steps are the prototypical implementation of the agent platform and a number of agents using a common framework like Enterprise Java Beans [20] or tools for the development of multi-agent systems like ZEUS [21]. Within this prototype the position depending scheduling knowledge and the extended negotiation protocols will be integrated and tested.

6 REFERENCES

[1] Scheer, A.-W., and Borowsky, R. 'Supply Chain Management: Die Antwort auf neue Logistikanforderungen'. *LM'99 - Intelligente I+K Technologien*, Bremen. Springer Verlag. 1999, pp. 3-14.

[2] Schönsleben, P. *Integrales Logistikmanagement: Planung und Steuerung von umfassenden Geschäftsprozessen*. Springer, Berlin, Heidelberg. 1998.

[3] Sauer, J. 'A Multi-Site Scheduling System'. *AAAI's Special Interest Group in Manufacturing Workshop on Artificial Intelligence and Manufacturing: State of the Art and State of Practice*. 1998.

[4] Appelrath, H.-J., Freese, T., Sauer, J., and Teschke, T. 'Konzept eines Multiagentensystems für die verteilte Ablaufplanung'. *Proceedings des Workshops "Agententechnologie" auf der KI '99*, Bonn. 1999, pp. S. 37-45.

[5] Henseler, H. *Aktive Ablaufplanung mit Multi-Agenten*. Dissertation, Carl von Ossietzky Universität Oldenburg, Oldenburg. 1998.

[6] Szyperski, C. *Component Software - Beyond Object-Oriented Programming*. Addison-Wesley Verlag. 1998.

[7] Swaminathan, J. M., Smith, S. F., and Sadeh, N. M. 'Modeling Supply Chain Dynamics: A Multiagent Approach'. *Decision Sciences Journal*, 29(3). 1998, pp. 607-632.

[8] Sauer, J., Appelrath, H.-J., Bruns, R., and Henseler, H. 'Design-Unterstützung für Ablaufplanungssysteme'. *KI-Künstliche Intelligenz*, 2/97. 1997, pp. 37-42.

[9] Dorn, J., and Froeschl, K. A. *Scheduling of Production Processes*. Ellis Horwood. 1993.

[10] Sauer, J., and Bruns, R. 'Knowledge-Based Scheduling Systems in Industry and Medicine'. *IEEE-Expert*(February). 1997.

[11] Smith, S. F. 'Knowledge-based production management: approaches, results and prospects'. *Production Planning & Control*, 3(4). 1992.

[12] Zweben, M., and Fox, M. S. *Intelligent Scheduling*. Morgan Kaufman. 1994.

[13] Franklin, S., and Graesser, A. 'Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents'. *Third International Workshop on Agent Theories, Architectures, and Languages*. Springer-Verlag. 1996.

[14] Wooldridge, M. 'Intelligent Agents'. *In:* G. Weiss, ed. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence.* The MIT Press. 1999, pp. 27-77.

[15] Henseler, H. 'From Reactive to Active Scheduling by using Multi-Agents'. *In:* R. M. Kerr and E. Szelke, eds., *IFIP TC5/WG.5.7 Second International Workshop on Knowledge Based Reactive Scheduling (KBRS) PREPRINTS.*, Budapest. 1994.

[16] Wooldridge, M., and Jennings, N. R. 'Intelligent Agents: Theory and Practice'. *Knowledge Engineering Review*, 10(2). 1995.

[17] Schierenbeck, H. *Grundzüge der Betriebswirtschaftslehre*. Oldenbourg Verlag GmbH, München. 1993.

[18] Fox, M. S., and Gruninger, M. 'Enterprise Modeling'. *AI Magazine*, 19(3 (Fall 98)). 1998, pp. 109-121.

[19] Barbuceanu, M., and Fox, M. S. 'The Information Agent: An Infrastructure Agent Supporting Collaborative Enterprise Architectures'. *3rd Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Morgantown, WV. 1994.

[20] Matena, V., and Hapner, M. *Enterprise JavaBeans Specification*., Sun Microsystems. 1999.

[21] http://www.labs.bt.com/projects/agents/zeus/