A Multi-Site Scheduling System

Jürgen Sauer
University of Oldenburg, Dept. of Computer Science
Escherweg 2, D-26121 Oldenburg
Germany
sauer@informatik.uni-oldenburg.de
http://www-is.informatik.uni-oldenburg.de/~sauer

Abstract

The objective of multi-site scheduling is to support the scheduling activities of a global scheduler and schedulers in distributed production plants in a cooperative way. A global schedule generated on a global level must be translated into detailed schedules as part of the local scheduling process. In case of disturbance, feedback between the local and global levels is essential. Previous methods focused on local production sites, in most cases without coordination. In this work we present an approach that considers the adequate modeling and processing of scheduling on the global and local level together with the coordination of the scheduling activities on the scheduling levels.

The Multi-Site Scheduling Problem

Scheduling problems can be found in several different application areas, e.g. the scheduling of production operations in manufacturing industry, computer processes in operating systems, truck movements in transportation, aircraft crews, etc. The main task of scheduling is the temporal assignment of activities to resources where a number of goals and constraints have to be regarded. Scheduling covers the creation of a schedule of the activities over a longer period (predictive scheduling) and the adaptation of an existing schedule due to actual events in the scheduling environment (reactive scheduling) (Smith 1992, Kerr and Szelke 1995). But scheduling has also a very important interactive dimension because we always find humans within the scheduling process who have to decide, interact or control. Several decisions have to be taken by the human scheduler (the user of the scheduling system), e.g. introducing new orders, cancel orders, change priorities, set operations on specific schedule positions, and these decisions have to be regarded within the scheduling process (Hsu et al. 1993).

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 (Dorn and Froeschl 1993, Sauer and Bruns 1997, Smith 1992, Zweben and Fox 1994). 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.

Figure 1 illustrates the hierarchical two-level structure of multi-site scheduling reflecting the organizational structure often found in business. On the 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. This scenario can easily be adapted to other application areas e.g. distributed software development or other projects.

The complexity of real-world scheduling as well as multi-site scheduling scenarios is mainly determined by

- the requirements imposed by numerous details of the particular application domain, e.g. alternative machines, cleaning times, set-up costs, etc.,
- the dynamic and uncertain nature of the manufacturing environment, e.g. unpredictable set-up times, machine breakdowns, etc.,
- conflicting organizational goals, e.g. minimize work-in-process time, maximize resource utilization, and
- the need of interaction with a human scheduler.

Due to the distribution of production processes to different plants some specific problems arise:

- Complex interdependencies between production processes that are performed in different plants, have to be regarded, e.g.
  - temporal relations between intermediate and final products, e.g. if an intermediate product that is manufactured in plant A is needed in plant B,
  - the same item can be manufactured in different plants (possibly at different costs),
  - the transport of parts between different plants needs transportation capacities and is time- and cost- intensive.

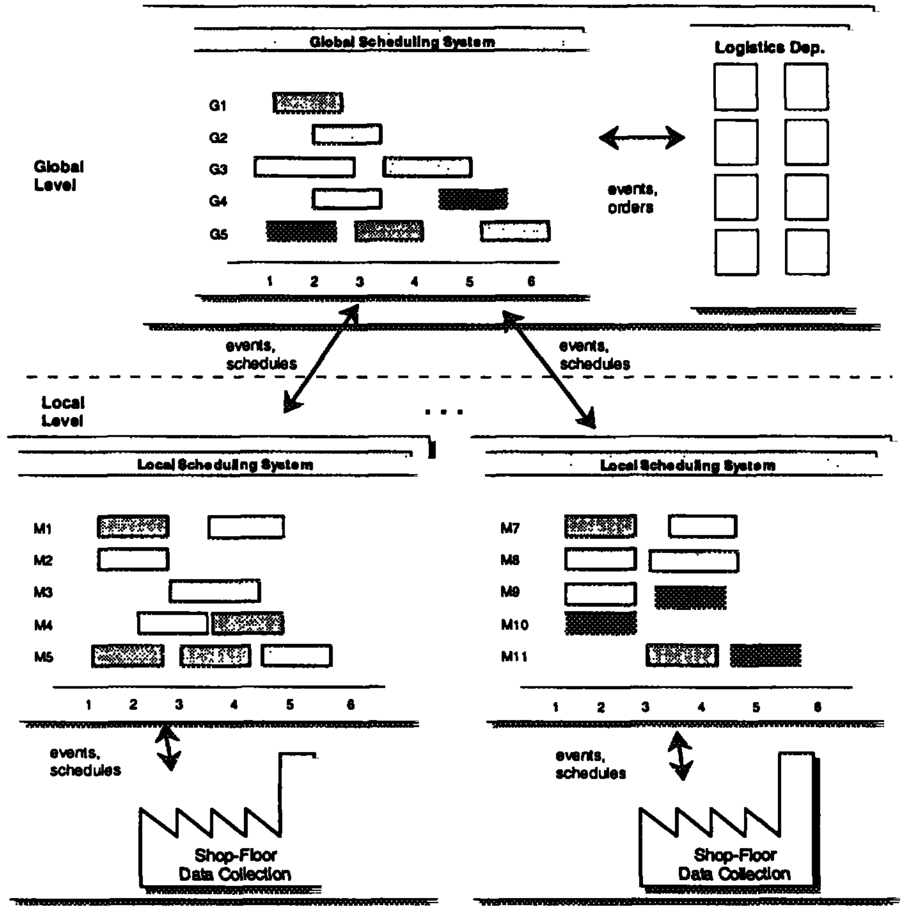

Figure 1: Multi-Site Scheduling

- In global scheduling generalized data are used instead of precise data, these are e.g.
  - capacity information referring to machine groups instead of single machines,
  - information on the duration of manufacturing processes for intermediate or final products that are often estimated values.
- 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 goals of the global scheduling activities such as meeting due dates of final products, minimizing transportation costs and work-in-process times. Furthermore the solution of the global scheduling problem should be as robust as possible, i.e. it should give enough flexibility for a local scheduler to react to local disturbances without affecting the other sites. This can be achieved amongst others by using buffer times in the time windows for local production and by trying to optimize the load balancing on the machine groups.

  The goals of the local scheduling level such as optimizing machine utilization, set-up times and meeting due dates of intermediates, which are often in contrast to each other.

  Additional goals, especially for the effectiveness of a multi-site scheduling system, are the early detection of capacity problems and in case of reactive scheduling, one of the main goals is to preserve as much as possible of the existing global schedule in order to minimize the subsequent effort on the local level.

Therefore, the multi-site scheduling problem can be divided into global and local scheduling tasks together with communication tasks. On the global level products must be distributed to plants where the intermediates have to be produced. On the local level the intermediates have to be scheduled within the local production sites. On both levels predictive and reactive scheduling is necessary to create respectively maintain the global or local schedules. Additionally, the coordination between these tasks has to be supported in order to provide all components with actual and consistent information.

As typical tasks of a multi-site scheduling can be identified:

- *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 shall be provided with data as actual and consistent as possible. At least the following items of information have to be sent from the global to the local level:
  - 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 level (e.g. the cancellation of an order).

  From the local to the global level these are amongst others:
  - the local realization of the global requirements with information on internal orders, affiliated intermediate products, start and end time of all locally scheduled activities and used machine groups,
  - appearance of failure events,
  - suggestions for possible local rescheduling.

The goals and events for the reactive tasks are quite similar on both levels. Examples of goals are

- conserve as much as possible of the former schedule,
- react almost immediately.

Examples of events are breakdowns of resources and new or cancelled orders.

A Multi-Site Scheduling System

Only a few approaches have been presented for multi-site scheduling, e.g. (Bel and Thierry 1993, Liu and Sycara 1993, Wauschkuhn 1992). Most of them try to solve the multi-site scheduling problem by generating a better initial distribution of orders to the different production sites, i.e. they are restricted to global predictive scheduling. In classical production planning and control systems there is only a simple distribution when machine groups are used instead of machines in the materials requirement and capacity planning. But here the actual situation at the local plants is not taken into account and feedback is not integrated.

The distributed knowledge-based scheduling system MUST (Multi-Site Scheduling System) has been designed to support the human experts in the management of the dynamic distributed manufacturing environment, in particular in the scheduling of the appropriate distribution of the orders to the different manufacturing plants as well as in the coordination of the decentralized scheduling activities for all plants within one enterprise. The MUST approach consists of a global scheduling level (logistics level) and a local scheduling level (single plant level) with predictive, reactive and interactive scheduling components on both levels. The objective of this approach is the reduction of complexity of distributed scheduling and improving the quality of the solution at the same time. Figure 2 shows the architecture of the MUST-system which reflects the tasks mentioned above.

On the global level the initial distribution of the orders for intermediates and final products to the individual plants is performed (global predictive scheduling). If any interplant constraints are affected by alterations of the local production schedules, the reactive part of the global scheduling (global reactive scheduling) reestablishes consistency, e.g. by means of due date relaxation, redistribution of orders, splitting of orders, etc. The local predictive scheduling component is used to generate detailed local production schedules for each plant. Possible disturbances caused by unexpected events, e.g. a machine breakdown, are handled by the local reactive scheduling component. If other plants are affected as well by local decisions the responsibility for solving the problem is passed again to the global reactive scheduling.

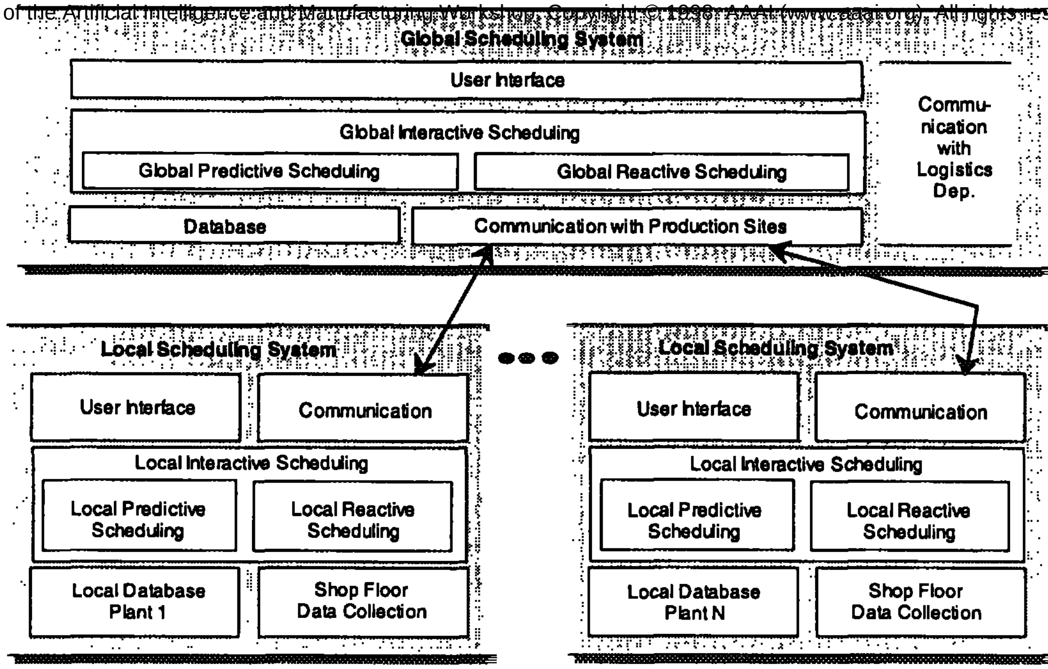

Figure 2: The MUST System Architecture

The MUST system consists of one global scheduling subsystem and several local subsystems, one for each individual production site. Common features of all subsystems of the multi-site approach are:

- All components are based on knowledge-based techniques, i.e. problem-specific knowledge is identified, represented, and applied for the solution of the addressed problem. The heuristic and domain knowledge, e.g. about preferable orders, alternative routings or machines, relaxable constraints, etc., is represented declaratively in terms of facts, rules, and constraints.
- Several problem solving techniques have been investigated for use in the scheduling components, see below.
- The reactive scheduling components on both scheduling levels are realized as a leitstand with a sophisticated graphical user interface, where the schedule is repaired interactively using knowledge-based reactive search algorithms, which make use of heuristic knowledge to solve the constraints violated.
- The user interfaces are window-oriented and most functions are mouse-sensitive. Thus providing a comprehensive presentation of and easy access to relevant information, e.g. graphical Gantt-chart representation of the current schedule (on different levels of abstraction) as well as tools for user-friendly manual interaction such as deletion, relocation, and substitution of resource allocations.
- The database components provide access to global respectively local databases containing global or local scheduling information, e.g. the master schedule, information about products, resources, orders, inventory, etc.
- Each subsystem contains two communication interfaces for the information exchange within MUST and the integration of the MUST system into an existing organizational environment.

Global and Local Scheduling

Global and local scheduling problems can be modeled similarly by the five-tuple (R, P, O, HC, SC) (Sauer 93), where R denotes the set of required resources, P the set of producible products, O the set of actual orders, and HC and SC stand for the sets of hard and soft constraints, respectively. Table 1 shows this model applied to global and local scheduling with examples for the items.

The solutions are a global schedule, which determines the requirements for the local scheduling level in terms of temporal assignments of intermediate products to machine groups and several local schedules showing the local assignment of production steps to machines. As mentioned earlier the events of the local and global level are also quite similar.

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 approach. Table 2 shows the tasks and some of the appropriate methods from which several are investigated in the MUST project.

	Local Scheduling	Global Scheduling
R	machines	groups of machines
P	intermediate products consisting of several production steps (operations)	final products consisting of several intermediate products
O	internal orders for intermediates	external orders for final products
HC	schedule all orders, regard production requirements (one variant, precedence constraints)	schedule all external orders, regard production requirements (one variant, precedence constraints, capacity)
SC	"optimal" machine utilization, meet due dates, minimize work-in-process costs.	meet due date, minimize transportation times/ costs, use production equally, reduce inventory costs.

Table 1: Global and Local Scheduling

Scheduling 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

Table 2: Scheduling Tasks and Methods

For the global predictive tasks a heuristic approach and a fuzzy-logic approach are realized.

The heuristic approach (Bruns and Sauer 1995) represents an order-based strategy which aims at creating a global schedule with a 'balanced' use of machine groups and time intervals trying to avoid bottlenecks and to provide as much latitude as possible for rescheduling activities that will not affect major parts of the global schedule. The heuristic knowledge used in this strategy is represented by heuristic rules or procedures and uses a dynamically updated worst case analysis of the capacity needed by the external orders. If a conflict has to be solved alternative machine groups are checked. If there is no feasible alternative, then new start times for the intermediates or alternative routings are tried. At last the given time interval for the final product is expanded in order to find a solution.

In the fuzzy-logic characteristic data are described by linguistic variables and fuzzy values, and fuzzy-rules are used to infer new information (the schedule) (Sauer, Suelmann, and Appelrath 1998). This gives an appropriate possibility of representing and processing the imprecise data of the global level, e.g. information about needed capacity by (`very small, small, medium, high, very high`). The soft constraints can be represented by fuzzy sets, too. The represented data is used by fuzzy rules, e.g.

```
IF capacity_needed(very_high)
FUZZY_AND time(already)
THEN priority(very_high),
```

to process new information, e.g. an ordering of the products to be scheduled. The system implemented uses a two level strategy for fuzzy scheduling. On the upper level a scheduling strategy is represented, e.g. an order-based problem decomposition like

```
find_ordering_of_products_to_be_scheduled
AND schedule_products_by_priority
```

and on the lower level the rule bases for the scheduling steps are evaluated.

The fuzzy logic approach leads to good results and provides a good representation and handling of imprecise knowledge. Thus it will be part of (not only of our) scheduling systems of the future.

The global reactive component of the MUST system has been designed as a (global) leitstand supporting interactive repair by means of a sophisticated graphical user interface as well as a heuristic reactive scheduling algorithm (Lemmermann 1995). It uses several repair strategies based on the conflicts that evolve from the events, e.g. capacity overflow of a machine group. Figure 3 shows a part of the user interface of the global leitstand, representing an order-based view of the global schedule with the local realization and communication information.

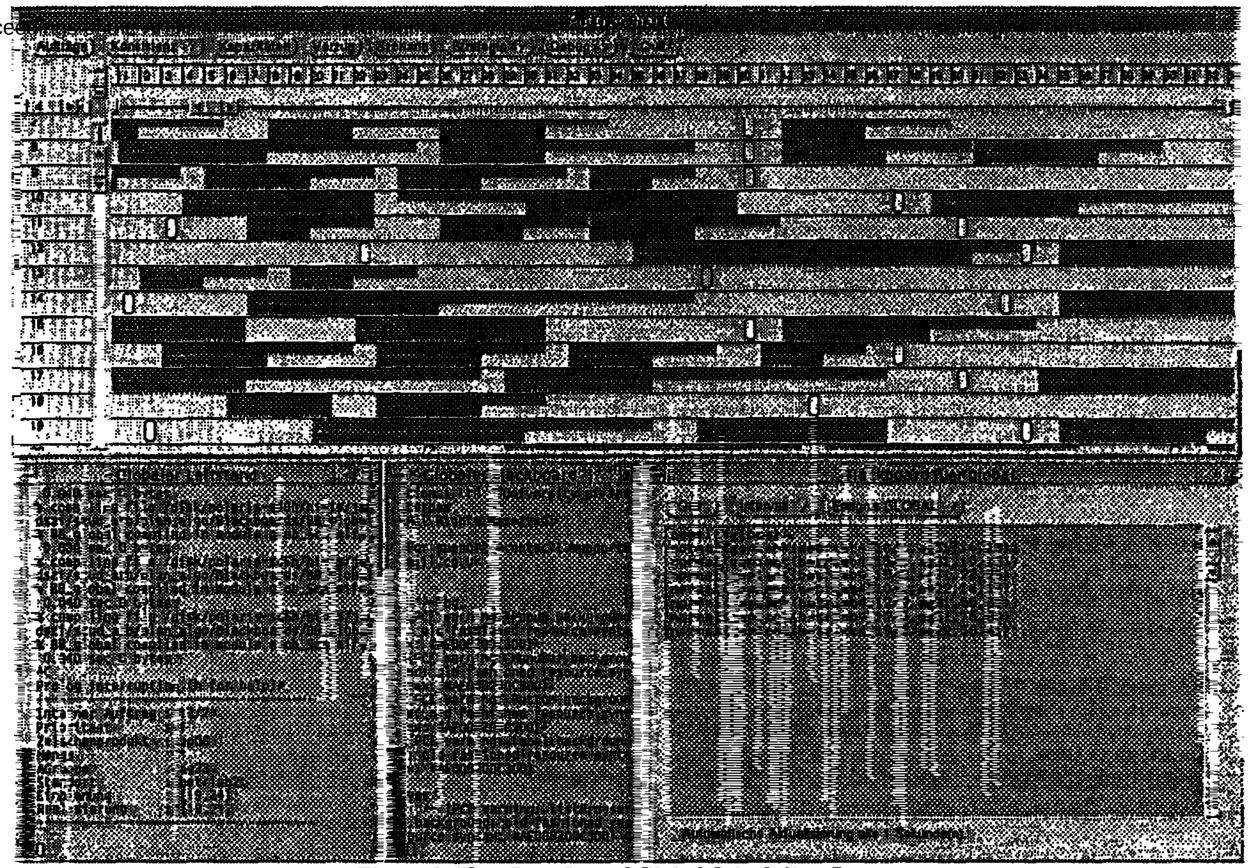

Figure 3: User Interface of Global Scheduling System

For local scheduling existing scheduling systems from previous projects and approaches from literature have been exploited.

For the local predictive scheduling tasks several approaches have been implemented. Early versions are based on heuristic search using order-based or resource-based problem decomposition as well as priority rules from operations research (Sauer 1993, Sauer and Bruns 1997). New approaches are based on iterative repair (Dorn 1995) and neural networks.

A genetic algorithm has been implemented that uses a new problem oriented direct representation of the individuals (the schedules) and new operators to deal with the representation (Bruns 1993). A schedule is represented by a set of operation/variant/machine/interval-assignments. The operators for crossover and mutation in the selection-crossover-mutation-evaluation cycle of the genetic algorithm have been designed to guarantee consistent schedules by using problem specific knowledge.

Additionally, an approach on the basis of neural networks was developed (Märtens and Sauer 1998). It views scheduling as combinatorial optimization problem and uses a combination of heuristic problem decomposition and neural networks to generate a schedule. To make to problem size small enough for handling it with a neural network, a three level problem decomposition consisting of the steps

- select variant for each order
- select machine for each operation of each variant
- select time interval for each machine/operation assignment

is used. Three networks are used to solve the reduced problems. For the selection of the variants and the machines a Hopfield-network is used where each neuron represents a possible assignment of an order to a variant resp. of a operation to a machine. The solution of the network is the best fitting assignment regarding the evaluation function. For the assignment of time intervals a linear programming formulation of the problem is realized with a LP-network showing the variables and the constraints imposed on the variables.

Both approaches lead to good results but they are very time consuming, especially the neural network, which restricts their applicability in a drastic way.

The local reactive scheduling is based on a similar approach as the global reactive part. A local leitstand has been realized including interactive as well as heuristically guided reactive repair possibilities (Henseler 1995).

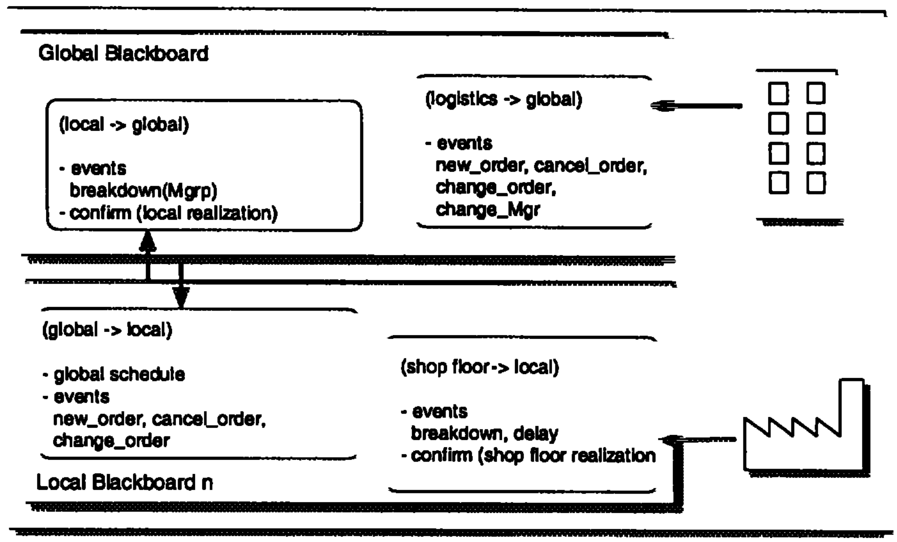

Figure 4: Communication in Multi-Site Scheduling

Communication

A vital part of the multi-site scheduling system is the communication between the scheduling levels, between the global level and the logistics department, and between the local level and the shop floor. Communication shall provide all participating subsystems with actual and consistent data. The actual global schedule has to be provided for all local schedulers and all local events affecting the global and other local scheduling systems have to be provided immediately for the global scheduling system. Figure 4 shows the data transferred between the levels.

The communication and event handling is realized with a blackboard approach. Every scheduling system has a blackboard for the presentation of the important events and tasks. Figure 4 shows also the blackboards of the two levels. A part of Figure 3 shows the global blackboard and other communication information in the actual prototype of the MUST system. The event handling implemented in the global and local scheduling systems allows in the normal active-state the processing of events and the performing of scheduling tasks. Other states of the event handling are the sending and receiving of data to resp. from the other systems.

Conclusion

The problem of multi-site scheduling together with an approach for an architecture and its implementation have been presented. The multi-site scheduling system supports all the scheduling and coordination tasks of a distributed production environment.

Within the system several existing methods of solving scheduling problems as well as new problem solving techniques have been evaluated. All scheduling subsystems of the two level multi-site approach consist of a user interface allowing interactive as well as predictive and reactive scheduling and of communication facilities for data exchange between the systems and the environment.

Actual research is done on further scheduling approaches, on other communication possibilities, e.g. based on protocols like contract nets, on a multi-agent realization of the system, on object-oriented techniques for realization of scheduling systems and on a design support system which integrates our experience to support the creation of knowledge-based scheduling systems (Sauer and Appelrath 1997). It will combine knowledge about the design of scheduling systems with the necessary components to build a system that fits the users needs.

References

Bruns, R. 1993. Direct Chromosome Representation and Advanced Genetic Operators for Production Scheduling. In *Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo: Morgan Kaufmann.

Bruns, R.; Sauer, J. 1995. Knowledge-Based Multi-Site Coordination and Scheduling. In R.D. Schraft et al. eds. *Flexible Automation and Intelligent Manufacturing 1995*. New York: Begell House.

Bel, G.; Thierry, C. 1993. A Constraint-Based System for Multi-Site Coordination and Scheduling. In *Workshop Notes of the IJCAI-93 Workshop on Knowledge-Based Production Planning, Scheduling and Control*, Chambery, France.

Dorn, J.; Froeschl, K.A. 1993. *Scheduling of Production Processes*. Chichester: Ellis Horwood.

Dorn, J. 1995. Iterative Improvement Methods for Knowledge-Based Scheduling. *AICOM*, Vol. 8, No. 1, March.

Henseler, H. 1995. REAKTION: a system for event independent reactive scheduling. In Kerr, R.; Szelke, E. eds. *Artificial Intelligence in Reactive Scheduling*, London: Chapman & Hall.

Hsu, W.-L.; Prietula, M.J.; Thompson, G.L.; Ow, P.S. 1993. A mixed-initiative scheduling workbench: Integrating AI, OR and HCI. *Decision Support Systems*, 9, 1993, 245-257, Amsterdam: North-Holland.

Kerr, R.M.; Szelke, E. 1995. *Artificial Intelligence in Reactive Scheduling*. London: Chapman & Hall.

Lemmermann, D. 1995. *Globaler Leitstand*. Diploma Thesis, FB Informatik, Universität Oldenburg.

Liu, J.S.; Sycara, K.P. 1993. Distributed Scheduling through Cooperating Specialists. In *Proc. of the IJCAI-93 Workshop on Knowledge-Based Production Planning, Scheduling, and Control*, Chambery, France.

Märtens, H.; Sauer, J. 1998. Ein Ablaufplanungssystem auf Basis neuronaler Netze. In Biethahn, J. et al. eds. *Betriebswirtschaftliche Anwendungen des Softcomputing*. Wiesbaden: Vieweg.

Sauer, J. Appelrath, H.-J. 1997. Knowledge-Based Design of Scheduling Systems. In *Proc. of World Manufacturing Congress*, Auckland, NZ.

Sauer, J. 1993. Meta-Scheduling using Dynamic Scheduling Knowledge. In Dorn, J., Froeschl, K.A. eds. *Scheduling of Production Processes*, Chichester: Ellis Horwood.

Sauer, J.; Bruns, R. 1997. Knowledge-Based Scheduling Systems in Industry and Medicine. *IEEE-Expert*, February.

Smith, S.F. 1992. Knowledge-based production management: approaches, results and prospects. *Production Planning & Control*, Vol. 3, No. 4.

Sauer, J. Suelmann, G. Appelrath, H.-J. 1998. Multi-site Scheduling with Fuzzy Concepts, *International Journal of Approximate Reasoning*, Amsterdam: Elsevier.

Wauschkuhn, O. 1992. *Untersuchung zur verteilten Produktionsplanung mit Methoden der logischen Programmierung*, IWBS Report 215, Wissenschaftliches Zentrum, IWBS, IBM, Stuttgart.

Zweben, M.; Fox, M.S. eds. 1994. *Intelligent Scheduling*. San Mateo: Morgan Kaufmann Publishers.