An Agent-based Approach to the Control of Flexible Production Systems

Bussmann, S. and Schild, K.

DaimlerChrysler AG, Research and Technology 3, Alt-Moabit 96a, 10559 Berlin (GERMANY)
T. +49-30-39982-215, F. +49-30-39982-209 Email: {Stefan.Bussmann, Klaus.Schild}@daimlerchrysler.com

We present an agent-based approach to the control of flexible production systems. This approach flexibly adapts to changing production conditions and is suitable for high volume production. In this approach, workpiece agents auction off their current task, while machine agents bid for tasks. When awarding a machine, a workpiece agent takes into account not only the machine's current workload, but also the workpieces leaving the machine. If a machine's outgoing stream is blocked, the machine agent eventually ceases to accept new workpieces, thus blocking its input stream as well. As a result, a capacity bottleneck is automatically propagated in the opposite direction of the material flow. A unique feature of this mechanism is that it does not pre-suppose any specific material flow; the current capacity bottleneck is always propagated in the opposite direction of the actual flow, no matter how this flow looks like. DaimlerChrysler has evaluated the new control approach as a bypass to an existing manufacturing line. A suite of performance tests demonstrated the industrial feasibility and the benefits of the approach.

I. INTRODUCTION

Today's manufacturing industry is facing a major shift from a supplier's to a customer's market. The growing surplus of industrial capacity provides the customer with a greater choice and increases the competition between suppliers. Aware of their power, customers have become more demanding and less loyal to a particular brand. As a result, companies must shorten product-life cycles, reduce time-to-market, increase product variety, and instantly satisfy demand, while maintaining high quality and reducing investment costs.

These trends have two major consequences for the manufacturing process [1]. First of all, processes must become more flexible in order to cope with constant product changes and an increasingly volatile demand. Secondly, processes must show more robustness with respect to disturbances in order to maximize the total use of the manufacturing equipment, and thus to further reduce the overall investment costs. Flexible and robust production systems, in turn, require an intelligent control system that makes efficient use of the (hardware) flexibility. Current control technology is no longer sufficient because it has been developed to optimize a fixed process. Future manufacturing processes will have to continuously adapt their process to changing production conditions.

Software agents are the right information technology to meet this challenge. They model the manufacturing process as it is with no artificial central control unit. Resources are allocated dynamically by a continuous coordination process among the relevant agents. Unlike in Computer Integrated Manufacturing, there is no need to handle all the contingencies of a complex manufacturing process at design time; rather, agents negotiate proper allocations among themselves during execution. Although some of their joint decisions may not be optimal, all decisions are, nevertheless, made on the basis of the current situation – in the long run leading to a better system performance.

Under the leadership of DaimlerChrysler, an industrial consortium was formed to meet the challenges of manufacturing in modern automotive industry. The consortium developed a new production system for flexible and robust manufacturing, called *Production 2000+*. To achieve the envisioned flexibility, the consortium addressed three important aspects:

(i) flexible machines providing the range of operations necessary to produce any variant of a product type;
(ii) the ability to download during operation the NC programs specifying how to process each variant; and
(iii) a flexible transportation system able to move a workpiece from any machine to any other machine as required by the processing graph of this variant.

In this paper, we will report on the development of the flexible transportation system and the associated agent-based control. While the technical details of the implementation are proprietary to the consortium, we will present the overall architecture of the system. We will also give a detailed analysis of the agent-based

publl. at the 8th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA 2001), pp. 169 – 174. Antibes Juan-les-pins, France, 2001.

control mechanism. The analysis is complemented by a brief report on a series of realistic simulations as well as performance tests on an industrial prototype. We begin with a short overview of software agent technology.

II. SOFTWARE AGENT TECHNOLOGY

Software agents offer a new approach to designing and building complex distributed systems that significantly extends previous technologies like object-oriented or distributed computing [4]. Instead of modeling distributed systems as rigid programs exchanging data and commands, agent technology creates autonomous decision makers which communicate their preferences, negotiate sub-goals, and coordinate their intentions in order to achieve individual or system goals [7,14]. This decision- and interaction-based approach to computing makes it possible to build systems that can dynamically react to unforeseen events, incorporate different preferences and attitudes, exploit different capabilities of components, and adapt flexibly to changes in the environment. The ability of agents to adapt their behavior during computation reduces the need for the designer to foresee all possible scenarios and changes the system will encounter [4]. Moreover, an agent-oriented design often corresponds to the distributed nature of decision making in many application domains and thus increases the understandability and maintainability of the software system.

There are many possible definitions of a software agent. Within the context of manufacturing control, we define a *software agent* to be a software process with two distinct properties:

(i) goal-based decision making in a dynamic environment, and
(ii) flexible interaction with other autonomous decision makers.

There is a wide range of agent architectures for autonomous goal-based decision making, as well as sophisticated techniques for flexible interaction between autonomous agents. The interaction techniques developed cover aspects like coordination, negotiation, planning [7]; distributed problem solving, distributed rational decision making, and multi-agent learning [14]. The coordination and negotiation techniques are of particular relevance to manufacturing control because they enable agents to allocate scarce resources in real time.

The advantages of agent technology have been widely recognized and have led to a wide range of applications [3]. The domain of manufacturing control, in particular, has been the target of many agent applications in the past [9]. To our knowledge, however, none of these applications in manufacturing control have been demonstrated under industrial conditions or have even been deployed in an industrial process. This paper presents an agent-based control system that has been tested in an industrial environment and is in operation at a DaimlerChrysler plant.

III. THE MANUFACTURING SYSTEM

In 1997, DaimlerChrysler initiated the project *Production 2000+* (P2000+ for short) in order to design a flexible and robust production system for large-series powertrain manufacturing. Large-series manufacturing requires high volume and low costs per product. Both requirements are met by transfer lines which are currently widely used in manufacturing. Transfer lines, however, have two essential disadvantages. First of all, transfer lines are designed to produce a specific product, and any (even minor) product changes require costly reconstruction of the line. Secondly, in terms of system throughput, transfer lines have poor overall performance because the failure of a single machine creates a backup of workpieces that soon blocks the preceding machine and eventually causes the whole line to stop.

To overcome the deficits of transfer lines, while maintaining high volume and low costs, the DaimlerChrysler plant at Stuttgart-Untertürkheim designed a new production layout which is both more flexible and more robust than existing transfer lines. The P2000+ system consists of flexible CNC machines (with three-, four-, and five-axes) which are installed along a flexible transportation system (see figure 1). The CNC machines are equipped with tools such that (i) each necessary operation is provided by at least two machines, and (ii) in case there are no disturbances, every machine is fully utilized. For each operation, the production system thus has an alternative machine without having any overcapacity. As a consequence, a single machine failure never causes the whole production system to stop.

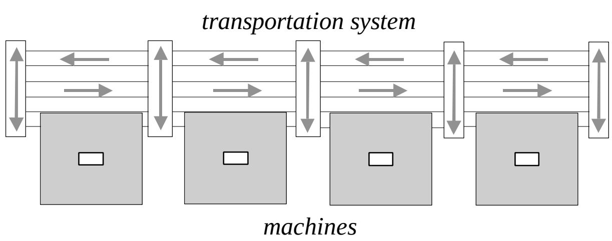

Fig. 1: The new production layout.

An example configuration of a P2000+ system is given in figure 2. The example configuration has six machines, of which each two machines are identical (i.e., machines M_1 and M_2, machines M_3 and M_4, and machines M_5 and M_6 are identical and have identical tools). All machines run at a cycle time of 60 seconds which results in a system cycle time of 30 seconds. Thus, all machines are fully utilized if there are no disturbances. If a single machine breaks down, the system throughput is reduced to 50%. In contrast to transfer lines, however, the production system still produces output as long as there is at least one alternative machine per operation available.

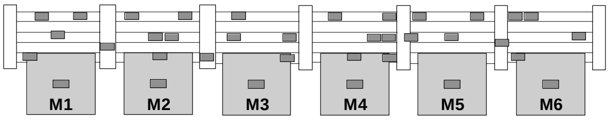

Fig. 2: Example configuration.

To supply each machine with workpieces even if some machines are disturbed, the machines are connected by a flexible transportation system. The transportation system consists of three conveyor: a *forward*, a *backward*, and a *supply* conveyor (see figure 3). A workpiece passes the machines on the forward conveyor until it reaches its goal machine. It is then moved by a shifting table from the forward conveyor to the supply conveyor. The machine takes the workpiece off the supply conveyor, processes it, and puts it back onto the supply conveyor. From there, the workpiece moves to the next shifting table which puts it either onto the forward or the backward conveyor, depending on the direction of the next goal machine. A shifting table is installed between every pair of adjacent machines in order to allow for individual supply of each machine.

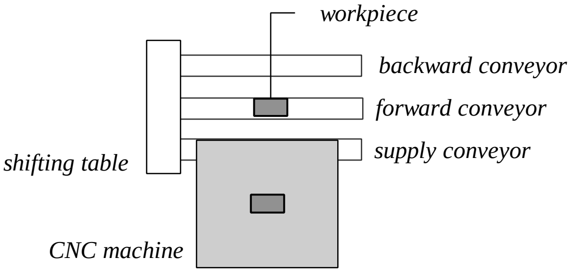

Fig. 3: System components.

Workpieces are fixed to pallets and remain on the pallet until all processing is finished. Pallets are uniquely identified by an identification number which is read in front of each shifting table and each machine. A P2000+ system has one or more loading stations where new workpieces are fixed to a pallet and introduced into the transportation system. Each loading station has a small buffer from which it can choose the next workpiece to be loaded. The system has also one or more unloading stations where the finished workpieces are taken off the pallet and shipped to their destination.

IV. THE AGENT-BASED CONTROL SYSTEM

A P2000+ layout consists only of standard hardware components, in particular CNC machines. The control of such a flexible system, however, requires a completely new approach. Workpieces should be assigned to machines at the latest moment possible because neither the necessary processing steps for each workpiece nor the current availability of the machines is known until the workpiece is introduced into the system. Consequently, planning and scheduling should be interleaved with the execution and distributed to those components which have the most up-to-date information, i.e., to the production components (cf. the design rationales in [1]).

DaimlerChrysler Research and Technology therefore developed a strictly decentralized approach to manufacturing control, called *West* (in German: *Werkstücksteuerung*). In this approach, a specific agent is associated with each workpiece, each machine, and each shifting table (cf. figure 4). A *workpiece agent* manages the processing state of the workpiece. A *machine agent* controls the machine and the material flow through the machine. To this end, every machine agent manages what we call a *virtual buffer*. This buffer includes not only the machine's current work in process, but also the workpieces trying to leave the machine; that is, all those workpieces which have already been processed by the machine without yet being able to find an appropriate new machine. A third type of agent, a *switch agent*, controls the shifting table. It decides autonomously in which direction a workpiece is forwarded.

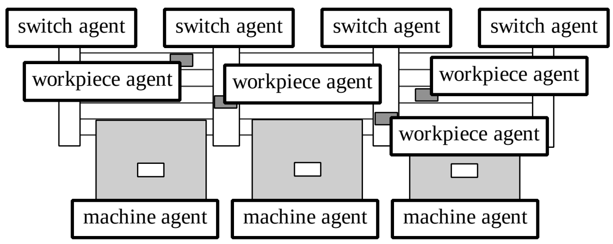

Fig. 4: Control agents.

All these agents constitute parallel processes. These processes are, of course, not independent; they have to be coordinated during execution. Proper coordination is achieved by special negotiation procedures, which also take place simultaneously. A single workpiece negotiates with the machines which of the machines should process the workpiece next.[1] The workpiece

[1] Whenever understood, we ignore the distinction between an agent and the physical component it controls.

auctions off its next operations by inviting machines to bid. Every machine bid includes information about the current state of the machine's virtual buffer. If a workpiece awards a specific machine, then this machine will be the next goal of the workpiece. The routing of a workpiece is organized through a sequence of bilateral negotiations, in each case between the workpiece and the next shifting table which the workpiece approaches.

The following subsections elaborate the different negotiation procedures.

A. Controlling buffer sizes

Each machine agent manages two buffers, an *input* and an *output buffer*. The input buffer contains all those workpieces which awarded the machine and have not been processed yet. This is the machine's *work in process*.

A machine's output buffer tracks all those workpieces that have already been processed by the machine, but have not been able yet to award an appropriate new machine. A workpiece thus moves from the input to the output buffer after it has been processed by the machine. The input and output buffer together represent the *virtual buffer* of the machine.

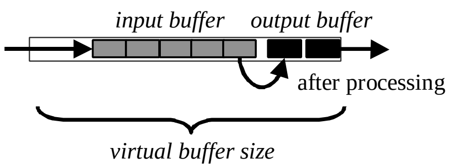

Fig. 5: Virtual buffer of a machine.

Every machine is assigned a maximal size of its virtual buffer. This size may vary from machine to machine. It should, however, never exceed the actual capacity of the physical buffer associated with the machine; that is, the section of the transportation system located between the two shifting tables adjacent to the machine (cf. figure 3).

A machine may only accept a new workpiece if its virtual buffer is smaller than its buffer limit. This rule ensures that the production system is never overflowed with workpieces and thus never runs into a transportation deadlock. Since every workpiece is always in at least one buffer and since buffer sizes are limited, the overall number of workpieces in the system never exceeds a global limit. We have proven in another paper [2] that for certain operational conditions *West* never violates this upper bound.

B. Dynamic task allocation

Each workpiece agent manages the processing state of a specific workpiece. To this end, the agent has to be aware of the particular product type of the workpiece. A *product type* is characterized by the exact sequence of operations which are to be applied to the workpiece. The *state* of the workpiece consists of the sequence of operations already performed, and the current task of the workpiece given a particular state is the set of operations that still need to be applied to the workpiece.

The current task (or at least a subtask of it) has to be allocated to a machine able to perform the operations required next. In *West*, this allocation is determined in parallel to the actual processing of workpieces. It would be impossible to make use of the flexibility provided by machines without such a late commitment. The workpieces choose the next machine at the latest moment possible (the moment they leave the last machine) and are thus able to take into account the current capability, availability, and workload of the machines.

The dynamic allocation of a workpiece task is carried out by a simple first-price, single-round auction, with which a workpiece auctions off its current task. Each round involves three steps: the call for bids, the bidding itself, and the awarding. These steps are described in detail below.

Step 1.

The protocol is always initiated by a workpiece agent: Whenever a workpiece first enters the manufacturing system and, thereafter, immediately after it leaves a machine. In all cases, the workpiece determines its current task and all machines which are configured to perform the task (e.g., by looking into a static configuration list). The workpiece then sends an invitation to bid to all these machines. This call always includes the current task of a workpiece.

Step 2.

If a machine M receives an invitation to bid for a task, it checks whether or not it is able to perform at least part of this task. If this is the case, then M issues a bid; otherwise, it simply ignores the call. Short-term disturbances of some of M's operations are also ignored here, i.e., the machine bids even if it is currently broken down. This is because the subject of the negotiation is a *future* allocation of the workpiece's next operations and the current situation obviously does not provide much information about a machine's state when the workpiece enters the machine.

M issues no bid without making sure that it is actually ready to accept a new workpiece; it therefore checks whether the virtual buffer has already reached its limit. If this is the case, then it does not answer the call. Otherwise, the machine sends a bid including the following information:

(a) the current size of M's virtual buffer, and
(b) the maximal subtask of the workpiece's task which contains only operations of M.

Step 3.

The workpiece collects all bids for a specific call. If there is no bid at all, then the workpiece issues another invitation to bid, continuing with step 1. Otherwise the workpiece compares the bids and awards the best bid. For this, both components (a) and (b) of a bid are relevant, with (a) having a higher priority. The lower the current size of the virtual buffer, the better. The more operations the maximal subtask offered by the machines contains, the better. The workpiece awards the bid which is best in this sense. The awarded machine then includes the relevant workpiece in its input buffer.

Usually, a number of auctions of this kind take place simultaneously, even in an interleaved manner. Thus a single machine may participate in more than one auction at a time.

C. Avoiding deadlocks

As it stands, the auction protocol described above is not able to handle deadlocks in the allocation process: A number of machines may find themselves in a cyclic dependency, each waiting for another machine to be ready to accept a new workpiece. Consider two machines M_1 and M_2. Let W_1 and W_2 be workpieces residing in the output buffer of M_1 and M_2 respectively. Moreover, suppose that W_1 may allocate its current task to only M_2, while W_2 may allocate its current task to only M_1. If M_1 and M_2 are not ready to accept a new workpiece, then M_1 and M_2 actually find themselves in a deadlock.

To resolve these kinds of deadlocks, *West* defines a forward direction of the main material flow. The basic idea is that most workpieces follow the forward direction during their processing, and that in only a few cases this forward direction is violated (this is certainly true for high volume production). Given this forward direction, *West* is able to distinguish between *forward* and *backward successors* of a machine (cf. figure 6). The bidding procedure is now adapted as follows.

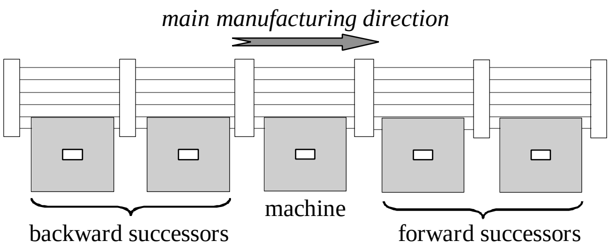

Fig. 6: Forward and backward successors of a machine.

A machine always bids if it has a subtask to offer, even if the machine is currently not ready to accept a new workpiece. Such a bid, however, is not made without including in it an explicit warning that the machine is currently not able to accept any new workpieces into its input buffer. When a workpiece awards a machine, it ignores any bid with such a warning. This rule is only violated if the following two conditions are met at the same time:

(a) There is no bid from a forward successor of the machine where the workpiece was processed last.
(b) There is no bid which does *not* contain a warning.

If both conditions are met, a machine with a warning may be selected in step 3 of the auction protocol. Although the chosen machine is not ready to accept a new workpiece, it is awarded anyway. Such an award is always combined with a request to include the workpiece into the input buffer, irrespective of the current size of the virtual buffer. This is what we call an *enforcement award*.

Such an enforcement award may exceed the upper bound of a machine's virtual buffer. However, under certain, general conditions, this violation of the machine's upper bound never results in a violation of the sum of the upper bounds at system level. A formal proof is given in [2].

D. Dynamic routing

Once a workpiece has awarded a specific machine for further processing, the workpiece must be moved to its new goal. In a layout like the one depicted in figure 1, there is usually a vast number of different paths ultimately leading to the same goal. Of course, the shortest of these paths should always be preferred. More important than optimizing the routing, however, is the avoidance of any congestion, which may have disastrous consequences on the performance of the overall system. In an unpredictable environment like a manufacturing system, jams can be avoided only by strictly separating the actual routing from the transportation goal itself.

In *West*, such dynamic routing is ensured through a sequence of bilateral communications between the workpiece and the next shifting table it approaches. A shifting table always tries to move a workpiece directly to its goal, thus trying to optimize the routing. If an exit is not available (e.g. because it is blocked), then an alternative route is taken. In this case, however, the priority of the workpiece is incremented by one. These priorities are used to decide which workpiece to prefer if a shifting table has more than one possibility: the workpiece with the highest priority is always served first. This is to avoid that a workpiece cycles eternally instead of reaching its actual goal eventually.

V. EVALUATION

So far we have defined the behavior of the individual agents used in *West* and described in detail how these agents interact with each other. In first subsection we shall now analyze the behavior which results at system level. Some observations are quite obvious; others give some deeper insights. In the second subsection, we will report on simulation results and the industrial prototype which was installed and tested at the DaimlerChrysler plant.

A. Analysis

Let us begin with some simple observations. Whenever a machine breaks down, the *West* auction mechanism automatically diverts the material flow to other machines, thus balancing the machines' workload. This is achieved by including the current workload of a machine (i.e., its current buffer size) in a bid, and by awarding the machine with the smallest workload. In case of a machine breakdown, the buffer of this machine runs full very quickly and workpieces start to avoid this machine because its buffer size is high. As soon as the machine is available again, it starts processing and workpieces begin awarding the machine again because the buffer size is decreasing. The workload of a machine thus adjusts itself automatically to its current processing capacity. *West* has this important feature in common with other agent-based control mechanisms proposed in the literature (see for instance [5,6,8,12]).

In mass production, it is also important to enforce a material flow in the main manufacturing direction. With a high volume, an unconstrained material flow would result in a high traffic in every direction and would thus require a transportation system with a high capacity. Such a transportation system would be very costly. It is therefore necessary to enforce a main manufacturing direction without sacrificing the possibility of violating this flow whenever it turns out to be inevitable. This is exactly what *West* does. When awarding a machine, forward successors (in the main manufacturing direction) are always preferred in favor of backward successors. Only if there is no bid from a forward successor, a backward successor may be awarded.

But there is even more to *West*. What really is a unique feature of *West* is that it automatically adjusts itself to the current capacity bottleneck while avoiding congestion in the transportation system. The system's current capacity bottleneck is automatically propagated in the opposite direction of the actual material flow; a process which continues until it has reached the loading stations. As a result, the loading stations feed only as many workpieces into the system as the system is currently able to handle.

To see how this works, let us again consider figure 5. A workpiece which wants to be processed by a machine is first put into the input buffer of the machine. After processing, the workpiece is moved from the input buffer to the output buffer and remains in the output buffer until the workpiece has been included into the input buffer of another machine. Entering the input buffer of another machine is the only way for a workpiece to leave the virtual buffer of the current machine. A machine's virtual buffer can therefore be thought of as a funnel whose input stream is controlled by its output stream (cf. fig. 7). As soon as the number of workpieces in the funnel reaches the maximal size of the virtual buffer the machine is no longer able to accept any new workpieces. This state lasts until a processed workpiece leaves the output buffer. The input stream is thus automatically restricted to the maximal throughput of the output stream.

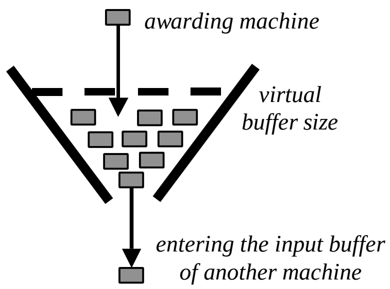

Fig. 7: Virtual buffer as a funnel.

One might object that as soon as the buffer capacity of the transportation system is at its limit the input stream is adjusted to the output stream anyway. However, a transportation system which is at its physical limit is jammed. Jams slow down the traffic and prevent workpieces from reaching a machine in time. Jams may even create transportation deadlocks (if the transportation system includes cycles as in the case of the P2000+ system). Jams may thus lead to a loss of performance which may be even greater than the cause of the congestion itself. The point is therefore to adjust the material flow through the machines *without* forcing the transport system to its physical limits.

But this is not even the end to the story. There is a whole network of funnels of the type just described. The current capacity bottleneck propagates through this network, eventually reaching the loading stations. An interesting feature of this kind of mechanism is that the topology of the network is not pre-defined; it is created dynamically by the workpieces themselves.

Take for example the production setting in figure 8. If the buffer of M_5 runs full (either because the machine is broken or the processed workpieces are not able to find a new machine), the machine no longer accepts any new workpieces in its input buffer. Any processed workpieces in machine M_2 or M_3 trying to award M_5 are now forced to wait for the virtual buffer of M_5 to decrease again. In the meantime, the virtual buffers of M_2 and M_3 run full also. Once full, these machines stop accepting workpieces, and the effect is propagated to the machines which supply M_2 and M_3 with workpieces. Machines M_1, M_4, and M_6, on the other hand, may continue processing because none of their workpieces award any of the affected machines.

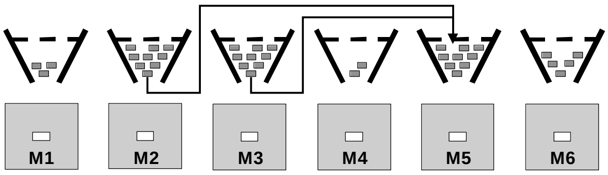

Fig. 8: Virtual buffers as a network of funnels.

The key aspect of this mechanism is that the machines do not have any knowledge about each other and their position in the processing sequence. The backpropagation of the current capacity bottleneck is solely determined by the workpieces and their processing requirements. A machine only stops processing if the virtual buffers of those machines are full which the workpieces in its output buffer want to award. If the processing requirements of the workpieces change, the direction of the backpropagation changes accordingly without the machines knowing about it. The backpropagation of the current capacity bottleneck in *West* is therefore a truly *self-organizing* mechanism.

B. Validation

In order to demonstrate the benefits of the new control approach, DaimlerChrysler has conducted a series of simulations, all of which are based on real product types and processing times. In particular, the disturbance characteristics have been taken from existing production systems. A typical simulation configuration consists of four blocks of identical machines. The number of machines in a block ranges from five to eleven machines, with 36 machines in total. The simulations have shown that the *West* mechanism is extremely robust against disturbances of machines as well as failures of control units. Its performance is nearly optimal. The following table summarizes the outcome of a typical simulation run.

Run time	**Number of machines**	**Average throughput**	**Theoretical optimum**
63h	36	70.54 workpieces/h	70.73 workpieces/h

Fig. 9: Example simulation results.

The *West* mechanism thus achieves about 99.7 % of the theoretical optimum.

We also tested a hypothetical production process with different product types. The operations were distributed over the machines in an irregular fashion and the average lot size of the workpieces fed into the system was one. Even under such unusual conditions, performance is still satisfactory. However, the complexity of the production process prevents us from computing the exact theoretical optimum here and, thus, also from conclusively assessing *West*'s performance in this particular setting.

DaimlerChrysler installed the new production system as a bypass to an existing large-series manufacturing line for cylinder heads. The bypass, located in a plant in Stuttgart-Untertürkheim (Germany), is shown in figure 10. The agent-based control system was implemented and installed by Schneider Electric [11]. The layout and the control system are basically as described in this paper.

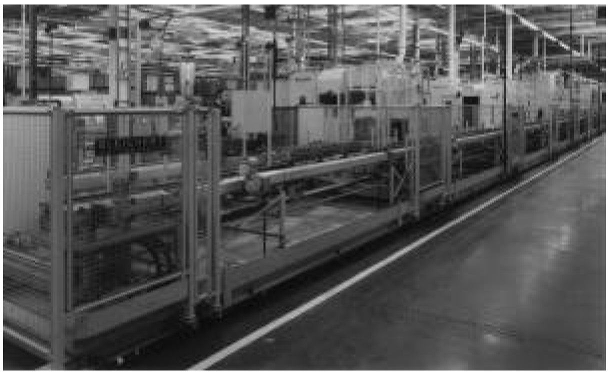

Fig. 10: DaimlerChrysler prototype in Stuttgart.
(Source: BLEICHERT Osterburken)

The bypass has undergone a series of performance tests. These tests showed that the results of the simulations are still valid under real manufacturing conditions. The prototype thus demonstrated the benefits and industrial feasibility of the agent-based control approach.

VI. CONCLUSION

In the sense of [10], manufacturing control is a *going concern*. The goal here is to continuously optimize the throughput rather than to solve a static problem where all relevant parameters are given from the start. Furthermore, manufacturing control is a kind of iterated game against nature [13], where the control system must cope with highly unpredictable events like resource disturbances and product changes. Consequently, our approach is to dynamically assign resources rather than to compute optimal schedules, which would be invalidated by constantly changing parameters anyway. To this end, we argued in favor of an agent-based approach, where workpieces auction off their current task while machines bid for a subtask. When awarding a machine a workpiece takes into account not only the machine's current workload, but also the stream of workpieces leaving the machine; that is, all those workpieces which have already been processed by the machine without yet being able to award an appropriate new machine. If a machine's outgoing stream is blocked, then the machine eventually ceases to accept any new workpieces, thus blocking its input stream as well. In this way, a capacity bottleneck is automatically propagated in the opposite direction of the actual material flow, leading to a self-organizing behavior of the control system.

With the DaimlerChrysler prototype, it has been demonstrated under industrial conditions that the agent-based control system developed in the project P2000+ is able to meet the challenges of flexible and robust manufacturing in the automotive industry. The production system was able to meet all performance expectations, in particular high product flexibility and nearly optimal production throughput. The prototype has thus proved that agent technology is able to meet the upcoming challenges of modern manufacturing control under industrial conditions.

ACKNOWLEDGEMENTS

We would like to thank Christian Anders, Christoph Siegel, and Gerold Winz from the plant in Stuttgart-Untertürkheim as well as Ralf Neubert and Ronald Schoop from Schneider Electric for the exciting time and fruitful cooperation during the course of the project Production 2000+. We would also like to thank our colleagues Hartwig Baumgärtel, Sven Brückner, and Harald Windisch for their invaluable contributions to the design and evaluation of the agent system.

REFERENCES

[1] S. Bussmann, D.C. McFarlane: Rationales for holonic manufacturing control. In *Proc. of the 2nd Int. Workshop on Intelligent Manufacturing Systems*, pp. 177–184. Leuven, Belgium, 1999.

[2] S. Bussmann, K. Schild: Self-organizing manufacturing control: An industrial application of agent technology. In *Proc. of the 4th Int. Conf. on Multi-Agent Systems (ICMAS 2000)*, pp. 87–94. Boston, MA, USA, 2000.

[3] N.R. Jennings, M.J. Wooldridge: Applications of Intelligent Agents. In NR. Jennings, M.J. Wooldridge (eds.), *Agent Technology – Foundations, Applications, and Markets*, pp. 3–28. Springer-Verlag, Berlin, Germany, 1998.

[4] N.R. Jennings: On agent-based software engineering. In *Artificial Intelligence*, Vol. 117, pp. 277–296, 2000.

[5] T. Kis, J. Vancza, A. Markus: Controlling distributed manufacturing systems by a market mechanism. In *Proc. of the 12th European Conf. on AI (ECAI'96)*, pp. 534–538. Budapest, Hungary, 1996.

[6] J. Maley: Managing the flow of intelligent parts. In *Robotics and Computer-Integrated Manufacturing*, Vol. 4, No. 3/4, pp. 525–530, 1988.

[7] G.M.P. O'Hare, N.R. Jennings (eds.): *Foundations of Distributed Artificial Intelligence*. John Wiley & Sons, New York, NY, USA, 1996.

[8] H.V.D. Parunak, J. White, P. Lozo, R. Judd, B. Irish, J. Kindrick: An architecture for heuristic factory control. In *Proc. of the American Control Conference*, pp. 548–558. Seattle, WA, USA, 1986.

[9] H.V.D. Parunak: Industrial and Practical Applications of DAI. In G. Weiss (ed.), *Multi-Agent Systems*, pp. 377–421. MIT Press, Cambridge, MA, USA, 1999.

[10] H.V.D. Parunak: From chaos to commerce: Practical issues and research opportunities in the nonlinear dynamics of decentralized manufacturing systems. In *Proc. of the 2nd Int. Workshop on Intelligent Manufacturing Systems*, pp. k15–k25. Leuven, Belgium, 1999.

[11] R. Schoop, R. Neubert: Agent-oriented material flow control system based on DCOM. In *Proc. of the 3rd IEEE Int. Symposium on Object-Oriented Real-Time Distributed Computing (ISORC 2000)*, 2000.

[12] M. Shaw, A. Whinston: Task bidding and distributed planning in flexible manufacturing. In *Proc. of the 2nd Conf. on AI Applications (CAIA'85)*, pp. 184–189. Miami, FL, USA, 1985.

[13] P. Wegner: Interactive foundations of computing. In *Journal of Theoretical Computer Science*, Vol. 192, pp. 315–351, 1998.

[14] G. Weiss (ed.): *Multi-Agent Systems*. MIT Press, Cambridge, MA, USA, 1999.