Advancing the Performance of Complex Manufacturing Systems Through Agent-Based Production Control

Arndt Lüder[1], Jacek Zawisza[1(✉)], and Alexander Becker[2]

[1] Otto-v.-Guericke University, Universitätsplatz 2, 39106 Magdeburg, Germany
{arndt.lueder,jacek.zawisza}@ovgu.de
[2] Volkswagen AG, Berliner Ring 2, 38440 Wolfsburg, Germany
alexander.becker@volkswagen.de

Abstract. Ever increasing competition is driving the efforts to improve productivity throughout nearly all domains. In the manufacturing context, digitalization of value networks and creation of autonomous, self-optimizing systems – a vision coined '*Industrie 4.0* – is an approach that promises competitive edge over other players. One field in which this vision could lead to great productivity potentials is order scheduling and sequencing in high variety, high volume manufacturing businesses like the automobile industry. A viable technology to realize the expected gains in productivity are software agents and multi-agent systems, since they provide autonomy, flexibility, adaptiveness, and robustness to unforeseeable events. This paper proposes an agent-based control architecture that enables communication between resources and customer orders within a car body shop, so that they can negotiate the best alternative schedule and order sequence in case of disturbances. The proposed architecture allows improvement of overall production system performance in terms of output, resource utilization, delivery reliability and others. Further, the paper describes the implementation and simulation of the multi-agent system with JADE framework and discusses the simulation results, which show that significant productivity leaps can be achieved.

Keywords: Industrie 4.0 · Multi-agent systems · Production control systems · Production scheduling · Car-sequencing

1 Introduction

Customer demand simultaneously fosters product segmentation and higher product individualization [1]. Consequentially, volume of each model and product variant declines, resulting in more complex and more competitive markets [2]. In order to keep up with demand and competition, manufacturing businesses have to constantly improve their productivity, adapt quickly to changes in customer demand, and eliminate waste throughout the value chain. However, to achieve new levels of productivity, new approaches become increasingly important [3]. In this context, *Industrie 4.0*, as one of the leading research and development initiatives, has envisioned an advanced production system control architecture and engineering methodology, to achieve leaps in

J.O. Berndt et al. (Eds.): MATES 2017, LNAI 10413, pp. 103–120, 2017.
DOI: 10.1007/978-3-319-64798-2_7

resource productivity and efficiency across entire value networks [4–7]. The underlying potentials are assumed to allow cost reductions of up to 40% in work in progress, up to 20% in processing, and even up to 70% in complexity reduction [2].

A field that could benefit greatly from advances in this area is the manufacturing sector, in particular, businesses with high product and production complexity, high product variety, and high volume. The automobile industry comprises a prominent representative of this kind of enterprise, where highly individual, complex products are being mass manufactured on mixed-model production lines. The same production method is also used in other segments like consumer electronics, white goods, furniture and clothing [8]. However, high product diversity requires detailed sequence planning in order to best exploit the potential of the production system [9].

This is achieved using mixed-model sequencing, which is an optimization problem from the domain of operations research and falls into the category of discrete and combinatorial optimization [10]. Research and industry have elaborated a broad range of approaches like using real options from finance domain, fuzzy goal programming, and particle swarm optimization to find satisfactory solutions in a given amount of time [11–13]. Those approaches factor in the restrictions of the underlying production system to give a near-optimal solution and have experienced considerable improvement over the last decades [14]. However, the underlying optimization problem is NP-hard, meaning that it cannot be solved in real-time [15]. Additionally, complex production systems with thousands of entities are subject to unforeseeable disruptions, that conventional, monolithic enterprise software is not designed to deal with. Multi-agent systems (MAS), on the other hand, provide the necessary properties to excel in dynamic environments [16]. This leads to the assumption, that the shortcomings of static scheduling algorithms could potentially be compensated by cooperating with dynamic multi-agent systems in order to achieve a better overall performance of production systems. Therefore, the following research questions will be discussed in this paper:

RQ1: How can decentralized control of production scheduling and sequencing with multi-agent systems improve the overall performance of complex production systems?

RQ2: How must a viable architecture for such a multi-agent system be designed and what tasks does each agent have to perform in order to realize productivity potentials?

To answer these questions, this work is structured as follows. In Sect. 2, the general mixed-model sequencing problem in the context of complex production systems is introduced and current challenges are illustrated. Following this, in Sect. 3, the approach of this work is described, requirements for the proposed MAS are derived, and the application case is presented. On this basis, the architecture of the MAS is explained and the elaborated agent communication is discussed. After this, Sect. 4 describes the basis of the performed simulations including input data and simulation approaches, and then focuses on the key performance indicators (KPI's) which are used to measure the performance of the system. The results of the simulations are discussed in Sect. 5 and the paper ends with a summary and outlook on future challenges.

In terms of scope and delimitations, the goal of this paper is to demonstrate the feasibility of applying multi-agent systems in the selected area and tapping into their potential. Therefore, focus is laid on the higher layers of the automation pyramid, mainly manufacturing execution system layer (MES). This, in turn, means that Enterprise Resource Planning (ERP) and field-level layers of the automation pyramid are not regarded. However, the work of [17, 18] shows that this is a realizable undertaking and a consequent next step. In real-world applications, deviations in cycle times of different product configurations can cause an overload of manufacturing equipment and, therefore, pose a challenge for production sequence planning [19]. However, due to the scope of this work, cycle times of each variant and within each variant type are assumed equivalent. This property can be implemented ex-post and is supported by existing sequencing algorithms as described in [8, 14], and alike. To enable future improvements, the application programming interface (API) of the designed software allows for easy exchange and adjustments of software modules and algorithms.

2 Production Scheduling and Control

2.1 Complex Products and Production Systems

Today, state of the art automobile factories produce several thousand cars per day. Each factory performs several thousand production steps to complete an order and has an equal amount of work in progress distributed on the production floor. This creates a very complex environment on production side already. On the product side, however, the situation is even more complex: due to high personalization of vehicles and thereby increasing product variety, manufacturers are confronted with billions of theoretically possible product configurations. For example, BMW offers up to 10^{17} configurations for its Series 7 and Daimler offers even up to 10^{24} for its E-Class [1, 20]. This product complexity raises production complexity even further and demands for suitable control approaches on the production side. Car wiring harnesses, for example, are each assembled individually and work only for the car it was designed for. It, therefore, is useless for any other car than this exact same one. To cope with the resulting complexity in supply chain and internal logistics, most automobile manufacturers have adopted the so-called “pearl chain logistics concept”. Here, the final assembly sequence is defined several days in advance to handle complexity and required materials are picked and sequenced in the predefined order [21–23].

2.2 Approaches for Sequencing Mixed-Model Assembly Lines

In research and industry, three main types of approaches for planning the optimal order sequence on mixed-model assembly lines have emerged: mixed-model-sequencing, car-sequencing, and level scheduling.

Mixed-Model-Sequencing. As a workload-oriented approach, mixed-model-sequencing focuses on avoiding or minimizing sequence-depending work-overload at individual workstations. This approach explicitly integrates operational characteristics like cycle times, personnel restrictions, station borders, etc. It, therefore, allows for high accuracy but requires significant effort considering data collection [8].

Car-Sequencing. In contrast, car-sequencing requires significantly less effort, since it considers the above-stated operational characteristics implicitly rather than explicitly by setting sequencing rules of type H_o:N_o (meaning that a maximum of H_o occurrences may be among N_o positions) [8]. The most successful implementations use heuristics like greedy algorithms, local search methods, genetic algorithms, and ant colony optimization methods [14]. Overall, this makes car-sequencing a valuable approach that is frequently used in practice, although it comes with the trade-off of lower accuracy.

Level Scheduling. The last approach to be mentioned in this paper seeks to optimize Just-in-Time (JIT) objectives, rather than workload. The overall goal is to distribute material requirements, which depend highly on the production sequence, as evenly as possible over the planning horizon. Therefore, target production rates are defined and product variants are sequenced according to those rates minimizing deviations [24].

Despite great efforts in academia describing alternative solutions, there is still a lack of empirical research evaluating the fit of sequencing approaches for real-world applications [8]. The next section tries to reduce this gap. Furthermore, manufacturers have a hard time following the specific sequence defined by the pearl chain concept, since perturbations and complex, parallel production lines tend to disrupt the planned order [25]. The resulting challenges are discussed in the following section.

2.3 Case Study: Challenges in the Automotive Sector

Both, high product and production system complexity, increase the possibility of disruptions during the manufacturing process. Even though concepts like Total Productive Maintenance (TPM) and Predictive Maintenance helped reduce previously unforeseeable failures in past decades, manufacturers constantly deal with disturbances.

Since disruptions in any of the thousands of participants of a production network are possible anytime, adaptation of the production sequence is a necessary ability. However, mixed-model scheduling is an NP-hard problem and a new near-optimal sequence cannot be calculated under real-time conditions [15]. Current algorithms require runtimes of about 30 min to calculate a new sequence, which is too slow for real-time adjustments [26]. Limiting runtime, e.g. to 10 min like in the ROADEF'05 challenge [14], enables faster reactivity, but generally leads to lower quality of the solution. As a consequence of both, dynamics of complex production systems and insufficient real-time abilities, the elaborated solution is often outdated the moment it comes into place. Therefore, it is inevitable for a robust production system to autonomously adapt to disturbances the moment they occur.

Conventional monolithic enterprise software, however, is not designed to cope with unexpected events, since it is based on a very detailed set of rules to cover very specific situations. On one hand, this allows excellent results in those predefined circumstances. On the other hand, monolithic systems tend to have poor performance when handling events that were not specifically defined in advance, because they cannot respond adequately and in a timely manner to such situations. Therefore, although many automobile factories have the possibility to virtually or physically resequence the production order at some point, they can only handle planned, predefined operations like e.g. building color blocks before entering paint shop [27]. Unplanned resequencing usually results in deviations from the originally planned sequence and causes problems down the production stream. Inconsistent data types can further increase such complications [28].

Additionally, the overall system complexity makes it extremely difficult for humans to manage the required information in real-time. However, it can be observed that humans are often the decision makers in such situations, which can lead to suboptimal results. As an example, the distribution of lead times often shows that the expected bell curve is stretched far to the right, with a significant part of orders having very long lead times [22]. However, in the experience of the authors, this effect can be found in about half of the factory output, and it can be argued, that ineffective production control is accountable for a significant part of this effect. The same can be observed in the distribution of product variants where unexpected events and disruptions often lead to unevenness of production as depicted in Fig. 1. This unevenness usually causes problems in assembly shop that can only process a certain number of variants in sequence. In a production control context, these kinds of deviations are often caused by disturbances like logistical restrictions, e.g. due to delayed material. Dispatchers counteract by releasing other orders to keep production running. These decisions, however are usually not entirely data-driven and must be taken quickly, resulting in the distributions illustrated below. Based on the experiences of the authors at different

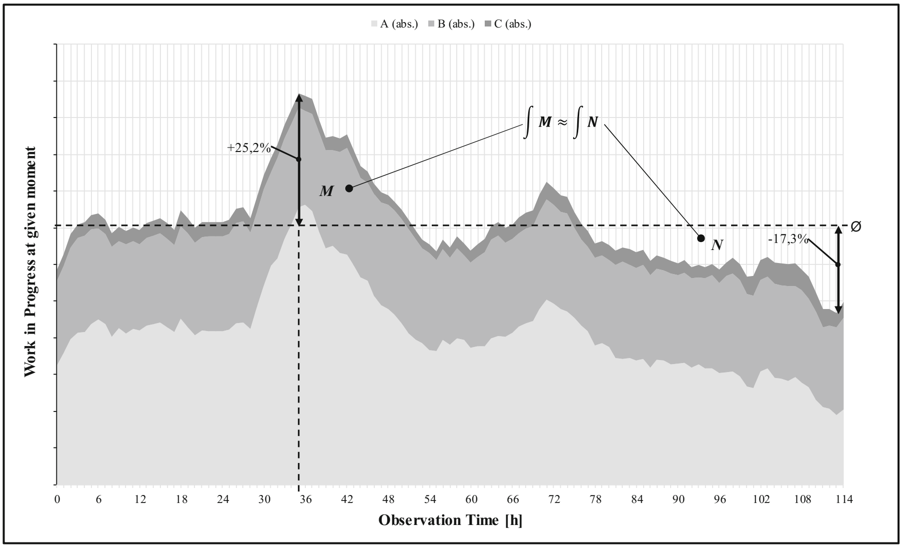

Fig. 1. Work in progress of three product variants A, B, and C

factories and manufacturers, these observations appear to be representative of the automobile sector.

To summarize, there are multiple approaches for sequencing mixed-model assembly lines and great effort is being made in both, academia and industry, to further improve the given tools. However, it can be observed that solutions like car sequencing are used in real-world applications, rather due to better interpretability for humans and lower computational effort then due to higher performance.

From this, two conclusions can be drawn: first, despite their complexity and real-time requirements, decisions in modern production systems are – to some extent – influenced by human decision-makers. And second, computation time is a critical factor for the success of a production control system in the manufacturing domain. Considering the fact that highly complex production systems with several thousand suppliers as well as hundreds of internal production resources are exposed to a substantial amount of unexpected disruptions, it seems rational to increase the amount of autonomy in that area. However, since the presented sequencing strategies already provide near-optimal solutions and significant runtime-improvements are not expected, it is difficult to achieve great productivity leaps in that area. It is far more likely to realize substantial improvements by adding more flexibility and autonomy to the production system itself, which is outlined in more detail in the following section.

3 Agent-Based Production Control System

3.1 Approach

A technology with high potential to address the challenges described above are multi-agent systems, because they are able to adapt to dynamic environments. Although there are many definitions of software agents, most researchers and authors agree on certain core properties characterizing software agents, like being autonomous (operating without external intervention), social (able to communicate with other agents), reactive (perceiving its environment and responding to changes), and proactive (taking initiative to achieve its goals) [29–31]. The agents' properties allow multi-agent systems to be highly flexible, adaptive, reconfigurable, and therefore also robust [31, 32]. However, by themselves, MAS do not deliver higher quality solutions for optimization problems, like e.g. mixed-model sequencing [33].

Therefore, the authors propose implementing a production control system that exploits the strengths of both approaches by combing existing scheduling algorithms and multi-agent systems. This way, on one hand, the system can continuously calculate new near-optimal production sequences for the next planning period that best suit the current situation and adapt iteratively to changes. On the other hand, unexpected disruptions that require real-time responses can be handled by the MAS, resulting in a highly optimized and highly adaptive system. In the next section, the requirements such a system must fulfill are discussed.

3.2 Requirements and Implementation-Framework

To address the defined challenges and equip the proposed production control system with the necessary properties a set of requirements must be met. Although multi-agent systems can provide a long list of valuable properties (see e.g. [29–31]), the following five requirements are considered especially important by the authors:

R1: Autonomy and Decentralized Control. To cope with its complex and dynamic environment, the designed system comprises of software-agents that are autonomous by definition. Each agent has its own goals and autonomously exercises plans to achieve them. Therefore, it controls its own behavior and operates without direct intervention of human supervisors [31]. It can and must, however, communicate with other agents.

R2: Flexibility and Adaptability. Unexpected disruptions must be handled effectively, meaning that agents must adapt automatically to dynamic changes in their environment. This was considered in the MAS by enabling agents to sense their environment, i.e. receiving the information they need, and to act upon it. As an example, if a resource suffers a malfunction and cannot process an order, the system must recognize this and find an alternative way to achieve its goal.

R3: Reconfigurability. Multi-agent systems offer the opportunity to provide plug & produce functionality to manufacturing businesses. In practice, this is a very valuable property, since it allows adding and removing resources depending on current demand. To ensure this functionality, the system is designed to allow registering and deregistering resources at runtime, providing all necessary information about the resource.

R4: Real-time capabilities. As shown in Sect. 2.3, it is necessary that the developed system can adequately respond to sudden changes. Since the system is working in the MES-context, response times in the range of minutes, like they are common in sequencing algorithms, are not allowed. Neither is the system responsible for direct control of field-layer equipment like PLCs, so it does not have to meet real-time requirements in the millisecond range. Therefore, the maximum response time for the system was designed to be below one second. Although most events are handled in significantly shorter time, this threshold is enough to make all required decisions.

R5: Modularity and extensibility. To facilitate modifications of the proposed system, the API was designed in a way that allows to easily replace or add modules. The system can thus be adjusted to the domain it is used in and the goals developers pursue. Examples for this are the scheduling and routing algorithms as well as simulation of resource failures. Furthermore, the API allows integrating sophisticated machine learning algorithms like reinforcement or deep learning and other kinds of artificial narrow intelligence that allow to further increase the autonomy and performance of the system.

Framework. To meet the above-stated requirements, the MAS was developed using the widespread Java Agent Development Framework (JADE) [34]. It has the advantage of being platform-independent, because it is Java-based and, in addition, provides

FIPA-compatibility and is distributed open source under the LGPL license. Furthermore, database management systems and several libraries were used to achieve the required functionality. Among them are MySQL, Apache Web Server, and phpMyAdmin for database functionality, apache commons library for mathematical functions like exponential and gamma distributions, and Dijkstra algorithm for finding the shortest path in a directed graph. The next section describes the application case on which those tools were applied.

3.3 Application Case: Car Body Weld Shop

The roots of many of the challenges described in Sect. 2.3 are linked to order release and resource allocation. Therefore, a state of the art car body weld shop was selected as an application case for this work, since it is an archetype for those challenges. It can be structured using the hierarchical structure model from [35] which divides production systems into 9 layers reaching from component to production network. The body shop covers levels 1 to 7 of those hierarchy layers. Due to the MES-focus of this work, only layers 5 to 7 are considered, i.e. work unit, production line segment, and production line. From top to bottom, on layer 7 we can find two production lines which produce different car body types, as depicted in Fig. 2. Going deeper to layer 6, these two lines consist of 13 different production line segments S11 to S80. Some of these segments, like S11 and S12, are used exclusively by either production line 1 or 2, others like S40 and S60 are shared by both production lines and build a bottleneck.

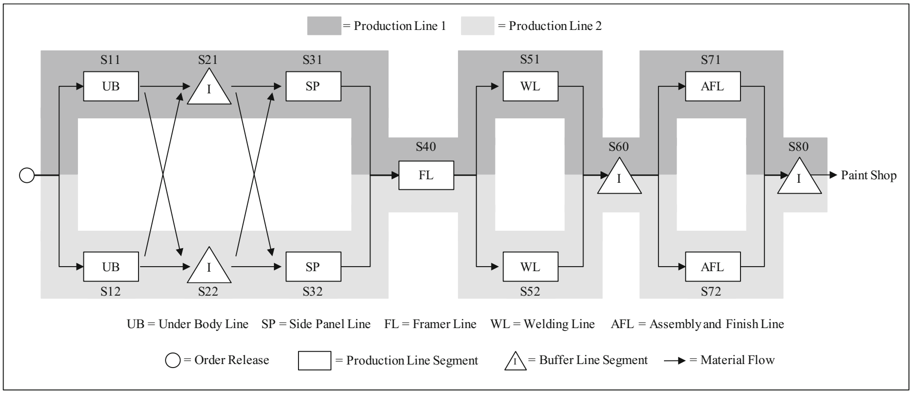

Fig. 2. Layout and technical capabilities of resources in the application case

Finally, on layer 5, each segment consists of multiple separate work units which are not depicted in the layout. They each perform separate production steps required in the production segment.

In total, the car body weld shop is able to manufacture around 70 different product variants, including different body types and market-specific models. However, all variants are based on three main body types: a two-door compact car, a four-door compact car, and a four-door compact van. To better illustrate the effects of the MAS, this paper focuses on those three main types which will be referred to as variants A, B, and C.

Production line 1 is able to produce variants A and B, while production line 2 can manufacture only variants A and C. Buffers can take all variants as shown in Table 1.

Table 1. Resource capabilities considering production of product variants A, B, and C

	Production line 1				Production line 2				Shared resources				
Variant	S11	S31	S51	S71	S12	S32	S52	S72	S21	S22	S40	S60	S80
A	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
B	✓	✓	✓	✓					✓	✓	✓	✓	✓
C					✓	✓	✓	✓	✓	✓	✓	✓	✓

Since not all production line segments can produce all variants, it is up to the agents in the system to decide which order will be produced by which resource to achieve the best possible performance. In the next section, the architecture for these processes is explained in more detail.

3.4 Multi-agent System Architecture

The agent architecture in this work was designed using the "Designing Agent-based Control Systems"-methodology (DACS) which consists of the three steps analysis of control decisions, identification of agents, and selection of interaction protocols [36].

For the first step, the authors build upon the work of [37], where control decisions of manufacturing systems were collected, categorized, and assigned to general control tasks. As part of a student work, the specific decisions in a car body weld shop have been analyzed further and mapped to the necessary agents.

The second step, identification of agents, requires a broader look on agent-based production system architectures. Over the past two decades, a large set of architectures for manufacturing control has been developed. The work of [38] analyzes those architectures and shows that common design patterns have emerged out of them. Following those design patterns, a distributed control system architecture was designed consisting of the seven agent types shown in Table 2.

Order, Product, and Resource Agents are the most elementary agents. While Order Agents represent individual customer orders, Product Agents represent whole product variants and contain information about their constitution. Resource Agents manage the production line segments of the system and the therein contained work units. Although work units could be represented as separate Resource Agents, for the selected scenario, it suffices to represent them internally in the Resource Agents of the production line segments. Furthermore, to control and supervise those resources, a Shop Management Agent is required and the Scheduling Agent is necessary to select the next best order to be released in the dynamic environment. A not so obvious but crucial entity is the Mediator Agent, whose job it is to match supply and demand so that the best overall

Table 2. Agent types and their corresponding tasks and information in the model

Agent type	Tasks	Required information
Order Agent (OA)	• Initialize and supervise all required production steps	• Order information (e.g. order ID, variant, delivery date, priority)
Product Agent (PA)	• Manage manufacturing information for every product variant • Provide information to OAs	• Product variant • Production steps (e.g. welding spot sequence in body shop, assembly sequence in assembly shop, etc.) • Technical and sales restrictions
Resource Agent (RA)	• Process designated orders • Keep track of reservation list • Document order status • Request new orders when not occupied • Inform SMA about status (available/disturbed)	• Resource capabilities (cycle time, number of work units, variants, etc.) • Resource status (time to failure, time to repair) • Reservation list • Order status
Shop Management Agent (SMA)	• Instantiate RAs • Keep track of resource status (available/disturbed) • Provide routing information for material flow to OAs	• Shop layout and material flow graph • Resource status (available/disturbed) via message from RA
Scheduling Agent (SA)	• Manage production program • Instantiate OAs when requested by RA or SMA	• Valid production schedule (order ID, variant, sequence, delivery date) • Output and work in progress • Capabilities of requesting resource • Availability of resources in the shop and other restrictions (e.g. logistical or technical restrictions)
Mediator Agent (MA)	• Collect mediation requests from OAs and get proposals for production from RAs during reservation time frame • Match orders to resources appropriately	• Demand for production steps (number and time of requests of OAs) • Supply for production steps (number and ESOP of RAs) • Rules for prioritization of orders (e.g. delivery time, reservation time, variant, priority)
Directory Facilitator (DF)	• Register and deregister every available service in the system • Provide information to all agents	• Registered services and related information

system productivity can be achieved. However, this can only be achieved through communication and cooperation between agents, which is the third step of the DACS-methodology and the subject of the following section.

3.5 Agent Communication

Out of the control tasks listed in Sect. 3.4, the following two are described in more detail in this paper, since they have the highest impact on system performance. The first one is scheduling and releasing orders. A task especially important during disruptions. The second one is allocating orders to resources, including a reservation mechanism allowing to book resources in advance and thereby control the production sequence.

Scheduling and Order Release. The Scheduling Agent (SA) builds upon the production sequence it receives from the sequencing algorithm. This sequence includes the order-ID and related data like product variant and delivery time and is only changed if necessary. On this basis, order release works as a pull mechanism: Resource Agents (RA) of root resources (first work unit of the first production line segment) request new orders from the SA, when they finish processing the predecessor. The SA then selects the next order and instantiates an Order Agent (OA). The SA is equipped with an algorithm that aims at leveling the order release. To do this, it uses the existing production schedule, applies the real-time information it has from other agents (e.g. resource status and capabilities to produce certain variants), and prioritizes orders in a way that target ratios – defined by production system restrictions – of the respective variants are met. However, conventional production systems do not adapt to them automatically in case of disruption. Therefore, if e.g. product variant B cannot be built, it balances production by releasing the variant with the least negative effect on the target ratio one at a time. This way, the backlog of variant B leads to fewer upheavals down the production stream, i.e. paint shop and especially assembly shop, and after the resource is repaired, it can be reduced more easily.

Resource Allocation. Two things were needed to implement the required resource allocation: an interaction protocol that supports the required negotiation between agents and a reservation mechanism that allows matching orders and resources in advance to minimize waiting time. For the negotiation part, the common contract net protocol [34] was selected and extended to match the application case requirements. As the sequence diagram in Fig. 3 suggests, the Mediator Agent (MA) plays an important role in the negotiation. It coordinates the negotiation process as a broker and is triggered by a mediation request from an OA. To avoid loss of productive time during negotiations and thus compromising performance, a resource reservation mechanism was implemented.

Figure 4 shows that a reservation time window is at its core. This window starts a predefined time period before the end of the current production process (e.g. one second prior to production end), so that (a) the probability of disturbances during the current production process is low, and (b) no productive time is lost due to negotiation processes. After receiving the production request, the MA looks up available resources and makes a call for proposition (CFP). The RAs provide the estimated start of production (ESOP) and wait for a response at the end of the reservation window.

The MA then calculates the best match for all orders and resources using a specifically engineered set of rules. Those include the variants and delivery times of the requesting orders as well as their successors, the currently available resources and variants in the resources that offer the desired operation, as well as the status and available variants on the subsequent production steps. As an example, if a subsequent

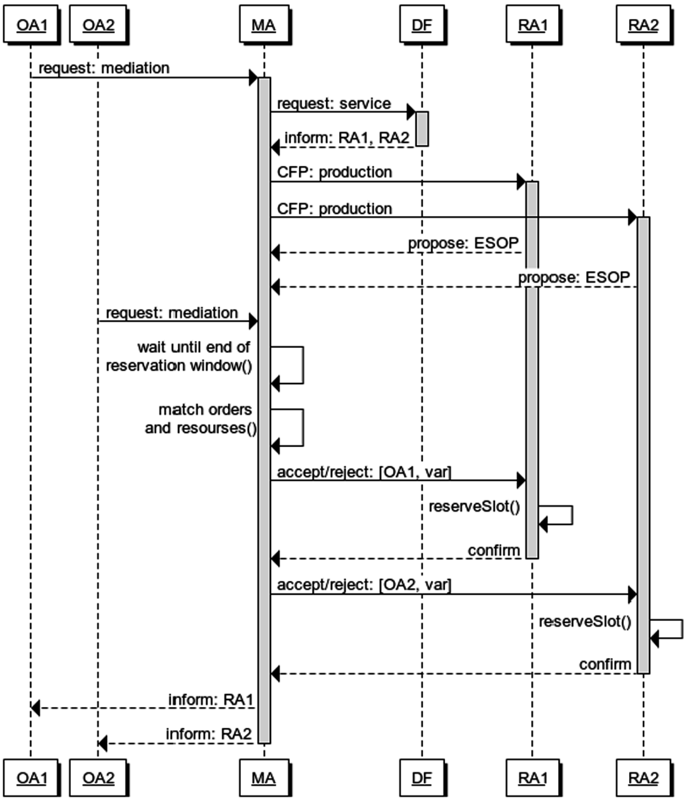

Fig. 3. Agent communication during resource allocation process

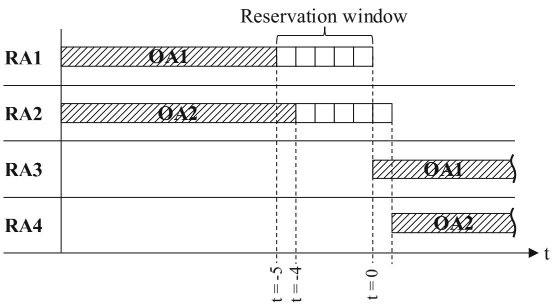

Fig. 4. Reservation mechanism with reservation window

resource offering variant 'B' is down (e.g. S51), resource S40 would not offer variant 'B', since it would block the resource for all other variants. It therefore offers 'A' and 'C' to continue production. Based on this architecture, a series of simulations was performed to examine the performance of the MAS, which is described in the following section.

4 Multi-agent Simulation

The objective of this work is to develop a solution that is tailored to the observed challenges in complex production environments. Against this background, the authors opted for using simulations with real-world data that allow emulating specific conditions instead of using abstracted benchmark-problems. The most important factors are summarized in the following.

4.1 Simulation Basis

Production Program. The starting point of the simulations is a real production program for 24 h, already sequenced by a sequencing algorithm that considers the technical restrictions of the factory. It includes a typical daily amount of about 2.000 orders of the three main body types A, B, and C. Internal delivery dates, i.e. the date when a car body must be delivered to the paint shop as an internal customer, range between 12 and 24 h from scheduled production start for regular orders. However, about 1% of the orders are fast orders with delivery dates between 10 and 12 h.

Resource Parameters. To reproduce the actual situation on the shop floor as realistically as possible, major resource parameters of the MES-layer were integrated into the model. That includes functional parameters like cycle times, variant capabilities, and capacity, as well as maintenance parameters like mean time between failure (MTBF) and mean time to repair (MTTR). Each Resource Agent is provided with the required data via a JSON-file on startup and manages its state by itself.

Resource Disturbances. To simulate resource failures, Resource Agents are provided with a function to calculate the time until the next breakdown and the time it will take to recover. The earlier is based on the MTBF-value of the resource and is approximated via an exponential distribution which is typically applied for lifetime distributions [39]. The latter, on the other hand, is based on the MTTR-value of the resource and is approximated by an Erlang distribution, since it better represents repair processes [40]. With the help of inverse transform sampling, the RA calculates those two times and takes down the resource during the disruption.

4.2 Evaluated KPI's in the Simulation Runs

To measure the performance of the system according to industry standards, the following seven categories were selected: output, resource utilization, delivery date, lead time, production program fulfillment, production sequence fulfillment, and work in progress. For each of these seven areas, specific KPI's were chosen that appropriately measure the system's performance. The list contains common KPI's used in the automobile industry as well as standard descriptive statistic methods like mean value and standard deviation, and are summarized in Table 3.

Since the simulation contains randomly selected events like MTBF and MTTR a simulation run is not replicable. To compensate outliers, the simulation was repeated multiple times for every configuration and results were averaged for each KPI. However, outliers can provide valuable information about the quality of the results and are therefore discussed in the next section.

Table 3. Selected KPI's for evaluation of production system performance

Category	KPI	Description	Unit
1. Output	• System output	• Number of good cars per time unit	jobs/hour (jph), %
	• Output-mix fulfill	• Deviation from planned output-mix	
2. Resources	• Resource-utilization	• Utilization of resources during uptime. Equal to OEE, if there are no scrap parts/orders	%
3. Delivery reliability	• Average delivery date deviation	• Mean value of delivery date deviations of all orders	Days
	• σ delivery date	• Standard deviation of delivery date	Days
4. Lead time	• Average lead time	• Mean value of lead time of all orders	Hours
	• σ Lead time	• Standard Deviation of lead time	Hours
5. Production program	• Production program fulfillment	• Degree to which the original production program is fulfilled with a tolerance window of zero	%
	• Average sequence deviation R000	• Mean value of sequence deviations at order release point (R000)	No. of cars
	• σ Production program	• Standard deviation of actual to target sequence at order release point	No. of cars
6. Production sequence	• Sequence fulfillment	• Degree to which the released order sequence mimics the planned production program with a tolerance window of zero	%
	• Average sequence deviation R800	• Mean value of sequence deviations at end of production (R800)	No. of cars
	• σ Production sequence	• Standard deviation of actual to target production sequence at end of production (R800)	No. of cars
7. Work in progress	• System filling level	• Degree to which the technical capacity of the production system is used by physical orders	%

5 Results and Critical Evaluation

To allow a direct performance comparison between a conventional production control system approach (CPCS) and the agent-based production control system (APCS), multiple simulation runs have been performed. The results of the CPCS were then set as an index, to display the direct performance delta of the APCS in a juxtaposition. The results are presented in Table 4.

As can be seen, the performance of the underlying production system is generally better when it is controlled by the agent-based control system. Good results are especially achieved in KPI's like output and resource-utilization which are interdependent. Higher resource utilization can be achieved through the systems adaptability properties, which allow it to act dynamically upon unpredicted events and therefore improve the

Table 4. Performance comparison of the two production control systems

KPI	Deterioration	Improvement
System Output [jph]		+8%
Output-Mix Fulfillment		+100%
Resource-Utilization		+8%
Avg. Delivery Date Dev. [d]		+1%
σ Delivery Date [d]		+8%
Avg. Lead Time [h]		+9%
σ Lead Time [h]		+21%
Production Program Ful.	-27%	
Avg. Seq. Dev. R000 [cars]	-36%	
σ Production Program [cars]	-32%	
Production Sequence Ful.	-22%	
Avg. Seq. Dev. R800 [cars]	-9%	
σ Production Seq. [cars]	-29%	

overall system yield. This also applies for KPI's in the area of delivery reliability. Since the system disfavors releasing orders that are subjected to current technical or logistical disturbances, the delivery reliability of the released orders increases.

On the other hand, the agent-based control system does not achieve quite as high levels of stability in the areas of production program and production sequence fulfillment. The adaptability of the system comes with the drawback of breaking a predefined production schedule and leads to a more mixed production sequence. This is a logical consequence of the adaptation of the system to influences of its environment and eventually leads to a more balanced workload. Although this could be seen as a challenge for manufacturers that follow the pearl chain concept, many production systems work with lead times that allow adaptation to those changes without compromising efficiency. Furthermore, on an absolute scale these KPI's decline at a single-digit rate and have therefore limited impact. So, by filling up production slots that would otherwise stay unused, production program conformity is sacrificed, but the business-wise more important KPI's considering output, resource utilization, and delivery reliability are increased. Considering the limitations of the proposed MAS, it should be noted that the system performance partly relies on predictions of the duration of disruptions which is an information many present production systems do not provide on their own. Instead, they depend on maintenance personnel or dispatchers to enter the data manually into the system, which usually comes with a delay. However, since the decisions of the MAS are being made in real-time, it can react immediately to the new information and e.g. release new orders or block them. Even in this scenario, the system would perform better than solely based on human decision making as it is common today.

6 Conclusions and Future Works

This paper has presented an approach for manufacturing business to cope with continuously rising complexity in production scheduling and sequencing in turbulent environments. The proposed approach is a hybrid between state of the art sequencing algorithms and an agent-based production control system. It allows high-quality solutions for sequencing problems while at the same time autonomously adapting to unexpected changes in the production system. The developed system architecture is based on common design patterns of agent-based production control systems and uses a Mediator Agent as a broker to allow optimal resource allocation during runtime. In simulations of an application case – a state of the art car body weld shop in the automobile industry – the system achieved significant improvements in important production system performance indicators such as output, resource utilization, and delivery time. It can therefore be assumed, that implementation of this kind of hybrid systems could help to reach substantial productivity improvements in manufacturing businesses in similarly complex environments.

Despite the achieved performance increase, there are still opportunities for further enhancements of the approach. Incremental system-specific improvements could be achieved by advancing e.g. message exchange efficiency, code heaviness, and action timing of agents. More potential, however, lies in the implementation of artificial intelligence like reinforcement learning to improve decision making. Finally, a consequent next step is the gradual implementation of the APCS into a state of the art factory to exploit the described potentials and demonstrate applicability in practice.

References

1. ElMaraghy, H., Schuh, G., ElMaraghy, W., Piller, F., Schönsleben, P., Tseng, M., Bernard, A.: Product variety management. CIRP Ann. Manuf. Technol. **62**(2), 629–652 (2013)
2. Bauernhansl, T.: Die vierte industrielle revolution – der weg in ein wertschaffendes produktionsparadigma. In: Vogel-Heuser, B., Bauernhansl, T., Hompel, M. (eds.) Handbuch Industrie 4.0 Bd.4. SRT, pp. 1–31. Springer, Heidelberg (2017). doi:10.1007/978-3-662-53254-6_1
3. Monger, M., Nicherson, J., Woerner, S.l.: Digital Transformation: The Race to Become Future-Ready. https://emarketing.alixpartners.com/rs/emsimages/2017/pubs/DIG/AP_The_race_to_become_future_ready_Apr_2017.pdf. Accessed 25 Apr 2017
4. Kagermann, H., Wahlster, W., Helbig, J. (eds.): Recommendations for implementing the strategic initiative INDUSTRIE 4.0 – securing the future of German manufacturing industry, Final report of the Industrie 4.0 Working Group, April 2013. http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_Website/Acatech/root/de/Material_fuer_Sonderseiten/Industrie_4.0/Final_report__Industrie_4.0_accessible.pdf. Accessed 01 Feb 2016

5. Arbeitsgruppen der Plattform Industrie 4.0: Umsetzungsstrategie Industrie 4.0 - Ergebnisbericht der Plattform Industrie 4.0, April 2015. https://www.bmwi.de/BMWi/Redaktion/PDF/I/industrie-40-verbaendeplattform-bericht,property=pdf,bereich=bmwi2012,sprache=de,rwb=true.pdf. Accessed 01 Feb 2016
6. Bauernhansl, T., ten Hompel, M., Vogel-Heuser, B. (eds.): Industrie 4.0 in Produktion, Automatisierung und Logistik. Springer, Wiesbaden (2014). doi:10.1007/978-3-658-04682-8
7. Manzei, C., Schleupner, L., Heinze, R.: Industrie 4.0 im internationalen Kontext. VDE Verlag, Offenbach (2016)
8. Sarker, B.R., Pan, H.: Designing a mixed-model, open-station assembly line using mixed-integer programming. J. Oper. Res. Soc. **52**, 545–558 (2001)
9. Boysen, N., Fliedner, M., Scholl, A.: Sequencing mixed-model assembly lines. Survey, classification, and model critique. Eur. J. Oper. Res. **192**, 349–373 (2009)
10. Waldmann, K.-H., Stocker, U.M. (eds.): Operations Research Proceedings. Springer, Heidelberg (2006). doi:10.1007/978-3-540-69995-8
11. Rahimi-Vahed, A., Rabbani, M., Tavakkoli-Moghaddam, R., Jolai, F., Manavizadeh, N.: Mixed-model assembly line sequencing using real options. In: Waldmann, K.-H., Stocker, U.M. (eds.) Operations Research Proceedings, pp. 161–167. Springer, Heidelberg (2006). doi:10.1007/978-3-540-69995-8_27
12. Rabbani, M., Rahimi-Vahed, A., Javadi, B., Tavakkoli-Moghaddam, R.: A new approach for mixed-model assembly line sequencing. In: Waldmann, K.-H., Stocker, U.M. (eds.) Operations Research Proceedings, pp. 169–174. Springer, Heidelberg (2006). doi:10.1007/978-3-540-69995-8_28
13. Mirghorbani, S.M., Rabbani, M., Tavakkoli-Moghaddam, R., Rahimi-Vahed, A.R.: A multi-objective particle swarm for a mixed-model assembly line sequencing. In: Waldmann, K.-H., Stocker, U.M. (eds.) Operations Research Proceedings, pp. 181–186. Springer, Heidelberg (2006). doi:10.1007/978-3-540-69995-8_30
14. Solnon, C., Cung, V.-D., Nguyen, A., Artigues, C.: The car sequencing problem. Overview of state-of-the-art methods and industrial case-study of the ROADEF’2005 challenge problem. Eur. J. Oper. Res. **191**, 912–927 (2008)
15. Brucker, P., Knust, S.: Complex Scheduling. GOR-Publications, 2nd edn. Springer, Heidelberg (2012). doi:10.1007/978-3-642-23929-8
16. Verstraete, P., Valckenaers, P., Van Brussel, H., Saint Germain, B., Hadeli, K., Van Belle, J.: Towards robust and efficient planning execution. Eng. Appl. Artif. Intell. **21**, 304–314 (2008)
17. Wannagat, A., Schütz, D., Vogel-Heuser, B.: Einsatz von Softwareagenten am Beispiel einer kontinuierlichen, hydraulischen Heizpresse. In: Göhner, P. (ed.) Agentensysteme in der Automatisierungstechnik. Xpert.press, pp. 169–185. Springer, Berlin (2013). doi:10.1007/978-3-642-31768-2_10
18. Göhner, P. (ed.): Agentensysteme in der Automatisierungstechnik. Xpert.press, pp. 169–185. Springer, Berlin (2013)
19. Schwede, C., Klingebiel, K., Pauli, T., Wagenitz, A.: Simulationsgestützte Optimierung für die distributions-orientierte Auftragsreihenfolgeplanung in der Automobilindustrie. In: März, L., Krug, W., Rose, O., Weigert, G. (eds.) Simulation und Optimierung in Produktion und Logistik, pp. 151–170. Springer, Heidelberg (2011). doi:10.1007/978-3-642-14536-0_13
20. Pil, F.K., Holweg, M.: Linking product variety to order-fulfilment strategies. Interfaces **34**, 394–403 (2004)
21. Meissner, S.: Logistische Stabilität in der automobilen Variantenfließfertigung. Dissertation, Technische Universität München, Garching b. München (2009)
22. Herlyn, W.J.: PPS im Automobilbau. Produktionsprogrammplanung und -steuerung von Fahrzeugen und Aggregaten, Hanser (Fahrzeugtechnik), Munich (2012)

23. Scheffels, G.: The pearl chain logistics concept. JOT **5**(2), 18–21 (2012)
24. Boysen, N., Fliedner, M., Scholl, A.: Level-Scheduling bei Variantenfließfertigung. Klassifikation, Literaturüberblick und Modellkritik. Journal für Betriebswirtschaft **57**, 37–66 (2007)
25. Boysen, N., Zenker, M.: A decomposition approach for the car resequencing problem with selectivity banks. Comput. Oper. Res. **40**, 98–108 (2013)
26. Downing, N., Feydy, T., Stuckey, P.J.: Explaining flow-based propagation. In: Beldiceanu, N., Jussien, N., Pinson, É. (eds.) CPAIOR 2012. LNCS, vol. 7298, pp. 146–162. Springer, Heidelberg (2012). doi:10.1007/978-3-642-29828-8_10
27. Boysen, N., Scholl, A., Wopperer, N.: Resequencing of mixed-model assembly lines. Survey and research agenda. Eur. J. Oper. Res. **216**, 594–604 (2012)
28. Lüder, A., Schmidt, N., Hell, K., Röpke, H., Zawisza, J.: Description means for information artifacts throughout the life cycle of CPPS. In: Biffl, S., Lüder, A., Gerhard, D. (eds.) Multi-Disciplinary Engineering for Cyber-Physical Production Systems, pp. 169–183. Springer, Cham (2017). doi:10.1007/978-3-319-56345-9_7
29. Wooldridge, M., Jennings, N.R.: Intelligent Agents. LNCS, vol. 890. Springer, Heidelberg (1995)
30. Botti, V., Giret, A.: ANEMONA: A Multi-agent Methodology for Holonic Manufacturing Systems, 1st edn. Springer, London (2008). doi:10.1007/978-1-84800-310-1
31. Unland, R.: Software agent systems. In: Leitão, P., Karnouskos, S. (eds.) Industrial Agents: Emerging Applications of Software Agents in Industry, pp. 3–22. Elsevier, Amsterdam, Oxford, Waltham (2015)
32. Roidl, M.: Kooperation und Autonomie in selbststeuernden Systemen. In: Günthner, W., ten Hompel, M. (eds.) Internet der Dinge in der Intralogistik, pp. 65–78. Springer, Heidelberg (2010). doi:10.1007/978-3-642-04896-8_9
33. Mönch, L.: Agentenbasierte Produktionssteuerung komplexer Produktionssysteme, Deutscher Universitäts-Verlag GWV Fachverlage GmbH (Wirtschaftsinformatik), Wiesbaden (2006)
34. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-agent Systems with JADE. Wiley Series in Agent Technology (Reprinted). Wiley, Chichester (2008)
35. Röpke, H., Lüder, A., Hell, K., Zawisza, J., Schmidt, N.: Identification of “Industrie 4.0” component hierarchy layers. In: 21st IEEE Conference on Emerging Technologies and Factory Automation (ETFA), 6–9 September 2016, pp. 1–8, Berlin. IEEE, Piscataway (2016)
36. Bussmann, S., Jennings, N.R., Wooldridge, M.: Multiagent Systems for Manufacturing Control. A Design Methodology. Springer, Heidelberg (2004). doi:10.1007/978-3-662-08872-2
37. Zawisza, J., Hell, K., Röpke, H., Lüder, A., Schmidt, N.: Generische Strukturierung von Produktionssystemen der Fertigungsindustrie. In: 17. Branchentreff der Mess- und Automatisierungstechnik - Automation 2016 - Secure & Reliable in the Digital World, VDI-Berichte 2284, 16 p., VDI-Verlag GmbH, Düsseldorf (2016)
38. Lüder, A., Calá, A., Zawisza, J., Rosendahl, R.: Design pattern for agent based production system control – a survey. Submitted to 13th IEEE Conference on Automation Science and Engineering August 20–23, Xi’an, China (2017)
39. Kütting, H., Sauer, M.J.: Elementare Stochastik. Mathematische Grundlagen und didaktische Konzepte. Springer, Berlin (2014)
40. Birolini, A.: Reliability Engineering. Theory and Practice, 7th edn. Springer, Heidelberg (2014). doi:10.1007/978-3-642-39535-2