MHP
A PORSCHE COMPANY

MHPWHITE PAPER

ALGORITHMIC PRODUCTION

A Framework for Planning, Design and Implementation of Manufacturing Control System Architectures

Abstract

This white paper proposes a framework for choosing the most appropriate solution architecture to support modern manufacturing facilities as part of an End-2-End-[I]IoT-Transformation. Modern manufacturing is characterized by processes that allow fast time-to-market, sustainability and highly customizable, multiple product variants. Growing requirements to production calls for new architecture solutions. Firstly, we will discuss the different approaches for implementing an architecture of this nature. Secondly, the building blocks encompassed by this architecture. And finally, the business scenarios that will serve this architecture.

Every architecture design suggested by the framework is linked to the new unique core figure, called "algorithmic production". In algorithmic production, schedules of entities in a distributed shop floor are repeatedly optimized. This is based on the predicted availability of machines and operations to be executed according to the work plan as well as on unforeseeable events during execution. To optimize the production plan, the framework selects from a repository the method that best suits the business scenario and objective function.

Chapter 1

Introduction

1.1. IoT Transformation — An End-to-End-Approach

Industrial Internet of Things (IIoT) and digital transformation are multilayered and complex concepts. On the one hand, fields of action and use cases for companies must be specifically defined and, if possible, via small projects or product concepts. On the other hand, embedding all aspects into an end-to-end approach is extremely important.

The end-to-end approach and the integration of the individual subject areas are essential success factors for the [I]IoT transformation. In addition, it is necessary to combine this approach with a consistent strategy, roadmap, conception, as well as the implementation and use of technological solutions.

The use of IIoT and its embedding in the overall architecture are elementary prerequisites for innovative production concepts - as in this example, algorithmic production. This requires smooth cooperation between all the disciplines shown in Figure 1. Only with the right roadmap and a focus on the relevant stations can solutions be implemented profitably - and above all sustainably.

1.2. Algorithmic Production as the Solution for Handling Increasing Industrial Challenges

In order to remain competitive, industrial companies must be able to handle an increasing number of product variants and ever shorter product life cycles. At the same time, even refining consumer requirements drives the demand toward products with a fast time-to-market. The result are products that are sustainable, have high quality and are highly customized, at a low price. In addition, the rapid development of cyber-physical systems and IIoT also promotes autonomy in dynamic shop floor processes. This concept is part of the so-called fourth industrial revolution or "Industry 4.0" [12, 15, 26].

To cope with highly competitive global markets, modern production systems are based on certain requirements and characteristics that go beyond traditional performance indicators:

- Increasing autonomy in decision-making at system and component level.
- Using fully self-aware components, e.g. resources and materials.
- Development of cyber-physical-oriented designs to individualize equipment and products.
- Improved overall system stability with focus on tolerance to faults and system changes.
- Adaptive capabilities for fast and immediate system adjustments as well as for realizing scalability.

These requirements are fundamental for handling new and highly dynamic environments. More self-aware and intelligent resources enable greater autonomy and self-control, which in turn, pushes existing shop floor systems toward innovation. Comprehensive data collection, in combination with agent technology and cloud computing, has led to the development of powerful technologies that could fundamentally change the way we think about manufacturing today.

The term "algorithmic production" encompasses any form of manufacturing control system that enables self-control and autonomy through calculating, optimizing and coordinating schedules of production entities from ubiquitous shop floor processes. Such entities include parts, machines, operations and job orders — or even any type of unforeseen process occurrences or disturbances.

For the algorithmic production, different architecture scenarios can be applied depending on the business use case. Each architecture scenario has its own functional and technical requirements, although they must all serve ISA95 Level 3 to Level 1 functionality in general.

A wide range of different designs for manufacturing control systems architecture have been discussed, including "Reconfigurable Mechatronic System" (RMS) [8, 12, 15, 18], "Multi Agent System" (MAS) [19, 29] or even "Holonic Manufacturing System" (HMS) [1, 6, 23]. This framework of algorithmic production considers different scenarios according to their particular needs in terms of implementing the algorithmic production principle. Here, we consider the architecture design rather than specific implementation structures like HMS. Any of the designs can be applied for implementation of algorithmic production processes.

The paper is organized as follows: In Chapter 2, the term "algorithmic production" will be amplified as a core principle found in any architecture design derived from the proposed framework. The characteristics of algorithmic production will be explained via an architecture design that encompasses several building blocks, each providing a specific functionality. Chapter 3 suggests various reliable architecture scenarios for implementing algorithmic production — from technical as well as functional perspectives. Moreover, Chapter 3 summarizes the related business cases for algorithmic production scenarios and explains the indicators to determine the most appropriate architecture scenario. In Chapter 4, an OEM implementation of the central architecture scenario, including genetic algorithm-based (GA-based) scheduling, will be presented. This chapter includes information about challenges and experiences that has been gained during implementation.

[I]IoT Transformation Tubemap

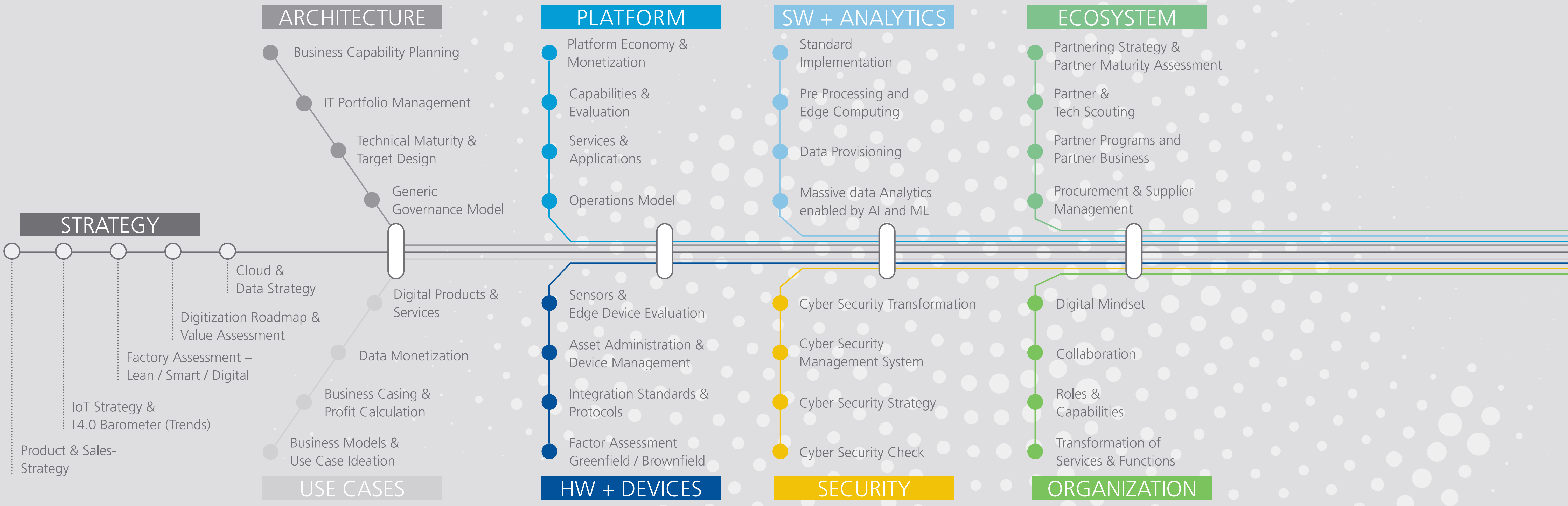

Figure 1: [I]IoT Transformation — MHP End2End Approach to Enable Digitalization

Chapter 2

Principles and Building Blocks of Algorithmic Production

2.1. Overall Principles of Algorithmic Production

In algorithmic production, several parallel running manufacturing schedules of production orders are coordinated and optimized. These actions are based on one or more predefined objective functions. Since unexpected deviations from planned schedules may occur at any time due to disturbances, these must also be resolved during the optimization. In algorithmic production, each possible sequence of activities for the execution of a production order is repeatedly calculated in advance for a short period of time and maximized against predefined objective functions. Availability, lead time and makespan are objective functions for optimization. At the same time, all material-related process sequences of the job conditions must be adhered to. Once implemented, algorithmic production guarantees maximum flexibility and reliability of the processes on the shop floor. Since the algorithmic production controls autonomous instances on different levels through constant schedule optimization, it can also react dynamically to changes. This also applies at machine, route and product level.

Scheduling in manufacturing is understood as the allocation of jobs to resources over a time related to an objective function. Jobs themselves always belong to order that are executed within a specific time period. The execution of the orders occurs either in parallel or in sequence to other orders that make manufacturing scheduling even more complex [2, 5, 11, 14, 32, 33]. Since operative manufacturing requires rescheduling within a short time, schedules must be calculated swiftly to allow for continuous process flows. Accommodated scheduling mechanisms and the chosen objective function must balance to achieve maximal efficiency. The framework offers a different combination of schedule mechanism and objective functions for each of the architecture scenarios.

In industrial practice, things seldom work as expected; different events always occur. For example:

- New tasks arrive continuously in the system and scheduled ones get canceled.

- Certain resources become unavailable and/or additional resources are introduced.
- Unexpected events occur in the system, such as machine failures, operator absence, rush orders or unavailability of (raw) materials.
- Scheduled tasks may take more or less time than expected.

In such dynamic environments, an optimized schedule that has been produced in advance can quickly become outdated. Dynamic rescheduling is required as fast as possible to avoid the risk of disruption to operational processes [3, 4, 5, 14]. This fact makes scheduling a crucial part of all manufacturing control systems. Many scheduling methods [4, 7, 11, 13, 14, 33, 36] have been evaluated based on heuristics, linear programming, constraint satisfaction techniques, neighborhood search techniques (e.g. simulated annealing or taboo search) and genetic algorithms [2, 11, 13]. However, it is necessary to define which method is most appropriate for whatever manufacturing control architecture is applied [26, 27].

2.2. Building Blocks of Algorithmic Production Architecture Design

The architecture design of algorithmic production encompasses four core building blocks that interact with each other. These building blocks sense data from the shop floor and transform it into meaningful events applicable for schedule calculation. The building blocks also provide an initial schedule in advance and optimize it in line with various criteria. Together they establish a closed-loop process with deep learning facilities to improve behavior over time.

Based on the lurking shop floor, data schedules are built for a certain time period t2 in advance, and in turn prepared for optimization. This optimization will be repeated increasingly after each time t1, until all shop floor jobs are accomplished. The time parameter t1 is significantly smaller compared to time t2 in order to catch as many events as possible. Both time parameters are unknown at the beginning but will be

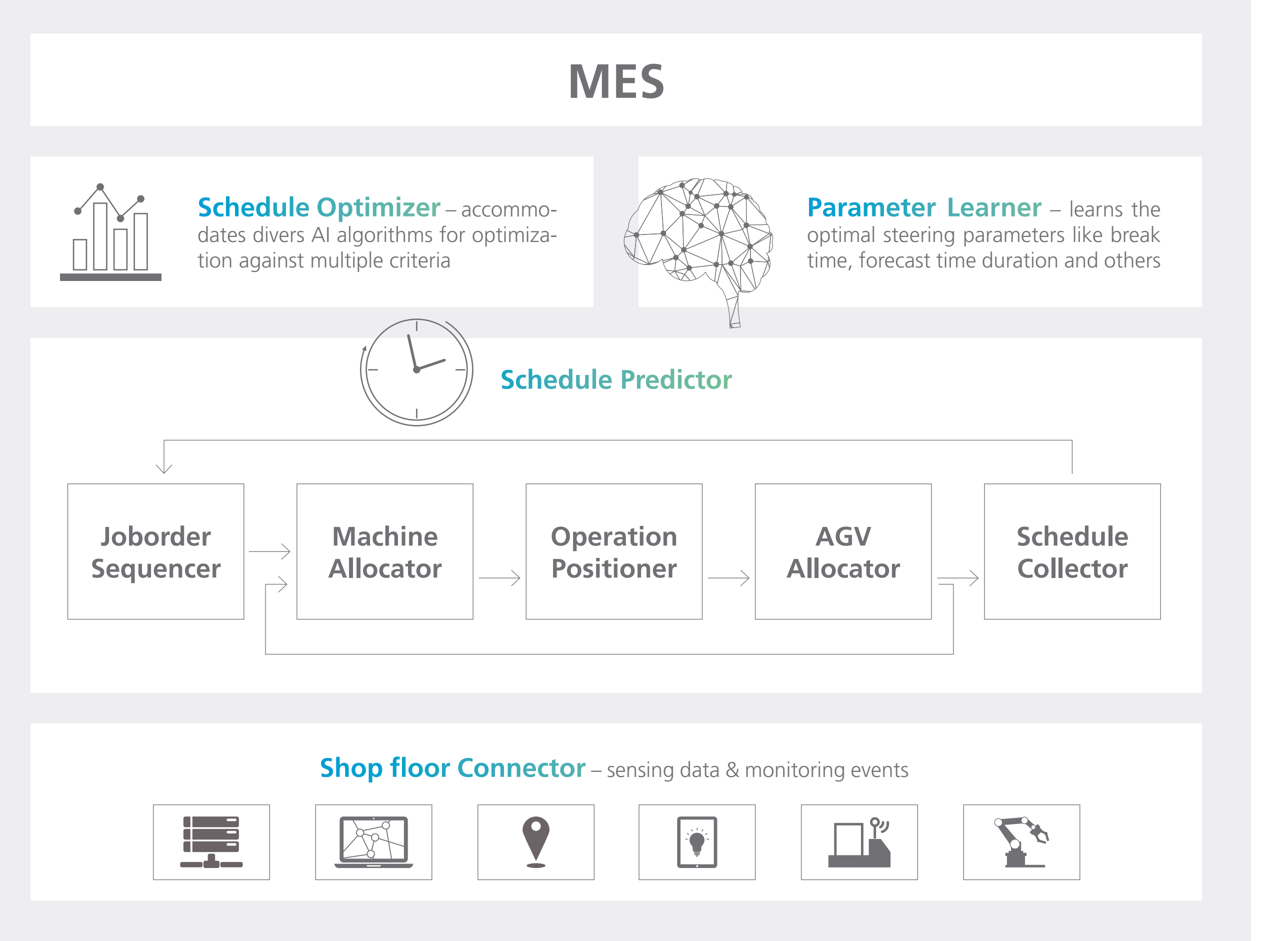

Figure 2: MES architecture extension by algorithmic production features

learned during runtime. This working principle has two consequences: Firstly, we may not be interested in specifying an optimal schedule during the prediction step, as the optimization will be executed afterwards. Secondly, the optimization itself has not the goal to provide a global optimum, but local ones only.

The principal architectural design of a conventional manufacturing execution system (MES) extended by the algorithmic production feature is described in Figure 2.

The following building blocks are provided within the architectural design:

Shop Floor Connector

Senses all schedule-related shop floor data that has been changed for the current time period, e.g. from Automated Guided Vehicles (AGVs), machines and operations together with start and finish times. All gathered data will be feed into the current schedule and, in turn, considered during schedule prediction for the next cycle.

Schedule Predictor

Calculates several schedule variants, each of which is eligible for processing within time t2. It is necessary for these schedules to adhere to the routing rules of their related jobs. Although the principle of earliest completion time will be applied to schedule design, it is not the intention of the "Schedule Predictor" to provide an optimal schedule but only to deliver several schedule variants.

As visualized in Figure 2, the "Schedule Predictor" accommodates five submodules to calculate a batch of single schedules. The schedule calculation starts with the "Joborder Sequencer" to build various sequences of job operations. The idea here is to provide different schedule variations that are gathered and finally consolidated by the "Schedule Collector". The creation of a single schedule includes the following steps: First, during machine allocation, a job operation to be executed is assigned to a machine whose availability is close to the start date of the operation. The "Operation Positioner" then determines the start and completion date of the operation based on the machine cycle time and the transport times to and from this machine. Finally, the "AGV Allocator" manages and controls the movements of the AGVs and their availability in order to optimize costs and maintenance times.

Schedule Optimizer

Provides a single optimal schedule by optimizing the schedule variations from the "Schedule Predictor" for the time period t2. The optimization step itself is restarted after time t1, until all job orders have been processed. Different tools for optimization are discussed in literature, including GA-based scheduling, Markov search trees, swarm models or conventional optimizations such as gradient descent. Any of the algorithm designs mentioned before may achieve satisfying optimization results. Depending on the business scenario, it must be determined which method is best suited regarding to the architectural design.

Parameter Learner

Applies learning mechanisms. We have already mentioned two parameters as relevant for algorithmic production. Firstly, by the time t2 each schedule will be calculated in advance. Secondly, by the time t1 the optimization step will be repeated. Both pre-defined parameters are made more precise over time by using learning mechanisms. The task of the "Parameter Learner" is to determine which value the parameters t1 and t2 should have in order to achieve the maximum optimization result.

Chapter 3

Framework for Algorithmic Production

The framework defines different manufacturing control architecture scenarios for the implementation of the algorithmic production with the corresponding use cases.

In general, manufacturing control architectures can be classified into one of four topological classes [31]. Architectures, which regulate the entire task spectrum for machines and transport routes via a central control system, form class 0. In class I the high processing effort of class 0 architectures is reduced by splitting them into different controlling (sub-)architectures. However, the hierarchical design of the overall architecture is retained when sub-architectures are used. In class II, the architectures are combined, which enable hybrid manufacturing control by combining hierarchical and partially heterarchical control. Finally, Class III architectures of manufacturing control systems offer completely decentralized control, where processing is performed by several independent and autonomous instances. Since there is no significant deviation between classes 0 and I with respect to the underlying process and system structure, they are considered as one single class in the framework of algorithmic production.

For the algorithmic production framework, the following architecture scenarios will be set out in more detail and applied to operational shop floor processing:

- **Centralized manufacturing architecture** — Orders, based on production planning, drive manufacturing: The overall central schedule accommodates all jobs, operations and resources. Disturbances in the form of unavailable resources will be resolved during schedule optimization. Thus, all critical paths throughout the shop floor must be redundant to avoid interruptions.
- **Federated manufacturing architecture** — Mixed approach depending on disturbances. There is a general central schedule only for the "Happy Path". In the case of disturbances, the affected jobs, operations and resources will negotiate new local schedules. Once disturbances are resolved, a new overall schedule is calculated, according to which production continues. This approach provides a high reliability of production in terms of output quantity.
- **Decentralized manufacturing architecture** — Machines lead and the shop floor drives manufacturing: Local schedules apply only at agent level. Communication between agents steers the overall manufacturing process. Agents work on all levels. Jobs, operations and resources all follow their own individual agenda and objectives. Negotiation balances individual interests and enables communication between agents.

	Centralized manufacturing architecture	Federated manufacturing architecture	Decentralized manufacturing architecture
Topology class	Class 0–I	Class II	Class III
Scheduling mechanism	Single overall schedule that applies: Approach by localization (AL) [13], disjunctive graph model (DSG) [3], earliest finish time (EFT) [4], HEFT [4, 30]	Coordination of distributed single schedules [16, 17]	Coordination of autonomous agents [19, 29]
Objective function	Makespan, machine utilization, goods produced	Makespan, machine utilization, tardiness, nervousness	Makespan, machine utilization, tardiness, nervousness
Optimization function	Particle swarm optimization (PSO), genetic algorithm (GA), automated bees colony (ABC), ant colony (ANC) [18, 33]	GA, multi-issue negotiation [9, 10]	Multi-issue negotiation Monte Carlo tree search (MCTS) [20, 25]
Architecture principle	Central control entity for all entities involved	Central control entity, distributed shop floor entities	Distributed, self-aware, autonomous entities
Technology stack	Conventional MES extended to shop floor gateway to steer level 2–0 resources	Holon manufacturing systems, multi-agent systems	IoT, multi-agent system, cloud manufacturing, swarm intelligence
Business principle	Order centric	Event centric	Machine/resource centric

Table 1: Manufacturing architecture scenarios for algorithmic production

3.1. Centralized Manufacturing Architecture

One architecture scenario for an algorithmic production framework is the centralized approach. There is one central instance in the centralized manufacturing architecture. This can be any system or entity that is responsible for one overall schedule for production. Here, we assume that production is driven by orders from one central production planning.

All order-related production processes are defined in the overall central schedule. It describes which tasks or operations need to be accomplished, as well as their allocation to physical resources. The schedule consists of several entities that are hierarchically structured into jobs, operations and resources. Furthermore, the schedule defines the sequence of all job-related processing and transportation tasks.

During processing, the central schedule is regularly updated and optimized. For the purpose of optimization, different objective functions can be applied to different instances, e.g. transport costs or speeds of AGVs as well as availability or processing times of machines. Unexpected disturbances in the form of unavailable resources are eliminated during optimization. Therefore, all critical paths should be redundant throughout the shop floor to avoid interruptions.

One possibility for implementing an optimization component into the schedule is by applying a genetic algorithm (GA). A GA is a search heuristic inspired by the theory of natural evolution. This algorithm reflects the process of natural selection in which the fittest individuals are selected for reproduction. The process of natural selection begins with the selection of the strongest individuals from a starting population [2, 13, 32]. Through crossover and mutation, recombination and optimization are realized. If this principle is applied to central scheduling, a global optimum can be achieved. In the context of algorithmic production, achieving a global optimum is not necessary even for a central production architecture. The principle of algorithmic production is based on the realization of short-sequential partial optimizations. According to the architecture design from chapter 2, the "Schedule Predictor" starts to determine all possible schedule variants for a firmly defined future time horizon (e.g. 10 minutes) based on the available information from "Shop Floor Connector". The calculated schedule variants can be used as an initial population for the genetic algorithm. The "Schedule Optimizer" continues to process the schedules until an optimum has been determined. This process is restarted after a fixed time interval has elapsed, so that successive partial-optimal schedules are generated within a short time.

To enable the central scheduling approach, a few requirements must be fulfilled. First of all, the ability to perform actions when scheduling approaches is necessary. Rescheduling is required in the event of machine breakdowns, rush orders, cancellations due to quality issues, etc. Moreover, the ability to apply optimization techniques is relevant for achieving a continuous improvement in overall system behavior.

3.2. Federated Manufacturing Architecture

The intention of the federated architecture is to combine centralized and decentralized control approaches in order to increase agility, flexibility and stability of the manufacturing control system. Such systems are as centralized as possible and as decentralized as necessary, which allows them to use a centralized approach when the objective is optimization, and a decentralized approach in response to unexpected events.

This architectural design was first handled in connection with so-called holons. According to Koestler's definition, a holon is a part of a system that has its own identity and a private schedule. Holons can be physical resources and logical instances that include both informational and physical parts [1, 6].

In federated architectures, various self-regulating holons are combined into a hierarchically structured overall system based on their private schedules [16,17,18]. In the following, central types of holons are introduced which, together with the formation of a hierarchy, can take over the execution of production orders:

- **Task Holon**: Is responsible for managing real-time execution of production orders on the shop floor.
- **Operational Holon**: Represents the system resources, e.g. machines and robots. An operational holon is responsible for governing its own agenda as well as managing the physical connection with the real resource.
- **Supervisor Holon**: Provides coordination and optimization services to the holons under its supervision, and thus introduces hierarchy in an otherwise decentralized system.

Holons will always constitute a hierarchy. However, any single holon can always leave that hierarchy in order to participate a temporarily built community of other holons, which mostly happen as result of a disturbance. The participation of a holon depends on how well the holon with its private schedule can optimize the local schedule from the hierarchy formed with the other holons. Only if the holon meets the defined optimization criteria, it will be integrated.

The integration takes place via the algorithmic production: On the one hand, the "Schedule Predictor" calculates the possible integration variants of the private holon schedule into the corresponding hierarchy. On the other hand, "Schedule Optimizer" determines whether the predefined optimization criteria have been met or not.

3.3. Decentralized Manufacturing Architecture

Another scenario for the development of algorithmic manufacturing control systems is the approach of decentralized architecture. In this scenario, shop floor objects (e.g. machines, products or workpiece carriers) lead and control production. These interacting and intelligent resources are called agents. Agents can be characterized as self-aware, autonomous and at least partially independent. Together they form so-called multi-agent systems, which aim to solve problems that would be too complex or even impossible for a single agent or a monolithic system [21, 22].

In contrast to the centralized and federated approach, the decentralized approach is based solely on the private schedules of the agents. A characteristic of the agents is their limited view of joined production occurrences. For this reason, continuous communication between the agents takes place in real time in order to get a global view and be able to solve higher-level problems. The global view makes it possible to steer the overall manufacturing processes.

Agents run jobs, operations and resources at all levels. Each agent follows a predefined individual work agenda (private schedule), which is linked to specific objectives and requires activities to be performed with other agents. The activities of all agents must be coordinated among themselves so that manufacturing control can always be carried out effectively and efficiently. Crucial for this is the execution of negotiations between the agents, which optimizes the distribution of tasks and the achievement of objectives by the agents. Negotiation balances individual interests and enables continuous communication. Agents communicate with each other about their objectives and the requested activities. A credit system regulates the negotiation process. Agents offer jobs to each other that can be accepted, declined or also canceled. Agents must immediately declare their decision to prevent several agents from performing the same task.

In the algorithmic production architecture, all agent decisions are collected by the "Shop Floor Connector" and transmitted to the "Schedule Predictor", which then bundles different variants of decision sequences into schedules. The "Schedule Optimizer" will then select the most suitable schedule and announce it to the involved agents. In addition, every negotiation will be dropped that has not been finalized before the schedule prediction process has been restarted. For dropped negotiation negative credits will be given to the agents involved.

Chapter 4

Industry Reference for Implementing Algorithmic Production in Electric Engine Manufacturing

Industry Reference is based on a central manufacturing architecture and has the primary purpose of explaining how algorithmic production can be implemented in operational practice. It was implemented as part of an automotive project to manufacture electric engines in a new production plant.

To introduce the reference, it is important to understand that this automotive project represents several innovative topics:

- The OEM strives for E-Mobility and has developed an electric engine for a fully electric vehicle. The product development process itself was very dynamic and volatile during the project. For example, changes to the product affected the implementation and realization of the required manufacturing processes as well as the technical parameters and configurations in the manufacturing control system.
- The new product was manufactured according to a completely new concept in modular design. The main goal of the OEM was to build a dynamic and flexible production facility that can react autonomously to changing events. For the first time in the industry, AGVs were used not only for logistical purposes, but also to map the value chain on the shop floor. AGVs therefore become a crucial part of the production system.
- The modular production layout and the redundant availability of machines have increased the flexibility in process execution. A control algorithm guided the AGVs through the modular production plant. This made it possible to shorten waiting times by reacting dynamically to unexpected events. Machine failures no longer led to an interruption of the production chain, as alternative machines with identical functions were offered.

At the same time, these innovative topics simultaneously entailed different customer requirements for the production system:

- An intelligent planning system was necessary that could react to changes in the production schedule as needed to maximize plant capacity utilization.
- Another important requirement was the flexible execution of operations in the routing. Due to the availability of several resources with the same or similar functions, the execution sequence of operations and resources should be dynamic. For this reason, operations in the work plan were broken down to the smallest sub-steps in order to then assign them to resources.
- Furthermore, there was the requirement to integrate logistical processes into the manufacturing control system. In this integrative system, it was crucial regarding productivity that the right material was available at the right time for the right resource and the right product variant. As it is the case in most production plants, the objectives of minimum throughput time and maximum output quantity were also set for this production system.

A completely new production architecture was required to fulfill the requirements of modular and flexible production described above. The centralized architecture approach was chosen to achieve the customer's objectives. With the central scheduling approach, a general schedule was created in an integrative manufacturing control system, which includes all orders, operations and resources. This schedule determined the respective sequence of processing activities of human and machine as well as the transport routs of the AGVs. On the one hand, the AGVs were used to transport the electric engines throughout the shop floor and between the machines. On the other hand, components from a supermarket were provided on the AGVs for assembly. Due to the multiple availability of systems, there were several manufacturing paths existing in the production plant. A regular update of the schedule made it possible to reschedule and optimize the production routes in case of disturbances or delays.

During implementation of this centralized production system, the project participants faced several technical and organizational challenges. The main challenges experienced are outlined below:

- **Integration and complexity:** A centralized approach always leads to the need to integrate all systems and technologies at all ISA levels. Seamless integration of all technical and logistical instances combined with a modular production concept increased the complexity of the overall system. This became particularly visible when changes were made in the system.
- **Performance:** Short latency of machine cycles creates a very tough time schedule that all processes must adhere to. The cycle time of each machine had to include all activities required for data acquisition and calculation from the surrounding interface systems.
- **Overall system availability:** Due to the pervasive nature of system availability, each individual system was critical to success. The systems in the architectural landscape were connected and dependent on each other.

A last challenge to be mentioned is the human influence on the new disruptive concept of modular and algorithmic production. First of all, the manufacturing processes for an electric engine differs significantly from that of a combustion engine. In addition, the transition from human control to automation and digitalization was completely new for employees. Intelligent instances that controlled the shop floor processes in a modular production led to a loss of human control and transparency. In addition, the intervention of an operator in the automated process could lead to a reduction in efficiency. Examples are increased waiting times for machines and longer transport times for AGVs.

The project represented a learning journey for all project participants, both in terms of process and system development, and due to the technical and organizational challenges involved in the introduction and operational execution of the centralized production system. For example, to success¬fully set up and run a human-machine environment, employees' awareness of how to operate a fully automated system must be achieved from the very beginning.

Conclusion

In this white paper, we introduced the principle of algorithmic production as a recommended solution to cope with the demands of modern manufacturing facilities regarding flexibility, robustness and availability to enable the production of highly customized goods. Algorithmic production relies on repeated schedule calculation within narrow time cycles for a fixed time in advance. This process gathers all emergent events that may impact the schedule. The schedule chosen for processing will be the one that best meets the predefined optimization criteria. Optimization is normally performed in line with various parameters such as makespan, lead time and machine load balance.

To implement algorithmic production, three different architecture scenarios (centralized, federated and decentralized) have been proposed. Each one is discussed from the perspective of its relevant building blocks. Due to a lack of experimental data, the selection of the architecture scenario must be based on business purpose only. It is assumed that the centralized scenario may be most appropriate for order-driven manufacturing processes. The centralized scenario differs from a decentralized scenario in which the machines control shop floor processes autonomously. The federated architecture scenario can be applied for a mixture of both business purposes. Finally, the industrial implementation of the central architecture approach was presented using a reference example for electric motor production. Technical and organizational challenges were also discussed.

In conclusion, it is clear that the [I]IoT-transformation is a highly complex topic. At the same time, it offers enormous potential for production efficiency if an integrative approach is taken. As the development cycles of software applications and innovative technologies will continue to accelerate, corporate management must also be viewed from a new perspective. An understanding of the technical changes and the changing tasks and roles of employees in the industry is becoming increasingly important.

References

- [1] Barbosa, J., Leitão, P., Adam, E., & Trentesaux, D. (2013). "Structural Self-organized Holonic Multi-Agent Manufacturing Systems" Industrial Applications of Holonic and Multi-Agent Systems (pp: 59–70). Springer.
- [2] Byung Joo Park, Hyung Rim Choi, Hyun Soo Kim (2003). "A Hybrid Genetic Algorithm for the Job Shop Scheduling Problem." Computers & Industrial Engineering, Vol. 4 No. 45 (pp: 597–613).
- [3] Peter Brucker (2006). "Scheduling Algorithms." Fifth Edition. Springer.
- [4] Zhengcai Cao, Lije Zhou, Biao Hu and Chengran Lin. "An Adaptive Scheduling Algorithm for Dynamic Jobs for Dealing with the Flexible Job Shop Scheduling Problem." BISE 3/2019 (pp. 300–309).
- [5] CHAUDHRY, I A. (2009). "Minimizing Flow Time for the Worker Assignment Problem in Identical Parallel Machine Models using GA [J]." International Journal of Advanced Manufacturing Technology 48 (5–8) (pp: 747–760).
- [6] Dias Ferreira, J., Ribeiro, L., Onori, M., & Barata, J. (2013). "Bio-Inspired Self-Organising Methodologies for Production Emergence." IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp: 3835–3841). IEEE.
- [7] Dias-Ferreira, J., Ribeiro, L., Akillioglu, H., Neves, P., Maffei, A., & Onori, M. (2014). "Characterization of an Agile Bio-Inspired Shop Floor." 12th IEEE International Conference on Industrial Informatics (INDIN) (pp: 404–410).
- [8] Farid, A. M. & Ribeiro, L. (2015). "An Axiomatic Design of a Multiagent Reconfigurable Mechatronic System Architecture." IEEE Transactions on Industrial Informatics, 11(5) (pp: 1142–1155).
- [9] Shaheen S. Fatima, Michael Wooldridge, Nicholas R. Jennings (2004). Artificial Intelligence, Vol. 152, 1–45, Elsevier.
- [10] Shaheen Fatima, Sarit Kraus, Michael Wooldridge (2015). "Principles of Automated Negotiation." Cambridge University Press.
- [11] Hong Zhou, Yuncheng Feng, Limin Han (2001). "The Hybrid Heuristic Genetic Algorithm for Job Shop Scheduling." Computers & Industrial Engineering, Vol. 3 No. 40 (pp: 191–200)
- [12] Jazdi, N. (2014). "Cyber Physical Systems in the Context of Industry 4.0." IEEE International Conference on Automation, Quality and Testing, Robotics (pp. 1–4).
- [13] KACEM I (2003). "Genetic Algorithm for the Flexible Job-Shop Scheduling Problem [J]." IEEE International Conference on Systems, Man, and Cybernetics 4 (pp: 3464–3469).
- [14] KACEM I, HAMMADI S, and BORNE P (2002). "Approach by Localization and Multi-Objective Evolutionary Optimization for Flexible Job-Shop Scheduling Problems [J]." IEEE Transactions on Systems, Man, and Cybernetics, Part C, 32(1) (pp: 408–419).
- [15] Kagermann, H., Helbig, J., Hellinger, A., & Wahlster, W. (2013). "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. Forschungsunion
- [16] Leitão P, Restivo F. (2006). "ADACOR: A Holonic Architecture for Agile and Adaptive Manufacturing Control." Comput Ind 57(2) (pp: 121–30).
- [17] Leitão P, Restivo F. (2005). "Experimental Validation of ADACOR Holonic Control System. System." Marı́k V, Brennan R, Pechoucek M, editors (2005). "Holonic and Multi-Agent Systems for Manufacturing." LNAI 3593. Springer (pp: 121–132).
- [18] Leitão, P., Barbosa, J., & Trentesaux, D. (2012). Bio-Inspired Multiagent Systems for Reconfigurable Manufacturing Systems." Engineering Applications of Artificial Intelligence, 25(5) (pp: 934–944).
- [19] Lepuschitz, W., Zoitl, A., Valleé, M., & Merdan, M. (2011). "Toward Self-Reconfiguration of Manufacturing Systems Using Automation Agents." IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 41(1) (pp: 52–69).
- [20] M. Lubosch, M. Kunath and H. Winkler (2018). "Industrial Scheduling with Monte Carlo Tree Search and Machine Learning." Procedia CIPR, Vol. 72 (pp: 1283–1287).
- [21] Maturana, F., Shen, W., & Norrie, D. H. (1999). Metamorph: "An Adaptive Agent-Based Architecture for Intelligent Manufacturing." International Journal of Production Research, 37(10) (pp: 2159–2173).
- [22] MacDougall, W. (2014). "Industrie 4.0 Smart Manufacturing for the Future." http://www.gtai.de/GTAI/Content/EN/Invest/_SharedDocs/Downloads/GTAI/Brochures/Industries/industrie4.0 -smart-manufacturing-for-the-future-en.pdf.
- [23] McFarlane, D. C. & Bussmann, S. (2003). "Holonic Manufacturing Control: Rationales, Developments and Open Issues." S. M. Deen (Ed.). "Agent-Based Manufacturing." (pp: 303–326). Berlin, Heidelberg: Springer.
- [24] Bilel Marzouki, Olfa Belkahla Driss, Khaled Ghedira (2017). "Decentralized Tabu Searches in Multi Agent System for Distributed and Flexible Job Shop Scheduling Problem." IEEE/ACS 14th International Conference on Computer Systems and Applications.
- [25] Sebastian Mayer, Nikolas Höhme, Dennis Gankin and Christian Endisch (2019). "Adaptive Production Control in a Modular Assembly System – Towards an Agent-Based Approach." IEEE Conference on Mechantronics (ICM).
- [26] Monostori, L. (2014). "Cyber-Physical Production Systems: Roots, Expectations and R&D Challenges." Procedia CIRP, 17 (pp: 9–13).
- [27] Pach, C., Berger, T., Bonte, T., & Trentesaux, D. (2014). "ORCA-FMS: A Dynamic Architecture for the Optimized and Reactive Control of Flexible Manufacturing Scheduling." Computers in Industry, 65(4) (pp: 706–720). [28] Ren, L., Zhang, L., Wang, L., Tao, F. & Chai, X. (2014). "Cloud Manufacturing: Key Characteristics and Applications." International Journal of Computer Integrated Manufacturing (pp: 1–15).
- [29] Ribeiro, L., Rocha, A., Veiga, A. & Barata, J. (2015). "Collaborative Routing of Products Using a Self-Organizing Mechatronic Agent Framework — A Simulation Study." Computers in Industry, 68 (pp: 27–39).
- [30] Topcuoglu H., Hariri S., Wu M. (2002). "Performance-Effective and Low- Complexity Task Scheduling for Heterogeneous Systems." IEEE Trans Parallel Distrib Syst 13(3) (pp: 260–274)
- [31] Trentesaux, D. (2009). "Distributed Control of Production Systems." Engineering Applications of Artificial Intelligence, 22(7) (pp: 971–978).
- [32] Xuan Guangnan and Cheng Runwei (2000). "Genetic Algorithm and Engineering Design [M]." Beijing: Sciences Press.
- [33] Zhang Guohui, and Shi Yang (2011). "Improved Genetic Algorithm for the Flexible Job-shop Scheduling Problem [J]." Mechanical Science and Technology 30(11) (pp: 1890–1894).
- [34] Zhang, L., Luo, Y., Tao, F., Li, B. H., Ren, L., Zhang, X., et al. (2014). "Cloud Manufacturing: A New Manufacturing Paradigm." Enterprise Information Systems, 8(2) (pp: 167–187).
- [35] Yu-Kwong Kwok and Ishfaq Ahmad (1997). "Dynamic Critical-Path Scheduling: An Effective Technique for Allocating Task Graphs to Multiprocessors." IEEE Transactions on Distributed Computing.

Publisher

MHP Management- und IT-Beratung GmbH

Welcome to the Future. MHP is a leading international management and IT consultancy. We develop pioneering mobility and manufacturing solutions for multinational corporations, mid-sized companies and disruptive startups. As a premium business and technology partner, we are shaping tomorrow's digital future, today.

Our consulting approach is unique, combining holistic IT and tech knowledge with deep expertise in management. This makes MHP the ideal partner for a successful digital transformation. As digitalization experts, we deliver innovative strategies on the basis of strong analysis. These turn your change processes into sustained success.

Over 3,000 employees are driving digital progress in 16 locations worldwide, for over 300 clients. We display excellence at every level. **MHP: DRIVEN BY EXCELLENCE**

Contact persons

Author
Daban Ahmad
Digital Supply
Chain Solutions

Author
Dr. Stefan Gerber
Digital Supply
Chain Solutions

Author
Andreas Fries
Digital Supply
Chain Solutions

Sponsor
Jens Fath
Strategic Lead [I]IOT
Transformation

Photo credits
Page 1 ©istock yoh4nn | Page 11 ©shutterstock Rawpixel.com
Page 18 ©istock ipopba

Layout
Freiland Design

MHP: DRIVEN BY EXCELLENCE

16 **MHP**Offices in Germany, United Kingdom, USA, China, and Romania.

International

Atlanta (USA)
Birmingham (United Kingdom)
Cluj-Napoca (Romania)
Timișoara (Romania)
Shanghai (China)

Germany

Ludwigsburg
(Headquarters)
Berlin
Essen
Frankfurt a. M.
Ingolstadt
Munich
Nuremberg
Wolfsburg