Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-024-02537-6

Optimizing operations of flexible assembly systems: demonstration of a digital twin concept with optimized planning and control, sensors and visualization

Thomas Sobottka[1,2] · Christoph Halbwidl[1] · Alexander Gaal[1] · Matthias Nausch[1] · Benedikt Fuchs[1] · Philipp Hold[1,2] · Leonhard Czarnetzki[1]

Received: 11 August 2023 / Accepted: 23 November 2024

Abstract

This paper presents the development of an optimized planning and control method for flexible manufacturing and assembly systems. While the significant potential of flexible manufacturing concepts to help producers adapt to market developments is recognized, the complexity of the flexible systems and the need to optimally plan and control them is a major obstacle in their practical implementation. Thus, this paper aims to develop a comprehensive digital planning method, based on a digital twin and to demonstrate the feasibility of the approach for practical application scenarios. The approach consists of four modules: (1) a simulation-based optimization module that applies reinforcement learning and genetic algorithms to optimize the module configuration and job routing in cellular reconfigurable manufacturing systems; (2) a synchronization module that links the physical and virtual systems via sensors and event handling; (3) a sensor module that enables a continuous status update for the digital twin; and (4) a visualization module that communicates the optimized plans and control measures to the shop floor staff. The demonstrator implementation and evaluation are implemented in a learning factory. The results include solutions for the method components and demonstrate their successful interaction in a digital twin, while also pointing towards the current technology readiness and future work required to transfer this demonstrator implementation to a full-scale industrial implementation.

Keywords Flexible manufacturing systems · Digital twin · IoT sensors · Reinforcement learning · AI · Simulation

Introduction, terminology, and concept

Production and assembly systems are predominantly either organized in fixed linear process chains, or in a more flexible workshop-style organization. The first is geared towards productivity and works best in high volume production, the latter towards flexibility, resembling the more small-scale manufaction approach of pre-industrial production (ElMaraghy, 2010; Wiendahl et al., 2007). Market trends however increasingly require both capabilities simultaneously: More customized products, ever shorter product life-cycles and environmental-sustainability induced limits for global economy-of-scale in production and associated logistics require a high degree of flexibility and changeability found in workshop production but unavailable in production and assembly lines. At the same time globalized cost pressure and scarce resources require the productivity line structures provide, but workshop production are lacking (Tolio et al., 2010). Two main categories of applications demanding more efficient flexible production organization are:

a) Factories currently working in line-structures that come under pressure from rising numbers of variants and increasing variance between products, due to required customization, specialization and quick development of products on international markets – this reduces the productivity of lines due to frequent set-up and waiting

✉ Thomas Sobottka
Thomas.sobottka@fraunhofer.at

1 Fraunhofer Austria Research GmbH, Vienna, Austria

2 Institute of Management Science, Department of Industrial and Systems Engineering, Vienna University of Technology, Vienna, Austria

Published online: 04 December 2024

times in the line equipment that would run most efficiently for high-volume with little-product variance.

b) Manufacturers operating workshop production with high flexibility but also slow throughput times, high inventory and are thus under (cost) pressure to increase the flowrate of their manufacturing process.

There is an evolution of manufacturing and assembly concepts trying to accommodate these seemingly contradicting requirements, starting from the baseline of inflexible but productive dedicated manufacturing system (DMS) (Bortolini et al., 2018). To enable more flexibility, the flexible manufacturing system (FMS) features optional processes and multiple variants of a product within the line structure. Even more flexibility is introduced with the cellular manufacturing system (CMS), where the strict line structure is relaxed, and processing capabilities are grouped in modules that production orders can utilize. In reconfigurable manufacturing systems (RMS), the structure can be adjusted according to business needs (e.g. product order mix), and the cellular reconfigurable manufacturing system (CRMS) combines all four flexibility elements. Each additional degree of flexibility potentially compromises the productivity of the production system – a comprehensive optimization of planning and control is thus crucial for CRMS to provide benefits in industry applications (Perwitz et al., 2022). Assembly systems can be considered a special form of manufacturing systems leading to a corresponding term of CRAS; since CRMS are the more common term and assembly is a part of production, we will use this term in the following when referring to assembly systems. Another term for CRMS that can be found in recent publications is matrix production (Foith-Förster & Bauernhansl, 2021) – in this paper we will continue using the term CRMS, acknowledging that the system developed herein could also be considered an example of matrix production.

Having introduced the concept of CRMS and their importance, the focus will now turn to how those can be implemented, so that their benefits can be utilized in industrial production applications, and how this paper plans to contribute to that: Digitalization is a major enabler for optimizing production systems, due to its ability to automate the optimization of planning and control as well as keeping the data behind the planning updated, detailed and reliable. The Digital Twin (DT) concept is one of the most advanced forms of this kind of digitalization. A DT is defined as a digital image of a production system(Brenner & Hummel, 2017) with a close coupling between the physical and virtual/digital world, enabled by a near-real-time data link between the status and control of physical objects of the production system and the planning and control level implemented in a software environment (Uhlemann et al., 2017). Thereby the digital model – a simulation – is constantly updated by sensor information and other current data sources from the real-word environment. The planning and control features a forecasting and optimized planning functionality (Rosen et al., 2015), thus closing a continuous loop: status data (e.g. productions order progress, machine availability) from the production system is feed to the simulation, thus ensuring an updated virtual image, which is then used by the optimization as an evaluation function to determine optimized planning and control decisions; in a last step, these measures are forwarded to the production system for execution, followed by a restart of the continuous loop. DTs are also a key component of cyber-physical production systems (CPPS) (Michael Schluse and Juergen Rossmann; Modoni et al., 2019).

The closed loop and the preferably real-time synchronization of planning/control with the production system complements the requirements of the CRMS particularly well, as more flexibility and modularity require more frequent and reactive planning updates. However, it also adds to the planning challenge, as conventional (computation-heavy) optimization methods may need to be supplemented with novel approaches to achieve the necessary reaction time. Both for CRMS and DTs, the still insufficient availability of practically applicable planning and control methods and reference demonstrators constitutes a major obstacle to their industry uptake (Dávid Gyulai et al. 2014; Uhlemann et al., 2017).

This paper aims at developing a dynamic planning and control method in the form of a DT, as well as a demonstrator implementation for CRMS. The demonstrator is situated in the pilot factory facility of the TU Wien in Vienna, Austria, offering a controlled environment necessary for the low technology readiness level while already providing realistic physical surroundings – i.e., production halls, machines and production equipment as well as it-infrastructure – thus ensuring practical relevance of the evaluation results.

Since there is a wide variety of variations of CRMS, we define the requirements for the system developed in this paper:

- The assembly system enables flexibility both in its structure (e.g. the position, combination and configuration of production equipment) and its order processing (e.g. flexible routing, flexible process sequence).
- The assembly structure is based on modules (i.e. workstations, tools, jigs and operators – the non-consumable production potential factors), which can be moved, (re-) combined and which provide certain processing capabilities for assembly tasks.
- Assembly jobs are work orders for a certain product. Each product is linked to a set of required assembly

operations. Unlike in traditional routing sheets, the order of operations is not predefined, but is flexible within limits of technological requirements. The routing of each job through the system can be determined dynamically according to the availability of processing resources, their current workload, costs or other considerations. This introduces considerable additional degrees of freedom and optimization variables compared to traditional production planning and control (PPC).

In order to achieve a DT based dynamic planning and control, four major required enablers can be identified, which constitute the four main elements that this paper aims to adapt and integrate to obtain a working system level CRMS optimization:

1. An optimization that can simultaneously optimize the variables that constitute the flexibility of CRMS, namely the re-configuration of the assembly structure and the dynamic routing of orders through the changing structure and with alternative capacities to choose from. As CRMS are meant to react swiftly and adapt to changing production (mix) requirements, the optimization must support this optimization speed, despite the greater flexibility. Here the paper seeks to investigate potential solutions based on artificial Intelligence (AI) to complement "slower" traditional optimization methods.
2. A DT setup that utilizes the optimization and enables a synchronization of virtual and physical side of the DT, to react to changes and constantly adapt the CRMS to the current production requirements. The paper seeks to build on theoretical DT concepts and develop a working DT method for CRMS.
3. A sensor system that collects status information from the physical assembly to update the virtual side of the DT, to enable the optimization to constantly respond to changes in the physical assembly system. The paper seeks to combine assembly planning methods with current sensor technology to create a suitable sensor module for the DT.
4. A visualization that transfers the results from the dynamically optimized planning to the shop floor and the operators working there, thus completing the DT information-loop. The paper seeks to propose a visualization concept that adequately informs different stakeholders working in CRMS.

The major contribution this paper seeks to achieve are not particularly advanced solutions in any of the four major elements but to achieve a system level functionality by determining the respective state-of-the-art, developing necessary incremental steps to achieve a working solution and combining the four enablers. Special emphasis lies on demonstrating the system with its four elements, the functionality that can be achieved and identifying major future development needs for a large scale industrial CRMS DT-optimization application.

The paper is structured as follows: In the following second section, the simulation, optimization, and control module at the core of the developed approach is presented. This is followed by a description in section three of how the planning and control, i.e. the virtual side of the DT, are synchronized with the physical assembly system. Section four explains how the data, processed by the synchronization module, is gathered in the physical assembly system via sensors. Closing the DT loop, section five explains how the optimized plans and control measures are visualized for management and shop-floor staff respectively, thus ensuring the optimized assembly system operations are executed in cooperation with the staff. The closing section six discusses the combined results of the developed approach and its demonstrator application.

The approach requires multi-disciplinary work to cover all four modules and the associated domain-specific content. For the sake of clarity, each of the four elements is presented individually in the same format in Sects. "Optimized planning and control"–"Visualization", owing to the different nature of the four associated sub-disciplines. This uniform format of subsections of Sects. "Optimized planning and control"–"Visualization" comprises: a detailed problem definition (derived from the desired system level functionality), followed by an analysis of related work, the method development, and the evaluation results. Section "Discussion and outlook" then provides the system level analysis of the combined results as a whole.

Optimized planning and control

Problem description and approach

The planning task for an CRMS can be categorized into the following two subtasks:

- Module (Re-)Configuration: The combination of modules (e.g., tools and equipment, such as screwdrivers, wrenches, and manual presses) and workstations must be optimized to fit the upcoming assembly orders and their technological requirements (e.g., processes and tools requirements). This configuration procedure should be repeated after a certain period or when the product mix has changed in order to continually adapt the assembly system layout to changes in assembly requirements.

- Dynamic, responsive routing and workstation assignment of jobs, i.e. selecting the next operation and station. This is constrained by the station configuration, the remaining to be completed operations for the job and the precedence rules, defining the possible process sequences for each product/assembly job.

Both configuration and routing/workplace assignment problems are NP-hard (Dou et al., 2016), thus requiring either a simplified modelling or the use of metaheuristics or other non-exact optimization methods. Since reaction time in the context of a DT is important, "classical" optimization methods such as metaheuristics, that usually require thousands of evaluations for intermediate solutions, are a limiting factor for the practical implementation and use of optimizing planning methods in complex planning tasks such as modular flexible assemblies. Thus, the presented approach will evaluate the application potential of AI, especially reinforcement-learning (RL), for these planning tasks. Reinforcement-learning can provide near-instant decisions, once the algorithm has been sufficiently trained beforehand.

The underlying system behavior is complex, since not only a "classical PPC" task is considered, but also the structure of the assembly system is dynamic and changeable. Thus, a dynamic simulation of the flexible assembly system was chosen to realistically consider the system behavior. As defined in the introduction, the simulation is integrated in a DT setting, featuring a simulation-based optimization. This means that the simulation is constantly updated by status data from the physical assembly system though sensor data (detailed in Sect. "Sensors and hardware"). The simulation is utilized by the optimization either as an evaluation function for intermediate solutions in the case of metaheuristics, or as a training environment to develop an optimal decision-making strategy for a RL agent.

Related work

Literature reviews for planning methods in the field of CRMS classify planning methods for structure and reconfiguration, and production planning and scheduling (Bortolini et al., 2018; Brahimi et al., 2019). The most common methods are mathematical programming, dynamic programming, metaheuristics and heuristics. Within metaheuristics, the most common methods are the genetic algorithm (GA), simulated annealing (SA) & Tabu Search (a form of local search).

Concerning the optimized planning of assembly system structure and reconfigurability, (Ashraf & Hasan, 2018) use a Non-dominated Sorting Genetic Algorithm-II (NSGA-II) to optimize reconfigurations for successive production stages. The approach is evaluated in an abstracted application setting. The authors of(Kumar et al., 2019) use a Multi-Objective Self-Organizing Migrating Algorithm (MOSOMA), a stochastic evolutionary algorithm, for optimizing the assembly sequence and minimizing reconfiguration effort, considering order due dates and balance of assembly work.

Concerning the routing and scheduling, there are multiple approaches to be found: In (Dou et al., 2016), an NSGA-II is used for the integrated configuration and scheduling of RMS. The results show the basic feasibility of the approach, although performance and calculation time is an issue. The test-case is evaluated with a simplified example featuring 5 machines and 5 parts. In (Gyulai & Monostori, 2017), constraint programming and GA are compared to solve assembly-operator allocation and order scheduling using a simulation-based optimization with metaheuristics. In (Petroodi et al., 2019), optimizing the product sequence in RMS (a reduced planning complexity, compared with CRMS) is implemented with simulated annealing in a simulation-based optimization setting.

For complex, real-world planning tasks, optimization efficiency and achieving good solutions in the available timeframe are a challenge, especially if complex simulation models are used as the evaluation function for the optimization (Sobottka et al., 2018). This becomes even more of a challenge if the changes and thus the re-planning must be frequent and responsive, as in DT planning scenarios, and still more challenging if there are additional degrees of freedom and action variables, such as in the CRMS concept. AI methods have the potential to address this limitation: (a) they can be used to provide a model-behavior approximation for the optimization by learning from a simulation (Sobottka et al., 2019), or (b) they can be used as the optimization method itself – here, the simulation provides a cost efficient training environment and the trained agent can provide immediate decisions, without the need for a lengthy optimization algorithm. For the latter approach, the most promising methods, aside from simple decision rules, are AI-based, especially learning based methods. The most prominent learning-based AI-method is RL, usually used for multi-agent models with decentralized optimization: In (Kuhnle et al., 2019), RL is used for order dispatching in dynamic manufacturing environments. RL in multi-agent decentralized market-based production control is presented in (Csáji et al., 2006). Due to the agent-based decision mode in RL, the "traditional" centralized optimization approach in PPC, i.e. all orders are scheduled, sequenced and assigned to workplaces and operators for the entire planning horizon, is substituted for a decentralized approach. The decentralized approach means, RL-agents make decisions over time, based on the system states at the time of the decisions.

Generally, this field of optimization methods for PPC applications is still in the initial stages of development, compared to the traditional methods, where quasi-standards for the optimization settings and algorithms exist. Consequently, there is no reference method available for the optimization approach at hand. The approach developed in this publication will be a hybrid method, based on both traditional optimization via metaheuristics, using simulation as an evaluation function, and AI in the form of RL agent-based optimization for the most dynamic parts of optimization task.

Developed solution

The system configuration as well as the planning and control process in the developed method are part of a DT featuring a complete simulation model of the assembly system. This simulation is used to evaluate the system's performance as well as a training environment for RL. The simulation itself was implemented in Python, in a custom-made simulation environment based on the SimPy Python library. The simulator features a standardized Excel-data import, which defines the production system, i.e. the resources and their capacity, the orders including routing information, as well as tabular representations of routing precedence graphs – as long as no complex spatial material transport has to be simulated, the modelling is done entirely with the input data, with delay functions for transport and handling times. If transport behavior needs to be modelled in detail, additional modules for the simulator must be used – however, in the paper at hand, this was not necessary, as the complex assembly process was of interest, which is the case in most PPC applications. Thus, the transport behavior was simplified as product- and process-step-specific transition times during the simulation. This lean approach to simulation modelling enables short simulation runtimes, which are critical for the required computational efficiency within the optimization algorithms.

The objective function comprises two main KPIs:

1. The mean job flow rate (i.e., the ratio of productive processing time to the total lead time per job). The flow rate is a standard KPI in the lean production concept to measure the system productivity.
2. Variable module costs per time unit of usage. This measure was chosen specifically for the CRMS setting, where multiple potential workstations are available for most assembly steps, compared to a determined workstation-sequence in conventional linear manufacturing. The unit- or workstation-specific costs can be caused by different levels of tool sophistication (e.g., level of automation) on the stations, scarcity of certain tools and thus processing capability (in that case, scarce tools should be prioritized for orders that benefit most from the tool specific advantages) or other priorities and qualities of workstations and their tools.

These KPIs are weighted and added together to form a total simulation score. The simulation's outcome and consequently its score are influenced by three main input variables:

1. The complete module configuration (i.e., the set of all module-to-station assignments, combining processing capabilities with assembly workstations).
2. The routing logic of individual jobs controlled by dedicated, autonomously acting agent functions, which dynamically chooses the next workstation an order travels to during the assembly.
3. The set of planned assembly jobs with their respective product types – while the first two variables are available as control parameters for the optimization, this third variable is considered a fixed input in the case-study setting, since the order mix and delivery times are in practice defined by the customers.

While the set of jobs is considered a fixed scenario-input, the first two factors are continually optimized. This is achieved by a hierarchical algorithm consisting of three optimization stages:

1. The first optimization stage deals with the training of a job agent function. Every job represents an intelligent entity capable of autonomously selecting its target assembly cell and upcoming assembly operation. This behavior is achieved using the Deep Q-Learning method which facilitates a Deep Neural Network (NN) to approximate the agent function. Initially and upon any fundamental system change (e.g., available modules and abilities, number of assembly stations, the product types in demand), the Neural Network must be re-trained. The training algorithm (see Fig. 1) is episodic: One episode represents a single simulation run for a given job set and module configuration, after which the job agents' actions are evaluated and rewarded accordingly. These steps are repeated for a given number of episodes. The state representation for the underlying Markov decision process uses the remaining process times for each machine queue as features. As the RL training requires existing module-assignments (i.e. workstation configurations) a number of random module-assignment configurations should be created for training purposes to avoid overfitting the routing strategy to certain module assignment configurations.

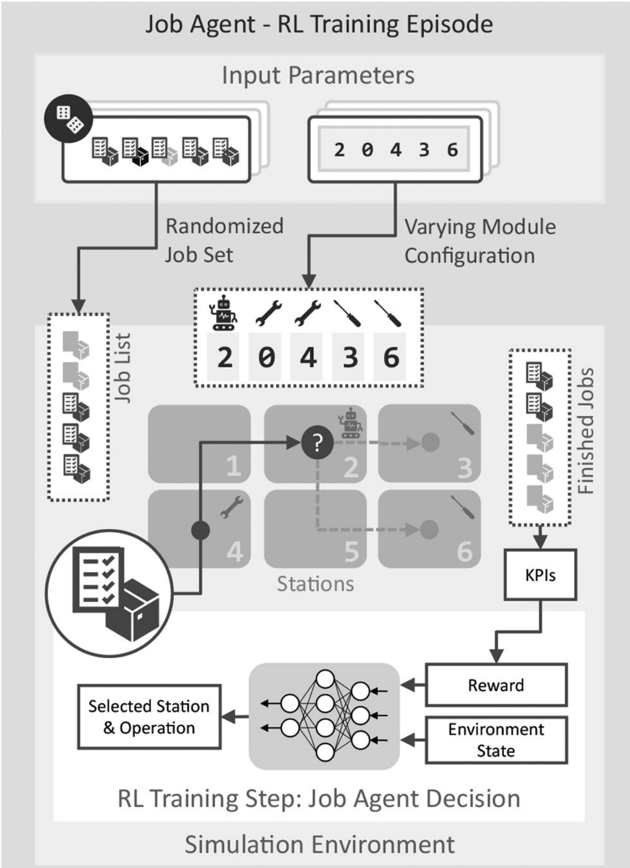

Fig. 1 RL training, optimization stage 1

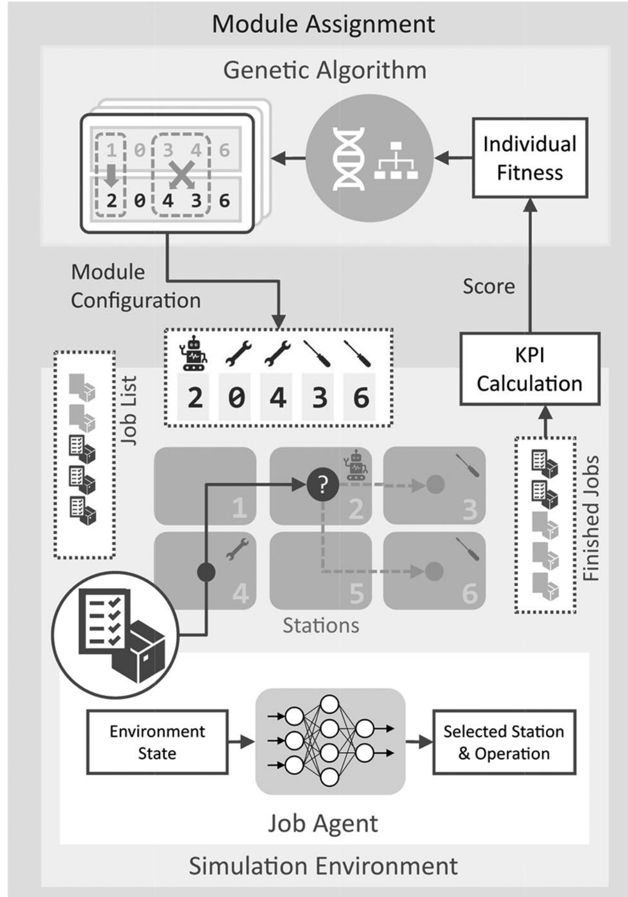

Fig. 2 Module configuration, optimization stage 2

2. After the training of the job agent functions, a GA optimizes the complete module assignment (see Fig. 2) – i.e., assigning assembly capabilities (tools) to the available workstations. We evaluated conducting this module assignment with RL also, but in all tested scenarios, a GA outperformed the RL implementation (see (Halbwidl et al., 2021) for more details). Since this optimization step is not as dynamic in nature as the routing (re-assigning modules is only sensible when the product mix and thus processing requirements change, plus re-assignment takes time and resources and thus cannot be conducted too often), potential efficiency benefits from a pre-trained RL over a GA are not as relevant, thus a GA was chosen for this optimization task – this is supported by findings from our previous publications. This optimization step is repeated after a given interval to adapt the assembly layout for the specific job set scheduled for the respective period. Hereby, the GA iteratively generates sets of different configuration solutions. For each solution and iteration, the simulation is run using the respective configuration, the period's job set, and the job agent function trained in stage 1) as input parameters. After completing a simulation run, the simulation score is used as the fitness value for the respective solution. The result of this stage is an optimized module-to-station assignment configuration.
3. Finally, in the operational method application for planning, the module assignment optimization is run using the pre-trained routing agents while additionally considering the real-time sensorial feedback from the physical system. This means that the optimization is conducted as a rolling horizon planning and executed (a) triggered by the event processing module of the DT (explained in the following Sect. "Synchronizing the DT with the production") when planned events do no longer match the real-word system state as reported by the sensors (see Sect. "Sensors and hardware" for the sensor based event detection), or (b) periodically to be able to initiate a re-assigning of modules for the upcoming jobs if the current assignment no longer fits the product mix. The resulting job schedule with selected assembly stations is then used as the real-world production schedule and sent to the shopfloor to be executed. In order not having to control all job routing centrally, the pre-trained routing agent logic is transferred to and executed on a computer

terminal at each workstation which determines the next destination for an order after having finished operations on the current workstation. This pre-trained decentrally executed routing logic (i.e. agents) enables the quick optimization response for routing in the flexible assembly system.

Results

In all tested scenarios, the GA module assignment optimization was able to achieve good results within seconds, since the problem complexity for a given product mix is limited – the GA can be stopped after a convergence criterion is reached. The population size was set to small values (10 in this scenario, chosen via a grid search), as prioritizing more generations over large population sizes was found to be more efficient use of a given number of simulation evaluations (2000 in this scenario). In this demonstration setting with 6 workstations, ca. 100 jobs (per scenario) and a planning horizon of 2 workdays, the GA-based reconfiguration was not at its performance limits, judging from the fitness-value trends.

Since the RL based routing optimization is more complex and a less established and researched method, this results section is focused on this part of the optimization. To demonstrate the RL job routing, we consider two example production scenarios and compare the performance of the RL routing schedule to different benchmarks. These scenarios have been simplified to fit the current state of RL optimization and support the development of this emerging method in this planning domain. Figure 3 shows the performance of the Deep Q-Learning agent during training and exploitation for a basic scheduling scenario. The performance is compared with a random machine selection strategy and two rule-based selection strategies. The rule-based approaches optimize for the utilization of machines and for usage costs of the machines, respectively.

All scores are reported as a mean value of five different runs (a mean of ten runs for the random selection strategy). In the first scenario, 40 jobs of a single product variant with a single operation are considered. We demonstrate that the Q-learning agent has learned to balance both objectives, in order to achieve a better overall simulation evaluated score than both rule-based and the random selection approach. The second scenario features 5 jobs of a single product variant with two operations. As shown in Fig. 4, the strategy found by the Q-learning agent approximates the optimal selection strategy, which is given by the selection rule that optimizes for usage cost.

However, for more complex scenarios with multiple operations per variant and multiple variants, due to the increase

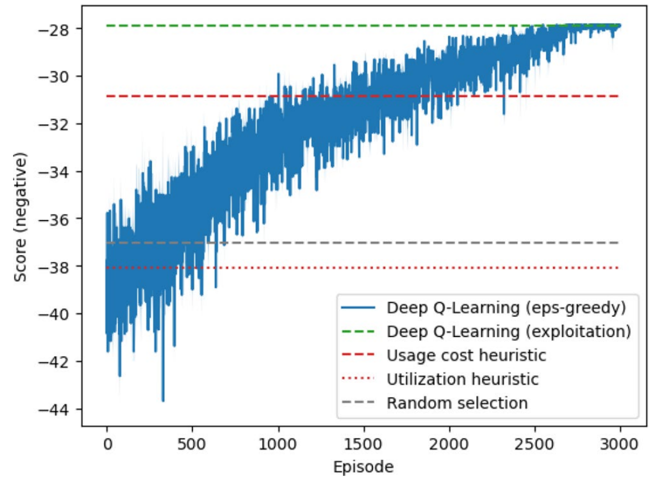

Fig. 3 Performance for the first scenario

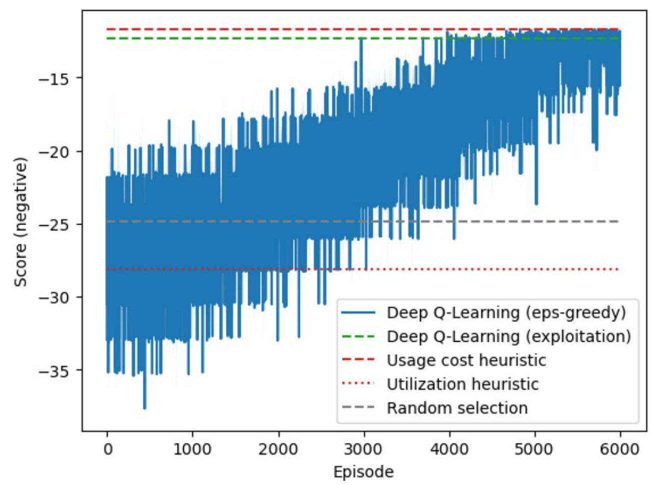

Fig. 4 Performance for the second scenario

in possible actions and therefore exponential increase in scheduling combinations per episode, the algorithm did not converge consistently anymore. Attempts to solve this problem by solely increasing the learning time failed. The increase in complexity can also be seen by comparing the first and the second scenario. Even for learning a "supposedly" easier strategy for a smaller number of jobs compared to the first scenario, convergence took twice the number of episodes in the second scenario, due to the two-stage assembly process. We expect that this problem can be solved to some extent by a combination of increasing the neural network size and the amount of learning time together with careful optimization of hyperparameters. Another approach worth trying might be to use policy gradient methods instead of Q-Learning. For some problems it can be easier to learn a policy instead of an action-value function (Sutton et al., 2020). All experiments were conducted on a single machine with a i7-6700 K CPU and 60GB RAM. The neural network for approximating the Q-function consisted of two hidden layers (48 neurons) with the hyperbolic tangent as the activation function, using Adam as an optimization algorithm for the weight updates. An additional target network and a replay memory (with a size of 100000) was used to increase the learning stability of the Q-learning algorithm (Mnih et al., 2015). After every episode, a weighted parameter update of the target network was performed, where the target and main network parameters were weighted 90 and 10%, respectively. During training an ϵ-greedy strategy was used, where ϵ was decayed from 0.9 to 0 during the first 90% of the episodes.

Although the results thus far do not support the optimization of the full-complexity case, the basic feasibility of applying RL for the more dynamic aspects of the planning tasks could be shown. In the current demonstrator, a fallback optimization for the full complexity case is a GA based approach, which has been successfully employed by the authors for related PPC optimization tasks (Sobottka et al., 2020). This method requires simulation runs as evaluations for the objective function, derived from the function used for the RL agent training. The downside is the longer optimization time since, unlike in the RL based optimization, the computationally intensive simulation evaluations must be conducted each time a new optimization run is required, e.g. also for re-planning, when small changes in the real-world system require a changed planning (e.g., when processes take longer, or when equipment is temporarily unavailable). Thus, in the case of quick re-planning events, the second fallback option is the machine usage cost heuristic used in the benchmark for the RL method above.

Synchronizing the DT with the production

Problem description and approach

Building on the definition from section one, a DT consists of a digital model (simulation), which is continuously updated with status data from the physical production system, a form of optimization, that uses the digital model to evaluate planning and control decisions (without the need for prior physical testing), and finally a channel to feed back the optimized decisions to the physical system to be executed. The required continuous synchronization between the real production, its digital counterpart and the optimization requires a hardware interface between sensors and other data-creating hardware and the hardware on which the simulation and optimized planning are run, plus a corresponding software interface. Internet of Things (IoT), with its principle of embedding sensors and data processing capability with physical objects, supports and accelerates the development of DTs to a large-scale system. However, challenges concerning time synchronization between the virtual part and the physical part, problems with data transparency and Quality of Service guarantee, etc. remain (Siddhartha Kumar Khaitan and James D. McCalley, 2015). This paper considers the IoT possibilities and proposes to use a middleware for the synchronization and to route the data stream from the physical objects to their digital representation and back. Detecting events and constantly monitoring/comparing timestamps of planned versus real-world events, and if necessary, initiating a re-planning, are the major requirements for this to be developed synchronization method.

Related work

Digital modelling and simulation technologies have proliferated thanks to the ubiquitous connectivity of devices and the amount of data shared directly between them or via the cloud. DT-related methodology and technology are applied in various industrial fields and show great potential. Industrial applications of DTs are mainly focused on those related to product design, manufacturing, prognostics, human-machine interaction and management, where DTs have already proven benefits over traditional solutions (Fei Tao et al. 2019; Alam & El Saddik, 2017). The DT concept has been introduced in manufacturing to make manufacturing systems more reliable and flexible. A conceptual DT framework was described in (Leng et al., 2019) for monitoring and optimizing physical manufacturing workshops based on production data, while recent conceptual work also comprises structural standardization and possible automatic deployment of DTs (Göppert et al., 2023). Another approach that enables communication and coordination between the operator with the production system using DT was proposed in (Iris Graessler and A. Poehler 2017) and a DT framework for reconfigurable manufacturing systems by (Kombaya Touckia et al., 2022). From the abovementioned DT frameworks, almost all are not directly geared towards handling the structural flexibility of CRMS, with only the last framework directly addressing this aspect. This last reference provides a detailed structural and procedural model of a form of CRMS, but the presented approach in MATLAB does not provide specifics concerning event handling – i.e. how are events in the physical and virtual side of the DT synchronized and how are optimizations and re-planning initiated; the use-case implementation in a factory lab appears to be on a conceptional stage. The paper at hand will use the basic concept found here and must develop a specific application-oriented solution for event handling/synchronization, as well as a system architecture that is not confined to a certain simulation behavior, so that it can be applied and adjusted to specific real-life system requirements that cannot be anticipated in advance. This structural flexibility is crucial for real-world application perspectives of the desired DT for CRMS.

Each component of the DT, which could each be standalone heterogeneous systems, must be linked via information exchange to enable interoperable systems. Communication middleware technology is used to create a communication environment for information exchange, where systems generally connect through an application programming interface (API) or web services. All participating systems have independent mechanisms for communication and the middleware takes over the mediation of the information between the systems. One of the many advantages is that the system can be upgraded without major changes to the existing integration. Various systems for communication middleware are used in the CPS domain, such as DDS, HLA/RTI, MQTT, or RT-CORBA, while each technology is subject to different system characteristics. In real-world applications, the state of the system's physical assets changes in real time, which requires the (near) real-time capability of the communication or connection of the physical and digital systems. Thus, the communication middleware should provide a real-time communication environment. High scalability and reliability are key requirements and thus standardized real-time communication enabling protocols and services such as MQTT-SN or webRTC should be used in the DT setting (Seongjin Yun et al. 2017; Fernando, 2020. Fontes et al.). MQTT will be used in this paper as the state-of-the-art data-exchange protocol behind the to be developed even handling logic.

Developed solution

The developed solution contains the physical and virtual production system, which are coupled by a process module named "Event Handling" – this module compares events in the physical production system with planned and forecast events computed by the optimization (presented in Sect. "Optimized planning and control"), which in turn uses the simulation as an evaluation function. An overview of the synchronization approach can be found in Fig. "Visualization". The physical assembly system consists of all resources on the shop floor, such as operators, workstations, equipment, materials, products, etc., the sensor system (Sect. "Sensors and hardware") is the data connection to the virtual side, while the virtual production system is a simulation and optimization (detailed in Sect. "Optimized planning and control"), and the visualization (detailed in Sect. "Visualization") is the data-interface from the virtual side (back) to the physical assembly system. For the system states to be synchronized on either side, real events in the physical assembly are transferred to the virtual system, and updated optimized planning decisions including their visualization are transferred back to the production floor. Within the event handling module, one part is responsible for event detection, one for the processing and another for publication of event information. The first part, the event detection (ED), detects events from generated sensor data and the current state of data logging in IT systems, while the second part, the event processing (EP), distributes the information to the requesting parties i.e., the simulation or dashboard.

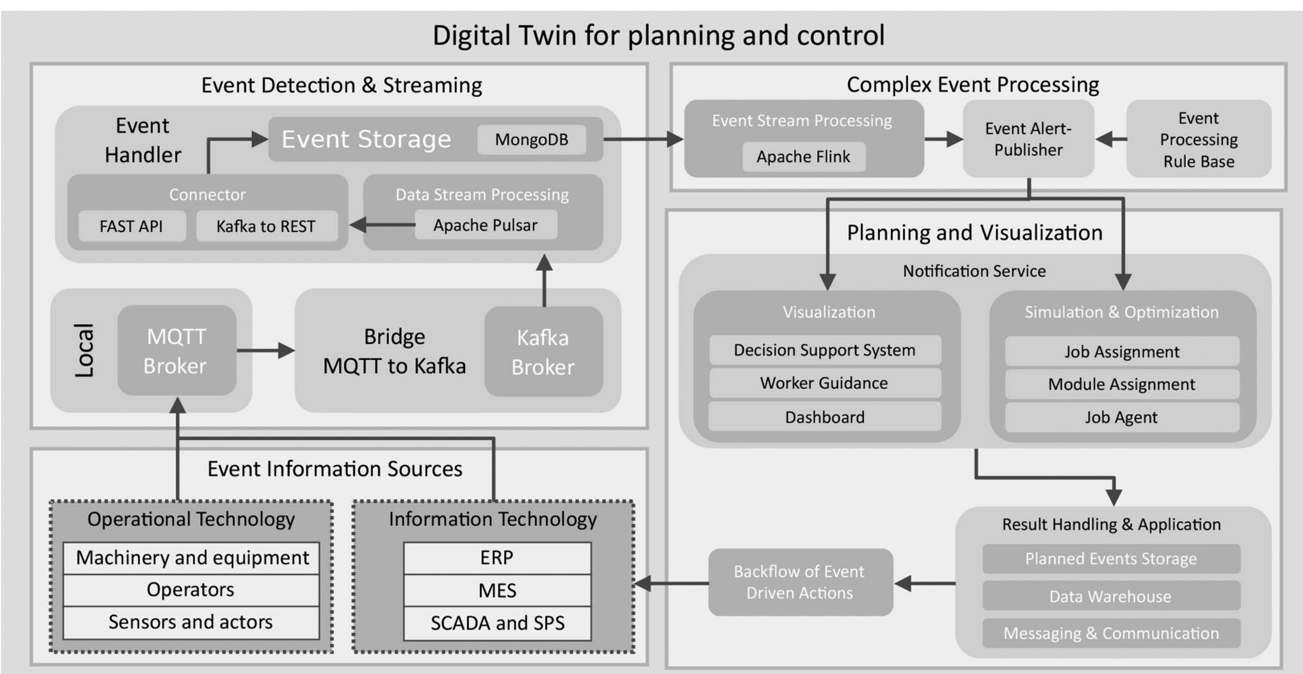

Fig. 5 Coupling of physical and virtual production system

In a physical production environment, data is collected from different sources, such as sensors, shop floor data collection systems, MES, ERP system, etc. Following a publisher-subscriber pattern, data is distributed within a network using suitable protocols – MQTT is used for the DT at hand. The publisher subscriber approach is combined with a central broker entity as an information distributor. The objective is the near real-time transmission of data to data-sinks that need them and are thus registered as information subscribers.

The task of the ED is to analyze the sensor data feeds and recognize process steps as events in the data. Several instances of ED receive the necessary data to make situation-dependent decisions. Sensor based event detection comprises all actions/events and states of the production system that can be determined by sensor data interpretation and cannot be derived from direct data, e.g., from the ERP system log data. For example, tool wear can be detected by interpreting vibration data during tool use – this indirect detection of events and status information enables a more detailed status reporting, e.g. in this case information on the deterioration of a tool rather than only the binary information once the tool is unusable and the machine blocked. Thus, sensor-based event and status reporting enables not only reacting to events but also anticipating them and acting foresightedly. In addition to that, sensor-based information can be used in a multiple sources approach, e.g. together with ERP logs to make the event and status detection more resilient against technical errors (e.g. sensor malfunction) or human error (e.g. unintended/erroneous ERP bookings). This lays the foundation for condition-based planning and optimization.

ED relies on measurements triggering activation functions, to be able to recognize events. According to defined patterns, e.g., reaching a threshold value in a motion detector or reaching it several times, ED creates an indexed entry in a central event database. This contains a unique ID as well as a timestamp and all information about the event itself. The interpretation of sensor data is described in detail in section four.

The virtual production system consists of the modules for visualization and optimization (production planning). In the optimization module, individual processes (e.g., dynamically assigning orders to workstations) and the structure (e.g., assignment of tools to workstations) of the production system are planned with the help of a simulation, as an evaluation function (described in section two), and transferred to the corresponding module for visualization. This involves job sequence and machine assignment planning, determining which operations are to be carried out on which machine, by which employee and at what time. The optimization output is a plan, containing expected events for the production system, e.g., timestamps for the execution of the operations.

The coupling between virtual and physical production system is conducted by the EP module. The EP monitors all the events that have taken place in the physical production system. This information comes from the central event database, fed by sensor data via the ED. The elapsed events are then compared with the expected events in the optimized plan. If there is a significant deviation of planned and actual event, the EP initiates a re-planning by the optimization module, based on the changed circumstances. Evaluating whether deviances in time necessitate a re-planning are implemented as rule-based heuristics in this current demonstrator implementation (e.g., if a threshold value is reached for a time difference between planned and actual event, a re-planning is initiated – this must be balanced so that only practically relevant deviations lead to a re-planning). In future implementations, this simple rule-based decision could be assigned to an AI-agent that evaluates additional status parameters to enable a more flexible, situation dependent decision-making. In addition to time-differences of events, the EP also considers other disturbances, e.g. machine breakdowns or other unavailability of capacities in the assembly system. These too can necessitate a re-planning (e.g. since orders requiring the unavailable capacity should not be temporarily released), which the EP recognizes as necessary and initiates.

A visualization module (see section five) displays both events that have taken place and events that are planned. As soon as an expected event occurs, the ID of the event is sent from the database to both the visualization and the optimization, so that they are always aware of the current state of production. Both modules obtain the required information about the event from the event database to ensure that only the information that is needed reaches the module. This reduces overhead and ensures faster data transfer and lower data volumes overall.

Results

The evaluation was conducted on the i7-6700 K CPU introduced in Sect. "Optimized planning and control". While the potential reaction time of the DT to real-world changes (i.e. synchronization speed) would be milliseconds in this demonstrator, as PLC data is directly accessed and available for the EP, this would not create real-world advantages as most reaction speeds towards deviations from planned to actual events are likely to be on the order of seconds to minutes. Thus, the demonstrator EP has been takted to 2 s – i.e. it compares planned to actual events every 2 s –, which is shorter than the set reaction thresholds for a re-planning, which can be set by the planners and have been set to 60 s in the evaluation scenarios. There is no "correct" value here: there is a tradeoff between reacting swiftly to changes in the physical production and overreaction leading to constantly changed planning that can lead to non-acceptance of the system by the human operators in the production system. Good practice for setting these values in real-world applications will likely only reliably be found through extensive testing, they will be case-dependent and a question of planners' preferences.

In addition to determining the proper reaction time for re-planning, the implementation efficiency of the DT and the programming language both influence the potential reaction times of the DT as well as the computing requirements. In this research the question of efficient implementation and programming language was not part of the intended goals – the implementation in Python was chosen for efficient development and was able to support the required reaction speed on the conventional PC. The DT demonstrator works as intended, with the MQTT publisher-subscriber approach shown to be a feasible approach for event updates between virtual and physical twin.

Sensors and hardware

Problem description

To create a valid database for a constantly updated virtual image of the production system, events from production must be recorded and made available to the simulation. For this purpose, the ED introduced in section three, identifies events in the physical production environment and instantly transmits them to the virtual side of the DT. The standard way to automatically capture events in predominantly manual assembly processes, that cannot be obtained by existing system logs or status data available in the equipment control layer, i.e. ERP/MES, is by means of (additional) sensors or cameras and AI-based machine vision. This section aims to, based on existing sensor concepts, develop an efficient systematic sensor and 10T concept for flexible assembly systems that can provide current process status data in the DT setting presented in the preceding section.

Related work

In recent years, Internet of Things (IoT) sensors have become widely available on the market. Features of these IoT sensors include a variety of environmental parameters that can be measured with a sensor, a wireless data transfer and wireless power supply integrated in one device. The integration and connection of these sensors into the IT-environment in production systems is therefore simple and enables a plug-and-play application – an example for available technology is documented in (Robert Bosch GmbH, 2022), which is a representative for similar IoT-sensor-technology on the market. Furthermore, wireless power and data transfer enables local integration of sensors in manual assembly areas (e.g., on parts, machines and materials) without limiting the flexibility of the assembly system. IoT sensors have already found their way into some areas of daily life, such as the smart home sector and production systems with a high degree of automation (Badarinath and Prabhu, 2017). The use in the field of manual assembly recognition is complicated by the complexity and diversity of assembly tasks, the multitude of new technologies and the lack of expert knowledge in the field of sensor-based assembly process recognition. Some approaches from the relevant literature deal with closing the gap between process and technology experts. Thramboulidis presents a cyber-physical microservice and IoT-based framework for manufacturing assembly systems to create a common vocabulary for assembly system experts and IoT experts (Thramboulidis et al., 2019).

The selection and integration of IoT sensors in specific applications, such as predictive maintenance, is usually based on critical machine parameters and is therefore decoupled from human interaction, thus requiring knowledge of relevant process parameters (Kanawaday & Sane, 2017). In manual assembly applications, the relationship between sensor metrics and assembly activity has not been studied extensively. This relationship poses a challenge for monitoring manual assembly processes. Two approaches can be identified in the area of sensor-based process recording:

- The first is a human-centered approach: Sensors are used to detect people and their actions so that positions, movements and gestures can be recognized, and operator's activities can be deduced. This can be done using body-worn sensors or camera systems with human movement detection software that provides information about the assembly task and the duration of the task. Ward et al. developed activity recognition of assembly tasks using body-worn microphones and accelerometers (Ward et al., 2006). Tao et al. present an approach for activity detection in assembly systems using a wrist-band and a deep learning algorithm for classification (Tao et al., 2018). Liu et al. presents a method to recognize hand movement using a camera system and an object detection software. This information is used to identify assembly steps (Liu et al., 2019). KACZMAREK uses camera sensors and generates depth data to detect gestures and assembly progress (Kaczmarek et al., 2015). In (Chen et al., 2020) machine learning is used to classify image data of assembly activities, enabling process detection. Critical aspects of these approaches are the privacy protection of employees, including differing local regulations, and the required effort in preprocessing and classifying training data.

- The second is an object-oriented approach: instead of capturing people directly, this approach focuses on capturing the product, tools or materials. Kärcher et al. use IoT sensors to detect ongoing assembly processes. Here, sensors are attached directly to tools and materials. Based on the sensor signal, the activity is identified (Kärcher et al., 2018). In this approach, there is no systematic method of how to select suitable points in the assembly processes to integrate sensors and how to choose sensor types. In several research projects, camera systems are used to detect objects using machine vision (Pierleoni et al., 2020). These approaches use image data of the current product to infer the assembly progress and the assembly quality. Rebmann et al. uses a Business Process Model and Notation process description, followed by using sensors and a classifier to identify assembly processes (Rebmann et al., 2020). Hu et al. presents an IoT-based cyber-physical framework for turbine assembly systems, where RFID sensors are used to determine the location of operators and materials and thus to identify the assembly process step (Hu et al., 2020). The selection of appropriate sensors as well as the definition of measuring points in the assembly process are the main challenges associated with this approach.

Assembly tasks	Elements involved		
Movement, visual inspection	Human		
Pick and place of material	Material	Human	Bin
Pick and place of tools	Tool	Human	Bin
Pick and use of resources	Resources	Human	Bin
Place material	Material	Human	Workspace
Use of material	Material	Human	
Use of tools	Tool	Human	
Use of resources	Resources	Human	

Fig. 6 Extended MTM representation, published by the authors in (Nausch et al., 2021)

Developed solution

In the approach at hand, we have chosen the object-oriented approach and thus developed a method to analyze sensor data from environmental sensors. The reason for this is on the one hand to minimize the integration and implementation effort for companies and on the other hand to ensure the protection of employees' privacy. The object-oriented approach has the advantage that no sensors must be attached directly to the person and at the same time no personal data of the employee is recorded. To record as many sensor measurement variables as possible, environmental sensors are ideal, since they can be very small and record several measurement variables at the same time. The sensors can be used without knowing in advance the exact measurand suitable to derive specific processes from the sensor signal curve. The iterative process and therefore the effort to integrate and test different sensors individually is significantly reduced with environmental sensors, compared to the human-centered approach. Currently, no analysis has been done on how suitable process steps can be identified for automated tracking with environmental sensors. Therefore, we developed a method to identify points or segments within assembly processes (e.g., specific movements within an assembly process step), that are suitable for a continuous process recording. The method also specifies how to identify suitable sensor measurement variables and a way to interpret the sensor data to automatically detect the executions of processes.

The first step of the developed method is to identify suitable movements within an assembly process sequence that can be recognized by sensors. For a structured approach, we use the Methods-Time Measurement (MTM) approach, a predetermined motion time system that provides standardized descriptions of movements for manual industrial processes combined with elements involved in each task (De Almeida & Ferreira, 2009). An extended MTM representation has been compiled by the authors (Nausch et al., 2021), were MTM process categories are matched with potential sensor carriers (Fig. 6).

Since the object-oriented approach is used, the human involved in the assembly process is not available as a sensor carrier, thus a "Movement, visual inspection" process would have to be detected by other means, e.g. with a confirmation button for the operator to push, once the step has been executed.

The result of an MTM analysis of an assembly process is a tabular process description. Within this description, recurring or otherwise characteristic elements in the description can be identified, which can then be identified as sensor signal patterns, e.g. from motion sensors, thus constituting suitable measuring points. Sensors can be classified and chosen depending on the use case and according to four dimensions (Jürgen Fleischer et al. 2018): Measurable quantities, energy source, external influence and data transmission. Each of them has several characteristics which are represented in the following Fig. 7.

From the general classification of sensors, this leads to the special sensors that are required for the application case. There are a couple of sensors which are suitable to identify different tasks.

An overview of these various types of sensors is listed in Table 1.

To determine if a certain environmental sensor is suitable for detecting the characteristic process of an object, a sensor must be attached to the object (see Fig. 8). Depending on the object, several manipulations (e.g.: a screwing process) are carried out and the sensor signal pattern of the individual executions is analyzed. If the sensor data from the individual runs do not show any correlation, either the sensor sensitivity or the sensor position must be adjusted, and new tests carried out. If still no clear correlation between the sensor signals of a process can be detected, it is necessary to revert to the MTM based identification of suitable measuring points in the process to find alternative measuring points. If a pattern is detected, the sensor signals can be stored and an activation function is set, depending on the deflection pattern.

Figure 9 shows the sensor signals from two magnetometers while performing an assembly operation (in this case a manual press operation) four times. Based on the sensor signal of the individual mounting executions, patterns in the signal were identified and an activation function was defined. In the above presented example, the activation function was defined according to the values of the two sensors: If the signals exceed the identified (though statistical analysis of the multiple monitoring of the assembly operation) threshold values, the activation function is triggered.

Since multiple different processes could be executed with one tool or piece of equipment with sensors attached, the sensor data feed can be used to detect those different processes. In that case, multiple activation functions are associated with that sensor data feed and can be triggered – e.g., a motion sensor on a manual press could detect different processes depending on the acceleration patterns. Multiple functions require sufficient differences between the activation functions – e.g., different threshold values or different value-patterns. The signals and patterns can become much more complex than in the example, depending on the process and sensor type, so that other pattern recognition methods could be necessary. Here, AI-based pattern recognition would be one of the most promising options. The same is true for camera and machine-vison based process detection, which is an example for very complex sensor signals – the advance of AI-methods has significantly improved options to automatically interpret complex (visual) data. Combining different types of sensors is another option to improve process detection accuracy potentially greatly: e.g., if a motion detection sensor is combined with an Automatic identification and data capture (AIDC), such as RFID or

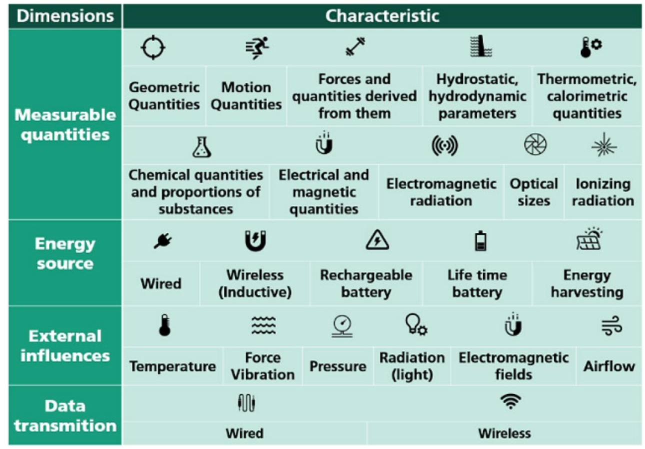

Fig. 7 sensor selection criteria (Nausch et al., 2021)

Table 1 Sensor types and measurable assembly parameters, based on (Kärcher et al., 2018) and extended by three more sensor types

Type of sensor	Task
RFID tags and reader	Localization of part, tool or jig Movement detection
Accelerometer	Movement and vibration detection
Magnetometer	Movement by magnetic field detection
Gyroscope	Movement detection
Thermometer and hygrometer	Temperature and humidity measurement
Ambient light sensor	Movement detection
Electronic scales	Weight measurement, part extraction check Movement detection by removing and placing part, tool or jig on the scale Localization and identification of part, tool or jig
Distance sensors (capacitive and inductive)	Part extraction check
6-axis force sensor	Presence of part, tool or jig by measurement of force and torque in x, y and z
Ultrasonic sensor	Measurement of distances between objects or fill levels
Gas sensor	Check concentration of certain gases in the air

Fig. 8 CISS sensor attached to a manual press

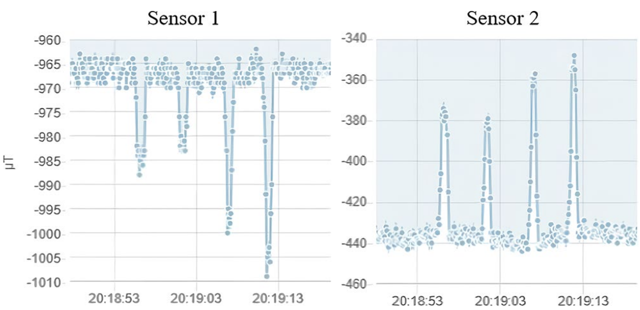

Fig. 9 Example of sensor signal during four manipulations, published by the authors in (Nausch et al., 2021)

barcode, a force sensor, or a machine vision system, the combined activation functions would lead to much higher detection accuracy then single sensor-detection. However, there is a trade-off between detection effort, investment and required accuracy, that must be considered when designing the system.

The sensor data feed for each sensor/process (a combination of sensors is also possible if a single signal is not sufficient to safely detect a process) is continuously analyzed via the activation functions. Upon triggering the function, an event with a timestamp is created and logged as an indexed entry in a central event database – this constitutes the ED process already described in the context of the entire synchronization process in Sect. "Synchronizing the DT with the production".

Results

In order to test and validate the procedure, an experimental setup was implemented in the TU Wien Pilot Factory Industry 4.0 (Hennig et al., 2019). In the case-study, a toy truck is assembled as a demonstration product. Figure 10 gives an overview of the workstation layout and modules.

The infrastructure of the pilot factory was first analyzed for the application of IoT sensors. Both wired and wireless sensors, with battery power supply and a Bluetooth connection for data transmission, were identified as suitable sensor technology for the application. The sensor data is transferred to a MySQL database.

The demonstration and evaluation were performed according to the approach described above:

- The process description was created using the extended MTM description. The assembly activities are described by 55 MTM codes. For each task, the corresponding elements involved have been added.
- Based on this MTM description, suitable measuring points were identified. The list of tools, resources,

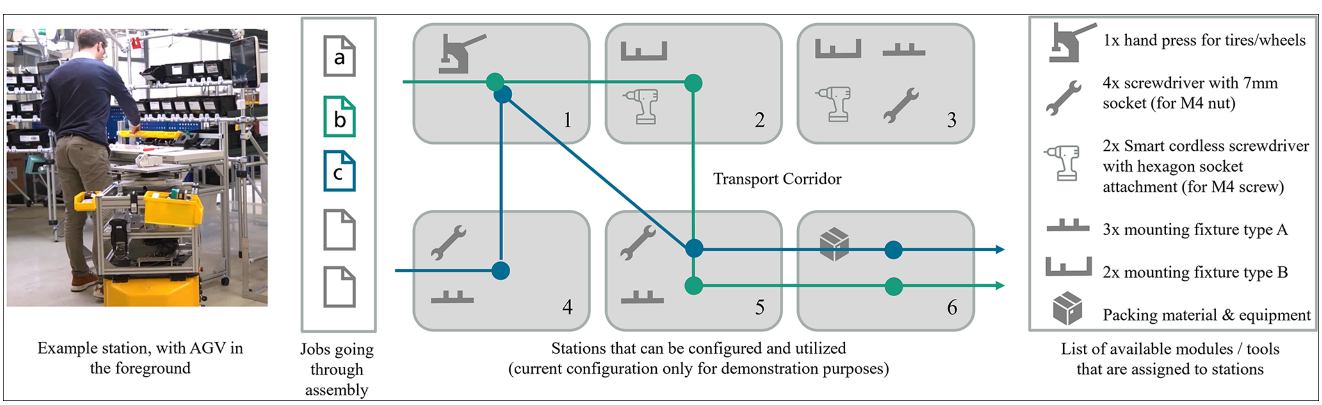

Fig. 10 Overview of setup in the pilot factory test environment

materials and supply devices and small load containers that are assigned to measuring points was derived.
- The tools, material, containers, supply devices for the measuring points were equipped with sensors.
- Through monitoring the sensor data feed for repeated process executions for each measuring point and each process, recognition patterns (i.e., threshold values and patterns) are derived and stored in activation functions.
- The sensor data feeds were assigned to the activation functions. The functions are directly linked to the assembly activities. Thus, the corresponding functions can be activated, and the matching activities can be marked as completed in the routing sheet by the EP module.

One of the analyzed and sensor-detected processes was the pressing of tires and rims of the toy truck, which we will show as an example for the measuring procedure. For this purpose, a CISS environmental sensor (Robert Bosch GmbH, 2022) was attached to the press and the sensor signal was measured during the performance of the activity (see Fig. 9). By operating the lever, the magnetic field and thus also the sensor signal changed. The activation function is based on a threshold value.

Of the 55 assembly tasks involved in the entire assembly process, 60% of the tasks could be clearly identified with environmental sensors. This offers enough possibilities to detect relevant steps within the assembly sequence. Since MTM analysis results in fine detail and thus small process segments, the steps most suitable for an easy sensor detection can be selected. The number of necessary sensor points depends on the availability of control-layer data (i.e., if a machine status is available and can be used to identify a process step, no further sensor point is necessary for that) and a combination of required reaction time from the DT and potential planning and control measures. For example, in high volume, highly automated assembly processes with short processing times, the reaction time should be very quick (e.g., within seconds), in order to be able to react to disturbances in the process – the time to detect disturbances, i.e. deviations from the plan, is shorter if there are more sensor or other data gathering points within the process chain. Conversely, in the case of a more manual process, with longer processing times and possibly more time variance, the reaction does not have to be as fast – this means fewer sensor or other data points are sufficient for the available sensible planning and control measures. Since more sensor points equal higher equipment cost and data processing effort, it is sensible to add sensor points as frugally as possible, especially since adding more sensor points later if required is always an option. Potential bottlenecks within a process sequence are usually the first choice for placing sensors as the disturbances here should be caught early to not lose time on the scarcest resources.

A major aspect to consider for the practical applicability of the developed ED concept is the ability to support the complexity and scale of real-life applications. Hundreds of processes and pieces of equipment are to be expected for an "average" production facility; thus, the ED system must support hundreds of measuring points and data feeds that must be analyzed. The CISS IoT technology used in the demonstration supports 7 sensors per IoT gateway and the industrial computer can connect to multiple IoT gateways, up to a total of 1.000 sensor feeds. In many industry applications these specifications could suffice, especially considering that not every single process has to be detected by the ED – a prioritization of relevant process gates could keep the number of sensors within practically feasible system specifications. It also must be noted that IoT technology is rapidly developing and the limits concerning system speed and size are likely to increase significantly in the coming years.

Visualization

Problem description and approach

The final step in the implementation of CRMS planning and optimization is to feed the information from the DT back into the physical world. The better and more timely the planning results can be communicated to the shop-floor staff, the better the prospects for a smooth execution of the plan (Matthias Tauber et al. 2019). For this purpose, a web-based dashboard was developed, which visualizes the information from the planning module, adapted to different roles, from operator on the shopfloor to an overview perspective for planners.

Related work

Gröger et al. (Gröger et al., 2013) have developed an operational dashboard for manufacturing, geared towards supporting shopfloor level operations, based on a literature analysis followed by expert interviews. They identify 4 major information needs:

- process context: conveying an understanding of the entire process, e.g. by informing about past, current and upcoming production orders.
- process performance: monitoring current performance metrics, e.g. current cycle time.
- process knowledge: information resources on the process, e.g. instructions.

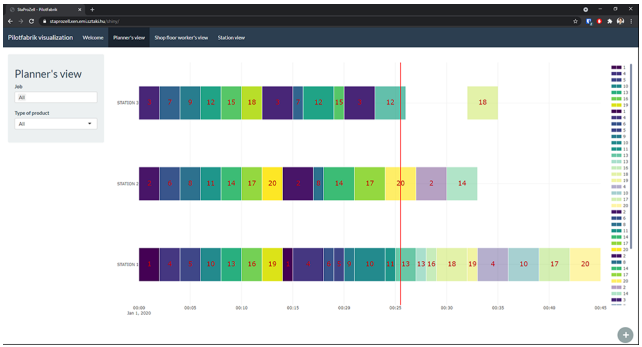

Fig. 11 Screenshot of planner's view

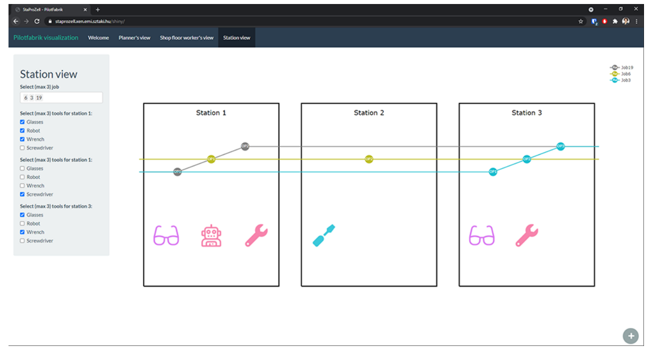

Fig. 12 Screenshot of the Station view

- process communications: enabling direct communication with other stakeholders within the process-chain.

Tokola et al. (Tokola et al., 2016) emphasize the importance of role specific information and KPIs in manufacturing dashboards and the need to develop user-centered views. The visualization developed in the paper at hand focusses mostly on the process context and process performance aspect, since those two aspects are closely linked to the implementation of optimized plans for the assembly process. The role specific information, following the user-centered paradigm, was considered by providing information for the planners, aimed at providing an overview of the plan and status of the entire assembly process, and for the shop-floor staff, with station- and process-step dependent focused information.

Developed solution

The main goal of developing a supporting dashboard was to make the displays required for planning and executing assembly activities as clear and easy to understand as possible, without losing essential information or taking it out of context. To transfer the continuously changing information from the two sources, namely the optimization model as well as the sensor data via the pre-processing module, to the respective employee, a web-based dashboard with three types of views was developed. The dashboard was implemented in a web application on a local server and is fed by the data of the DT. It is based on the programming language R (version 4.0.4) and data visualization was implemented using the package R Shiny application. The elements displayed in the graphs include information about the variants to be produced, running, completed and pending jobs, the active stations including the active modules, as well as the duration of the activities. The following three user group-specific views have been developed:

- Planner's view (Fig. 11): the planners view consists of a dynamic Gantt chart which displays executed and future jobs on a single timeline. The already completed jobs and assembly steps are played back to the system based on the sensor data, while the upcoming jobs in the respective stations future are sent to the system from the DT's optimized planning. On the left side, a drop-down menu is provided, where planners can select those jobs or product variants that should be highlighted. A color code helps to distinguish the jobs from each other. A mouse-over function provides more information on each order by clicking on the orders in the chart. The red vertical line is the current time, with past processes on its left and upcoming processes on its right.
- Station view (Fig. 12): The station view displays the currently calculated assembly paths of a certain selection of jobs through the assembly stations. The station view shows the worker the active assembly stations including the active modules, i.e. which tools and processing capabilities are currently assigned to the workstation, as well as the allocation by assembly orders. Each job is represented by a colored line, the individual work steps as nodes within the respective station. The available modules in each station are displayed to the workers as simple icons. Like in the Planner view, the worker has the possibility to customize the dashboard by filtering jobs and active modules per station.
- Shopfloor worker's view (Fig. 13): The Shopfloor View is a detailed view for the operational employee, analogous to the Gantt chart representation from the Planners View.
- In addition to the features from the Planner's View, i.e. and overview of jobs and job sequence, the workers are provided with information about running, soon to be completed and soon to be started jobs (see table on the right). For each section, the IDs of the operation, the jobs and the assigned station are displayed, as well as the remaining time until the end or start of a corresponding activity.

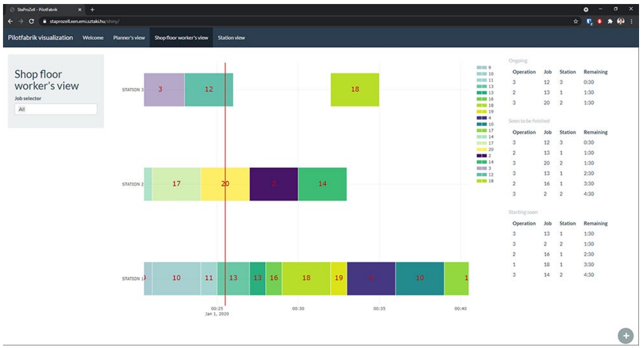

Fig. 13 Screenshot of the Shop floor worker's view

Results

Since the visualization is the main interface of the planning method with the human operators in the assembly, a major element of the development is incorporating feedback from operators on the three different layers (planner, station, shopfloor worker) in iterative steps. In the demonstrator development included in this presentation, the included steps were: an initial concept workshop from the development team concerning the basic functionality (i.e. providing status information on orders at different workstations and the upcoming orders as well as the basic concept of different user-groups), followed by a workshop with students and staff working at the TU Wien Pilot Factory Industry 4.0 in which the three layer views were conceptualized and finally a test workshop were the entire DT as presented in this paper was tested. This test was to establish the basic functionality and feasibility of the concept – a more thorough testing and evaluation must follow, which will enable detailed feedback and iterative improvement, especially of the visualization module. However, it must be noted that the goal is not the development of a ready-to-use software for industry use but rather a holistic concept that is demonstrated, potential-evaluated and ready for implementation in existing industry IT environments. Thus, a significant part of the visualization development in terms of usability will have to be included in that final development step, beyond research papers.

Discussion and outlook

The main goal of this paper was to develop a demonstrator for a DT approach for a CRMS or matrix production setting. All four identified necessary elements, from simulation and optimized planning and control, through synchronization of the DT with the physical manufacturing system, to IoT sensor-based status data generation and a visualization module for different user groups, were developed and integrated into a working demonstrator in a learning factory environment. The main contributions of the paper are:

- The development of a simulation-based optimization module that applies reinforcement learning and GAs to optimize the module configuration and job routing in CRMS, considering a multi-criteria objective function and constraints. The most innovative aspect lies in in the inclusion of RL to achieve an almost instant optimization response for the most dynamic parts of the optimization task, i.e. dynamic routing and job-workstation assignments, with encouraging initial results pointing towards a principal feasibility of the approach. However, it also shows the current limitations and technology readiness level of this specific method-application aspect, i.e. restriction of RL support to a simplified system with backup-algorithms for now necessary for the full assembly system complexity. At the same time, this is an important indicator for the prioritization of research needs in this area of Optimized Control of CRMS.
- The design of a synchronization module that links the physical and virtual systems via sensors and event handling, ensuring the consistency and reliability of the data exchange and enabling (near) real-time feedback and adaptation. Building on more theoretical and conceptual frameworks, the development in this paper shows how the crucial event handling to synchronize the physical and virtual sides of the DT for planning and control purposes can be realized: (a) on a structural/conceptual level, but (b) also in practical terms, by providing a demonstrator implementation and evaluation of the feasibility of the approach. This should not be understood as a prescribed blueprint for industrial DT implementations for CRMS. Rather, it is intended as an incremental step within a development progression aiming to increase the efficiency and adaptability of DT setups, creating a technology readiness level that supports the execution of large-scale industrial applications.
- The development of a sensor concept that utilizes IoT sensors to detect assembly-process steps to update status information for the processes in the assembly system, thus enabling a continuously updated digital shadowing of the real-world assembly system. This paper contributes a systematic approach to utilize state of the art lot sensor technology, based on the MTM process design method.
- The creation of a visualization module that communicates the optimized plans and control measures to the shop floor staff, enhancing their awareness and understanding of the system status and performance. Here, the principle of user-centric information provision, identified in the literature review, was transferred into

a working demonstrator, for which initial user-feedback could be collected.

While this is a demonstration of the feasibility of the concept of a constantly optimized flexible production system that provides the productivity benefits of flowline production systems with considerably more flexibility, there are still major challenges to overcome until this concept is ready for industry use:

The simulation is sufficiently developed, as the necessary adaptions for simulations in conventional linear production systems were relatively easy to implement, i.e. the process sequence flexibility or the ability to re-configure the production system structure during the simulation. The optimization could be demonstrated as well, although the RL approach for dynamic routing and workstation allocation must be developed further in order to handle the full complexity case with multiple products, hundreds of orders and multiple stations, which can be expected in full-scale production environments. The RL approach is meant to show the most promising research trajectory in providing a responsive optimization, which is especially important for CRMS, where flexibility and constant changes are part of the concept.

Concerning the synchronization of DT and real-world production system, the approach developed in this demonstrator is meant to show the feasibility of the event management with planned events and those that have already taken place and were detected by sensors. While the demonstrator features a solution that directly accesses the control layer of the devices and sensors, an alternative option would be that the event manager could have a data interface only with an existing MES or process control software and not access the control layer directly – or a combination of both is also possible: i.e., accessing most status date from a central data link to the MES or process control software and directly accessing the status of certain IoT sensors for especially time-critical status data. This decision is process dependent and can be illustrated with two examples from the meat production industry were the author's research group have created DTs for PPC in conventional automated production settings (i.e. not for flexible manufacturing systems): in the production of bacon, the processes are on a timescale of minutes to many hours, while in a poultry processing plant, processes can be on a timescale of fractions of a second (here, the speed of a line can reach 5 products per second, which are then automatically distributed to workstations for automatic processing units). While in the first case a status update from the ERP/MES system is sufficient, in the second case it might be beneficial to directly access sensor data from the control layer to enable responsive split-second decisions. The presented event handling approach is relevant and applicable irrespective of this question of how the status data connection is established.

Concerning the sensors, the basic feasibility of the approach of using functions to detect events could be shown. In a next step it is planned to include other sensor systems like cameras with an object recognition method, in addition to "traditional" sensors, for contactless identification of further parts and materials to record assembly processes more reliably. This can also reduce the cost and necessary infrastructure as cameras could be considered a universal sensor, with the process specific detection capability "outsourced" to an AI detection algorithm that can be trained to work for a large variety of processes. For such systems the activation function must be adapted to the interface of the object recognition, but this does not represent a contradiction in the methodical procedure and will be the content of further research projects. Scaling the system with IoT components together with the synchronization of the DT for larger systems, with hundreds of objects, is another subject of the ongoing research track.

Concerning the visualization and the interaction of human planners with the planning method, a basic concept could be demonstrated that considers requirements for operators on the shopfloor as well as those of planners. The visualization must be part of the graphic user interface of a planning tool and usability aspects must be considered in the context of implementing the status visualization in traditional planning views of MES and process control software; while some aspects of this can be part of research, a big part will likely be addressed in user-centric software development.

A major next step, encompassing and addressing most of the limitations, will be a demonstration in an industrial demonstrator where the developed method is transferred from the current learning factory to a real-world industrial application. This will enable a more comprehensive evaluation (a) of the real-world optimization benefits of the approach, e.g. compared to manual planning or traditional planning approaches, and (b) of the usability and it-implementation and performance aspects of the approach. Next to the mentioned technical challenges, there are also practical questions concerning the applicability and real-life benefits of flexible manufacturing concepts, i.e. evaluating the cost benefit ratio of additional implementation effort versus the flexibility benefits for production systems (Perwitz et al., 2022), or evaluating acceptance and for these production concepts. This is another category of future research that will impact mainly the relevance of methods. Intelligent methods to optimally plan and control flexible assembly systems in turn also have the potential to reduce their implementation and operational effort while increasing the benefits though increased resource utilization due to a better planning quality.

Funding The research presented is part of the research project StaProZell, funded by the Austrian Research Promotion Agency (FFG) (Project number 864792). The authors would like to thank all project partners for their contributions, especially to our colleagues at SZTA-KI for their support in implementing the visualization module.
Open access funding provided by TU Wien (TUW).

Data availability The data used in this study consists of various sets of data generated during the demonstrator evaluation, consisting of sensor-log data and simulation data. There is no large central data source we that could be shared and the relevant data underlying the findings are included in this article. Should additional access to specific data be required, it is available from the corresponding author upon request.

Declarations

Competing interests The authors declare no competing interests.

References

Alam, K. M., & El Saddik, A. (2017). C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems. In *IEEE access*, 1. https://doi.org/10.1109/ACCESS.2017.2657006

Ashraf, M., & Hasan, F. (2018). Configuration selection for a reconfigurable manufacturing flow line involving part production with operation constraints. *International journal of advanced manufacturing technology*, *98*, 5–8. https://doi.org/10.1007/s00170-018-2361-7

Badarinath, R., & Prabhu, V. V. (2017). Advances in Internet of Things (IoT) in manufacturing. *513*, 111–118. https://doi.org/10.1007/978-3-319-66923-6_13

Bortolini, M., Galizia, F. G., & Mora, C. (2018). Reconfigurable manufacturing systems: Literature review and research trend. *Journal of Manufacturing Systems*, *49*, 93–106. https://doi.org/10.1016/j.jmsy.2018.09.005

Brahimi, N., Dolgui, A., Gurevsky, E., & Yelles-Chaouche, A. R. (2019). A literature review of optimization problems for reconfigurable manufacturing systems. *IFAC-PapersOnLine*, *52*(13), 433–438. https://doi.org/10.1016/j.ifacol.2019.11.097

Brenner, B., & Hummel, V. (2017). Digital twin as enabler for an innovative digital shopfloor management system in the ESB logistics learning factory at Reutlingen - University. In *Procedia Manufacturing*, *9*, 198–205. https://doi.org/10.1016/j.promfg.2017.04.039

Chen, C., Zhang, C., Wang, T., Li, D., Guo, Y., Zhao, Z., & Hong, J. (2020). Monitoring of assembly process using deep learning technology. *Sensors (Basel Switzerland)*, *20*(15). https://doi.org/10.3390/s20154208

Csáji, B. C., Monostori, L., & Kádár, B. (2006). Reinforcement learning in a distributed market-based production control system. *Advanced Engineering Informatics*, *20*(3), 279–288. https://doi.org/10.1016/j.aei.2006.01.001

Dávid, G. (2014). Botond Kádár; András Kovács; László Monostori: Capacity management for assembly systems with dedicated and reconfigurable resources. In *CIRP Annals - Manufacturing Technology*. https://doi.org/10.1016/j.cirp.2014.03.110

De Almeida D. L. M, & Ferreira J. C. E. (2009). Analysis of the Methods Time Measurement (MTM) methodology through its application in Manufacturing companies. https://doi.org/10.13140/RG.2.1.2826.1927

Dou, J., Li, J., & Su, C. (2016). Bi-objective optimization of integrating configuration generation and scheduling for reconfigurable flow lines using NSGA-II. *International Journal of Advanced Manufacturing Technology*, *86*, 5–8. https://doi.org/10.1007/s00170-015-8291-8

ElMaraghy, H. A. (2010). *Changeable and reconfigurable manufacturing systems*. Springer. Springer Series in Advanced Manufacturing. https://doi.org/10.1007/978-1-84882-067-8

Fei Tao; He Zhang, Ang Liu, A. Y. C., & Nee (2019). Digital Twin in Industry: State-of-the-art. *IEEE Transactions on Industrial Informatics*, *15*(4), 2405–2415. https://doi.org/10.1109/TII.2018.2873186

Fernando, F. (2020). Bruno Rocha; Alexandre Mota; Paulo Pedreiras; Valter Silva: Extending MQTT-SN with Real-Time Communication Services. 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 2020, pp. 1–4. https://doi.org/10.1109/ETFA46521.2020.9212147

Foith-Förster, P., & Bauernhansl, T. (2021). Axiomatic design of matrix production systems. *IOP Conf Ser.: Mater Sci Eng*, *1174*(1), 12022. https://doi.org/10.1088/1757-899X/1174/1/012022

Göppert, A., Grahn, L., Rachner, J., Grunert, D., Hort, S., & Schmitt, R. H. (2023). Pipeline for ontology-based modeling and automated deployment of digital twins for planning and control of manufacturing systems. *Journal of Intelligent Manufacturing*, *34*(5), 2133–2152. https://doi.org/10.1007/s10845-021-01860-6

Gröger, C., Hillmann, M., Hahn, F., Mitschang, B., & Westkämper, E. (2013). The operational process dashboard for manufacturing (7). https://doi.org/10.1016/j.procir.2013.05.035

Gyulai, D., & Monostori, L. (2017). Capacity management of modular assembly systems. *Journal of Manufacturing Systems*, *43*, 88–99. https://doi.org/10.1016/j.jmsy.2017.02.008

Halbwidl, C., Sobottka, T., Gaal, A., & Sihn, W. (2021). Deep reinforcement learning as an optimization method for the configuration of adaptable, cell-oriented assembly systems. In *Procedia CIRP* 104, pp. 1221–1226. https://doi.org/10.1016/j.procir.2021.11.205

Hennig, M., Reisinger, G., Trautner, T., Hold, P., Gerhard, D., & Mazak, A. (2019). TU Wien pilot factory Industry 4.0. In *Procedia Manufacturing* 31, pp. 200–205. https://doi.org/10.1016/j.promfg.2019.03.032

Hu, X., Wan, J., Wang, T., & Zhang, Y. (2020). An IoT-based cyber-physical framework for turbine assembly systems. *Ieee Access: Practical Innovations, Open Solutions*, *8*, 59732–59740. https://doi.org/10.1109/ACCESS.2020.2983123

Iris Graessler, A., & Poehler (2017). Integration of a digital twin as human representation in a scheduling procedure of a cyber-physical production system. *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Singapore, pp. 289–293. https://doi.org/10.1109/IEEM.2017.8289898

Jürgen Fleischer; Benedikt Klee, Spohrer, A., & Simon Merz. (2018). *Leitfaden Sensorik für Industrie 4.0 - Wege zu kostengünstigen Sensorsystemen*. VDMA, KIT wbk Institut für Produktionstechnik.

Kaczmarek, S., Hogreve, S., & Tracht, K. (2015). Progress monitoring and gesture control in manual assembly systems using 3D-image sensors. In *Procedia CIRP* 37, pp. 1–6. https://doi.org/10.1016/j.procir.2015.08.006

Kanawaday, A., & Sane, A. (2017). Machine learning for predictive maintenance of industrial machines using IoT sensor data. In *2017 8th IEEE International Conference*, pp. 87–90. https://doi.org/10.1109/ICSESS.2017.8342870

Kärcher, S., Cuk, E., Denner, T., Görzig, D., Günther, L. C., Hansmersmann, A. (2018). Sensor-driven analysis of manual assembly systems. In *Procedia CIRP* 72, pp. 1142–1147. https://doi.org/10.1016/j.procir.2018.03.241

Kombaya Touckia, J., Hamani, N., & Kermad, L. (2022). Digital twin framework for reconfigurable manufacturing systems (RMSs): Design and simulation. *International Journal of Advanced Manufacturing Technology*, *120*, 7–8. https://doi.org/10.1007/s00170-022-09118-y

Kuhnle, A., Röhrig, N., & Lanza, G. (2019). Autonomous order dispatching in the semiconductor industry using reinforcement learning. In *Procedia CIRP* 79, pp. 391–396. https://doi.org/10.1016/j.procir.2019.02.101

Kumar, A., Pattanaik, L. N., & Agrawal, R. (2019). Optimal sequence planning for multi-model reconfigurable assembly systems. *International Journal of Advanced Manufacturing Technology*, *100*, 5–8. https://doi.org/10.1007/s00170-018-2826-8

Leng, J., Zhang, H., Yan, D., Liu, Q., Chen, X., & Zhang, D. (2019). Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop. In *Journal of Ambient Intelligence and Humanized Computing* 10, pp. 1155–1166. https://doi.org/10.1007/s12652-018-0881-5

Liu, L., Liu, Y., & Zhang, J. (2019). Learning-based hand motion capture and understanding in assembly process. *IEEE Trans Ind Electron*, *66*(12), 9703–9712. https://doi.org/10.1109/TIE.2018.2884206

Matthias Tauber; Andreas Gallmetzer, Erwin, R., Christopher, A., Brown; Dominik, T., & Matt (2019). Concept design of a digital shop floor information system for assembly operators in machine industry. In *MATEC Web Conf*, *301*, The 13th International Conference on Axiomatic Design (ICAD 2019). https://doi.org/10.1051/matecconf/201930100017

Michael Schluse; Juergen Rossmann (2016). From simulation to experimentable digital twins: Simulation-based development and operation of complex technical systems. *2016 IEEE International Symposium on Systems Engineering (ISSE)*, Edinburgh, UK, pp. 1–6. https://doi.org/10.1109/SysEng.2016.7753162

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533. https://doi.org/10.1038/nature14236

Modoni, G. E., Caldarola, E. G., Sacco, M., & Terkaj, W. (2019). Synchronizing physical and digital factory: benefits and technical challenges. In *Procedia CIRP* 79, pp. 472–477. https://doi.org/10.1016/j.procir.2019.02.125

Nausch, M., Hold, P., & Sihn, W. (2021). A methodological approach for monitoring assembly processes. *Procedia CIRP*, *104*(2), 1233–1238. https://doi.org/10.1016/j.procir.2021.11.207

Perwitz, J., Sobottka, T., Beicher, J. N., & Gaal, A. (2022). Simulation-based evaluation of performance benefits from flexibility in assembly systems and matrix production. *Procedia CIRP*, *107*, 693–698. https://doi.org/10.1016/j.procir.2022.05.047

Petroodi, S., Ehsan Hashemi; Eynaud, B. D., Klement, N., & Tavakkoli-Moghaddam, R. (2019). Simulation-based optimization approach with scenario-based product sequence in a reconfigurable manufacturing system (RMS): A case study. In *IFAC-PapersOnLine*, *52*(13), 2638–2643. https://doi.org/10.1016/j.ifacol.2019.11.605

Pierleoni, P., Belli, A., Palma, L., & Palmucci Michela; Sabbatini Luisiana. (2020). A machine vision system for manual assembly line monitoring. In *International Conference on Intelligent Engineering and Management (ICIEM)*, pp. 33–38. https://doi.org/10.1109/ICIEM48762.2020.9160011

Rebmann, A., Knoch, S., Emrich, A., Fettke, P., & Loos, P. (2020). A multi-sensor approach for digital twins of manual assembly and commissioning. In *Procedia Manufacturing* 51, pp. 549–556. https://doi.org/10.1016/j.promfg.2020.10.077

Robert Bosch GmbH (2022). *Connected industrial sensor solution*. Available online at https://www.bosch-connectivity.com/products/industry-4-0/connected-industrial-sensor-solution/downloads/, updated on 1/10/2022, checked on 3/18/2022.

Rosen, R., Wichert, G., Lo, G., & Bettenhausen, K. D. (2015). About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine*, *48*(3), 567–572. https://doi.org/10.1016/j.ifacol.2015.06.141

Seongjin, Y., & Won-Tae, J. H. P. (2017). Kim: Data-centric middleware based digital twin platform for dependable cyber-physical systems. *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, Milan, Italy, pp. 922–926. https://doi.org/10.1109/ICUFN.2017.7993933

Siddhartha Kumar Khaitan, James, D., & McCalley (2015). Design techniques and applications of cyberphysical systems: A Survey. *IEEE Systems Journal*, *9*(2), 350–365. https://doi.org/10.1109/JSYST.2014.2322503

Sobottka, T., Kamhuber, F., Rössler, M., & Sihn, W. (2018). Hybrid simulation-based optimization of discrete parts manufacturing to increase energy efficiency and productivity. In *Procedia Manufacturing* 21, pp. 413–420. https://doi.org/10.1016/j.promfg.2018.02.139

Sobottka, T., Kamhuber, F., Faezirad, M., & Sihn, W. (2019). Potential for machine learning in optimized production planning with hybrid simulation. In *Procedia Manufacturing* 39, pp. 1844–1853. https://doi.org/10.1016/j.promfg.2020.01.254

Sobottka, T., Kamhuber, F., & Heinzl, B. (2020). Simulation-based Multi-criteria optimization of parallel heat treatment furnaces at a casting manufacturer. *JMMP*, *4*(3), 94. https://doi.org/10.3390/jmmp4030094

Sutton, R. S., Bach, F., & Barto, A. G. (2020). *Reinforcement learning. An introduction* (2 ed.). MIT Press Ltd (Adaptive Computation and Machine Learning series).

Tao, W., Lai, Z. H., Leu, M. C., & Yin, Z. (2018). Worker activity recognition in smart manufacturing using IMU and sEMG signals with convolutional neural networks. In *Procedia Manufacturing* 26, pp. 1159–1166. https://doi.org/10.1016/j.promfg.2018.07.152

Thramboulidis, K., Vachtsevanou, D. C., & Kontou, I. (2019). CPuS-IoT: A cyber-physical microservice and IoT-based framework for manufacturing assembly systems. *Annual Reviews in Control*, *47*, 237–248. https://doi.org/10.1016/j.arcontrol.2019.03.005

Tokola, H., Gröger, C., Järvenpää, E., & Niemi, E. (2016). Designing manufacturing dashboards on the basis of a key performance indicator survey. In *Procedia CIRP* 57, pp. 619–624. https://doi.org/10.1016/j.procir.2016.11.107

Tolio, T., Ceglarek, D., ElMaraghy, H. A., Fischer, A., Hu, S. J., Laperrière, L., et al. (2010). SPECIES—Co-evolution of products, processes and production systems. *CIRP Annals*, *59*(2), 672–693. https://doi.org/10.1016/j.cirp.2010.05.008

Uhlemann, T. H. J., Lehmann, C., & Steinhilper, R. (2017). The digital twin: Realizing the cyber-physical production system for Industry 4.0. In *Procedia CIRP* 61, pp. 335–340. https://doi.org/10.1016/j.procir.2016.11.152

Ward, J. A., Lukowicz, P., Tröster, G., & Starner, T. E. (2006). Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(10), 1553–1567. https://doi.org/10.1109/TPAMI.2006.197

Wiendahl, H. P., ElMaraghy, H. A., Nyhuis, P., Zäh, M. F., Wiendahl, H. H., Duffie, N., & Brieke, M. (2007). Changeable manufacturing - Classification, design and operation. *CIRP Annals*, *56*(2), 783–809. https://doi.org/10.1016/j.cirp.2007.10.003