PRODUCTION MANAGEMENT

The concept of a maximal network plan to enhance automated decision making in production steering

Roland Larek[1] · Jan Cetric Wagner[1] · Heiko Grendel[2] · Andreas Nüchter[3]

Abstract

Digitalization is a major driver of competitiveness in industry. Especially, automated scheduling systems are broadly available to deal with production steering in complex production systems. Their underlying concept is the network plan to describe the logical relation between different production steps for the completion of a production process. The network plan, however, is limited to contain only one standard process and allows therefore basically only the automated variation of execution time of each step within the given logical sequence. This is sufficient as long as the production process and the boundary conditions, such as resource availability, are stable to a certain degree. This work introduces an enhancement of the concept of the network plan, the Maximal Network Plan, that is designed to automatically cope with disruptions. It stores additional alternative sequences, weighted according to their priority in comparison to the standard process. Therefore, a numerical decision making is possible in terms of also varying the order of production tasks to achieve a maximum performance for the overall process. Instead of waiting for a disruption in the standard process to be solved, an alternative sequence will be proposed to increase overall progress by circumventing the blocked step. This concept has been developed and tested for complex assembly tasks, such as the production of air- and spacecraft structures and systems. In conjunction with the use of mobile computing devices instead of paper-based work order documents it allows a very fast and dynamic production steering, also in the case of substantial disruptions in the standard process.

Keywords Digitalization · Production steering · Industry 4.0 · Assembly planning · System integration

1 Introduction

The maturity of digitalization in industry is highly depending on the specific environment. Automated production processes, large batch sizes and a high integrational level of software based production management systems are advantages in terms of the implementation of the concepts and ideas of Industry 4.0 [1]. On the other hand, especially the high-tech industry, such as the production of air- and spacecrafts and other special machinery lacks of these factors. In comparison small batch sizes and complex assembly sequences hinder automation and manual work is the most important factor for value adding [2]. Therefore, production

planning systems that are used for coordinating the manual work are seen as a platform for pushing digitalization [3]. This work deals with the automation of decision making in production planning systems used in assembly and system integration, where especially a fast and reliable reaction to disruptions in the regular process is needed. Until now, individual manual decisions and paper-based documents slow down the production to a high extend, although the knowledge of alternative procedures is given.

The concept of production planning systems, such as Manufacturing Execution Systems (MES) and respective modules in Enterprise Resource Planning Systems (ERP), bases on the principle of structuring information into product, process and resource (PPR) [4]. The manufacturing of a given product is being performed through a process, which again requires certain resources [5]. Whereas the product, its sub-assemblies and components are documented in the product breakdown structure [6], the manufacturing process is described in a more or less visible manner in a network plan. It holds the order of predecessors and successors for

✉ Roland Larek
roland.larek@hs-wismar.de

1 Hochschule Wismar, 23966 Wismar, Germany

2 Hochschule Bremen, 28199 Bremen, Germany

3 Universität Würzburg, 97070 Würzburg, Germany

each production step. In the simplest case this network plan is a short linear sequence for parts machining but can consist of hundreds of nodes in a highly branched network for elaborate assembly operations. The available resources, such as personnel with different qualifications, fixed and moveable equipment, machines, workplaces, transport containers and even work or storage space are managed in a database, following a pooling principle. The material is also treated as a resource but is managed as inventory. This concept allows the computerized scheduling of large and complex production processes with multiple products, processes and resources and is state of the art for several decades already.

In complex assembly operations, often a range of different assembly strategies is feasible but the planner has to decide for one because due to its concept the network plan can contain only one. This leads to the main issue of describing production processes in a network plan. If the due successor of a finished production step cannot be performed because one of the resources is not available, there are only two options to deal with; either the process is stopped until all required resources are available or a manual replanning outside of the planning system is being carried out. Both options are not sufficient, especially if the time to gain the missing resource is very long and / or the number of manual replannings is increasing. The latter leads to an inevitable loss of validity of the existing network plan. Furthermore, manual decision making in case of an urgent problem is highly reliant on the knowledge and experience of the specific person in charge. That often leads to poor decisions due to incomplete information on interdependencies between different production steps and the resulting consequences of spontaneously changing their order. After a replanning new paper-based documents have to be created and distributed to the shop-floor personnel, which again takes additional time.

This work represents latest results from research projects dealing with the digitalization of process knowledge and its automated evaluation and use in production management. In the focus is the body of theories of the Maximal Network Plan. The name is intended as a reference to the Maximal Work Plan ("Maximalarbeitsplan"), that is a well-known concept for documenting variants of a product in production planning [7].

2 State of the art

Research activities aiming at increasing the efficiency in production scheduling can be divided into different categories which are discussed in the following.

Recent work in the context of Industry 4.0 deals in a more general perspective with the optimization of data collection, exchange and use in a factory for more efficient interaction and decision making. Exemplary, an overall generic architecture for the interlinkage of all mechanical and computational systems in a factory is being proposed in [8]. Hošovský et al. discuss the role of computational intelligence including concepts such as machine learning, optimization algorithms and approximate reasoning to achieve an overall optimization of performance [9]. Virtual cross-company networks with the same goal are in the focus of [10, 11].

A broad range of literature deals with scheduling problems, that are typically to be found in parts manufacturing; a given number m of assumed identical, parallel or unrelated machines [12] is available as a static system. Jobs can either be kept in a pool and therefore are known in advance or arrive randomly and can, depending on the author, vary regarding the lot size, setup time or lead time. The order of jobs is considered the main variable to achieve a high performance of the overall system. The aim is to identify scheduling strategies that maximize throughput, minimize waiting times or maximize machine utilization by varying the rules of prioritization of the incoming jobs, which are waiting for execution in a queue in front of this system. Basic so-called dispatch rules are "first come first serve" or "shortest process time". [13] introduces a new rule called RTSLACK with the objective to select the one job out of the list of jobs waiting to be processed by one or more manufacturing resources, that would maximize the total slack time of the remaining jobs. [12] describes an approach based on simulated annealing to keep the work load within the system, consisting of m identical parallel machines, on a constant level. [14] discusses an approach based on a genetic algorithm to maximize throughput of a system called the hybrid flow shop. It consists of several layers of m parallel machines and assumes that all jobs must pass through every layer. [15] introduces an approach also based on genetic algorithms called dynamic scheduling because the arrival of jobs is random and unpredictable machine breakdowns are taken into account. [16] analyses the interaction of computational effort and quality of the decision. In opposition to the approach introduced in this work, however, none of the authors above uses a network plan or other kind of diagram to predefine the order of jobs. Therefore, the use for the scheduling of complex assembly tasks is very limited.

The available literature considering the use of diagrams to describe and schedule assembly operations is much more narrow compared to the job shop scheduling problem above. A number of authors has tried to overcome the limitation of having only one manually planned variant by generating assembly sequences automatically based on product data. Rochow et al. developed a precedence graph based on the product breakdown structure. The graph contains all technical constraints between its elements. They transfer the precedence graph via an adjacency matrix to an algorithm, which computes all possible process sequences. However,

they do not discuss the usability of the high number of computed solutions. Furthermore, no differentiation of the several alternative sequences exists, whereby all solutions have to be seen as equally feasible and efficient [17]. A further work uses an AND/OR-Graph to describe different adequate assembly sequences. Homem de Mello and Sanderson presented a graph that visualizes on the basis of disassembly operations alternative feasible assembly sequences. However, they include no differentiation of the presented solutions in the graph. Additionally, the authors mention that their approach is potentially not usable for highly engineered products. That assumption is based on the complexity of the graph due to the high amount of potential assembly sequences [18]. The work of Crowston et al. deals with a method to optimize the overall cost of a process which incorporates alternative methods for given tasks by introducing the so-called Decision CPM. It includes additional nodes to represent those alternatives and allow decision making by integer programming. They also state that this approach leads to large and complex graphs and is therefore predestinated for small problems [19]. Petri nets are used by [20] to store assembly sequences including process data such as assembly time. The sequences are generated out of a CAD model of the assembly. However, the demonstrated sample assembly consists of only 4 parts. In [21] the use of the simulated annealing method is demonstrated to generate all possible assembly sequences for a given assembly of 9 parts and design for assembly principles are applied to delete practically unfeasible solutions. None of the authors discuss the active use of their data in production scheduling, especially in terms of a dynamic reaction to disturbances and weighted alternative sequences. Due to the mainly practical and less scientific context only in [22] are strategies to be found regarding the handling of errors occurring on the production line. Depending on the specifics of the error there are named basically three options; stopping the assembly line until it is fixed, bypass the faulty product to a repair line or simply document the error and proceed.

Whereas in this work the adjacency matrix is used as a means for storing process data, another method is the incidence matrix. Both differ in terms of their structure and memory requirements. The memory requirement O of the incidence matrix is dependent of the number of nodes (n) and the number of edges (m) within the graph and equals $O(n*m)$. The memory requirement for the adjacency matrix is calculated using the number of nodes in the graph, regardless of how many edges exist $O(n^2)$. Therefore, for graphs with a small number of edges, the incidence matrix is most suitable in terms of memory requirements [23]. The maximal network plan has an increased number of edges due to the additional representation of the alternative sequences. In such cases, when there are more edges than nodes, it is therefore well justified to use

the adjacency matrix to store the graph [24]. In opposition to AND/OR graphs, Petri nets or other methods to store process data, the principle of the network plan is used as a standard in commercial production planning systems and therefore well understood and accessible in this context. Furthermore, it is capable of storing large processes with hundreds of nodes thus still being easy to visualize and to read.

The aim of this work is to develop a method based on the commonly accepted network plan that allows to contain a range of valid production sequences and information on their priority in relation to each other. Therefore, it allows a numerical replanning that enhances the performance of the available production scheduling systems also in case of substantial disruptions. The exhaustive calculation of the solution space, as it is done in [17] to [21], is intentionally not taken into account. The primary source of the initial network plan is the planner who provides one variant as a basis which is considered the most reliable one.

3 Theory of the maximal network plan (MNP)

3.1 Theoretical basis

Russel and Norvig described the principle of a network plan in their definitive book on artificial intelligence [25]. It is not the aim to reach all nodes or edges once, as it is the case in Euler or Hamilton circuit problems, nor is it necessary to compute the shortest path from start to end as in travelling salesman or routing problems. Rather, the network plan visualizes the predecessor-successor-relationship between each node by using directed edges. Nodes are understood as jobs or work orders in this context. An edge between two nodes implies that the predecessor has to be executed before the successor is begun. A timed relation is not defined.

A network plan is a directed graph that can be described by an adjacency matrix with the size $n \times n$, with n being the number of nodes of the graph. Each edge of the graph is stored in the corresponding field of the upper triangular matrix. All edges are equivalent in principle. Therefore, a *1* describes the specific entry in the matrix. The main diagonal has no entries respectively a *0* because the nodes of the production process cannot interact with themselves. Moreover, there are no backwards directed edges to describe the production process whereby no entries appear in the lower triangular matrix. That leads also to a *0* as input. Figure 1 shows a typical network plan and its corresponding adjacency matrix.
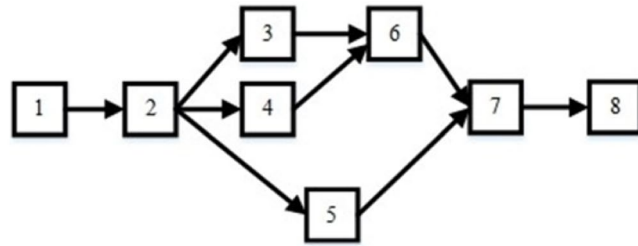
Fig. 1 A conventional network plan with eight nodes (right) and its corresponding adjacency matrix (left)

3.2 Enhancement of the concept of the network plan

Additional relations between nodes can be described by additional entries in the upper triangular matrix but they have to be considered as subordinate in comparison to the ideal sequence. Therefore, their respective entry is higher than *1*, which can also be read as a penalty. In the simplest case it is *2* which means the respective edges are of second priority to the standard process. Higher penalties are possible. The determination can be done intuitively by the planner or follow strict rules. This work proposes that the value in the first step is determined based on the number of edges between the predecessor node of the interrupted one and its alternative successor. That equates to the number of skipped nodes + *1*. The principle is that the more the alternative sequence is apart from the standard process, the less desirable it is. The aim is to stay as close to the standard process as possible and return to it as soon as possible. Figure 2 illustrates a respective extension of the example from Fig. 1.

Edges that have a penalty higher than *1* are referred to in the following as alternative edges. Nodes that are linked through an alternative edge and are considered for execution thereby are referred to as alternative nodes.

The introduction of alternative edges should never lead to circumventing nodes that are not due yet because their regular predecessor is not finished. It is only intended to circumvent a due successor of a finished node that is "blocked". It is acceptable on the other hand if an alternative edge with a priority higher than *2* leads to circumventing nodes in the direct sequence after one that is "blocked", as it is the case in the example above regarding node *6*.

In the interpretation of a typical network plan as soon as a node has been executed it is of no further importance in the ongoing sequence and its successor is the next node due. By introducing alternative edges this basic principle is not sufficient anymore. As a basis for numerical decision making further information on the status of each node is required. Nodes can be "finished" after their execution and are therefore of no further relevance unless they are a precondition to another node as explained in the following. While they are executed they bind resources and are being understood as "in progress". Due nodes according to the common interpretation of a network plan are considered "workable". In addition, there are nodes which are due but considered as "blocked", because a required resource is temporarily not available. As soon as the missing resource is available, the node should be executed with a higher priority to get closer to the standard sequence again. This status is treated as "repaired". The following differentiations are therefore introduced in this concept (Table 1). The color coding is for illustrational purposes only.

The actual status of each node can be stored in the respective field of the main diagonal of the adjacency matrix, as illustrated in Fig. 3.
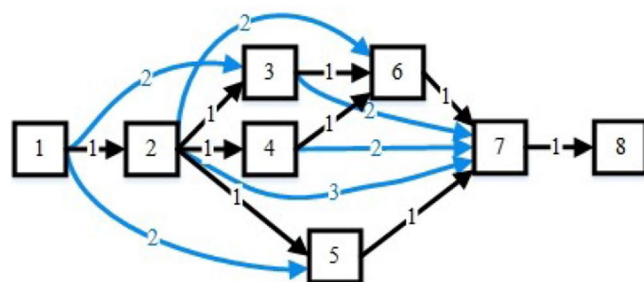

Fig. 2 Network plan with alternative edges and its corresponding adjacency matrix

Table 1 Overview of the possible node statuses

Status	Integer	Colour Coding	
Blocked	-2	■	(Red)
Repaired	-1	■	(Blue)
Workable	0	☐	(White)
In Progress	1	■	(Orange)
Finished	2	■	(Green)

A further extension is aiming to prevent failure in highly complex network plans by specifying preconditions and thus limiting the range of possible alternatives in case a blocked node is circumvented by an alternative edge. The example in Fig. 4 shows that an edge with priority *3* would bypass nodes number *3*, *4* and *6* in case nodes *3* and / or *4* are blocked. To perform node *7* on the other hand a precondition could be that node *3* must be finished yet before for technical reasons. This would limit the meaning of the alternative edge to only bypassing node *4* but not *3* and *6*. These preconditions can be defined by the planner individually for each node and are independent from alternative edges. Preconditions can be understood as edges directing backwards from a specific node, who's precondition they are, to a specific node that must be completed before the node can be performed. Figure 4 illustrates this with additional edges in yellow.

Two different levels of preconditions have proofed beneficial. A precondition can be specified as "restrictive" and is coded in this example with integer *1* in the lower triangular matrix if the specified node must be completed only to a certain degree but not fully. A manual decision would be necessary to determine if the progress reached so far is sufficient, unless the production data acquisition does provide this information. According to the example above the blue alternative edge with priority *2* would bypass nodes *3* and *4* (see Fig. 3). But the precondition with level *1* pointing from node *6* to node *2* (see Fig. 4) means that node *3* can only be bypassed if node *2* has been started yet and completed to a certain degree. Integer *2* in the lower triangular matrix identifies nodes that have to be fully completed. These are declared as "indispensable".

3.3 Selection and prioritization of alternative nodes

The introduction of alternative edges and the circumventing of blocked nodes lead to the situation that nodes with a different status may be available for execution:

1. nodes that have been blocked but are now available ("repaired");
2. nodes that are the due successor of a finished node ("workable");
3. alternative nodes pointed to by alternative edges with varying priority.

	v1	v2	v3	v4	v5	v6	v7	v8
v1	2	1	2	0	2	0	0	0
v2	0	-1	1	1	1	2	3	0
v3	0	0	1	0	0	1	2	0
v4	0	0	0	-2	0	1	2	0
v5	0	0	0	0	0	0	1	0
v6	0	0	0	0	0	0	1	0
v7	0	0	0	0	0	0	0	1
v8	0	0	0	0	0	0	0	0
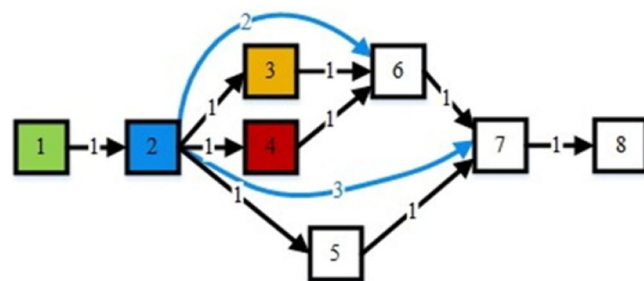
Fig. 3 Storing the status information in the main diagonal of the adjacency matrix

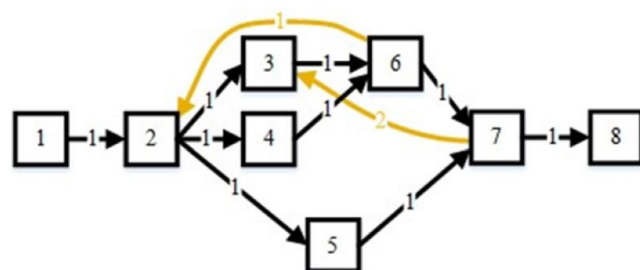|	v1	v2	v3	v4	v5	v6	v7	v8
v1	2	1	2	0	2	0	0	0
v2	0	-1	1	1	1	2	3	0
v3	0	0	1	0	0	1	2	0
v4	0	0	0	-2	0	1	2	0
v5	0	0	0	0	0	0	1	0
v6	0	1	0	0	0	0	1	0
v7	0	0	2	0	0	0	0	1
v8	0	0	0	0	0	0	0	0

Fig. 4 Definition of preconditions in the lower triangular matrix and the corresponding representation in the network plan

Furthermore, in the assembly of a complex system the aim is not to identify only one node to continue a production process but to identify all available process steps and then assign the available resources, such as personnel, to them in a useful order. Therefore, a set of rules is needed to prioritize. One paradigm of this concept is to stay as close to the standard process as possible. Therefore, "repaired" nodes must be executed as soon as they are available. In the following also nodes must be executed that are located behind a formerly "blocked" node in the same branch of the network plan and could therefore not be executed earlier. This is minimizing interference with other process steps. If no "repaired" nodes are left, the "workable" nodes have the highest priority. This also fulfils the paradigm mentioned above. It is assumed that nodes which are considered as "workable" have a released documentation and the required material is available. Otherwise they would have to be treated as "blocked".

If there are no or not enough "repaired" and "workable" nodes available to ensure sufficient progress on the one hand and a number of the due nodes is "blocked" because specific resources are missing on the other hand, alternative nodes are taken into account. They are only to be considered if their individual preconditions are satisfied. To prioritize between them requires different considerations. It is assumed that the impact on the overall process is lower if the penalty of the corresponding alternative edge is lower, thus meaning the number of circumvented or skipped nodes is lower. Therefore, a prioritization is proposed beginning with the lowest penalty. In a complex network plan a number of nodes with an equally low penalty might be available. A further parameter therefore is the process relevance of the alternative node to be evaluated. In graph theory, the neighborhood term describes the quantity of related nodes to a specified node [26]. In this context this term is used to declare the significance of a node in the production process. The higher the number of related nodes is, the more important is the node for the overall process. To reduce the impact of circumventing nodes, alternative nodes are prioritized higher the lower their process relevance is. To get back to the standard process soon it is furthermore useful to execute alternative nodes with short execution times before nodes with long execution times. Once the alternative node has been finished, a replanning takes place and the circumvented node might be repaired and available for execution. A useful extension could be to compare the expected time until the missing resource is available with the duration of the considered alternative node. Table 2 provides an overview of the rules described above.

This procedure can be automated to a high extent based on operations using the adjacency matrix. Manual intervention is possible but not strictly necessary for the approval of

Table 2 Overview of the prioritization steps

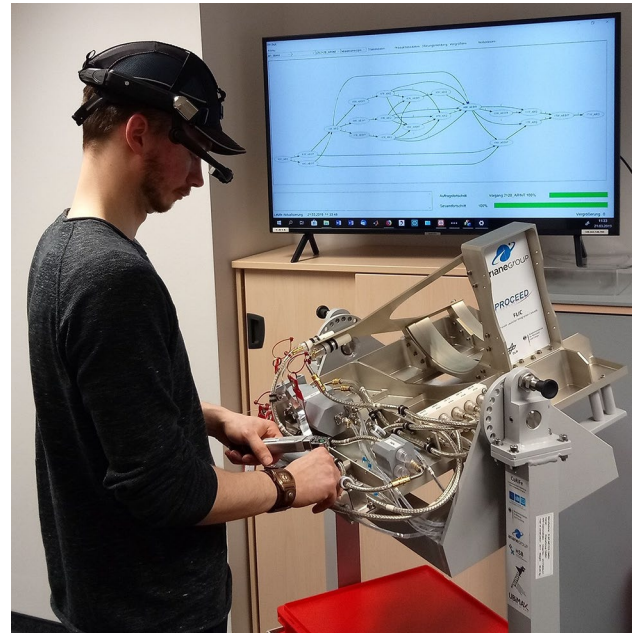| No.	Description
1	Repaired nodes and subsequent workable nodes
2	Workable nodes in the standard process
3	Alternative nodes
	3.1 Lowest penalty of the alternative edge
	3.2 Lowest process relevance
	3.3 Shortest execution time of the alternative node

Fig. 5 Set-up of a demonstrator using a component of the Ariane upper stage as sample assembly

the final order of the prioritized nodes and the release of the corresponding new work orders.

4 Application example

The concept of the Maximal Network Plan (MNP) has been developed and tested during the planning phase of the assembly plant for the Ariane 6 upper stage in Bremen, Germany. A demonstrator has been set up using operational data of a subassembly of the actual product (Fig. 5). The use of the MNP has been demonstrated, functioning as the back end of a novel planning system that provides a high dynamic especially in its reaction to substantial disruptions of the standard process, such as missing material and components. This high dynamic can only play to its full extent if it is combined with digital documents instead of static paper-based documentation. Therefore, in parallel mobile computing

devices to be used as mobile front ends have been selected and equipped with specific user interfaces to provide the required data as well as input abilities for collecting process data. The demonstrator consists of a database containing the MNP of the sample assembly and the corresponding algorithms. A status monitor is illustrating the overall process and its current state and is intended to be used by production planners. The worker wears a head mounted display that provides technical and organizational information on the current process step. Voice control allows to change menus and to provide input to the system.

The overall system is designed in a way that allows multiple users to communicate with the back end at the same time via a SQL-database (see Fig. 6). It can be distinguished between stationary and mobile users. Stationary users are located in an office and are responsible for planning and controlling of the overall process and administer and manage the overall set of data. They manage the overall network plan, determine the status of blocked and repaired nodes, approve the prioritization list and release work orders. Mobile users are workers and quality management staff on the shop floor. Their focus is a single process step that is currently being worked on. Whenever one of the mobile users has finished a process step and reported it accordingly via its mobile device to the planning system, the procedure described above of selecting the next due production step is being executed automatically. The corresponding operational data is provided on the mobile device. Also information on blocked nodes can be provided from the shop floor if the reason is related to the actual situation there. An extension of the system is feasible that holds an individual instance of the MNP for each product in the production line but manages a central pool of resources, especially working personnel with different qualifications.

As this project has been carried out before serial production has started, no operational data is available yet. The intention was to avoid the known difficulties, especially occurring in the production ramp up of a totally new

product, beforehand. The first flight of Ariane 6 is scheduled for 2022.

5 Discussion and further work

The aim of the Maximal Network Plan is to enable automated decision making in production planning systems also beyond the single sequence of a conventional network plan. The concept has been deployed in a use case scenario of a complex industrial environment. A number of advantages, but also limitations and further work have become visible.

The general concept allows to digitalize process knowledge to a much higher extent as it has been possible so far, thus extending automated decision making significantly. The planner can document far more than one single process sequence, especially in terms of locally limited workarounds in the case of missing resources. Furthermore, the MNP is intended to be a living and growing store of knowledge. Whereas network plans in typical user scenarios are only developed once and not modified very often when in use, the MNP can and should be refined further continuously, especially in terms of feasible alternative edges and their prioritization. New alternative edges are included based on manual decisions in a certain situation and are available for automated decisions in the future. Actual experience is being collected with executing alternative nodes and can be mirrored to the penalty of the corresponding alternative edge. Good alternative processes are rated better, bad alternatives are rated lower or even deleted. If boundary conditions change, alternative edges could even become the standard process.

An underlying assumption of this work is that a higher work load compared to the standard process is acceptable to achieve an overall progress. This is different from applications for example in mass production of consumer products, which on the other hand have a much more stable resource availability and therefore don't need the flexibility. It has

Fig. 6 Structure of the implemented demonstrator
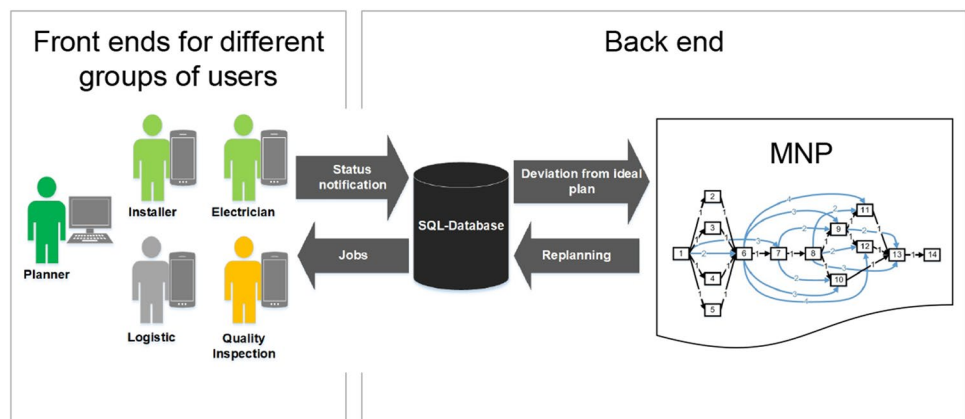
to be criticized that the penalty, represented by the edge weight of the alternative edges as seen in Fig. 3, so far has no relation to the real additional expenditure caused by circumventing the standard process. This can be low or even zero but range also very high. However, the actual amount of additional work is not known. Whereas it could be represented in the given data structure by the penalty being any positive value, it seems not practically feasible to manually calculate the actual amount for each possible alternative scenario. Further work therefore deals with the automated collection of process performance data and its reflection in the penalty of alternative edges. The actual process duration and work load of an alternative process in a specific scenario is available in the production data acquisition once it has been executed. This data can be used to specify the difference to the standard process and automatically improve the quality of the penalty information.

A further limitation should be seen regarding the prioritization steps provided in Table 2 as they have not been evaluated on large scale experiments outside of the application example above yet. Ongoing work deals with this issue. A test environment is being developed that allows the automated generation of large MNPs for imaginary scenarios and their automated evaluation by simulating their execution and typical user input. This is aiming to collect data on the performance of different prioritization strategies and their interdependence with boundary conditions such as the structure of the MNP, the pool of resources and the profile of the occurrence of disruptions.

Further work also deals with the automated collection of data in case of disruptions in the process. The aim is to identify patterns and eliminate the reasons systematically as far as they lie within the influential reach.

References

1. Rauch E, Vickery AR, Brown CA, Matt DT (2020) "SME requirements and guidelines for the design of smart and highly adaptable manufacturing systems," in Industrie 4.0 for SMEs, Cham, Switzerland, Palgrave Macmillan, pp. 39–72

2. Grendel H, Larek R, Riedel F, Wagner JC (2017) Enabling manual assembly and integration of aerospace structures for Industry 4.0-methods. Proced Manuf 1(14):30–37

3. Larek R, Grendel H, Wagner JC, Riedel F (2019) "Industry 4.0 in manual assembly processes–a concept for real time production steering and decision making," in 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, Naples, Italy

4. Cutting-Decelle AF, Young RIM, Michel JJ, Grangel R, Cardinal JL, Bourey JP (2007) ISO 15531 Mandate: a product-process-resource based approach for managing modularity in production management. Concurr Eng 15(2):2017–2235

5. Schleipen M, Drath R (2009) "Three-View-Concept for modeling process or manufacturing plants," in International Conference on Emerging Technologies and Factory Automation (ETFA), Palma de Mallorca

6. Yu B, MacCallum KJ (1996) "Modelling of Product Configuration Design and Management by Using Product Structure Knowledge," in *Workshop on Knowledge Intensive CAD*. Springer, Boston, pp 115–124

7. Günthner WA, Wilke M, Zäh MF, Aull F, Rudolf H (2006) "Produktion individualisierter Produkte," in *Individualisierte Produkte — Komplexität beherrschen in Entwicklung und Produktion*. Springer VDI-Buch, Wiesbaden, pp 63–87

8. Trunzer E, Calà A, Leitão P, Gepp M, Kinghorst J, Lüder A, Schauerte H, Reiferscheid M, Vogel-Heuser B (2019) System architectures for Industrie 4.0 applications Derivation of a generic architecture proposal. Prod Eng Res Devel 13:247–257

9. Hošovský A, Piteľ J, Trojanová M, Židek K (2021) "Computational Intelligence in the Context of Industry 4.0," in Implementing Industry 4.0 in SMEs Concepts, Examples and Applications, Cham, Switzerland, Palgrave Macmillan, pp. 27–94

10. Saniuk S, Saniuk A, Cagáňová D (2021) Cyber Industry Networks as an environment of the Industry 4.0 implementation. Wireless Netw 27:1649–1655

11. Ferreira F, Faria J, Azevedo A, Marques AL (2016) "Industry 4.0 as enabler for effective manufacturing virtual enterprises," in working conference on Virtual Enterprises, Porto, Portugal

12. Hamzadayi A, Yildiz G (2016) Event driven strategy based complete rescheduling approaches for dynamic m identical parallel machines scheduling problem with a common server. Comput Ind Eng 91:66–84

13. Papakostas N, Chryssolouris G (2009) A scheduling policy for improving tardiness performance. Asian Int J Sci Technol 2(3):79–89

14. Zhan Y, Qiu C (2008) "Genetic algorithm application to the hybrid flow shop scheduling problem," in 2008 IEEE International Conference on Mechatronics and Automation, Takamatsu, pp. 649–653

15. Chryssolouris G, Subramaniam V (2001) Dynamic scheduling of manufacturing job shops using genetic algorithms. J Intell Manuf 12(3):281–293

16. Chryssolouris G, Lee M (1994) An approach to real-time flexible scheduling. Int J Flex Manuf Syst 6(3):235–253

17. Rochow P, Burggraef P, Reuter C, Prinzhorn H, Wagner H, Schmitz T (2015) "Identification of alternative assembly sequences for large-scale products," in Production and Operations Management Society POMS 26th Annual Conference, Washington

18. Homem-de-Mello LS, Sanderson AC (1990) AND/OR graph representation of assembly plans. IEEE Trans Robot Autom 6(2):188–199

19. Crowston W, Thompson L (1965) Decision CPM: A Method For Simultaneous Planning, Scheduling And Control Of Projects. Massachusetts Institute of Technology, Cambridge

20. Zha XF, Lim S, Fok S (1998) Integrated knowledge-based assembly sequence planning. Int J Adv Manuf Technol 14:50–64

21. Gunji B, Deepak B, Bahubalendruni MR, Biswal B (2017) "Optimal assembly sequence planning towards design for assembly using simulated annealing technique," in International Conference on Research into Design

22. Kropik M (2009) Produktionsleitsysteme in der Automobilfertigung, Heidlberg. Springer Verlag, Berlin

23. Thulasiraman K, Arumugam S, Brandstädt A, Nishizeki T (2016) Handbook of graph theory, combinatorial optimization, and algorithms, Boca Raton, London. Chapman & Hall, New York

24. Jungnickel D (2013) Graphs, Networks and Algorithms, vol 5. Springer Verlag, Berlin, Heidelberg

25. Russell S, Norvig P (2003) Artificial Intelligence A Modern Approach, 2nd edn. Prentice Hall, Upper Saddle River

26. Diestel R (2017) Graph Theory, 5th edn. Springer, Berlin