Paper:

Towards Decentralized Production: A Novel Method to Identify Flexibility Potentials in Production Sequences Based on Flexibility Graphs

Lennart Bochmann[*1,*2], **Lars Gehrke**[*1,*3], **Adrian Böckenkamp**[*4],
Frank Weichert[*4], **Rainer Albersmann**[*5], **Christian Prasse**[*6], **Christoph Mertens**[*6],
Marco Motta[*6], **and Konrad Wegener**[*2]

[*1] Volkswagen AG
Berliner Ring 2, 38440 Wolfsburg, Germany
E-mail: lennart.soeren.bochmann@volkswagen.de
[*2] Institute of Machine Tools and Manufacturing, ETH Zurich, Zurich, Switzerland
[*3] Chair of Enterprise Logistics, Technical University of Dortmund, Dortmund, Germany
[*4] Department of Computer Science VII, Technical University of Dortmund, Dortmund, Germany
[*5] F/L/S Fuzzy Logik Systeme GmbH, Dortmund, Germany
[*6] Fraunhofer Institute for Material Flow and Logistics, Dortmund, Germany

[Received December 19, 2014; accepted April 13, 2015]

Due to higher degrees of individualization, shorter product life cycles, and volatile selling markets, fulfilling customer demands – the main task of automotive companies – has become very complex. In order to tackle this complexity, new concepts that enable the decentralization of decision-making within the production process have become promising solutions. The advancement towards self-organized production requires novel approaches in the field of production program planning. This work introduces the concept of the *volume cycle* as a new design factor in program planning. Additionally, a novel method to identify flexibility potentials in production sequences based on *flexibility graphs* is proposed, and the method is validated through a case study considering a segment of the assembly process for an automobile. Suitable visualization techniques for flexibility graphs are also discussed. Furthermore, in order to allow automatic analysis and evaluation of the flexibility potentials, methods of graph mining are introduced and the application possibilities of these techniques in terms of analyzing flexibility graphs are clarified. The results obtained from the case study illustrate that routing flexibility is not leveraged in today's production lines, thus revealing a potential optimization domain.

Keywords: decentralized production, flexibility potentials, modeling of flexibility graphs, volume cycle, Industry 4.0

1. Introduction

Nowadays, mass-market vehicles are commonly produced with a continuous flow manufacturing system that consists of a combination of highly efficient production and assembly lines. In a broader sense, all vehicles follow the same designated sequence of production steps until they are readily assembled. A centralized production planning and controlling approach enables a smooth production flow through the described system but requires significant investment in IT infrastructure and production equipment. Additionally, the approach is best suited for deterministic customer demands, which do not exist at all times and in all markets [2, 57].

New trends within the automotive industry, such as the market introduction of electric vehicles and various technologies in the field of Industry 4.0 [6], challenge the existing production systems. Specifically, these challenges include an increased need for flexibility through volatile selling markets, unforeseeable technological developments (e.g., in the field of electric vehicles), shorter production cycles, and the increasing individualization of products [16, 46, 48, 57]. Particularly with respect to volatile selling markets, investing in expensive production plants that are capable of producing several thousand vehicles per day, is economically infeasible because the return of investment is not guaranteed even on a mid-term basis. Therefore, alternative methods of planning, operating, and controlling smaller production sites by exploiting the potentials of modern information technologies have become more attractive to Original-Equipment-Manufacturers (OEMs) [32, 51]. A number of promising approaches using advanced technologies, with a significant potential for self-controlled complex systems, are currently developed in the scope of Industry 4.0 activities. Industry 4.0 is the focal part of the high-tech strategy of the German government and describes the technological evolution from embedded systems to Cyber-Physical Systems (CPS) [6, 21, 44]. More specifically, it subsumes the conjunction of all physical and virtual components within the value net of manufacturing. All real components are equipped with autonomous and decentralized communication and decision-making competences to fulfill customer demands. The main technological enablers for Industry 4.0 solutions are the Internet of Things (IoT) [24], Multi Agent Systems (MAS) [37, 42], and CPS [10].

With respect to industrial production, Industry 4.0 aims at realizing a one-piece flow ("lot size one") at the cost of mass production [21, 30], which would be suitable for addressing the challenges of future (European) manufacturing markets [19]. Following the vision of Industry 4.0, these challenges could be resolved through the concept of smart factories. These factories are scalable and operate at high flexibility through decentralized control strategies, like MAS approaches for material flow and program planning, and the utilization of technologies such as the IoT and CPS [6, 58]. Within the scope of these smart factories, the decentralized program planning approach as well as the application of CPS to assembly tasks have to be explored. An approach towards decentralized production planning is introduced in this paper.

The main objective of this study is to provide the vision of Industry 4.0 with a tool box of planning methods for developing a specific profile. Therefore, this work focuses on:

- the definition and introduction of a *volume cycle*, which replaces the traditional sequence, to create a degree of freedom for the operation of CPS on the shop floor and
- the development of a concept to quantify the flexibility potential by analyzing an abstraction of the assembly dependencies using *flexibility graphs*.

The proposed *volume cycle*, which allows a temporal sequence of orders and does not require a fixed production sequence, was identified as a promising solution to the intended production planning approach. Furthermore, the degree of flexibility concerning the potential to "open" strict production sequences has to be investigated. The concept of a new methodology was thus developed to qualify and quantify this potential.

The application of the proposed methods and therefore the identification of the flexibility potential of production sequences entail a number of advantages. For industries in general and the automotive industry in particular, the utilization of flexibility potentials can maximize the profits of the production schedules. Additionally, the implementation of a decentralized planning and control approach constitutes a convenient way to manage the complexity that arises in individualized industrial production [12]. By mastering this complexity, the quantity and effects of down-times in production lines can be reduced. Consequently, shorter delivery times can be realized for the customers. Furthermore, the enhanced flexibility enables OEMs to allow customers to adjust their existing orders during the production process. This new flexibility leads to operational improvements and will have a positive effect on customer satisfaction. From a technical point of view, decentralized control approaches enable scalability of production and material flow systems in an economical manner [43].

The remainder of this paper is organized as follows. Section 2 describes the existing State-of-the-Art technologies of current (centralized) production systems. Additionally, precedence diagrams are reviewed briefly, as they constitute the standard technique for analyzing (centralized) production lines. In Section 3, the idea of future program planning is explained by introducing the terms *flexibility graph* and *volume cycle*. This is exemplarily supported by a process analysis of the (pre-)assembly lines at an OEM within the automotive industry. Section 4 presents a graph-based representation of the production as well as a formalization of the flexibility graphs. In addition, visualization concepts for the (flexibility) graphs are surveyed. In Section 5, the analysis of the flexibility graph is explained to elaborate a deeper understanding of the dependencies of the required (pre-)assembly process steps. Finally, Section 6 concludes with a summary of the main results and presents an outlook regarding further research.

2. Related Work

In order to optimize production planning and control, the flexibility potentials of current production processes need to be investigated. Ten different types of flexibility are defined in the literature [9, 18, 50]. In the context of this work, the routing flexibility is the most relevant type of flexibility, and it is also a topic addressed frequently in visions of Industry 4.0. It describes the ability of a manufacturing system to produce a part by alternative routes through the system [50].

To leverage the routing flexibility of a production system, the flexibility potentials need to be known during the planning phase. Different planning techniques exist, and a general overview is shown in **Fig. 1**. At the current stage of our research, the production times of the considered manufacturing operations are assumed to be deterministic. A common method to determine the production times of manual operations is the Methods-Time Measurement (MTM) [40]. In this work, the Precedence Diagram Method (PDM) is used, as indicated in **Fig. 1** [25, 38, 45]. The PDM belongs to the category of Critical Path Methods (CPM) and is characterized by the representation of actions using nodes and dependencies using arrows connecting the nodes (task-on-node representation) [3, 7]. The advantages of a precedence diagram are its simplicity and avoidance of redundancies [36]. Furthermore, the routing flexibility can be represented by buffer times [38]. The Program Evaluation and Review Technique (PERT) [3, 4, 15] is another planning method that allows the determination of the critical path and flexibility potentials within a process chain. By assuming determined activities, the requirement for the application of PERT, i.e., probabilistic activities, is not met. Therefore, PERT cannot be utilized for the analysis within this work. This work introduces a method that expands the

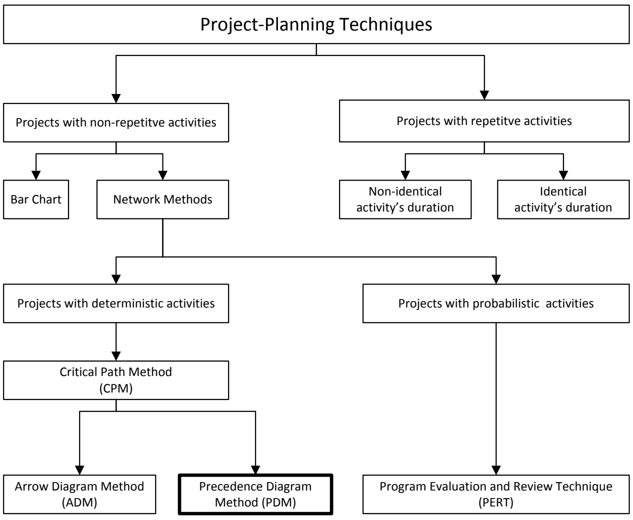


Fig. 1. Overview of existing planning techniques [3].

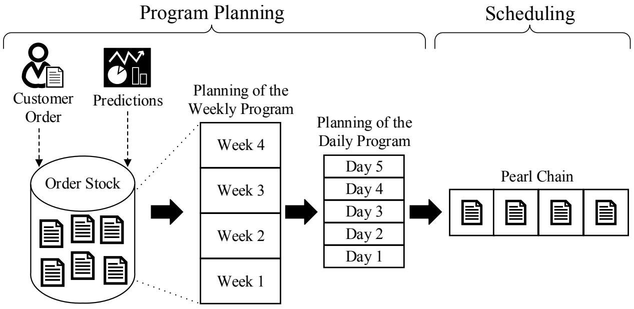


Fig. 2. Program planning and sequencing in the automotive industry (based on [8]).

static graph derived from the application of PDM to a dynamic graph. Additionally, semantic groups are added to the dynamic graph in order to investigate specific characteristics of the graph, in this case the routing flexibility, in more detail.

An overview of the currently applied techniques for production program planning and scheduling in the context of the automotive industry is given in [11, 27, 28, 47]. The production program planning and scheduling is part of the order-to-delivery process, which is defined by the following [23, 55]:

- program planning and capacity management,
- order management,
- production program planning and scheduling,
- production, and distribution.

The process of program planning is a multi-level process with different time horizons [33]. Three different perspectives are distinguishable in practice: strategic program planning (long term), tactical planning (medium term), and operative planning (short term). The following explanations only focus on the production program planning and scheduling shown in **Fig. 2**, in which a short-term planning interval is considered. The program planning can be subdivided into two steps: the planning of the weekly program and the planning of the daily program. These two steps are followed by the scheduling, which organizes the orders from the daily program into a sequence that can be illustrated as a pearl chain, i.e., the different orders flow through the production in the designated sequence.

On a monthly basis, the order stock, which holds the customer orders as well as the predictions from the markets, is partitioned into weekly portions. Therefore, a target week is allocated for each customer order on basis of the delivery date demanded by the customer. In this case, the customer may be the end customer, the market, or a particular dealer. The horizon for this planning step can vary depending on the OEM. In **Fig. 2**, the horizon is depicted as four weeks. At this point, a linking between the order, plant, and capacities should have already taken place. The planning of the daily program is a further fine-tuning of the weekly program, in which all the technical and capacitive restrictions of the plant and the production line are considered and the orders are related to a concrete shift on the already planned production day.

The latter part of the whole process is the scheduling. Here, the orders of a shift are sequenced considering all the technical and capacitive restrictions. These are the same rules that have to be applied in the program planning as well. Otherwise, the order mix of the weekly and daily portions could be incompatible to the conditions of the related production line.

For instance, the distance and number of item restrictions for certain properties (equipment features) are defined in this step. A distance restriction defines that a certain number of orders excluding a specific property has to be produced in between two orders including this property. An example could be that between two Right-Hand-Drive (RHD) vehicles a certain number of non RHD vehicles must be produced. An item restriction defines the number of orders including a specific property (e.g., air conditioning) that have to be manufactured during the day. In some cases, up to 100 different restrictions or characteristics are defined during the fine planning process, which have to be considered later during the sequencing process. The fine planning process normally results in a so-called Just-In-Sequence (JIS) release order placed at the component manufacturers. The plant issues the JIS sequence orders a certain number of days (empirical value: 6 to 10 days) before the scheduled start of production. The corresponding JIS deliveries of the different suppliers are scheduled exactly in the same sequence, in which the orders have been scheduled on the production lines.

3. Program Planning for Industry 4.0

This section introduces a new concept for future program planning of decentralized production structures. In Subsection 3.1, the concept and its relevance to the vision of Industry 4.0 are clarified. The knowledge about the flexibility potentials of the production processes is the foundation of the new methodologies for a future decentralized production planning and control approach for smart factories. A process mapping of the current assembly processes at an OEM is shown in Subsection 3.2 and used as a case study within this work.

3.1. Concept for Future Planning Tasks

The concept of program planning and scheduling described in Section 2 can be found with variations at almost every car manufacturer operating with continuous flow production structures. In terms of Industry 4.0, the advancement towards decentralized production implies a change from fixed, scheduled flow production to a flexible network of process cells that can be easily rearranged. The overall capacities of the production system and its resources have to be synchronized with the customer demand, and the supply of the process cells requires dynamic logistic units, e.g., Automated Guided Vehicles (AGV). Therefore, the decentralized production concept necessitates new and tailored production planning and scheduling approaches.

In the future, it will be impossible to predict the exact point in time when a specific production step will start or end because of the decentralized decision-making process. This leads to the problem of providing the assembly cells with the demanded parts and required product and production information on time. Even though a production plan indicates the day a specific order has to be produced, the exact hour of value adding is a dynamically determined result of the decentralized production control (depending on the assembly cells and the material flow). Because the assembly sequences of JIS parts can be dynamically reorganized by a decentralized production control, the exactly scheduled JIS part supply, synchronized to a specific production sequence, can no longer be done according to the traditional patterns.

To resolve this issue, the present work introduces an enlarged time slot for JIS part deliveries, called the *volume cycle* (cf. **Fig. 3**). The *volume cycle* is conceived as a non-sequenced amount of production orders that is allocated to be produced within a certain time frame of the production plan. Thus, the *volume cycle* is a result of the production planning and is used to pass orders from the production planning side (from the order stock) to the production control side (into the order pool). It is flexible with respect to the number of orders it can contain and the time frame the production plan provides for the actual production process. Orders within a *volume cycle* have no longer a particular sequence like in traditional production plans (pearl chain). The production control side determines the *volume cycles*, i.e. the possible number of orders, by allocating the actual capacities and resources of the production system. This information is then communicated to the planning side to ensure that the *volume cycles* are planned according to the actual situation on the shop floor.

In the production planning process, pre-filled volume cycles have to be determined based on of real-time data from the production system such as capacities and resources. The quality of the planning result depends on the knowledge about the configuration and capabilities of the actual production system. The basic interrelation between the production and the assembly steps could be visualized by a *flexibility graph*, as described in Section 4. Because of the flexibility within the production system, the planning process has to incorporate simulation methods to predict future states with high probabilities.

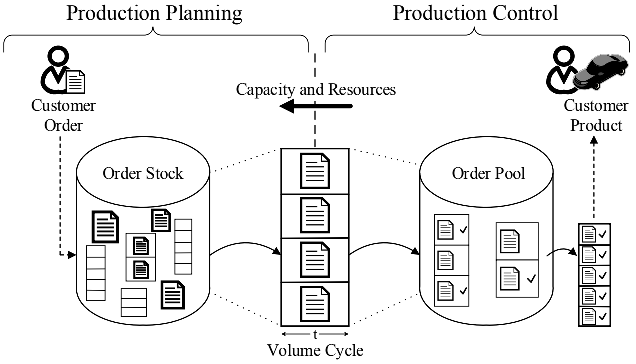


Fig. 3. Visualization of the suggested concept of the *volume cycle*.

In order to exploit the benefits of the decentralized program planning and scheduling concept, Subsection 3.2 gives an overview of the flexibility potentials within the production process of OEMs.

3.2. Process Mapping

During the phase of process mapping, the different steps of the production process at an OEM plant are mapped manually. This process mapping constitutes the basis for the subsequent process analysis. The final assembly line as well as the different pre-assembly lines are within the scope of the process mapping phase, as shown in **Fig. 4**. In this way, differences between the production and assembly areas can be identified, and more detailed statements can be made.

As a matter of principle, the availability of process information for the subsequent analysis grows with the number of process steps mapped. Hence, the room for flexibility increases. The data and information regarding the different process steps are taken from a knowledge base of the OEM and will be verified by assessments on the shop floor. The available degree of detail ranges from detailed process step levels (e.g., assembly instruction for single parts) to general descriptions of the different production areas.

In this study, the level of detail was set to the description of the process steps (e.g., Step X: "Assembly of headlights") to enable a significant analysis. Eventually, a list of all the process steps evaluated in the analysis has to be generated. Additionally, time-wise information, such as the lead time or the required time for transportation between two steps, has to be available for each process step. Logistical information, such as the level of inventory or the type of material staging at the production sections,

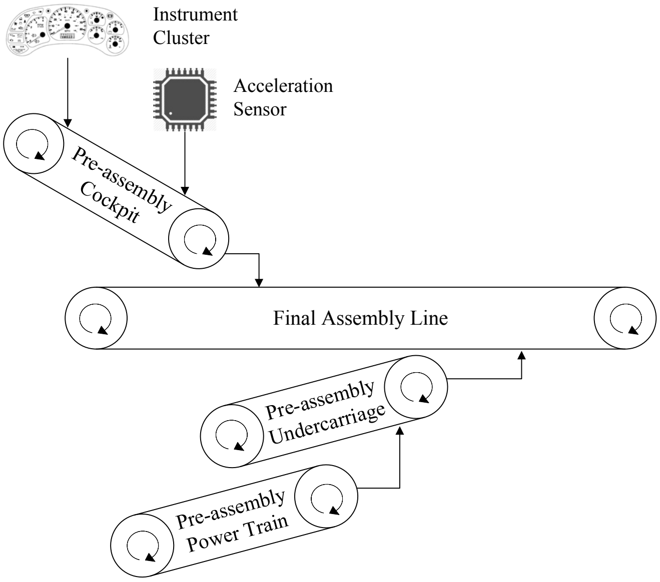


Fig. 4. Production areas considered for the analysis.

is collected as well. The large amount of specific information and the mapped production processes of the OEM constitute the starting base for the development of flexibility graphs (see Section 4) and for the subsequent process analysis (see Section 5).

4. Flexibility Graph for the Novel Program Planning

The information from the process mapping described in Section 3 is analyzed in this chapter. As a first step, the PDM is applied to highlight the dependencies in the process order (see Subsection 4.1). As the PDM is executed manually, a method to model *flexibility graphs* automatically is introduced in Subsection 4.2 and exemplarily applied to the considered case study. Furthermore, visualization techniques for *flexibility graphs*, which are necessary to face the complexity of the process structure, are presented in Subsection 4.3.

4.1. Precedence Diagram

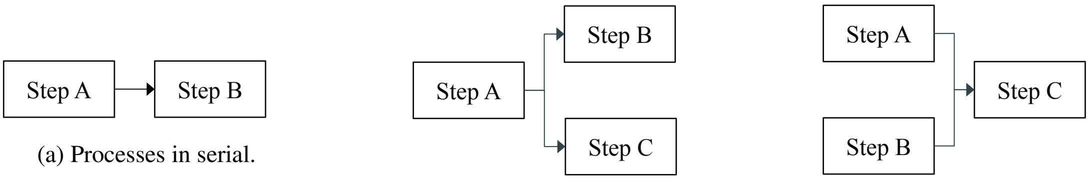
The main objective of the process analysis is to derive the routing flexibility potential with respect to the sequence of production steps. Browne et al. [9] introduced routing flexibility as "the ability to handle breakdowns and to continue producing the given set of part types." ElMaraghy [18] adapted this definition and explained the routing flexibility as the "number of feasible routes of all part types/number of part types." A precedence diagram describes the predecessor-successor relationships and is therefore useful for describing the relationships and coherence between the production steps as well [56]. Therefore, this method was found to be particularly useful for the modeling and analysis of production processes and is a tool capable of achieving the mentioned main objective. Precedence diagrams are widely used in industrial contexts, both in theory and in practice [43, 45]. For example, once the predecessor-successor relationships for a set of production steps are identified, a variety of analyses (e.g., critical path analysis) can be performed on the set of steps. A number of these analyses are described below and applied to the relevant production processes. Further analysis methods based on precedence diagrams can be found in [36]. To ensure proper understanding of precedence diagrams, it is important to clarify that only dependencies in the production order between processes are included and every process step shown in a precedence diagram needs to be performed in order to manufacture a product.

The listed production steps (see Subsection 3.2) are iteratively transformed into a precedence graph by the following method: For each production area, i.e., cockpit pre-assembly, power train pre-assembly, and final vehicle assembly, the direct predecessors and direct successors of every production step are identified. Based on this identification, which is commonly done on a spreadsheet and is termed the precedence matrix, a precedence graph as shown in **Fig. 5** can be derived. Additionally, the different production steps, which are portrayed as boxes or "knots," are connected by arrows pointing in the direction of the succeeding processes. Therefore, an arrow from the box of process step A to the box of process step B indicates that A has to be completed before B can start or B can only start if A is completed (**Fig. 5(a)**). In case an arrow points from process step A to process steps B and C and if there are no other arrows, then process steps B and C run simultaneously and can start once process step A is completed (**Fig. 5(b)**). This also means that, if the arrows of A and B both point to C, then process step C can only be executed after the completion of process steps A and B (**Fig. 5(c)**).

Additionally, using the lead time of each production step, the time aspect can be incorporated into the analysis, i.e., once the duration (D) of every production step is available, the total lead time can be calculated as the sum of the durations of all the process steps. In a complex situation, such as the production of an automobile, many processes can flow simultaneously and have different start and end points. In project management, it is common to identify the critical path in the previously mentioned precedence diagram. The critical path is the sequence of process steps that has the longest total duration. This path is said to be critical because a delay of any lead time included in the path will have an effect on the total lead time of the observed processes. On the other hand, a reduction of any lead time on the critical path will cause a reduction in the total lead time [54].

4.2. Modeling the Flexibility Graph

This section focuses on the representation of process steps and their dependencies in terms of graph theory. By employing graph algorithms, the possibly complex precedence diagrams (see Section 4.1) can be analyzed and visualized (semi-) automatically. Thus, in this section, the



(a) Processes in serial.

(b) A serial of a single process followed by two simultaneous processes.

(c) A serial of two simultaneous processes followed by a single process.

Fig. 5. Process examples that clarify the logical relationships within a precedence diagram.

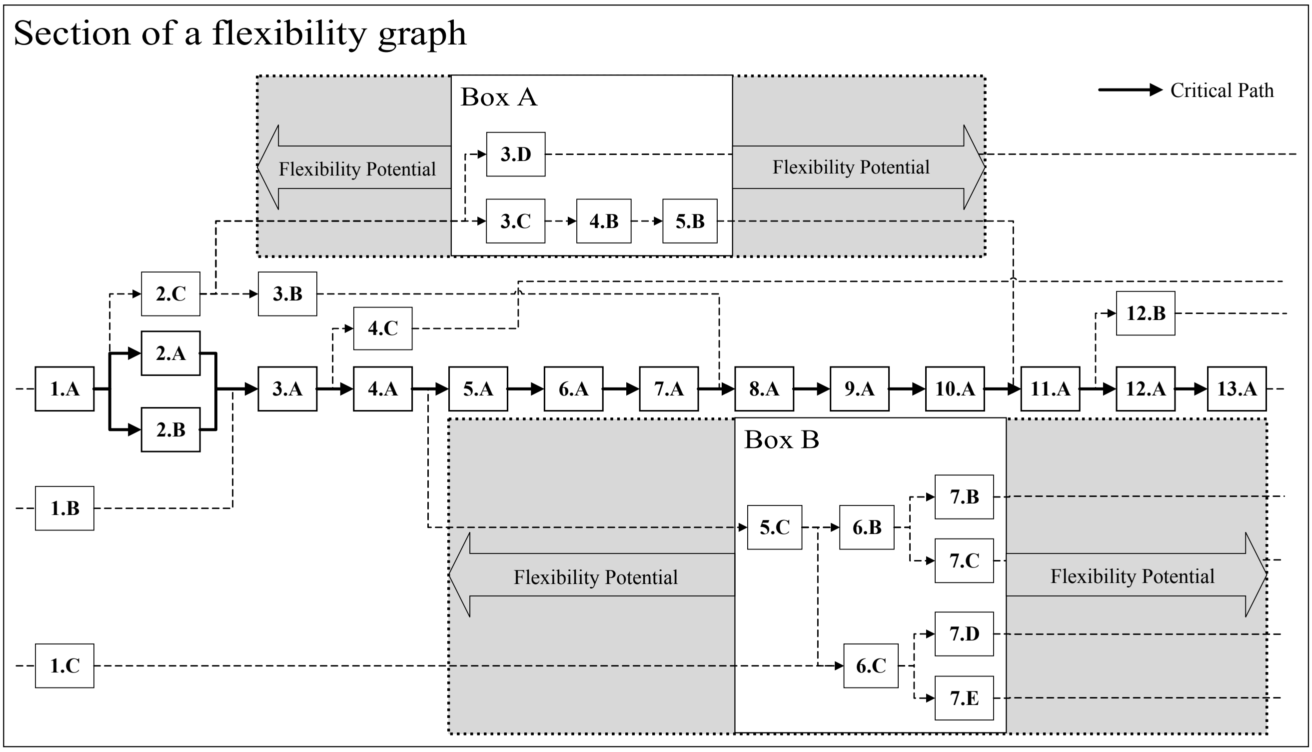


Fig. 6. Excerpt of an exemplary graph (segment of a production process) indicating the complexity and demonstrating the flexibility potential with respect to the process steps in boxes A and B; the critical path is visualized as a solid line. It is important to note that the vertices represent the process steps of an automobile (pre-) assembly and the edges represent the dependencies of these steps.

preliminaries of graph theory and required formalisms are presented so that precedence diagrams can be modeled as graphs.

Precedence diagrams, as discussed in Section 4.1, may be modeled as a directed weighted (and connected) *graph* $G = (V, E)$, where the set V of vertices equals the process steps and the set $E \subset V \times V$ of edges specifies the dependencies of each process step. It is important to note that the edges do *not* (necessarily) represent the flow of material. Thus, an edge $e = (u, v) \in E$ with $u, v \in V$ requires u to be completed before v can start. As each process step $v \in V$ is annotated with additional information gathered during the process mapping, the vertices have associated weights $w_i(v)$. For example, the weights may denote the workload level of a machine in the production flow, which in turn may be exploited to dynamically select the least used machine when multiple alternatives are available in a specific production step. Because of the introduction of weights, further information is incorporated in the precedence diagram. As this work utilizes the incorporated information mainly to determine the routing flexibility, it is proposed to name this type of graph a *flexibility graph*.

Figure 6 depicts an exemplary excerpt of a *flexibility graph*, which models a partial aspect of an automobile assembly. Note that all the process steps shown need to be passed through to complete the modeled assembly excerpt. This means that, even if some of the process steps may be executed in parallel (e.g., processes 2.A and 2.B), all the processes must still be carried out finally. The process steps condensed in boxes A and B can be executed in parallel to the process steps of the critical path (shown as a solid line). Moreover, they may even be delayed, which is visualized by the gray dashed boxes. For example, process 5.C must not start before 4.A and must be completed before process 13.A has finished. Note that the additional information (collected in the process mapping) for each vertex is not shown here for the sake of readability. This information is taken into account to compute the final weight $w(v)$ of each vertex for the optimization algorithms applied on the graph.

The vertex weights are transformed into weighted edges because many well-known graph algorithms work on edge-weighted graphs only, e.g., [17, 26]. This is achieved by replacing each weighted vertex v with two new synthetical vertices v_1 and v_2 connected by a new edge $(v_1, v_2) \in E$ having the weight $w(v)$. All preexisting (i.e., non-weighted) edges in G are set to some constant weight $c \in \mathbb{R}^+$. Note that the possible execution orders of the process steps can simply be determined by using topological sorting on the graph [31].

4.3. Visualization of Flexibility Graphs

With increasing complexity of the graphs, automatic visualization and layout algorithms are beneficial to support the visual inspection of the production structure. Thus, efficient layout algorithms [5, 53] are needed; examples will be presented in the following. Note that graph features, like the centrality indices, which will be presented in Subsection 5.2, can simply be visualized by mapping the centrality values of a vertex to a color scheme.

Visualization concepts can be categorized into (pure) algorithmic and declarative approaches. Algorithmic approaches are designed to solve a specific optimization problem like reducing the edge crossings or the size of the required canvas [52]. On the other hand, declarative approaches incorporate use-case related aspects such as concurrency or non-determinism of process steps [1, 29].

Sugiyama et al. [52] described a hierarchical algorithm to draw directed graphs in a layered manner so that the graph acts "as a visual aid to understand overall images of the structure of the [modeled] complex system." The algorithm assumes a two-dimensional surface onto which the graph should be drawn, i.e., each vertex $v \in V$ is mapped to a position $(x,y) \in \mathbb{R}^2$ – **Fig. 7** shows a small example with 12 vertices. The algorithm can be outlined as follows: In the first step, appropriate edges, which make the graph cyclic, are inverted. This effectively makes the graph acyclic (cf. **Fig. 7**); however, as no cyclic graphs are expected within the context of (pre-) assembly flow modeling (cf. Section 5.2), this step may be omitted. The next step assigns each vertex a horizontal layer, which determines the y-coordinates. To achieve this, a repetitive two-step procedure is pursued: Firstly, all sinks are placed in the lowermost layer L_1 (layer 1, see **Fig. 7**). Secondly, all sinks are conceptually removed, and the procedure continues again with the first step, except the new sinks are placed in the next layer (e.g., L_2 in case of the second iteration). This effectively places a vertex v in the layer L_{i+1}, where i denotes the length of the path from v to a sink having the largest possible number of edges. The next step of the algorithm aims at reducing the number of edge crossings. Therefore, dummy vertices are inserted for all edges that cross multiple layers. If this step is omitted, the reduction of edge crossings would only consider edges crossings pairwise between two layers. Using the dummy vertices, the algorithm analyzes the layers of the graph pairwise (starting at L_1). Given two layers L_i and L_{i+1}, the vertices in layer L_{i+1} are sorted based on the number of crossings they induce with respect to all the other vertices on the same layer. In this manner, the vertices in L_i are fixed, and those in L_{i+1} are re-arranged so that the edge crossings between L_i and L_{i+1} are reduced. Finally, x-coordinates are assigned to the vertices by minimizing the number of bucklings at the dummy vertices. This is accomplished by quadratic programming. Note that during the last step, the vertices are not re-arranged anymore. Further details on this algorithm may be found in [52]. This approach is specifically useful for the visualization of Directed Acyclic Graphs (DAG); Section 5.2 presents more details.

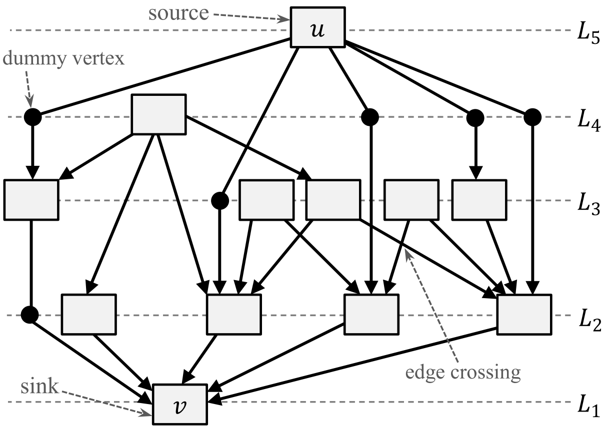


Fig. 7. Exemplary result of applying the hierarchical visualization algorithm by Sugiyama et al. on a small flexibility graph (labels of the vertices have been dropped for the sake of readability); notice the 5 *layers* $L_1,...,L_5$ on which the vertices are aligned. The figure also illustrates the terms *source*, *dummy vertex*, *edge crossing*, and *sink* (refer to [49, 52] for more details). The graph shown is *acyclic* because it does not contain any cycles; it would be *cyclic* if, e.g., it had an edge (v,u).

It should be noted that visualizations with non-intersecting edges (so-called *planar graphs*) are the most favored because of their improved readability. The Open Graph Drawing Framework (OGDF) provides tools for rendering (planar) graphs [14].

5. Revealing Flexibility Potentials by the Analysis of Flexibility Graphs

Section 4 already provided the fundamentals of mapping and modeling the process steps of an (pre-) assembly line along with their properties. These include the precedence diagrams and the flexibility graphs. Subsection 5.1 employs the concept of precedence diagrams to describe a manual analysis for computing the *flexibility potential* based on various timing constraints. Subsection 5.2 extends the analysis by using flexibility graphs (introduced in Subsection 4.2) to quantify, for example, the amount of parallelizable process steps or to automatically determine the critical paths.

5.1. Manual Analysis of Precedence Diagrams

Critical paths are determined for each production area (i.e., cockpit pre-assembly, power train pre-assembly, and final assembly). In order to identify the path, the starting and final process steps were defined for each of these areas. After identifying the critical path within each production area, the flexibility potentials for all the process steps can be determined. It should be noted that flexibility potentials may never be found along the critical path. By providing information about the process, the lead times, and the critical path, the earliest start (*ES*), latest start (*LS*), earliest finish (*EF*), and latest finish (*LF*) can be determined using

$$EF_i = \max\left(ES_{i-1}\right) + D_i \quad \text{. . . . .} \quad (1)$$

$$LS_i = \min\left(LF_{i+1}\right) - D_i \quad \text{. . . . .} \quad (2)$$

where $i = 1,\ldots,n$ and n is the length of the critical path [36]. Notice that ES and EF are calculated working forwards, whereas LS and LF are calculated working backwards. Finally, the flexibility potential of each process step can be calculated using

$$FP_i = LS_i - ES_i = LF_i - EF_i \quad \text{. . . . . . .} \quad (3)$$

The resulting description for each process step is shown in **Fig. 8**.

Based on the gathered information about the process relationships, the total flexibility potential FP_{tot} of the considered process chain is determined by applying

$$FP_{\text{tot}} = 1 - \frac{n_{\text{crit}}}{n_{\text{tot}}} = \frac{n_{\text{noncrit}}}{n_{\text{tot}}} \quad \text{. . . . .} \quad (4)$$

where n_{tot} is the total number of considered processes, n_{crit} is the number of processes in the critical path, and n_{noncrit} is the number of process not belonging to the critical path. The values of n_{tot}, n_{crit}, and n_{noncrit} are obtained from the critical path analysis of the conducted case study and summarized in **Table 1**. The information obtained by applying CPM is necessary for the calculation of the flexibility potential.

Thus, for the process areas considered, the flexibility potential is determined to be $FP_{\text{tot}} = 76$

- FP_{low}: Low flexibility potential, if $1 \leq FP_i \leq 5$.
- FP_{medium}: Medium flexibility potential, if $6 \leq FP_i \leq 10$.
- FP_{high}: High flexibility potential, if $FP_i \geq 11$.

The additional categorization shows that 47% of the processes have a low flexibility potential FP_{low}, 18% have a medium flexibility potential FP_{medium}, and 11% have a high flexibility potential FP_{high}. A visualization of the distribution of the flexibility potentials is shown in **Fig. 9**. It is important to mention that the categorization only groups the investigated flexibility potentials into three (almost) equally sized categories.

<table>
<tr><td>Earliest Start
(<i>ES</i>)</td><td>Duration
(<i>D</i>)</td><td>Earliest Finish
(<i>EF</i>)</td></tr>
<tr><td colspan="3">Part Name / Activity</td></tr>
<tr><td>Latest Start
(<i>LS</i>)</td><td>Flexibility Potential
(<i>FP</i>)</td><td>Latest Finish
(<i>LF</i>)</td></tr>
</table>

Fig. 8. Resulting process description, adapted from [36].

Table 1. Results of the critical path analysis applied on the conducted case study.

Category	Quantity
n_{tot}	79
n_{crit}	19
n_{noncrit}	60

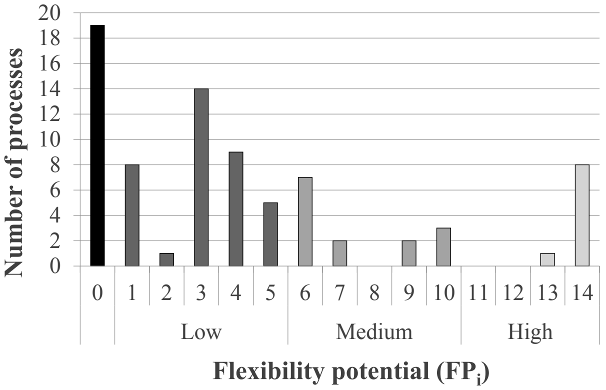


Fig. 9. Histogram of the flexibility potentials in the conducted case study.

The results obtained from the analysis of the routing flexibility show a significant potential, which can be used in the dimensioning and designing of reconfigurable manufacturing systems (RMS) [34, 35, 41], i.e., the process steps that have a certain flexibility potential could be grouped to form a reconfigurable manufacturing station. In this manner, a production system consisting of non-dedicated machines could be designed. Moreover, the gained flexibility enables the system to face the complexities that arise from higher degrees of individualization, shorter product life cycles, and volatile selling markets. Therefore, RMS constitute one of the key research topics related to the German high-tech strategy Industry 4.0.

5.2. Graph Mining

Graph features [13] can be used to explore the structural coherence of a graph, which could in turn reveal the importance and interaction of the process steps (modeled as graph vertices) described in Subsection 4.1. In this manner, a graph analysis could help in (semi-) automatically identifying the flexibility mentioned in Subsection 5.1. As these graphs can have a large number of vertices (5000 and more), efficient algorithms are needed.

The most widely used graph features are *centrality indices*, which were originally used in the analysis of social networks. Centrality indicates the importance of the vertices in a graph [13]. For example, within the context of this paper, centrality can be considered as the significance of a machine in the production flow, provided the machine is modeled using graph vertices. Given a vertex $v \in V$, the *degree centrality* equals the number of adjacent edges tied to v. The degree of a vertex can be further classified into *in-* and *outdegrees*, where the indegree equals the number of incident edges and the outdegree corresponds to the number of outgoing edges. A high indegree centrality requires many process steps to be joined, whereas a high outdegree allows the parallel execution of all the process steps that are connected through an outgoing edge. Additionally, vertices with outgoing edges can be started independently of other processes.

Closeness centrality builds up on the concept of shortest paths. A *shortest path* from a vertex $v \in V$ to another vertex $u \in V$ is defined by the sequence of edges that connects the two vertices such that the sum of the weights is minimized (with respect to all possible paths between v and u). The *distance* from v to u is then given by the number of edges in their shortest path. If there is no path from v to u (e.g., because v is executed after u), the distance is defined to be zero. The closeness centrality of a vertex v is the reciprocal of the sum of distances to all other vertices $u \in V \setminus \{v\}$. Thus, process steps that have to be executed at the beginning of the assembly flow have the smallest closeness centralities not equal to zero. Thus, such processes can be identified by estimating the values of closeness centrality, which can be considered as an indicator of the relevance within the production flow. The final process steps without any successors are defined to have a closeness centrality of zero. For example, these may include cleaning and/or checking the assembled car. Furthermore, process steps that require multiple assembly steps to be completed beforehand (i.e., they are located at a "later" stage in the assembly flow) have higher values of closeness centrality compared with the processes that must be executed first.

Betweenness centrality refers to the importance of a vertex v with respect to the shortest paths across v [13]. Specifically, let $\#_{st}$ be the total number of shortest paths from $s \in V$ to $t \in V$ and let $\#_{st}(v)$ be the number of shortest paths from s to t that pass through v. Adding up the fraction $\frac{\#_{st}(v)}{\#_{st}}$ for all vertex pairs $s,t \in V$ with $s,t \neq v$ yields the betweennes centrality of v.

Both closeness centrality and betweenness centrality require to solve the All-Pairs-Shortest-Path (APSP) problem, which can be done using the Floyd-Warshall algorithm [20]. However, single shortest paths may be of interest as well: Consider a process step v that requires a specific apparatus for its completion. Assume further that the apparatus has, for example, a high failure rate or requires maintenance soon. Completing some pre-assembled products (located at some process step u) before the next maintenance that only need to pass through the processing of the aforementioned apparatus is desirable. For example, this may be required for the adherence to a deadline in order to avoid contractual penalties or a customer's urgent order. In this case, computing a single shortest path between u and v in terms of a weighting function that possibly combines many criteria can be done with the *Dijkstra algorithm* [17] or its (typically faster) extension – the *A* algorithm* [26]. Note that there are also algorithms available that tackle the multi-criteria shortest path problem (SPP) [39]. The multi-criteria SPP involves the consideration of many criteria during the optimization in the algorithm, whereas, for example, the Dijkstra algorithm only considers weights attached to edges. Clearly, multiple criteria can be mathematically combined into a single weight and thus be solved by the Dijstra algorithm. Given two vertices u (source) and v (sink), the Dijkstra and A* algorithms roughly proceed as follows: Initially, the distance $d(u)$ of u is set to zero, and the distances of $v \in V, v \neq u$ are assigned ∞. At first, all the neighbors $t_i \in N(u)$ of u are visited, and their distances are set to $w(u,t_i) + d(u)$. Recall that $d(u) = 0$ because u is the source vertex. The successor of u is stored as the neighboring vertex with the smallest distance, i.e.,

$$\min_{\forall t_i \in N(u)} d(t_i). \quad \ldots\ldots\ldots\ldots\ldots \quad (5)$$

The algorithm then repetitively examines the non-visited neighboring vertices of the previously selected vertex t_i and selects its successor similarly. Note that the "distance" is computed by adding up the distance of the previous vertex and the weight of the connecting edge. If the algorithm encounters a visited vertex t that stores a previously-assigned larger distance $d(t)$, t is updated to reflect the smaller distance. The procedure terminates when v is visited. This approach is extended by the A* algorithm, which utilizes a heuristic to estimate the most-promising direction of search. As in the Dijkstra algorithm, each vertex stores its current minimum distance to the source vertex u. The heuristic $\delta(t)$ estimates the remaining distance from a neighboring vertex t to the final vertex v by

$$\delta(t) = \alpha(t) + \beta(t) \quad \ldots\ldots\ldots\ldots \quad (6)$$

where $\alpha(t)$ denotes the current distance to t and $\beta(t)$ estimates the distance from t to v. For the algorithm to work properly, $\beta(t)$ must not overestimate the distances [26]. Clearly, this heuristic depends on the type of weights attached to the graph. For example, assume each process step has a single associated processing station and the weight of each edge $e = (s,t)$ states the real distance from the s-station to the t-station. A "shortest path" then minimizes the transportation route, and $\delta(t)$ can be set to the distance of the beeline from t to v. In this manner, the directed edges maintain a valid order of the production flow while the transportation costs are minimized. Within this context, the previously explained betweenness centrality reflects the importance of the vertices (i.e., process steps) on a shortest path from u to v that need to be passed through when the earliest possible completion of v is required.

As mentioned in Subsection 4.1, critical paths are of particular interest because delays during the process steps in such paths lead to delayed delivery. In terms of graph theory, critical paths may be understood as the problem of finding the *longest path* in a graph between two vertices. When considering automotive assembly processes, the incorporation of customer wishes *during* the assembly is worthwhile to be as late as possible. Process steps that may be affected by spontaneous customer wishes should therefore be started as late as possible, which in turn demands longest paths to such vertices. Because the flexibility graph describes the dependencies of each process step, it cannot contain any cycles. Accordingly, the assumption of a DAG is suitable; see Subsection 4.3. The longest path G in a DAG can be by found by determining the shortest path in $\overline{G}$ obtained by negating all the weights [49].

Finally, it should be mentioned that there are algorithms for property testing [22], i.e., to check whether a given graph has a specific property. With respect to the longest path problem, one may be interested in testing whether a graph is actually a DAG.

6. Conclusion and Outlook

To achieve the goal of identifying flexibility potentials in decentralized production sequences, a novel approach based on the concepts of *volume cycle* and *flexibility graphs* was introduced. The proposed method enables the identification of flexibility potentials in manufacturing processes in terms of routing flexibility. *Flexibility graphs* provide a tool to analyze the dependencies in the production sequence. The results obtained from the case study clarify that the routing flexibility is not leveraged in today's production lines and thus exposes a potential optimization domain. Therefore, it is concluded that the proposed benefits of Industry 4.0 can be achieved by developing and implementing decentralized production planning and scheduling concepts that utilize the exposed flexibility potentials.

Nevertheless, the developed method for deriving *flexibility graphs* considers only the technical restrictions to determine the predecessors and successors. To validate the practical relevance of the results, other restrictive factors, e.g., the spatial accessibility after changing the assembly order, need to be considered as well. Furthermore, to achieve a holistic optimization, the scope of the analysis needs to be extended to assimilate the entire production system. For this reason, visualization techniques as well as graph mining methods have been presented to face complexities by (semi-)automation of the analysis process.

The obtained information about the routing flexibility of the production processes constitutes the basis for decentralized program planning concepts. The introduced concept of the *volume cycle* is only the first approach to leverage the investigated flexibility potentials. Additionally, a concept for decentralized program scheduling is required to plan and produce the orders of a volume cycle within the available time slot. It will be part of future works to determine the amount of orders that can be grouped in a volume cycle and the amount of the corresponding time slot for producing the orders of this volume cycle. Further boundary conditions for defining the volume cycle are, among others, the size of the logistics warehouse and the number of transport vehicles. These and other parameters will be evaluated in further research.

Acknowledgements

This work was supported by the German Federal Ministry for Economic Affairs and Energy (BMWi) under the "AUTONOMIK fur Industrie 4.0" research program (Grant no. 01MA13007) within the project "Smart Micro Factory fur Elektronikfahrzeuge mit schlanker Produktionsplanung" (SMART FACE). Additionally, we would like to thank Mr. Ulrich Weidemann from the Volkswagen AG for his technical input and valuable discussion regarding the topic of flexibility graphs.

References:

[1] D. Archambault, T. Munzner, and D. Auber, "TopoLayout: Multilevel Graph Layout by Topological Features," IEEE Transactions on Visualization and Computer Graphics, Vol.13, No.2, pp. 305–317, March, 2007.

[2] J. Aurich, "Automobilproduktion," Springer Vieweg, 2014.

[3] R. F. Aziz, "RPERT: Repititive-Projects Evaluation and Review Technique," Alexandria Engineering Journal, Vol.53, pp. 81–93, 2013.

[4] K. Baker, "Introduction to Sequencing and Scheduling," Wiley, New York, 1974.

[5] G. Battista, P. Eades, R. Tamassia, and I. Tollis, "Graph Drawing: Algorithms for the Visualization of Graphs," Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 1998.

[6] T. Bauernhansl, M. ten Hompel, and B. Vogel-Heuser (Eds.), "Industrie 4.0 in Produktion, Automatisierung und Logistik," Springer Vieweg, Wiesbaden, 2014.

[7] J. Blazewicz and D. Kobler, "Review of properties of different precedence graphs for scheduling problems," European Journal of Operational Research, Vol.142, pp. 435–443, 2002.

[8] K. Brauer, F. Zesch, M. Motta, C. Schwede, A. Wagenitz, C. Reeker, K. Liebler, J. Maaß, C. Engmann, S. Schneider, P. Thomas, M. Preuß, J. Kemper, A. Hermes, M. Marr, and M. Florian, "Integrierte Terminierung und Transportplanung für komplexe Wertschöpfungsstrukturen: Partnerbericht zum Projektabschluss: Fraunhofer-Institut Materialfluss und Logistik," Technical report, Fraunhofer Institute for Material Flow and Logistics, Berlin, 2011.

[9] J. Browne, D. Dubois, K. Rathmill, S. P. Sethi, and K. E. Stecke, "Classification of flexible manufacturing systems," Flexible Manufacturing Systems Magazine, Vol.2, pp. 114–117, 1984.

[10] M. Broy, H. Fischer, K. Beetz, and W. Damm, "Cyber-Physical Systems," Springer, 2010.

[11] J. Buzacott, "Production Planning and Control: Basics and Concepts," Oldenbourg, München, 2012.

[12] Capgemini Consulting, "Industry 4.0 – The Capgemini Consulting View," Website, 2014, http://www.de.capgemini-consulting.com/sites/default/files/resource/pdf%/capgemini-consulting-industrie-4.0_0.pdf [accessed December 15, 2014]

[13] D. Chakrabarti and C. Faloutsos, "Graph Mining: Laws, Tools, and Case Studies," Synthesis Lectures on Data Mining and Knowledge Discovery, Morgan & Claypool Publishers, 2012.

[14] M. Chimani, C. Gutwenger, M. Jünger, G. Klau, K. Klein, and P. Mutzel, "The Open Graph Drawing Framework (OGDF)," In R. Tamassia, editor, Handbook of Graph Drawing and Visualization, chapter 17, pp. 543–569, CRC Press, 2013.

[15] W. D. Cottrell, "Simplified Program Evaluation and Review Technique (PERT)," Journal of Construction Engineering and Management, 1999.

[16] M. Dijk, R. Orsato, and R. Kemp, "The emergence of an electric mobility trajectory," Energy Policy, Vol.52, pp. 135–145, 2013, Special Section: Transition Pathways to a Low Carbon Economy.

[17] E. Dijkstra, "A note on two problems in connexion with graphs," Numerische Mathematik, Vol.1, No.1, pp. 269–271, 1959.

[18] H. ElMaraghy, "Flexible and reconfigurable manufacturing systems paradigms," International Journal of Flexible Manufacturing Systems, Vol.17, No.4, pp. 261–276, October, 2006.

[19] European Factories of the Future Research Association, "Multi Annual Roadmap for the Contractual PPP under Horizon 2020," 2013.

[20] R. Floyd, "Algorithm 97: Shortest Path," Communications of the ACM, Vol.5, No.6, p. 345, June, 1962.

[21] German Government, "Hightech Strategie Industrie 4.0," http://www.hightech-strategie.de/de/Industrie-4-0-59.php [accessed December 9, 2014]

[22] O. Goldreich, S. Goldwasser, and D. Ron, "Property testing and its connection to learning and approximation," In Proceedings of the 37th Annual Symposium on Foundations of Computer Science, pp. 339–348, October 1996.

[23] B. Grafen, "Prozessorientierte Auftragsabwicklung in der Automobilindustrie," Ph.D. thesis, Philipps-Universität Marburg, 2001.

[24] W. Günthner and M. ten Hompel (Eds.), "Internet der Dinge in der Intralogistik," Springer, Berlin, 2010.

[25] T. Gupta, "Applying the Critical Path Method to Manufacturing Routing," Computers industrial Engineering, Vol.21, Nos.1–4, pp. 519 – 523, 1991.

[26] P. Hart, N. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, Vol.4, No.2, pp. 100–107, July, 1968.

[27] W. Herlyn, "PPS im Automobilbau: Produktionsprogrammplanung und steuerung von Fahrzeugen und Aggregaten," Fahrzeugtechnik, Hanser Verlag, München, 2012.

[28] A. Hermes, "Modellbasierte Bewertung von Potenzialen einer distributionsorientierten Programm- und Reihenfolgeplanung in der Automobilindustrie," Verlag Praxiswissen, Dortmund, 2011.

[29] T. Jéron and C. Jard, "3D Layout of Reachability Graphs of Communicating Processes," In Proceedings of the DIMACS International Workshop on Graph Drawing, GD '94, pp. 25–32, Springer-Verlag, London, UK, 1995.

[30] H. Kagermann, W. Wahlster, and J. Helbig, "Recommendations for implementing the strategic initiative Industrie 4.0," Technical report, acatech, 2013.

[31] A. Kahn, "Topological Sorting of Large Networks," Communications of the ACM, Vol.5, No.11, pp. 558–562, November, 1962.

[32] K.-C. Kim, C. Sohn, T. Roemer, and A. Yassine, "Configuration and Coordination of Activities within a Supply Chain," International Journal of Automation Technology, Vol.6, No.1, pp. 6–19, 2006.

[33] F. Klug, "Logistikmanagement in der Automobilindustrie: Grundlagen der Logistik im Automobilbau," Springer-Verlag, Heidelberg, 2010.

[34] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, and H. Van Brussel, "Reconfigurable Manufacturing Systems," Annals of the CIRP, Vol.48, No.2, pp. 527–540, 1999.

[35] Y. Koren and M. Shpitalni, "Design of reconfigurable manufacturing systems," Journal of Manufacturing Systems, Vol.29, pp. 130–141, 2010.

[36] A. Lester, "Project Management, Planning and Control," Vol.5, Butterworth Heinemann, 2007.

[37] H. Li, F. Karray, O. Basir, and I. Song, "Multi-Agent Based Control of a Heterogeneous System," Journal of Advanced Conputational Intelligence and Intelligent Informatics, Vol.10, No.2, pp. 161–167, 2006.

[38] M. Lu and H. Li, "Resource-Activity Critical-Path Method for Construction Planning," Journal of Construction Engineering and Management, 2003.

[39] E. Martins, "On a multicriteria shortest path problem," European Journal of Operational Research, Vol.16, No.2, pp. 236–245, May, 1984.

[40] H. B. Maynard, G. J. Stegemerten, and J. L. Schwab, "Methods-Time Measurement," McGraw-Hill Book Company Inc., New York, 1948.

[41] M. G. Mehrabi, Y. Ulsoy, and G. Koren, "Reconfigurable manufacturing systems: Key to future manufacturing," Journal of Intelligent Manufacturing, Vol.11, pp. 403–419, 2000.

[42] L. Monostori, J. Váncza, and S. Kumara, "Agent-based systems for manufacturing," CIRP Annals-Manufacturing Technology, Vol.55, No.2, pp. 697–720, 2006.

[43] X. Niu, H. Ding, and Y. Xiong, "A hierarchical approach to generating precedence graphs for assembly planning," International Journal of Machine Tools & Manufacture, Vol.43, No.14, pp. 1473–1486, 2003.

[44] Plattform Industrie 4.0, http://www.plattform-i40.de [accessed December 9, 2014]

[45] T. O. Prenting and R. M. Battaglin, "The precedence diagram: A tool for analysis in assembly line balancing.," Journal of Industrial Engineering, Vol.15, No.4, pp. 208–213, 1964.

[46] P. Schoensleben, "Changeability of strategic and tactical production concepts," CIRP Annals – Manufacturing Technology, Vol.58, No.1, pp. 383–386, 2009.

[47] G. Schuh and V. Stich, "Produktionsplanung und steuerung 1: Grundlagen der PPS," Vol.4, Springer Berlin Heidelberg, Berlin and Heidelberg, 2012.

[48] G. Schuh, N. Wemhöner, and C. Friedrich, "Scenario-based Lifecycle Analysis of Manufacturing Systems," CIRP – Journal of Manufacturing Systems, Vol.35, No.2, 2006.

[49] R. Sedgewick, "Algorithms," Addison-Wesley, Upper Saddle River, NJ, 2011.

[50] A. Sethi and S. Sethi, "Flexibility in Manufacturing: A Survey," The International Journal of Flexibility Manufacturing Systems, Vol.2, pp. 289–328, 1990.

[51] M. Sugi, M. Cheng, M. Yamamoto, H. Ito, K. Inoue, and J. Ota, "System Rescheduling in Seminconductor Manufacturing," International Journal of Automation Technology, Vol.4, No.2, pp. 184–197, 2010.

[52] K. Sugiyama, S. Tagawa, and M. Toda, "Methods for Visual Understanding of Hierarchical System Structures," IEEE Transactions on Systems, Man and Cybernetics, Vol.11, No.2, pp. 109–125, February, 1981.

[53] R. Tamassia, "Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)," Chapman & Hall/CRC, 2007.

[54] M. Vanhoucke, "Project Management with Dynamic Scheduling," Spirnger Berling Heidelberg, 2012.

[55] A. Wagenitz, "Modellierungsmethode zur Auftragsabwicklung in der Automobilindustrie," PhD thesis, TU Dortmund, 2007.

[56] G. Weigert, T. Heinlich, and A. Klemmt, "Methods for Modelling and Optimisation of Assembly Processes," Advances in Simulation for Production and Logistics Applications, 2008.

[57] A. Zubaryeva, C. Thiel, E. Barbone, and A. Mercier, "Assessing factors for the identification of potential lead markets for electrified vehicles in Europe: expert opinion elicitation," Technological Forecasting and Social Change, Vol.79, No.9, pp. 1622–1637, 2012.

[58] D. Zuehlke, "Smart Factory – Towards a factory-of-things," Annual Reviews in Control, Vol.34, No.1, pp. 129–138, 2010.

Name:
Lennart Soren Bochmann

Affiliation:
Ph. D. Candidate, Volkswagen AG
Supervised at the Institute of Machine Tools and Manufacturing, ETH Zurich

Address:
Berliner Ring 2, 38440 Wolfsburg, Germany

Brief Biographical History:
2014 Master of Science ETH in Mechanical Engineering, ETH Zurich, Switzerland
2014 Visiting Researcher at The University of California, Berkeley, USA
2014 Ph. D. Candidate at Volkswagen AG and ETH Zurich

Main Works:

- C. Dold, M. Henerichs, L. Bochmann, and K. Wegener, "Comparison of Ground and Laser Machined Polycrystalline Diamond (PCD) Tools in Cutting Carbon Fiber Reinforced Plastics (CFRP) for Aircraft Structures," 5th CIRP Conference on High Performance Cutting, pp. 178–183, 2012.
- C. Bayley, L. Bochmann, M. Helu, R. Transchel, and D. Dornfeld, "Understanding Error Generation in Fused Deposition Modeling," ASPE Spring Topical Meeting on Dimensional Accuracy and Surface Finish in Additive Manufacturing, Berkeley, USA, 2014.
- L. Bochmann, C. Bayley, M. Helu, R. Transchel, K. Wegener, and D. Dornfeld, "Understanding Error Generation in Fused Deposition Modeling," Surface Topography: Metrology and Properties, Vol.3, No.1, 014002, 2015.

Name:
Lars Gehrke

Affiliation:
Ph. D. Candidate, Volkswagen AG
Supervised at the Chair of Enterprise Logistics, Technical University of Dortmund

Address:
Berliner Ring 2, 38440 Wolfsburg, Germany

Brief Biographical History:
2012 Non-degree Business Engineering Studies, University of Waterloo, Canada
2013 Visiting Researcher in Automotive Engineering, Tongji University Shanghai, China
2014 Master of Science in Industrial Engineering, TU Braunschweig
2014 Ph. D. Candidate at Volkswagen AG and TU Dortmund

Main Works:
Advanced Manufacturing Technologies, Industry 4.0, Industrial Management, Organization and Process Improvement

Membership in Academic Societies:
Association of German Engineers (VDI)

Name:
Adrian Böckenkamp

Affiliation:
Ph. D. Candidate, Department of Computer Science VII, Technical University of Dortmund

Address:
Otto-Hahn-Str. 16, 44227 Dortmund, Germany

Brief Biographical History:
2011 Research Assistant at the Embedded System Software (ESS) Group, Department of Computer Science XII, TU Dortmund
2013 Master of Computer Science (M. Sc.), TU Dortmund
2014 Ph. D. Candidate at the Department of Computer Science VII, TU Dortmund

Main Works:
- F. Weichert, A. Bockenkamp, C. Prasse, C. Timm, B. Rudak, K. Holscher, and M. ten Hompel, "Towards Sensor-Actuator Coupling in an Automated Order Picking System by Detecting Sealed Seams on Pouch-Packed Goods," Journal of Sensor and Actuator Networks, MDPI, Vol.3, pp. 224–252, 2014.
- C. Prasse, J. Stenzel, A. Bockenkamp, B. Rudak, K. Lorenz, F. Weichert, H. Muller, and M. ten Hompel, "New Approaches for Singularization in Logistic Applications using Low Cost 3D Sensors," Sensing Technology: Current Status and Future Trends IV, Springer, pp. 191–215, 2014.

Name:
Frank Weichert

Affiliation:
Research Associate, Dr., Department of Computer Science VII, Technical University of Dortmund

Address:
Otto-Hahn-Str. 16, 44227 Dortmund, Germany

Brief Biographical History:
2000 Diploma in Computer Science, TU Dortmund
2010 Ph. D. in Computer Science, TU Dortmund

Main Works:
Research interests include image processing, simulation, signal analysis, computer graphics, scientific visualization, algorithms and geometry processing in a variety of different domains (e.g., (bio) medicine, logistics, GIS-related areas)

Name:
Rainer Albersmann

Affiliation:
Sales Manager, F/L/S Fuzzy Logik Systeme GmbH

Address:
Joseph-von-Fraunhofer-Str. 20, 44227 Dortmund, Germany

Brief Biographical History:
1992 Diploma in Computer Science, TU Dortmund
1992 F/L/S Fuzzy Logik Systeme GmbH

Main Works:
Industrial applications based on fuzzy logic, decision support and optimization

Name:
Christian Prasse

Affiliation:
Researcher, Department of Machines and Facilities, Fraunhofer Institute for Material Flow and Logistics

Address:
Joseph-von-Fraunhofer-Str. 2-4, 44227 Dortmund, Germany

Brief Biographical History:
2005 Master in Logistics, Technical University of Dortmund
2005 Fraunhofer Institute for Material Flow and Logistics

Main Works:
Vision systems, decentralized control concepts, generative manufacturing

Name:
Christoph Mertens

Affiliation:
Researcher, Department of Supply Chain Engineering, Fraunhofer Institute for Material Flow and Logistics

Address:
Joseph-von-Fraunhofer-Str. 2-4, 44227 Dortmund, Germany

Brief Biographical History:
2011 Diploma in Computer Science, TU Dortmund
2011 Fraunhofer Institute for Material Flow and Logistics

Main Works:
Process analysis and optimisation, automotive supply chains, and production planning

Membership in Academic Societies:
Gesellschaft fur Informatik e. V. (GI)

Name:
Marco Motta

Affiliation:
Head of Department Supply Chain Engineering, Fraunhofer Institute for Material Flow and Logistics

Address:
Joseph-von-Fraunhofer-Str. 2-4, 44227 Dortmund, Germany

Brief Biographical History:
2004 Diploma in Mechanical Engineering, TU Dortmund
2004 Researcher, TU Dortmund
2005 Researcher, Fraunhofer Institute for Material Flow and Logistics

Main Works:
Automotive logistics, supply chain management, simulation Methods, and logistical decision support solutions

Name:
Konrad Wegener

Affiliation:
Prof. Dr., Head of Institute of Machine Tools and Manufacturing (IWF), ETH Zurich

Address:
Leonhardstr. 21, 8092 Zurich, Switzerland

Brief Biographical History:
1990 Ph. D. in Mechanical Engineering, TU Braunschweig
1990 Head of Design and Layout Planning Department, Schuler Press GmbH & Co. KG
1999 General Manager, Schuler Press GmbH & Co. KG
2003 Head of Institute of Machine Tools and Manufacturing (IWF), ETH Zurich
2003 Head of Manufacturing Section, inspire AG

Main Works:
Optimization of machine tools, cutting technology, additive manufacturing, laser material processing, and electro discharge machining

Membership in Academic Societies:
International Academy for Production Engineering (CIRP)