Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 81 (2019) 276–281

www.elsevier.com/locate/procedia

52nd CIRP Conference on Manufacturing Systems

Approach to Generate Optimized Assembly Sequences from Sensor Data

Susann Kärcher*, Thomas Bauernhansl

Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Nobelstrasse 12, 70569 Stuttgart, Germany

* Corresponding author. Tel.: +49-711970-3838. E-mail address: susann.kaercher@ipa.fraunhofer.de

Abstract

Manual assembly processes are often plagued with a lack of transparency while holding great potential for optimization. As preparing instructions takes a lot of time, assembly processes are often poorly documented, particularly for small volumes. This paper presents a general approach to automatically generate optimized process step sequences in manual assemblies from sensor data. The idea is to analyze the data and compare different assembly processes captured by the analysis. From this, a best practice can be derived and standard times can be defined for the individual process steps.

Keywords: assembly; sensor; analysis; manual; best practice; benchmark; sequence

1. Introduction

Manual assembly workstations, especially for products with small batch sizes, offer great potential for optimization. One reason is the increasing number of product variants while the number of units per product decreases down to single-piece production [1].

This generally implies that the smaller the quantity, the less work processes have been standardized and documented. Nevertheless, the time available for planning the assembly of products is limited. As a result, most of the assembly instructions, work plans, and bills of material are insufficiently prepared.

Small wonder that this results in a lack of transparency. On the one hand, workers get unclear work instructions (e.g. concerning the assembly process, a correct execution of work steps, information about quality and time). On the other hand, planners get no feedback from assembly (e.g. actual times, causes for irregularities). This makes planning even more challenging.

So far, assembly planning is a time-consuming business, e.g. when the times for assembly processes are externally recorded [2], following a technique developed by the association for work design, industrial organization and company development (REFA) [3] or through MTM (Methods-Time-Measurement-Method) [4]. Often, there are several ways to assemble a product. After recording the process times, the best option needs to be selected for the existing conditions. The search for the most suitable assembly sequence also calls for an experienced planner. In case of inadequate planning, workers often help to overcome 'planning gaps' by providing flexibility and technical know-how.

In general, the planning of assembly sequence influences the cost and the time of assembly [5]. With automated assembly planning, however, the high planning effort can be reduced and often inaccurate planning can be overcome.

The demand for an automated creation of work plans has existed for a long time. Automated work plans can reduce the time and effort required by a company, cut personnel costs, increase data accuracy, and improve up-to-dateness. [6]

One way to take a step towards automated assembly planning is to establish a system using sensors for recording assembly processes [7], so that the collected data can be used for automated planning.

This paper presents a general approach to automate the generation of optimized process step sequences in manual

2212-8271
10.1016/j.procir.2019.03.048

assemblies based on sensor data. The idea is to develop a best practice by combining solution strategies identified by the sensor analysis. The approach aims not to find the optimal solution but to identify the best solution in the context of the analysis.

The underlying ideas work for a one-time application within the framework of an optimization project as well as for continuous application.

2. Automated Manual Assembly Planning and Optimization

The following section outlines the state-of-the-art in the optimization of manual assembly processes and automated assembly planning using sensor data.

2.1. Automatic Analysis of Manual Assembly Systems

The manual recording of assembly times requires a significant amount of effort [2, 3, 4]. As a result, there are multiple approaches for automating the analysis of manual assembly processes. They differ in their goals and the way of data acquisition, as shown in Fig. 1 below.

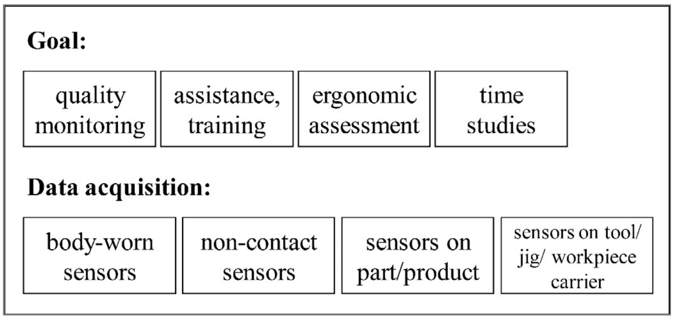

Fig. 1. Overview of automated analysis of manual assembly systems.

The goals of automated recording of manual assembly processes include quality monitoring, assistance and training, ergonomic assessment, and time studies. Data can be acquired from body-worn sensors, non-contact sensors such as cameras, sensors on parts or products and sensors on tools, jigs or workpiece carriers. There is also a combination possible of sensors used, so-called hybrid forms.

Systems are used for tracking assembly activities to *monitor the quality* of the (intermediate) product. For instance, sensors for monitoring the proper execution of all steps and the correct assembly of a product. It is the same for *assistance or training,* because the system recognizes the process steps and confirms when they are successfully completed. Examples are presented by Müller et al. [8] or Stiefmeier et al. [9]. For an *ergonomic assessment* of manual assembly processes, many approaches are in use, in recent years particularly research in motion capture and analysis [5]. Examples are presented by Gudehus [10], who does an ergonomic analysis with body-worn sensors, or Härtel [11], who uses ergonomic analysis via a marker-based system combined with IMU sensors. There are also approaches for *time studies*. An example is presented by Ma et al., who compares recorded times to standard times and thus determines efficiency [12]. Agethen et al. [13] present an approach to improve planned times by a comparison of actual and planned work paths of the operators.

2.2. Automatic Generation and Optimization of the Assembly Sequence

In general, a distinction can be made between the acquisition and optimization of real data and the determination and optimization of planned data.

There are a number of approaches to automatically derive an optimal assembly sequence from CAD data. Computer-aided assembly planning (CAAP) generates assembly sequences by examining the disassembly process [5]. Approaches to plan assembly from CAD models are, for example, Belhadj et al. [14], who present an approach to generate a subassembly algorithm from CAD model or Bikas et al. [15]. In addition, there are approaches to virtually plan assembly from CAD models (see [5] for an overview).

Fewer solutions exist in the area of optimization of real data. For automated assembly and manufacturing, there are approaches to collect data. Müller [16] analyzes machine and sensor data and optimizes overall equipment effectiveness (OEE). After evaluating the data, all machines are subdivided into their individual process steps to combine the shortest time units to a benchmark machine.

There are various approaches and optimization methods to determine assembly sequences. Recent approaches show that evolutionary algorithms can also be applied for assembly sequence planning (e.g. [17, 18]).

3. Approach to Generate Optimized Assembly Sequences from Sensor Data

This paper shows how to apply a benchmarking approach to manual assembly.

Camp [19] defines benchmarking as a 'continuous process of measuring products, services, and practices against the toughest competitors or those companies recognized as industry leaders' and as 'the search for industry best practices that lead to superior performance'.

In the following, the idea of benchmarking to learn from other better practices is applied to the optimization of assembly. Instead of planning the assembly sequence from the beginning, existing knowledge is used.

There are a number of benchmarking procedure models that differ, for example, in the number of phases and the possibility of repetitions [20].

Watson [21] describes the benchmark process through the following steps:

- planning
- data gathering
- data analysis
- making improvements

Not every benchmarking study requires to be carried out through the four steps given in the model. The model should be seen as a guideline and to help understand the process. [21]

Watson suggests guidance in each phase: In the first step of 'plan study' the target, measure of performance and the investigation method of the process are selected. Also, the method of data collection and criteria for comparison are defined. In the second step 'data gathering', internal, secondary, and external data are collected. Moreover, suitable comparative entities are searched and selected. 'Data analysis' as third procedure starts by preparing the data, then identifies performance gaps, and finally derives the best practices. Moreover, it must be examined why a best practice achieves better performance. The last phase is that of 'making improvements'. The selected improvement actions are introduced to the company based on the knowledge gained from benchmarking and actually the best practice is implemented. [21]

Benchmarking is more than just imitation. It is crucial to understand why the best practice is better than other solutions. [22]

In the following, an approach is presented based on the steps above. The focus of the paper lies on the first three steps.

3.1. Planning

In the planning phase, the system is analyzed and the model is generated.

The planning aim is to record the assembly processes of different workers with different solution strategies and to derive the best solution strategy.

The target value is to minimize time. Consequently, the best process is the sequence that results in the shortest assembly time. At present, conventional assembly planning also includes collecting times for the assembly processes. In order to avoid the target system becoming too complex, this approach uses time as the primary target value for optimization. However, later other aspects (e.g. ergonomics, hybrid forms of assembly) must also be taken into account for the design of the assembly system.

The following Fig. 2 shows the system considered within the approach in a schematic way.

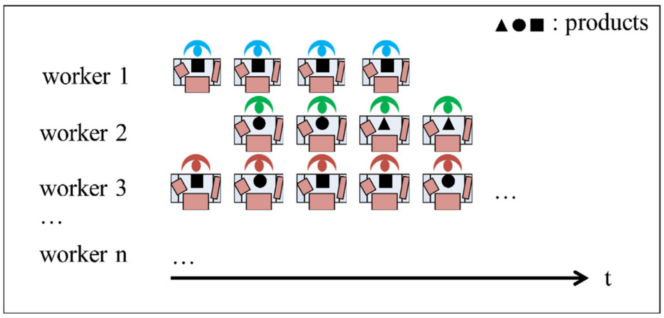

Fig. 2. Manual assembly system - recording data over time.

The figure shows that, within the framework of the analysis, the assembly activities of several workers (e.g. blue, green, red) are recorded over time. They assemble different products (e.g. triangle, circle, and square) at different work stations.

Recording and optimizing manual assemblies is a complex issue: various characteristics of a manual assembly system need to be taken into account to specify the data. The challenges essentially have to do with manual execution, especially, if not all of the steps are documented, or with recording the processes via sensors:

- *Different types, variants and options:* Especially small lot sizes differ in types, variants and options. This means, that different products have to be considered during an analysis.
- *Different workers*: In manual assembly, as the term suggests, there are workers who vary in their assembly skills and who develop different solution strategies.
- *Diverse assembly sequences:* Often, the definition of the assembly sequence is not clear. As a result, the assembly sequences of different workers might vary widely.
- *Different processes:* Another challenge arises from operators who execute different processes to achieve the same goal. Though the result of the process can be compared, the processes themselves might differ.
- *Work station:* An operator's time spent to carry out a process step heavily depends on the set-up at the work station.
- *Data gaps:* Another great challenge are data gaps detected during a manual assembly through sensors and automated process recognition.

3.2. Data Gathering

The optimization is enabled by actual data instead of plan data. Data gathering is based on the data acquired from a system that is currently being developed for the automated analysis of manual assembly processes. The system can be used in every manual assembly due to its scalable and modular structure. [7] The following figure shows the structure of the previously outlined system.

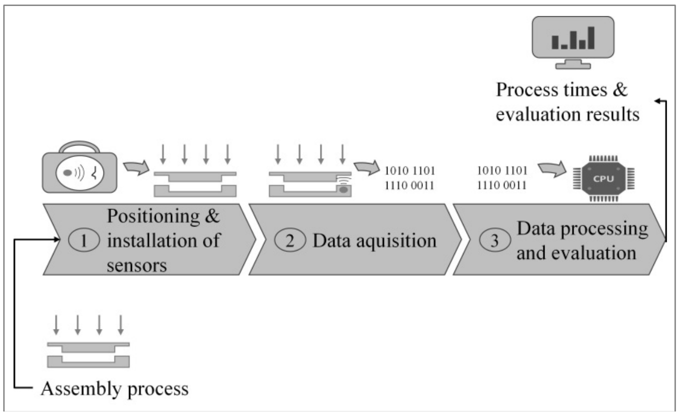

Fig. 3. Structure of the analysis system [7].

The system covers three levels: the setting (on the shop floor), the IT architecture, and the evaluation. The data is captured on the shop floor by sensors on parts, products, tools, and other equipment. Afterwards, the data is transmitted to the database and analyzed. The main advantage of the system is a flexible way of analysis. [7]

A combination of different sensors is used for the recording of the processes, mainly accelerometer, magnetometer and gyroscope. The recognition of the process steps and process times results from sensor fusion and following evaluation. Data from video and an app are recorded as an auxiliary until the system recognizes all processes reliably.

The system provides information on an operators' work during a process and indicates the time, when and for how long she/he worked. Therefore, all process data include the following information:

- process_ID
- process_name
- worker_ID
- start_time
- end_time
- predecessor
- successor
- product_ID
- work_station_ID

The distinction into process_IDs is highly important. This means that processes sharing the same ID describe the same assembly step.

3.3. Data Analysis and a Development of Best Practice

Once the solutions have been recorded, the best practice has to be found from the recorded solutions. The approach does not aim to find the optimal solution but the best one identified in the analysis (depending on the selected optimization method). This is done in three steps:

Step 1: Generate graph

This problem can be considered as an SSP problem (Single Source Shortest Path Problem): There is a digraph $G = (V, E)$ with weights $c: R \rightarrow R$, as well as a vertex $s \in V$. Each directed edge (u,v) is an ordered pair of vertices, where u is the starting vertex and v is the destination vertex of the edge (u,v). The shortest path from s to v for all $v \in V$ is searched for. [23]

The first solution demonstrated in the analysis is the first path of the digraph. The nodes describe the process results. Moreover, the edges define the predecessor-successor-relationship and the costs at the edges describe the process duration. The final digraph does not necessarily represent an assembly precedence graph, as it does not cover all precedence relations.

There are two options for adding further solutions to the digraph:

- A) The assembly was carried out in the same order: This means, the individual process duration needs to be compared. If the new process time is shorter, the original solutions are updated; if the process time is slower, the new solution is discarded. It must be ensured that outliers in the time data are recognized and not stored as new process times.
- B) The assembly was performed in a different order: This means, a new path has to be created in the digraph. For each new assembly process sequence identified in the analysis, a new path is created in the digraph. New nodes must be generated because of the precedence dependencies and the assembly progress.

Step 2: Check graph

After all solutions have been illustrated in the digraph, it first must be checked, whether the graph is cycle-free or if process steps appear multiple times. A cycle occurs after rework or adjustment. If reworking is carried out, the time needed must be added to the costs of the actual process.

Step 3: Find shortest path

In the following, the shortest path through the digraph has to be found. For this, heuristics, evolutionary algorithms, or machine learning are used. Also, data sets must be used to prove the best optimization methods (quality of the solution, computing time).

It is not possible to calculate the total time by summing up the shortest individual times, as the sequence of execution differs and there are dependencies between the processes (in contrast to, for example, in the machine benchmark illustrated in chapter 2.2. Individual processes are carried out in a precisely defined sequence.). Moreover, a worker may have assembled the product in the shortest total time, but other workers have better partial solutions or better assembly sequences but a slower execution (e.g. due to less experience).

As mentioned in chapter 3.1, the target value is to minimize time. At the beginning, assembly is considered as if a product was completely assembled at a work station by only one operator. In the next step, however, this assumption must be scaled up and the assembly has to be optimized holistically. This consideration is particularly important when differences in cycle times occur. For this purpose, it is crucial to maintain the different solutions represented in the graph. Only the individual process times are updated, but none of the solution strategies (corresponds to paths in the graph) are deleted. If the associated assembly graphs of all products are known, the whole assembly can be optimized. The deviation from the individual optimum is possible.

3.4. Make Improvements

In the last step, the newly found solution actually has to be implemented.

The presented approach to generate a best practice from identified solutions can be used both for a temporary optimization project and for a continuous optimization. Currently, the derivation from automated recording and optimization to implementation has not yet been finally realized.

It is important to be aware that not all results are directly transferable [22]. For example, not all operators are able to carry out the assembly process in the same way and the same time.

New work instructions are generated on the basis of the results. It will be crucial to make them available to the workers in a suitable way and to properly communicate the new results.

4. Practical Example

In the following, the basic ideas of the approach presented are applied to a simple exemplary assembly. Fig. 4 illustrates the component to be assembled: an axle.

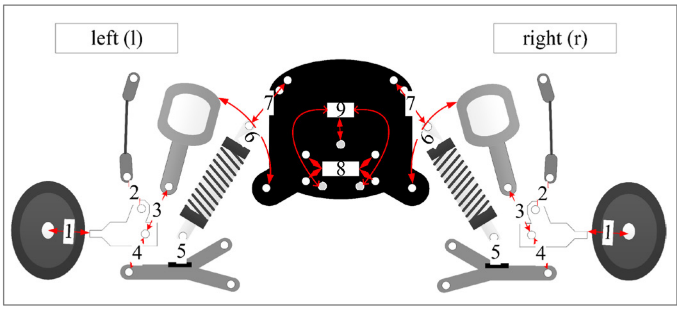

Fig. 4. Axle assembly.

The following Table 1 describes each process step:

Table 1: Process steps

Number	Description	Both-sided	Process_number
9	Rear Middle Part (3 screws)		9
8	Front Middle Part (4 screws)		8
7	Spring on Middle Part	X	7l, 7r
6	Upper Wishbone to Middle Part	X	6l, 6r
5	Spring to Lower Wishbone	X	5l, 5r
4	Lower Wishbone on Axle Mount	X	4l, 4r
3	Upper Wishbone on Axle Mount	X	3l, 3r
2	Tie Rod on Axle Mount	X	2l, 2r
1	Axle Mount on Wheel	X	1l, 1r

The process_number is transformed into process_ID by adding ‘_’ and run_number, for example a process in the first run and with process_number 4l receives the process_ID ‘4l_1’.

The product has been chosen because it allows different assembly sequences. Process times of three different persons (worker 1, worker 2 and worker 3) were recorded. Video data were used to support the data gathering. The attendants had different experiences. No work instructions were provided but the assembled product could be inspected in advance. To simplify the situation, all workers assembled an identical product at the same workplace.

Fig. 5 shows an overview of the assembly times. W refers to the worker, R to the run. Therefore, the figure visualizes three runs comparing the three workers. The assembly time is between 8.9 and 14.8 min at an average of 11 min.

Fig. 6 gives an overview of a single process level and shows that the times required for the individual assembly processes (1-9) sometimes differ significantly. The colors show in which run and for which worker the process times were recorded.

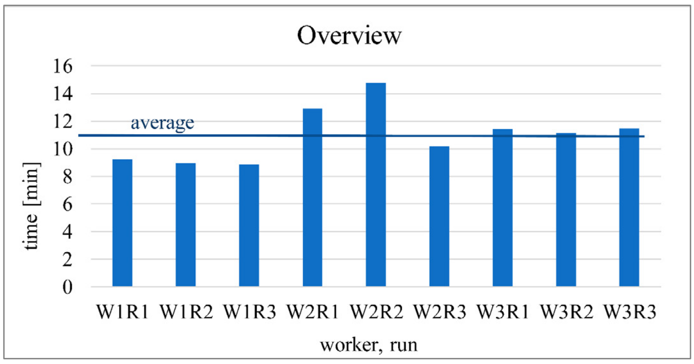

Fig. 5. Overview.

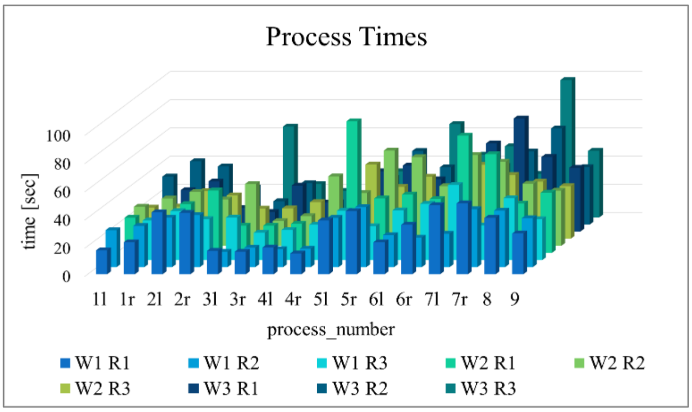

Fig. 6. Overview of a single process level.

It turned out that the workers had implemented different solution strategies, i.e. different assembly sequences. Fig. 7 gives an overview.

As described in chapter 3.3, a new path and new nodes are created in the graph for each new solution strategy. The colors white, orange and blue refer to the workers. For a clearer visualization, process times (costs) are not shown in the graph.

It results in the following requirements for evaluation:

- The evaluation must be able to deal with errors and rework. In the given example, it turns out that worker 3 made some mistakes, e.g. process steps 1 and 2 are completely missing in the lowest branch of the graph. Processes 7l and 7r appear several times because rework is required.
- The data recorded should be as detailed as possible. Thus, processes executed twice could possibly be better assigned to a substep.
- In some cases, workers vary in the beginning but then the processes are in the same order. The evaluation should consider whether these processes are still comparable (depending on the precedence dependencies).

In the next step, optimization methods must be tested.

5. Conclusion and Outlook

This paper presents a general approach to optimize manual assembly processes by using sensor data and deriving a best

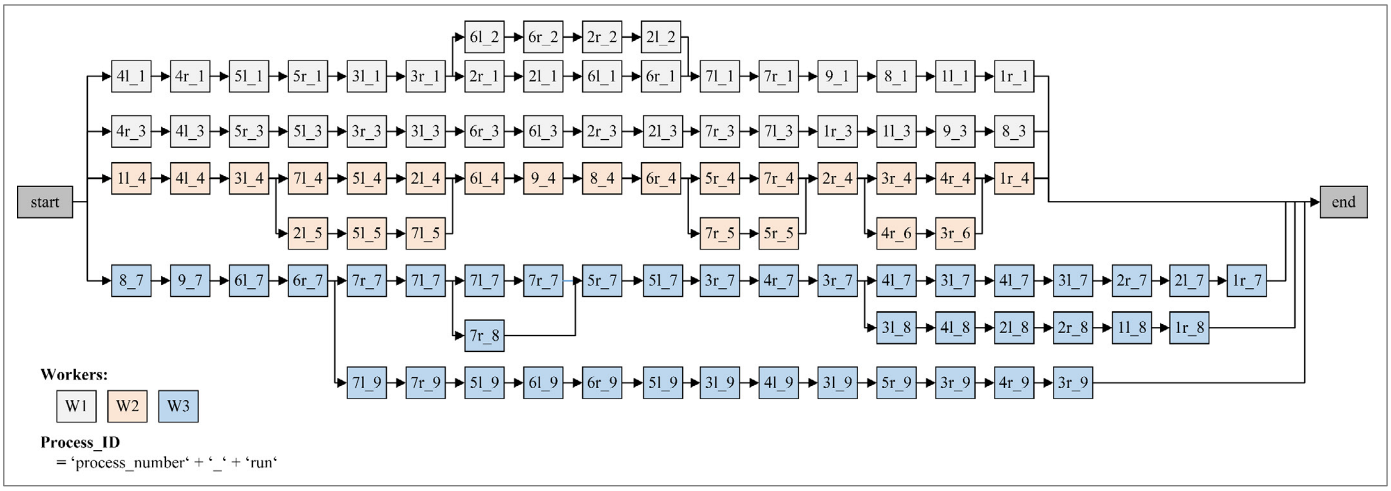

Fig. 7. Visualization of solution strategies.

practice. A comparison of workers' different solution strategies, as in the context of the analysis, lead to a best practice.

It is shown that manual assemblies, especially if insufficiently planned, can become highly complex.

The advantage lies in using real data as opposed to plan data. This way, assembly can be planned automatically, i.e. with less time and effort. This approach might not find the optimal solution, but an appropriate solution with less effort. The approach offers the possibility to externalize existing knowledge and to improve planned times.

Next, optimization methods providing the best possible solution within an acceptable computing time have to be investigated in practice. Hence, several data sets have to be considered. Moreover, an evaluation of additional data needs to be performed. Furthermore, a way to identify incorrect solutions must be found as well as to ensure whether an assembly process achieves the required objective at all. Another challenge in complex assemblies is to recognize two processes that are similar and to label them with the same process_ID. It might also be interesting to examine the influence of worker qualifications on the outcome of the approach.

References

[1] Koren Y. The Global Manufacturing Revolution. Hoboken, New Jersey: John Wiley & Sons, Inc; 2010. p. 15.

[2] Lotter B, Wiendahl H-P. Montage in der industriellen Produktion. 2. Auflage, Berlin Heidelberg: Springer Verlag; 2012.

[3] REFA Bundesverband e. V. Methodenlehre der Betriebsorganisation – Grundlagen der Arbeitsgestaltung. München. 1993. p. 81-125.

[4] Picker C. Prospektive Zeitbestimmung für nicht wertschöpfende Montagetätigkeiten. Technische Universität Dortmund, Fakultät Maschinenbau, Dissertation. Aachen: Shaker Verlag; 2006.

[5] Leu MC, ElMaraghy HA, Nee AY, Ong SK, Lanzetta M, Putz M, Zhu W, Bernard A. CAD model based virtual assembly simulation, planning and training. CIRP Annals 2013; 62(2):799–822.

[6] Opitz H, Olbrich W, Steinmetz G. Automatische Arbeitsplanerstellung. Opladen: Westdeutscher Verlag; 1972. p. 10-11.

[7] Kärcher S, Cuk E, Denner T, Görzig D, Günther LC, Hansmersmann A, Riexinger G, Bauernhansl T. Sensor-driven Analysis of Manual Assembly Systems. Procedia CIRP 2018; 72:1142-1147.

[8] Müller BC, Nguyen TD, Dang QV, Duc BM, Seliger G, Krüger J, Kohl H. Motion Tracking Applied in Assembly for Worker Training in different Locations. Procedia CIRP 2016; 48:460-465.

[9] Stiefmeier T, Roggen D, Ogris G, Lukowicz P, Tröster G. Wearable Activity Tracking in Car Manufacturing. IEEE Pervasive Computing 2008; 7(2):42-50.

[10] Gudehus T. Entwicklung eines Verfahrens zur ergonomischen Bewertung von Montagetätigkeiten durch Motion-Capturing. Kassel: kassel university press GmbH; 2009.

[11] Härtel T, et al. Capturing and Assessment of Human Motion during Manual Assembly Operation. First International Symposium on Digital Human Modeling, Conference Proceedings 2011.

[12] Ma L, Zhang W, Fu H, Guo Y, Chablat D, Bennis F. A Framework for Interactive Work Design based on Motion Tracking, Simulation, and Analysis. Human Factors and Ergonomics in Manufacturing 2010; 20(4):339-352.

[13] Agethen P, Otto M, Gaisbauer F Rukzio E. Presenting a novel motion capture-based approach for walk path segmentation and drift analysis in manual assembly. Procedia CIRP, 2016; *52*: 286-291.

[14] Belhadj I, Trigui M, Benamara A. Subassembly generation algorithm from a CAD model. The International Journal of Advanced Manufacturing Technology 2016; 87.9-12: 2829-2840.

[15] Bikas C, Argyrou A, Pintzos G, Giannoulis C, Sipsas K, Papakostas N, Chryssolouris G. An automated assembly process planning system. Procedia CIRP, 2016; 44: 222-227.

[16] Müller F. Fraunhofer IPA Referenzprojekte. Neues Auswertungsmodell entdeckt in den Zykluszeiten von Maschinen Produktivitätspotenziale zwischen 6 und 10 Prozent bei Freudenberg. https://www.ipa.fraunhofer.de/de/ referenzprojekte/ Maschinenbenchmark.html

[17] Özmen Ö, Batbat T, Özen T, Sinanoğlu C, Güven A. Optimum Assembly Sequence Planning System Using Discrete Artificial Bee Colony Algorithm. Mathematical Problems in Engineering 2018.

[18] Maadi M, Javidnia M, Ramezani R. Modified Cuckoo Optimization Algorithm (MCOA) to solve Precedence Constrained Sequencing Problem (PCSP). Applied Intelligence 2018; 48(6):1407-1422.

[19] Camp R. Benchmarking: the search for industry best practices that lead to superior performance. ASQC/Quality Press, 1989. p.10-21.

[20] Böhnert A. Benchmarking: Charakteristik eines aktuellen Managementinstruments. Kovač, 1999.

[21] Watson GH. Benchmarking – Vom Besten lernen. Landsberg/Lech: Verlag Moderne Industrie, 1993. p. 82-97.

[22] Watson GH. Benchmarking – Vom Besten lernen. Landsberg/Lech: Verlag Moderne Industrie, 1993. p. 20-21.

[23] Krumke SO. Graphentheoretische Konzepte und Algorithmen, 2012. p. 169.