

PLANNING AND ANALYSIS OF MECHANICAL ASSEMBLY SEQUENCES IN DESIGN ENGINEERING – PART I: THE METHOD

Michał Szaśiadek

Original scientific paper

The article presents theoretical notions related to assembly sequence planning in the process of elements design engineering and design for machines assembly. Basic modules of the designed method are described and shown on simple examples. The concept is based on the assumption that the method should help the engineer-constructor in specifying the best assembly sequence, taking into account the principles of design for assembly at an early stage of development of the product design. The second part of the article explains how to use the described method on the basis of an authentic industrial situation.

Keywords: *assembly sequence planning; design for assembly; engineering design*

Planiranje i analiza toka operacija montaže u projektiranju – Dio I: Metoda

Izvorni znanstveni članak

U radu se predstavljaju teorijski pojmovi vezani uz planiranje toka operacija montaže u postupku konstruiranja elemenata i projektiranja za montažu strojeva. Opisani su osnovni moduli projektirane metode i prikazani na jednostavnim primjerima. Pristup se zasniva na pretpostavci da bi ta metoda trebala pomoći inženjeru-konstruktoru u određivanju najboljeg toka operacija montaže, uzimajući u obzir principe konstruiranja za montažu u ranom stadiju razvoja projekta proizvoda. Drugi dio rada objašnjava kako primijeniti opisanu metodu u autentičnoj industrijskoj situaciji.

Ključne riječi: *planiranje toka operacija montaže; projektiranje za montažu; tehnički projekt*

1 Introduction

In the last few decades there has been noted a trend of quick introduction of products to the market. This trend is a result of increasing competition and technological progress. To meet the market conditions it is appropriate to introduce the product as quickly as possible. The quality of an industrial product: an element, a machine, a device or an apparatus is formed mostly during processes such as: design and construction, production, and assembly. Of these three operations design is of particular importance because it should include a number of requirements, such as technological or assembly.

The traditional method of designing and building machines is characterized by sequencing its particular stages and is relatively long. An alternative to this process is concurrent design, which eliminates inconveniences. It is characterized by team work and simultaneous realization of its stages, which contributes to overt inclusion of many subsequent development phases in the early stage of design. Dynamically developing methods of concurrent design have become the basic and indispensable form of design work.

In the concurrent design trend there have emerged a lot of methods aimed at different design aspects – DfX ('Design for X' where 'X' is a distinguished group of requirements), all of which should be included in the project to a satisfying degree. The methods contain design for manufacturing, assembly, etc. A particularly important method, due to the subject matter of this article, is DfA – Design for Assembly [1, 2, 3]. It requires taking into account the assembly process requirements over the course of the product design cycle, so that, in terms of speed and cost, the construction would be primarily adjusted to an efficient assembly. Taking into account all the requirements of the assembly process at the design stage is a complicated task. The practical application of

the principles of design for assembly is often limited to the intuitive consideration of these requirements during the development of the project. It is commonly known that manufacturing costs constitute the major part of the implementation process, while the installation costs can be even 30 ÷ 50 % of the cost of manufacturing the product. Therefore, design for assembly is one of the most important methods in the wide field covered by the general name of DfX [1, 3, 4, 5].

The assembly process, which consists of combining the individual components in the assembly units of a higher order, up to the complete assembly, is one of the most important steps in the process of production. Therefore it is important to carry out the process as quickly and accurately as possible. The importance of the assembly process for the costs of implementation and the quality of the product suggests the explicit inclusion of the process in the time of construction of machines and their components. The author's practical observation and insight show that the assembly process is often planned after the completion of the project, or even after the components are manufactured. Even then there is a possibility to incorporate various variants of the assembly process, but they are determined implicitly by the design. It would be much better for the designer to include the assembly requirements explicitly during the design and construction process. It is important then to develop a product design that ensures correct, flawless, and as fast and inexpensive as possible installation. Labour intensity of the assembly process increases with the number of the assembly parts in the product, and the proper conduct of this process depends on the order and the correct connection of the parts. Hence, it is important to determine the best sequence of connecting the components (assembly sequence) of the product. With a large number of components, the issue of determining the best assembly sequence is combinatorial complex, and

that requires the development of an efficient algorithm for determining the set of feasible solutions [6, 7].

Therefore, the designer should have a tool to check and analyse the created structure for assembly requirements. He should also be able to generate assembly sequence variants and to assess their influence on the progress and costs of the manufacturing process. The opportunity to evaluate and select an optimal (according to accepted criteria) assembly sequence during production would eliminate the necessity to adapt the design to the installation needs in the future. Such a tool would enable the product design to simultaneously plan the best sequence of connecting the components. The product would be concurrently designed with the project of its installation.

2 Characteristics of the method

The method described in this section is used to determine the best assembly sequence of the product components. The assumption is that the method should be applied at the early stage of design (preliminary design) once the structure of the product has been determined. At this stage, after the proposed method had been applied, the combinations for joining components of the product can be generated. Also, it is possible to evaluate designated sequences and every combination in them due to efficiency of its assembly. As a result, after an analysis of the output (best) sequences, it is possible to change some of the design features of the product components (or even its structure) to streamline the process of assembly. In this case, any changes to the initial solution will be implemented before the detailed design phase, without the involvement of unnecessarily large amount of time, and, above all, without the excessive cost [6, 7].

It is assumed that application of the proposed method could contribute to: high product quality, reduction of time and cost of launching the product to the market, simplification of the process of connecting the product components, minimization of positioning, adjustment of the structure components for ease of handling, selection (or change) of the connection type to shorten the process, reduced complexity of the components shape, reduction of the force necessary to set the elements in relation to each other, etc. [2].

2.1 Problem of determining the assembly sequence

Determination of the feasible (possible) sequence assemblies is a complex issue, and as the complexity of an assembly (machine) increases it leads to the phenomenon of the so-called 'combinatorial explosion'. The issue of variants of the assembly sequence is shown on two simple, but sufficient to illustrate, examples of a four-element sub-assemblies. The sub-assemblies are presented in Fig. 1.

Theoretically, without taking into account the constraints of construction components, there are $4! = 24$ ways of assembly of both components.

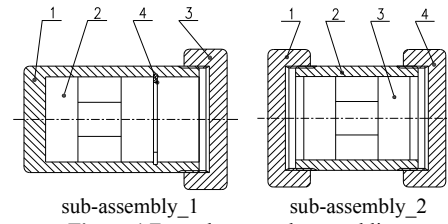


Figure 1 Four-element sub-assemblies

The number of possible combinations for increasing number of elements grows according to the factorial function (permutations of n -element set, where n is the number of the product components). Hence, when the elements number is greater, a systematic analysis of all variants is virtually impossible without a proper instrument. Such an analysis is not necessary after all, as there are plenty of unfeasible and not recommended variants in this theoretical set. It can be seen that there are only 4 permissible sequences for the sub-assembly 1 [(1-2-4-3), (2-4-1-3), (2-1-4-3), (4-2-1-3)] due to the possibility of their implementation. The sub-assembly 2 has 12 proper assembly sequences. Although the examples are simple there are 12 and 20 not permissible sequences for the first and second sub-assembly respectively. From the analysis of the listed cases it can be stated that effective assembly sequence generation algorithm should be characterised primarily by elimination of all the solutions from the result set, which in terms of actual implementation are unfeasible or incorrect. It can be accomplished by defining the appropriate constraint requirements.

2.2 General method characteristics

The proposed method for determining the set of feasible assembly sequences and their evaluation and selection belongs to the category of classic formal methods. In this method four basic and implemented sequentially modules can be distinguished. In the first module the product design is mapped in the form of design structure matrix (M_k). In the second module, all relations stored in the M_k matrix are evaluated. The result of the implementation of the first two modules is a record of the product design structure in the form of M_k matrix and assigning an evaluation indicator to each relation stored in this matrix. In the third module precedence constraints for the assembly connections are defined (generated on the basis of the M_k matrix). They are taken into consideration to determine the set of feasible assembly sequences. The last module of the method is the algorithm for generating the set of feasible assembly sequences. The resulting assembly sequences are evaluated and recorded. It is possible to repeat the analysis with appropriate modifications in the individual steps of the method. The next section describes the method's modules.

3 Description of the method's modules

3.1 Record of the product structure design

All contact relations between the components of the product are identified on the basis of product design documentation. Contact relation is understood as the

possibility of combining two parts. Established relations (connections) are stored in the form of a graph and the corresponding matrix - called further the relationship matrix or structure design matrix - M_k . This matrix has a size of $n \times n$, where n is the number of the product components. Relations between the product components can assume three forms. They are presented in Tab. 1.

Table 1 Forms of the matrix record of relations between the components

M_k matrix	Relation direction									
<table border="1"> <tr><td></td><td>1</td><td>2</td></tr> <tr><td>1</td><td></td><td></td></tr> <tr><td>2</td><td>x</td><td></td></tr> </table>		1	2	1			2	x		 $1 \leftrightarrow 2$
	1	2								
1										
2	x									
<table border="1"> <tr><td></td><td>1</td><td>2</td></tr> <tr><td>1</td><td></td><td>x</td></tr> <tr><td>2</td><td></td><td></td></tr> </table>		1	2	1		x	2			 $2 \leftarrow 1$
	1	2								
1		x								
2										
<table border="1"> <tr><td></td><td>1</td><td>2</td></tr> <tr><td>1</td><td></td><td>x</td></tr> <tr><td>2</td><td>x</td><td></td></tr> </table>		1	2	1		x	2	x		 $2 \leftarrow 1$ $1 \leftarrow 2$
	1	2								
1		x								
2	x									

If there is no relationship between the parts (or if it is not possible to connect two parts) no type of relation is assigned and the corresponding M_k matrix field stays empty. The Tab. 2 presents forms of the directed graph (digraph) for the two sub-assemblies mentioned earlier (Fig. 1), as well as the relationship matrix and all the possible connections between the components.

Table 2 Directed graph and relationship matrix of analysed sub-assemblies

Sub-assembly 1																										
	<table border="1"> <tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>1</td><td></td><td>x</td><td>x</td><td>x</td></tr> <tr><td>2</td><td>x</td><td></td><td></td><td>x</td></tr> <tr><td>3</td><td>x</td><td></td><td></td><td></td></tr> <tr><td>4</td><td>x</td><td>x</td><td></td><td></td></tr> </table>		1	2	3	4	1		x	x	x	2	x			x	3	x				4	x	x		
	1	2	3	4																						
1		x	x	x																						
2	x			x																						
3	x																									
4	x	x																								
Sub-assembly 2																										
	<table border="1"> <tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>1</td><td></td><td>x</td><td>x</td><td></td></tr> <tr><td>2</td><td>x</td><td></td><td>x</td><td>x</td></tr> <tr><td>3</td><td>x</td><td>x</td><td></td><td>x</td></tr> <tr><td>4</td><td></td><td>x</td><td>x</td><td></td></tr> </table>		1	2	3	4	1		x	x		2	x		x	x	3	x	x		x	4		x	x	
	1	2	3	4																						
1		x	x																							
2	x		x	x																						
3	x	x		x																						
4		x	x																							

Table 3 Reduced matrix connections record in analysed sub-assemblies

Sub-assembly_1					Sub-assembly_2				
	1	2	3	4		1	2	3	4
1			x		1		x		
2	x				2	x			x
3	x				3		x		
4	x				4		x		

The large size of the M_k matrix resulting from the product complexity and large number of defined connections adversely affect the time of solving the task. Therefore, it is assumed that only connections practically feasible will be defined in the matrix. Remaining connections resulting only from the contact between two parts and whose realisation is irrelevant in installation process can be omitted (excluded) in further discussion. For the sub-assembly 1 such connections are e.g. $2 \leftarrow 4$, $4 \leftarrow 1$, and for the sub-assembly 2: $1 \leftarrow 3$ and $3 \leftarrow 4$. The Tab. 3 presents filled structure design matrix for both sub-assemblies after elimination of such connections.

In relationship matrices mapping relations between components of the sub-assemblies 1 and 2 the number of connections for further analysis is reduced from 8 to 4 and from 10 to 5 respectively.

3.2 Evaluation of the components connection

To evaluate a connection of two parts information from the literature [2] was applied together with q_a indicator. It was developed on the basis of experts' knowledge and multiple analyses conducted in actual companies, and described in publications [2, 3, 5]. The indicator was used to assess the set of connections defined earlier in the form of M_k matrix in order to evaluate assembly sequences and find the most propitious ones in the generated set. Moreover, it is assumed that it is going to be used to obtain information on the degree of complexity of the analysed structure and its component parts.

The possibility of defining values other than in the original study has been introduced. The values serve to evaluate particular components of the q_a indicator. This gives a chance to adjust the assessment with the use of q_a indicator to the specific conditions of a particular company, in which literature indicators would be wrongly applied for various reasons. In addition, the assessment value could be represented by cost or connection realisation time, which would facilitate defining of sequences characterised by the shortest time or the lowest realization cost. The components of $q_a = h_p \cdot f_p$ [2] indicator:

- characteristics of passing and catching the component marked as h_p ,
- characteristics of the process of combining the components f_p .

Indicator $f_p = A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G \cdot H$ takes into account:

- A - correctness of combining the parts in terms of the assembly function,
- B - necessity of precise mutual positioning of two parts to be combined,
- C - orienting of the parts to be combined,
- D - direction of combining the parts,
- E - type of connection, it depends on the contact surface between them,
- F - constraints of access and / or control of the connection,
- G - alignment and other possible obstacles,
- H - resistance of the parts combining [2].

3.3 Defining the constraints

Determination of the correct assembly sequences requires appropriate precedence constraints. They are related to the set of connections recorded in the M_k matrix. Each connection can be assigned to one of three designators:

- starting connections (p_s) - connections of two parts from which the creation of the assembly sequence variants of the product starts,
- connection 'skip' (p_p) - this connection is not taken into account when generating the assembly sequence variants of the product,

- blocking connection (p_b) - connection which prevents or limits getting a complete assembly in the later course of the assembly process.

The first type of constraints (starting connection) is used predominantly to define base components and parts from which the assembly sequence formation starts.

Connections of the 'skip' type are defined in the case of reduction of a generated feasible assembly sequences set. This constraint can help to exclude resulting sequences with unfavourable sub-sequences.

The last of the constraints, and the most important one, is blocking connection, which has a direct influence on generating correct order of combining the parts, in terms of the selection completeness. This constraint is characteristic of those preceding connections, which prevent the realisation of the connection for which they are defined. This way the possibility of incorrect sequence when combining the parts is eliminated. It is assumed the blocking connections need to be defined with the operator 'and' (\wedge) and 'or' (\vee). In the first case, assigning the ' \wedge ' operator to the blocking connections ($p_{b1} \wedge p_{b2} \wedge \dots \wedge p_{bn}$) means that connection p_n , for which the blocking connections are defined, can be executed before every blocking connection is made. Thus, it is possible to make $n-1$ blocking connections before the connection p_n , for which n blocking connections were defined. If all the blocking connections are executed, it is impossible to

achieve complete assembly of the whole product because realisation of the connection p_n is blocked. In the second case, assigning the ' \vee ' operator to the blocking connections ($p_{b1} \vee p_{b2} \vee \dots \vee p_{bn}$) means that connection p_n , for which the blocking connections are defined, has to be executed before any of them. Even if one of the blocking connections is made, it is impossible to achieve complete assembly of the whole product because realisation of the connection p_n is blocked. Furthermore, it is possible to define blocking connection sequences (with the ' \wedge ' operator) separating them by the use of the ' \vee ' operator. For model sub-assemblies (Fig. 1) defined constraints are presented in Tab. 4. In this table 'x' stands for starting connections.

In the case of the first sub-assembly (Fig. 1) realisation of one of the connections $1 \leftarrow 4$, $1 \leftarrow 3$, $3 \leftarrow 1$ prevents subsequent assembly of part 2 with part 1. Similarly, realisation of one of the connections $1 \leftarrow 3$, $3 \leftarrow 1$ prevents subsequent assembly of part 4. The mentioned connections, which prevent assembling of part 2 and 4 are defined as blocking connections of 'OR' type. They cannot appear earlier in a proper assembly sequence. In the second sub-assembly (Fig. 1) only blocking connections of 'AND' type are defined. They appear only with the connection $2 \leftarrow 3$. Installation of components 1, 2, and 4 prevents the possibility of adding component 3.

Table 3 List of defined constraints for the sub-assemblies from figures 2 and 3

No.	Assembly connection	Starting connection	Blocking connection	
			'OR' ; \vee	'AND' ; \wedge
Sub-assembly 1				
1	$1 \leftarrow 2$	x	$1 \leftarrow 4 \vee 1 \leftarrow 3 \vee 3 \leftarrow 1$	--
2	$1 \leftarrow 3$	--	--	--
3	$1 \leftarrow 4$	--	$1 \leftarrow 3 \vee 3 \leftarrow 1$	--
4	$3 \leftarrow 1$	--	--	--
Sub-assembly 2				
1	$1 \leftarrow 2$	x	--	--
2	$2 \leftarrow 1$	--	--	--
3	$2 \leftarrow 3$	--	--	$(1 \leftarrow 2 \wedge 2 \leftarrow 4) \vee (4 \leftarrow 2 \wedge 2 \leftarrow 1)$
4	$2 \leftarrow 4$	--	--	--
5	$4 \leftarrow 2$	x	--	--

3.4 Algorithm for generating feasible sequences

The proposed algorithm for determining and evaluating the assembly sequences allows generation of all permissible variants for assembly sequences with simultaneous evaluation. It belongs to the category of classic (scientific) methods. In Fig. 2 the algorithm is presented in a schematic form, along with its particular steps.

In the algorithm three databases have been distinguished. The first of them contains data related to the product structure and relations between its components. Directly from the database – 1 a list of possible connections is created. The first step of the algorithm is to choose the first available connection from the starting connections list and create an assembly subsequence from its components. At the same time, when selecting a starting connection, its evaluation from the database – 2 is taken. This database contains information pertaining to evaluation of all the relations between the connections' components. This subsequence

is recorded as the M_{K+1} matrix, which decreases the size of the M_K matrix by 1. Relations recorded in the M_K matrix are changed into the form of the M_{K+1} matrix and constraints for the current subsequence are checked. All the constraints (connections of 'skip' type, blocking connections of 'OR' and 'AND' type) are recorded in the database 3. If there are any constraints, the current sequence is excluded from further consideration. If the constraints allow continuous building of the assembly sequence, more components are added. Subsequences of a higher order are created until a complete sequence meeting all the constraints is built. Produced sequences are then recorded and the starting connection used in the process is deleted from the list of available connections. Next, the algorithm chooses another available starting connection and the process of sequences creation is repeated. After every starting connection is used a set of all the possible assembly sequences is received.

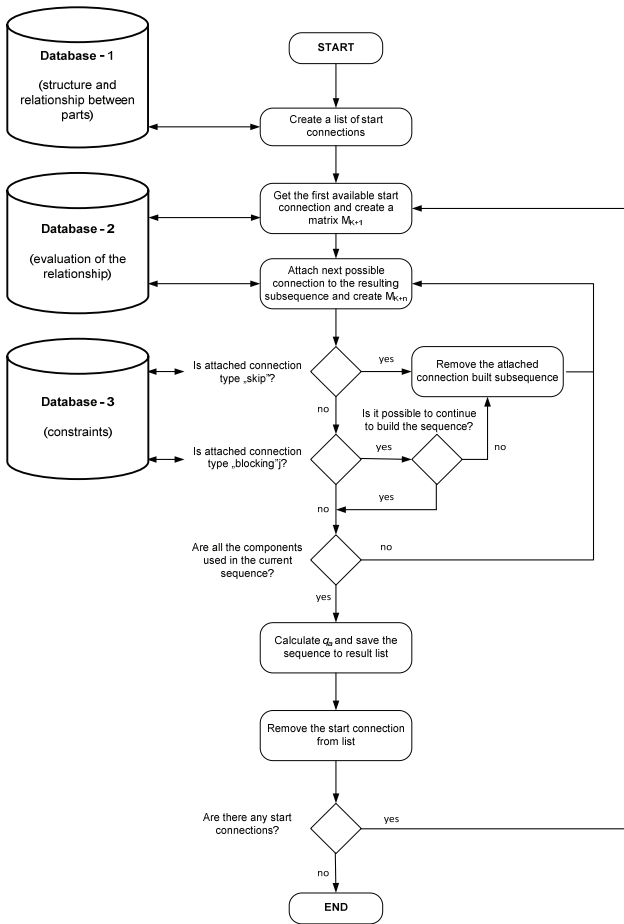


Figure 2 Algorithm for generating feasible assembly sequences

Subsequent steps of the proposed algorithm for designating a set of permissible sequences of the sub-assembly₁ from Fig. 1 are presented schematically in Fig. 3. The next steps of the algorithm are shown in the matrix of structure design mapping the relations between the components of this sub-assembly.

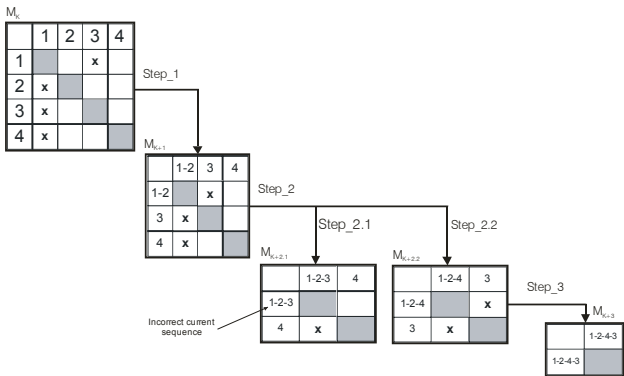


Figure 3 Algorithm functioning on the basis of the sub-assembly₁

The first available starting connection in the sub-assembly₁ is 1←2, which is recorded in the subsequence (1←2). The reduced M_{K+1} matrix of the 3×3 size containing the current subsequence is a result of Step₁ (Fig. 5). In the next step it is possible to add components 3 and 4 to the existing subsequence (due to the relations in the M_{K+1} matrix): (1←2); 1←3 or (1←2); 1←4. However, the first possibility is eliminated by the blocking connection 1←3, whose realisation before adding component 4 is impossible (Tab. 4). Thus, there is

only one way to join another element (component 4), which results in the subsequence ((1←2); (1←4)) recorded in the $M_{K+2,2}$ matrix of 2×2 size.

The last step of the algorithm for the analysed example is adding the last element (component 3) to the present subsequence, which leads to the complete sub-assembly₁. It is recorded in the M_{K+3} matrix of 1×1 size as the sequence (((1←2)(1←4))(1←3)), or 1-2-4-3.

The following Fig. 4 presents consecutive steps of operations in the matrix of structure design of the sub-assembly₂, aimed at generating feasible sequences of its installation.

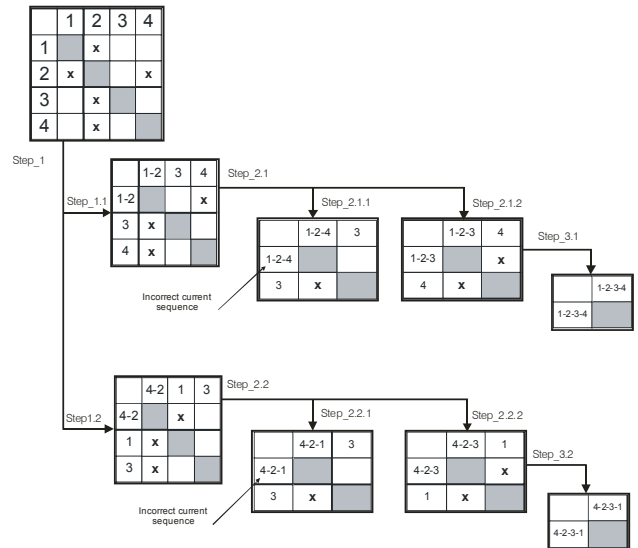


Figure 4 Algorithm functioning on the basis of the sub-assembly₂

In the first step there are two options for choosing the starting connection: 1←2 and 4←2. When the connection 1←2 is chosen, the subsequence (1←2) is created and recorded in the form of the $M_{K+1,1}$ matrix of 3×3 size. Then, there is a possibility of adding component 3 or 4. Adding component 4 is excluded because connections 2←4 and 1←2 block the possibility of further joining of part 3. Hence, there exists only one way to add component 3 to the subsequence (1←2) and record the current subsequence ((1←2); (2←3)) in the form of the $M_{K+2,1,1}$ matrix of 2×2 size. The last element to be joined is component 4.

The resulting sequence (((1←2)(2←3))(2←4)) is recorded in the $M_{K+3,1}$ matrix of 1×1 size. The sequence for the starting connection 4←2 is created in an analogous way.

4 Conclusion

Due to the extensive influence of assembly work on the cost and quality of machines and mechanical devices, a constructor should have at his disposal an efficient tool for planning the proper assembly sequence and evaluating a designed construction in terms of assembly requirements. He should be able to choose the most advantageous sequence of assembly operations. Hence, design for assembly should be included concurrently in the design process. Meeting these conditions will prevent constructions not following assembly process requirements.

The following article presents the method designed to plan and analyse assembly sequences in design engineering oriented to design for assembly. Its practical application is going to be presented in the second part of the article.

5 References

- [1] Anderson, D. M. Design for Manufacturability & Concurrent Engineering, CIM Press, 2010.
- [2] Booker, J. D.; Raines, M.; Swift, K. G. Designing Capable and Reliable Products, Butterworth-Heinemann, 2001.
- [3] Boothroyd, G.; Dewhurst, P. Design for Assembly, Boothroyd Dewhurst Inc., Wakefield, 1991.
- [4] Eder, E. W.; Hosnedl, S. Introduction to design engineering. Systematic creativity and management, Taylor & Francis Group, London, UK, 2010. DOI: 10.1201/b10536
- [5] Whitney, D. E. Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development, Oxford University Press, 2004.
- [6] Sasiadek; M.; Rohatyński, R. A method of computer aided design for assembly. // Proceedings of the Design methods for industrial practice Conference, Oficyna Wydaw. Uniwersytetu Zielonogórskiego, (2008), pp. 119-126.
- [7] Sasiadek, M. A method of mechanical assembly sequence planning with accordance to the DFA principles. // Modelování a optimalizace podnikových procesů - MOPP, 13. ročník mezinárodního semináře. Plzeň, Czech Republic, (2011), pp. 8 (CD-ROM).

Author's address

Ing. Michal Sasiadek Ph.D.

University of Zielona Góra, Faculty of Mechanical Engineering,
Institute of Management and Production Engineering
ul. Prof. Szafrana 4, Zielona Góra, 65-516, Poland
E-mail: m.sasiadek@iizp.uz.zgora.pl