Marek MAGIERA

AGH University of Science and Technology

doi:10.15199/48.2018.04.42

A no-wait scheduling method for assembly operations concerning multi-option electric devices

Streszczenie. *Przedstawiono monolityczną metodę szeregowania operacji montażowych dotyczących sprzętu elektrycznego uwzgledniającego indywidualne wymagania odbiorców. Metodę zbudowano dla linii montażowych z maszynami równoległymi, bez buforów międzyoperacyjnych. Skonstruowane modele matematyczne zadań programowania całkowitoliczbowego, w których uwzględniono opisane konfiguracje linii montażowych, przeznaczone są do budowy harmonogramów montażu zgodnie z ideą szeregowania „bez czekania". Zamieszczono wyniki eksperymentów obliczeniowych – porównano m.in. dwie różne organizacje przepływów produktów dotyczące sztywnych i alternatywnych marszrut montażu. (**Metoda szeregowania "bez czekania" operacji montażowych dotyczących wielowariantowego sprzętu elektrycznego**).*

Abstract. *The monolithic method of no-wait scheduling is presented. The individual requirements of recipients of the electric devices are regarded. The method is for assembly lines with parallel machines, without intermediate buffers. The mathematical models of integer programming are constructed for this configuration of assembly lines – for no-wait scheduling. The results of computational experiments with the proposed method are presented – fixed and alternative assembly routes are compared, among others.*

Słowa kluczowe: szeregowanie operacji, harmonogramowanie montażu, maszyny równoległe, programowanie całkowitoliczbowe, produkty wielowariantowe.
Keywords: scheduling, assembly scheduling, parallel machines, integer programming, multi-option products.

Introduction

The method presented herein concerns scheduling of assembly operations – the operations are distributed in space (between the machines), and in time [1]. The result of this operation distribution is a schedule of performing assembly operations. The developed method is monolithic, as both tasks (distribution of operations between the machines, designation of the operation starting time) are solved at the same time. The utilization of such a single-level approach to task-solving and integer programming guarantees that an optimum solution will be defined. This is a significant feature of the constructed method. Task solutions are not burdened with deviation from the optimum, the resulting schedules are the shortest and take into account the assembly line setup and the flow of products along this line. It is also recommended to complete the optimization as soon as at the stage preceding the production process - at the stage of the product design [2], which influences the quality of the product.

The benefits of applying the integer programming in assembly scheduling are described, for instance, in works [3], [4], and [5]. It is applied both in the monolithic methods (e.g. [5]), and hierarchical, that is multilevel methods (e.g. [3], [4]).

The developed method is distinguished by the organization of the product flow along the assembly line – scheduling with constraints no-wait method was used [6]. This means that the breaks between the assembly operations for the given product are permitted only in order to complete transport operations. Such a product flow organization is usually used on assembly lines which are not equipped with intermediate buffers - local storages where the products can wait for further operations. This results in reduced costs of the assembly system. An alternative organization of the product flow along an assembly line without buffers is a flow where the machines can be blocked by products which wait for further operations. In such a case, the machines act as buffers – like, e.g., in certain methods described in work [4]. Whereas the setup of an assembly line equipped with intermediate buffers of limited capacities was taken into account in the operation scheduling method described in work [7], by the author of this article. This method applies to multi-option products. Taking such products – characterized below – into account is another distinctive feature of the method presented herein.

Multi-option products are diversified, due to the individual requirements of the customers. These requirements may differ in terms of technical parameters (e.g. motor power), dimensions or, for instance, extra equipment. So a product of the given type can be assembled in different variants.

The following chapters describe the idea of the method, constructed mathematical models concerning this method, as well as results of computational experiments where these mathematical models were used.

Idea and application of the method

The method is used to solve the following task: to build the shortest possible operation schedule, while having specific parameters which describe a one-direction assembly line with parallel machines. There are no intermediate buffers between the machines, making it necessary to use the no-wait scheduling. An example setup of such a line, also referred to as a hybrid flow system [8], is shown in Figure 1.

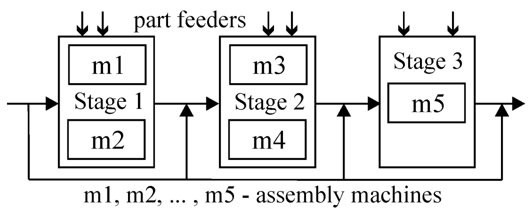

Figure 1. Example of diagram of assembly line with parallel machines and without intermediate buffers

Every stage is a set of machines which work in parallel. As visible in Figure 1, some stages can be omitted. The product which flows through the given stage loads only the assembly machine. In such a system, multi-option electric devices of various types can be assembled. Each product is assembled in accordance with the given assembly sequence. The first operation consists in installing the base part in the assembly jig. Further assembly operations consist in attaching a single component or a subassembly to the parts put together earlier. Components are fed from the part feeders, located at individual stages. The developed method includes two variants of the part feeders arrangement:

– the part feeder, which allows to perform the given type of operation (attachment of the given component) is located in exactly one stage. In such a case, the path of the product flow along the assembly line is called the fixed assembly route.
– the part feeder, which allows to perform the given type of operation is located at least in one stage. In such a case, the path of the product flow along the assembly line is called the alternative assembly route. Yet, for each multi-option product, exactly one route is outlined. The operation of the given type, which applies to several various products, can be performed on different machines.

The idea of the monolithic method of solving the task described above is presented in the block diagram in Figure 2.

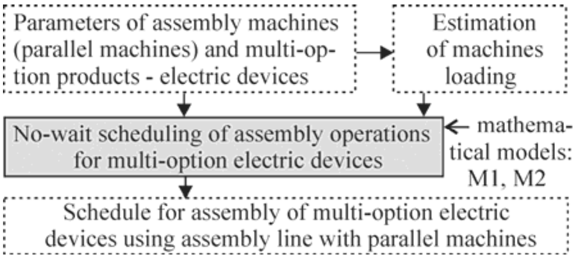

Figure 2. Block diagram of the method for assembly line with parallel machines and no-wait scheduling

Input data include parameters and sets which describe the assembly line and the multi-option electric devices. Among these data, there are two parameters which inform about availability of individual machines. This allows to take into account the scheduled machine downtimes, e.g. for the purpose of repair or maintenance. The data which describe the machine's availability are used in the machine load estimation procedure. The parameter which specified the machine load estimation, along with the other ones, are then used in the mathematical models of the integer programming tasks. The following models were built:

– the mathematical model M1 for no-waiting scheduling of assembly operations in assembly lines with parallel machines and fixed assembly routes;
– the mathematical model M2 for no-waiting scheduling of assembly operations in assembly lines with parallel machines and alternative assembly routes.

Data and analysis of input parameters

The list of all the sets, parameters and variables used in the linear mathematical models M1 and M2 built for the method is given in Table 1.

Table 1. Sets, parameters and variables used in the models

Basic sets:		
I	–	the set of assembly machines; $I = \{1, \ldots, M\}$;
J	–	the set of assembly operations; $J = \{1, \ldots, N\}$;
K	–	the set of types of products; $K = \{1, \ldots, W\}$;
L	–	the set of periods; $L = \{1, \ldots, H\}$;
S	–	the set of product indices; $S = \{1, \ldots, U\}$;
V	–	the set of stages; $V = \{1, \ldots, A\}$;
Others sets:		
D	–	the set of pairs (i, v), in which machine i is placed in the stage v;
V_j	–	the set of the stages in which the machine are capable of execution of operation j;
J^C	–	the set of tasks which require using the part feeder, $J^C \subset J$;
J^1	–	the set of basic operations $J^1 \subset J$;
J^2	–	the set of additional operations, that distinguished multi-option products, $J^2 \subset J$;
O^1	–	the set of pairs (k, j), in which the basic operations $j \in J^1$ is required for type of product $k \in K$;
O^2	–	the set of pairs (s, j), in which the additional operation $j \in J^2$ is required for product $s \in S$;
R^1	–	the set of elements (k, r, j), in which operation j is executed immediately before task r, $r, j \in J^1$ and operations are required for type of product k;
R^2	–	the set of elements (s, r, j), in which operation j is executed immediately before task r; $r, j \in J$ and operations are required for product s, one of this operation or both belong to set J^2;
P	–	the set of pairs (s, k), in which product s is type k;
Parameters:		
a_{vj}	–	working space of machine in stage v required for execution of operation j;
b_v	–	total working space of the machine placed in stage v;
g_{ev}	–	transport time between machines in stage e and in stage v;
m_v	–	number of machines in stage v;
p^1_{jk}	–	processing time for basic operation $j \in J^1$ for type of product k;
p^2_{js}	–	processing time for additional operation $j \in J^2$ for product s;
n_{il}	=	1, if machine i is available during period l, otherwise $n_{il} = 0$;
Variables:		
x_{vj}	=	1, if type of assembly operation j is assigned to stage v, otherwise $x_{vj} = 0$;
y_{ijsl}	=	1, if product s is assigned to machine i to perform assembly operation j in period l, otherwise $y_{ijsl} = 0$;

Allocation of the assembly machines to individual stages is defined by set D (Table 1). The setup of the assembly line shown in Figure 1 may be coded as follows: D = {(1,1), (2,1), (3,2), (4,2), (5,3)}. For instance, let us assume that in such a hybrid system a product type k = 1 is installed in variants s = 1 and s = 2. Assignment of individual variants to the given product types, in accordance with designation from Table 1, can be notated as: T = {(1,1), (2,1)}. The restrictions regarding the order of performing operations for the example product k = 1 are illustrated in Figure 3.

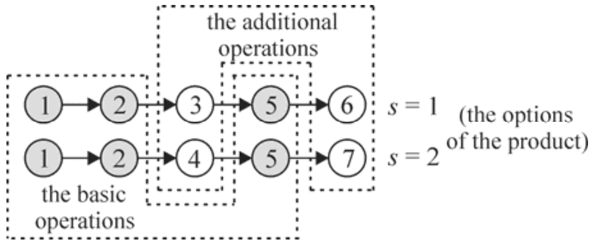

Figure 3. Example of restrictions on the order of operations in the case of the product that is assembled in two options

As shown in Figure 3, for both variants (s = 1 and s = 2) it is necessary to perform the same operations which pertain to set J^1 = {1, 2, 5}. There are also operations the performance of which grants specific properties to the products – operations that belong to the set J^2 = {3, 4, 6, 7}. The multi-option electric appliances are diversified by operations 3 and 4 (e.g. various motor power), as well as by operations 5 and 6 (e.g. various casings). The sets which apply to the sequential restrictions on operation performance are as follows: R^1 = {(1, 1, 2)}, R^2 = {(1, 2, 3), (1, 3, 5), (1, 5, 6), (2, 2, 4), (2, 4, 5), (2, 5, 7)}.

The data listed in Table 1 include parameters n_{il} , which describe the planned downtimes of individual machines i at periods l. In order to estimate the number of the periods H, it is suggested to use the procedure described in work [5], which provides for the limited availability of the machines.

Mathematical models: M1 and M2

In order to solve the described task of no-wait scheduling for operations concerning multi-option electric devices, performed on assembly lines with parallel machines, linear mathematical models of integer programming tasks were built: M1 (concerns fixed assembly routes), M2 (concerns alternative assembly routes). These models are used to determine the shortest possible assembly schedules.

The mathematical models M1 and M2:

(1) Minimize: $\sum_{i\in I}\sum_{j\in J}\sum_{s\in S}\sum_{l\in L} l\, y_{ijsl}$

(2) Subject to: $\sum_{i\in I}\sum_{j\in J} y_{ijsl} \le 1;\ l\in L; s\in S$

(3) $\sum_{i\in I_j}\sum_{l\in L:n_{il}=1} y_{ijsl} = p^1_{jk};\ j\in J^1; (s,k)\in P : (k,j)\in O^1$

(4) $\sum_{i\in I_j}\sum_{l\in L:n_{il}=1} y_{ijsl} = p^2_{js};\ j\in J^2; s\in S : (s,j)\in O^2$

(5) $\sum_{j\in J}\sum_{s\in S} y_{ijsl} \le n_{il};\quad i\in I;\ l\in L$

(6) $y_{ijsl} + y_{\tau jsf} \le 1;\ i,\tau\in I; \tau\ne i; j\in J; l,f\in L; s\in S$

(7) $y_{ijsl} + y_{\tau rsf} \le 1;\ (i,v),(\tau,v)\in D;\ \tau\ne i;\ r,j\in J;\ r\ne j;$
$l,f\in L;\ s\in S$

(8) $x_{vj} \ge y_{ijsl};\ (i,v)\in D;\ j\in J;\ l\in L;\ s\in S$

(9) $a_{vj}x_{vj} \le b_v m_v;\ j\in J^c;\ v\in V$

(10) $\sum_{j\in J} x_{ij} = 1;\ v\in V$ - for the model M1 only

(11) $x_{vj} = 0;\ j\in J;\ v\notin V_j$

(12) $iy_{ijsl} \ge \tau y_{\tau rsf} \ge -(H+1)(1-y_{ijsl});$
$i,\tau\in I; (s,k)\in P; l,f\in L; (k,r,j)\in R^1 \vee (s,r,j)\in R^2$

(13) $ly_{ijsl} - fy_{\tau rsf} \ge 1 + g_{\varepsilon v} - (H+1)(1-y_{ijsl}); (\tau,\varepsilon),(i,v)\in D;$
$l,f\in L;\ ((s,k)\in T; (\kappa,r,j)\in R^1; \kappa=k)\vee(s,r,j)\in R^2$

(14) $ly_{ijsl} - fy_{\tau rsf} - 1 \le g_{\varepsilon v} + \sum_{\eta\in L: f<\eta<l}\left(y_{ijs\eta} + y_{\tau rs\eta}\right) +$
$+(H+1)(1-y_{\tau rsf}); (\tau,\varepsilon),(i,v)\in D; (s,k)\in P;$
$l,f\in L; f<l; r,j\in J; (k,r,j)\in R^1 \vee (s,r,j)\in R^2$

(15) $x_{vj}, y_{ijsl} \in \{0,1\};\ i\in I; j\in J; s\in S; l\in L; v\in V$

The shortest possible assembly schedules using mathematical models M1 and M2 are determined by minimizing the value of sum (1). As a result, the assembly operation completion times on individual machines are the shortest. The constraints built for models M1 and M2 ensure: (2) performance of no more than one operation for each product in any given moment; (3) distribution of all the basic operations among the machines; (4) distribution of all the additional operations among the machines; (5) performance of no more than one operation at a time by the machine, if in this period the machine is available for operation performance; (6) indivisibility of operations between the machines – the given operation is performed on no more than one machine; (7) product flow through at least one machine of the given stage; (8) designation of allocation of individual operation types to the machines, as a result of which the distribution of the part feeders between the machines is known (if the operation consists in attaching a components taken from the feeder); (9) taking into account the limited working area of the machines – the part feeders must fit in a specific working area; (10) allocation of each type of assembly operation to one stage only – this restriction applies only to model M1; (11) elimination of assigning the operations to wrong machines; (12) unidirectionality of the product flow along the assembly line; (13) taking into account the given sequence of operations (assembly sequence) and reservation of time for transporting the products between the machines; (14) continuity of operations and no-wait scheduling; this means that the pauses between operations for individual products are used only for transporting these products between the machines which belong to different stages; (15) binarity of the decision-making variables.

Computational experiments, verification of the method

The described method was verified. For this purpose, computational experiments were carried out. Additional objectives of these experiments included:

- comparison of the lengths of the schedules determined in the case of fixed and alternative assembly routes;
- comparison of the lengths of the schedules determined in the case of no-wait scheduling and scheduling in assembly lines equipped with intermediate buffers with limited capacities;

The length of the schedule created using the mathematical model *M* is determined on the basis of the equation (16):

(16) $C^M_{\max} = \max_{i\in I, j\in J, s\in S, l\in L} ly_{ijsl}$

The experiments were carried out on five groups of test tasks. For each group, 25 tasks were solved. The parameters of the test tasks are listed in Table 2. A GUROBI solver was used [9].

Table 2 also presents the results of the computational experiments. They are the average values of the following indexes:

- α – index intended for comparing the schedule lengths for fixed routes (designates using the M1 model) and the alternative routes (designated using the M2 model). It is estimated based on equation (17):

(17) $\alpha = \dfrac{C^{\mathrm{M1}}_{\max} - C^{\mathrm{M2}}_{\max}}{C^{\mathrm{M2}}_{\max}} \cdot 100\%$

- β_1, β_2 – indexes used to compare the length of the designated schedules with the schedules for assembly lines with parallel machines equipped with intermediate buffers of limited capacities . In this case, the constraint thanks to which the pauses between the operations for the given product are used only to transport this product between various stages was omitted. Therefore, the constraint typical of the no-wait scheduling is rejected. If the operation concerning the given product cannot be performed on another machine (it is loaded by another product), the product waits in a buffer for another operation to be performed. This buffer is located directly before the stage in which the next operation concerning this product is to be performed. The mathematical models concerning the line with the parallel machines and the buffer are described in work [7]. Indexes β_1, β_2 were defined in equations (18), built similarly to equation (17). The β_1 index applies to fixed assembly routes. It is used to compare the lengths of the schedules determined using the M1 model and the model which applies the assembly lines with buffers, marked as MBF. Whereas index β_2 concerns alternative assembly routes. This index allows to compare schedules determined using the models M2 and MBA – a model applicable to buffers and alternative assembly routes. The original designations of the MBF and MBA mathematical models presented in work [7] are M1 and M2.

$$(18)\quad \beta_1 = \frac{C_{\max}^{\mathrm{M1}} - C_{\max}^{\mathrm{MBF}}}{C_{\max}^{\mathrm{MBF}}} \cdot 100\%; \quad \beta_2 = \frac{C_{\max}^{\mathrm{M2}} - C_{\max}^{\mathrm{MBA}}}{C_{\max}^{\mathrm{MBA}}}$$

Table 2. Parameters of groups of test tasks and average values of indexes [%]

Group	Parameters of test tasks						Indexes		
	A	M	N	W	H	S	α	β_1	β_2
1	2	4	10	4	16	8	4,7	8,5	5,2
2	3	5	12	4	18	8	5,1	7,9	5,1
3	3	6	14	5	20	10	6,4	6,2	4,9
4	4	6	16	5	24	10	7,6	5,7	4,2
5	4	8	18	6	24	12	7,8	5,9	3,8

Numbers of: A - stages, M - machines, N - types of assembly tasks , W - types of products, H - periods S – multi-option products

The schedules for fixed routes, determined using the M1 model, were almost 8% longer than the schedules which include alternative routes, created on the basis of the M2 models. This is confirmed by the values of the α index presented in Table 2. Compared to systems with intermediate buffers, the schedules determined using the M1 and M2 models were longer – even by about 8% for fixed routes (index β_1), and by about 5% (index β_2).

A great benefit is to compare the developed method with hierarchical method. An example of a two-level method is described in the monograph [4] (pp. 117–128). The top-level is a machine loading, i.e., allocation of operations among the machines. The base-level is a no-wait scheduling. Compared to this hierarchical method, the schedules determined using the method presented in the paper were shorter by about 5–7% for fixed routes, and by about 4–5% for alternative routes.

Conclusion

An obvious benefit of no-wait scheduling is avoidance of the costs of using intermediate buffers, or using the machines as buffers. The products flow along the assembly system in a continuous way. The breaks between the assembly operations are uses solely to transport the products between the machines. Yet, the no-wait scheduling may slightly extend the schedule length, as evidenced by the results of the computational experiments presented herein.

Another benefit of the developed method is that is takes into account the requirements faced by the manufacturers of electric devices. These manufacturers must take into account the individual requirements of their customers. The result is assembly of multi-option products made to order. This aspect was emphasized not only in the created mathematical models concerning the developed method, but also in the data structure. The described structure of data and variables has a positive impact on the computational complexity of the problems being solved – there is a distinction between the basic and additional operations, which diversify the multi-option electric devices.

Of course, every mathematical model is a simplification of reality, which is described using mathematical relationships. In the constructed models, there was an attempt to reflect the assembly of multi-option electric devices to the highest extent possible. For this purpose, for instance, the scheduled downtimes of the assembly machines were taken into account, and assembly operations which require utilization of part feeders were singled out.

The observed development of computer technology, software and algorithms promotes the development of methods based on binary programming which include the presented concepts of no-wait scheduling. The discrete optimization packages feature higher computing power, which allows solving of real problems of increasing sizes and significant shortening of computational time.

Author: *dr inż. Marek Magiera, AGH University of Science and Technology, Faculty of Management, Department of Operation Research, AL. Mickiewicza 30, 30-059 Kraków, Poland, E-mail: mmagiera@zarz.agh.edu.pl*

REFERENCES

[1] Pochet Y., Wolsey L.A., Production planning by mixed integer programming. *Series in Operations Research and Financial Engineering*, Springer, New York (2006)
[2] Bobalo Y., Nedostup L., Kiselychnyk M., Melen M., Optimization of processes designed for ensuring the quality and reliability of electronics using cumulative models of defectiveness and total production expenses. *Przegląd Elektrotechniczny* (Electrical Review), 92 (2016) n. 1, 155-158
[3] Sawik T., Loading and scheduling of a flexible assembly system by mixed integer programming. *European Journal of Operational Research*, 154 (2004), n.1, 1-19
[4] Magiera M., Wybrane metody planowania przepływów produktów przez linie produkcyjne i łańcuchy dostaw (Selected methods of planning product flows through production lines and supply chains). Wydawnictwa AGH, Kraków (2016)
[5] Magiera M., Monolityczna metoda planowania montażu dotyczącego wielowariantowego sprzętu elektrycznego i elektronicznego (A monolithic method of assembly planning for multi-option electric and electronic devices). *Przegląd Elektrotechniczny* (*Electrical Review*), 93 (2017), n. 8, 192-195.
[6] Pindeo M.L. Scheduling. Theory, Algorithms, and Systems. Springer, New York (2008)
[7] Magiera M., Monolityczna metoda harmonogramowania montażu wielowariantowych produktów w hybrydowych systemach przepływowych (A monolithic method of scheduling for assembly of multi-option products in hybrid flow shop systems), *Technologia i Automatyzacja Montażu*, 2 (2017), 13–18
[8] Ribas I., Leinstein R., Framinan J.M., Review and classification of hybrid flowshop scheduling problems from a production system and a solution procedure perspective. *Computers & Operations Research*, 37 (8), (2010),1439-1454
[9] www.gurobi.com (October 2017)