ͻmper

AN OPTIMUM SCHEDULING METHOD FOR MULTI-OPTION PRODUCT FLOWS THROUGH PRODUCTION LINES WITHOUT INTERMEDIATE BUFFERS

Marek Magiera

AGH University of Science and Technology, Faculty of Management, Chair of Operations Research, Poland

Corresponding author:

Marek Magiera
AGH University of Science and Technology
Faculty of Management
Chair of Operations Research
Al. Mickiewicza 30, 30-059 Kraków, Poland
phone: +48 12 6173992
e-mail: mmagiera@zarz.agh.edu.pl

ABSTRACT

The presented method is constructed for optimum scheduling in production lines with parallel machines and without intermediate buffers. The production system simultaneously performs operations on various types of products. Multi-option products were taken into account – products of a given type may differ in terms of details. This allows providing for individual requirements of the customers. The one-level approach to scheduling for multi-option products is presented. The integer programming is used in the method – optimum solutions are determined: the shortest schedules for multi-option products. Due to the lack of the intermediate buffers, two possibilities are taken into account: no-wait scheduling, possibility of the machines being blocked by products awaiting further operations. These two types of organizing the flow through the production line were compared using computational experiments, the results of which are presented in the paper.

KEYWORDS
Flexible Manufacturing Systems, integer programming, linear programming, no-wait scheduling, hybrid flow shop, makespan, decision making.

Introduction

The purpose of the paper is to present a method intended for building production schedules in production lines devoid of intermediate buffers. Therefore, this method applies to production systems without local storages, where the products can await further operations. Operations concerning individual products are carried out in consecutive stages of a unidirectional flow system. The production lines are hybrid – each stage contains a set of machines operating in parallel. In a hybrid flow shop (HFS) there are two or more identical production machines in one or more of the processing stages. Some authors refer the hybrid flow shop as a flexible flow shop or flexible production line [1].

The main distinctive feature of the developed method is that it provides for multi-option products.

A multi-option product is characterized by the fact that individual pieces of a specific type (variants) differ in terms of details, e.g. extra finishing, additional handles, and differences in overall dimensions. Taking into account multi-option products is a response to requirements made by the market, characterized by significant competitiveness. To face this competitiveness means, for instance, to take into account individual requirements of the customers.

The method presented later on is used to determine the shortest possible schedules. This method is also distinguished by that it takes into account scheduled downtimes of the machines, e.g. for maintenance, retooling.

The method described in the paper is intended for planners who deal with production planning and control, particularly in the case of shorter time horizons. The engineers who prepare production on

production lines can use this method to determine the shortest schedules for production lines without intermediate buffers. Hence, the shortest schedules are determined in terms of the optimum criterion of schedule length. If a production plant uses another method which does not guarantee determination of an optimum solution, the method presented in this paper can be used for the purposes of comparison. First of all, it is possible to compare the lengths of the determined schedules, in order to determine the deviation of the optimum, as well as computation times.

The developed method is significant for the planners not only because it determines the shortest schedules. Another, very important advantage of this method is that it provides for multi-option products, as already mentioned in the paper. Thanks to these two advantages, the method can give an advantage in a competitive market. On one hand, optimum schedules are determined; on the other, individual requirements of the product recipients are taken into account – the products can be made in different variants. Another important fact is that the mathematical models concerning the method try to reflect the conditions of a production process as accurately as possible, e.g. by taking into account machine downtimes.

The motivation for developing the method described in the paper were the aforesaid benefits of the applied production scheduling concept. So far, the issues regarding production of multi-option products have not been broadly described. This article is a supplement in the scope of mathematical description of building production schedules regarding multi-option products.

According to the classification of mathematical problems [2], the method presented herein concerns production planning, as part of which task scheduling is performed. The scheduling which the developed method concerns is a part of the tasks solved during operating control, which is also referred to as operating planning [3]. Usually, it encompasses a short time horizon, and takes into account the current status of the production system. Scheduling sometimes also connected with planning regarding longer time horizons and seen as an integrated process of planning and scheduling. This can be seen, e.g., in publications [4] and [5].

The literature concerning scheduling for production systems is extensive. Soualhia et al. [6] presented an overview of IT tools used for task scheduling. An overview of the task scheduling in production hybrid systems, which this paper pertains to, was made by Ruiz and Vasquez-Rodriguez [7]. They analysed over

two hundred papers. Their analysis of the applied optimality criteria showed that the time criteria predominate, constituting over 90% of the applied criteria. Among these criteria, schedule length, also referred to as operation scheduling length, dominates. The criterion applied in the developed method concerns this very group of criteria. The second group of criteria included in scheduling are the cost criteria.

Yet, it should be emphasized that minimizing the schedule length also reduces production costs, as the time of completing a production order is shorter.

Production line scheduling makes use of optimum or approximate methods. The developed method is an optimum one. Optimum solutions were determined e.g. using integer programming. The use of mathematical programming in the developed method was inspired, e.g., with works [8] and [9]. These works show very good perspectives for using mathematical programming in production planning. This is the result of the observed development of software and computer technology.

Approximate methods are an alternative for optimum methods. These methods are used mostly for solving problems of relatively significant size. Approximate methods are characterized by short computing time. But an obvious downside of these methods is that they determine solutions encumbered with a certain deviation from the optimum. The rudiments of creating approximate methods and the issues regarding construction of approximate algorithms were described, e.g., by Gonzales [10]. Another disadvantages of the approximate methods include: premature convergence to the relative extremum, as well as stagnation in the search for solution sets. These disadvantages contributed to the development of hybrid methods. The idea of hybrid methods was described in detail in the paper [11]. The detailed overview of the hybrid methods was presented in the paper [12].

Scheduling employs one of two concepts: monolithic or hierarchical. The monolithic (single-level) concept is characterized by a global approach to the problem. This means that all the partial tasks are solved simultaneously. In the case of adopting the alternative approach to problem solving – the hierarchical (multi-level) method – the problem to be solved is divided into partial tasks. These tasks are assigned to individual levels of the method. E.g. in the case of building a production schedule, operations may be assigned to machines at the top level, while at the bottom level operations are assigned to machines. Examples of using this concept are presented, e.g. in papers [13] and [14]. Hierarchical methods are usually applied to problems of

relatively significant size. Dividing the problem into tasks solved consecutively results in solving of smaller-sized problems – assigned to individual levels of the method. The results of solving a lower-tier problem are input parameters for the task solved at the upper tier. Therefore, applying the hierarchical concept usually results in defining solutions encumbered with a certain deviation from the optimum solution.

The aforementioned downside of the hierarchical concept contributed to application of the monolithic concept in the developed method. This means that all the input data must be taken into account simultaneously. Problems solved with single-level approach (monolithic) are often large-sized. The size of the problem to be solved, constituting an integer being the measure of the input data quantity, has a significant impact on the computational complexity – memory and time [15]. Yet the aforesaid development of computer technology and software allow solving of problems of increasing sizes.

It should be emphasized that the monolithic method propose in this paper has a significant advantage – it enables creation of the shortest flow schedules (optimum) for various types of products through production lines. This was achieved by employing using the monolithic approach and using integer programming.

Another significant advantage of the developed method, which should also be pointed out, is dedicating this method to multi-option products. This required construction of a data structure which takes into account various variants of the products of a given type. The products were grouped by technological operations, some of which are identical, while some may be similar (in the case of individual product variants). Thus, the developed method concerns group technology [12]. Product grouping simplifies planning and control of production. Primary advantages of group technology include reduction of unit costs and flexibility of production.

Flexibility of production system which the method concerns is mostly related to flexibility of the production assortment. It is possible to perform operations concerning various types of products. The production system is also characterized by flexibility of the production volume. Production of small batches is also profitable. This is particularly important for multi-option products, which often vary in terms of details – depending on the customer's requirements. Production machines are also characterized by flexibility. The method described further on is also characterized by flexibility of the production routes. This means that operations of the same type can be performed on different machines which are capable of performing them. The aforesaid types of flexibility are typical of Flexible Manufacturing Systems (FMS) [13].

Scheduling is a field of production engineering, which is still developing. Not only are new concepts for solving complex problems developed, but new tasks are formulated, which are solved simultaneously to the creation of a new production schedule. One of these concepts is rescheduling. It consists in creation of a new schedule, usually if the original schedule can no longer be followed. Another reason for rescheduling is to provide for new, urgent orders. In the methods concerning rescheduling often, assignment of operations to machines regarding selected products is often retained, and the operation starting times for specific machines are re-determined only for certain products. This means that rescheduling can apply only to certain products. The issues concerning this trend is described, e.g. in studies [18] and [19]. This direction of research was mentioned, as many mathematical dependencies concerning mathematical models presented further on can be used for rescheduling. However, these models apply primarily to another direction of developing the scheduling studies – creation of schedules where products can be manufactured in short batches, often with piece production, on account of providing for individual requirements of the customers.

The detailed description of the developed method is included in the following sections of the paper. Section 2 is dedicated to the general description of the problem and the concept of its solution. Section 3 features the detailed presentation of the proposed method. The results of computational experiments used for verification of the developed method are presented in Sec. 4.

General description of the problem and monolithic method

The monolithic method concerns a unidirectional, multi-stage production line without intermediate buffers. Each product encumbers no more than one machine of the given stage. Certain stages may be omitted. This results from machine specialization, their capacity for performing operations. If in the case of a given product, a specific operation which can be performed at the given stage is not performed, this stage is omitted. A model setup of such a production line is shown in Fig. 1.

In the described production line, it is necessary to simultaneously perform operations concerning multi-option products of various types. Seeing as the line is

not fitted with intermediate buffers, two cases should be taken into account:

- no-wait scheduling – breaks between operations are intended solely for transporting the products between machines belonging to various stages [20];
- possibility of the machines being blocked by products awaiting performance of consecutive operations – the machines can act as buffers (local storages).
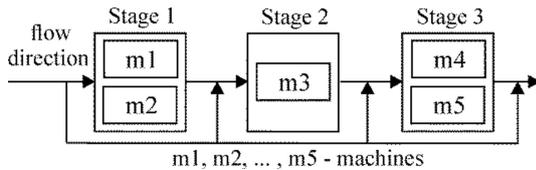

Fig. 1. Diagram of the hybrid production line without intermediate buffers.

Taking the aforesaid cases into account, it is necessary to build the shortest possible production schedules. When building these schedules, it is necessary to take into account the machinery stock setup, production capacities of the machines and constraints regarding the order of operations assigned to individual multi-option products. The schedules must be built for alternative production routes, which means that operations of a given type concerning different multi-option products do not have to be performed on the same machine – they can be assigned to various machines capable of performing these operations.

In order to solve the described task, it is proposed to use a monolithic method, employing two models of integer programming tasks. Below are the designations of these mathematical models: M1 – the mathematical model for no-wait scheduling; M2 – the mathematical model for possibility of the machines being blocked by products.

Figure 2 shows the block diagram of the monolithic scheduling method for multi-option product flows through production lines without intermediate buffers.
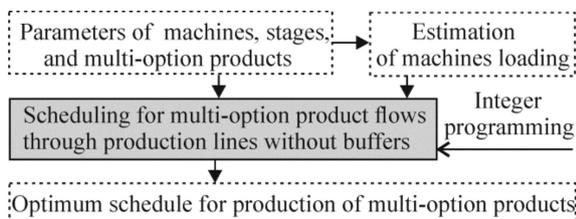

Fig. 2. Block diagram of the monolithic method.

The structure of data and variables, as well as mathematical models of the integer programming tasks, are presented in the following section.

Mathematical description of the method

Taking into account multi-option products required construction of a data set dedicated solely to these products. Table 1 presents a list of sets, parameters and variables concerning this structure.

Distinctive features of this method include taking into account limited availability of the machines and the related division of the estimated schedule into periods (intervals) l. The number of these periods (intervals) H can be estimated, e.g., using the procedure developed by the author of this article, presented in the paper [13].

Table 1 presents a set of stages containing machines able to perform operation j. This set was marked as V_j. This results from the setup of the production line which the described method concerns. This line includes specialized machines.

The aspect of products made in various variants is visible in the description of defined sets and parameters, provided in Tabl 1. Finishing operations concerning individual variants of the product types are listed in set O^2. Whereas operations concerning all the products of a given type, referred to as primary operations, are listed in set O^1. In a similar manner, two sets concerning order constraints are distinguished: Set R^1 describes the order of performing primary operations, while set R^2 includes operations which give the products their distinctive features, thanks to which the products are described as multi-variant. Distinction of these sets, as well as the concept of order constraint notation presented further on, have a positive effect on computational complexity.

An example of a multi-option product (sleeve) is shown in Fig. 3.

Product type $k = 1$ (the sleeve shown in Fig. 3a) is produced in different options $s = 1$, $s = 2$ and $s = 3$. The restrictions regarding the order of operations performance for the three options of the product $k = 1$ are presented in Fig. 3b. Assignment of options to the given product types (the sleeve) can be notated as: $T = \{(1,1),(2,1),(3,1)\}$. For three options of the product ($s = 1$, $s = 2$, and $s = 3$) it is necessary to perform the same operations which pertain to set $J^1 = \{1,2,3\}$. There are also operations the performance of which grants specific properties to the products – operations that belong to the set $J^2 = \{4,5,6,7,8\}$. The sets which apply to the sequential restrictions on operation performance are as follows: $R^1 = \{(1,1,2),(1,2,3)\}$, $R^2 = \{(1,3,4),(1,4,5),(2,3,6),(2,6,7),(3,3,8)\}$.

Table 1
Summary of sets, parameters and variables used in the methods.

|		
Basic sets:		
I	–	the set of machines: $I = \{1, ..., M\}$;
J	–	the set of types of operations: $J = \{1, ..., N\}$;
K	–	the set of types of products: $K = \{1, ..., Z\}$;
L	–	the set of periods (intervals): $L = \{1, ..., H\}$;
S	–	the set of product indices: $S = \{1, ..., U\}$;
V	–	the set of stages: $V = \{1, ..., A\}$.
Other sets:		
D	–	the set of pairs (i, v), in which production machine i is placed in the stage v;
J^1	–	the set of types of primary operations, $J^1 \subset J$;
J^2	–	the set of types of secondary operations, that distinguished multi-option products, $J^2 \subset J$;
O^1	–	the set of pairs (k, j), in which the primary operations $j \in J^1$ is required for product of type $k \in K$;
O^2	–	the set of pairs (s, j), in which the secondary 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, and operations r, $j \in J^1$ are required for product of type k;
R^2	–	the set of elements (s, r, j), in which operation j is executed immediately before task r, and operations r, $j \in J$ are required for product s, one of this operation or both belong to set J^2;
T	–	the set of pairs (s, k), in which product s is type k;
V_j	–	the set of the stages in which the machines are capable of execution of operation j.
Parameters:		
a_k	–	system readiness for performing operation concerning product of type $k \in K$;
g_{ev}	–	transportation time between machines in stage e and in stage v;
n_{il}	=	1, if machine i is available during period l, otherwise $n_{il} = 0$;
p^1_{jk}	–	processing time for primary operation $j \in J^1$ for product of type k;
p^2_{js}	–	processing time for secondary operation $j \in J^2$ for product s.
Decision variables:		
x_{ijsl}	=	1, if product s is assigned to machine i to perform operation j in period l, otherwise $x_{ijsl} = 0$.
Variables formulated for the M2 model only:		
w_{is}	–	time of starting operation j on machine i;
z_{is}	–	time of ending operation j on machine i;
y_{isl}	=	1 if machine i is blocked during period l by product s, otherwise $y_{isl} = 0$.
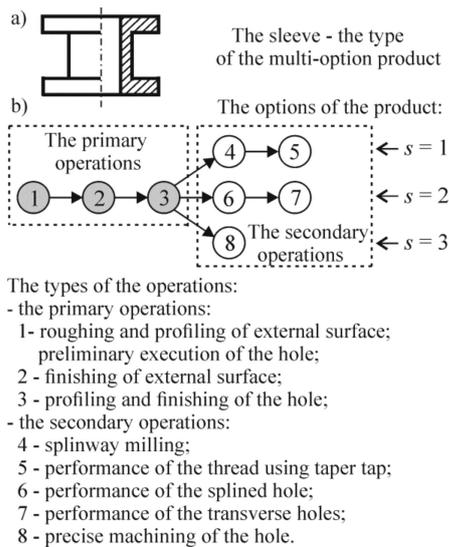
Fig. 3. An example of multi-option product – the sleeve:
a) engineering drawing of the sleeve, b) graph of operations for the three options of the sleeve.

According to the designations listed in Table 1, the production line can simultaneously manufacture Z types of products $k \in K$. The production volume is determined by parameter U, constituting the number of all the products $s \in K$, which can be produced in different variants.

The M1 and M2 mathematical models:

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

subject to:
$$\sum_{i \in I} \sum_{j \in J} x_{ijsl} \leq 1; \; l \in L; \; s \in S, \quad (2)$$

$$\sum_{j \in J} \sum_{s \in S} x_{ijsl} \leq n_{il}; \; i \in I; \; l \in L, \quad (3)$$

$$\sum_{i \in I} \sum_{l \in L:n_{il}=1} x_{ijsl} = p^1_{jk}; \; j \in J^1;$$
$$(s, k) \in T : (k, j) \in O^1, \quad (4)$$

$$\sum_{i \in I} \sum_{l \in L: n_{il}=1} x_{ijsl} = p_{js}^2; \tag{5}$$
$$j \in J^2; \ s \in S : (s, j) \in O^2,$$

$$x_{ijsl} = 0; \ (i, v) \in D; \ v \notin V_j; \tag{6}$$
$$j \in J; \ l \in L; \ s \in S,$$

$$x_{\tau r s f} + x_{ijsl} \leq 1; \ (\tau, v)(i, v) \in D; \ rj \in J; \tag{7}$$
$$fl \in L; \ s \in S; \ r \neq j; \ \tau \neq i,$$

$$ix_{ijsl} \geq \tau x_{\tau r s f} - (H + 1)(1 - x_{ijsl}); \tag{8}$$
$$i\tau \in I; \ (s, k) \in T; \ lf \in L;$$
$$(k, r, j) \in R^1 \vee (s, r, j) \in R^2,$$

$$lx_{ijsl} - fx_{\tau r s f} \geq 1 + g_{ev} - (H + 1)(1 - x_{ijsl}); \tag{9}$$
$$(i, v)(\tau, e) \in D; \ fl \in L;$$
$$\left((s, k) \in T; \ (\kappa, r, j) \in R^1; \ \kappa = k\right) \vee (s, r, j) \in R^2,$$

$$lx_{ijsl} \geq a_k; \ i \in I; \ j \in J; \ l \in L; \ (s, k) \in T, \tag{10}$$

$$x_{ijsl} \in \{0, 1\} i \in I; \ j \in J; \ l \in L; \ s \in S. \tag{11}$$

For the model M1 only (for no-wait scheduling):

$$lx_{ijsl} - fx_{irsf} - 1 \leq g_{ev} \sum_{q \in L: f < q < l} (x_{ijsq} + x_{irsq})$$
$$+ (H + 1)(1 - x_{ijsl}); \tag{12}$$
$$(s, k) \in T; \ (i, v)(\tau, e) \in D; \ lf \in L; \ f < l;$$
$$(k, r, j) \in R^1 \vee (s, r, j) \in R^2.$$

For the model M2 only (for possibility of blocking machines):

$$lx_{ijsl} - fx_{irsf} - 1 \leq \sum_{q \in L: f < q < l} (x_{ijsq} + x_{irsq})$$
$$+ (H + 1)(1 - x_{ijsl}); \tag{13}$$
$$(s, k) \in T; \ i \in I; \ lf \in L; \ f < l;$$
$$(k, r, j) \in R^1 \vee (s, r, j) \in R^2,$$

$$lx_{ijsl} - fx_{\tau r s f} - 1 \leq g_{ev} + \sum_{q \in L: f < q < l} (x_{ijsq} + x_{\tau r s q})$$
$$+ \sum_{q \in L} y_{\tau s q} + (H + 1)(1 - x_{\tau r s f}); \tag{14}$$
$$lf \in L; \ f < l; \ (i, v)(\tau, e) \in D; \ e < v;$$
$$\left((s, k) \in T; \ (\kappa, r, j) \in R^1; \ \kappa = k\right) \vee (s, r, j) \in R^2,$$

$$w_{is} \geq lx_{ijsl} - \sum_{f \in L} \sum_{r \in J} x_{irsf} + 1 - (H + 1)(1 - x_{ijsl}); \tag{15}$$
$$i \in I; \ j \in J; \ l \in L; \ s \in S,$$

$$w_{is} \leq lx_{ijsl} + (H + 1)(1 - x_{ijsl}); \tag{16}$$
$$i \in I; \ j \in J; \ l \in L; \ s \in S,$$

$$z_{is} \geq lx_{ijsl} - (H + 1)(1 - x_{ijsl}); \tag{17}$$
$$i \in I; \ j \in J; \ l \in L; \ s \in S,$$

$$ly_{\tau s l} \leq w_{is} - g_{ev} - 1 + (H + 1)(1 - y_{\tau s l}); \tag{18}$$
$$(\tau, e)(i, v) \in D; \ e < v; \ l \in L; \ s \in S,$$

$$ly_{\tau s l} \geq z_{\tau s} + 1 - (H + 1)(1 - y_{\tau s l}); \tag{19}$$
$$(\tau, e) \in D; \ v \in V; \ e < v; \ l \in L; \ s \in S,$$

$$\sum_{s \in S} (x_{ijsl} + y_{isl}) \leq 1; \ i \in I; \ j \in J; \ l \in L, \tag{20}$$

$$y_{isl} \in \{0, 1\} \, w_{is} z_{is} \geq 0; \ i \in I; \ l \in L; \ s \in S. \tag{21}$$

The shortest possible schedules using M1 and M2 mathematical models are determined by minimizing the value of sum (1). As a result, the production operation completion times on individual machines are the shortest. The constraints built for M1 and M2 mathematical models ensure: (2) – performance of no more than one operation for each multi-option product in any given period (time interval); (3) – performance of no more than one operation at a time by the machine, if in this period the machine is available for production operation performance; (4) and (5) – distribution of all the primary operations (4) and secondary operations among the machines; (6) – elimination of assignment of operations to inappropriate machines; (7) – no more than one machine at each stage being loaded by the given multi-option product; (8) – unidirectional product flow along the production hybrid line; (9) – taking into account the given sequence of primary and secondary operations; (10) – performance the multi-option products only when the production line is ready for this; (11) – binarity of the decision-making variables.

The next constraint is formulated for the M1 model only. Constraint (12) ensures not only no-wait scheduling (intervals between the operations are used solely to transport the product), but also continuity of performing consecutive operations. Moreover, thanks to this constraint, operations concerning a product, performed on a single machine, are not separated with other operations, assigned to different products.

The last constraints regarding the M1 model ensure: (13) – continuity of performing operations regarding a specific product on the given machine – just like in the case of constraint (12); (14) – determination of the length of the period in which the machine must be blocked by the multi-option product; (15)–(17) – determination of starting and ending times for operations concerning individual products on the given machines (if product s does not encumber machine i, then values of variables $w_{is} = z_{is} = 0$); (18) – machine blocked by product (acting as a buffer) before performing the next operation and before transport to the following stage;

(19) – machine blocked by product immediately upon completing the operations concerning this product on that machine; (20) – machine cannot act as a buffer while an operation is performed on this machine; (21) – relevant types of variables.

Verification of the method – computational experiments

The presented mathematical models have been verified and compared by means of computational experiments. The described M1 and M2 mathematical models were coded in the AMPL language (*A Mathematical Programming Language*) [21], and next *.mps files were generated. Computations were performed using the GUROBI optimizer [22]. The computational experiments covered 5 groups of tasks. For each one of the groups of tasks, 25 examples were solved.

Lengths are compared for schedules determined on the basis of the M1 and M2 models. Length of each schedule $C_{\max}$ was determined based on formula (22). Computational times are provided as well.

$$C_{\max}^{M} = \max_{i \in I, j \in J, l \in L, s \in S} lx_{ijsl} \qquad (22)$$
– for the M mathematical model.

Indexes concerning comparison of the M1 and M2 models are defined in formulas (23). Index c is intended for comparing the lengths of the determined schedules, whereas index t concerns comparison of the computation times, where CPU^{M} – computation time when using the M model.

$$c = \frac{C_{\max}^{M1} - C_{\max}^{M2}}{C_{\max}^{M2}} \cdot 100\%;$$

$$t = \frac{CPU^{M1} - CPU^{M2}}{CPU^{M2}} \cdot 100\%. \qquad (23)$$

Parameters of test task groups and average values if indexes defined in Eqs (23) are listed in Table 2.

Table 2
Specification of parameters of groups of tasks and results of computational experiments.

Group	Parameters of groups						Results of experiments	
	A	M	N	Z	S	H	c [%]	t [%]
1	2	4	10	4	8	16	17.3	10.3
2	3	6	12	4	8	18	15.3	10.0
3	3	6	14	5	10	18	14.2	10.4
4	4	8	14	5	12	20	14.0	11.2
5	4	8	16	6	14	22	14.1	11.4

Numbers of: A – stages, M – machines, N – types of operations, Z – types of products, U – products, H – periods.

The average values of indexes c listed in Table 2 show that use of the M1 model (regarding no-wait scheduling) resulted in determination of schedules up to 17% longer – compared to schedules where machines can be used as intermediate buffers. The longest schedule built for no-wait scheduling, concerning a group of 5 test tasks, was almost 20% longer than a schedule which permitted the machines to be blocked by products waiting for consecutive operations.

The average values of the determined factors t indicate that in the case of no-wait scheduling, the computation time was over 10% longer than when the M2 model was used. The computations were performed using CPU Intel Core i7-8550U 4GHz.

In the case of no-wait scheduling, schedules determined based on the presented monolithic approach (using model M1) and schedules created using the hierarchical method presented in the paper [23] – after modifying this method by removing constraints regarding intermediate buffer and adding a restraint which ensures no-wait scheduling, i.e. counterpart to constraint (12), were also compared. Application of the monolithic method described herein had a positive impact on the quality of the solutions – the defined schedules were optimal, about 4–6% shorter than in the case of the M1 model – compared to the hierarchical method. Application of the hierarchical method resulted in determination of schedules which were 3.8–5.7%, but solutions were defined in a shorter time – by about 20%.

Conclusions

It should be emphasized that the developed method is related to group technology [16]. The products are grouped according to technological operations, some of which are identical (in the case of a product group – these operations are called primary), and some of which can be similar (in the case of individual product variants). Product grouping simplifies scheduling of multi-option products.

One advantage distinctive of the described monolithic method, related to group technology, is that it provides for multi-option products – the details which set the products apart can be adapted to the requirements of the customers. Thanks to the constructed data structure regarding these products, the sizes of the tasks to solve are about 8–10% smaller compared to the data structure where each multi-variant product would be described (digitally) separately, without detailing identical operations for a specific product type (primary operations) and without taking into account the order constraints

– identical for option product types. Such a data structure has a positive impact on the time required for the computations, which is important in the case of the monolithic approach to problem solving. Thanks to this one-level approach and using integer programming, the propose method allows to determine optimum solutions – the shortest schedules are constructed.

This method is also unique in that it takes into account limited availability of the machines. Therefore, it provides for any scheduled stoppages, e.g. for maintenance.

Further studies concerning creation of production schedules for multi-option products are planned. It is possible to distinguish three main directions of these studies. The first one concerns creation of a scheduling method where the schedules are determined in a relatively short time. It is planned to build a relaxation heuristics. This heuristics will make use of the presented mathematical models, without the integer requirements of decision variables. Continuous variables will be taken into account. This will reduce the time of solving the task which the method applies to. Procedures for modifying the original solution will be developed, in order to meet all the constraints featured in the mathematical models. The second direction of the planned studies concerns assembly systems. The developed method will be modified in order to take into account the issues of assembly on unidirectional assembly lines. The created schedule will not only solve the task of assigning assembly operations to the machines, but also the task of arranging the feeders of components for the assembled products. It is planned to develop both a method intended to determine optimum solutions and heuristics which, in a relatively short time, create an optimum assembly schedule, yet the solution is encumbered with a certain deviation from the optimum. Another direction of further studies provides for the possibility of rescheduling. The newly-created schedule will also provide for product operations which could not be performed in accordance with the original schedule, e.g. due to machine malfunctions.

Therefore, the described method may be the basis for further studies, as demonstrated in the previous paragraph. Of course, the studies may also be directed elsewhere, such as e.g. simultaneous scheduling and lot-sizing problem. The issues concerning such studies are described in the paper [24].

The presented mathematical models can be, of course, modified, to adapt them to the changing conditions of the production process. The current development of computer technology and software offers good perspectives for the method presented here.

Thanks to this development, it is possible to solve increasingly large problems in decreasingly short time, and to determine optimum solutions.

References

[1] Varela M.L.H., Trojanowska J., Carmo-Silva S., Costa N.M.L., Machado J., *Comparative simulation study of production scheduling in the hybrid and the parallel flow*, Management and Production Engineering Review, 8, 2, 68–70, 2017.

[2] Chan F.T.S, Li N., Chung S.H., Saadat M., *Management of sustainable manufacturing systems – a review on mathematical problems*, International Journal of Production Research, 55, 4, 1210–1225, 2017.

[3] Hassani Z.I.M., Barkany A.E., Abbasi I.E., Jabri A., Darcherif A.M., *Planning and scheduling of production system in conditioning line: industrial application, optimization and simulation approach*, Management and Production Engineering Review, 10, 4, 3–10, 2019.

[4] Li X., Gao L., Zhang C., Shao X., *A review on Integrated Process Planning and Scheduling*, International Journal of Manufacturing Research, 5, 2, 161–180, 2010.

[5] Sugimura N., Shrestha R., Tanimizu Y., Iwamura K., *A study on integrated process planning and scheduling system for holonic manufacturing*, Process Planning and Scheduling for Distributed Manufacturing, Springer, London, UK, 311–334, 2006.

[6] Soualhia M., Khomh F., Tahar S., *Task Scheduling in Big Data Platforms: A Systematic Literature Review*, The Journal of Systems and Software, 134, 170–189, 2017.

[7] Ruiz R., Vázquez-Rodriguez J.A., *The Hybrid Flow Shop Scheduling Problem*, European Journal of Operational Research, 205, 1, 1–18, 2010.

[8] Sawik T., *Loading and scheduling of a flexible assembly system by mixed integer programming*, European Journal of Operational Research, 154, 1, 1–19, 2004.

[9] Ronconi D.P., Birgin E.G., *Mixed-Integer Programming Models for Flowshop Scheduling Problems Minimizing the Total Earliness and Tardiness*, [in:] Ríos-Mercado R., Ríos-Solís Y. [Eds], Just-in-Time Systems. Springer Optimization and Its Applications, vol. 60, Springer, New York, 2012.

[10] Gonzales T.F. [Ed.], *Handbook of Approximation Algorithms and Metaheuristics*, Chaoman and Hall/CRC, New York, 2007.

[11] Behnamin J., Fatemi Ghomi S.M.T., *Hybrid flowshop scheduling with machine and resource depen-*

dent processing times, Applied Mathematical Modelling, 35, 3, 1107–1123, 2011.

[12] 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, 1439–1454, 2010.

[13] Magiera M., *A relaxation heuristic for scheduling flowshops with intermediate buffers*, Bulletin of the Polish Academy of Sciences. Technical Sciences, 61, 4, 929–942, 2013.

[14] Magiera M., *Two-level of production scheduling for flow-shop systems with intermediate storages*, in: Total Logistic Management, Annual 1, AGH University of Science and Technology, Kraków, 101–110, 2008.

[15] Castillo L. Borrajo D., Salido M.A., Oddi A. [Eds], *Planning, scheduling and constraint satisfaction: from theory to practice*, IOS Press, Amsterdam, 2005.

[16] Debnarova L., Krchova D., Kuric I., *Group technology in context of the product classification*, Advances in Sciences and Technology Research Journal, 8, 21, 78–81, 2014.

[17] Yadov A., Jayswal S.C., *Modelling of flexible manufacturing system: a review*, International Journal of Production Research, 56, 7, 2464–2487, 2017.

[18] Katragjini K., Vallada E., Ruiz R., *Flow shop rescheduling under different types of disruption*, International Journal of Production Research, 51, 3, 780–797, 2013.

[19] Vieira G.E., Herrmann J.W., Lin E., *Rescheduling manufacturing systems: a framework of strategies, policies, and methods*, Journal of Scheduling, 6, 39–62, 2003.

[20] Magiera M., *A no-wait scheduling method for assembly operations concerning multi-option electric devices*, Przegląd Elektrotechniczny (Electrical Review), 94, 4, 187–190, 2018.

[21] Fourer R., Gay D., Kernighan B., *AMPL, A Modelling Language for Mathematical Programming*, Duxbury Press, Pacific Grove, CA, 2003.

[22] www.gurobi.com (Gurobi Optimizer), January 2019.

[23] Magiera M., *A hierarchical method of planning of multi-option product flows through production lines* [in Polish: *Hierarchiczna metoda planowania przepływów wielowariantowych produktów przez linie produkcyjne*], Automatyzacja procesów dyskretnych. Teoria i zastosowania, vol. I, Świerniak A., Krystek J. [Eds], Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice, 171–183, 2016.

[24] Kaczmarczyk W., *Valid inequalities for proportional lot-sizing and scheduling problem with fictitious microperiods*, International Journal of Production Economics, 229, 236–247, 2020.