Marek MAGIERA

AGH University of Science and Technology

doi:10.15199/48.2015.04.20

Comparative analysis of two hierarchical methods of assembly planning for producers of electric and electronic equipment

Streszczenie. *Porównano dwie dwupoziomowe metody planowania montażu, przeznaczone dla producentów sprzętu elektrycznego i elektronicznego. Zbudowano je dla elastycznych linii montażowych z maszynami równoległymi. Na górnym poziomie opracowanych metod rozwiązywane jest zadanie równoważenia obciążeń maszyn (w metodzie I) lub równoważenia obciążeń stadiów (w metodzie II). Równocześnie dla każdego produktu wybierana jest jedna sekwencja montażowa. Na dolnym poziomie szeregowane są operacje montażowe. Przedstawiono wyniki eksperymentów obliczeniowych. (**Analiza porównawcza dwóch hierarchicznych metod planowania montażu dla producentów sprzętu elektrycznego i elektronicznego**)*

Abstract*. The two two-level methods of assembly planning for producers of electric and electronic equipment are compared. The methods are constructed for flexible assembly lines with parallel machines. At the upper level of the prepared methods, a task is solved for balancing machine workload (in the method I) or a task for balancing assembly stages workload (in the method II). Simultaneously, only one assembly plan is selected for each product type. At the lower level, assembly tasks are scheduling. The results of calculation experiments are presented.*

Słowa kluczowe: montaż, szeregowanie, programowanie całkowitoliczbowe, planowanie hierarchiczne
Keywords: Assembly, Scheduling, Integer programming, Hierarchical planning

1. Introduction

Flexible Assembly System (FAS) is one of the basic types of Flexible Manufacturing Systems. It consists of automated assembly stations and transport devices. It is dedicated for simultaneous assembly of many different products in short series. The basic problems related to FAS functioning are described in the works [1] and [2].

The basic planning and control tasks in FAS include [3]:
- selection of only one assembly plan from among the given alternative plans for each product;
- balancing machine workload, that is optimization of distribution of tasks and resources in order to execute the coming production orders;
- scheduling assembly tasks in order to build detailed assembly schedules.

The solution of the first of the tasks (selection of assembly plans) affects quality of the solutions of the further listed tasks. Assembly machine workload and assembly schedule length are dependent on the selected assembly plans. The overview of the assembly planning methods, including selection of assembly plans, is given in [4]. This work shows that a major part of the assembly planning methods is based on discrete optimization. This mathematical tool has been also used in the methods presented in this paper.

Another task for FAS is balancing machine workload. In case of the assembly systems in which parts are collected from the feeders, this task has one more aspect. The solution of this tasks contributes not only to determination of allocation of assembly tasks to machines, but also to planning location of the part feeders. Majority of assembly tasks consists in adding assembly of single parts or components to the already assembled parts. The execution of this tasks, which includes limited working space for individual machines, has also been included in the methods presented in the article. Taking into consideration of planning location of the part feeders has been inspired by paper [5]. The issues related to breaking down the part feeders is also given in the paper [6]. As compared with the assembly planning methods presented in these papers, this work defines not only a set of types of tasks which require using part feeders, but also a set of types of tasks which do not need them (e.g. welding, soldering).

The last of the stated tasks is scheduling assembly tasks in order to develop the product flow schedule through the assembly system. Task scheduling methods of assembly planning may be broken down into two groups: those which allow determination of optimum solutions (in view of the criteria taken into account) and those which are used to determine approximate solutions which are slightly deviated from the optimum. The literature covering the issues of scheduling of assembly tasks for flow systems is very rich. The overview of methods is described in [8].

An approach for integration of process planning and scheduling was inspired by article [9]. The method described in [9] consists of modules: plan selection modules, scheduling module, schedule analysis and process plan modification module. However, these interesting methods are not suited to the Flexible Assembly Systems.

One of two approaches is used for solving problems in assembly planning: monolithic (one-level) or hierarchical (multilevel) [7]. The monolithic approach takes into account many parameters, variables included in a large number of constraints at the same time. With this approach, difficulties arise in solving problems of significant size. For this reason, an alternative approach is often used, which allows solving problems of relatively larger volumes. It is usually achieved with some deviation from the optimum solution. In this hierarchical approach, the global problem is divided into a number of problems solved in succession. The division into sub-problems allows simultaneous accounting of a relatively smaller number of parameters, variables, constraints.

The methods built by the author of the paper are related to the presented issues. These methods are based on the hierarchical approach. They take into consideration various configurations of assembly machines for production of electric and electronic equipment.

2. Concepts of the methods

The methods have been prepared for multi-stage, unidirectional flow systems in which many different types of the assembled products may flow at the same time. Each stage is a set of identical machines operating in parallel. Passing through the given stage, the product constitutes workload for one machine only. Some stages may be omitted by individual products. A sample configuration of the assembly line with parallel machines is presented in Figure 1.

These methods take into account the following configurations of the production system:

- a flexible assembly line with intermediate buffers of limited capacities (Figure 1). If a product does not constitute workload for the given machine, it is awaiting in the buffer preceding the stage of the given machine.

– a flexible assembly line without intermediate buffers. When there are no buffers, 2 cases are analysed: 1: machines perform the role of buffers: when the next task cannot be executed, the product blocks the machine in which the last task has been completed; 2: the so-called "no-waiting" schedule is built: breaks between execution of consecutive tasks result only from transport time between the machines of different stages.

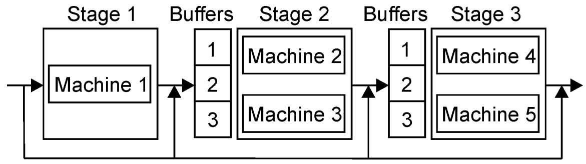

Figure 1. Diagram of multistage assembly line with buffers

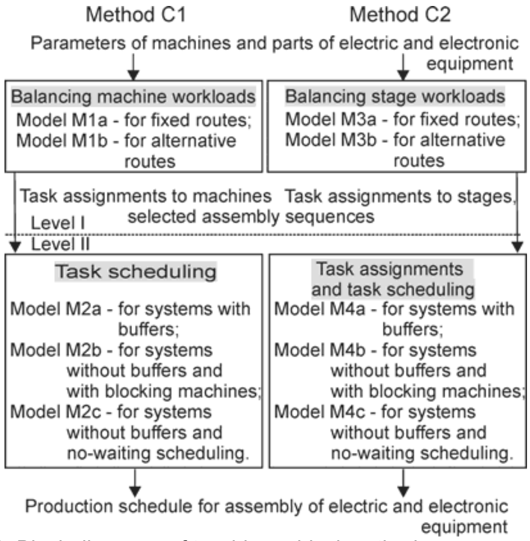

Figure 2. Block diagrams of two hierarchical methods

The concepts of the two developed methods (C1 and C2) are illustrated in the block diagrams in Figure 2. These diagrams include also markings of the linear mathematical models of integer programming built for individual method levels. On the first level of the C1 method, machine workloads are balanced. The purpose is to prevent the so-called "bottlenecks" in the assembly system. On the first level of the C2 method, stage workloads are balanced. The allocation of the tasks to the machines takes place at the second level of the method. For both methods, two types of production routes have been taken into account:

– a fixed assembly routes: each task type is allocated to the machines of the same stage;

– an alternative assembly routes: each task type is allocated to at least one machine; these machines may belong to different stages.

The following are solutions of the problem formulated for the level I of the methods: allocation of the tasks to the machines (for the C1 method), allocation of the tasks to the stages (for the C2 method). These allocations take into consideration the selected assembly plans. Exactly one assembly plan is selected for each product. Machine workload by assembly tasks are assigned to the selected plans are as low as possible.

The assembly plans and the allocations of the tasks to the machines (for the C1 method) or the allocations of the tasks to the stages (for the C2 method) selected at the level I, each of the methods constitute data for the level II of the methods, in which assembly tasks are scheduling. The schedules with the shortest possible lengths are constructed. The mathematical models have been built for 3 different cases: the system with the intermediate buffers (M2a, M4a); the system without the buffers, with the possibility of blocking the machines by the products awaiting the next tasks (M2b, M3b); the system without the buffers, where the tasks are "no-waiting" scheduling (M2c, M4c).

3. Parameters and variables

The problems stated in the previous chapter have been described mathematically. The sets of assembly plans are given. For each product, at least one series of consecutive tasks is generated. The sets of product types, assembly tasks, the machines belonging to individual assembly stages are also known. The list of all the indices, parameters and variables used in the linear mathematical models built for the methods is given in Table 1.

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

Indices:	
i	= machine; $i \in I = \{1, \ldots, M\}$;
j	= type of assembly task; $j \in J = \{1, \ldots, N\}$;
k	= type of product; $k \in K = \{1, \ldots, W\}$;
l	= period (time interval); $l \in L = \{1, \ldots, H\}$;
s	= assembly plan; $s \in T = \{1, \ldots, Q\}$;
v	= assembly stage; $v \in V = \{1, \ldots, A\}$.
Parameters:	
a_{vj}	= working space of machine in stage v required for execution of task j;
b_v	= working space of the machine placed in stage v;
d_v	= number of intermediate buffers before stage v;
g_{vk}	= transport time for product k from the machine in which assembly has been completed to machine in stage v (only for the level I);
m_v	= number of the assembly machines in stage v;
p_{js}	= assembly time of task j for assembly plan s;
η_k	= priority of the execution of the task of product k;
μ_{il}	= 1, if machine i is available during period l, otherwise $\mu_{il} = 0$;
α	= any integral number larger than the number of the analysed time intervals (periods);
F	= the set of arranged pairs (i, v), such that the machine i belongs to the stage v;
I_j	= the set of machines capable of performing task j (the parameter for the C1 method);
J_k	= the set of tasks required for product k, $J_k \subset J$;
J_c	= the set of tasks which require using the feeder; ;
R_s	= the set of pairs of tasks (j, r) executed in succession according to assembly plan $s \in T$;
S_k	= the set of assembly plans for product type k, $S_k \subset S$;
T	= the set of all assembly plans;
V_j	= the set of the stages in which the machine are capable of execution of task j (for the C2 method);
λ	= maximum break in the assembly.
Decision variables:	
• for the models built for the level I:	
u_s	= 1, if the plan s has been selected, otherwise $u_s = 0$;
Variables for the M1a and M1b models (for the C1 method):	
$\tilde{P}_{\max}$	= maximum machine workload;
$\tilde{x}_{ij}$	= 1, if type of assembly task j is assigned to machine i, otherwise $\tilde{x}_{ij} = 0$;
$\tilde{z}_{ijs}$	= 1, if assembly task j belonging to plan s has been assigned to machine i, otherwise $\tilde{z}_{ijs} = 0$.
Variables for the M3a and M3b models (for the C2 method):	
$P_{\max}$	= maximum stage workload;
x_{vj}	= 1, if type of assembly task j has been assigned to stage v, otherwise $x_{vj} = 0$;
z_{vjs}	= 1, if assembly task j belonging to the plan s has been assigned to stage v, otherwise $z_{vjs} = 0$;
• for the models built for the level II:	
q_{ikl}	= 1, if during period l task of product k is executed on machine i , otherwise $q_{ikl} = 0$;
y_{vkl}	= 1, if during period l product k is in the buffer located before stage v, otherwise $y_{vkl} = 0$ (for the M2a, M4a);
w_{ikl}	= 1, if during period l machine i is loaded by product k, awaiting the execution of the next task (the machine performs the role of the buffer), otherwise $w_{ikl} = 0$.

4. Level I: selection of assembly plans

In accordance with the diagram in Figure 1, at the level I of the methods is solved the problem of balancing machine workloads (the C1 method) or is solved the problem of balancing stage workloads (the C2 method) with the simultaneous selection of the assembly plans.

The all mathematical models built for the C1 method are presented in the papers [10] and [11] of the author of this article. The models constructed for the level I of the C1 method are described in [10]: the M1a model (us M1-I in [10] and the M1b model (us M2-I in [10]).

These are the mathematical models M3a and M3b:

(1) Minimize: $P_{\max}$

Subject to:

(2) $\sum_{s\in T}\sum_{j\in J} p_{js} z_{vjs} + \sum_{l\in L}\sum_{i\in I:(i,v)\in F}(1-\mu_{il}) \le P_{\max}; v\in V$

(3) $\sum_{v\in V_j} x_{vj} = 1;\ j\in J$ - for the M3a model only

(4) $\sum_{v\in V_j} x_{vj} \ge 1;\ j\in J$ - for the M3b model only

(5) $\sum_{j\in J_c} a_{vj} x_{vj} \le b_v m_v; v\in V$

(6) $\sum_{s\in S_k} u_s = 1; k\in K$

(7) $\sum_{v\in V} z_{vjs} = u_s; s\in S_k;\ j\in J_k; k\in K$

(8) $z_{vjs} \le x_{vj}; v\in V;\ j\in J_k; s\in S_k; k\in K$

(9) $\sum_{v\in V} v\, z_{vjs} \le \sum_{v\in V} v\, z_{vrs}; s\in S_k; k\in K; (j,r)\in R_s$

(10) $x_{vj}, z_{vjs}, u_s \in \{0,1\}; v\in V;\ j\in J; s\in T$

In the linear mathematical models M3a and M3b, the load of the most loaded stage (1), determined according to (2), is minimized. The second component of the inequality (2) allows for a limited availability of machines in the makespan. The remaining constraints guarantee: (3) - the allocation of each type of tasks to one stage only (for the M3a model); (4) - the allocation of each type of tasks to at least one stage (for the M3b model); (5) - taking into consideration limited working space of the machines; (6) - the selection of exactly one assembly plan for each product flowing through the assembly line; (7) - distribution of all the assembly tasks (assigned to the selected plans) between the stages; (8) - the allocation of the tasks assigned to the particular product to these stages which are assigned (according to (3), (4)) the possibility of execution of the task of this type; (9) - maintaining the order of execution of the tasks according to the given assembly plans with an unidirectional product flow; (10) - binary of decision variables.

5. Level II: scheduling of assembly tasks

The models constructed for the level II of the C1 method are described in [10] and [11]: the M2a model (us M1-II in [10], the M2b model (us M1-II in [11]) and the M2c model (us M2-II in [10]). The models constructed for the level II of the C2 method are presented in this chapter.

The results of the problem solved at the level I constitute the input data for the problem of scheduling assembly tasks solved at the level II. These data include: t_{vk} – time of loading stage v by product k, determined using (11).

(11) $t_{vk} = \sum_{s\in S_k}\sum_{j\in J_k} p_{js} z_{vjs}; k\in K; v\in V$

These are the mathematical models M4a, M4b and M4c:

(12) Minimize: $\sum_{i\in I}\sum_{k\in K}\sum_{l\in L} l\eta_k\, q_{ikl} + \sum_{k\in K}\sum_{v\in V}\sum_{l\in L} y_{vkl}$ - for M4a

(13) Minimize: $\sum_{i\in I}\sum_{k\in K}\sum_{l\in L} l\eta_k q_{ikl}$ - for M4b and M4c

(14) Subject to: $\sum_{k\in K} q_{ikl} \le \mu_{il}; i\in I; l\in L$

(15) $\sum_{i\in I:(i,v)\in F}\sum_{l\in L} q_{ikl} = t_{vk}; k\in K; v\in V$

(16) $lq_{ikl} - fq_{ikf} \le t_{vk} - 1 + \alpha(1-q_{ikf});$
$(i,v)\in F; k\in K; l,f\in L; l>f$

(17) $\sum_{i\in I:(i,v)\in F}\sum_{l\in L}\frac{lq_{ikl}}{t_{vk}} - \sum_{\tau\in I:(\tau,\varepsilon)\in F}\sum_{l\in L}\frac{lq_{\tau kl}}{t_{\varepsilon k}} +$
$-0.5(t_{vk}+t_{\varepsilon k}) \ge g_{vk}; k\in K; v,\varepsilon\in V; t_{vk}, t_{\varepsilon k}>0$

(18) $q_{ikl} + q_{\tau kf} \le 1;\ k\in K; (\tau,v),(i,v)\in F; i\ne\tau$

(19) $q_{ikl}\in\{0,1\}; i\in I; k\in K; l\in L$

For the M4a model only:

(20) $\sum_{i\in I:(i,v)\in F}\sum_{l\in L}\frac{lq_{ikl}}{t_{vk}} - \sum_{\tau\in I:(\tau,\varepsilon)\in F}\sum_{l\in L}\frac{lq_{\tau kl}}{t_{\varepsilon k}} - \frac{t_{vk}+t_{\varepsilon k}}{2} +$
$-g_{vk} = \sum_{l\in L} y_{vkl};\ k\in K; l,f\in L; v\in V\setminus\{1\};$

$\varepsilon\in V;\ v>\varepsilon; t_{vk}, t_{\varepsilon k}>0; \sum_{\psi=\varepsilon}^{v} t_{\psi k} = t_{vk}+t_{\varepsilon k}$

(21) $ly_{vkl} \ge \sum_{f\in L}\sum_{\tau\in I:(\tau,\varepsilon)\in F}\frac{fq_{\tau kf}}{t_{\varepsilon k}} + \frac{t_{\varepsilon k}+1}{2} + g_{vk} - \alpha(1-y_{vkl});$
$k\in K; l\in L; v\in V\setminus\{1\}; \varepsilon\in V; v>\varepsilon;\ t_{vk}, t_{\varepsilon k}>0$

(22) $\sum_{f\in L}\sum_{i\in I:(i,v)\in F}\frac{fq_{ikf}}{t_{vk}} - \frac{t_{vk}+1}{2} - ly_{vkl} \ge 1;$
$k\in K; l\in L; v\in V\setminus\{1\}; t_{vk}>0$

(23) $\sum_{k\in K:t_{vk}>0} y_{vkl} \le d_v;\ v\in V\setminus\{1\}; l\in L$

(24) $y_{vkl}\in\{0,1\}; v\in V; k\in K; l\in L$

For the M4b model only:

(25) $w_{ikl} \le \sum_{f\in L} q_{ikf}; i\in I; k\in K; l\in L$

(26) $\sum_{i\in I:(i,v)\in F}\sum_{l\in L}\frac{lq_{ikl}}{t_{vk}} - \sum_{\tau\in I:(\tau,\varepsilon)\in F}\sum_{l\in L}\frac{lq_{\tau kl}}{t_{\varepsilon k}} - \frac{t_{vk}+t_{\varepsilon k}}{2} +$
$-g_{vk} = \sum_{\tau\in I:(\tau,\varepsilon)\in F}\sum_{l\in L} w_{\tau kl};\ k\in K; l,f\in L; v\in V\setminus\{1\};$

$\varepsilon\in V; \varepsilon<v; t_{vk}, t_{\varepsilon k}>0; \sum_{\psi=\varepsilon}^{v} t_{\psi k} = t_{\varepsilon k}+t_{vk}$

(27) $lw_{\tau kl} + \alpha(1-y_{\tau kl}) \ge \sum_{\tau\in I:(\tau,\varepsilon)\in F}\sum_{f\in L}\frac{fq_{\tau kf}}{t_{\varepsilon k}} + \frac{t_{\varepsilon k}+1}{2};$

$$(\tau,\varepsilon)\in F; k\in K;\ t_{\varepsilon k}>0;\ \sum_{\rho\in V:\varepsilon\le\rho} t_{\rho k}>t_{\varepsilon k};\ l\in L$$

(28) $$l w_{\tau k l}\le \sum_{i\in I:(i,v)\in F}\ \sum_{f\in L}\frac{f\,q_{ikf}}{t_{vk}}-\frac{t_{vk}+g_{vk}}{2}+\alpha(1-w_{\tau k l})$$

$$(\tau,\varepsilon)\in F; v\in V; v>\varepsilon; k\in K; t_{vk},t_{\varepsilon k}>0;$$

$$\sum_{\rho\in V:\varepsilon\le\rho\le v} t_{\rho k}=t_{\varepsilon k}+t_{vk};\ l\in L$$

(29) $$q_{ikl}+w_{ikl}\le 1;\ i\in I; k\in K; l\in L$$

(30) $$w_{ikl}\in\{0,1\};\ i\in I; k\in K;\ l\in L$$

Moreover, to the models M4a and M4b, the constraint (31) may be added, whereas for the model M4c, λ = 0 should be assumed: the maximum break in the execution of the assembly tasks.

(31) $$l\cdot\sum_{i\in I:(i,v)\in F} q_{ikl}-f\cdot\sum_{\tau\in I:(\tau,\varepsilon)\in F} q_{\tau kf}\le g_{vk}+t_{vk}-1+$$

$$\alpha\left(1-\sum_{\tau\in I:(\tau,\varepsilon)\in F} q_{\tau kf}\right)+\lambda;\ \varepsilon,v\in V; v>\varepsilon; k\in K;$$

$$t_{\varepsilon k},t_{vk}>0;\ \sum_{\rho\in V:\varepsilon\le\rho\le v} t_{\rho k}=t_{\varepsilon k}+t_{vk};\ l,f\in L$$

In order to build schedules with the shortest possible schedule lengths, the sums are minimized: (12) – for the systems with the intermediate buffers, (13) – for the system without the buffers. Minimization of the schedule length is approximated. Minimization of the sum of (12) or (13) ensures building not only shortest-length schedules, but also relatively short times of completion of assembly of each product. The value of the η_k parameter may be used to control the order of leaving the system by the assembled products. Minimization of the sum (12) ensures the selection from among the schedules the solution in which the intermediate buffers are least loaded, that is the products move directly between the assembly machines. The constraints guarantee: (14) - the execution of at the most one task on the machine at the given time, if this machine is made available for the execution of the task in the analysed period; (15) - distribution of all the assembly tasks between the machines and their execution in the given time; (16) - ensuring indivisibility of the execution of the assembly tasks; (17) - maintaining the order of the execution of the assembly tasks in the unidirectional flow system; (18) - ensuring loading of one machine at the most by the product flowing through the given stage; (19), (24) and (30) - binary of the decision variables; (20) - determining duration of stay of particular products in the buffers; (21) and (22) - ensuring placing of the product in the buffer directly before the execution of the following tasks; (23) - verification of using the buffers by the products, taking into consideration the limited capacity of the buffers; (25) - ensuring the possibility of blocking by the product only these machines to which the relevant operations have been assigned; (26) - determining the time of blocking the machine by the product awaiting the execution of the following tasks; (27) and (28) - determining the time ranges in which the machine performs the role of the buffer; (29) - elimination of the machine performing the role of the buffer during the execution of the assembly tasks; (31) - protection against exceeding the maximum duration of break in the assembly of the product.

For the schedules built based on the C1 or C2 method, the length of schedule may be determined using (32).

(32) $$C_{\max}=\max_{i\in I,k\in K,l\in L}(l\,q_{ikl})$$

6. Calculation experiments

The conducted calculation experiments were aimed at verification of the built mathematical models and comparison of two concepts of assembly planning for electric and electronic equipment. The discrete optimization package has been used along with the AMPL language (*A Modelling Language for Mathematical Programming*) [12]. The assumption was adopted that the priorities are identical for the execution of the individual products. The experiments were conducted for 4 groups of test problems. For each of the groups, 30 examples were solved (for producers of electric equipment). The parameters of these groups are given in Table 2.

Table 2. Parameters of groups of test tasks

Group	*A*	*M*	*N*	*W*	*S*	*H*	
1	2	4	10	4	10	14	Numbers of: *A* - stages, *M* - machines, *N* - types of assembly tasks ,*W* - types of products, *S* - assembly plans, *H* - periods.
2	2	6	12	5	10	16	
3	3	6	14	5	10	18	
4	3	8	16	6	12	20	

The first group of calculation experiments was aimed at comparison of the determined schedule lengths according to (32) with the global estimate of the schedule length $LBC_{\max}$ [1, p. 139]. In order to compare the obtained values $C_{\max}$ with the lower bound of the schedule length $LBC_{\max}$, the value of the γ coefficient has been determined. This coefficient, which is the deviation of the schedule length from the estimate from the bottom of the length of the schedule has been defined in the relationship (33). The average values of this coefficient for the built methods are given in Table 3. The header of the table gives the markings of the mathematical models used to solve the test examples.

(33) $$\gamma=\frac{C_{\max}-LBC_{\max}}{LBC_{\max}}\cdot 100\%$$

Table 3. Average values of index γ [%]

	Group	M1a, M2a	M1a, M2b	M1a, M2c	M1b, M2a	M1b, M2b	M1b, M2c
For the C1 method	1	13.7	16.8	22.7	9.3	12.8	17.8
	2	12.7	16.5	21.0	9.1	11.9	16.4
	3	12.5	15.6	21.4	8.8	10.5	15.7
	4	11.9	15.8	21.2	7.9	9.8	15.4
	Group	M3a, M4a	M3a, M4b	M3a, M4c	M3b, M4a	M3b, M4b	M3b, M4c
For the C2 method	1	10.0	13.8	19.3	6.3	8.3	14.4
	2	9.1	13.6	18.1	6.0	8.4	13.8
	3	8.4	12.9	17.4	5.7	7.8	12.3
	4	8.2	12.8	17.3	5.4	7.7	12.3

The analysis of the results given in Table 3 shows that when the C1 method is used, longer makespans have been obtained than in case of the C2 method. For the systems with the buffer (the models M2a, M4a), the value of the deviation γ did not exceed for the C2 method: 10% for fixed routes and 6.3% for alternative routes, and for the C1 method: 13.7% for fixed routes and 9.3% for alternative routes. After modification of the data, which consisted in taking into account the settings of two buffers before each stage (one product may be waiting in the buffer), the value of the deviation γ changed slightly and did not exceed 10.5% for the C2 method and 14% for the C1 method.

The average values of the deviations given in Table 3 show that the longest schedules were determined for the systems in which breaks are forbidden between the execution of the consecutive tasks for each product. Here, the average value of the deviation γ did not exceed 22.7%

for the C1 method and 19.3% for the C2 method. However, it should be emphasized than "no-waiting" scheduling always result in construction of longer schedules. The makespans for "no-waiting" scheduling were 8 ÷ 12% longer than the times of task schedules for the systems with the buffer. With blocking of the machines by the products awaiting the execution of the task allowed, the schedule length increased by 4 ÷ 9% as compared with the makespan for the systems with buffers.

The results given in Table 3 show also the effect of the allowed types of assembly routes on the schedule length of the assembly tasks. In case of fixed production routes, the makespans were longer than in case of alternative routes, for which differences in loading of individual machines are smaller. For the test examples, the schedule length increased to 12% (in reference to alternative routes) in case of assembly planning for fixed routes.

The lengths of the schedule built according to the C2 method constituted 89% ÷ 94% of the lengths of schedules obtained with the C1 method and the schedules with the C2 method were on the average 7.8% shorter than the solutions obtained for C1. Assignment of the possibility of allocation of the tasks to the machines in the lower level of the C2 method had to be compensated with the calculation time longer by about 20% ÷ 30% as compared with the calculation times for the C1 method. The calculation times for "no waiting" scheduling were longer by about 28% than the times of solving problems for the systems with the possibility of blocking the machines and by about 40% longer than the calculation times for the systems with the buffers.

Scheduling of the tasks was also compared with the optimum solutions. The Johnson algorithm [13] was used here for the systems with two stages (the groups 1 and 2 of the test problems, Table 2) and for the systems with three stages (the groups 3 and 4 of the test problems). In case of the comparison with the Johnson algorithm, the makespans were longer by 4.8 ÷ 6.9% (on the average by 6.5%) for the C1 method and 3.3 ÷ 5.5% (on the average by 4.4) for the C2 method. The comparison related to the systems with three assembly stages presented extending the makespans by 4.7 ÷ 6.6% (on the average by 5.9%) for the C1 method and 3.2 ÷ 5.3% (on the average by 4.3) for the C2 method.

In comparison of the developed methods with the known algorithm, attention has to be paid to the advantages related to the possibilities of the developed methods. The C1 and C2 methods take into consideration limited availability of the machines, and the tasks which require the use of parts the feeder are set aside from the other tasks. Moreover, the presented methods are used for selection of the assembly plans.

7. Conclusion

The presented comparison of the two methods allowed measuring of shortcomings and advantages of two alternative approaches to assembly planning. The results of the comparison proved that leaving a larger space for the level of task scheduling positively affects the length of scheduling assembly tasks. Assigning the tasks to the machines at the lower level increases duration of the calculations in reference to machine loading at the upper level, but the schedules are shorter.

In the mathematical models built for the lower levels of the methods, the time criterion has also been taken into account. The scheduling problem uses also, obviously enough, other criteria, for example related to costs. Using the available resources, including machines, parts feeders, is obviously related to incurring some costs.

The built mathematical models may be modified, expanded, and used for construction of other algorithms. In order to shorten the time of calculations for the problems of significant size, application of the developed models in relaxation heuristics is recommended. An example of such an algorithm in which binary variables are replaced with continuous variables and then the integral solutions are determined with specific rules is given in the work [14].

The observed development of computer technology, software and algorithms [15] promotes the development of methods based on integer programming which include the presented concepts of assembly planning. The discrete optimization packages feature higher calculation power, which allows solving of problems of increasing sizes and significant shortening of calculation time.

REFERENCES

[1] Sawik T., Production Planning and Scheduling in Flexible Assembly Systems. Springer-Verlag, Berlin (1999)

[2] Blazewicz J., Ecker K., Pesch F., Schmidt G., J. Weglarz, Handbook Scheduling. Springer, Berlin (2007)

[3] Pochet Y., Wolsey L.A., Production planning by mixed integer programming. Series in Operations Research and Financial Engineering, Springer, New York (2006)

[4] Abdullach T. A., Popplewell K., Page C.J., A review of the support tools for the process of assembly method selection and assembly planning, *International Journal of Production Research* 41 (2003), n.11, 2391-2410

[5] Barutçuoğlu Ş., M. Azizoğlu M., Flexible assembly line design problem with fixed number of workstations. *International Journal of Production Research* 49 (2011), n.12, 3691-3714

[6] 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

[7] Magiera M.: Heurystyczna metoda wyboru dostawców i środków transportu dla elastycznych systemów montażowych, *Technologia i automatyzacja montażu*, z. 2-2012 (76), IMBiGS, Warszawa (2012), 20-27

[8] Quadt D., Kuhn H., A taxonomy of flexible flow line scheduling procedures. *European Journal of Operational Research* 178 (2007) n.3, 686-698

[9] Phanden R., Jain A., Verma R., An approach for integration of process planning and scheduling. *International Journal of Computer Integrated Manufacturing* 26 (2013), n. 4, 284-302

[10] Magiera M., Analiza porównawcza dwóch metod sterowania przepływem produktów przez elastyczną linię montażową z ograniczoną dostępnością maszyn. *Automatyzacja procesów dyskretnych. Sterowanie procesami dyskretnymi. Zarządzanie i inżynieria produkcji*, pod red. M. Zaborowskiego. Wydawnictwa Naukowo-Techniczne, Warszawa (2004), 115-123

[11] Magiera M., Hierarchiczna metoda planowania produkcji dla systemu przepływowego bez magazynów. Wydawnictwo Politechniki Śląskiej, *Automatyka*, z.144 (2006), 57-62

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

[13] Johnson S.M., Optimal two- and three stage production schedules with setup times included. *Naval Research Logistics Quarterly* (1954), n.1, 61-68

[14] Magiera M., Methods of supplier selection for producers of electric and electronic equipment. *Przegląd Elektrotechniczny* (2013), nr 89, 66–70

[15] Kwiecień J., Filipowicz B., Algorytmy stadne w optymalizacji strukturalnej systemów niezawodnościowych. *Przegląd Elektrotechniczny* (2014), nr 6, 195-197

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