Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Towards the Leveling of Multi-Product Batch Production Flows. A Multimodal Networks Perspective

Bocewicz, G. K.; Nielsen, I. E.; Smutnicki, C.; Banaszak, Z. A.

TOWARDS THE LEVELING OF MULTI-PRODUCT BATCH PRODUCTION FLOWS. A MULTIMODAL NETWORKS PERSPECTIVE.

G. K. Bocewicz*, I. E. Nielsen.**, C. Smutnicki***
Z. A. Banaszak*

*Faculty of Electronics and Computer Science, Koszalin University of Technology, Poland,
bocewicz@ie.tu.koszalin.pl; z.banaszak@wz.pw.edu.pl
**Department of Materials and Production, Aalborg University, Denmark,
izabela@make.aau.dk
***Faculty of Electronics, Wroclaw University of Technology, Poland
czeslaw.smutnicki@pwr.edu.pl

Abstract: The problem studied in this paper is a cyclic job-shop problem with multiple Automated Guided Vehicles (AGVs). Job batches, which follow specific production routes, are processed, in the order of their operations, on multiple machines with standard processing times, and the fleet of AGVs perform the transportation operations of moving these job batches between the workstations. In this system, part sets of items are produced at fixed time intervals (takts). In the adopted model of the system, one can distinguish a layer of station-to-station transport, which is a network of local loops connecting subsets of workstations serviced cyclically by dedicated AGVs, and a layer of repetitive production flows which comprise job batches following a given set of production routes. The relationship between the elements of the structure of the system being modelled and its potential behavior is described by a system of integer equations. The resulting solutions enable the fast evaluation of production flow parameters including part sets, takt time, as well as repetitive-flow balancing aimed at maximization of the rate of system resource utilization. The high efficiency of the proposed approach, enabling the online prototyping of the production flow, is a consequence of omitting the time-consuming calculation of the sequencings of jobs within a cycle.

Keywords: cyclic robotic job shop, repetitive-flow balancing, takt time, flow time

1. INTRODUCTION

The planning and organization of multi-item repetitive flow production is one of the most difficult production problems, but at the same time it is a problem that is rarely studied. Cyclic job-shop problems, which are a unique and relatively little investigated subclass of job sequencing problems [20] – as well as the problems of production leveling and flow line balancing – [2, 19] are good examples here. These issues have seen rapidly growing interest both on the part of practitioners and theorists, mainly because of their great practical significance and the difficulty of constructing effective algorithms for solving them. It is noteworthy that the NP-completeness and/or strong NP-harness of the simplest versions of the mentioned classes of problems limit the scope of the application of exact algorithms to small-sized instances only [24].Given this background, the present study is an attempt to collate two perspectives: the basic (academic) perspective, focused on finding optimal solutions, in particular with respect to the minimization of the makespan (completion time); and the applied (production- and manufacturing-oriented) perspective, which concentrates on flow-time as the response time to user requests for job execution [10, 15, 23]. An observation that these two perspectives diverge is made in [3], which emphasizes the multi-objective character of the problems under consideration. The author of that study notes that since the makespan is the maximum value of the completion time of all jobs, and flow-time is the sum of the completion times of all jobs, hence the minimization of

makespan results in the maximization of flow-time. Another example of this discrepancy is given in Section 3, which discusses a case in which an organization of production flow that guarantees minimal flow time makes it impossible to achieve a minimum takt time. The two foci of this example are repetitive flow, in which a batch (a set) of items is produced at fixed time intervals (takt times), and its generalization, based on the concept of *heijunka* (production leveling). Production leveling is usually understood as a form of cyclic scheduling and is defined as "the distribution of production volume and mix evenly over time" [23]. In the case under consideration, flow time is understood as a period required for completing a specific job or a defined amount of work, i.e., the amount of time equal to the time it takes to finish one unit of product. If there is more than one path through the process, the flow time is equivalent to the length of the longest path. In turn, by takt time, we mean the desired time between the units of production output, synchronized to customer demand.

In further parts of this work, it is assumed that in the multi-machine cyclic production system considered, any element from a fixed batch (mix) passes in a predefined technological sequence through machines along a given technological route. The system consists of m workstations (machines with unitary throughput) designated as $R = \{R_1, \ldots, R_k, \ldots, R_m\}$. The system is designed to cyclically (repetitively) perform n jobs (production processes) given by the set $W = (W_1, \ldots W_i, \ldots W_n)$. A set of jobs performed in a single cycle is called a Part Set (PS). PS items are processed one after another

in a cyclic manner, providing a batch of jobs (a part set) in quantities appropriate for each cycle. That is because the processes must be able to be scaled to takt time, or the rate of customer demand.

All the job batches are processed in order of their operations on multiple machines with standard processing times, and the fleet of AGVs perform transportation operations between the workstations. The goal is to find an AGV fleet assignment for transportation operations and a sequencing of the AGVs' moves that can minimize production takt time, i.e., the time after which the next mix (part set) of the same elements can be produced (maximizing the throughput rate), and ensure a balanced flow (repetitive-flow balancing). Here, production flow balancing is used to mean "leveling work content" between all resources. Throughput rate (flow rate), in turn, is understood as the number of flow units passing through the production (manufacturing) process per unit in time.

In the declarative model of the production system considered in this study, one can distinguish a layer of station-to-station transport, which is a network of local loops connecting subsets of workstations serviced cyclically by dedicated AGVs, and a layer of repetitive production flows, which comprises sequences of operations, executed as part of the processes of moving/processing job batches, following a given set of production routes. The model, which implements a set of integer equations describing the behavior of the system being modelled, can be used to determine the set of permissible solutions characterizing the batch sizes of the PSs produced under the specific constraints imposed by the structure and organization of the technical appliance used (AGVs, workstations, intermediate storage facility, load/unload devices, and so on). In summary, this model allows one to determine the permissible size of *PS* batches and the takt times which determine the production completion times for those batches, as well as scenarios that allow repetitive flow-balancing aimed at leveling the production and increasing the rate of resource utilization. In particular, in situations in which the parameters of the production system) characterizing its machinery and technical layout (including operation times, buffer capacity, and so on) as well as the production and transportation routes are known, this model makes it possible to seek answers to the following questions: *What batches of PSs can be produced at what takt times in a system with a given structure and a given organization of the technical appliances used within it? (And in particular: What cyclic schedules can result in the production of feasible [in terms of composition and batch sizes of component products] PSs?) What execution times of technological and transport operations, and what capacity of intermediate storage buffers guarantee the production of specified compositions of PSs completed at the given takt time?*

In this context, the present work is a continuation of previous research on production flow balancing [7] and the robust scheduling of multi-item cyclic production [5, 8]. The novelty of this work is based on the formulation and analysis of a model of integer equations, a model that enables one to depart from the analysis of events represented in a Gantt chart (by omitting the calculation of the timings and sequences of jobs within a takt time), thus allowing a radical reduction in the size of the search space of permissible solutions.

The remainder of this paper is organized as follows: Section 2 provides a brief overview of the related research. Section 3 presents a model of considered batch flow production system and a dedicated for it constraint satisfaction problem. Section 4 provides computational experiments illustrating the proposed approach to PS batches cycle time scheduling and repetitive-flow balancing. Finally, Section 5 offers some concluding remarks.

2. RELATED WORK

Numerous papers address the issue of optimization of cyclic scheduling problems, such as the Basic Cyclic Scheduling Problem (BCSP) [16, 20] and its extensions associated with scheduling in production job-shops (so-called Job-shop Problems): the general Cyclic Job-shop Problem [4], the Cyclic Flow-shop Problem [9, 20], and the Cyclic Open-shop Problem [18]. With the exception of BCSP, all the problems listed above belong to the class of NP-hard problems, which means their solution requires the use of Artificial Intelligence methods [12, 14, 19, 20, 22]. The widely-recognized advantages of cyclic scheduling include, in particular, more efficient material handling, better station utilization, and simpler shop floor control, which however all come at the cost of schedule inflexibility, which means cyclic schedules are difficult to modify, and overallocation can cause a whole schedule to fail. Also, cyclic scheduling is not very suitable for systems with both periodic and aperiodic jobs. A special place in this context is occupied by studies on models oriented towards cyclic production methods in which a set of units is produced at fixed time intervals [11, 21]. The optimization of a process usually amounts to minimizing the cycle period or ensuring that a certain (usually Minimum) Part Set (MPS) of products is manufactured [1]. An alternative to this predominant, academic approach, is a paradigm based on the concept of heijunka, requiring that a base period referred to as EPEI (Every Part Every Interval) be determined, in which the whole mix of products has to be produced [23]. The core idea of the principle of production leveling is the heijunka box. It is a cyclic production schedule divided into a grid of boxes in which the columns represent a specific period of time and the rows represent product types. Attention is especially paid to the cyclic job-shop scheduling problem with multiple AGVs, which is recognized as being an effective way of processing various manufacturing and transportation processes, including those where setup and transportation times are relevant [20, 13, 10, 18]. In this context, a special interest should be taken in studies devoted to frameworks combining path routing and AGV fleet scheduling, which provide an integrated and unified approach to the modelling and design of cyclic production flow schedules ensuring efficient material handling, better workstation utilization, and simpler shop floor control [20]. The models and the methods developed on this basis, which are used in solving cyclic scheduling problems, include operational research techniques, such as branch-and-bound search or mixed-integer programming [24], or artificial intelligence techniques, such as constraints programming [5, 24], tabu search [14] and ant colony [12]. An alternative approach is offered by methods based on Max-plus algebra. The formalism of this algebra can be used to model and analyze a production system within the linear framework,

allowing, for example, the use of Petri nets to simulate system behavior. The major advantage of this framework is that it eliminates the need to formulate scheduling problems as non-linear optimization problems [5, 7, 8]. Its main drawback, however, is that it has limited potential use for the analysis of production flows in models that do not take into account the structures responsible for material handling. From this point of view, the approach presented in this study, in which both production flows and the accompanying transport operations between pairs of workstations are modelled by a set of linear/nonlinear integer equations, may also bring interesting research options. Models of this type allow for the quick determination of the basic parameters (takt time, size of part sets, etc.) of permissible production flow variants in a cyclic job-shop problem with multiple AGVs. The advantage of the approach discussed here, however, comes at the cost of a lack of solutions determining the sequencing and timing of the particular jobs performed within a takt time period.

3. MODELLING

3.1. Motivating Example

Given below is a multi-item batch flow production system composed of five stations $R = \{R_1, \dots, R_5\}$, as shown in Fig. 1. This system cyclically performs two jobs related to the execution of two production processes: $W = (W_1, W_2)$. The processes are executed along the following routes: $W_1 = (R_1, R_3, R_5)$, $W_2 = (R_2, R_1, R_4, R_5)$. The execution times $t_{i,j}$ of workstation operations $O_{i,j}$ (j-th operation executed along the technological route of i-th product) marked in Fig. 1 are given in Table 1. A system of equations describing the steady state of the flow production of two products is shown below [7]:

$$TP = \max\{tz_1 \dots tz_5\};$$
$$t_{1,1} \cdot b_1 + t_{2,2} \cdot b_2 = tz_1; \quad t_{2,1} \cdot b_2 = tz_2; t_{1,2} \cdot b_1 = tz_3;$$
$$t_{2,3} \cdot b_2 = tz_4; t_{1,3} \cdot b_1 + t_{2,4} \cdot b_2 = tz_5; \quad (1)$$

where: tz_k – resource occupation time R_k, $t_{i,j}$ – time of j-th operation of job W_i, b_i – batch size for W_i.

It is easy to see that when operation times are assumed to have integer values, the model considered (1) forms a system of integer equations. The set of solutions of this system includes: $b_1 = b_1^{(1)} = 1$, $b_2 = b_2^{(1)} = 1$, $TP = TP^{(1)} = 7$; $b_1^{(2)} = 2$, $b_2^{(2)} = 2$, $TP^{(2)} = 14$; … ; $b_1^{(c)} = c \cdot b_1^{(1)} = c$; $b_2^{(c)} = c \cdot b_2^{(1)} = c$, $c \in \mathbb{N}$, $TP^{(c)} = c \cdot TP^{(1)} = c \cdot 7$. Each of the potential solutions of the system of equations includes a so-called part set, for example: $PS^{(1)} = (1, 1)$, produced during one takt period $TP^{(1)} = 7$. In the case discussed here, each job is performed once during one takt period.

Given the above part set $PS^{(1)}$, a schedule (understood as a set of starting times of job operations) in which takt period $TP^{(1)}$ can be feasibly obtained is one of the permissible solutions of the following system of inequalities.

$$x_{1,1}(k) \geq x_{1,1}(k-1) + t_{1,1}; \; x_{1,2}(k) \geq x_{1,2}(k-1) + t_{1,2};$$
$$x_{1,2}(k) \geq x_{1,1}(k) + t_{1,2}; \quad x_{1,3}(k) \geq x_{1,3}(k-1) + t_{1,3};$$
$$x_{1,3}(k) \geq x_{1,2}(k) + t_{1,2}; \quad x_{2,1}(k) \geq x_{2,1}(k-1) + t_{2,1};$$
$$\left(x_{1,1}(k) \geq x_{2,2}(k) + t_{2,2}\right) \vee \left(x_{2,2}(k) \geq x_{1,1}(k) + t_{1,1}\right)$$
$$x_{1,1}(k) \geq x_{2,2}(k-1) + t_{2,2}; x_{2,2}(k) \geq x_{1,1}(k-1) + t_{1,1}$$

$$\left(x_{1,3}(k) \geq x_{2,4}(k) + t_{2,4}\right) \vee \left(x_{2,4}(k) \geq x_{1,3}(k) + t_{1,3}\right)$$
$$x_{i,j}(k) = x_{i,j}(k-1) + TP^{(1)}, i = 1,2; \; j = 1 \dots 4$$

where: $x_{i,j}(k)$ – starting time of j-th operation of job W_i in k-th cycle, $TP^{(1)}$ – takt time for part set $PS^{(1)}$, $t_{i,j}$ – time of j-th operation of job W_i (given in Table 1).
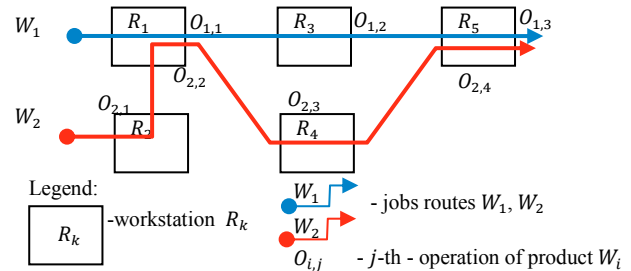

Fig. 1. Structure of a multi-item batch flow production system

Table 1 Workstation operation times $t_{i,j}$

	R_1	R_2	R_3	R_4	R_5
W_1	$t_{1,1} = 3$		$t_{1,2} = 6$		$t_{1,3} = 4$
W_2	$t_{2,2} = 4$	$t_{2,1} = 4$		$t_{2,3} = 3$	$t_{2,4} = 3$

These inequalities describe sequence relations between the job operations W_1 and W_2 in the system of Fig.1. By finding solutions which satisfy the inequalities, one can determine operation starting times that will result in flow production (with takt time $TP^{(1)}$) of the given part sets $PS^{(1)}$. The jobs are performed on the assumption that: (1) each station can perform at most one operation per unit of time, (2) each operation can be performed on at most one station per unit of time, and (3) operations may not be interrupted during their execution. Examples of two achievable schedules are shown in Figs. 2 and 3. Fig. 2 shows a schedule that guarantees a minimum production rate $TP^{(1)} = 7$ at flow times $FT_1^{(1)} = 13$ (for job W_1) and $FT_2^{(1)} = 17$ (for job W_2). Fig.3, on the other hand, shows a schedule with a longer takt time ($TP^{(1)} = 8$) but a shorter flow time of job W_2: $FT_2^{(1)} = 16$ (value $FT_1^{(1)}$ is unchanged). It should also be noted that in the first case there are two bottlenecks R_1 and R_5, and in the second one, there are two critical processes (i.e., processes in which component operations are executed without downtime). Of course, the workstations are more fully utilized in the first case. It is worth noting that changes in production flow caused by increasing the batch size to integer multiples of batch size $PS^{(1)}$ do not affect the utilization rate of the system resources. Increased batch size leads to longer waiting times for the pickup/delivery of successive batches at non-bottleneck workstations, as illustrated in Fig. 4. This means that by changing the size of production batches while keeping the bottleneck of the system unchanged, one can increase the availability of other non-bottleneck workstations. The surplus availability of workstations obtained in this way makes it possible to produce an additional range of products.

3.2. Model formulation

Fig. 5 shows the structure of a multi-item batch flow production system (Fig. 5c) in which two layers have been distinguished: the production flow layer (Fig. 5a) and the layer of transport and storage of production batches (Fig. 5b).
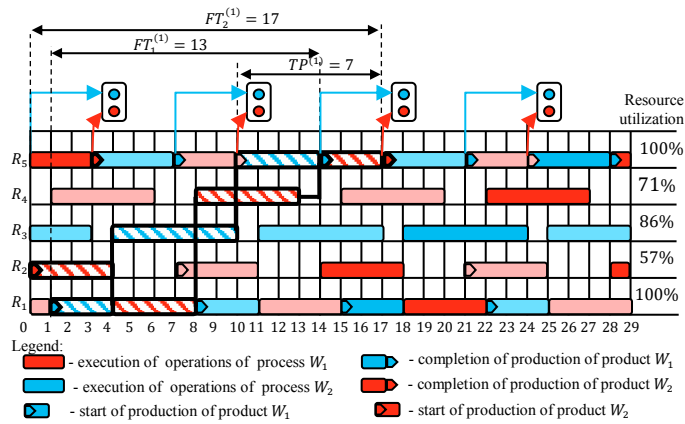
Fig. 2. Gantt diagram illustrating the execution of process operations for flow production of part set $PS^{(1)} = (1,1)$ with $TP^{(1)} = 7$ u.t.
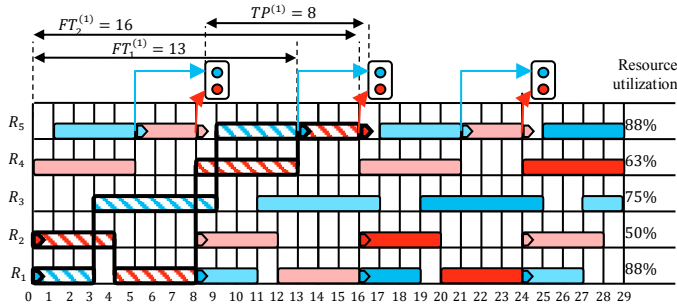

Fig. 3. Gantt diagram illustrating the execution of process operations for flow production of part set $PS^{(1)} = (1,1)$ with $TP^{(1)} = 8$
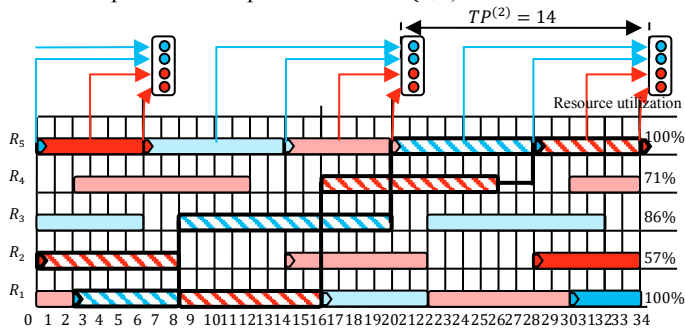

Fig. 4. Gantt diagram illustrating how process operations are executed in flow production of part set $PS^{(2)} = (2,2)$, with production takt time $TP^{(2)} = 14$ u.t.

Items are transported between workstations by AGVs periodically servicing the selected workstations, i.e., neighboring workstations linked via local cyclic transportation routes. This means that the production capacity of the system (i.e., the size of the production batches and the value of the production takt time) are further limited by the capacity of the available transport subsystem.

Assuming that the adopted mechanism of synchronization of the AGVs does not allow for resource conflicts leading to deadlocks and/or starvation, and assuming that the transport operations and the related loading/unloading operations are carried out at a takt time governed by the production system's bottleneck, the model of this system is determined by the following parameters:

Sets:

R: the set of resources (workstations), indexed by k,

W: the set of jobs, (products of multimodal processes) indexed by i,

P: the set of transportation means e.g. AGVs (local processes), indexed by j,

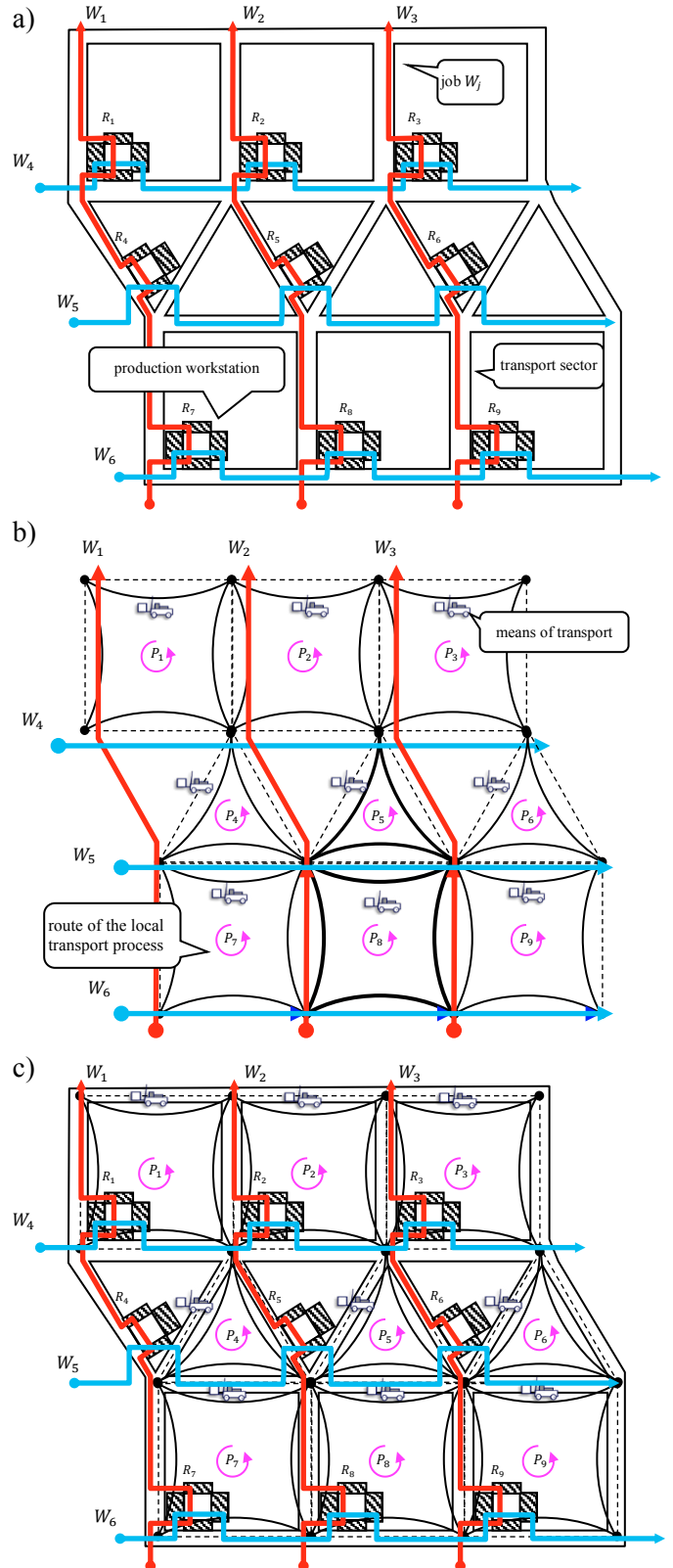Q_k: the set of jobs using resource k.

Fig. 5. A multi-item batch flow production system: a) production flow layer, b) transport layer, c) structure encompassing both the transport and the production flow layers

Parameters:

m: number of resources,

n: number of products manufactured,

R_k: resource k,

W_i: job i,

q_k: number of jobs using resource k, $q_k = |Q_k|$,

$ts_{i,k}$: operation time for one item of product W_i on resource R_k,

Δt: travel time between workstations,

tu: loading/unloading time,

ti_k: time of transport operations executed on resource R_k,

T: period of local process executions, $T \leq ti_k$,

TP^*: maximum value of production takt time TP

Variables:

PS: sequence of batch sizes of part set $PS = (b_1, \dots, b_i, \dots, b_n)$; to simplify the notation, it was assumed that PS stands for $PS^{(1)}$ (i.e. the sequence of batch sizes produced once in a production cycle),

b_i: size of production batch of product W_i, $b_i \in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$ ($b_i = 0$ means that job W_i is not being executed), value of b_i determines also the capacity of intermediate storage buffers.

TP: production takt time of PS; to make the notation simpler, it was assumed that TP means $TP^{(1)}$ (i.e. the production takt time for part set $PS^{(1)}$),

tz_k: resource occupation time R_k, if the following condition is satisfied for resource R_k: $\forall R_c \in R : tz_c \leq tz_k$, then resource R_k is called the system's bottleneck) ,

$to_{i,k}$:   workstation processing time for batch (b_i) of product W_i, on resource R_k.

The relationships among the above variables are described by the following constraints:

• Takt time TP of a system is determined by the bottleneck - resource R_k with the longest occupation time tz_k,

$$TP = max\{tz_1, \dots, tz_k, \dots, tz_m\} ,   (1)$$

where: tz_k – resource occupation time R_k representing the sum of times $to_{i,k}$ and ti_k: times of operations related to the processing of products of set Q_k (i.e. the set of products using resource R_k); times of operations associated with delivering and picking up products to and from resource R_k:

$$tz_k = \sum_{i \in Q_k} to_{i,k} + ti_k .   (2)$$

• Workstation processing time $to_{i,k}$ for a batch of product W_i on resource R_k:

$$to_{i,k} = b_i \cdot ts_{i,k}   (3)$$

• Transport operation time ti_k (delivery/pickup and loading/unloading of product) on resource R_k:

$$ti_k = q_k \cdot (2t_{ul} + \Delta t)   (4)$$

Times tz_k, $to_{i,k}$ and ti_k are shown in Fig. 6.

The periodic flow of production which is governed by the takt time of completion of the successive part set batches (the production flow layer, Fig. 5a) is of course determined by the period of the system of simultaneously executed local cyclic processes (the transport layer Fig. 5b).
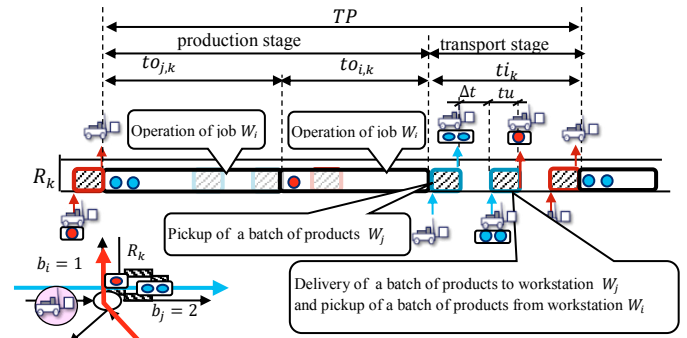

Fig. 6. Execution of operations at workstation R_k, including transport operations (batch delivery and pickup operations)

In a system with a transport structure in which the transport of products takes place after the production stage has been completed and in which the movement of the AGVs is collision- and deadlock-free (as in Figure 5), the following condition is satisfied:

$$mod\{TP, T\} = 0, \text{ for } = 1 \dots m .   (5)$$

According to this condition, production takt time TP is a multiple of the period T of AGVs, during which products are delivered to workstations. The satisfaction of this condition enables the synchronization of simultaneous deliveries to multiple workstations – as illustrated in Figure 7.
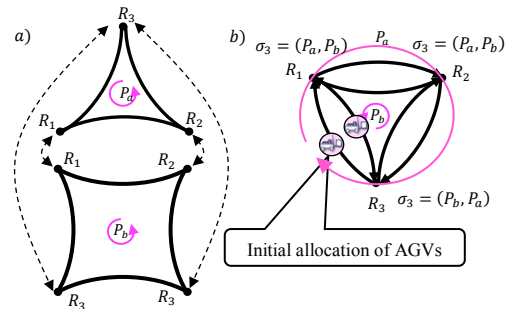

Fig. 7. a) An elementary substructure of structure from Fig. 4, and b) its convoluted representation

For the purposes of this work, it is assumed that the times of the particular station-to-station transport operations and the accompanying loading/unloading activities, which are components of the production cycle, are fixed (independent of the size of part set batches), and that their aggregate share in the transport stage does not exceed 20% of the takt time of the production stage when production is executed on the system's bottleneck. The cyclic, deadlock-free and starvation-free execution of concurrently flowing local periodic transport and delivery/pickup processes creates conditions which determine the admissible initial allocation of AGVs and a set of priority rules which resolve resource conflicts (synchronize the order of access of the AGVs to shared resources). The method of determining such sufficient conditions, based on the concept of convoluting the elementary substructure of the local-transport network, is presented in article [6]. For example, in the network shown in Fig. 5b, one can identify the elementary substructure of the form shown in Fig. 7a and 7b, and the corresponding convoluted form as presented in Fig. 7c, to determine the initial allocation in which AGVs allocated to

transport sectors (as in Fig. 7c) will perform their operations at production workstations following the dispatching priority rules: $\sigma_1 = (P_c, P_b, P_a)$, $\sigma_2 = (P_a, P_d, P_c)$, $\sigma_3 = (P_b, P_a, P_d)$, and $\sigma_4 = (P_d, P_c, P_b)$ where: σ_k is a dispatching priority rule that specifies the order of access of the processes to R_k.
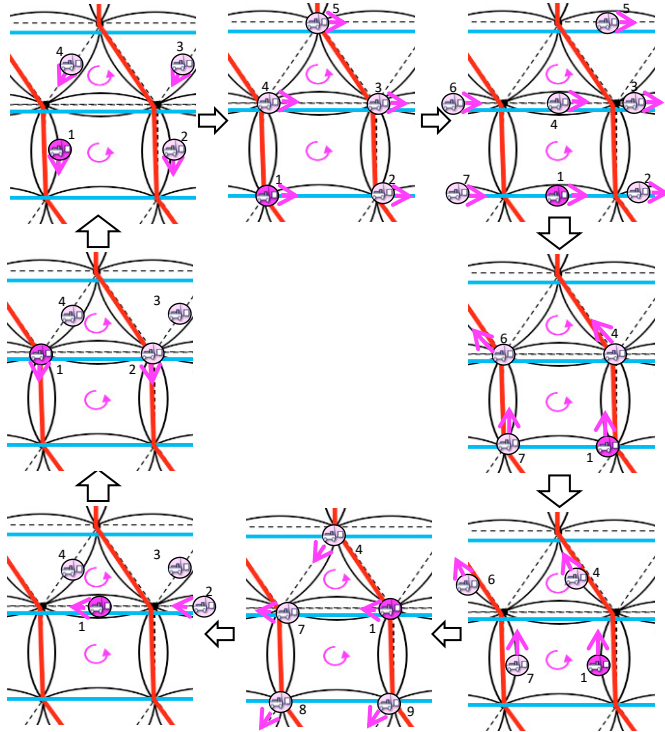

Fig.8 Flow of production batches between workstations in the system from Fig. 5

The cyclic nature of the processes occurring in the transport and production flow layers (Fig. 5c) is also illustrated in Fig. 8. Focusing on the elementary, repetitive structures of the production flow network and the network of local transport processes, this figure shows 8 successive phases of simultaneously executed operations of station-to-station transport. Due to limited space, further considerations are confined to the production flow layer. Assuming that the appropriate conditions guaranteeing a deadlock- and starvation-free execution of transportation operations performed by the fleet of AGVs are satisfied, the formulation of the following problem is considered.

3.4. The Constraint Satisfaction Problem

Consider the multi-item batch flow production system shown in Fig. 5, which has a regular structure and parameters TP^*, $ts_{i,k}$, Δt, t_{ul}, Q_k, q_k, $i = 1 \dots n$; $k = 1 \dots m$. The behavior and structure of the system is described by the set of constraints (1)–(5). The goal is to find an answer to the following question: *Does a non-empty set of PSs exist that can be produced with takt times $TP \leq TP^*$?*

$$C = \begin{cases} TP = \max\{tz_1, \dots, tz_k, \dots, tz_m\}, \ k = 1 \dots m \\ tz_k = \sum_{i \in Q_k} to_{i,k} + ti_k, \ i = 1 \dots n, \ k = 1 \dots m \\ to_{i,k} = b_i \cdot ts_{i,k}, \ i = 1 \dots n, \ k = 1 \dots m \\ ti_k = q_k \cdot (2tu + \Delta t), i = 1 \dots n, \ k = 1 \dots m \\ mod\{TP, T\} = 0, k = 1 \dots m \\ TP \leq TP^* \end{cases} \quad (11)$$

In other words, the goal is to find sets of the form $= \{b_i \mid i = 1 \dots n\}$, the elements of which form sequences of permissible batch sizes of mix $PS = (b_1, \dots, b_i, \dots, b_n)$.. To solve the above decidability problem, it is enough to solve an appropriate corresponding CS problem.

$$CS = (B, D, C), \quad (12)$$

where $B = \{b_i \mid i = 1 \dots n\}$ is a set of decision variables (batch sizes of PS), $D = \{d_i \mid i = 1 \dots n, \}$ is a set of domains of variables b_i, and C is a set of constraints (11).

It is worth noting that the solutions obtained for the different values of TP^* make it possible to consider feasible variants of production flow balancing. In particular, they allow one to determine different batch sizes of part sets produced in the given takt time, to evaluate the rate of system resource utilization and to estimate the possibility of starting additional production. Also note that the fact that mix $PS^{(c)}$ for $c > 1$ is a multiple of mix $PS^{(1)}$ can be exploited to determine takt time $TP^{(c)}$ from the following relationship:

$$TP^{(c)} = c \cdot TP^{(1)} + ti_k \cdot (1 - c) + \delta, \quad (13)$$

where $c \in \mathbb{N}$, a multiple of batch size of mix $TP^{(1)}$, $\delta \in \mathbb{N}$, $\delta < (ti_k + \Delta t)$ – a correction taking into account constraint $mod\{TP^{(c)}, T\} = 0$.

4. COMPUTATIONAL EXPERIMENTS

Given here is a system such as the one shown in Fig. 5, in which six jobs (W_1–W_6) are executed; workstation operation times $t_{i,j}$ for these jobs are given in Table 2. Product batches are transported by 15 industrial trucks (P_1–P_{15}). It is assumed that station-to-station transport cycle times Δt and loading/unloading times t_{ul} equal 1, i.e. $\Delta t = t_{ul} = 1$ unit of time (u.t.). Each resource R_k is used to execute two workstation operations (two production routes intersect at each workstation) and therefore the number of jobs using resource k is $q_k = 2$. This means that the duration of transport operations for each station is $ti_k = 5$ u.t and period $T = 8$ (see Fig. 8). The goal is to find an answer to the following question: *Does there exist, for such a system, a non-empty set of PSs (part sets) that can be produced in takt times of $TP \leq 35$ u.t.?* To answer this question, it is necessary to solve CS problem (12), for which the set of constraints C takes the form:

$8b_4 + 4b_1 + 5 = tz_1$; $6b_4 + 2b_2 + 5 = tz_2$; $4b_4 + 6b_3 + 5 = tz_3$; $6b_5 + 8b_1 + 5 = tz_4$; $10b_5 + 3b_2 + 5 = tz_5$; $4b_5 + 7b_3 + 5 = tz_6$; $4b_6 + 5b_1 + 5 = tz_7$; $6b_6 + 4b_2 + 5 = tz_8$; $4b_6 + 8b_3 + 5 = tz_9$; $mod\{TP, 8\} = 0$; $TP = \max\{tz_1, \dots, tz_9\}$, $TP \leq 35$ u.t.

A solution that satisfies the above set of constraints is a set of sequences $PS = (b_1, \dots, b_6)$ of production batch sizes (for jobs W_1–W_6) which can be manufactured in the system of Fig. 5 in a cyclic manner at takt time TP of less than 30 u.t. The problem considered was implemented and solved in the constraint programming environment OzMozart (Windows 10, Intel Core Duo2 3.00 GHz, 4 GB RAM). The set of all permissible solutions includes 239 part sets, of which only 20 ensure that all of the jobs are involved in the production (in other cases at least one of the jobs is not executed i.e. there

exists a batch with a size of zero: $b_i = 0$). The first permissible solution (obtained in less than 1 second) has the form: $PS = (1,1,3,1,1,1)$. The resulting mix guarantees a smooth flow of production at a takt time of $TP = 32$ u.t. With production planned in this way, the Average Resource Utilization (ARU) is 49% (that means each resource is idle for a half production cycle - for simplification, we assume that all resources are equally important). In this context, it is only natural to ask another question: *Is there, among all the permissible solutions, a solution which allows us to increase the utilization of the system's resources ARU by increasing the size of the batches of selected groups of products when the production takt time remains unchanged?*

Table 2. Workstation operation times $t_{i,j}$ [t.u.]

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9
W_1	$t_{1,3}=4$			$t_{1,2}=8$			$t_{1,1}=5$		
W_2		$t_{2,3}=2$			$t_{2,2}=3$			$t_{2,3}=4$	
W_3			$t_{3,3}=6$			$t_{2,3}=7$			$t_{2,3}=8$
W_4	$t_{4,1}=8$	$t_{4,2}=6$	$t_{4,3}=4$						
W_5					$t_{5,1}=8$	$t_{5,2}=6$	$t_{5,3}=4$		
W_6							$t_{6,1}=8$	$t_{6,2}=6$	$t_{6,3}=4$

To answer this question, it suffices to determine the successive solutions to CS problem (12). As it turns out, such a mix exists among the solutions to the CS problem: $PS = (2,5,3,2,1,1)$. This mix is characterized by a larger production batch size of jobs W_1, W_2, W_4 (change in the values of: b_1 from 1 to 2, b_2 from 1 to 5 and b_4 from 1 to 2). As before, the production takt time is 28 u.t., but the rate of workstation utilization has increased to 74%. In other words, the production flow parameters were improved as a result of increasing the utilization of the work stations servicing job W_1, W_2, W_4 (workstations R_1-R_5, R_7, R_8). The set of all 20 permissible solutions is presented in Table 3. The mixes listed in Table 3 guarantee the same takt time $TP = 32$ u.t, but differ in the workstation utilization rates. The obtained set of solutions provides a basis for determining the production takt time $TP^{(c)}$ for successive part sets $PS^{(c)}$ with c-fold production batch sizes. Limiting the further considerations to mix $PS^{(1)} = (2,5,3,2,1,1)$, we can ask the following question: *What production takt time $TP^{(c)}$ and what workstation utilization rate is achievable for part sets $PS^{(c)}$?*

Relationship (13) makes it possible to determine production takt time $TP^{(c)}$ on the basis of production takt time $TP^{(1)}$. For example, the production takt time determined by mix ($c = 2$): $PS^{(2)} = (4,10,6,4,1,1)$ is equal to $TP^{(2)} = 64$ u.t. and the resource utilization rate is 74%. In turn, when mix $PS^{(1)}$ is produced three times in one production batch: $PS^{(3)} = (6,15,9,6,3,3)$, the production takt time is $TP^{(3)} = 88$ u.t. and the resource utilization rate is 81%.

To recapitulate, it can be seen that with the increase in batch size (i.e., for batch sizes which are c-fold multiples of the batch size of $PS^{(1)}$), the utilization rate of production workstations and the production takt time determining the completion time of part set $PS^{(c)}$ also increase. These observations are illustrated in Fig. 9. The results of the experiments, on the one hand, confirm the promise of the new approach proposed in this study, which offers solutions alternative to the scheduling methods commonly used in cyclic job-shop problem. On the

other hand, the results also encourage further in-depth study of issues such as the irregular, non-monotonous increase in the rate of utilization of the system's resources (see Fig. 9 a).

Table 3 A set of permissible solutions that guarantee the flow of production at takt time $TP = 32$ u.t (the mixes described in the example are highlighted in orange)

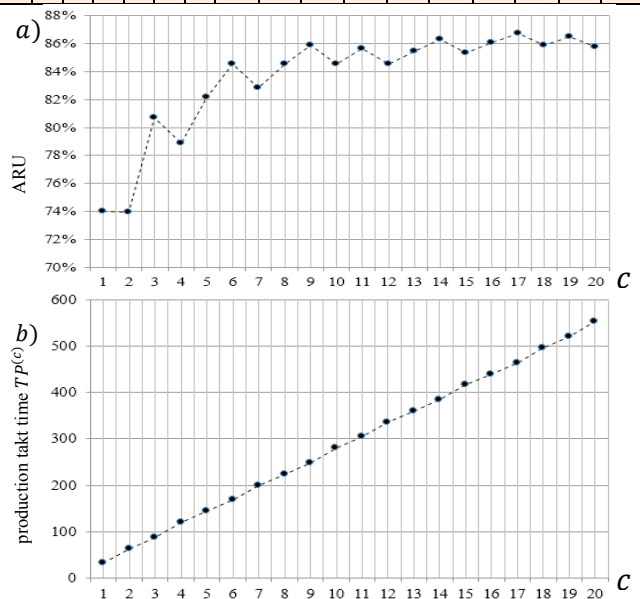| No.	Mixes of PS						Resource loading with workstation operations [u.t.]									mean
	b_1	b_2	b_3	b_4	b_5	b_6	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	
1	1	1	3	1	1	1	12 38%	8 25%	22 69%	14 44%	13 41%	27 84%	9 28%	10 31%	27 84%	49%
2	1	2	3	1	1	1	12 38%	10 31%	22 69%	14 44%	16 50%	27 84%	9 28%	14 44%	27 84%	52%
3	1	3	3	1	1	1	12 38%	12 38%	22 69%	14 44%	19 59%	27 84%	9 28%	18 56%	27 84%	56%
4	1	4	3	1	1	1	12 38%	14 44%	22 69%	14 44%	22 69%	27 84%	9 28%	22 69%	27 84%	59%
5	1	1	3	2	1	1	20 63%	14 44%	26 81%	14 44%	13 41%	27 84%	9 28%	10 31%	27 84%	56%
6	1	2	3	2	1	1	20 63%	16 50%	26 81%	14 44%	16 50%	27 84%	9 28%	14 44%	27 84%	59%
7	1	3	3	2	1	1	20 63%	18 56%	26 81%	14 44%	19 59%	27 84%	9 28%	18 56%	27 84%	62%
8	1	4	3	2	1	1	20 63%	20 63%	26 81%	14 44%	22 69%	27 84%	9 28%	22 69%	27 84%	65%
9	1	5	3	1	1	1	12 38%	16 50%	22 69%	14 44%	25 78%	27 84%	9 28%	26 81%	27 84%	62%
10	1	5	3	2	1	1	20 63%	22 69%	26 81%	14 44%	25 78%	27 84%	9 28%	26 81%	27 84%	68%
11	2	1	3	1	1	1	16 50%	8 25%	22 69%	22 69%	13 41%	27 84%	14 44%	10 31%	27 84%	55%
12	2	2	3	1	1	1	16 50%	10 31%	22 69%	22 69%	16 50%	27 84%	14 44%	14 44%	27 84%	58%
13	2	3	3	1	1	1	16 50%	12 38%	22 69%	22 69%	19 59%	27 84%	14 44%	18 56%	27 84%	61%
14	2	4	3	1	1	1	16 50%	14 44%	22 69%	22 69%	22 69%	27 84%	14 44%	22 69%	27 84%	65%
15	2	5	3	1	1	1	16 50%	16 50%	22 69%	22 69%	25 78%	27 84%	14 44%	26 81%	27 84%	68%
16	2	1	3	2	1	1	24 75%	14 44%	26 81%	22 69%	13 41%	27 84%	14 44%	10 31%	27 84%	61%
17	2	2	3	2	1	1	24 75%	16 50%	26 81%	22 69%	16 50%	27 84%	14 44%	14 44%	27 84%	65%
18	2	3	3	2	1	1	24 75%	18 56%	26 81%	22 69%	19 59%	27 84%	14 44%	18 56%	27 84%	68%
19	2	4	3	2	1	1	24 75%	20 63%	26 81%	22 69%	22 69%	27 84%	14 44%	22 69%	27 84%	71%
20	2	5	3	2	1	1	24 75%	22 69%	26 81%	22 69%	25 78%	27 84%	14 44%	26 81%	27 84%	74%

Fig. 9 a) ARU and b) production takt time $TP^{(c)}$ for c-fold multiples of batch size of part set $PS^{(1)}$.

5. CONCLUDING REMARKS

The approach presented in this study implicitly assumes that all of the operation times specifying production flows are integers. In that context, an implementation of the integer equations to modelling and providing variant solutions for part sets cycle time scheduling and repetitive-flow balancing seems to be justified. The model proposed in this study allows the bottom-up or top-down organization of production flow by integrating the level of station-to-station transport and the level of flows of batches of different (simultaneously manufactured) products. In the first case, the calculated production takt time determines the cycle of the local transport system. In the second case, the period of cyclically moving AGVs determines the takt time. In this model, which incorporates the concept of the levelling of the flow of the production processes, and is also based on the observation that a system's bottleneck determines its production flow takt time, the number of admissible part set batches processed within a given takt time as well as a system's resource utilization rate can be easily calculated. Moreover, the declarative character of the model enables the utilization of commercially available software tools, such as CPLEX/ECL^{i}PS^e/Gurobi, etc., and their easy supplementation and/or enlargement, depending on the specific context. Future research should be focused on finding sufficient conditions that would ensure the consistency of the set of integer equations which comprise an analytical model of a production flow configuration, while integrating the level of station-to-station transport with the level of flow of batches of different products. Apart from the research perspective presented in this article, other directions of study worth mentioning are those aimed at investigating the conditions that would allow one to reschedule cyclic production according to customers' changeable demands. The changes in demands may regard both delivery dates and the quantities of the batches ordered by customers. Other interesting areas of investigation for the future relate to the smooth transition between two successive cyclic steady states.

5. REFERENCES

1. Al-Ashhab S. M., An optimization model for multi-period multi-Product multi-objective production planning. International Journal of Engineering & Technology Vol.16, No.01, 43–56
2. Battaïa O., Delorme X., Dolgui A., Finel B., Grimaud F.. Flow Line Balancing Problem. Proc. Of the International Conference on Industrial Engineering and Systems Management, 1065-1071, 2015, DOI: 10.1109/iesm.2015.7380287
3. Bhuvaneswari M.C. (Ed.), Application of Evolutionary Algorithms for Multi-objective Optimization in VLSI and Embedded Systems, Springer 2015
4. Błażewicz J., Ecker K., Pesch E., Schmidt G., Węglarz J. Handbook on Scheduling: From Theory to Applications. Springer. 2007, 647
5. Bocewicz G., Banaszak Z., Nielsen I., Muszyński W., Re-scheduling of AGVs Steady State Flow, Preprints of the 20th World Congress The International Federation of Automatic Control Toulouse, France, July 9-14, 2017, 3552- 3557
6. Bocewicz G., Janardhanan M.Ni., Krenczyk D., Banaszak Z., Traffic flow routing and scheduling in a food supply network, Ind. Management & Data Systems, 117 (9), 2017, 1972-1994
7. Bocewicz G., Nielsen P., Banaszak Z., Wojcik R., An Analytical Modeling Approach to Cyclic Scheduling of Multiproduct Batch Production Flows Subject to Demand and Capacity Constraints. Advances in Intelligent Systems and Computing 656, 277-289
8. Bocewicz G., Nielsen I., Banaszak Z., Majdzik P., Cyclic scheduling approach to maintenance of production flow robustness. Advances in Mechanical Engineering, 2018, Vol. 10 issue 1, doi.org/10.1177/1687814017746245
9. Bożejko W., Uchroński M., Wodecki M. Block approach to the cyclic flow shop scheduling, Computers & Industrial Engineering, Issue 81, 158-166
10. Dileep G., Sumer D., Aggarwal C. Multi-Machine, Multi-Product Production Scheduling And Inventory Control. Decision Sciences, Vol.9, Issue 4, 1978, 577–595
11. Dobson G., Yano C.A., Cyclic scheduling to minimize inventory in a batch flow line. European Journal of Operational Research, Vol. 75, 1994, 441–461
12. Elmi A., Topaloglu S., Cyclic job shop robotic cell scheduling problem: Ant colony optimization. Computers & Industrial Engineering, Vol.111, 2017, 417-432
13. El-Najdawi M.K. Multi-cyclic flow shop scheduling: An application in multi-stage, multiproduct production processes. International Journal of Production Research, Vol. 35, Issue 12, 1997, 3323-3332
14. Greistorfer P., Hybrid Genetic Tabu Search for a Cyclic Scheduling Problem. In: Voß S., Martello S., Osman I.H., Roucairol C. (eds) Meta-Heuristics. Springer, Boston, MA, 1999
15. Klos S., Patalas-Maliszewska J. The Topological Impact of Discrete Manufacturing Systems on the Effectiveness of Production Processes. Advances in Intelligent Systems and Computing, Springer, Vol. 571, 2017, 441-452.
16. Kampmeyer T., Cyclic scheduling problems. Ph.D. Dissertation. Mathematik/Informatik Universit at Osnabruck, 2006
17. Korytkowski P., Grimaud F., Dolgui A., Exponential smoothing for multi-product lot-sizing with heijunka and varying demand. Management and Production Engineering Review, Vol. 5 (2), 2014, 20–26
18. Kubale M., Nadolski A. Chromatic scheduling in a cyclic open shop. European Journal of Operational Research. Vol. 164(3), 2015, 585–91
19. Li Z., Janardhanan M.N., Tang Q., Nielsen P. Co-evolutionary particle swarm optimization algorithm for two-sided robotic assembly line balancing problem, (2016) Advances in Mechanical Engineering, Vol. 8 (9), 1-14
20. Levner E., Kats V, Lopez A.P., Cheng T.C.E. Complexity of cyclic scheduling problems: A state-of-the-art survey. Computers and Industrial Engineering, 59, 2010, 352-361
21. Moreno M.S., Montagna J.M., A multiperiod model for production planning and design in a multiproduct batch environment. Mathematical and Computer Modelling, Vol. 49 (7–8), 2009, 1372–1385
22. Nilakantan, J.M., Li, Z., Tang, Q., Nielsen, P. Multi-objective co-operative co-evolutionary algorithm for minimizing carbon footprint and maximizing line efficiency in robotic assembly line systems, Journal of Cleaner Production, 2017, 156, 124-136.
23. Pascal D., Lean Production Simplified. CRC Press, 2007.
24. Sitek, P. and Wikarek, J., A hybrid programming framework for modeling and solving constraint satisfaction and optimization problems, Scientific Programming, Vol. 2016 No. 2016, Article ID 5102616