

New models and algorithms to solve integrated problems of production planning and control taking into account worker skills in flexible manufacturing systems

Norbert Tóth^a and Gyula Kulcsár^{b*}

^a*Division for SMART Systems, Bay Zoltán Nonprofit Ltd. for Applied Research, Hungary*

^b*Department of Information Engineering, University of Miskolc, Hungary*

CHRONICLE

Article history:

Received April 1 2021
Received in Revised Format
May 12 2021
Accepted May 12 2021
Available online
May, 12 2021

Keywords:

Production planning and control
Scheduling
Search algorithm
Worker skills
Flexible manufacturing system

ABSTRACT

The paradigm of the cyber-physical manufacturing system is playing an increasingly important role in the development of production systems and management of manufacturing processes. This paper presents an optimization model for solving an integrated problem of production planning and manufacturing control. The goal is to create detailed production plans for a complex manufacturing system and to control the skilled manual workers. The detailed optimization model of the problem and the developed approach and algorithms are described in detail. To consider the impact of human workers performing the manufacturing primary operations, we elaborated an extended simulation-based procedure and new multi-criteria control algorithms that can manage varying availability constraints of parallel workstations, worker-dependent processing times, different product types and process plans. The effectiveness of the proposed algorithms is demonstrated by numerical results based on a case study.

© 2021 by the authors; licensee Growing Science, Canada

1. Introduction

Due to the increasing globalization of production networks, developments aimed at the cost-effective operation of production systems are becoming more and more important for companies to increase their competitiveness. In addition, companies aim to satisfy the customers' requirements at the highest possible level while meeting the rigorous order deadlines. The large number and diverse composition of production orders require an increasing degree of flexibility in production systems considering all levels of flexibility. Shivanand (2006) and Kumar et al. (2017) presented these levels of flexibility in detail. Flexible manufacturing systems (FMSs) integrate manufacturing and logistics resources (e.g. machining cells, workstations, automated material handling and storage equipment). The coordinated and efficient operation of FMSs is ensured by computer control. Michalos et al. (2010) presented the effect of system flexibility and degree of automation on product type variants and series sizes through the example of an automotive assembly system.

Market trends also affect the operation of production systems. It is necessary to produce several types of products at the same time with small series sizes and high flexibility. By ensuring high flexibility, the degree of automation of the systems is reduced, manual processes come to the fore, and the structure of production systems is also transformed. The matrix layout of workstations is able to ensure a high degree of simultaneous production of different product types to meet customer needs. Between workstations, one-piece material flow is realized through dynamically changing material flow relations. In this case, the role of human resources (workers) is twofold in the production process: on the one hand, the worker ensures the movement of the workpiece among the workstations, and on the other hand, he or she performs the corresponding technological operation on the workpiece at the given workstation. Efficient operation of production systems requires the use of intelligent production

* Corresponding author Tel.: +36709412685
E-mail: itkgv@uni-miskolc.hu (G. Kulcsár)

planning, scheduling, and control systems (Zijm, 2000), which are greatly supported by Industry 4.0 developments and solutions (Gania et al., 2018).

In many discrete manufacturing processes, the type and number of jobs to be performed may vary. Jobs involve technological operations that must be performed on certain machines or workstations according to the corresponding process plans. The order of operations of a given job is constrained by precedence relations. Fundamental manufacturing schemes include the flow shop, job shop, open shop, mixed shop and general shop models, whose classification, characteristics and scheduling algorithms are detailed by Brucker (2007). Regarding the complexity of job shop scheduling problems, the vast majority of models belong to the NP-hard problem class (Sotskov and Shakhlevich, 1995; Zinder et al., 2005; Brucker et al., 2007) and so it is not possible to create an optimal schedule with polynomial runtime. Considering the logistical resources of real production systems significantly increases the complexity of production scheduling models. Zhang et al. (2013) extended the job shop model with delivery tasks, Liu et al. (2017) considered intermediate buffers, and Chawla et al. (2018) inserted automated guided vehicles (AGVs) into the job shop scheduling problems. These extended problems were solved with heuristic algorithms (e.g. modified shifting bottleneck, PSO with MA, etc.).

In addition, human resources also appear to be an important influencing factor in examining the efficiency of the production system. For this reason, the impact of human resources on a flexible production system has also become an important area of research. Karwowski (2005) provided a theoretical perspective on human factors and ergonomics (HFE), defined as a unique and independent discipline that focuses on the nature of human-artefact interactions. Chung (1996) examined the effects of human factors in order to introduce an advanced manufacturing technology. Oliveira et al. (2019) presented the impact of Industry 4.0 on workers through the results of two European research projects. Peruzzini and Pellicciari (2017) focused on research and design of human-centred adaptive manufacturing systems (AMS) for modern companies, where aging workers are becoming more common. The purpose of AMS is that the system can adapt to the aging workers' needs considering their reduced capability for work. Longo et al. (2017) demonstrated the application possibilities of Industry 4.0 technologies that help workers perform their daily tasks. Despite technological advances, digitization, and increasing levels of automation, the human workforce remains an integral part of manufacturing and assembly systems. As the efficiency of production processes is also influenced by workers, the human factor must also be taken into account in planning, scheduling and control tasks.

Several studies deal with the examination of the production performance of workers. Ferjani et al. (2017) developed a heuristic algorithm for associating workers and jobs with different abilities. In their proposed model, the processing times of operations depend on the degree of worker fatigue. The paper demonstrated the efficiency of the proposed method through the results generated by a simulation model. The new method was also compared with other methods. Sirovetnukul and Chutima (2010) created a model for a U-shaped assembly cell in which workers perform one-piece material flow and the walking times are not negligible. Their goal was to - the number of workers. Sungur and Yavuz (2015) presented a new assembly line balancing problem, in which the workers have different qualifications. The goal is to find the optimal assignment of jobs and workers in case of each workstation in order to achieve the minimal cost. Durmaz and Koyuncu (2019), in their research, aimed to minimize the production cost of a production line. In the proposed model, the abilities of workers were taken into account when assigning jobs and workers. The results of their mathematical model showed a reduction in manufacturing costs when assigning the most appropriate worker to the job. In many cases, discrete manufacturing-assembly systems are based on job shop models and they employ human workers with different skills. The near-optimal assignment of jobs and workers contributes greatly to the efficient operation of the entire system. For this reason, job shop type production planning and scheduling models and methods need to manage human workers who have different skills in an integrated way. Al-Hinai and Piya (2015) presented, in a job shop environment, the effect of worker-dependent processing times on lead time. The goal was to minimize the lead time. Lin et al. (2016) presented a job shop scheduling model augmented with workers, in which processing time varies depending on the ability and experience of the workers.

The objective function was the maximum completion time, which was minimized with an improved simulated annealing genetic algorithm (ISAGA). Workers' skills are the basis for assigning jobs and workers. Lin and Gen (2008) presented a mathematical model in which the assignment of workers and jobs is solved regarding maximizing benefits and minimizing costs. A multi-objective hybrid genetic algorithm (mohGA) is proposed to solve the problem. Bányai et al. (2018) described the optimization problem of designing the number and structure of human resources. They use a simulation model based on Markov chains for optimizing the composition of worker groups. To solve the job shop (JSP) and flexible job shop (FJSP) scheduling problems, several studies have been performed in which heuristic and meta-heuristic search algorithms have been applied.

The most common methods include the genetic algorithm (e.g. Sun et al., 2010; Ida and Oka, 2011; Amjad et al., 2018; Fu and Liu, 2019; Wang et al., 2019; Yang et al., 2019), the simulated annealing (e.g., Yazdani et al., 2009; Zhang, 2013), and the tabu search (e.g., Bożejko et al., 2010; Jia and Hu, 2014; Ziaee, 2014). In addition, many other evolutionary meta-heuristics are used to solve the problem. Some examples are the particle swarm (PSO) algorithm (Pongchairerks and Kachitvichyanukul, 2009), ant colony (ACO) algorithm (Wang et al., 2017), bee colony (BCA) algorithm (Gao et al., 2016), bacteria (BA) algorithm (Luh and Lee, 2006) and bat (BAT) algorithm (Zaher et al., 2017). Furthermore, the application of other algorithms

can also be found: firefly (FFA) algorithm (Udaiyakumar and Chandrasekaran, 2014), black hole (BHO) algorithm (Jeet et al., 2016), flower pollination (FPA) algorithm (Yang et al., 2013) and harmony search (HS) algorithm (Gao et al., 2015).

Ruiz and Vázquez-Rodríguez (2010) presented a study that summarizes hybrid flow shop (HFS) problem classes, their objective functions and solving algorithms. The study shows that in most cases the goal was to minimize the maximum completion time (C_{max}). For small problems, the branch and bound (B&B) algorithm can be applied. However, the application of heuristic methods came to the fore to the greatest extent. Based on these experiences, it can be assumed that heuristic algorithms can also be used effectively to solve FJSP problems. Recently, due to the complexity of these problems, discrete event-driven simulation models have been used more and more frequently. These simulation frameworks are increasingly supporting the planning of manufacturing processes. For example, Zupan and Herakovic (2015) dealt with production line balancing. Kikolski (2016) performed efficiency improvements and optimization of a manufacturing system by eliminating the negative effects of bottlenecks. Barrera-Diaz et al. (2018) developed a simulation-based solution for an automotive company by developing an automatic data-handling system. Kundu et al. (2019) used a simulation model and PSO algorithm for supporting the KANBAN based material supply system of an assembly line.

In this paper, we summarize the results of our research aimed at optimizing the operation of flexible production systems in which human workers are integrated in addition to machine resources. The goal is to assign jobs, workers, and workstations in a way that maximizes the produced quantity and the free capacity of the system over time. The developed heuristic solving method has been implemented in discrete event driven (DES) software.

2. Problem Description

2.1 Main Characteristics of the Examined Manufacturing Processes

The motivation for our research was the production planning and scheduling problems that arise in a real production system. One specific industrial case study made it clear that there are scheduling problems in which not only the primary resources (machines, workstations) but also the secondary resources (workers) need to be considered in order to create effective production plans and to control the manufacturing processes. The examined production system can produce specific types of products. In order to create a product, a given sequence of operations has to be carried out according to a pre-developed technological plan. In the production system, different workstations are located in specific positions. The routes, the distances and the permissible traveling speeds are known. A given product type can be made according to a given technological plan that specifies the operations to be performed. Each product type has a given technological plan. The starting workpiece is selected by a worker (human resource), who visits the suitable workstations in the order fixed in the technological plan, and the worker performs the planned operations on the workpiece at the workstation. Finally, the worker places the finished product into the output storage. The worker selects the next workpiece from the input storage and performs the planned operations. In the production system, several workers perform their tasks at the same time.

The set of suitable workstations is given for each operation. The resources are limited in the time allowed to perform operations. The production management defines the actual shift calendar. It is a series of time intervals (shifts) that prescribes the resource availability as time constraints. Workstations and workers work according to the given shift calendar. The scope of scheduling decisions does not extend to changing the shifts, because the active shifts for the scheduling period are given. A factor that further increases the complexity of the constraints is that specific workers are assigned to each shift. Only one operation can be performed at a time at a given station. Only one operation can be performed on a workpiece at a time. A given worker can only work on one workpiece at a time. The standard processing times of operations are known for each product type and each workstation. The workers' working abilities are different. Our model includes special parameters to consider human skill factors. This means that the processing time of a given operation depends on the worker's abilities. The decision-making scope of the production planning and scheduling system extends to planning the type and number of products to be manufactured in the shifts, and to control the work of the workers during the shift taking into account the strict restrictions. The control module selects which of the alternative options the worker chooses in the decision situations. This control task includes selecting the product type at the beginning of the cycle and selecting the specific workstation for each operation.

2.2 Requirements and Purpose

The essence of the solution of the outlined resource-constrained multi-level scheduling problem is that a detailed production schedule must be created for the given time horizon considering the actual input data. The primary goal is that the schedule fulfils the quantity of products required by production management, and the secondary goal is that the utilization of resources must be maximized while respecting the strict constraints of the applied manufacturing technology and resources. This scheduling problem does not directly correspond to any of the known types of scheduling problems, mainly due to the specifics of the regulations for workers. Therefore, a new integrated problem type of production planning, scheduling and control must be solved. Hereinafter, we use the term Extended Flexible Job Shop with Human Resources (EFJSHR) to denote this problem

type. At the company under study, the problem was handled by human decision-making. Production planning engineers made the necessary decisions and prepared the production plan that specifies the quantities of products to be manufactured of each product type in the given shifts. The control of the processes within the shift was performed by the workers themselves. Workers could choose the next type of product to be manufactured themselves and also selected the specific workstation from the set of suitable workstations for each operation to be performed. The main purpose of our research was to replace this decision-making based on human experience with a software solution in order to achieve more efficient production.

3. Modelling and Solving the EFJSHR Problem

3.1 An Approach to Solve the Problem

The main goal of the research was to develop a detailed model and solving algorithms for the EFJSHR problem in order to implement a production planning and scheduling module in a simulation system that can create near-optimal detailed plans to meet current production needs and that can guide the workers' activities in the strict resource-constrained production system. The EFJSHR problem is very complex, so we have created a two-level decision hierarchy from the two interrelated scheduling problems. Each level uses its own special optimization model:

- At the upper level, the production planning problem is handled by a special heuristic search algorithm.
- At the lower level, the workers' activities are managed by special multi-criteria reactive scheduling algorithms.

Each solution variant created at the upper production planning level delivers input data and constraints to the lower level. The set of constraints originated from the upper level is supplemented by the lower level of the hierarchy with its own constraints of the worker control problem (Fig.1).

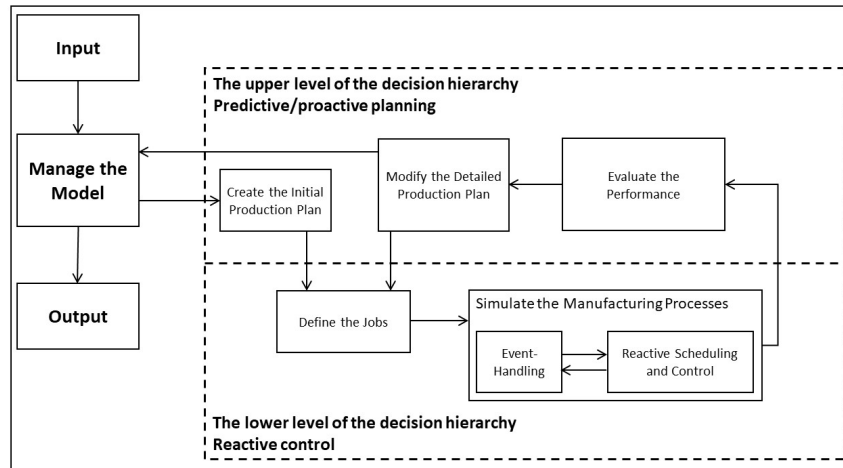


Fig. 1. Simplified flow chart of the solving method

In the following subsections, a new model and new algorithms are presented to realize this two-level integrated method. We first define a formal optimization model for the entire problem, then we describe the algorithms for solving the subproblems of production planning and worker control.

3.2 Problem Formulation

To solve the EFJSHR problem, the workpieces, shifts, workers and workstations have to be assigned to each other and the timely execution of operations must be controlled in such a way that the goals are met and the strict constraints are satisfied. Basic elements, notations and relationships of the formulation will be described in the following order:

- Input data;
- Primary decision variables;
- Values calculated from primary decision variables;
- Secondary decision variables;
- Values calculated from secondary decision variables;
- Constraints and
- Objective functions.

Input data

P_p	product type can be produced ($p = 1, 2, \dots, N_p$)
Q_p	quantity to be produced from the product type P_p ($p = 1, 2, \dots, N_p$)
\mathcal{O}_l	operation-type ($l = 1, 2, \dots, N_{\mathcal{O}}$)
$\mathcal{O}_{p,k}$	the k^{th} item in the sequence of operation-types to be performed on product type P_p ($k = 1, 2, \dots, N_{\mathcal{O}p}$)
T_s	the s^{th} prescribed resource availability time frame (shift) on the planning time horizon ($s = 1, 2, \dots, N_s$)
H_h	worker (human resource) ($h = 1, 2, \dots, N_H$)
$\omega_{p,h}$	the ability of worker H_h to perform operation-type $\mathcal{O}_{p,k}$ of product type P_p during the standard processing time (extent of compliance with the standard time)
$H(p)$	the set of workers who can manufacture product type P_p in the current shift ($H(p) \subseteq \{H_h \mid h = 1, 2, \dots, N_H\}$)
M_m	workstation (machine) ($m = 1, 2, \dots, N_M$)
$M(p,k)$	the set of workstations that can execute operation-type $\mathcal{O}_{p,k}$ in the current shift ($M(p,k) \subseteq \{M_m \mid m = 1, 2, \dots, N_M\}$)
$v_{p,k,m}$	the standard processing time of operation-type $\mathcal{O}_{p,k}$ executed at workstation M_m
$t_{p,k,m,h}$	the processing time of operation-type $\mathcal{O}_{p,k}$ executed at workstation M_m by worker H_h
$\tau_{m,r,h}$	the time need for worker H_h to move from workstation M_m to workstation M_r

Primary decision variables

Primary decision variables determine the detailed production plan.

$q_{p,s}$	the planned number of workpieces to be produced from product type P_p in shift T_s ($p = 1, 2, \dots, N_p$), ($s = 1, 2, \dots, N_s$)
-----------	---

Values calculated from primary decision variables

The actual values of the primary decision variables clearly determine the number and type of workpieces to be manufactured in the given shifts. The job is defined as manufacturing a given workpiece.

J_i	the i^{th} item in the full set of jobs, $\{J_i \mid i = 1, 2, \dots, N_J\}$
$J_{s,j}$	the j^{th} item in the set of jobs assigned to the shifts T_s , ($s = 1, 2, \dots, N_s$), ($j = 1, 2, \dots, N_{s,j}$)
$i(J_{s,j})$	the global index of job $J_{s,j}$
$p(i)$	the product type of job J_i
$O_{i,k}$	the k^{th} item in the technological sequence of operations of job J_i , ($k = 1, 2, \dots, N_{\mathcal{O}p(i)}$)
$t_{p(i),k,m,h}$	the processing time of operation $O_{i,k}$ executed at workstation M_m by worker H_h
$t_{p(i),k,m,h} = v_{p,k,m} + v_{p,k,m} \left(1 - \frac{\omega_{p,h}}{100}\right)$	(1)
r_i	the release time of job J_i
d_i	the due date of job J_i

Secondary decision variables

Secondary decision variables determine the detailed schedule of the execution of jobs. We use the following four types of secondary decision variables:

$\lambda_{i,h}$	a given item (decision variable) of a two-dimensional matrix to assign jobs $\{J_i \mid i = 1, 2, \dots, N_J\}$ to workers $\{H_h \mid h = 1, 2, \dots, N_H\}$:
$\lambda_{i,h} := \begin{cases} 1, & \text{if worker } H_h \text{ perform job } J_i \\ 0, & \text{otherwise} \end{cases}$	(2)

$x_{i,k,m}$	a given item (decision variable) of a three-dimensional matrix to assign operations $\{O_{i,k} \mid i = 1, 2, \dots, N_J, k = 1, 2, \dots, N_{\mathcal{O}p(i)}\}$ to workstations $\{M_m \mid m = 1, 2, \dots, N_M\}$:
$x_{i,k,m} := \begin{cases} 1, & \text{if operation } O_{i,k} \text{ is performed at workstation } M_m \\ 0, & \text{otherwise} \end{cases}$	(3)

$y_{i,k,u,v}$	a given item (decision variable) of a four-dimensional matrix to determine the execution sequence of operations $\{O_{i,k} \mid i = 1, 2, \dots, N_J, k = 1, 2, \dots, N_{\mathcal{O}p(i)}\}$:
---------------	---

$$y_{i,k,u,v} := \begin{cases} 1, & \text{if operation } O_{i,k} \text{ precedes operation } O_{u,v} \text{ at workstation } M_m \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

$S_{i,k,m}$ the starting time of operation $O_{i,k}$ at workstation M_m

Values calculated from secondary decision variables

$$\begin{aligned} \rho(O_{i,k}) & \text{ the worker who performs operation } O_{i,k} \\ \delta(O_{i,k}) & \text{ the workstation at which operation } O_{i,k} \text{ is performed} \\ C_{i,k,m} & \text{ the completion time of operation } O_{i,k} \text{ at workstation } M_m \\ C_i & \text{ the completion time of job } J_i \text{ (} i = 1, 2, \dots, N_j \text{)} \\ \pi_i & \text{ the function that indicates the completion of job } J_i \\ \pi_i & := \begin{cases} 1, & \text{if } 0 < C_i \text{ and } C_i \leq d_i \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (5)$$

The value $\pi_i = 1$ indicates that job J_i has already started ($0 < C_i$) and its completion time does not exceed the prescribed due date d_i . Due date d_i equals the finishing time of the shift that is assigned to the job J_i .

Constraints

We use the following five groups of constraints:

1. Restrictions on the quantities of products to be manufactured:

- Negative quantity cannot be produced:

$$q_{p,s} \geq 0 \quad (p = 1, 2, \dots, N_p), \quad (s = 1, 2, \dots, N_s) \quad (6)$$

- The total quantity to be produced has to be distributed among all shifts in the case of each product type:

$$\sum_{s=1}^{N_s} (q_{p,s}) = Q_p \quad (p = 1, 2, \dots, N_p) \quad (7)$$

2. Restrictions on selected resources:

- A given job can only be assigned to a suitable worker:

$$\sum_{H_h \notin H(p(i))} \lambda_{i,h} = 0 \quad (i = 1, 2, \dots, N_j) \quad (8)$$

- A given job can be assigned to exactly one suitable worker:

$$\sum_{H_h \in H(p(i))} \lambda_{i,h} = 1 \quad (i = 1, 2, \dots, N_j) \quad (9)$$

- A given operation can only be assigned to a suitable workstation:

$$\sum_{M_m \notin M((p(i),k))} x_{i,k,m} = 0 \quad (i = 1, 2, \dots, N_j), \quad (k = 1, 2, \dots, N_{\theta p(i)}) \quad (10)$$

- A given operation can be assigned to exactly one suitable workstation:

$$\sum_{M_m \in M((p(i),k))} x_{i,k,m} = 1 \quad (i = 1, 2, \dots, N_j), \quad (k = 1, 2, \dots, N_{\theta p(i)}) \quad (11)$$

3. Restrictions on worker activities:

- A given worker can perform only one job at a time. A given worker can start the next job only if he or she has completed the previous one:

$$C_{u,v,r} < S_{i,k,m} \text{ where } u \neq i \wedge \rho(O_{u,1}) = \rho(O_{i,1}) \wedge \delta(O_{u,v}) = r \wedge \delta(O_{i,k}) = m \wedge C_{u,1,r} < S_{i,1,m} \quad (12)$$

- A given worker can perform only one operation at a time and the time for moving between workstations cannot be neglected:

$$C_{i,k,\delta(O_{i,k})} + \tau_{\delta(O_{i,k}),\delta(O_{i,k+1}),\rho(O_{i,k})} \leq S_{i,k+1,\delta(O_{i,k+1})} \text{ where } (i = 1, 2, \dots, N_j), \quad (k = 1, 2, \dots, N_{\theta p(i)} - 1) \quad (13)$$

4. Restrictions on the workload of workstations:

- Only one operation can be performed at a given workstation at a time:

$$C_{i,k,m} \leq S_{u,v,m} \text{ where } \delta(O_{i,k}) = \delta(O_{u,v}) \wedge y_{i,k,u,v,m} = 1 \quad (14)$$

- The processing time of a given operation depends on the workstation and the worker:

$$S_{i,k,\delta}(O_{i,k}) + t_{p(i),k,\delta}(O_{i,k},\rho(O_{i,k})) = C_{i,k,\delta}(O_{i,k}) \text{ where } (i = 1, 2, \dots, N_j), (k = 1, 2, \dots, N_{\theta p(i)}) \quad (15)$$

5. Restrictions on the execution of jobs in the given shifts:

- The execution of job J_i cannot start earlier than its release time (the beginning of the assigned shift):

$$r_i \leq S_{i,1,\delta}(O_{i,1}) \text{ where } (i = 1, 2, \dots, N_j) \quad (16)$$

- The operations of job J_i can only be performed in the assigned shifts:

$$C_{i,k,\delta}(O_{i,k}) \leq d_i \text{ where } (i = 1, 2, \dots, N_j), (k = 1, 2, \dots, N_{\theta p(i)}) \quad (17)$$

Objective functions

The optimization model is defined as finding the best solution by considering two objective functions at the same time. The objective functions are the following:

1. Primary objective function: Quantity of manufactured products (the number of completed jobs) [pcs]:

$$f_1 := \sum_{i=1}^{N_j} (\pi_i) \rightarrow \max. \quad (18)$$

2. Secondary objective function: Total remaining time [min]:

$$f_2 := \sum_{s=1}^{N_s} \left(\min_{j=1}^{N_s, j} \left(d_{i(J_{s,j})} - C_{i(J_{s,j})} \right) \right) \rightarrow \max. \quad (19)$$

The value of objective function f_2 expresses the free capacity of the production system over time. This is a new performance indicator that measures the length of the last section at the end of the shift when no worker is active. We refer to this time value by the term “remaining time”. The remaining time of a given shift is calculated by subtracting the completion time of the last workpiece from the finishing time of the given shift. The total remaining time of the whole system is obtained by summing the remaining times of the shifts.

3.3 A New Method to Solve the EFJSHR Problem

3.3.1 Advanced Simulation-Based Searching Algorithm to Generate Detailed Production Plans

We have developed a heuristic method to determine the current values of primary and secondary decision variables of the EFJSHR problem formalized in the previous subsection. Our method is called Simulation-Based Production Planning and Control (SBPPC). The core of SBPPC is based on the integrated usage of an iterative improvement searching algorithm and two reactive rule-based control algorithms. These components are embedded in a discrete event-driven simulation framework. The simulation is able to calculate the current values of the objective functions based on the values of the decision variables. In a preparatory phase, an initializing method loads the input data, builds the current simulation model and initializes all the variables and parameters of the system. The pseudocode of SBPPC is presented in Fig. 2.

Method SBPPC to solve the problem EFJSHR	
1:	{
2:	Load the input data;
3:	Set the maximum number ε_{max} of iterations;
4:	Create the initial production plan by using algorithm IPP to calculate the values of primary decision
5:	variables $q_{p,s}$;
6:	$\varepsilon \leftarrow 1$;
7:	while ($\varepsilon \leq \varepsilon_{max}$)
8:	{
9:	Simulate the production plan with reactive control of manufacturing processes;
10:	Calculate the values of objective functions f_1 and f_2 ;
11:	if ($f_1 < N_j$ and $f_2 > 0$)
12:	{
13:	Modify the production plan by using the algorithm MPP to recalculate the values of primary decision
14:	variables $q_{p,s}$;
15:	}
16:	else
17:	{
18:	break;
19:	}
20:	$\varepsilon \leftarrow \varepsilon + 1$;
21:	}
22:	Return the results;
23:	}

Fig. 2. Pseudocode for the SBPPC method

The SBPPC method uses the IPP algorithm to create the initial production plan. This algorithm assigns the jobs to the shifts to create quasi-uniform loads.

Algorithm IPP

```

1: {
2:   p = 1;
3:   while (p ≤ Np)
4:     {
5:       s ← 1;
6:       while (s ≤ Qp mod Ns)
7:         {
8:           qp,s ← (Qp div Ns) + 1;
9:           s ← s + 1;
10:        }
11:      while (s ≤ Ns)
12:        {
13:          qp,s ← (Qp div Ns)
14:          s ← s + 1;
15:        }
16:      p ← p + 1;
17:    }
18:  Return the primary decision variables qp,s;
19: }
```

Fig. 3. Pseudocode for the IPP algorithm

In the definition of the IPP algorithm presented in Fig. 3. , the operator *div* returns the integer value of the result of the division operation. The operator *mod* means the modulo division that returns the remainder of the division operation. The SBPPC method uses the MPP algorithm (Fig. 4) to improve the production plan. This algorithm modifies the assignment of jobs and shifts by moving unfinished jobs into suitable shifts that can perform the selected jobs.

Algorithm MPP

```

1: {
2:   Φ ← 1;
3:   while (Φ ≤ Ns)
4:     {
5:       Create an empty list W;
6:       Select the index i of each job Ji assigned to shift TΦ where πi = 0;
7:       Add the selected job indices to list W such that symbol Ww represents the w-th item of the list and symbol
8:       NW equals to the number of items stored in list W;
9:       w ← 1;
10:      while (w ≤ NW)
11:        {
12:          φ ← Search the earliest suitable shift by ESS(Ww, Φ + 1, Ns);
13:          if (φ ≠ 0)
14:            {
15:              qp(Ww),φ ← qp(Ww),Φ - 1;
16:              qp(Ww),φ ← qp(Ww),φ + 1;
17:            }
18:          else if (2 < Φ)
19:            {
20:              φ ← Search the earliest suitable shift by ESS(Ww, 1, Φ - 1);
21:              if (φ ≠ 0)
22:                {
23:                  qp(Ww),φ ← qp(Ww),Φ - 1;
24:                  qp(Ww),φ ← qp(Ww),φ + 1;
25:                }
26:            }
27:          w ← w + 1;
28:        }
29:      Φ ← Φ + 1;
30:    }
31:  Return the primary decision variables qp,s;
32: }
```

Fig. 4. Pseudocode for the MPP algorithm

In the presented formal definition of the MPP algorithm, the instruction $q_{p(W_w),\Phi} = q_{p(W_w),\Phi} - 1$ expresses that the selected job J_{W_w} is taken out of the set of jobs assigned to shift T_Φ . The instruction $q_{p(W_w),\Phi} = q_{p(W_w),\Phi} + 1$ shows that the selected job J_{W_w} is added to the set of jobs assigned to shift T_Φ . Function $ESS(i, lb, ub)$ is used to find the earliest shift where the selected unfinished job J_i can be performed. The searching range is bounded with two parameters. The parameter lb defines the lower bound. This expresses the smallest index from which to start the searching method. The parameter ub specifies the largest allowed index of the searching range as upper bound. This function examines the searching range of shifts T_κ ($lb \leq \kappa \leq ub$), and it returns the index κ of the earliest shift T_κ that has remaining time and includes at least one worker H_h who can produce the product type $P_{p(i)}$ of the given job J_i . If there is no suitable shift then the function returns zero. The pseudocode of ESS is presented in Fig. 5.

Function ESS(i, lb, ub)	
1:	{
2:	$s \leftarrow 0;$
3:	$\kappa \leftarrow lb;$
4:	while ($\kappa \leq ub$)
5:	{
6:	if $\min_{j=1}^{N_{\kappa,j}} (d_{i(J_{\kappa,j})} - C_{i(J_{\kappa,j})}) > 0$ and $H(p(i)) \neq \emptyset$
7:	{
8:	$s \leftarrow \kappa;$
9:	Return $s;$
10:	}
11:	$\kappa \leftarrow \kappa + 1;$
12:	}
13:	Return $s;$
14:	}

Fig. 5. Pseudocode for Function ESS

As shown earlier, the solution of the entire optimization problem can be divided into two main levels. The upper level focuses on optimizing the production plan. This part has been presented so far. In the following, we propose new control algorithms based on reactive scheduling to perform the given production plan.

The SBPPC Method includes an event-driven simulation method called “Simulate the production plan with reactive control of manufacturing processes”. At the beginning of the simulation, the framework initializes the virtual manufacturing system, actualizes the production plan and sets the current values for the control parameters. The essence of the simulation is that different events appear at specific times and they are processed by event-handler algorithms. In each decision-making situation, in which there are several options for handling the event, reactive control algorithms make the decision. The most important types of decision-making situations are the following:

- When a worker arrives at the input container, the next job to be performed must be selected.
- When a worker has performed a given operation, the workstation must be selected at which the next operation will be executed.

To effectively solve these decision problems, we have developed new control algorithms. These are used in the simulation to determine the values of secondary decision variables.

3.3.2 Control Algorithm to Assign Jobs and Workers (AJW)

To solve this subproblem, we use a reactive method based on a weighted multi-criteria evaluation function. When worker H_h arrives at the input container, the control system computes the value of function g for each selectable job, and the job with the largest value of function g is selected to be next job ($g(J_i) \rightarrow \max$). The selected job is assigned to worker H_h . This can be expressed with the following secondary decision variable: $\lambda_{g(J_i) \rightarrow \max, h} := 1$. The definition of function g is the following:

$$g(J_i) := \beta_1 b_1(J_i) + \beta_2 b_2(J_i) + \beta_3 b_3(J_i) + \beta_4 b_4(J_i) + \beta_5 b_5(J_i) \quad (20)$$

where:

- $b_1(J_i)$ (worker’s skill): the capability $\omega_{p(i),h}$ of worker H_h to perform job J_i in percentage [%];
- $b_2(J_i)$ (manufacturing flexibility): the ratio of the number of workers able to perform job J_i and the total number of workers in shift T_s [%];
- $b_3(J_i)$ (waiting jobs): the ratio of the current number of jobs still to be performed from product type $P_{p(i)}$ and the planned number $q_{p(i),s}$ of jobs assigned to shift T_s according to the same product type $P_{p(i)}$ [%];

- $b_4(J_i)$ (inhomogeneity): This aspect expresses that we strive to choose the product type that has the smallest number of currently running jobs in the manufacturing system. Simultaneous production of different product types can be ensured by increasing the degree of heterogeneity of the product types. In order to prepare the definition of $b_4(J_i)$, we introduce the following notations:
 - $L(P_p)$: the number of currently running jobs of the product type P_p ,
 - $R(P_p)$: the number of starting jobs of product type P_p :
$$R(P_p) := \begin{cases} 1, & \text{if the product type } P_p \text{ is selected to start} \\ 0, & \text{otherwise} \end{cases} \quad (21)$$

In each decision-making situation, only one job can be selected to start, so the following relation is obtained:

$$\sum_{\mu=1}^{N_p} R(P_\mu) = 1 \quad (22)$$

Using these formulas, we define the probability of product type P_p as follows:

$$\xi(P_p) = \frac{L(P_p) + R(P_p)}{\sum_{\mu=1}^{N_p} L(P_\mu) + 1} \quad (23)$$

By adapting the concept of entropy used in the information theory (Shannon, 1948), we define the entropy of any product type P_p as follows:

$$\eta(P_p) = -\xi(P_p) \log_2 \xi(P_p) \quad (24)$$

If job J_i of product type $P_{p(i)}$ is selected to start, the entropy of the system can be calculated with the following formula:

$$E(P_{p(i)}) = \sum_{p=1}^{N_p} \eta(P_p) = -\sum_{p=1}^{N_p} \xi(P_p) \log_2 \xi(P_p) \quad (25)$$

The entropy of the system would be maximal if all the product types were in production with equal number of jobs at the same time. In such an ideal case, the probability ξ of each product type P_p is the following:

$$\xi(P_p) = \frac{1}{N_p} \quad (26)$$

Consequently, the maximal entropy of the system is the following:

$$E_{max} = -\sum_{p=1}^{N_p} \xi(P_p) \log_2 \xi(P_p) = \log_2 N_p \quad (27)$$

The value of factor $b_4(J_i)$ is defined in percentage as follows:

$$b_4(J_i) := \frac{E(P_{p(i)})}{E_{max}} \quad (28)$$

From the point of view of inhomogeneity, we would like to select to start the job J_i where the value $b_4(J_i)$ is maximal.

- $b_5(J_i)$ (average lead time): the ratio of the average lead time of product type $P_{p(i)}$ of job J_i and the average lead time of all product types of shift T_s [%]. Note that the values of all factors b are expressed as a percentage.
- $\beta_1, \beta_2, \beta_3, \beta_4, \beta_5$ integer weights from the closed interval $[-5, 5]$. Adjusting the weight of a given aspect determines the degree of its consideration.

3.3.3 Control Algorithm to Assign One of the Suitable Workstations to the next Operation (AWO)

To execute a given operation $O_{i,k}$ of job J_i , we have a set of suitable workstations:

$$M(p(i), k) \subseteq \{M_m \mid m = 1, 2, \dots, N_M\} \quad (29)$$

Worker H_h should choose one of the suitable workstations. In order to solve this subproblem, our control algorithm computes the function z for each suitable workstation and the workstation with the smallest value of function z is selected to execute the operation ($z(M_m) \rightarrow \min$). The selected workstation is assigned to the given operation. This can be expressed with the following secondary decision variable: $x_{i,k,z(M_m) \rightarrow \min} := 1$. The definition of function z is the following:

$$z(M_m) := \alpha_1 c_1(M_m) + \alpha_2 c_2(M_m) \quad (30)$$

where:

- $c_1(M_m)$: the number of currently waiting workers in front of the workstation M_m [pcs].
- $c_2(M_m)$: the number of operations completed at the workstation M_m in the current shift T_s [pcs].
- α_1, α_2 integer weights from the closed interval $[0, 100]$, Adjusting the weight of a given aspect determines the degree of its consideration.

The simulation extended with the above presented control algorithms clearly determine the execution of the jobs. During the simulation, all the secondary decision variables get their values according to reactive scheduling when handling the occurring events:

- Decision variables $\lambda_{i,h}$ obtain values when the control algorithm selects and assigns jobs to workers.
- Decision variables $x_{i,k,m}$ take on their values when the control algorithm selects and assigns workplaces to operations.
- Decision variables $y_{i,k,u,v,m}$ and $S_{i,k,m}$ are set when operations start at workstations.

3.3.4 Applicability of the Proposed Model and Algorithms

The presented model and algorithms can solve the integrated production planning, scheduling and control problem of flexible manufacturing systems in which human resources are working with different individual skills. The proposed algorithms can flexibly and efficiently handle the changing requirements, varying resource constraints, uncertainties, and unexpected events. In this approach, detailed production plans are generated by the SBPPC method that includes a special search algorithm and advanced event-driven simulation. Reactive control algorithms are embedded into the simulation to make the necessary decisions at the same time as the execution of manufacturing processes. The presented production planning phase and the manufacturing control phase together create a two-level decision hierarchy to manage the production activities. The elaborated optimization model of the investigated problem, the solving approach and algorithms can be used in practice. The effectiveness of the proposed methods will be demonstrated with some running results in the next section.

In cyber-physical environments, the real system and the simulation system can work together in real time, Taking advantage of this fact, the proposed algorithms can also be used to supplement such systems; therefore, the control algorithms can directly handle real events as well. Classical enterprise application systems (e.g. ERP, MES, APS, and SCADA) and modern technologies (e.g. Industry 4.0, Internet of Things, Cloud Computing and Big Data) are able to support the practical application of the proposed approach and algorithms by providing appropriate input data, computational capacities, and application environment.

4. Some Numerical Results

The presented algorithms have been implemented in Plant Simulation (Siemens, v15). The software ran on a personal computer equipped with an Intel® Core™ i7-4810MQ 2.80 GHz processor, with 8 GB of RAM memory. The operating system of the computer was Microsoft Windows 10 (64-bit).

4.1 Examining the AWO Control Algorithm

Simulation studies were performed, according to the parameter values given in Table 1 to Table 4, to adjust the values of α_1 and α_2 parameters. During the simulation studies performed in the test environment, the values of α parameters were independent of the values of β parameters. Assignments of jobs and workers were performed according to the First Suitable Job Selection (FSJS) strategy, which means that the worker selects the first suitable item from the queue of waiting jobs.

Table 1

The details of the illustrative test environment for adjusting α parameter values

Settings	Values
Number of shifts	5
Processing times of operations	Depending on workers' skills
Assignments of jobs and workers	First Suitable Job Selection method (FSJS strategy)
Groups of parallel workstations	Group_1: M ₃₆ , M ₁₆ , M ₅₁ Group_2: M ₉ , M ₁₉ , M ₇₁
Assignments of suitable workstations and operations	AWO control
Production planning method	IPP

Table 2
Illustrative test problem data for adjusting α parameters

Input data	Values
Production plans	Production plan A: $P_j = \text{Prod13}; Q_j = 10000$ Production plan B: $P_j = \text{Prod111}; Q_j = 2500;$ $P_2 = \text{Prod122}; Q_2 = 2500$
Workers in shifts	8 workers and 5 shifts: $T_1: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_2: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_3: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_4: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_5: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ 16 workers 5 shifts: $T_1: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15}, H_{16}$ $T_2: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15}, H_{16}$ $T_3: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15}, H_{16}$ $T_4: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15}, H_{16}$ $T_5: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9, H_{10}, H_{11}, H_{12}, H_{13}, H_{14}, H_{15}, H_{16}$

Table 3
Illustrative process plans for product types

Product type	Operations	Parallel workstations	Product type	Operations	Parallel workstations	Product type	Operations	Parallel workstations
Prod13	OP0	Start (Source)	Prod111	OP0	Start (Source)	Prod122	OP0	Start (Source)
Prod13	OP1	M1	Prod111	OP1	M1	Prod122	OP1	M1
Prod13	OP2	M47	Prod111	OP2	M30	Prod122	OP2	M39
Prod13	OP3	M39	Prod111	OP3	M29	Prod122	OP3	M47
Prod13	OP4	M48	Prod111	OP4	M25	Prod122	OP4	M43
Prod13	OP5	M44	Prod111	OP5	M15	Prod122	OP5	M44
Prod13	OP6	M49	Prod111	OP6	M40	Prod122	OP6	M45
Prod13	OP7	M50	Prod111	OP7	M41	Prod122	OP7	M46
Prod13	OP8	M16,M36,M51	Prod111	OP8	M35	Prod122	OP8	M16,M36,M51
Prod13	OP9	M7	Prod111	OP9	M42	Prod122	OP9	M7
Prod13	OP10	M8	Prod111	OP10	M16,M36,M51	Prod122	OP10	M8
Prod13	OP11	M9, M19, M71	Prod111	OP11	M7	Prod122	OP11	M9, M19, M71
Prod13	OP12	End (Drain)	Prod111	OP12	M8	Prod122	OP12	End (Drain)
			Prod111	OP13	M9, M19, M71			
			Prod111	OP14	End (Drain)			

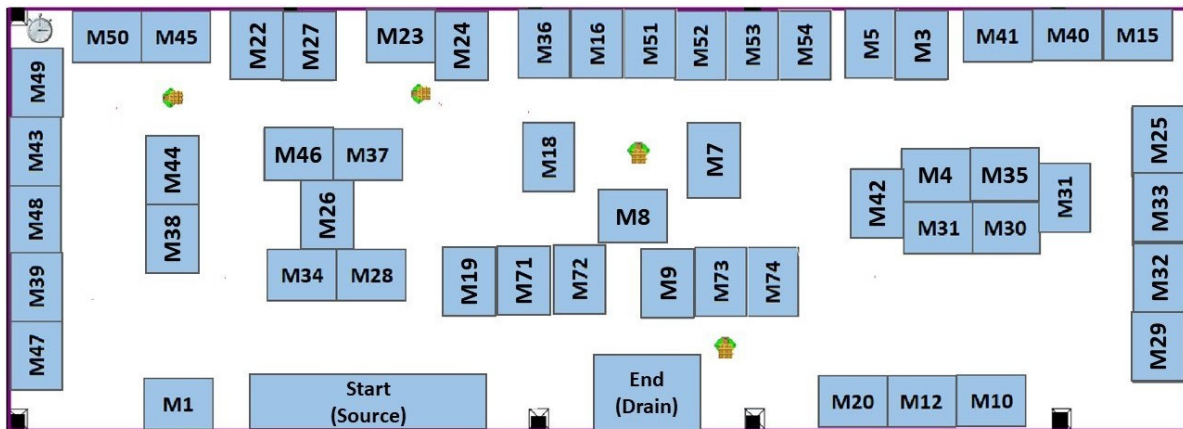


Fig. 6. Simplified layout of the investigated flexible manufacturing system with parallel workstations (M36, M16, M51) and (M9, M19, M71)

Different production plans (A and B) were performed with different teams of workers. Some illustrative examples can be found in Table 2. The goal was to define the best value pair for α_1 and α_2 parameters, where the quantity of the produced products is maximal and the waiting time of workers in front of workstations is minimal. In the simulation, the amount of waiting time of workers was quantified by comparing the sum of the waiting times to the sum of availability times, then the average waiting rate was expressed with the obtained quotient value as percentage. Parameters α_1 and α_2 were given all combinations of integer values from the range[0, 100]. Some illustrative running results of the simulation studies on the test problem data are shown in Table 5.

Table 4
Illustrative workers' skills ($\omega_{p,h}$) [%]

Product types	Workers															
	H ₁	H ₂	H ₃	H ₄	H ₅	H ₆	H ₇	H ₈	H ₉	H ₁₀	H ₁₁	H ₁₂	H ₁₃	H ₁₄	H ₁₅	H ₁₆
Prod13	108	93	114	94	91	84	100	101	108	103	87	51	83	89	92	0
Prod111	101	99	98	107	88	92	111	108	110	95	84	53	84	80	89	53
Prod122	105	102	99	103	86	82	99	111	105	109	86	0	86	90	86	0

Table 5
Some numerical results of simulation studies focused on α parameters

Parameters		Number of produced products [pcs]				Average waiting rate of workers in front of the workstations [%] (percentage of availability time)			
α_1	α_2	8 workers		16 workers		8 workers		16 workers	
		Production	Production	Production	Production	Production	Production	Production	Production
100	40	734	839	1061	1175	3.01	4.87	25.50	24.20
90	50	734	840	1061	1177	3.01	4.75	24.71	24.17
70	90	734	843	1061	1181	2.99	4.53	24.71	23.96
60	30	734	840	1061	1177	3.01	4.75	24.75	24.17
50	20	734	839	1061	1175	3.01	4.87	24.71	24.20
40	100	734	841	1058	1178	3.01	4.58	24.71	24.15
90	60	734	841	1061	1177	3.01	4.75	24.58	24.17
30	100	734	841	1058	1180	3.14	4.58	24.71	24.04
80	30	734	839	1061	1175	3.01	4.87	24.71	24.20
1	100	726	827	1046	1160	4.06	5.91	24.71	25.04

When searching the best values for α_1 and α_2 parameters, the primary consideration was the quantity of produced products and the secondary consideration was the average waiting rate of workers. The best result was obtained with the combination $\alpha_1 = 70$ and $\alpha_2 = 90$. This value pair was used in further examinations.

4.2 Examining the AJW Control Algorithm

Additional simulation studies were also performed to adjust the values of $\beta = \{\beta_1, \beta_2, \beta_3, \beta_4, \beta_5\}$ parameters. During these simulation studies, we used the best values of α parameters, Assignments of jobs and workers were performed according to the AJW control algorithm. By changing the β parameter values, the assignments of jobs and workers can be influenced, which affects the productivity of the production system. The most important settings of the test environment are summarized in Table 6.

Table 6
Settings of the test environment for adjusting β parameters

Settings	Values
Number of shifts	5
Processing times of operations	Depending on workers' skills
Assignments of jobs and workers	AJW control algorithm
	Group_1: M ₃₆ , M ₁₆ , M ₅₁ , M ₅₂ , M ₅₃ , M ₅₄
Groups of parallel workstations	Group_2: M ₉ , M ₁₉ , M ₇₁ , M ₇₂ , M ₇₃ , M ₅₄
Assignments of suitable workstations and operations	AWO control
Control parameters	$\alpha_1 = 70, \alpha_2 = 90$
Production plan	$P_1 = \text{Prod5}; Q_1 = 181; P_2 = \text{Prod4}; Q_2 = 231;$ $P_3 = \text{Prod9}; Q_3 = 122; P_4 = \text{Prod111}; Q_4 = 165;$ $P_5 = \text{Prod7}; Q_5 = 156; P_6 = \text{Prod6}; Q_6 = 88;$ $P_7 = \text{Prod8}; Q_7 = 101$
Production planning method	IPP
Workers in shifts	$T_1: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_2: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_3: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_4: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$ $T_5: H_1, H_2, H_4, H_5, H_6, H_7, H_8, H_9$

The input data and initial situation of the simulation studies were fixed and the control aspects of the AJW algorithm were considered with different weights. The β parameters were given all combinations of integer values from the range $[0, 5]$. As a result, the productivity of the production system changed. The goal was to find the best values of $\beta = \{\beta_1, \beta_2, \beta_3, \beta_4, \beta_5\}$ parameters where the quantity of produced products (f_1) and the sum (f_2) of remaining times of the system are maximal. Accordingly, we performed simulation studies with different β values and with the given settings of the test environment. Some typical results are summarized in Table 7.

Table 7
Running results of simulations performed with different β parameters

ExpNo.	β_1	β_2	β_3	β_4	β_5	Planned quantity	Produced quantity	Remaining time
						[pcs]	[pcs]	[%]
1	4	2	2	5	2	1044	993	9.10
2	1	4	2	5	5	1044	987	1.98
3	3	4	1	3	4	1044	990	0.20
4	4	4	5	3	3	1044	992	2.48
5	2	3	4	3	5	1044	993	5.06
6	5	4	5	3	2	1044	993	10.32
7	1	3	4	2	4	1044	987	3.74
8	4	3	2	4	5	1044	992	6.14
9	5	4	2	2	1	1044	989	10.47
10	5	1	2	1	4	1044	988	6.58
11	2	4	2	3	5	1044	991	1.69
12	3	1	2	3	1	1044	992	10.75
13	4	1	1	4	4	1044	987	9.68
14	4	5	2	2	2	1044	991	11.03
15	2	5	1	4	2	1044	985	5.75
16	5	1	1	1	5	1044	982	9.83
17	2	4	3	5	2	1044	988	2.61

The best result was given by the following combination: $\beta_1 = 5, \beta_2 = 4, \beta_3 = 5, \beta_4 = 3, \beta_5 = 2$. The presented simulation examinations were able to find the values of the α and β parameters that ensure the most efficient operation of the control algorithms. In the following studies, their values do not change.

4.3 Examining the SBPPC Method

4.3.1 An Illustrative Example for Using the MPP Algorithm in the SBPPC Method

After finding the best values for parameters of the control algorithms, we examined the efficiency of the SBPPC method. Before describing the details, we present how our MPP-based search method works through an example. Table 6 summarizes the settings of the simulation model used in this example.

Table 8
Settings of simulation for testing SBPPC method

Settings	Values
Processing times of operations	Depending on workers' skills
Number of shifts	5
Algorithm and control parameters for assigning jobs to workers	AJW control $\beta_1=5, \beta_2=4, \beta_3=5, \beta_4=3, \beta_5=2$
Groups of parallel workstations	Group_1: M ₃₆ , M ₁₆ , M ₅₁ , M ₅₂ , M ₅₃ , M ₅₄ Group_2: M ₉ , M ₁₉ , M ₇₁ , M ₇₂ , M ₇₃ , M ₇₄
Algorithm and control parameter for assigning suitable workstations and operations	AWO control $\alpha_1=70, \alpha_2=90$
Active workers in shifts	T_1 : H ₈ , H ₉ , H ₁₅ , H ₁₉ T_2 : H ₁ , H ₂ , H ₃ , H ₇ , H ₁₃ T_3 : H ₄ , H ₅ , H ₆ , H ₁₄ , H ₁₈ T_4 : H ₁ , H ₂ , H ₃ , H ₇ , H ₁₃ T_5 : H ₁₀ , H ₁₁ , H ₁₂ , H ₁₆ , H ₂₀
Production requirements	$P_1=$ Prod5; $Q_1=$ 66; $P_2=$ Prod6; $Q_2=$ 81; $P_3=$ Prod4; $Q_3=$ 322; $P_4=$ Prod9; $Q_4=$ 48; $P_5=$ Prod11; $Q_5=$ 220; $P_6=$ Prod7; $Q_6=$ 73; $P_7=$ Prod10; $Q_7=$ 61; $P_8=$ Prod8; $Q_8=$ 110
Method to create initial production plans for shifts	IPP

The MPP algorithm, which modifies the detailed production plan for shifts, is an integral part of the SBPPC method. An illustrative simulation study was performed based on the settings summarized in Table 8. The results are shown in Table 9. The first column of the table contains the iteration number of SBPPC method. In the first iteration (#1), the IPP algorithm creates the initial solution. This is represented with the primary decision variables that determine the detailed production plans for shifts. The simulation, which uses the reactive control algorithms, shows that this initial solution leads to unfinished jobs while the manufacturing system has remaining time. In the second iteration (#2), MPP algorithm modifies the solution by moving unfinished jobs into suitable shifts, and then the simulation is run again with the new solution. This iterative process continues until the stopping conditions are met. The results of the simulation performed in each iteration are summarized in the rows of Table 9.

Table 9
The results of the MPP algorithm included in the SBPPC method

Iteration (#)	Number of finished jobs [pcs]	Number of unfinished jobs [pcs]	Sum [min]	Remaining time Shifts				
				T_1	T_2	T_3	T_4	T_5
1	957	24	115.41	0.00	21.37	45.11	26.12	22.82
2	973	8	75.40	5.24	21.37	0.00	23.09	25.70
3	976	5	52.56	5.24	21.37	0.00	25.95	0.00
4	980	1	40.76	5.24	7.25	1.32	26.94	0.00
5	981	0	42.43	5.24	7.25	1.32	21.17	7.45

The SBPPC method can achieve a significant improvement in the productivity of an EFJSHR model even with a small number of iterations by creating a quasi-optimal production plan for shifts. Table 9 also shows that all jobs are finished (all products are produced) finally in this example.

4.3.2 Comparison of MPP-based Search Method and Genetic Algorithm

The proposed MPP-based search method was compared with the genetic algorithm of the Plant Simulation software. In this investigation, we focused on the production planning method that assigned the jobs to shifts. This method calculated the values of the primary decision variables of the EFJSHR problem. These values formed the detailed production plan. The control of the execution of jobs within shifts was performed by the control algorithms with the previously adjusted parameters. When using our MPP-based heuristic search algorithm and the genetic algorithm, we used the same simulation method and environment. The only difference between the two approaches was the method of assigning jobs to shifts.

The genetic algorithm (PSGA) of Plant Simulation is not directly suitable for optimizing the assignments of jobs and shifts. To solve this problem, we developed a model transformation technique that can transform the detailed production plan into a permutation scheme. This can be mapped to an array that can be fully manageable with PSGA. Each candidate array is transformed back into a detailed production plan before simulation. Thus, the detailed production plan can be mapped to the permutation of given elements. Using this model transformation technique, the optimization problem of production planning (assigning jobs to shifts) corresponds exactly to a general combinatorial optimization problem in which the objective function value depends on the order of the given elements. This allows the PSGA to solve the problem. The representation scheme of detailed production plan was designed so that the length of the array is equal to the multiplication of the number of jobs and the number of shifts. The array contains the job identifiers and zero values. Each job can appear in the array exactly once. Zero values play a “position-filling” role. These are used to ensure that the length of the array (encoding scheme) is constant for a given problem, while all possible assignment options (permutations) can be represented. The position index of each element in the array clearly determines the target shift.

For example, we have 4 jobs to be performed: J_1, J_2, J_3, J_4 . There are 3 shifts: T_1, T_2, T_3 . In this illustrative case, the array (chromosome) contains $4 * 3 = 12$ elements (genes). In the chromosome, genes 1-4 represent shift 1, genes 5-8 represent shift 2, and genes 9-12 represent shift 3. In this problem, one possible chromosome, which describes a given solution (assignments of jobs and shifts), can be seen in Table 10.

Table 10
An illustrative chromosome for PSGA

Chromosome	Shifts											
	T_1				T_2				T_3			
chromosome 1	J_1	J_2	0	0	J_3	0	0	0	J_4	0	0	0

In addition, an example population of 5 individuals is shown in Table 11.

Table 11

An example population for PSGA

Chromosome	Shifts											
	T_1				T_2				T_3			
	J_1	J_2	0	0	J_3	0	0	0	J_4	0	0	0
chromosome 1	J_1	J_2	0	0	J_3	0	0	0	J_4	0	0	0
chromosome 2	0	J_3	0	J_2	0	0	0	0	J_1	J_4	0	0
chromosome 3	J_1	J_2	J_3	0	0	J_4	0	0	0	0	0	0
chromosome 4	0	0	0	0	J_3	0	J_4	0	J_1	J_2	0	0
chromosome 5	J_4	0	0	0	J_3	0	0	J_1	0	0	0	J_2

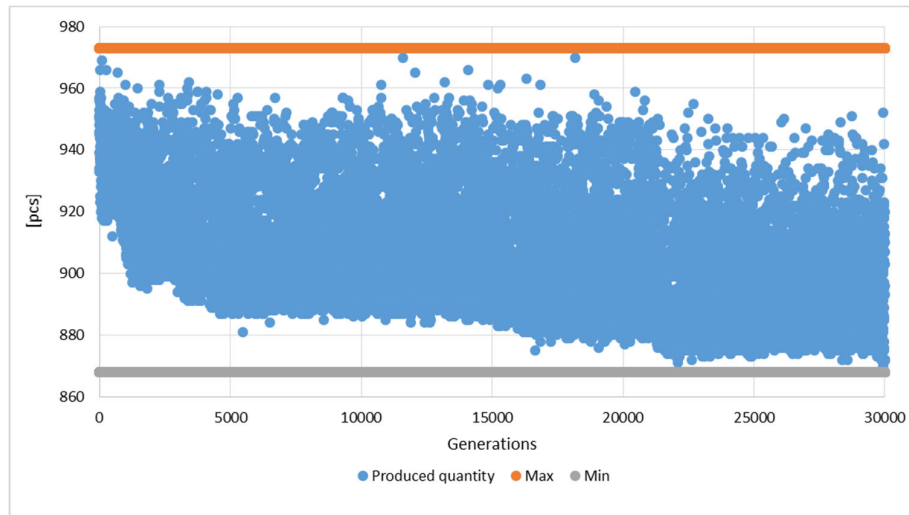
In terms of efficiency, we compared our MPP-based method with PSGA. Table 12 shows the parameter settings of PSGA.

Table 12

Settings for Genetic Algorithm of Plant Simulation (PSGA)

Settings	Values
Decision variables	the array of assignments of jobs and shifts
Objective function (fitness):	the number of the produced product (f_1) the sum of the remaining time (f_2)
Weight	f_1 : 0.7 f_2 : 0.3
Direction of optimization	Maximum
Number of generations	3000
Size of population	10
Settings for genetic operations	Crossover type: OX (order crossover), initial probability: 0.8 Genetic operation: Mutation, initial probability: 0.1

The simulation settings for the PSGA tests were the same as the settings for the MPP-based SBPPC (Table 8). The initial populations for PSGA was generated by using the IPP algorithm. Fig. 7 shows the results of a simulation study of 30,000 detailed production plans generated by PSGA.

**Fig. 7.** Running results of PSGA

During PSGA optimization, the values of primary objective function (number of performed jobs) ranged from 868 to 973. Despite generating and simulating 30,000 candidate solutions, PSGA could not find the solution that was determined by the MPP-based search algorithm, whose objective function value was 981 and where only 5 iterations were required to find it. The disadvantage of PSGA appears when generating newer populations. PSGA only and exclusively takes into account the settings of genetic operations. In contrast, our proposed search method uses purposefully designed modification operations.

Further comparative studies were performed with different problem instances and parameter settings. These demonstrated that the proposed MPP-based search algorithm solves EFJSHR problems more efficiently than the genetic algorithm.

5. Conclusions

In our research, we focused on modelling and solving an integrated production planning, scheduling and control problem of on-demand flexible manufacturing. Key features of the modelled manufacturing system include a resource-constrained manufacturing environment, skilled manual workers, worker-dependent individual processing and material handling times, alternative technological routes, limited available workstations, and predefined shifts. During the research, we developed production planning, scheduling and control algorithms that can flexibly adapt to changing requirements and resource constraints. To generate detailed production plans, we developed a predictive search algorithm that includes advanced event-driven simulation. We also developed reactive control algorithms that are embedded into the simulation to make the necessary decisions at the same time as the execution of manufacturing processes. To solve the integrated problem, we proposed a new approach based on a two-level decision hierarchy. The upper level uses a special multi-objective heuristic search algorithm to generate detailed production plans by determining the values of primary decision variables, while at the lower level, the values of the secondary decision variables are calculated by multi-criteria control algorithms. In this paper, we have described the detailed optimization model of the investigated problem, the solution method and the key algorithms. The effectiveness of the proposed algorithms is demonstrated by presenting some running results. We compared our search algorithm with the genetic algorithm of the Plant Simulation software.

The results show that the proposed search algorithm is much more efficient than the genetic algorithm. The optimization model of the integrated problem of production planning and production control was formulated in such a way that the individual skills of the workers were also taken into account. Due to the complexity of the reactive control and the large number of influencing factors, classical one-way rule-based heuristic algorithms cannot work effectively in different extreme situations. The proposed multi-criteria decision-making algorithms are able to adapt to changing conditions due to their flexibility. Considering several aspects simultaneously is very important because it allows practical needs to be incorporated into the model. By modifying the weights (control parameters) of decision aspects, the management strategy can be fine-tuned. Our further research plans include the study of the possibilities of systematic and/or flexible change of decision aspects within the operation of control algorithms.

The results of the experiments, which focused on the α and β control parameters, have shown that the usage of all decision aspects provides better results than the consideration of only one aspect. When adjusting weights, we have found that any aspect alone works less effectively than all together. Moreover, ignoring any of aspects produces a worse result than the usage of all aspects together. These results demonstrate that the usage of workers' intuitive decisions is less effective than the application of the proposed control algorithms.

Our simulation studies have also shown that taking workers' skills into account contributes greatly to increasing performance. In addition, proactively generated production plans can be performed more accurately, thus fewer corrections have to be made during the execution of processes.

The presented models and solution algorithms are suitable for control of various specific production systems operating in industrial practice. In a cyber-physical environment, the real system and the simulation system can work together in real time, so the control algorithms are able to handle directly real events as well. Workers' skills can be continuously updated in the model. As time progresses, stored values that express the actual ability of workers to perform operations can be continuously refined. The method can also be used to control automated production systems, for example, in the case of automated guided vehicles (AGVs) for material supply, to assign tasks and resources. The proposed approach and model can be adapted to any manufacturing or service processes using appropriately chosen abstract objects.

Cyber-Physical Production Systems, Industry 4.0, Internet of Things (IoT), Cloud Computing and Big Data paradigms and technologies are playing an increasingly important role in the development and management of production systems and virtual enterprises. These technologies allow the practical application of the proposed decision models and algorithms for companies by providing the appropriate input data, computational capacities, and application environment. This makes it possible to integrate the models into applied systems of enterprises.

Acknowledgements

This research was supported by the European Union and the Hungarian State, cofinanced by the European Regional Development Fund in the framework of the GINOP 2.3.4-15-2016-00004 project, aimed to promote the cooperation between the higher education and the industry.

References

Al-Hinai, N., & Piya, S. (2015). Jobshop scheduling for skill-dependent Make-to-order system. *2015 International Conference on Industrial Engineering and Operations Management (IEOM)*, 1354–1358.

- Amjad, M. K., Butt, S. I., Kousar, R., Ahmad, R., Agha, M. H., Faping, Z., Anjum, N., & Asgher, U. (2018). Recent Research Trends in Genetic Algorithm Based Flexible Job Shop Scheduling Problems. *Mathematical Problems in Engineering, 2018*, 1–32.
- Bányai, T., Landschützer, C., & Banyai, Á. (2018). Markov-Chain Simulation-Based Analysis of Human Resource Structure: How Staff Deployment and Staffing Affect Sustainable Human Resource Strategy. *Sustainability, 10*(10), 3692
- Barrera-Diaz, C. A., Oscarsson, J., Lidberg, S., & Sellgren, T. (2018). Discrete Event Simulation Output Data-Handling System in an Automotive Manufacturing Plant. *Procedia Manufacturing, 25*, 23–30.
- Bożejko, W., Uchroński, M., & Wodecki, M. I. (2010). Parallel hybrid metaheuristics for the flexible job shop problem. *Computers & Industrial Engineering, 59*(2), 323–333.
- Brucker, P. (2007). *Scheduling Algorithms* (5th ed.). Springer.
- Brucker, P., Sotskov, Y. N., & Werner, F. (2007). Complexity of shop-scheduling problems with fixed number of jobs: a survey. *Mathematical Methods of Operations Research, 65*(3), 461–481.
- Chung, C. A. (1996). Human issues influencing the successful implementation of advanced manufacturing technology. *Journal of Engineering and Technology Management, 13*(3–4), 283–299.
- Chawla, V. K., Chanda, A. K., & Angra, S. (2018). Scheduling of multi load AGVs in FMS by modified memetic particle swarm optimization algorithm. *Journal of Project Management, 39*–54.
- Durmaz, H., & Koyuncu, M. (2019). Optimization of Assignment Problems In Production Lines With Different Skilled Labor Levels. *International Advanced Researches and Engineering Journal, 123*–136.
- Ferjani, A., Ammar, A., Pierreval, H., & Elkosantini, S. (2017). A simulation-optimization based heuristic for the online assignment of multi-skilled workers subjected to fatigue in manufacturing systems. *Computers & Industrial Engineering, 112*, 663–674
- Fu, H. C., & Liu, P. (2019). A Multi-Objective Optimization Model Based on Non-Dominated Sorting Genetic Algorithm. *International Journal of Simulation Modelling, 18*(3), 510–520.
- Gao, K. Z., Suganthan, P. N., Pan, Q. K., & Tasgetiren, M. F. (2015). An effective discrete harmony search algorithm for flexible job shop scheduling problem with fuzzy processing time. *International Journal of Production Research, 53*(19), 5896–5911.
- Gao, K. Z., Suganthan, P. N., Pan, Q. K., Tasgetiren, M. F., & Sadollah, A. (2016). Artificial bee colony algorithm for scheduling and rescheduling fuzzy flexible job shop problem with new job insertion. *Knowledge-Based Systems, 109*, 1–16.
- Gania, I. P., Stachowiak, A., & Oleśków-Szlapka, J. (2018). Flexible manufacturing systems: Industry 4.0 solution. *DEStech Transactions on Engineering and Technology Research, icpr*, 57–62.
- Ida, K., & Oka, K. (2011). Flexible job-shop scheduling problem by genetic algorithm. *Electrical Engineering in Japan, 177*(3), 28–35.
- Jeet, K., Dhir, R., & Singh, P. (2016). Hybrid Black Hole Algorithm for Bi-Criteria Job Scheduling on Parallel Machines. *International Journal of Intelligent Systems and Applications, 8*(4), 1–17.
- Jia, S., & Hu, Z.-H. (2014). Path-relinking Tabu search for the multi-objective flexible job shop scheduling problem. *Computers & Operations Research, 47*, 11–26.
- Karwowski, W. (2005). Ergonomics and human factors: the paradigms for science, engineering, design, technology and management of human-compatible systems. *Ergonomics, 48*(5), 436–463.
- Kikolski, M. (2016). Identification of production bottlenecks with the use of Plant Simulation software. *Ekonomia i Zarzadzanie, 8*(4), 103–112.
- Kumar, S., Goyal, A., & Singhal, A. (2017). Manufacturing Flexibility and its Effect on System Performance. *Jordan Journal of Mechanical and Industrial Engineering, 11*(2), 105–112.
- Kundu, K., Rossini, M., & Portioli-Staudacher, A. (2019). A study of a kanban based assembly line feeding system through integration of simulation and particle swarm optimization. *International Journal of Industrial Engineering Computations, 421*–442
- Lin, C., & Gen, M. (2008). Multi-criteria human resource allocation for solving multistage combinatorial optimization problems using multiobjective hybrid genetic algorithm. *Expert Systems with Applications, 34*(4), 2480–2490.
- Lin, R., Zhou, G., Liu, A., Lu, H., & Li, T. (2016). Impact of Personnel Flexibility on Job Shop Scheduling. *Scientific Programming, 2016*, 1–8.
- Liu, S. Q., Kozan, E., Masoud, M., Zhang, Y., & Chan, F. T. S. (2017). Job shop scheduling with a combination of four buffering constraints. *International Journal of Production Research, 56*(9), 3274–3293.
- Longo, F., Nicoletti, L., & Padovano, A. (2017). Smart operators in industry 4.0: A human-centered approach to enhance operators' capabilities and competencies within the new smart factory context. *Computers & Industrial Engineering, 113*, 144–159.
- Luh, G.-C., & Lee, S.-W. (2006). Abacterial evolutionary algorithm for the job shop scheduling problem. *Journal of the Chinese Institute of Industrial Engineers, 23*(3), 185–191.
- Michalos, G., Makris, S., Papakostas, N., Mourtzis, D., & Chryssolouris, G. (2010). Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach. *CIRP Journal of Manufacturing Science and Technology, 2*(2), 81–91.
- Oliveira, M., Arica, E., Pinzone, M., Fantini, P., & Taisch, M. (2019). Human-Centered Manufacturing Challenges Affecting European Industry 4.0 Enabling Technologies. *HCI International 2019 – Late Breaking Papers, 507*–517.

- Peruzzini, M., & Pellicciari, M. (2017). A framework to design a human-centred adaptive manufacturing system for aging workers. *Advanced Engineering Informatics*, 33, 330–349.
- Pongchairerks, P., & Kachitvichyanukul, V. (2009). A Particle Swarm Optimization Algorithm On Job-Shop Scheduling Problems With Multi-Purpose Machines. *Asia-Pacific Journal of Operational Research*, 26(02), 161–184.
- Ruiz, R., & Vázquez-Rodríguez, J. A. (2010). The hybrid flow shop scheduling problem. *European Journal of Operational Research*, 205(1), 1–18.
- Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(4), 623–656.
- Shivanand, H. K. (2006). *Flexible Manufacturing System*. New Age International.
- Sirovetnukul, R., & Chutima, P. (2010). The Impact of Walking Time on U-Shaped Assembly Line Worker Allocation Problems. *Engineering Journal*, 14(2), 53–78.
- Sotskov, Y. N., & Shakhlevich, N. V. (1995). NP-hardness of shop-scheduling problems with three jobs. *Discrete Applied Mathematics*, 59(3), 237–266.
- Sun, W., Pan, Y., Lu, X., & Ma, Q. (2010). Research on flexible job-shop scheduling problem based on a modified genetic algorithm. *Journal of Mechanical Science and Technology*, 24(10), 2119–2125.
- Sungur, B., & Yavuz, Y. (2015). Assembly line balancing with hierarchical worker assignment. *Journal of Manufacturing Systems*, 37, 290–298.
- Udaiyakumar, K. C., & Chandrasekaran, M. (2014). Application of Firefly Algorithm in Job Shop Scheduling Problem for Minimization of Makespan. *Procedia Engineering*, 97, 1798–1807.
- Wang, L., Cai, J., Li, M., & Liu, Z. (2017). Flexible Job Shop Scheduling Problem Using an Improved Ant Colony Optimization. *Scientific Programming*, 2017, 1–11.
- Wang, Y., Yang, O., & Wang, S. N. (2019). A Solution to Single-Machine Inverse Job-Shop Scheduling Problem. *International Journal of Simulation Modelling*, 18(2), 335–343.
- Yang, M. S., Ba, L., Liu, Y., Zheng, H. Y., Yan, J. T., Gao, X. Q., & Xiao, J. M. (2019). An Improved Genetic Simulated Annealing Algorithm for Stochastic Two-Sided Assembly Line Balancing Problem. *International Journal of Simulation Modelling*, 18(1), 175–186.
- Yang, X.-S., Karamanoglu, M., & He, X. (2013). Multi-objective Flower Algorithm for Optimization. *Procedia Computer Science*, 18, 861–868.
- Yazdani, M., Gholami, M., Zandieh, M., & Mousakhani, M. (2009). A Simulated Annealing Algorithm for Flexible Job-Shop Scheduling Problem. *Journal of Applied Sciences*, 9(4), 662–670.
- Zaher, H., Ragaa, N., & Sayed, H. (2017). A novel Improved Bat Algorithm for Job Shop Scheduling Problem. *International Journal of Computer Applications*, 164(5), 24–30.
- Zhang, Q., Manier, H., & Manier, M.-A. (2013). A modified shifting bottleneck heuristic and disjunctive graph for job shop scheduling problems with transportation constraints. *International Journal of Production Research*, 52(4), 985–1002.
- Zhang, R. (2013). A Simulated Annealing-Based Heuristic Algorithm for Job Shop Scheduling to Minimize Lateness. *International Journal of Advanced Robotic Systems*, 10(4), 214.
- Ziaee, M. (2014). Job shop scheduling with makespan objective: A heuristic approach. *International Journal of Industrial Engineering Computations*, 5(2), 273–280.
- Zijm, W. H. M. (2000). Towards intelligent manufacturing planning and control systems. *OR Spectrum*, 22(3), 313–345.
- Zinder, Y., Ha Do, V., & Oğuz, C. (2005). Computational complexity of some scheduling problems with multiprocessor tasks. *Discrete Optimization*, 2(4), 391–408.
- Zupan, H., & Herakovic, N. (2015). Production line balancing with discrete event simulation: A case study. *IFAC-PapersOnLine*, 48(3), 2305–2311



© 2021 by the authors; licensee Growing Science, Canada. This is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).