BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 70(6), 2022, Article number: e143830
DOI: 10.24425/bpasts.2022.143830

SPECIAL SECTION

Solving scheduling problems with integrated online sustainability observation using heuristic optimization

Anna BURDUK[1], **Kamil MUSIAŁ**[1], **Artem BALASHOV**[1], **Andre BATAKO**[2]*, **Andrii SAFONYK**[3]

[1] Wroclaw University of Science and Technology, Faculty of Mechanical Engineering, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
[2] Liverpool John Moores University, Faculty of Engineering and Technology,70 Mount Pleasant Liverpool L3 3AF, UK
[3] National University of Water and Environmental Engineering, Department of Automation, Electrical Engineering and Computer-Integrated Technologies, Rivne 33000, Ukraine

Abstract. The paper deals with the issue of production scheduling for various types of employees in a large manufacturing company where the decision-making process was based on a human factor and the foreman's know-how, which was error-prone. Modern production processes are getting more and more complex. A company that wants to be competitive on the market must consider many factors. Relying only on human factors is not efficient at all. The presented work has the objective of developing a new employee scheduling system that might be considered a particular case of the job shop problem from the set of the employee scheduling problems. The Neuro-Tabu Search algorithm and the data gathered by manufacturing sensors and process controls are used to remotely inspect machine condition and sustainability as well as for preventive maintenance. They were used to build production schedules. The construction of the Neuro-Tabu Search algorithm combines the Tabu Search algorithm, one of the most effective methods of constructing heuristic algorithms for scheduling problems, and a self-organizing neural network that further improves the prohibition mechanism of the Tabu Search algorithm. Additionally, in the paper, sustainability with the use of Industry 4.0 is considered. That would make it possible to minimize the costs of employees' work and the cost of the overall production process. Solving the optimization problem offered by Neuro-Tabu Search algorithm and real-time data shows a new way of production management.

Key words: production scheduling; sustainable development; genetic algorithm; Tabu Search; meta-heuristics; intelligent optimization methods of production systems.

1. INTRODUCTION

Production process scheduling is a very complex task, and this complexity results from the necessity of taking into account simultaneously a range of independent and interdependent factors. Thus, in order to remain competitive, companies are compelled to improve by developing and introducing new methods of the process organization. Traditional methods are usually based on the knowledge and experience of process engineers, with the possible use of basic information technology (IT) tools such as simple spreadsheets. These techniques are typical for small and medium-sized enterprises where employees' know-how is the basis for the proper functioning of processes.

In more advanced and often larger enterprises the ERP (Enterprise Resources Planning) software is used to support resource planning and management. This solution is often adequate for the needs of a given enterprise. However, it requires the purchase of appropriate software, special training along with technical support after the purchase. These commercial software packages have the advantage of being versatile in several generic aspects, yet they are very expensive. However, in the case of production processes which require individual solutions, it can be an obstacle. Thus, with the science and technology development, more and more methods are based on intelligent solutions, i.e. heuristic algorithms [1–3].

Among the discrete optimization algorithms, two types of methods can be distinguished: exact methods, which allow one to find the optimal solution, and approximate methods that do not guarantee to provide the optimal solution. With the current size of production issues, approximate methods are much more often used [4].

It is impossible to single out the best method for a given type of problem. The following are widely used in production scheduling problems: greedy algorithm, genetic algorithm, Tabu Search, Scatter Search and Simulated Annealing. In the literature, a given problem is often solved with the help of two or more algorithms with different results [5].

Finding the most beneficial solution to this type of problem is possible thanks to the use of artificial intelligence methods [6]. They allow finding the solutions close to optimal in a relatively short time, which makes them more and more popular [4]. The application of intelligent algorithms allows the reduction of time, labour input and human errors in the processes. Therefore, they can be used to support decision-making processes [7, 8].

1.1. Research goals and aims of the present article

The research analyses the experiences of a large company operating in manufacture industry. The paper examines the case where various parts are produced by two types of employees

*e-mail: A.D.Batako@ljmu.ac.uk

Manuscript submitted 2022-08-02, revised 2022-10-04, initially accepted for publication 2022-10-23, published in December 2022.

with different skills and permissions. The aim of the study was to build a scheduling system that would allow obtaining lower costs of a production process by planning a lower number of employees.

Any kind of production optimization should be provided as a global and total procedure which affects each aspect of the product delivery to the final customer, from its initial design and manufacturing to the customer satisfaction surveys. Efficiency and sustainability of the chained process are always defined as efficient and they sustain its weakest chain [9–11]. There is not much of an efficiency boost in introducing a brand new scheduling model if there is a chance to stop the whole manufacturing process in case of breakdowns, repairs or another last day maintenance. Thus, the company implemented a production optimization approach to the Total Productive Maintenance.

The present article is a logical continuation of our previous study [12] on scheduling in the case of multi-objective production process with the use of heuristic optimisation. This study is aimed to illustrate the significance of considering real-time observation on machines sustainability and interoperating this data in scheduling system.

The work consists of five sections. Section 1 describes the need to write this work. Section 2 thoroughly discusses the analysed production problem and the proposed approach to the solution. Section 3 describes production organization supported by a set of remote observation sensors. Section 4 presents the algorithms proposed for the solution to the problem. Section 5 presents the results of the operation of the algorithms, and then the outcome of this work is presented in Section 6.

2. DECISION-MAKING PROBLEM

A decision-making problem, inspired by a real company which is a global manufacturer of tools and components for the production of electric machines, is considered in the paper. Up to now, all decision-making processes have been based only on a human factor. Over time an observation was made that such an approach often generates financial losses. Then, a possibility was noticed to reduce costs through the implementation of new decision-making methods.

One of the areas subjected to observation was the main production process in one production hall. The company produces 80 products in 28 production modules. It means that several products are assigned to each module and cannot be produced parallelly. Each Friday, the company analyses customers' orders. It prepares a production plan for the nearest week and calculates the number of employees needed to fulfill this plan (consisting of up to 80 products in various volumes). For the described production process, two types of employees are necessary. Each has different skills, permissions and is allowed to use different workstations.

A situation where, among all processes, a certain amount of them require an employee of one type and other processes require an employee of another type can be presented as a simplified example of different production plans. Assuming, in this case, that the factory has enough machines and workstations for each and every employee, two scheduling plans can be considered – the one of minimizing overall production time by considering new employees for each process and the one of "minimizing production costs" by assigning only two employees of different types to work through these processes sequentially. "Minimizing production costs" is in quotation marks here because the unit costs of processes remain constant although auxiliary expenses (such as insurance, paid breaks and lunches, holiday bonuses) are reduced.

The company does not make a schedule assigning specific employees neither to the products nor to the workstations. During the production process decisions are made quickly by a foreman based only on his subjective opinions.

From the point of view of the company oriented to profit maximisation, the proper usage of human resources is an important issue. In order to maintain the competitive advantage, it is necessary to minimize all costs, especially those of production. Therefore, an employee scheduling management system should be flexible enough to adapt itself to the ambiguous and changing specificity of production and customer orders. Considering various conditions of orders, such as delivery deadlines and unit values as well as some unforeseen factors (sick leaves, birthdays, lateness as well as equipment failures, delivery delays, etc.), the system should offer various scenarios (which will be called further "options") of the production schedule. These variants are based on defining a specific production program (see Fig. 1) for each part being in production in the observed moment of time.

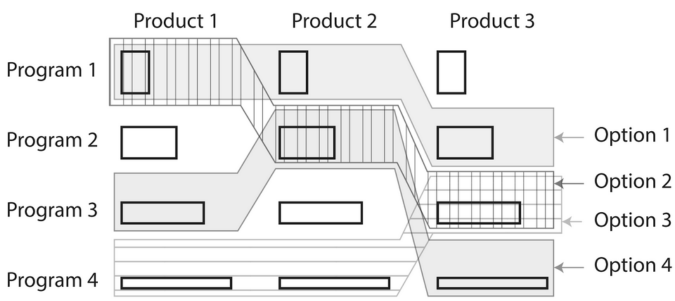

Fig. 1. Selecting a production plan for a different scheduling variant

As step one the programs were calculated – several for each specific product, assuming a various number of employees and production time needed. An option is a set of the selected programs assuming the determined total demand of time and the number of employees. Here, general analytical algorithms were used. In some particular situations and for the further development of the idea as general, some empirical or statistical coefficients can be applied in order to achieve more precision.

3. REMOTE OBSERVATION OF MACHINE PARK SUSTAINABILITY AND VIRTUAL SUPPORT FOR TPM FEATURES

Any kind of production optimization should be provided as a global and total procedure that affects each aspect of a product delivery to the final customer, from its initial design and manufacturing to the customer satisfaction surveys. Efficiency and sustainability of the chained process is always defined as efficient and sustain its weakest chain [13]. There is not much of an efficiency boost of introducing a brand new scheduling model if there is a chance to stop the whole manufacturing process in case of breakdowns, repairs or another last-day maintenance. Thus, the company implemented a production optimization approach to the Total Productive Maintenance (TPM) [14, 15]. MP is a well-known approach that was developed in the 1960s, and hence, it will not covered entirely in this report. However, the main idea to migrate maintenance activities from reactive to proactive and then, to fit them flexibly into a production schedule, is correlating with a methodology of flexible scheduling described in the present work. That is what led to the idea of including maintenance activities in the scheduling methodology [16]. The traditional approach to TPM consists of 5S as a foundation and eight supporting pillars. Only four of them have a relation to our scheduling methodology. Therefore, a short description of them is provided below. Autonomous maintenance is a set of activities carried out by operators such as cleaning, lubricating and inspecting, ensuring that machinery is always clean and lubricated, and it helps to identify issues before they become failures and frees up maintenance staff for higher-level tasks. Planned maintenance is a set of preventive maintenance activities which are carried out by a prepared maintenance team and happen following a schedule that was developed on the basis of the analysed metrics and history of the previous breakdowns and machine manufacturer's service guidelines. Quality maintenance is an approach of the quality control of each part after each process (or at least as much as reasonably possible) and prevents the defected products from moving down the line, which could lead to a lot of rework [17,18]. Safety, health and environmental activities ensure that employees can perform their tasks in a safe place without health risks. Apart from the obvious benefits, a safe environment also provides a lack of incidents, positively influencing the following production scheduling [19]. Since it is necessary to observe the current online state of equipment and machines in order to have the ability to fit maintenance activities in a production schedule along with retooling operations, a set of remote observation sensors was introduced. These sensors are used to remotely inspect machine condition and sustainability as well as to forecast necessary supporting events such as maintenance, repair, calibration, etc. These sensors are categorized into 3 classes:

- Machine condition remote observation sensors that directly inspect machine characteristics and behaviour such as temperature, wear, humidity, deformation, magnetic field, vibration, sound frequency, etc.;
- Machine behaviour remote observation sensors which indirectly (based on parts quality) observe machine condition – a computer image for the parts quality control;
- Work, health and safety (referred to as WHS) environment supporting sensors – the sensors which support human health conditions such as air humidity sensors, CO2 sensors, and air pollution. The sensors and their data from this category at this moment are not analysed and interpreted in the described production organization optimization algorithm. Implementing WHS environment observation to the scheduling algorithm is one of the top priorities.

3.1. General statement of observation sensors implementation

The first thing that shall be mentioned is that since this is a proof of concept project, sensors and other electrical components were chosen for fast and robust prototyping. Each aspect and decision that were made during this study proceeded, keeping in mind a prototyping approach, providing the ability for fast and flexible corrections. Assuming the positive outcome of the implementation of this strategy, all the sensors, other hardware components and software are to be replaced with more sustainable ones. Currently, each machine and workplace for unique operations have its own set of sensors connected to the main control board (also personal for each machine and process). The schematics of the sensors and their communication is presented in Fig. 2.

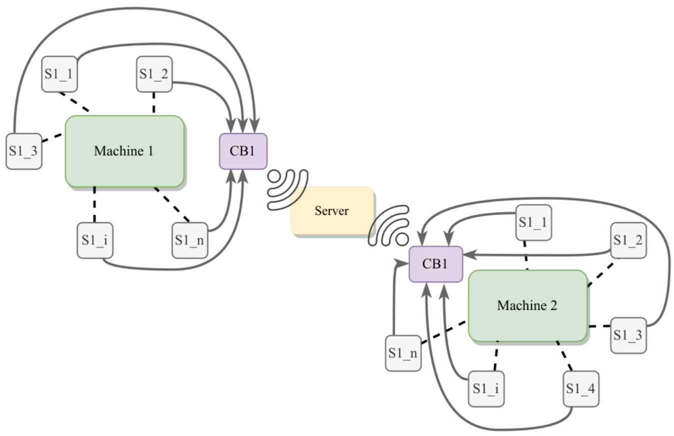

Fig. 2. Sensor maintenance support (M_i – machines, CB_i – control boards, S_i – sensors)

Thus, the main control board for each machine was chosen to be the ESP32 development kit (ESP-WROOM-32). There were several reasons for this decision. First, it has wireless communication protocols onboard, which are necessary for the convenient implementation in a wide machine park. It also eliminates further actions of creating some custom solutions for wireless communication with a different board. Second, it possesses two CPU cores that ensure smooth data receiving from the sensors and analysing without additional interferences. Next, there is a sufficient amount of ADC pins for analog sensors, as well as a number of prebuilt communication protocols, such as I2C, I2S, SPI, and UART. The operation voltage of 3.3V is also considered to be more precise for analysing ADC signals from analog sensors (including MAX9814, which has an output signal of 2.5V). The overall characteristics of this board (32bit architecture, 160Mhz CPU frequency, 512Kb RAM, 4Mb Flash memory, etc.) are considerably better than its concurrent boards (Arduino, STM32, ESP8266).

Another thing worth mentioning is that most of the machines already have, in one part or another, some kind of sensors by themselves: ammeters in welding machines, temperature fuses in power blocks, temperature sensors in coolant thermostats, etc. These sensors were not used in the structure of the presented sensor. Moreover, in some cases additional sensors, which replicate the existing ones, were installed as part of the global structure. Caused by various reasons, depending on a specific situation, in some cases it was impossible to group the existing sensors with the control board, in some cases it was impossible to reach these sensors without damaging the integrity of covering components and potentially losing the machine warranty and in some cases these sensors tolerances were defined as not precise enough.

Although the analog-digital converter on the control board is considered to be moderately better than its competitors, yet it is still not sufficient to read data directly from analog sensors. Moreover, both analog and digital sensors are affected by interferences which usually are either constant noise of the sensor itself with the same deviation, or random noise (3a) that occurs

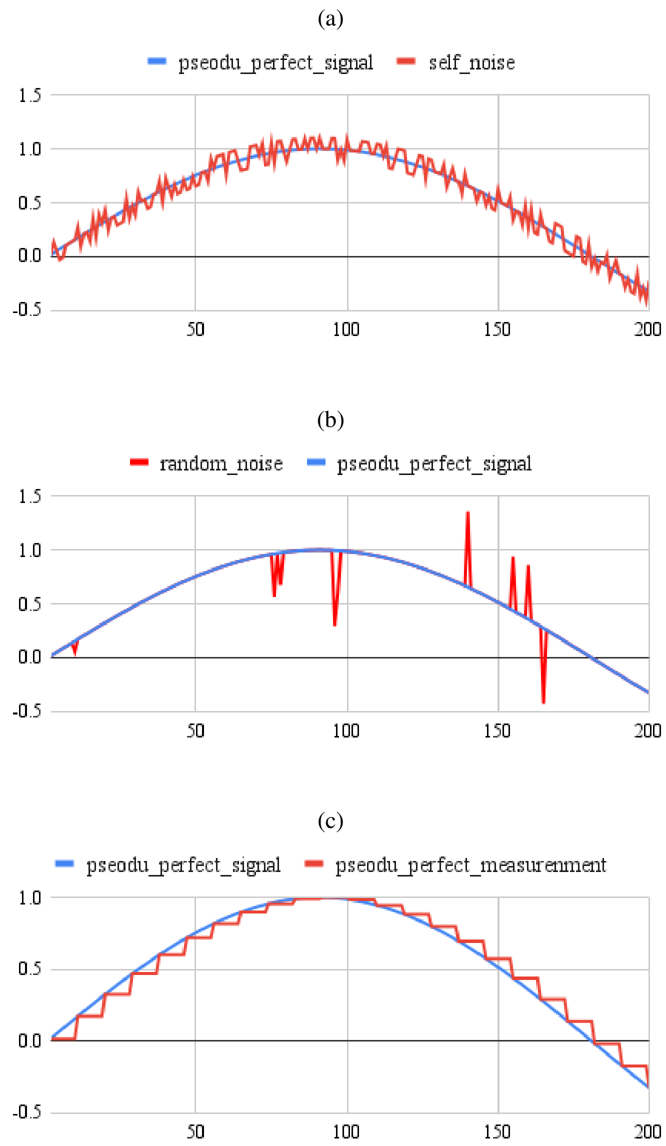

Fig. 3. Sensor noise and iterative measurement

under various random (most often external) circumstances (3b). In addition, the data receiving also happened iteratively (3c).

All the circumstances mentioned above lead to the necessity of data filtering. Thus, several filtering algorithms were applied to the sensor data.

- The median filter is used to remove high peaks and sudden value changes. It finds the median value, not averaging but choosing the median from the presented. The biggest advantage of this filter is that it does not calculate but simply compares the numbers. This makes it faster than other types of filters. However, it does not present sufficient results for filtering small noise and it also detains the data.
- Exponential moving (running) average also performs positively removing both noise and random peaks. Its primitive description would be "filtered data += (new data – filtered data) * coefficient". The coefficient is used to adjust the filter – the smaller it is, the smoother the data, but the more affected by high fluctuations.
- In order for the running average to work correctly with sharply changing signals, the coefficient can be made adaptive so that it adapts to sudden changes in the value, for example: if the filtered value is "far away" from the real – the coefficient increases dramatically, allowing to quickly reduce the "gap" between the values. If the value is "close", the coefficient is set small in order to filter the noise well".

Even though the data from sensors is filtered, sometimes (37 out of 40 trials, according to the preliminary observations) inadequate behaviour was observed, which is considered to have an interference nature. However, since these interferences are interpreted by the algorithm as an inadequate machine behaviour, and then analysed along with the regular machine wear, that leads to preliminary maintenance events. In other words, these occurrences of inadequate behaviour leads to redundancy maintenance, which is still a type of proactive maintenance, and, nevertheless, do not allow skipping service events up to reactive maintenance. Which, considering above is considered as success.

3.2. Sensors observation outcome and maintenance events

The data received from the sensors in the 60-day period, which was enough to cover the most possible orders change, was analysed. Currently, there is no universal solution for machine sensoring data interpretation. Partially, this data was analysed empirically, by comparing current behaviour observation with the information from sensors, partially by referencing with machine service manual. After careful consideration of 6 different types of machines, 29 different maintenance events were established, such as tool sharpening, coolant replacement, guides cleaning, oil refiltering, sliders greasing, etc.

In addition to all the information behind these events, there are only 3 types of data operating which are considered by scheduling algorithm: duration of the maintenance process, safe period during which this maintenance event could be maintained without substantial loss for production line, and identification of the specific machine, which require undergoing maintenance.

4. SELECTED ALGORITHMS

Many kinds of decision-making problems can be solved with using meta-heuristic algorithms [20] because their application allows one to find a near-optimal solution in a reasonable time without the transformation into mathematical formulations [21–23]. Among the widely used are: genetic algorithm (GA) [7] and Tabu Search (TS) [24]. Papers comparing TS and GA algorithms for selected issues [12, 25] agree that both of them are suitable for production scheduling optimization problems and it is impossible to unequivocally indicate the better one. For the considered problem of a decision-making process in production scheduling, GA and TS algorithms were chosen to use and additional Neuro-Tabu Search (NTS) uniting in itself TS with neural network was taken under consideration.

4.1. Genetic algorithm and Tabu Search

Genetic algorithm is an example of an algorithm analyzing many samples on each step. Process of solution searching consists of several steps: initiating – generating an initial group of solutions, solutions evaluation in terms of acceptability and quality in accordance with defined objective function and choosing solutions that can be used for next population generating – next group of solutions. Genetic algorithm belongs to class of evolution algorithms [25].

The Tabu Search algorithm is one of the most effective methods of constructing heuristic algorithms for scheduling problems. In the method proposed by Glover [26], the neighbourhood of the base solution is checked in each iteration in order to find the solution with the best value of the objective function. The neighbourhood of the solution is generated with the modifications base solution called movements. The method owes its name to the prohibition mechanism [27]. This mechanism is implemented in the form of a list on which the attributes of the solutions generated in several previous iterations are stored. Therefore, the list is a sort of memory of the searching process. During the neighbourhood search for the base solution, based on the contents of the forbidden solution list, the neighbourhood is divided into two subsets: the set of forbidden solutions and the set of non-forbidden solutions. Only the set of non-forbidden solutions is considered in selecting the best neighbour solution.

As it was mentioned, the study of present article is based on our previous research [12], where implementation of Tabu Search and genetic algorithm were observed with the same initial historical company data to present research. Results obtained are published in the mentioned article and could be briefly interpreted as moderate success of both algorithms. However, the main difference is that previous research did not consider forecasting delays, caused by service and maintenance events. For valid verification of algorithms, history of occurred maintenance delays was plotted on a top-optimized schedule. The algorithms of Tabu search with neural network mechanisms are presented in this study as proceeding investigation of the use of heuristic algorithms.

4.2. Neuro-Tabu Search

Neuro-Tabu Search is a combination of Tabu Search algorithm and simple (one perceptron) neural network [28]. TS (described above) has a very restrictive mechanism of taboo movements. Self-organizing neural network implemented in following case weakens the restrictiveness of the mechanism which allows to analyze the solutions rejected by TS. In the considered problem, the solution is represented as pairs (π, a). In the most effective algorithms based on the search methods with prohibition, the neighbourhood is generated with the use of "insert" movements.

Let $v = (x, y, z)$ be a triplet, where x and y represent the position in π, and z is the manner of execution of $\pi(x)$. The adjacent solution (π_v, a_v) resulting from π and a by making the v move takes the form:

$$\pi_v = (\pi(1), \ldots, \pi(x-1), \pi(x+1), \ldots, \pi(y), \pi(x), \ldots, \pi(n)), \quad \text{if } a < b, \tag{1}$$

$$\pi_v = (\pi(1), \ldots, \pi(x), \pi(y) \ldots, \pi(x-1), \pi(x+1), \ldots, \pi(n)), \quad \text{if } a > b, \tag{2}$$

whereas

$$a_v = (a_1 \ldots, a_{x-1}, z, a_{x+1}, \ldots, a_n). \tag{3}$$

The set of all such moves is defined as follows:

$$V = \{(x,y,z)\colon\ y \notin \{x-1,x\},\ x,y \in \{1,\ldots,n\},\ z \in \{1,\ldots,r_{\pi(a)}\}\}. \tag{4}$$

The $y \notin \{x-1, x\}$ condition was added to eliminate generating the moves solutions identical to (π, a) as well as redundant movements (generating solutions, that can be generated with other moves from this set). The number of solutions in the vicinity of $\mathscr{N}(V, \pi, a)$ solution (π, a) generated by the moves from the set V is $n(n-2)\sum_{i=1}^{n} r_i$. Taking into account the computational complexity of determining the value of the objective function $O(n^2)$, the complexity of the neighbourhood search generated by the moves from the set V is $O(n^4, ave(r))$, where $ave(r)$ is an average value of the number of manner of the task execution.

4.3. Tabu mechanism with neural network

The variables used in the algorithm description are:
$J = 1, \ldots, n$ – set of tasks to be performed,
$M = 1, \ldots, m$ – set of workstations,
$\pi = (\pi(1), \ldots, \pi(n))$ – the permutation that determines the order in which the tasks are entered into the system,
$a_j = (a_j(1) \ldots, a_j(n))$ – the task execution method.

In the algorithms based on the search method with prohibition for scheduling problems, the prohibition mechanism is implemented in the form of a list of constraints on which forbidden sequential relations between tasks are stored.

Applying this approach to the considered problem, the elements of the prohibition list T are triples of the form (a,b,c), where a and b denote the tasks and c is the way of performing the task. The solution (π,a) is forbidden if there is a triplet (a,b,c) in the list T such as task a is performed by method c and is executed in π before task b.

In each iteration of the algorithm, the list T is updated, i.e. the attributes of the newly generated solution are added. If $v=(x,y,z)$ is a movement generating the solution selected in a given iteration, then the list T is added:

- $(\pi(x),\pi(x+1),a_{\pi(x)})$ if $x<y$,
- $(\pi(x-1),\pi(x),a_{\pi(x)})$ if $x>y$.

The described prohibition mechanism is very restrictive, i.e. the mechanism prohibits the generation of the previously generated solutions and many others. For example, the triple (a,b,c) prohibits the insertion of the a task directly by the b task and to all positions in π before the a task position. The high restrictiveness of this mechanism has its advantages, namely it allows one to limit the penetration of subspaces consisting of similar solutions.

The use of such a restrictive mechanism for the considered problem significantly reduces the set of non-forbidden neighbouring solutions in the vicinity of the base solutions. This is due to the fact that in the discussed planning period, as a rule, the critical sequence consists of several tasks and it often happens that the surroundings of the base solution do not have non-prohibited solutions. Therefore, it was decided to weaken the restrictiveness of the mechanism by using a simple self-organizing neural network. The proposed neural network consists of one layer of neurons and to each neuron one adjacent solution of the base solution is assigned (Fig. 4).

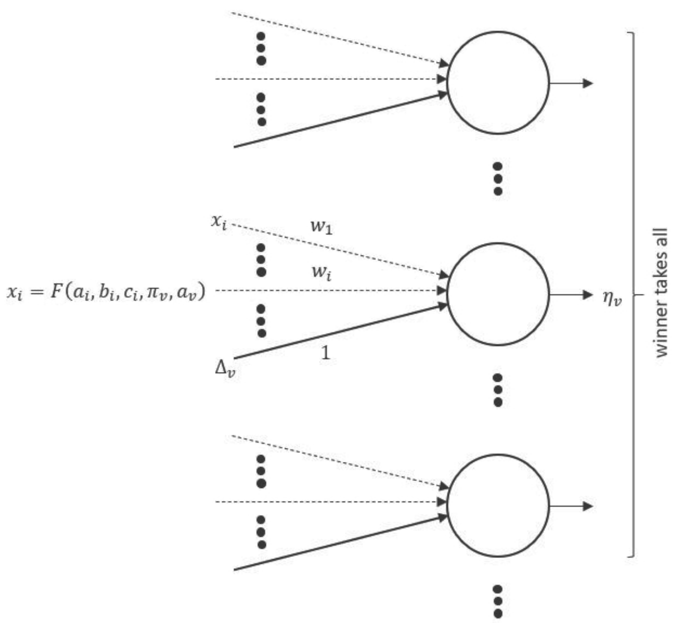

Fig. 4. Structure of the neural network

Binary numbers (0 or 1) are given at the neuron inputs. A value of 1 means that the neighbouring solution has some features, and 0 that it does not. The value of the function to the input i of the neuron assigned to the solution (π_v,a_v) is given:

$$\mathscr{F}(a_i,b_i,c_i,\pi^{(v)},\alpha^{(v)})=\begin{cases}1, & \text{if in } (\pi^{(v)},\alpha^{(v)}) \text{ task } a_i \\ & \text{is made by method } c_i \\ & \text{before task } b_i, \\ 0, & \text{otherwise.}\end{cases} \tag{5}$$

So each neuron has $|I|$ number of enters, where the set

$$I=\{(a_i,b_i,c_i): a_i\neq b_i, a_i,b_i\in J, c_i\in\{1,\ldots,r_{a_i}\}\}. \tag{6}$$

The neuron has an additional input, the value of which depends on the quality of the solution and is equal to

$$\Delta_v=\frac{C_{\max}(\pi_v,a_v)-C^*_{\max}}{C^*_{\max}}, \tag{7}$$

where $C^*_{\max}$ is the value of the objective function of the best solution found so far.

The output value of v neuron is defined as follows

$$\mu_v=\alpha\cdot\Delta_v+\sum_{i\in I}(\beta+\omega_i)\cdot\mathscr{F}(a_i,bi,ci,\pi_v,a_v). \tag{8}$$

The v neuron is activated, according to the winner-take-all principle, if it has the lowest μ_v value of all neurons. The neuron activation is accompanied by two activities: generation of a new base solution, i.e. in the next iteration, the base solution will be solution (π_v,a_v) and the reorganization of the network by changing the input weights of neurons. In the case of activation of the neuron corresponding to the solution (π_v,a_v) for $v=(x,y,z)$, the weight of the i input corresponding to the triplet is modified:

- $(\pi(x)),\pi(x+1),a_{\pi(x)}$ if $x<y$,
- $(\pi(x-1)),\pi(x),a_{\pi(x)}$ if $x>y$,

according to the following relationship

$$\omega_i=\beta(1+\omega_i), \tag{9}$$

while the remaining neurons according to the relationship

$$\omega_i=\beta\omega_i. \tag{10}$$

The proposed method of modifying the network distinguishes forbids remembered in the original prohibition list. The most recent forbids have greater weight (9) than the earlier forbids. This is the result of being forgotten by the expression (10). Additionally, the network remembers that a given prohibition has been applied once or more times before, and this fact is amplified by the neuron.

The schematic diagram of the Neuro-Tabu Search (NTS) algorithm is presented in Algorithm 1.

Algorithm 1. Neuro-tabu search algorithm (NTS)

Variables

(π^0, α^0) – *initial solution*,

(π^*, α^*) – *best solution found*,

(π, α) – *current solution*.

1. $(\pi^*, \alpha^*) \leftarrow (\pi^0, \alpha^0)$;
2. $(\pi, \alpha) \leftarrow (\pi^0, \alpha^0)$;
3. Until the STOP criterion is met, execute

   3.1 Designate neighborhood $\mathcal{N}(\pi, \alpha)$ of current solution π;

   3.2 Designate a solution $(\pi^{(v)}, \alpha^{(v)})$ using a neural network;

   3.3 If $C_{\max}(\pi^*, \alpha^*) > C_{\max}(\pi^{(v)}, \alpha^{(v)})$ then

   3.3.1 $(\pi^*, \alpha^*) \leftarrow (\pi^v, \alpha^v)$;

   3.4 $(\pi, \alpha) \leftarrow (\pi^{(v)}, \alpha^{(v)})$;

   3.5 Perform a self-reorganization of the neural network

5. OBTAINED RESULTS AND DISCUSSION

For the evaluation of the algorithm, 9 company production plans from 3 factories have been used. Each production plan includes 3 month production period with machines unavailability information.

To provide a valid verification methodology, optimised schedules obtained as a result of presented algorithms were compared to company history data. However, the system of machine sustainability remote observation was not designed yet to respond to a real-time forecasting of machine service necessity and it is not possible to present a direct analysis based on historical data. In order to observe significance of adopting selected algorithms and proposed remote observation system, the data were analysed separately for algorithm usage with and without considering to history of machines unavailability.

In the first case, delays, which occurred due to maintenance, were simply plotted on top of optimised schedule to imitate actual machine unavailability. The same non-operational time was assumed. The obtained results were compared with the real company result.

In the second case, it was assumed that system of remote observation responded to 100% of existent threats. Hence, all existent delays were scheduled to a position of less profit loss, in accordance with their "safety notification periods" as early limit and the time they actually occurred as late limit. Due to the fact that during examination of the system of machine sustainability remote observation cases of false alarms were observed (inadequate sensors behaviour, described in Section 3.2), and no cases of failed alarm were observed (when maintenance was actually needed, but not detected), this assumption is considered valid. Occurrences of false alarm were considered.

Two simulations have been performed:

1. GS, TS and NTS solutions generated and compared with the company production plans excluding history of machines unavailability,
2. GS, TS and NTS solutions generated including history of machines unavailability and additional maintenance planned.

5.1. TS, GA and NTS compared with company production plan

In Fig. 5 (presenting data from Tables 1 and 2) the total company profit was presented and compared with solutions proposed by algorithms TS, GA and NTS. Results have been obtained based on:

- a number of retooling including the production time and the number of retooling that would be included applying the production schedule generated by TS, GA and NTS algorithms,
- a number of penalties the company had received and would receive applying the production schedule generated by TS, GA and NTS algorithms.

Table 1
Company profit

sample	products	orders	retooling	penalties	profit*
1	27	86	62	3	1.311
2	12	93	64	4	1.186
3	17	72	47	3	0.738
4	29	134	55	7	0.960
5	36	112	74	4	0.854
6	37	99	70	5	1.540
7	15	86	42	8	1.034
8	18	102	58	5	1.163
9	22	126	74	9	1.040

[*] mil. €

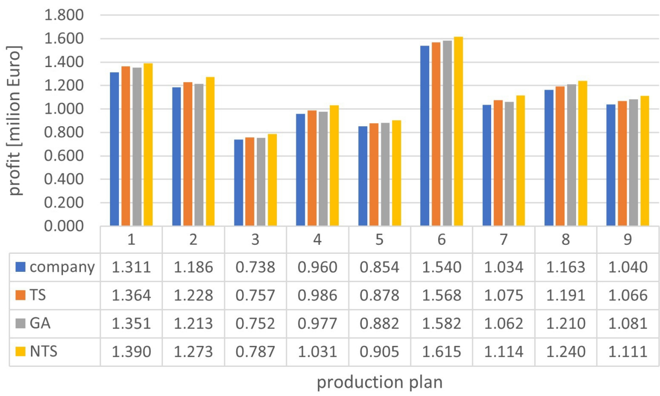

Fig. 5. TS, GA and NTS solutions compared with company profit

The solutions delivered by TS, GA and NTS improves the total profit by respectively 1.82%–4.07%, 1.84%–4.03% and 4.92%–7.79% compared with company solution. During work, the company does not manage all the orders in the required time and the penalties are given. Schedules proposed by algorithms can prevent penalties. As mentioned before and expected in some cases (samples 1, 2, 3, 4, 7) TS algorithm provided better solution (higher profit) than GA and vice versa (samples 5, 6, 8, 9). The research confirms that both algorithms can be successfully used for scheduling problem and it is impossible to arbitrarily highlight better ones. In all 9 cases schedule proposed by NTS algorithm exceeds profit proposed by TS and GA that confirms that this approach is suitable for this type of scheduling problems.

5.2. TS, GA and NTS with IoT data

The second comparison took under consideration machine unavailability time from the company production data and preventive actions proposed by methodology. As mentioned, the majority of failures existing in the production under investigation were checked and preventive actions were proposed.

Table 2

Profit calculated by algorithms

Sample	TS		GA		NTS	
	profit*	gain [%]	profit*	gain [%]	profit*	gain [%]
1	1.364	4.07	1.351	3.07	1.39	6.04
2	1.228	3.53	1.213	2.27	1.273	7.29
3	0.757	2.61	0.752	1.92	0.787	6.59
4	0.986	2.77	0.977	1.84	1.031	7.41
5	0.878	2.89	0.882	3.30	0.905	6.01
6	1.568	1.82	1.582	2.72	1.615	4.92
7	1.075	3.96	1.062	2.76	1.114	7.79
8	1.191	2.37	1.210	4.03	1.240	6.58
9	1.066	2.50	1.081	3.91	1.111	6.78

* mil. €

Figure 6 (presenting data from Table 3) shows the solutions proposed by TS, GA and NTS algorithms. Also "gain to TS" and "gain to GA" have been specified to compare NTS algorithms with other ones. As seen, NTS algorithm can deliver profit higher of 1.81%–5.69% and 1.65%–4.75% then respectively TS and GA algorithm. The following comparison presents profit that could be generated if IoT solution was deployed together with NTS algorithm.

Table 3

Comparison of profit calculated by algorithms including company IoT data

Sample	TS	GA	NTS		
	profit*	profit*	profit*	gain to TS [%]	gain to GA [%]
1	1.354	1.341	1.405	3.71	4.75
2	1.215	1.216	1.267	4.36	4.20
3	0.764	0.763	0.787	3.13	3.22
4	0.994	0.987	1.012	1.81	2.57
5	0.875	0.875	0.903	3.15	3.19
6	1.562	1.587	1.631	4.41	2.76
7	1.066	1.061	1.107	3.85	4.33
8	1.185	1.208	1.253	5.69	3.72
9	1.071	1.081	1.099	2.58	1.65

* mil. €

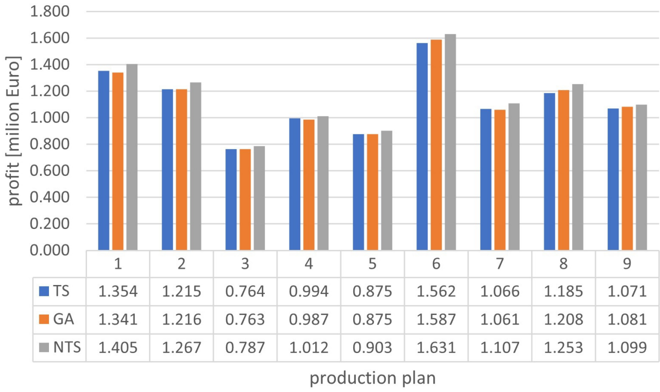

Fig. 6. Profit calculated by algorithms including company IoT data

6. CONCLUSIONS

In accordance with the obtained results, a moderate improvement of company's scheduling system could be concluded as it is with the use of proposed algorithms: Tabu Search, genetic algorithm and Neuro-Tabu search and a moderate improvement of scheduling using heuristic approach using real-time machine maintenance forecasting system.

The NTS algorithm for the production scheduling with machines unavailability management with the use of IoT can potentially revolutionize the way production systems work. Early detection of potential faults and adjusted preventive actions are essential for an efficient, constant and sustainable production process. Together with the efficient production scheduling, it allows the company to increase its market competitiveness. The next step is to implement a schedule in the current production process and to verify the result obtained in this paper.

Scheduling algorithms have a long history of implementation in manufacture. This article aimed to underline their potential and ability to consider various factors. In addition to the deterministic case, the algorithms can also be solved quickly in real time, while obtaining an optimal or close to optimal solution that will help to deal with unexpected changes. It is highly recommended to continue the research at a bigger scale where the complexity of a process is much higher than computing possibilities of the current computers.

REFERENCES

[1] G. Bocewicz, I. Nielsen, and Z. Banaszak, "Iterative multimodal processes scheduling," *Annu. Rev. Control*, vol. 38, no. 1, pp. 113–122, 2014.

[2] E.M. Tachizawa and C.G. Thomsen, "Drivers and sources of supply flexibility: an exploratory study," *Int. J. Oper. Prod. Manage.*, vol. 27, no. 10, pp. 1115–1136, 2007.

[3] P. Pawlewski, P. Golinska, M. Fertsch, J.A. Trujillo, and Z.J. Pasek, "Multiagent approach for supply chain integration by distributed production planning, scheduling and control system," in *International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI 2008)*, 2009, pp. 29–37.

[4] W. Bożejko, M. Uchroński, and M. Wodecki, "Parallel tabu search algorithm with uncertain data for the flexible job shop problem," in *Artificial Intelligence and Soft Computing*. Lecture Notes in Computer Science, vol. 9693, 2016, pp. 419–428.

[5] I. Rojek, "Tooling selection in technological processes using neural networks," *Arch. Mech. Technol. Mater.*, vol. 35, pp. 41–50, 2015.

[6] K. Antosz and D. Stadnicka, "The results of the study concerning the identification of the activities realized in the management of the technical infrastructure in large enterprises," *Eksploat. Niezawodn. (Maintenance and Reliability)*, vol. 16, no. 1, p. 112–119, 2014.

[7] A. Burduk and K. Musiał, "Genetic algorithm adoption to transport task optimization," in *International Joint Conference SOCO'16-CISIS'16-ICEUTE'16*, 2016, pp. 366–375.

[8] A. Burduk and K. Musiał, "Optimization of chosen transport task by using generic algorithms," in *Computer Information Systems and Industrial Management*. Lecture Notes in Computer Science vol. 9842, 2016, pp. 197–205.

[9] M. Jasiulewicz-Kaczmarek and A. Gola, "Maintenance 4.0 technologies for sustainable manufacturing-an overview," *IFAC-PapersOnLine*, vol. 52, no. 10, pp. 91–96, 2019.

[10] J. Patalas-Maliszewska and H. Łosyk, "An approach to assessing sustainability in the development of a manufacturing company," *Sustainability*, vol. 12, no. 21, p. 8787, 2020.

[11] I. Rojek, M. Macko, D. Mikołajewski, M. Sága, and T. Burczyński, "Modern methods in the field of machine modelling and simulation as a research and practical issue related to industry 4.0," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 69, no. 2, p. e136717, 2021.

[12] K. Musiał, A. Balashov, A. Burduk, A. Batako, and A. Safonyk, "Solving scheduling problems in case of multi-objective production using heuristic optimization," in *International Scientific-Technical Conference MANUFACTURING*, 2022, pp. 13–24.

[13] I. Rojek, D. Mikołajewski, and E. Dostatni, "Digital twins in product lifecycle for sustainability in manufacturing and maintenance," *Appl. Sci.*, vol. 11, no. 1, p. 31, 2020.

[14] H. Khadiri, S. Sekkat, and B. Herrou, "An intelligent method for the scheduling of cyber physical production systems," *Manage. Prod. Eng. Rev.*, vol. 13, p. 44—51, 2022.

[15] P. Tambare, C. Meshram, C.-C. Lee, R.J. Ramteke, and A.L. Imoize, "Performance measurement system and quality management in data-driven industry 4.0: A review," *Sensors*, vol. 22, no. 1, p. 224, 2021.

[16] R. Thakur and D. Panghal, "Total productive maintenance," in *Lean Tools in Apparel Manufacturing*. Elsevier, 2021, pp. 355–379.

[17] S. Chaurey, S.D. Kalpande, R. Gupta, and L.K. Toke, "A review on the identification of total productive maintenance critical success factors for effective implementation in the manufacturing sector," *J. Qual. Maint. Eng.*, 2021, doi: 10.1108/JQME-11-2020-0118.

[18] M.M. Saxena, "Total productive maintenance (tpm); as a vital function in manufacturing systems," *J. Appl. Res. Technol. Eng.*, vol. 3, no. 1, pp. 19–27, 2022.

[19] A. Tayal and N.S. Kalsi, "Review on effectiveness improvement by application of the lean tool in an industry," *Mater. Today: Proc.*, vol. 43, pp. 1983–1991, 2021.

[20] M. Dorigo, "Optimization, learning and natural algorithms," Ph.D. dissertation, Politecnico di Milano, 1992.

[21] A. Goyal, W. Lu, and L.V. Lakshmanan, "Celf++ optimizing the greedy algorithm for influence maximization in social networks," in *Proceedings of the 20th international conference companion on World wide web*, 2011, pp. 47–48.

[22] D. Górnicka, M. Markowski, and A. Burduk, "Optimization of production organization in a packaging company by ant colony algorithm," in *International Conference on Intelligent Systems in Production Engineering and Maintenance*, 2017, pp. 336–346.

[23] J. Grabowski and M. Wodecki, "A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion," *Comput. Oper. Res.*, vol. 31, no. 11, pp. 1891–1909, 2004.

[24] J.-F. Cordeau, M. Gendreau, and G. Laporte, "A tabu search heuristic for periodic and multi-depot vehicle routing problems," *Networks: Int. J.*, vol. 30, no. 2, pp. 105–119, 1997.

[25] A. Burduk, K. Musiał, J. Kochańska, D. Górnicka, and A. Stetsenko, "Tabu search and genetic algorithm for production process scheduling problem," *LogForum*, vol. 15, no. 2, pp. 181–189, 2019.

[26] M. Laguna, J.W. Barnes, and F.W. Glover, "Tabu search methods for a single machine scheduling problem," *J. Intell. Manuf.*, vol. 2, no. 2, pp. 63–73, 1991.

[27] X. Chi, S. Liu, and C. Li, "Research on optimization of unrelated parallel machine scheduling based on ig-ts algorithm," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 70, p. e141724, 2022.

[28] W. Bożejko, A. Burduk, K.Musiał, and J. Pempera, "Neurotabu search approach to scheduling in automotive manufacturing," *Neurocomputing*, vol. 452, pp. 435–442, 2021, doi: 10.1016/j.neucom.2020.01.121.