Journal of Civil Engineering and Management
ISSN 1392-3730 / eISSN 1822-3605
2022 Volume 28 Issue 4: 247–260
https://doi.org/10.3846/jcem.2022.16383

EFFECTIVE ALLOCATION OF MANPOWER IN THE PRODUCTION OF PRECAST CONCRETE ELEMENTS WITH THE USE OF METAHEURISTICS

Michał PODOLSKI*

Faculty of Civil Engineering, Wrocław University of Science and Technology, Wrocław, Poland

Received 24 March 2021; accepted 8 December 2021

Abstract. Planning problems are particularly important for the production processes of precast reinforced concrete elements. Currently used modeling of these processes is based on the flow shop problem. Flow shop models are usually used in Enterprise Resource Planning systems, which, however, may not take into account the specifics of the production of such elements. The article presents a new model for scheduling the production of reinforced concrete prefabricated elements, which is distinguished by the possibility of carrying out activities by more than one working group. An additional new constraint is the possibility of parallel performance of some works, which may occur during their production. Also, there will be an individual order of elements assumed for each of the activities. New objective functions will be considered – the sum of idle times of working groups and the total type changes of precast components. The presented scheduling model contains an NP-hard discrete optimization problem. For this reason, metaheuristics were used in the article to solve optimization problems: the simulated annealing algorithm and the tabu search algorithm. Verification of the results obtained with the use of these algorithms confirmed their high efficiency. The application of the presented scheduling model illustrates a practical case study showing the effectiveness of the used algorithms.

Keywords: scheduling, hybrid flow shop, precast concrete production, optimization, management, metaheuristics.

Introduction

Precast reinforced concrete is an economical and functional solution for building constructions. They are more often used on many construction sites. Ready-made reinforced concrete prefabricated elements not only reduce the time of performed works, but also guarantee reliability and high quality (Bennett, 2005). Their production is usually carried out in factory conditions, which allows it to be independent from the effects of weather conditions: precipitation, wind and temperature. Due to the high repeatability of activities during the production of precast elements, these processes can be classified as industrial production. Their production processes are usually carried out by working groups consisting of qualified employees servicing the necessary equipment and using the space in the production halls. An important issue occurring during the production of such elements is the planning process.

Precast production can be divided into two categories according to the difference in production methods: flow shop production and fixed location production. The flow shop divides the precast production into activities, which are described in Section 2 of the article. Each activity is handled in a particular workstation by a particular working group. For the fixed location production, the division of precast production is similar, while all steps of a component are handled in a fixed workstation by the same or different working group. The production capacity and resource utilization rate of the fixed location production is lower than that of the flow shop production. Among these two kinds of shop floor scheduling, flow shop scheduling is of the most importance, because the flow shop is often chosen as the shop floor organization form for precast production due to their high production capacity. The method presented in the article can be used for optimizing the working group work schedule when the flow shop production method of precast components is applied.

Production planning problems are most often analyzed using various theoretical models of deterministic or random nature. One of the most important elements of production planning is striving to determine the optimal production schedule for a given production program, i.e. providing optimal start and end dates for orders based on current production capacity. It comes down to establish-

*Corresponding author. E-mail: *michal.podolski@pwr.edu.pl*

ing a detailed work schedule for work groups, machines, and material consumption schedules. In addition, not only optimal use of equipment is implemented during production, but also optimal use of the surface of production halls should be strived for. The purpose of this article is to present the problem of optimizing the working group work schedule during the implementation of a given program for the production of precast reinforced concrete elements.

In the article there will be presented a new model for scheduling the production of precast elements based on the hybrid flow shop problem (HFS problem, i.e., flow shop scheduling problem with parallel machines (Ruiz & Vázquez-Rodríguez, 2010)). In previous studies, the assumption was to perform a given type of activity only by one working group. The model presented in the article assumes the possibility of performing a given type of activity by more than one working group (in HFS scheduling problem more than one machine). The number of working groups can be individually determined for each type of activity. The basic application of HFS problem for scheduling the production of precast elements only with makespan criterion was presented in the conference paper (Podolski & Rejment, 2019). In presented article new, additional constraint and new objective functions are introduced in the prefabricated elements scheduling model, compared to conference paper (Podolski & Rejment, 2019). The important new constraint is the possibility of performing certain activities in parallel, which is set by a directed graph. This constraint models order of the activities (for instance – preparing reinforcement blanks), which are performed independently of other activities in the production process. Contrary to the previous research, the presented model assumes the possibility of changing the order of elements for various types of activities. This assumption allows us to significantly increase the number of acceptable solutions and thus, increases the possibility of achieving a schedule with a better value of the objective function. The objective functions, apart of makespan, are considered in the model: the sum of idle times of working groups and the total type changes of precast components. These objectives are minimized and were used in previous research (e.g., Yang et al., 2016), but only for the flowshop scheduling models with one working group. These three criteria create a two-criteria and three-criteria discrete optimization problems. The solution of these problems is proposed in the article. To solve discrete optimization tasks in the presented model metaheuristics will be used, such as: simulated annealing and tabu search algorithms.

In conclusion, the following new elements have been adopted in the model of production of precast elements presented in the article:

- the possibility of performing certain activities in parallel, which is set by a directed graph;
- the possibility of changing the order of elements for various types of activities;
- the application of objective functions: the sum of idle times of working groups and the total type changes of precast components in model with possibility of performing a given type of activity by more than one working group;
- the formulation of three-criteria discrete optimization problem (objective functions: the sum of idle times of working groups, the total type changes of precast components, makespan);
- the application of simulated annealing and tabu search algorithms for solving discrete optimization tasks in the presented model.

In the article, Section 1 presents a literature review of previous research on precast production scheduling problems. Section 2 examines the technology of producing precast elements. Section 3 presents the optimization model for the production of precast elements analyzed in the article. Section 4, in turn, presents the algorithms for solving optimization problems in the analyzed model. Section 5 presents verification of the results obtained with the use of the algorithms. The case study will be presented in Section 6. Results of optimization process with discussion for the case study will be presented in Section 7. Finally, the summary and conclusions will end the article.

1. Literature review

Due to the usefulness for production planning, scheduling modeling is most often used, as for problems of deterministic character. One of the most commonly used methods of production scheduling is, among others, the so-called Line of Balance (LOB) (Lumdsen, 1968) method. However, it has significant limitations, including failure to adapt to a rapidly changing production program and the size of the components themselves (Su & Lucko, 2016). It resulted in the fact that nowadays flow shop models have been most commonly used, which have their source in the achievements of the task scheduling theory. Flow shop models are usually used in Enterprise Resource Planning (ERP) systems (Huang & Yasuda, 2016; Addo-Tenkorang & Helo, 2011), which, however, may not take into account the specifics of the production of precast reinforced concrete elements.

Scheduling of this type of production was presented in many works as a discrete optimization problem with various parameters, constraints and objective functions. Approach using mathematical programming (Warszawski, 1984) has been developed for precast production planning and scheduling. The advantage of this approach is that one can obtain a satisfactory schedule with little computer effort. The disadvantage is the small size of the tasks solved by this approach due to the NP-hardness (nondeterministic polynomial time) of the solved problems. Another approach is the use of heuristic algorithms based on different priority rules (Chan & Hu, 2002a). There are simple, computationally inexpensive, less sensitive to input data disturbances methods used in production systems to quickly determine the solution. The disadvantage of these algorithms is the quality of the obtained solutions, which are distant from the optimal solutions from 25% (average) to even 500% (extreme). The practical usefulness of algorithms based on priority rules should be confirmed by detailed experimental research. These algorithms are now often used in ERP systems.

The methods based on simulation techniques were presented in the works of Dawood (1995, 1996), Dawood and Neale (1993). These are easy-to-apply techniques that require more computing power than the previously described algorithms. As a result of their use, an approximate result is also obtained. This result may be quite distant from the optimal one due to the lack of rules of "moving" through the space of solutions to a given problem, which allows us to achieve a solution close to the optimal one. This imperfection has been resolved in metaheuristic algorithms, where rules are introduced, that most often use phenomena occurring in nature, such as e.g. evolution (evolutionary algorithms, including genetic algorithms), searching for a solution to a problem by humans (tabu search algorithm) and many others (e.g., African vulture optimization algorithm (Abdollahzadeh et al., 2021a), artificial gorilla troops optimizer (Abdollahzadeh et al., 2021b), black widow optimization (Hayyolalam & Kazem, 2020). These algorithms achieve results much better than simulation techniques and therefore they are currently most often used in solving various discrete optimization problems (e.g., multi-unit construction projects optimization problems: Podolski, 2016; Podolski & Sroka, 2019), including the problems of planning the production of precast elements.

Chan and Hu (2002b) presented the flow shop model taking into account the parallel use of a limited number of devices accelerating curing and taking into account the continuous laying of the concrete mix and curing. The following objective functions were considered: minimizing the duration time, cost of storage of elements, penalties resulting from the terms of the contract with the customer. Ko and Wang (2010, 2011) introduced a limited buffer time between positions and adjusted the restrictions of the flow shop model accordingly to the situation. In order to solve the optimization problem, they used a suitably modified genetic algorithm to solve the tasks of minimizing the duration of production and penalties for delays in tasks' execution. Tharmmaphornphilas and Sareinpithak (2013) presented a model of scheduling the production of precast elements with a fixed location during the execution of processes and a heuristic solution for this model given. The model of scheduling the production of prefabricated elements on many production lines is presented in the article of Yang et al. (2016). An additional constraint was also taken into account, such as avoiding frequent changes in the type of precast element. The development of this model with new limitations and objective functions was given in the article of Ma et al. (2018). Anvari et al. (2016) presented a scheduling model based on the job shop problem, which is a special case of the flow shop problem. They used a genetic algorithm to solve a two-criteria (cost/time) optimization problem with the dominant objective function, which was the duration of the production task. The two-criteria problem of scheduling with regard to uncertainty was described by Wang et al. (2018). It consisted of seeking a compromise between two contradictory goals: delivery time for a given precast element and production costs. In order to solve optimization problems in the above articles, the genetic algorithm (Ko & Wang, 2010, 2011; Yang et al., 2016; Ma et al., 2018; Anvari et al., 2016) or its variant – the hybrid genetic algorithm (Wang et al., 2018) was most often used due to the NP-hardness of the models presented in them.

2. Production technology of the precast reinforced concrete elements

Precast reinforced concrete elements manufactured by the prefabrication company are e.g. beam elements, girders, wall elements, footings, retaining walls, floor slabs, elements of communication shafts. These elements are made stationary in metal forms, with a demountable structure. The entire production process consists of the following activities: assembly of the reinforcing blank, assembly of the mold, lubrication of the mold and assembly of the reinforcement, then, laying of the concrete mix with compaction, accelerated curing, demolding of the finished element. After unforming the previously made element, the mold is cleaned manually with pneumatic scrapers.

The reinforcement blanks, regardless of other activities, are made in the reinforcement preparation department. Then, the mold is reinstalled manually or hydraulically. After the mold is assembled, it is lubricated from the inside with an anti-adhesive agent with the use of emulsion spray. The reinforcement is installed from above. The ready reinforcement blank is transported and assembled into the mold by means of a crane with a sling. Laying the previously prepared concrete mix takes place from a hopper which is suspended on a sling attached to the crane block of a crane.

The compaction of the concrete mix is carried out using adhesive vibrators mounted on the walls of the mold. The next process is accelerated concrete curing, which is carried out by means of thermal hardening. using contact method. The heating medium is steam supplied to the mold. After the accelerated concrete curing process, the mold is disassembled manually or hydraulically, and the element is removed from the mold by lifting it with a crane and transported to the storage location. Finished elements can be transported by tractor units with a semi-trailer to the built-in location.

3. Optimization model of the production process of precast elements

The optimization model for the production of precast reinforced concrete elements considered in the article is based on the assumptions of the HFS model (Ruiz & Vázquez-Rodríguez, 2010) and it can be presented as follows:

Parameters:

The project is constituted by a set of precast components:

$$Z = \{Z_1, Z_2, Z_3, ..., Z_j, ..., Z_n\}. \quad (1)$$

For the execution of the activities in the precast production, there are teams of working groups created, each of which performs one type of activity. They form a set of

$$B = \{B_1, B_2, B_3, ..., B_k, ..., B_m\}. \quad (2)$$

In each team of working group $B_k \in B$ there is $m_k \geq 1$ of the same working groups (of the same performance and constitution)

$$B_k = \{B_{k1}, B_{k2}, B_{k3}, ..., B_{ki}, ..., B_{km_k}\}. \quad (3)$$

Each precast component $Z_j \in Z$ requires performance of m activities which constitute the set

$$O_j = \{O_{j1}, O_{j2}, O_{j3}, ..., O_{jk}, ..., O_{jm}\}. \quad (4)$$

It is assumed that activity $O_{jk} \in O_j$ can be realized by the working group $B_{ki} \subset B_k$. Activity duration time O_{jk} performed by the group B_{ki} equals $p_{jk} > 0$. The set of duration of p_j activities from the set O_j is defined by vector

$$p_j = \{p_{j1}, p_{j2}, p_{j3}, ..., p_{jk}, ..., p_{jm}\}. \quad (5)$$

The sequence dependence of O_j activities for precast component Z_j are given on the example of directed, acyclic graph $K = (M, F)$. M is a set of nodes representing activities O_j of precast component Z_j ($|M| = mn$). F is a set of arcs. The arcs are use to show the precedence relationships that exist between the activities in precast components.

Constraints:

The order of execution of the activities resulting from activity technology is assumed such that: $O_{j,k-1} \prec O_{j,k} \prec O_{j,k+1}$. It is assumed that activity $O_{jk} \in O_j$ is performed without any stops by one working group B_{ki} from B_k team in $p_{jk} > 0$ time. It is assumed that the activity $O_{jk} \in O_j$ is performed continuously by the working group B_{ki} in time $p_{jk} > 0$.

Decisive variable:

For rigorous explanation of the decisive variable there are introduced two notions: batches and precast components processing order. A batch is a subset of precast components assigned to working group B_{ki} from the team of working groups $B_k \subset B$; due to interchangeable working group identity, a batch is not associated with any particular working group. In each team of working groups $B_k \subset B$ the set of precast components Z has to be partitioned into m_k batches $Z_{ki} \subset Z$, $n_{ki} = |Z_{ki}|$. The precast components processing order from batch Z_{ki} can be expressed by a permutation $\pi_{ki} = (\pi_{ki}(1), \pi_{ki}(2), ...\ \pi_{ki}(l), ..., \pi_{ki}(n_{ki}))$, where $\pi_{ki}(l)$ denotes the element of Z_{ki} which is in position l in π_{ki}. Thus, the precast components processing order for each team of working groups $B_k \subset B$ can be determined by the set of m_k permutations:

$$\pi_k = (\pi_{k1}, \pi_{k2}, ..., \pi_{ki}, ..., \pi_{km_k}), \quad (6)$$

each permutation for one batch. The set of m_k permutations π_k determines performing of the activity k by the team of working group B_k in all precast components. Permutation π_{ki} determines the processing order of precast components by working group B_{ki} from the team of working groups $B_k \subset B$ of size m_k, n_{ki} determines the number of precast components assigned to working group B_{ki}. Decisive variable is precast components processing order, which is defined by m-tuple:

$$\pi = (\pi_1, \pi_2, ..., \pi_k, ..., \pi_m). \quad (7)$$

In the presented model of the precast production the task is to find a schedule (S, A), (where $S = [S_{jk}]_{n \times m}$, $A = [a_{jk}]_{n \times m}$ and S_{jk} defines earliest start time for carrying out activities in precast component Z_j by team of working groups B_k and a_{jk} defines number of working group allocated to carrying out activities for precast component Z_j) in order to minimize or maximize value of objective function, while meeting the accepted constraints.

The first criterion (objective function) is a term $C_{\max}(\pi)$ of implementation of all activities in all precast components (makespan):

$$C_{\max}(\pi) = \max_j \{C_{m,j}\}. \quad (8)$$

The makespan value associated with π will be denoted by $C_{\max}(\pi)$. Optimization task in the model is to find a schedule for the activity realization which minimizes the value of the objective function $C_{\max}(\pi)$, satisfying the constraints given above. The considered model can be represented in the form of a disjunctive graph. Any graph for the present model has the property of the critical path of length $C_{\max}(\pi)$. The earliest finish time for activities can be determined from the recursive formula:

$$C_{k,\pi_{ki}(l)} = \max\left\{C_{k,\pi_{ki}(l-1)}, \max_f \left\{C_{f,\pi_{ki}(l)}\right\}\right\} + p_{k,\pi_{ki}(l)}, \quad (9)$$

where: $j = 1, ..., n$, $l = 1, ..., n_{ki}$, $i = 1, ..., m_k$, $k = 1, ..., m$, $\pi_{ki}(0) = 0$, $C_{k,0} = 0$, $C_{0,j} = 0$, $f = 1, ..., m_{pred}$. Variable m_{pred} defines the number of predecessors of k work, whereas the variable f denotes the number of works on the list of predecessors of k works. For given π, the earliest finish times for all activities can be found in time $O(nm)$ using recursive Eqn (9). In the scheduling theory presented model is a kind of flow shop problem with parallel machines (i.e., HFS problem (Ruiz & Vázquez-Rodríguez, 2010)) with the $C_{\max}$ criterion. In literature this problem is strongly NP-hard (non-deterministic polynomial-time hardness).

In the formulated model it is possible to distinguish the second optimization criterion, which is important for maintaining the continuity of work of working groups. Such a criterion is the sum of idle times of working groups $D(\pi)$:

$$D(\pi) = \sum_{k=1}^{m}(\sum_{i=1}^{m_k}(\sum_{l=2}^{n_{ki}}(C_{k,\pi_{ki}(l)} - p_{k,\pi_{ki}(l)} - C_{k,\pi_{ki}(l-1)}))). \quad (10)$$

The schedule desired from the viewpoint of the manufacturer of precast elements is one that meets the condition of minimizing its duration and at the same time there are minimally possible idle times of working groups. Therefore, it is possible to consider a two-criteria approach in the model. It involves the simultaneous minimization of both independent criteria: the duration of the schedule and the sum of idle times of working groups in the work. Due to the fact that both criteria do not converge, the solution to this problem is a set of optimal (non-dominated, effective) Pareto solutions. One of the Pareto optimal solutions s, being an acceptable solution, must meet the condition that for the opposite criteria F_1 and F_2 there is no other solution q such that the following inequalities are met: $F_1(q) \leq F_1(s)$ and $F_2(q) \leq F_2(s)$ and at least one of these inequalities is sharp. The set P will be called the Pareto set if it is a set of optimal Pareto solutions such that it does not contain two solutions $s, q \in P$ with the values $F_1(q) = F_1(s)$ and $F_2(q) = F_2(s)$. Point $(F_1(s), F_2(s))$ will be called a compromise point in the space of criterion functions F_1 and F_2, if $s \in P$. The set of all compromise points between criteria F_1 and F_2 will be marked by K. In the model under consideration, criteria F_1 and F_2 are: the sum of idle times of working groups $D(\pi)$ (Eqn (10)) and the total duration of the schedule (makespan) $C_{\max}(\pi)$ (Eqn (8)).

The set of optimal Pareto solutions s will be marked for the considered model by P, while the set of all compromise points between the criteria by K. In the presented production scheduling model, a part of the set of compromise points K corresponding to the idle time threshold values $D(\pi)$ was determined using the following algorithm for determining part of the set of compromise points K:

– **Step 1.** Determine possible minimum $D_{\min}$ and maximum $D_{\max}$ total working group idle time.
– **Step 2.** Determine the set of thresholds values for the total idle time of working groups, which will be between the values of $D_{\min}$ and $D_{\max}$.
– **Step 3.** For the values of $D_{\min}$ and $D_{\max}$ and the set thresholds values of the total idle time of working groups, find the minimum duration of the entire project corresponding to these idle time values, using the model with the criterion of the duration of the entire project with the assumed limit on the total idle time of working groups.

Threshold values of total idle time of working groups are understood as such idle time values that were set individually by the element planner. In the extreme case, with a sufficiently high density of idle time threshold values, we can get al. or almost all compromise points from the K set. The presented algorithm for determining compromise points has a useful value for people planning the production of precast elements in the company, who will receive information about the form of idle time/makespan for the current production program and will take appropriate actions regarding its planning.

In the formulated model it is possible to distinguish the third optimization criterion – the total type changes of precast components $H(\pi)$:

$$H(\pi) = \sum_{k=1}^{m}(\sum_{i=1}^{m_k}(\sum_{l=2}^{n_{ki}}(h_{k,\pi_{ki}(l)}))), \tag{11}$$

where:

$$h_{k,\pi_{ki}(l)} = \begin{cases} 0, \text{ if } \pi_{ki}(l) - \pi_{ki}(l-1) = 0 \\ 1, \text{ if } \pi_{ki}(l) - \pi_{ki}(l-1) \quad 0 \end{cases}, \tag{12}$$

$\pi_{ki}(l)$ – determines the processing order of precast components by working group B_{ki} using numbering with the rule: one type of precast component has only one number. It is important to keep the type changes to the minimum, because frequent type change causes unnecessary equipment adjustments and worsen efficiency of working groups and quality of precast components.

The schedule desired from the viewpoint of the manufacturer of precast elements is one that meets the condition of minimizing at the same time its duration, possible idle times of working groups and the total type changes of precast components. Therefore, it is possible to consider a three-criteria approach in the model. It involves the simultaneous minimization of independent criteria: the duration of the schedule and the sum of idle times of working groups in the work and the total type changes of precast components. Due to the fact that these criteria do not converge, the solution to this problem is also a set of optimal Pareto solutions. To solve this problem there is introduced the single objective function $R(\pi)$:

$$R(\pi) = w_1\left(\frac{D(\pi)}{D^*(\pi)}\right) + w_2\left(H(\pi) - H^*(\pi)\right), \tag{13}$$

where w_1 and w_2 are positive weights ($w_1 + w_2 = 1$); $D(\pi)$ is the sum of idle times of working groups; $H(\pi)$ is the total type changes of precast components; $D^*(\pi)$ is the fittest value for $D(\pi)$; $H^*(\pi)$ is the fittest value for $H(\pi)$. The single objective function $R(\pi)$ is introduced to transfer two criteria $D(\pi)$ and $H(\pi)$ to a single criterion by weighted sum approach (Coello et al., 2007). The values of criteria $D(\pi)$ and $H(\pi)$ are distributed in a different range. Therefore, they should be normalized before the weighted-sum operation using fittest values $D^*(\pi)$ and $H^*(\pi)$ (Cochran et al., 2003). The weights w_1 and w_2 are adopted by the manufacturer of precast elements. The value of the weights depends on the importance of the given criterion for the manufacturer. The three-criteria problem (criteria $C_{\max}(\pi)$, $D(\pi)$ and $H(\pi)$) can be transferred to two-criteria problem thanks to the use of the single objective function $R(\pi)$. Due to the fact that both criteria $C_{\max}(\pi)$ and $R(\pi)$ do not converge, the solution to this problem is a set of optimal Pareto solutions too. The problem of getting the set of all compromise points between the criteria $C_{\max}(\pi)$ and $R(\pi)$ in article will be analogous to the solution of two-criteria problem $C_{\max}(\pi)$ and $D(\pi)$, which is described above.

4. Metaheuristic algorithms for solving optimization problems in the analyzed model

The problem presented in Section 3 is a strongly NP-hard discrete optimization problem. This means that the optimal solution can be found in the optimization tasks of the presented model only with the use of algorithms whose calculation time is increasing exponentially with the increase in the size of the task, e.g. algorithms based on the branch and bound method or dynamic search. Such time is too long for practical applications of the presented optimization model.

Therefore, currently metaheuristic algorithms are most often used to solve them. The algorithms provide solutions close to optimal, which in practical applications are fully satisfying. The most commonly used metaheuristic algorithms are: evolutionary (e.g., genetic), hybrid (e.g., memetic), simulated annealing, tabu search, ant or particle swarm algorithms. In order to find a solution to the optimization task in the production model presented in Section 3, the simulated annealing algorithm and the tabu search algorithm will be used.

The first of them is the simulated annealing (SA) algorithm, which was proposed in Kirkpatrick's work (Kirkpatrick et al., 1983). This algorithm uses, analogous to the thermodynamic process of cooling, solid in order to introduce the trajectory of the search of the local extremum. States of solid matter are seen analogously as individual solutions to the problem, whereas the energy of the body as the value of the objective function. During the physical process of cooling the temperature is reduced slowly in order to maintain energy balance. The SA algorithm starts with the initial solution, usually chosen at random. Then, in each iteration, according to established rules or randomly, there is solution π' selected from the base neighbourhood $N(\pi)$. It becomes the base solution in the next iteration, if the value of the objective function is better than the current base solution or if it otherwise may become the base solution with the probability of: $p = \exp(-\Delta/T_i)$, where $\Delta = c(\pi') - c(\pi)$, T_i – the temperature of the current iteration i, c – the objective function. In each iteration there are m draws from the neighbourhood of the current basic solution performed. The parameter called the temperature decreases in the same way as in the natural process of annealing. The most frequently adopted patterns of cooling are: geometrical ($T_{i+1} = \lambda_i\, T_i$) or logarithmic ($T_{i+1} = T_i\, /\, (1+ \lambda_i\, T_i)$), where $i = 0, ..., N - 1$, T_0 – initial temperature, T_N – final temperature, N – number of iterations, λ_i – parameter. In the algorithm there are usually parameter values T_0, T_N, N adopted and parameter λ_i is calculated. The relationship $T_0 > T_N$ should take place, whereas T_N should be small, close to zero. Below, there is presented a general method of SA algorithm used to solve the flow shop problem.

Step 0. Determine the initial solution $\pi^0 \in \Pi$. Substitute $\pi_{SA} = \pi^0$, $k = 0$, $T = T_0$.

Step 1. Perform steps 1.1–1.3 x-times.

Step 1.1. Substitute $k := k + 1$. Choose random $\pi \in N(V, \pi_{k-1})$.

Step 1.2. If $c(\pi) < c(\pi_{SA})$ then substitute $\pi_{SA} = \pi$.

Step 1.3. If $c(\pi) < c(\pi_{k-1})$ then substitute $\pi_k = \pi$. Otherwise, accept solution π with a probability of $p = \exp((c(\pi_{k-1}) - c(\pi))/T$, i.e. $\pi_k = \pi$, if solution π was not accepted.

Step 2. Change the temperature T according to a defined pattern of cooling.

Step 3. If $T > T_N$, return to step 1, otherwise STOP.

SA algorithms are used to solve many optimization problems, including flow shop problems considered in the context of discrete optimization problems (e.g., Ogbu & Smith, 1990; Ishibuchi et al., 1995). The SA algorithm used in the article works as follows. Algorithm starts with the initial solution obtained randomly (precast components processing order π). Neighbourhood $N(\pi)$ contains precast components processing orders generated from π with the use of "insertion" move. Boltzmann function of acceptance was adopted. Geometric cooling scheme was adopted, i.e., $T_{i+1} = \lambda T_i$ and $T_0 = 60$, $\lambda = 0.99$. The number of considered solutions at a set temperature is $0.5n$. Maximum number of iterations of the algorithm SA is 10000.

The second algorithm used in the article is the tabu search algorithm (abbreviated as TS) (Glover & Laguna, 1995). It replicates the natural searching process to find solutions to problems posed by men. Basic version of the TS algorithm starts its working with a particular startup solution. Then, for this solution, there is neighborhood found. It is defined as a set of solutions that can be created after performing of moves in a given solution, i.e., transformation of a given solution into another one according to established rules. The solution with the smallest value of the objective function is sought in the neighborhood. This solution is a base solution for the next iteration. The result of the algorithm is the best solution of the whole search trajectory. Below there is a general algorithm of TS method used to solve flow shop problems in scheduling theory:

Let $\pi \in \Pi$ be any permutation, LT – tabu list, c – adopted objective function, whereas π^* – the best solution found so far (in the beginning we adopt permutation π for π^*).

Step 1. Determine neighborhood N_π of permutation π which does not contain a list of items prohibited by LT;

Step 2. Find permutation $\delta \in N_\pi$ such that:
$c(\delta) = \min\{c(\beta) : \beta \in N_\pi\}$;

Step 3. If $c(\delta) < c(\pi^*)$, then $\pi^* \leftarrow \delta$;
Put attributes δ on LT list;
$\pi \leftarrow \delta$;

Step 4. If Finish_Condition, then STOP
otherwise go to Step 1.

Tabu search method has many degrees of freedom: the choice of a move and determining the neighborhood, forms implementing the mechanism of tabu (e.g., the length of the tabu list, form of attributes), search strategy. Currently TS algorithm is one of the most effective tools used in the scheduling theory. Its form used to solve the optimization problem in the presented model is based on the article (Nowicki & Smutnicki, 1998). TS algorithm used in the presented model starts with the initial solution. It is assumed that initial solution (precast components processing order) is obtained randomly. It is assumed that move is of "insertion" type. The insert-type move operates on set π_k (7) and removes a job placed at a position in this set and inserts it in another position of this set. The neighborhood N_π consists of precast components processing orders π (6) with the set $\pi_k{}^v$ generated by all possible "insertion" moves in a given randomly chosen set π_k. The next basic element in presented TS algorithm is the tabu list *LT*. The tabu list *LT* is a list including attributes of moves of recently examined solutions which are described in article (Nowicki & Smutnicki, 1998). The only parameter of tabu list is its length (the number of its elements). In the presented TS algorithm the length of tabu list equals 1/3*nm*. The completion condition is as following: the TS algorithm stops after 10000 iterations. Programming implementation of SA and TS algorithms for the considered model was made by author of the article in Mathematica environment.

5. Verification of the results obtained using the SA and TS algorithms

For the presented above form of the SA and TS algorithms, the author evaluated the results which were obtained with them. For this purpose, the dedicated software in the Mathematica system has been created for the model of the problem presented in the paper. The examples from article (Wittrock, 1988) were used to verify the obtained results. They are related to the practical problems of the task scheduling of production line in a company that produces printed circuit boards. This company accepts six orders a day to produce a certain amount of different printed circuit boards. The assembly of each board is a task that always consists of three operations and is performed on three types of machines. Each of the three operations can be performed by a number of machines working in parallel. The problem to solve is to find the order of assembly of boards included in the daily order, so that the duration of work is as short as possible. These orders are test examples that can be modelled as a hybrid problem. The sizes of these examples are the following: $n \times m$ = 51×3 ("day 1"), 38×3 ("day 2"), 38×3 ("day 3"), 36×3 ("day 4"), 40×3 ("day 5"), 30×3 ("day 6"). Each of the considered test example was solved seven times. Then the arithmetic mean of these calculations was related to the value of the lower bound of objective function (LBC_{max}), given for these examples in article (Wittrock, 1988), by calculating for each example the percentage relative deviation *PRD*(SA) of the SA algorithm and the percentage relative deviation *PRD*(TS) of the TS algorithm:

$$PRD(\mathrm{SA}) = 100\%(C^{\mathrm{SA}} - LBC_{max}) \,/\, LBC_{max}; \quad (14)$$

$$PRD(\mathrm{TS}) = 100\%(C^{\mathrm{TS}} - LBC_{max}) \,/\, LBC_{max}, \quad (15)$$

where: C^{SA} – the average value of the objective function obtained by the SA algorithm, C^{TS} – the average value of the objective function obtained by the TS algorithm, LBC_{max} – the value of the estimation of the lower bound of the objective function from article (Wittrock, 1988). In order to compare the quality of the obtained results, the relative deviations of the heuristic W algorithm were presented as proposed in article (Wittrock, 1988) and the TSAB algorithm, which is an advanced version of the metaheuristic tabu search algorithm given in article by Nowicki and Smutnicki (1998). The results of calculation of the SA and TS algorithms for the test examples from article of Wittrock (1988) are presented in Table 1.

The presented above results of verification calculations of the SA and TS algorithms confirm the great usefulness of these algorithms for solving optimization tasks. The SA algorithm provided the vast majority of better results than the W algorithm (for five out of six tested examples) and worse for the vast majority of examples than the

Table 1. The results of verification calculations of SA and TS algorithms for the test examples (table partially reproduced from paper (Podolski & Rejment, 2019), copyright by IOP Publishing Ltd, license CC BY 3.0 https://creativecommons.org/licenses/by/3.0/)

Notation of the example acc. to Wittrock (1988)	Average value C^{SA}	Average value C^{TS}	LBC_{max} acc. to Wittrock (1988)	C^{W} acc. to Wittrock (1988) – algorithm W	C^{TSAB} acc. to Nowicki and Smutnicki (1998) – algorithm TSAB	*PRD*(SA) [%]	*PRD*(TS) [%]	*PRD*(W) [%]	*PRD*(TSAB) [%]
"day1"	784.43	783.57	720	784	765	8.95	8.83	8.89	6.25
"day2"	780.86	783.71	715	789	753	9.21	9.61	10.35	5.31
"day3"	784.00	784.57	694	785	760	12.97	13.05	13.11	9.51
"day4"	792.57	786.29	694	796	761	14.20	13.30	14.70	9.65
"day5"	963.00	967.43	963	964	963	0.00	0.46	0.10	0.00
"day6"	684.86	680.43	584	686	661	17.27	16.51	17.47	13.18
mean *PRD*(SA) [%]:						10.43			
mean *PRD*(TS) [%]:							10.29		
mean *PRD*(W) [%]:								10.77	
mean *PRD*(TSAB) [%]:									7.32

advanced TSAB algorithm. The quality of the solutions obtained with the SA algorithm results from the adopted controlling parameters of this algorithm: the method of transforming the base solution into disturbed (type of the so-called move) scheme and speed of the cooling, number of considered solutions at a set temperature, maximum number of iterations, which was presented in section 4. These parameters were selected so as to obtain solutions close to the optimal (satisfactory) in the acceptable time of the algorithm run. For example, if cooling is too slow, the algorithm runtime may become unacceptably long. If cooling is too fast, the SA algorithm may provide far from optimal results. The results provided by this algorithm in comparison to the results of the TSAB algorithm indicate that these are the results of the so-called local extremes (minimums) of the tested objective function to some extent distant from the global minimum.

The TS algorithm also provided the vast majority of better results than the W algorithm (for five out of six tested examples) and worse for all examples than the advanced TSAB algorithm. The mean deviation PRD calculated for the TS algorithm is lower than the mean deviation PRD calculated for the SA algorithm by 0.14 percentage points, which indicates the possibility of providing better results than the SA algorithm. This is due to the construction of the TS algorithm, which, in the course of its operation, searches the entire neighborhood of the base solution thoroughly, and does not select random solutions as in the SA algorithm. However, this significantly extends the run time of the TS algorithm compared to the run time of the SA algorithm. As in the case of the SA algorithm, the quality of the solutions obtained with the TS algorithm results from the adopted control parameters of this algorithm: the method of determining the neighborhood of the base solution (type of move), the form of attributes of solutions in the neighborhood, the length of the tabu list, the maximum number of iterations. The worse results obtained with this algorithm compared to the results of the TSAB algorithm indicate the possibility of obtaining approximate solutions to some extent distant from the global minimum. The TS algorithm in the form presented in the article may tend to look for local optimum in the so-called big valleys of solutions and not leaving them by the time the algorithm runs. The escape of the search trajectory from such valleys would allow for the achievement of better results, closer to the optimal ones.

An important problem for the tested algorithms is the comparison of their search time of optimal solutions. During the experimental analysis, the following numbers of iterations were set: SA algorithm – 10000 iterations, TS algorithm – 10000 iterations. Exemplary times of searching for optimal solutions using the SA algorithm ranged from approximately 1 minute ($n \times m = 5 \times 5$) to approximately 18 minutes ($n \times m = 25 \times 5$). For the TS algorithm, the times of searching for optimal solutions were about 6 times longer (for computation purposes the author used: Intel Core i5-4440, 4GB RAM, OS Windows 10). From these data it can be concluded that the SA algorithm is more computationally effective than the TS algorithm. The short search time of the SA algorithm results from the fact that random searches performed in the neighborhood of a given solution are the basis of this algorithm. Generating a random solution from the neighborhood is less time-consuming than a direct research of the entire neighborhood in the TS algorithm.

6. Case study

In the precast plant, a set of n = 11 precast reinforced concrete elements is planned. This set includes 5 types of elements: precast component No. 1 – 2 pieces, precast component No. 2 – 3 pieces, precast component No. 3 – 3 pieces, precast component No. 4 – 2 pieces, precast component No. 5 – 1 piece. Each of the prefabricated elements requires six different activities: A – demolding of the previous element, B – assembly of the reinforcement blank, C – assembly of the mold, D – applying anti-adhesive agent to mould and reinforcement assembly, E – laying the concrete mix with compaction, F – thermal curing. Activities A, B, C, D, E are performed by teams of working groups that are specialized to perform only one type of activity. Activity F, i.e., thermal curing, is carried out without the direct participation of working groups and for each element it lasts 1005 minutes (16.75 hours). The most important problem for the organization of work in the precast plant is the creation of an appropriate work schedule of the teams of working groups performing the first five activities in one work shift. Therefore, the optimization task was limited to only the first five activities. The order of activities A, B, C, D, E results from the network plan presented in Figure 1.

At the workplace, during the work shift, 2 working groups are available for each of the five activities. Based on the labor intensity of the activities and the composition and performance of the working groups, the duration of activities for n = 11 precast elements were determined, which are given in Table 2.

The initial (reference) solution was adopted with the assumption of making the elements in the order of their numbering (numbers 1 and 2 were adopted for two elements No. 1, numbers 3, 4, 5 were adopted for three elements No. 2, numbers 6, 7, 8 were adopted for three elements No. 3, numbers 9 and number 10 were adopted for two elements No. 4, number 11 was adopted for one element No. 5), i.e., for the following decision variable $\pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$, where:

π_1= (((1,2,3,4,5), (6,7,8,9,10,11));

π_2= (((1,2,3,4,5), (6,7,8,9,10,11));

π_3= (((1,2,3,4,5), (6,7,8,9,10,11));

π_4= (((1,2,3,4,5), (6,7,8,9,10,11));

π_5= (((1,2,3,4,5), (6,7,8,9,10,11)).

In the above decision variable, it is assumed that for activities A, B, C, D, E, the first working groups from the given teams will perform their type of activities for the

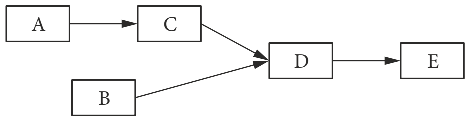

Figure 1. Sequential relationships between activities for precast reinforced concrete elements

Table 2. Duration of activities performed by working groups expressed (in hours)

Element type number	No. 1	No. 2	No. 3	No. 4	No. 5
Quantity of elements of a given type	2	3	3	2	1
A. Demolding of the previous element	0.3	0.3	0.2	1.2	1.5
B. Installation of the reinforcement blank	1.2	0.8	1.0	1.4	2.4
C. Mold assembly	0.9	0.2	0.3	0.8	1.5
D. Applying anti-adhesive agent to mould and reinforcement assembly	0.6	0.5	0.5	1.1	1.8
E. Laying of concrete mix with compaction	0.5	0.3	0.4	0.7	1.6

first five elements (more specifically, two pieces of No. 1 element and then three pieces of No. 2 element). The second working groups from the given teams will perform their type of activities for the other six elements (more precisely, three pieces of No. 3 element, then two pieces of No. 4 element and then one element of No. 5 element). If the element numbering method is used in accordance with the Table 2, the form of the above decision variable will be as follows: $\pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$, where:

π_1= (((1,1,2,2,2), (3,3,3,4,4,5));

π_2= (((1,1,2,2,2), (3,3,3,4,4,5));

π_3= (((1,1,2,2,2), (3,3,3,4,4,5));

π_4= (((1,1,2,2,2), (3,3,3,4,4,5));

π_5= (((1,1,2,2,2), (3,3,3,4,4,5))).

The production schedule implementation time n = 11 precast elements for the adopted decision variable π is C_{max} = 11.6 hours.

In addition, the total time of breaks in the work of individual working groups and the total type changes of precast components were calculated in the received schedule. After performing the operation of moving non-critical activities to critical activities within the existing stocks of non-critical activities, a schedule was received in which the total time of breaks in the work of working groups was D = 0 hours. This means that all working groups in the received schedule can work without unnecessary idle time. The total type changes of precast components H was 15 (minimum total type changes of precast components are H = 15 too, because total type changes for each activity for the initial solution is 3 and total type changes for five activities is $3 \times 5 = 15$). The next step is to calculate the value of single objective function $R(\pi)$ for the initial solution. The values of functions $D^*(\pi)$ and $H^*(\pi)$ were adopted so that the values of the quotient $D(\pi)/D^*(\pi)$ and the difference $H(\pi) - H^*(\pi)$ were minimally 0 and the minimal change of values of this expression was 1. The value of $D^*(\pi)$ was adopted as 0.1h and the value of $H^*(\pi)$ was adopted as 15 (minimum total type changes of precast components). In this way, the objective function $R(\pi)$ can be normalized to a stable level. The values of weights w_1 and w_2 were adopted both as 0.5, because it was assumed that the minimization of both functions $D(\pi)$ and $H(\pi)$ is equally important for the person planning the production of precast reinforced concrete elements. Assuming the above, the value of function $R(\pi)$ for the initial solution is 0 (minimum). The work schedule of the teams of working groups for the initial solution is presented in Figure 2.

7. Results of optimization process for the case study and discussion

Using dedicated software created in the Mathematica environment, the optimal solution was sought in the presented calculation example taking into account the criterion of minimizing the duration of activities of a set of precast elements. By performing three tests using the SA algorithm, a schedule was obtained for which the duration of all activities C_{max} = 8.6 hours and the sum of idle times D = 1.1 hours. Then, by performing three tests using the TS algorithm, a schedule was obtained for which C_{max} = 7.6 hours and the sum of idle times D = 1.0 hour. The obtained result with the use of the TS algorithm is better than the result obtained by the SA algorithm by 11.6%. This is due to the principle of operation of the SA algorithm, which searches randomly a much smaller space of neighborhood than the TS algorithm, which directly searches the entire neighborhood of the base solution. The obtained result with the use of the TS algorithm better by 34.5% than the result obtained for the reference (initial) solution. The decision variable obtained using the TS algorithm and numbering method in accordance with the Table 2 is $\pi = \pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$, where:

π_1 = ((5,2), (3,2,1,2,4,3,3,4,1));

π_2 = ((5,4,1,3,2), (2,3,3,4,1,2));

π_3 = ((2,2,3,3,3,1,1), (5,2,4,4));

π_4 = ((5,4,1,2), (2,3,3,4,1,3,2));

π_5 = ((4,1,3,4), (3,2,3,5,1,2,3)).

In further research on the presented example, only the TS algorithm will be used. This algorithm was chosen due to the possibility of achieving better results than in case of using the SA algorithm, which results from the verification analysis presented in section 5. Additionally, the values of the $H(\pi)$ function and the $R(\pi)$ function were determined for the obtained solution: H = 36, R = 15.5. The values of the $D(\pi)$, $H(\pi)$ and $R(\pi)$ functions for the schedule obtained with the TS algorithm are quite remote from the minimum values due to $D(\pi)$, $H(\pi)$ and $R(\pi)$ functions and makespan function $C_{max}(\pi)$ do not converge.

The found minimum $C_{max}(\pi)$ is in the area of solutions far from the optimum (minimum) of the other calculated functions. This problem is analogous to the phenomenon of time-cost tradeoff known in project scheduling. Time-cost tradeoff involves accelerated activity durations that are obtained by allocating more resources, and lead to shorter project duration and lower indirect cost at the expense of higher direct cost.

An important problem for the work of working groups in the above solution is the existence of idle time in their work (the sum of idle times D = 1.0 hour). The best possible solution for organizing the work of working groups will be a schedule in which the sum of idle times will be as low as possible, i.e., D = 0 hours. Therefore, it was decided to modify the presented model by introducing an additional restriction in searching the set of all solutions for acceptable solutions, for which the sum of idle times D = 0 hours. This restriction was introduced into existing software using the TS algorithm. Then, by performing three tests using this algorithm, a schedule was obtained for which C_{max} = 8.0 hours, and the sum of idle times D = 0 hours. This result is 5.3% worse than the result obtained without imposing an additional limit on the value of the total idle time in the schedule, but 31.0% better than the result obtained for the reference (initial) solution. The reason for this is that the function $D(\pi)$, and makespan function $C_{max}(\pi)$ do not converge. The decision variable obtained using the TS algorithm with the limitation of the sum of idle times to the value of D = 0 hours is $\pi = \pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$ (using numbering method in accordance with the Table 2), where:

π_1 = ((2,2,1,3,3), (5,2,3,1,4,4));
π_2 = ((5,1,4,3), (4,1,2,2,3,3,2));
π_3 = ((5,3), (1,3,3,1,2,2,4,2,4));
π_4 = ((1,5,2,4,3,2), (1,2,4,3,3));
π_5 = ((2,1,2,1,3,4,3), (5,4,3,2)).

Additionally, the values of the $H(\pi)$ function and the $R(\pi)$ function were determined for the obtained solution: H = 37, R = 11.0. The work schedules of the working group teams for both of the above-mentioned decision variables are presented in Figure 3 and Figure 4.

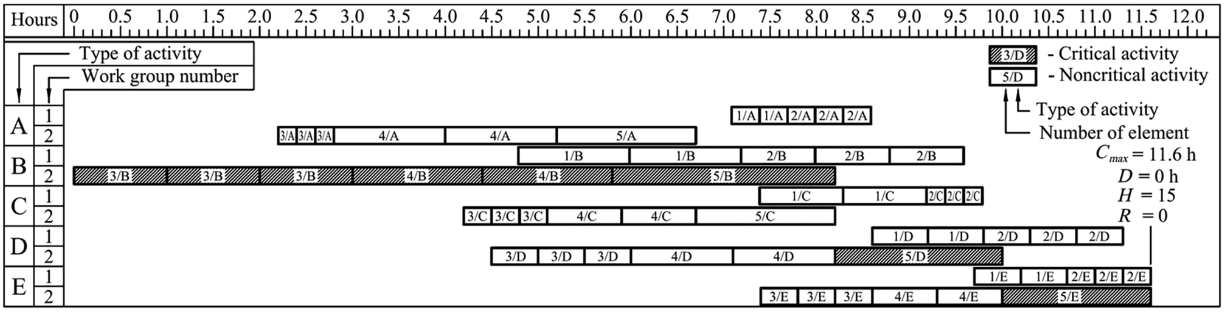

Figure 2. Working groups' work schedule for the initial solution

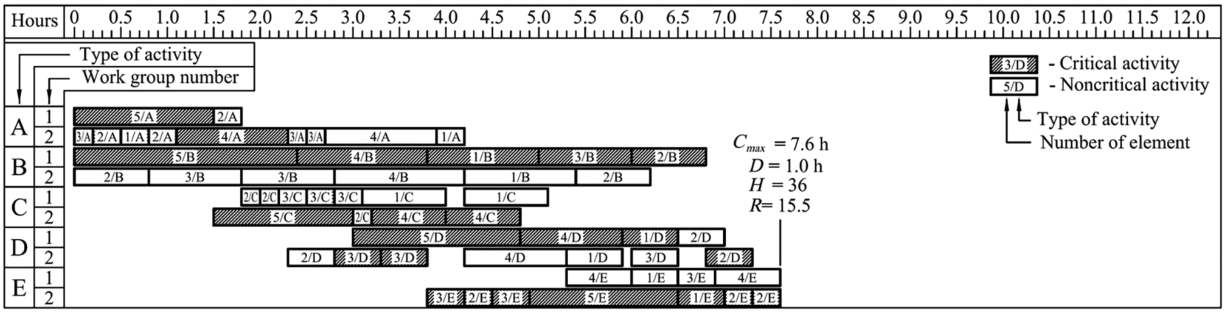

Figure 3. Work schedule of workgroup teams for a suboptimal solution without taking into account the limitation of the sum of idle times

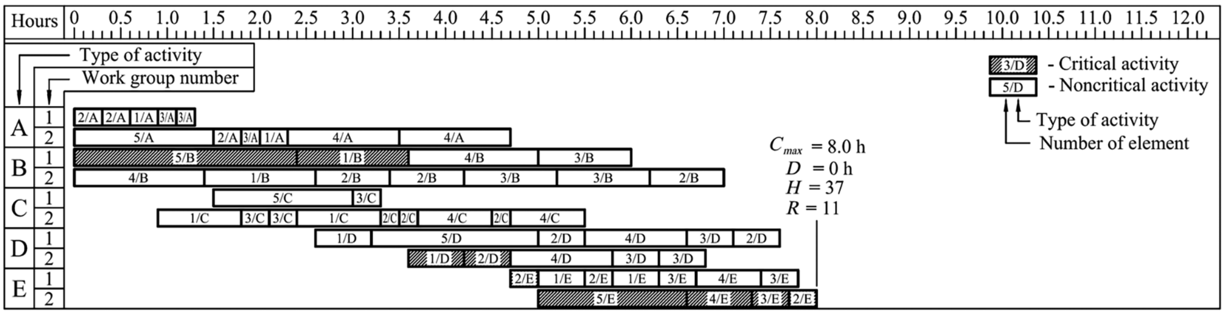

Figure 4. Work schedule of working group teams for a suboptimal solution including the limit of the sum of idle times D = 0 hours

The next step in the subject calculation was to determine a set of non-dominated solutions (a set of Pareto-optimal solutions) for the objective function under consideration – the duration of all $C_{\max}$ activities and the sum of idle times D using the algorithm for determining part of the set of compromise points K presented in section 3. This set contains two extreme points, i.e., the previously found schedule for which the minimum duration of all activities $C_{\max}$ = 8.0 hours for the sum of idle times assumed $D_{\min}$ = 0 hours and the schedule for which the minimum duration of all activities $C_{\max}$ = 7.6 hours with the sum of idle times $D_{\max}$ = 1.0 hour.

The remaining points of the set of Pareto-optimal solutions were determined by solving the tasks of optimizing single-criteria minimization of the duration of the $C_{\max}$ elements for the threshold values limiting the sum of idle times D. Threshold values D were established between the values $D_{\min}$ = 0 hours and $D_{\max}$ = 1.0 hour, assuming subsequent D values every 0.1 hours. The result of searching for suboptimal points of the set of Pareto-optimal solutions (makespan/sum of idle times trade off) obtained using the TS algorithm is presented in Figure 5. Obtaining the minimum value of the function $D(\pi)$ increases the value of the $C_{\max}(\pi)$ function. This is due to the fact that the production schedule of precast elements with the function value $D(\pi) = 0$ is not in the same area of solutions with the minimal $C_{max}(\pi)$.

The next step is to consider a three-criteria approach in the case study. It involves the simultaneous minimization of independent criteria: the duration of the schedule and the sum of idle times of working groups in the work and the total type changes of precast components. Due to the fact that these criteria do not converge, the solution to this problem is also a set of optimal Pareto solutions. To solve this problem there is introduced the single objective function $R(\pi)$, which is described in the Section 3. The set of all compromise points between the criteria $C_{\max}(\pi)$ and $R(\pi)$ was determined using the algorithm, which is described in the Section 3. The possible minimum $R_{\min}$ is 0. The maximum $R_{\min}$ was determined as 20. The values R were established between the values $R_{\min}$ = 0 and $R_{\max}$ = 20, assuming subsequent R values every 1. The result of searching for suboptimal points of the set of Pareto-optimal solutions (makespan/function $R(\pi)$ trade off) obtained using the TS algorithm is presented in Figure 6. In this figure it can be noticed, that the solutions with the minimum value of the function $C_{\max}(\pi)$ do not coincide with the minimum values of the function $R(\pi)$. This is due to the fact that the area of schedules with the minimum $R(\pi)$ is not in the area of solutions with the minimum value of the function $C_{\max}(\pi)$. $R(\pi)$ and $C_{\max}(\pi)$ functions do not converge. In addition, it can be seen that it was possible to obtain the minimum values of the $C_{\max}(\pi)$ function from the value of R = 8, which translates into low values of the total type changes of precast components $H(\pi)$ and the sum of idle times $D(\pi)$.

The last step is to indicate the schedule, which was chosen by the manufacturing planner. In the factory is used eight-hour working day. The eight-hour working day should be treated as an additional constraint in the optimization model. The solution with minimal function $R(\pi)$ from the set Pareto-optimal solutions (makespan/function $R(\pi)$ trade off), which take into account above constraint is $\pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$ (using numbering method in accordance with the Table 2), where:

π_1 = ((1,1,4,4), (5,2,2,3,3,3,2));

π_2 = ((5,3,3,1,1), (4,2,3,2,2,4));

π_3 = ((3,3,3,2,4,2,1), (5,1,2,4));

π_4 = ((3,3,2,2,2,3,1,1), (5,4,4));

π_5 = ((2,2,2,4,4), (3,3,5,3,1,1)).

The value of the $R(\pi)$ function was determined for the obtained solution: R = 6.5 (H = 25, D = 0.3 h). This schedule is longer than the schedule with a minimum makespan $C_{\max}(\pi)$ of only 5.3%. It is characterized by only an 18-minute idle time in the work of the second workgroup for activity D and 10 additional changes of the types of precast elements during production. The received schedule confirms the effectiveness of the TS algorithm used. It also shows that it is possible to design it for the assumptions presented in the model so that it meets the minimum possible constraints imposed on it. The work

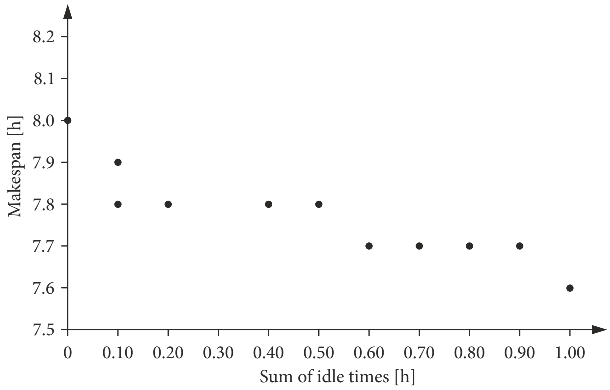

Figure 5. An approximation of makespan/sum of idle times trade off obtained with the use of the TS algorithm

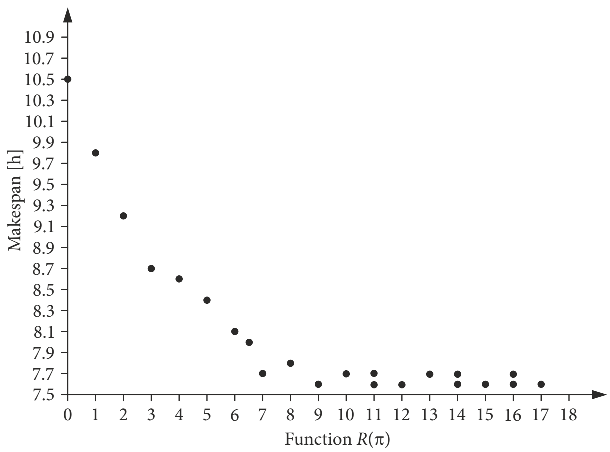

Figure 6. An approximation of makespan/function $R(\pi)$ trade off obtained with the use of the TS algorithm

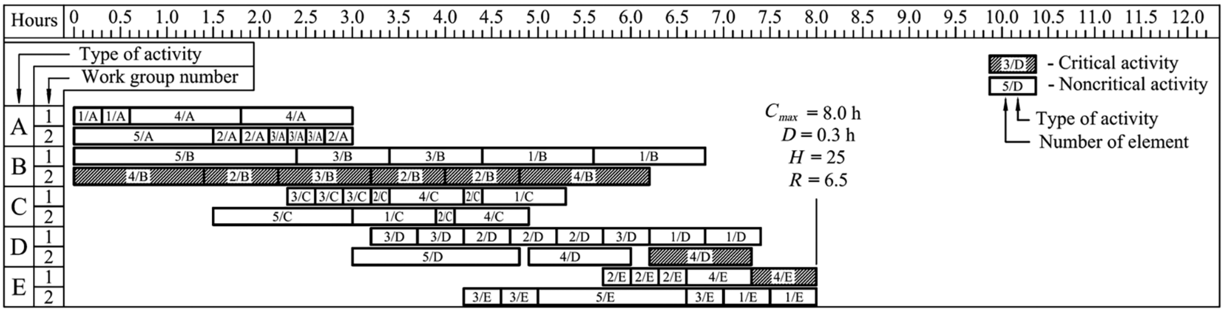

Figure 7. Working groups' work schedule for the solution, which was chosen by the manufacturing planner

schedule of the teams of working groups for the solution, which was chosen by the manufacturing planner is presented in Figure 7.

The optimization results obtained in this section cannot be compared with the results for the models presented in the literature review. This is due to the fact that the presented model adopts different constraints and criteria than in the previously presented models of scheduling the production of reinforced concrete prefabricated elements.

The case study presented in Section 6 and the results of the search for the optimal schedule (Section 7) are a real example of the application of the scheduling model in the production of precast components. Therefore, the theoretical scheduling model presented in the article should be used in a real application for planning this type of production. The flow shop models to which the model presented in this article belongs are typically used in ERP systems. They are usually used with single machines, for one-off or short-run production. The most common recipients of these planning systems are the consumer, automotive, electronics and defense industries. The form of ERP systems is determined by the expectations of recipients from these industries. For this reason, the specificity of the production of precast components may not be included in them. The problem presented in the article may find its application in a dedicated job scheduling system for the production of precast components, which is part of the ERP system for this type of production. Such a job scheduling system will determine the production schedule for a given set of activities using a given set of machines. This system will allocate activities to machines with regard to a certain optimization criterion using the optimal solution search technique. Currently used professional job scheduling systems are dedicated to specific production structures. The universal use of such a scheduling system (e.g., IBM ILOG CPLEX Optimizer (International Business Machines Corporation, 2021)), which is applicable to most planning systems, may be quite limited. Therefore, there is a need to adapt the job scheduling system dedicated to the production of precast components, consisting in the use of the scheduling model presented in the article.

Conclusions

The article presents a new optimization model for scheduling the production of precast reinforced concrete elements based on the hybrid flow shop problem. An important novelty in the model is the use of more than one workgroup to perform a given activity, the ability to conduct selected activities in parallel and the ability to change the order of performing elements for different activities. The optimization tasks in the model are NP-hard discrete optimization tasks. This means that for such problems only exact algorithms can be constructed, whose calculation time increases exponentially. A multiple increase in computing power of computers does not significantly improve the speed of solving such problems. Therefore, in order to solve optimization tasks, a metaheuristic algorithm of simulated annealing and a tabu search algorithm were used, the results of which were verified in comparison to other algorithms. The model includes a new objective functions, which were the sum of idle time of working groups and the total type changes of precast components. These are important criteria for those planning production because of the need to ensure continuity of the work and ensure high efficiency of working groups and quality of precast components. The article considers a two-criteria approach (duration of production and total idle time) and three-criteria approach (duration of production, total idle time, total changes of type of precast components). The algorithm for obtaining points of a set of Pareto-optimal solutions is given.

The proposed method of organizing working groups in the production of precast components has limitations of its use. In the presented model optimization objective functions are associated with the start/finish times for activities (makespan and sum of idle times of working groups) and the total type changes of precast components. These are the most important objective functions from the point of view of ensuring the highest efficiency of working groups in the production of precast components. Other optimization constraints that may exist during the production of precast components are limitation of the available equipment for their production, e.g. the number of available molds or production pallets or financial penalties for exceeding the eight-hour day working duration. In production plants, their number is usually sufficient to complete the intended production program. However, in some cases, the number of molds or pallets available during production may be limited, which can have a significant impact on the work schedule of the workgroups. In the model presented in the article, these constraints are not taken into account. In case of existing financial penalties for exceeding the eight-hour day working duration, it would be necessary to formulate a new objective function for the cost of work of the working groups that would take into account the existence of such penalties. Such the objective function would be minimized in the model. It is possible to take into account the above-mentioned additional constraints and the new objective function in the presented model, which will broaden the scope of its application. Works on extending the possibilities of using the presented model of scheduling of working groups during the production of precast components are planned in the near future.

The scheduling model presented in the article can be successfully used in planning work of working groups in factory conditions due to the possibility of significantly reducing the production time of precast elements. In addition, the use of the model allows us for suboptimal schedules with minimally possible working group idle time and minimally possible type changes of precast components. This results in a better, more efficient use of their working time and better quality of precast components.

Disclosure statement

Author have not any competing financial, professional, or personal interests from other parties.

References

Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021a). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers & Industrial Engineering*, *158*, 107408. https://doi.org/10.1016/j.cie.2021.107408

Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021b). Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *International Journal of Intelligent Systems*, *36*(10), 5887–5958. https://doi.org/10.1002/int.22535

Addo-Tenkorang, R., & Helo, P. (2011, October). Enterprise resource planning (ERP): A review literature report. In *Proceedings of the World Congress on Engineering and Computer Science* (*WCECS 2011*) (Vol. 2), San Francisco, USA.

Anvari, B., Angeloudis, P., & Ochieng, W. (2016). A multi-objective GA-based optimisation for holistic manufacturing, transportation and assembly of precast construction. *Automation in Construction*, *71*, 226–241. https://doi.org/10.1016/j.autcon.2016.08.007

Bennett, D. (2005). *Precast concrete: Tone texture form*. Birkhaeuser.

Chan, W. T., & Hu, H. (2002a). Constraint programming approach to precast production scheduling. *Journal of Construction Engineering and Management*, *128*(6), 513–521. https://doi.org/10.1061/(ASCE)0733-9364(2002)128:6(513)

Chan, W. T., & Hu, H. (2002b). Production scheduling for precast plants using a flowshop sequencing model. *Journal of Computing in Civil Engineering*, *16*(3), 165–174. https://doi.org/10.1061/(ASCE)0887-3801(2002)16:3(165)

Cochran, J. K., Horng, S. M., & Fowler, J. W. (2003). A multi-population genetic algorithm to solve multi-objective scheduling problems for parallel machines. *Computers and Operations Research*, *30*(7), 1087–1102. https://doi.org/10.1016/S0305-0548(02)00059-X

Coello, C. A. C., Lamont, G. B., & Veldhuizen, D. A. V. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer.

Dawood, N. N. (1995). Scheduling in the precast concrete industry using the simulation modelling approach. *Building and Environment*, *30*, 197–207. https://doi.org/10.1016/0360-1323(94)00039-U

Dawood, N. N. (1996). A simulation model for eliciting scheduling knowledge: An application to the precast manufacturing process. *Advances in Engineering Software*, *25*, 215–223. https://doi.org/10.1016/0965-9978(95)00096-8

Dawood, N. N. & Neale, R. H. (1993). Capacity planning model for precast concrete building products. *Building and Environment*, *28*, 81–95. https://doi.org/10.1016/0360-1323(93)90009-R

Glover, F., & Laguna, M. (1995) *Tabu search*. Kluwer Academic Publishers. https://doi.org/10.1007/978-1-4615-6089-0

Hayyolalam, V., & Kazem A. A. P. (2020). Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, *87*, 103249. https://doi.org/10.1016/j.engappai.2019.103249

Huang, T., & Yasuda, K. (2016). Comprehensive review of literature survey articles on ERP. *Business Process Management Journal*, *22*(1), 2–32. https://doi.org/10.1108/BPMJ-12-2014-0122

International Business Machines Corporation. (2021, November 10). *IBM ILOG CPLEX Optimizer*. https://www.ibm.com/products/ilog-cplex-optimization-studio

Ishibuchi, H., Misaki, S., & Tanaka, H. (1995). Modified simulated annealing algorithms for the flow shop sequencing problem. *European Journal of Operational Research*, *81*, 388–398. https://doi.org/10.1016/0377-2217(93)E0235-P

Kirkpatrick, S., Gelatt, C. D., & Vecchi M. P. (1983). Optimization by simulated annealing. In M. Mezard, G. Parisi, & M. Virasoro (Eds), *World Scientific lecture notes in physics: Vol. 9. Spin glass theory and beyond* (pp. 339–348). https://doi.org/10.1142/9789812799371_0035

Ko, C. H., & Wang, S. F. (2010). GA-based decision support systems for precast production planning. *Automation in Construction*, *19*(7), 907–916. https://doi.org/10.1016/j.autcon.2010.06.004

Ko, C. H., & Wang, S. F. (2011). Precast production scheduling using multi-objective genetic algorithms. *Expert Systems with Applications*, *38*(7), 8293–8302. https://doi.org/10.1016/j.autcon.2016.08.021

Lumdsen, P. (1968). *The line of balance method*. Pergamon Press.

Ma, Z., Yang, Z., Liu, S., & Wu, S. (2018). Optimized rescheduling of multiple production lines for flowshop production of reinforced precast concrete components. *Automation in Construction*, *95*, 86–97. https://doi.org/10.1016/j.autcon.2018.08.002

Nowicki, E., & Smutnicki, C. (1998). The flow shop with parallel machines: A tabu search approach. *European Journal of Operational Research*, *106*, 226–253. https://doi.org/10.1016/S0377-2217(97)00260-9

Ogbu, F., & Smith, D. (1990). The application of the simulated annealing algorithm to the solution of the n/m/Cmax flowshop problem. *Computers & Operations Research*, *17*(3), 243–253. https://doi.org/10.1016/0305-0548(90)90001-N

Podolski, M. (2016). Scheduling of job resources in multiunit projects with the use of time / cost criteria. *Archives of Civil Engineering*, *62*(1), 143–158. https://doi.org/10.1515/ace-2015-0057

Podolski, M., & Sroka, B. (2019). Cost optimization of multiunit construction projects using linear programming and metaheuristic-based simulated annealing algorithm. *Journal of Civil Engineering and Management*, *25*(8), 848–857. https://doi.org/10.3846/jcem.2019.11308

Podolski, M., & Rejment, M. (2019). Scheduling the production of precast concrete elements using the simulated annealing metaheuristic algorithm. *IOP Conference Series: Materials Science and Engineering*, *471*, 112083. https://doi.org/10.1088/1757-899X/471/11/112083

Ruiz, R., & Vázquez-Rodríguez, J. A. (2010). The hybrid flow shop scheduling problem. *European Journal of Operational Research*, *205*(1), 1–18. https://doi.org/10.1016/j.ejor.2009.09.024

Su, Y., & Lucko, G. (2016). Linear scheduling with multiple crews based on line-of-balance and productivity scheduling method with singularity functions. *Automation of Construction*, *70*, 38–50. https://doi.org/10.1016/j.autcon.2016.05.011

Tharmmaphornphilas, W., & Sareinpithak, N. (2013). Formula selection and scheduling for precast concrete production. *International Journal of Production Research*, *51*(17), 5195–5209. https://doi.org/10.1080/00207543.2013.795250

Wang, Z., Hu, H., & Gong, J. (2018). Framework for modelling operational uncertainty to optimize offsite production scheduling of precast components. *Automation in Construction*, *86*, 69–80. https://doi.org/10.1016/j.autcon.2017.10.026

Warszawski, A. (1984). Production planning in prefabrication plant. *Building and Environment*, *19*(2), 139–147. https://doi.org/10.1016/0360-1323(84)90039-8

Wittrock, R. J. (1988). An adaptable scheduling algorithm for flexible flow lines. *Operational Research*, *36*(3), 445–453. https://doi.org/10.1287/opre.36.3.445

Yang, Z., Ma, Z., & Wu, S. (2016). Optimized flowshop scheduling of multiple production lines for precast production. *Automation in Construction*, *72*, 321–329. https://doi.org/10.1016/j.autcon.2016.08.021