ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2023, Vol. 84, No. 5, pp. 594–608.
Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2023.

Linear Integer Programming Model as Mathematical Ware for an Optimal Flow Production Planning System at Operational Scheduling Stage

A. I. Kibzun[*,a] **and V. A. Rasskazova**[*,b]

[*]*Moscow Aviation Institute (National Research University), Moscow, Russia*
e-mail: [a]*kibzun@mail.ru,* [b]*varvara.rasskazova@mail.ru*

Received November 8, 2022
Revised February 22, 2023
Accepted March 20, 2023

Abstract—The problem of optimal flow production planning at the operational scheduling stage is being studied, using the example of the out-of-furnace department of a converter-based steel-making production in the iron metallurgy industry. To solve this problem, a linear integer programming model is proposed, which fully describes the specifics of the investigated technological processes. A major advantage of this approach is its scalability for solving related optimization problems in the industry of plant logistics, as well as flexibility in adapting to changes and fine-tuning the system of constraints and objective function. The software implementation of the developed model forms the basis of the operational scheduling module of the optimal flow production planning system, which is used for a large-scale computational experiment on real-world data.

Keywords: mathematical ware, linear integer programming, flow production, operational planning

DOI: 10.25728/arcRAS.2023.33.80.001

1. INTRODUCTION

Linear integer programming (LIP) is widely used in various fields of science and technology, including transportation and production planning. In [1, 2], LIP models were developed to solve applied problems in railway transportation planning. In [3], LIP methods were considered in the context of solving flow problems in networks. In [4, 5], various scheduling problems were reduced to the corresponding LIP problems. In [6], LIP models were proposed to solve some industrial planning problems. The main difficulties in reducing applied problems to the corresponding LIP models are related to describing the complex system of constraints inherent in many applied problems. In this paper, an LIP model is proposed to solve the problem of optimal flow production planning at the operational scheduling stage, which fully reflects all significant features of the considered technological processes. The software implementation of the proposed model forms the basis of the operational scheduling module of the intelligent decision-making system in planning and logistics problems of flow production in the black metallurgy industry.

In [7–10], an extended review of LIP model applications and modern solution methods is provided. In [7, 8], classical formulations and solution methods of LIP problems are discussed in detail, including boolean LIP. In [9, 10], particular attention is paid to the development of LIP models for solving various applied problems in the field of management, planning and decision making. The approach proposed in this paper is also scalable and can be extended to solving related optimization problems in the technological processes of metallurgical production. Such related optimization problems may include plant logistic transportation, crane fleet management, reassignment of technological routes, and more.

As the LIP problem is a well-known $\mathcal{NP}$-hard, its solution methods continue to be actively researched and developed. An extended review of modern methods for solving LIP problems is presented in [11, 12]. In this paper, solution methods for LIP problems are not discussed in detail. The main goal here is to develop an adequate and scalable mathematical model that fully describes the technological features of the processes under consideration. Any modern software can be used to obtain a solution for the proposed model.

In [13–15], some applied problems related to optimal planning of technological processes in metallurgical production were discussed, mainly focusing on improving the quality of the final product. In [13], a methodology for solving complex decision-making problems in metallurgical production management was proposed. Robust optimization approaches to solving related problems in the steelmaking industry were proposed in [16–18]. In [14, 15], algorithms were also developed to improve the quality of the final product in a hot rolling mill. In this paper, a fundamentally different approach to production optimization is considered, where the main focus is on energy efficiency at each stage of the production chain. In particular, the optimal implementation of the operational scheduling stage considered in this paper will significantly improve the overall production quality by increasing plan feasibility and ensuring even equipment utilization.

It is well-known that various classes of Machine Scheduling and Shop Scheduling problems can be successfully reduced to the Resources Constrained Project Scheduling Problem (RCPSP). The applied problem of flow production planning at the operational scheduling stage investigated in this paper also exhibits properties similar to RCPSP. Given a set of resources (machines, out-of-furmace units), it is necessary to construct a schedule for processing requirements (jobs, ladles of hot metal) within a given set of technological constraints. Thus, the investigated problem of operational scheduling of flow production can be classified as RCPSP with fixed processing durations and constraints on the start and finish times of processing each requirement. A wide review of different classes of RCPSP problems is presented in [19].

RCPSP is an $\mathcal{NP}$-hard problem in the strong sense, even in its simplest form. The best exact algorithm for solving it was proposed in [20] and provides solutions for instances of size up to $n = 60$, where n is the number of requirements. It is clear that this size is insufficient for practical problems. As for polynomial-time algorithms with guaranteed accuracy (when the error does not exceed a given constant), they are not known even for the case of $K = 1$, where K is the number of resources. A popular approach to solving RCPSP is based on forming upper and lower bounds for the optimal solution. In [21, 22], upper and lower bounds for the RCPSP solution were obtained using linear programming methods. Among efficient polynomial-time algorithms for solving RCPSP, the List Scheduling (LS) algorithm should be highlighted. An important drawback of the LS algorithm is the limitation on the size of the problem – even in the case of a small (several units) number of requirements, the algorithm does not guarantee the optimality of the obtained solution. Moreover, LS is not flexible enough to implement specific application constraints, such as shift production planning (when a specified number of requirements must be fulfilled within a given time interval across all machines). The same is true for various LS modifications, including Ant Colony and other metaheuristics. Therefore, the development of a generalized model that allows for fine-tuning of constraints and functionality, taking into account the peculiarities of the processes under consideration, is a relevant issue. In this paper, a LIP model is proposed for these purposes.

The paper has the following structure. Section 1 provides a general statement of the problem, including in terms of the subject area. Section 2 is devoted to describing the LIP model for solving the investigated problem and algorithms for forming the functional space. Section 3 presents the results of a computational experiment using the developed LIP model. In conclusion, paths for further development of the topic are discussed, including expanding the functionality of the model in the case of infeasible system of constraints, as well as continuing the methodology for solving related problems of shop floor logistics.

2. MATHEMATICAL MODEL OF THE PROBLEM OF FLOW PRODUCTION PLANNING

The problem of operational scheduling of flow production refers to the formation of a detailed schedule for the movement of demands (jobs to be executed) on machines in a shop floor with restrictions on the sequence of machines in the technological chain, intervals, and duration of processing of each demand depending on specified characteristics and type of machine. Let's consider a section of flow production as an example of the out-of-furnace processing department of a converter steel-making shop floor. We denote:

T — the number of different types of machines for out-of-furnace processing of steel (steel finishing unit, circulating vacuum steel unit, etc.),

$k(i) \in \mathbb{N}$ — the number of machines of the ith type, where $i = \overline{1, T}$ (steel finishing unit no. 1, steel finishing unit no. 2, out-of-furnace unit no. 1, etc. for each type of machine),

$$K = \sum_{i=1}^{T} k(i)$$

— the total number of machines in the shop floor.

The normative duration of transportation (minimum time in minutes) for each pair of machines in the shop floor is determined by a square matrix of the form

$$\Delta = \|\delta_{ij}\|, \quad i, j = \overline{1, K},$$

where $\delta_{ij} \in \{0\} \cup \mathbb{N}$ and the following conditions are hold:

1) the equality $\delta_{ij} = 0$ is achieved if and only if the transportation of the job (steel ladle) from machine i to machine j is prohibited;
2) transportation is allowed only between different machines, i.e.

$$\delta_{ii} = 0 \text{ for all } i = \overline{1, K};$$

3) the duration of transportation for any pair of machines does not depend on the start of movement, i.e.

$$\delta_{ij} = \delta_{ji} \text{ for all } i, j = \overline{1, K}.$$

In the case of assigning multiple jobs to the same machine sequentially, time is required for its setup. This is because different jobs may have different technological characteristics and processing requirements, and the machine (out-of-furnace processing unit) must be prepared to meet these requirements. Even with identical technological requirements, checking the operability of the machine is necessary before assigning it to perform the next job. Assuming that the setup norms are determined only by the type of machine, we denote the vector of setup durations as

$$\Pi = (\pi_1, \ldots, \pi_T),$$

where π_i is the minimum required time in minutes.

During the planning period, repair and maintenance measures, such as technical inspection (TI) or planned preventive maintenance (PPM), may be provided for any machine. We denote the set of all scheduled TIs and PPMs (hereinafter referred to as TIs) by

$$R = \{\rho_i | i = 1, \dots, Ro\},$$

where Ro is the number of such inspections and repairs. For each element ρ_i, parameters $(s(\rho_i), f(\rho_i), m(\rho_i))$ are defined, where $s(\rho_i), f(\rho_i)$ are the start and finish times, and $m(\rho_i) \in 1, \dots, K$ is the identifier of the machine (out-of-furnace processing unit) for which the TI is planned.

The grade of steel is the most important characteristic of jobs to be assigned to machines. It is the grade of steel that determines the technological instruction for job formation at the level of forecasting scheduling. For instance, different grades of steel have different norms for minimum and maximum allowable metal holding times in steel ladles. At the operational scheduling stage, the grade of steel also plays a decisive role, as different grades of steel require different out-of-furnace processing procedures (in terms of the type and number of units in the technological route, the length of steel processing on a unit of a specific type, etc.). Let $S = \{\sigma_i | i = \overline{1, s}\}$ be the set of steel grades for the considered flow production (converter shop floor). The minimum and maximum normative processing durations for each steel grade and each machine type are determined by the matrices

$$M = \|\mu_{ij}\|, \quad i = \overline{1, s}, \quad j = \overline{1, T}$$

and

$$\hat{M} = \|\hat{\mu}_{ij}\|, \quad i = \overline{1, s}, \quad j = \overline{1, T},$$

where $\mu_{ij}, \hat{\mu}_{ij} \in \{0\} \cup \mathbb{N}$ characterize the minimum and maximum processing time (in minutes) for steel of grade i on machine of type j, and $\mu_{ij} = \hat{\mu}_{ij} = 0$ if the steel of grade i is not subject to processing on machine of type j.

Let $P(\sigma_i) = \{p_1(i), \dots, p_n(i)\}$ be the set of permissible types of technological routes (the set of generalized technological routes, GTRs) for each steel grade $\sigma_i \in S$, where $n = n(\sigma_i) \in \mathbb{N}$ is the number of different GTRs for processing steel grade σ_i, and each GTR $p_j(i)$ is a sequence of machine types of the form

$$p_j(i) = (\tau_1(i, j), \dots, \tau_l(i, j)),$$

where $l = l(\sigma_i, j) \in \mathbb{N}$ is the number of machines in GTR j for processing steel grade σ_i, and $\tau_k(i, j) \in \{1, 2, \dots, T\}$ is the type of the kth machine in the considered GTR. For all σ_i, the set $P(\sigma_i)$ contains the main GTR as the first element and possible alternatives in descending order of priority. Essentially, the priority is determined by the energy efficiency of production following the given GTR. For instance, the main GTR envisages a shorter total transportation time (even considering the minimum processing time on each machine), so that machines do not idle and the raw material does not cool down (additional heating requires significant resource consumption).

2.1. Source Data

Let $Z = \{\zeta_i | i = \overline{1, z}\}$ be a set of jobs for the forecast schedule. For each job ζ_i, $(k_i, r_i, u_i, d_i, \sigma_i)$ are given, where k_i is the start position (converter), r_i is the start time, u_i is the finishing position (continuous steel casting machine), d_i is the finishing time, $\sigma_i \in S$ is the steel grade. It is necessary to determine specific technological routes and deadlines for all jobs.

Since the start and finish positions of the forecast schedule jobs are essentially some reference points of the shop, let us determine the normative duration of transporting the job (steel-ladle) from each such position to other machines. Let us define:

$$\Delta' = \|\delta'_{ij}\|, \quad i = \overline{1, K'}, \quad j = \overline{1, K},$$

where K' is the number of the initial positions in the shop (number of converters), δ'_{ij} is the minimum transportation time (in minutes) from the initial position i to the machine j, and $\delta'_{ij} = 0$ if transportation is prohibited. Similarly,

$$\Delta'' = \|\delta''_{ij}\|, \quad i = \overline{1, K''}, \quad j = \overline{1, K},$$

where K'' is the number of the final positions in the shop (number of continuous steel casting machines), δ''_{ij} is the minimum transportation time (in minutes) from the machine j to the final position i, and $\delta''_{ij} = 0$ if transportation is prohibited.

2.2. Statement of the Problem

The problem of forming an operational schedule for assigning heats (the problem of operational scheduling) can be formulated as follows. For each job $\zeta_i \in Z$, $i = \overline{1, z}$ a detailed time and machine-specific technological route (extended TR) of the form

$$f(\zeta_i) = \big(s_1(i), m_1(i), f_1(i), \ldots, s_l(i), m_l(i), f_l(i)\big)$$

must be assigned, subject to the following constraints:

1) for any $f(\zeta_i)$ such that $\sigma = \sigma_i$, and some $j \in \{1, \ldots, n(\sigma)\}$, $l = l(\sigma, j)$ is satisfied, i.e. the length (in terms of the number of machines) of the extended TR corresponds to the length of some GTR for processing the steel grade σ;

2) for any $f(\zeta_i)$ and the corresponding $j \in \{1, \ldots, n(\sigma)\}$, it is satisfied that

$$t\,(m_k(i)) = \tau_k(i, j)$$

for all $k = \overline{1, l}$, where $t\,(m_k(i))$ is the type of the kth machine in the extended TR;

3) for all $f(\zeta_i)$ and $m_k(i), m_h(i)$ such that $k < h$, it is satisfied that

$$s_h(i) - f_k(i) \geqslant \delta_{kh},$$

where δ_{kh} is the minimum transportation duration from machine $m_k(i)$ to machine $m_h(i)$;

4) for any $f(\zeta_i)$ and $m_k(i)$ such that $t\,(m_k(i)) = t$, it is satisfied that

$$\mu_{\sigma t} \leqslant f_k(i) - s_k(i) \leqslant \hat{\mu}_{\sigma t},$$

where $\mu_{\sigma t}, \hat{\mu}_{\sigma t}$ respectively are the minimum and maximum durations of processing the steel grade σ on the machine of type t;

5) for any $f(\zeta_i), f(\zeta_j)$ and $m = m_k(i) = m_h(j)$ such that $t(m) = t$, it is satisfied that

$$\begin{cases} s_h(i) - f_k(j) \geqslant \pi_t, \text{ if } s_k(i) \leqslant s_h(j), \\ s_k(i) - f_h(j) \geqslant \pi_t \text{ otherwise,} \end{cases}$$

where π_t is the minimum time (in minutes) required to set up the machine of type t when assigning jobs sequentially;

6) for all $f(\zeta_i)$ and TR $\rho_j \in R$ such that $m_k(i) = m = m(\rho_j)$ for some k, it is satisfied that

$$\begin{cases} s_k(i) > s(\rho_j), \\ s_k(i) \geqslant f(\rho_j) \end{cases}$$

or

$$\begin{cases} s_k(i) < s(\rho_j), \\ f_k(i) \leqslant s(\rho_j), \end{cases}$$

where $s(\rho_j)$, $f(\rho_j)$ are the start and finish times of the TR on machine m;
7) for all $f(\zeta_i)$ such that $k_i = k$ and $m_1(i) = m$, it is satisfied that

$$s_1(i) - r_i \geqslant \delta'_{km},$$

where $s_1(i)$ is the time when job i arrives at the first machine in the extended TR $f(\zeta_i)$, r_i is the start of the execution of the ith job, k_i is the start position of the job's execution, δ'_{km} is the minimum transportation duration (in minutes) from the start position of the job's execution to the first machine in the extended TR;
8) for all $f(\zeta_i)$ such that $u_i = u$ and $m_l(i) = m$, it is satisfied that

$$d_i - f_l(i) \geqslant \delta''_{um},$$

where $f_l(i)$ is the time when job i is finished on the last machine in the extended TR, d_i is the finish time of the execution of the ith job, and δ''_{um} is the minimum transportation duration (in minutes) from the last machine in the extended TR to the position of the job's completion.

The objective function is defined as the minimum total transportation duration, which corresponds to the maximum total processing duration of all requirements on each machine in the processing TR, i.e.

$$\sum_{i=1}^{z} \sum_{j=1}^{l(i)} (f_j(i) - s_j(i)) \longrightarrow \max,$$

where $l(i)$ is the length (in terms of number of machines) of the TR $f(\zeta_i)$ assigned to execute the job $\zeta_i \in Z$. The choice of this function is due to the fact that lengthy transportation leads to the need for adjustment of requirements on the next machine in the TR and incurs additional resource costs (such as temperature or chemical heating). Thus, the total transportation duration, as previously noted in discussing TR priorities, reflects the energy efficiency indicator of the process, and minimizing it (i.e. maximizing the processing duration, time on the machine) corresponds to the goals of increasing energy efficiency.

In other words, the problem of real-time scheduling is to construct a detailed TR for each job under specified constraints on the sequence and duration of processing on each machine, as well as taking into account the normative transportation duration. To solve this problem, a LIP model is proposed, the discussion and description of which is dedicated to the following section.

3. LIP MODEL FOR THE OPERATIONAL SCHEDULING OF FLOW PRODUCTION PROBLEM

In modeling applied problems using LIP methods, a fundamental role is assigned to the structure of the functional space (model variables), since the efficient and adequate implementation of this stage lays the foundation for both the structure of the constraint system and subsequent opportunities for fine-tuning the model with regard to additional technological features of the studied processes. Let us take a closer look at this procedure.

In Algorithm 1, a set of boolean variables is formed for each job, corresponding to the processing of the job on one of the machines in the shop floor, indicating the start and finish times, as well as the duration of operations. In this process:

— the function $end(i, s, j, k)$ fixates $l(s, j) - k$ machines in the route (starting from the end) with the minimum processing and transportation duration, where i is the identifier of the job, j is the GTR for processing the grade $s = \sigma_i$, and k is the number of fixated machines;

— the function $start(i, s, j, k)$ fixates $k - 1$ machines in the route (starting from the beginning) similarly to the function $end(i, s, j, k)$;

— the function $move(i, s, j, k, r)$, for fixed start and finish times in the route, forms options for the start and finish of processing job i on the kth machine with the minimum processing duration.

Algorithm 1. Formation of functional space

$c = 0$ ▷ global counter of variables
$s_1 = 0, s_2 = 0, f_1 = 0, f_2 = 0$ ▷ auxiliary counters
For all $i = \overline{1, z}$ **do**
  $s = \sigma_i$ ▷ the grade
  $n = n(s)$ ▷ number of GTRs
  For all $j = \overline{1, n}$ **do**
    For all $k = \overline{1, l(s, j)}$ **do**
      If $k \neq l(s, j)$ **then**
        $end(i, s, j, k)$ ▷ fix $l(s, j) - k$ machines from the end of the TR
      Else
        $s_1 = 0, s_2 = 0$
      If $k \neq 1$ **then**
        $start(i, s, j, k)$ ▷ fix the first $k - 1$ machines from the beginning
      Else
        $f_1 = 0, f_2 = 0$
      $t = \tau_k(s, j)$ ▷ type of the kth machine
      For all $r = \overline{1, k(t)}$ **do**
        $move(i, s, j, k, t, r)$ ▷ moving the kth machine

Algorithm 2. Function $end(i, s, j, k)$

For all $h = \overline{1, l(s, j) - k}$ **do**
  $t = \tau_{l(s,j)-h+1}(s, j)$
  For all $r = \overline{1, k(t)}$ **do**
    $m = m(t, r)$ ▷ identifier of the rth machine of type t
    For all $ii = \overline{f_1, f_2}$ **do**
      $c = c + 1$
      If $ii = 0$ **then**
        $u = u_i$
        $f(c) = d_i - \delta''_{um}$ ▷ move the transportation duration to the left
      Else
        $f(c) = s(ii) - \delta_{m(ii)m}$
      $s(c) = f(c) - \mu_{st}$ ▷ move the processing duration to the left
      $id(c) = i, p(c) = j, num(c) = l(s, j) - h + 1, \alpha(c) = j$
  If $f_2 = 0$ **then**
    $f_1 = c - k(t) + 1$
  Else
    $f_1 = f_2 + 1$
  $f_2 = c$

Algorithm 3. Function $start(i, s, j, k)$

For all $h = \overline{1, k-1}$ **do**
  $t = \tau_h(s, j)$
  For all $r = \overline{1, k(t)}$ **do**
    $m = m(t, r)$
    For all $ii = \overline{s_1, s_2}$ **do**
      $c = c + 1$
      If $ii = 0$ **then**
        $v = k_i$
        $s(c) = r_i + \delta'_{vm}$ ▷ move the transportation duration to the right
      Else
        $s(c) = f(ii) + \delta_{m(ii)m}$
      $f(c) = s(c) + \mu_{st}$ ▷ move the processing duration to the right
      $id(c) = i, p(c) = j, num(c) = h, \alpha(c) = j$
  If $s_2 = 0$ **then**
    $s_1 = c - k(t) + 1$
  Else
    $s_1 = s_2 + 1$
  $s_2 = c$

Algorithm 4. Function $move(i, s, j, k, t, r)$

$m = m(t, r)$
For all $ii = \overline{s_1, s_2}$ **do**
  For all $jj = \overline{f_1, f_2}$ **do**
    If $ii = 0$ **then**
      $t_1 = r_i, v = k_i, \delta = \delta'_{vm}$
    Else
      $t_1 = f(ii), \delta = \delta_{m(ii)m}$
    If $jj = 0$ **then**
      $u = u_i, t_2 = d_i - \delta''_{um} - \mu_{st}$
    Else
      $t_2 = s(jj) - \delta_{m(ii)m} - \mu_{st}$
    While $t_2 - t_1 \geqslant \delta$ **do**
      $c = c + 1,\ s(c) = t_2,\ f(c) = s(c) + \mu_{st},\ id(c) = i,\ p(c) = j,\ num(c) = k,\ m(c) = m,\ \alpha(c) = j,$ $t_2 = t_2 - step$ ▷ move the start of processing to the left on the kth machine

It is clear that the implementation of Algorithm 1 is possible in a different configuration, where the start of the route is fixed first and then its end. In general, the subsets of variables formed in different approaches may be different, leading to structural differences in subsequent constraints and functionality. In addition, in the proposed variant, the minimum processing duration is fixed for each machine for each job i, reducing the optimization potential. However, with the aim of reducing the dimensionality of the model as a whole, the problem of real-time scheduling of flow production can be decomposed into two stages, the first of which is the proposed configuration of Algorithm 1 with a fixed minimum processing duration. The subsequent expansion within allowable limits for each machine can be implemented within an auxiliary LIP model with a guaranteed solution, which constitutes a direction for development of the proposed methodology.

It is worth noting the substantive meaning of the parameter *step* in Algorithm 4 — this is a tunable parameter that corresponds to the discretization of the functional space. For example, when the value of the parameter *step* is sufficiently small, variables corresponding to the same job i

and processing on the same TR j will have proportional differences in transportation duration based on the *step* parameter value. Correspondingly small values of the parameter *step* (in the extreme case, equal to 1 min) significantly increase the dimensionality of the functional space. Therefore, the natural problem arises of finding a balance between the size of the parameter *step* and the performance of the model as a whole (since a large dimensionality of the functional space entails a significant increase in computational resource consumption).

Based on the results of Algorithm 1 (including the nested functions of Algorithms 2, 3, and 4), arrays of parameters for the variables of the LIP model will be formed. For each variable $x_i \in \{0, 1\}$, $i = \overline{1, c}$, the following are defined: id_i — the identifier of the job for which the machine is assigned; p_i — the GTR; num_i — the ordinal number of the machine in the TR; m_i — the identifier of the machine; s_i — the start time of processing; f_i — the finish time of processing; α_i — the coefficient of the objective function, determined in the simplest case as the ordinal number of the selected GTR.

The main limitation in solving the operational scheduling problem is the completion of all jobs in the set Z. Then,

$$\sum_{i=1}^{c}\{x_i\colon id_i = id, num_i = 1\} = 1 \tag{1}$$

for all $id = \overline{1, z}$. The condition (1) ensures the selection of a single extended TRs for the execution of each job. Moreover, the condition $num_i = 1$ allows avoiding the construction of an "ifthen" statement associated with the length of the selected TRs.

Furthermore, the selected starting point of the TR in constraint (1) must be continued according to its length in terms of the number of machines, i.e.,

$$\sum_{i=1}^{c}\{x_i\colon id_i = id, p_i = p\} = l(\sigma, p)\cdot\sum_{i=1}^{c}\{x_i\colon id_i = id, p_i = p, num_i = 1\} \tag{2}$$

for all $id = \overline{1, z}$ and $p = \overline{1, n(\sigma)}$, where $\sigma = \sigma_{id}$. In other words, constraint (2) is designed to "track" the choice of a complete (in terms of the number of machines) TR whose starting point is determined by constraint (1). At the same time, constraint (2) does not contradict the choice of such $x_i = x_j = 1$, where $num_i = num_j = k$ for some $k > 1$. In order to exclude such errors, the following constraints are introduced:

$$\sum_{i=1}^{c}\{x_i\colon id_i = id, p_i = p, num_i = k\} \leqslant 1 \tag{3}$$

for all $id = \overline{1, z}$, $p = \overline{1, n(\sigma)}$ and $k = \overline{2, l(\sigma, p)}$, where $\sigma = \sigma_{id}$.

Thus, due to constraints (1), a single TR will be selected in the solution. Furthermore, due to constraints (2), this TR will be extended to $l(\sigma, p)$ in terms of the number of machines. Moreover, each machine k will be selected only once due to constraint (3).

Let us now formulate constraints on machine performance. In the case of sequential assignment of jobs to the same machine, the setup time should be no less than the specified performance parameter for the respective machine type, i.e.,

$$x_i + \sum_{j=1, j\neq i}^{c}\{x_j\colon m_i = m_j, |s_i - f_j| \leqslant \pi_t\} \leqslant 1 \tag{4}$$

for all $m_i = \overline{1, K}$, where $t = \tau_k(\sigma, p)$ if $k = num_i, id_i = id, \sigma = \sigma_{id}, p = p_i$, and π_t is the normative setup time for machine type t. It is clear that constraints (4) are redundant, as some of them may be subsets of others. To form a set of maximal inclusion constraints of the form (4), a simple lexicographic rule can be used (pairwise comparison of binary strings of length K, where each string corresponds to a constraint for variable x_i, and the jth element of the string takes the value 1 if m_j is included in the constraint for x_i).

Constraints on maintenance operations can be introduced at the stage of functional space formation by setting to zero all variables for which

$$\sum_{i=1}^{c} \{x_i\colon s(\rho) \leqslant s_i \leqslant f(\rho), m_i = m(\rho)\} = 0, \tag{5}$$

$$\sum_{i=1}^{c} \{x_i\colon s_i \leqslant s(\rho) \leqslant f_i, m_i = m(\rho)\} = 0 \tag{6}$$

for all planned maintenance operations $\rho \in R$, where $m(\rho)$ is the machine on which the maintenance operation is planned, and $s(\rho)$ and $f(\rho)$ are the start and completion times. In other words, for any x_i for which the start or completion time of processing job i on machine m_i falls within the period of the scheduled maintenance of that machine, x_i should be set to zero. This condition can be satisfied either at the level of formation of the model variable set (by means of pre-checking), or at the level of constraints. The latter option is more preferable in the conditions of the modified problem of finding the maximum feasible subsystem of constraints, where some of them may be violated according to established priorities. However, this problem is beyond the scope of the present paper and presents a direction for future research.

In general, the objective function in the problem of operational scheduling of flow production is the indicator of energy efficiency. However, with the aim of decomposing the problem and reducing the dimensionality of the main model in the simplest configuration, ordinal numbers of GTR are used as variable coefficients — the main GTR is more energy efficient compared to the alternatives, and for each subsequent alternative GTR there is a decrease in processing energy efficiency indicators. Thus, the problem is to minimize the objective function of the form

$$\sum_{i=1}^{c} \alpha_i \cdot x_i \ \longrightarrow \min \tag{7}$$

subject to the constraints

$$\begin{cases} \sum\limits_{i=1}^{c} \{x_i\colon id_i = id, num_i = 1\} = 1, \\ \sum\limits_{i=1}^{c} \{x_i\colon id_i = id, p_i = p\} = l(\sigma, p) \cdot \sum\limits_{i=1}^{c} \{x_i\colon id_i = id, p_i = p, num_i = 1\}, \\ \sum\limits_{i=1}^{c} \{x_i\colon id_i = id, p_i = p, num_i = k\} \leqslant 1, \\ x_i + \sum\limits_{j=1, j\neq i}^{c} \{x_j\colon m_i = m_j, |s_i - f_j| \leqslant \pi_t\} \leqslant 1, \\ \sum\limits_{i=1}^{c} \{x_i\colon s(\rho) \leqslant s_i \leqslant f(\rho), m_i = m(\rho)\} = 0, \\ \sum\limits_{i=1}^{c} \{x_i\colon s_i \leqslant s(\rho) \leqslant f_i, m_i = m(\rho)\} = 0, \\ x_i \in \{0, 1\}, \end{cases} \tag{8}$$

where

- $id = \overline{1, z}$ — the set of jobs to be assigned to a sequence of machines,

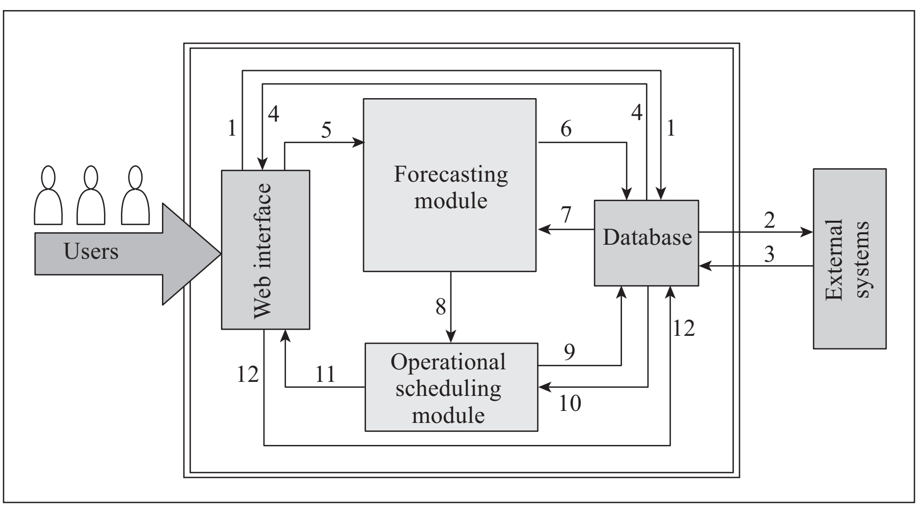
- $p = \overline{1, n(\sigma)}$ — the set of GTR for processing grade $\sigma = \sigma_{id}$,
- $l(\sigma, p)$ — the length of the TR according to the number of machines,
- $k = \overline{1, l(\sigma, p)}$ — the ordinal number of the machine in TR p,
- $t = \tau_k(\sigma, p)$ — the type of the kth machine in TR p,
- π_t — normative duration of setup for machine type t,
- $\rho \in R$ — the set of scheduled maintenance operations,
- $m(\rho)$ — the machine on which maintenance ρ is planned,
- $s(\rho), f(\rho)$ — start and finish of maintenance ρ.

For solving the problem (7) with constraints (8), any application software (IBM CPLEX, Analytic Solver VBA Excel, Gurobi, Google OR-Tools, etc.) for solving mathematical programming problems or built-in libraries of high-level programming languages (Optimization Toolbox, CVXOPT, HeO, etc.) can be used.

4. COMPUTATIONAL RESULTS

The LIP model (7), (8) was implemented in Python 3.8.5 using the PuLP library for finding solutions in the operational scheduling module of the flow production planning system. The simplest system component diagram and data flow are presented in the figure, where

[1] request for parameters of the daily task, transfer of a set of casting series indicating the start time of casting, the number of heats, and the series identifiers;
[2] request for parameters of the daily task by series identifiers (steel grade, casting cycle, recommended technological route for out-of-furnace processing);
[3] transfer of the daily task parameters from external systems to the database;
[4] transfer of the daily task parameters from the database to the User WEB interface;
[5] request for calculation of the forecast schedule, transfer of a set of casting series indicating the start of casting, the number of heats, and the parameters of the daily task;
[6] request for reference information on allowed holds for grades and statistics of energy resource consumption depending on the processing duration (forming the target function coefficients);
[7] transfer of reference information from the database to the forecast scheduling module;
[8] transfer of the forecast schedule parameters (list of heats indicating the grade, hold, end time of blowing in the converter and start of casting, recommended technological route for out-of-furnace processing);



System component diagram.

[9] request for reference information (alternative technological routes for out-of-furnace processing, processing duration on each machine depending on the grade and hold, duration of steel-ladle transportation with a heat between shop floor units);
[10] transfer of reference information from the database to the operational scheduling module;
[11] transfer of the results of the forecast and operational scheduling calculation (full processing route of each heat in the daily task indicating the start and finish time of processing at each unit, including the converter and continuous casting machine);
[12] saving the results of the forecast and operational scheduling calculation for subsequent display (upon request) and analysis.

The computational experiment was carried out on a personal computer with Intel Core m3 1.2 GHz, 8Gb 1867 MHz LPDDR3, macOS 10.13.6. The source data were the actual production scenarios implemented at converter shop floor of the Novolipetsk Metallurgical Entherprise (Lipetsk, Russia). For each fixed daily task (for months of 2020 and 2021), an optimized out-of-furnace processing schedule (operational schedule) was proposed, and the weighted-average costs for each type of energy resource consumption were compared in monetary equivalent. The results of the

Table 1. May, 2020

Date	F_1	F_2	Δ, %	t, sec	Date	F_1	F_2	Δ, %	t, sec
01	1961.6	1852.1	5.58	7.11	02	1859.8	1759	5.42	5.20
03	1730.9	1690.2	2.35	3.22	05	1953.4	1843.8	5.61	6.64
06	1816.3	1627.7	10.38	3.92	07	1894.8	1835	3.16	4.64
08	1893.3	1832.8	3.20	9.25	09	1801.9	1717.2	4.70	8.69
10	2162.4	1823.8	15.66	60.75	11	1673.3	1251.6	25.20	1.08
12	1426.8	1101.8	22.78	0.55	13	2921.7	1131.3	61.28	0.74
14	1632.0	931.5	42.92	0.44	15	1364.2	1080.5	20.80	0.88
16	1434.8	1086.9	24.25	0.50	17	1488.4	1133.2	23.86	0.42
18	1527.8	1283.7	15.98	0.74	19	1641.3	1233.4	24.85	0.51
20	1781.4	1383	22.36	0.52	21	1877.2	1704.7	9.19	2.65
22	1820.3	1723.2	5.33	16.07	24	1990.7	2065.5	–3.76	33.47
25	1787.1	1632.2	8.67	34.85	26	1887.2	1925.6	–2.03	10.34
27	1905.0	1770.6	7.06	3.94	28	2136.1	1982.8	7.18	3.71
29	1887.9	1990.8	–5.45	25.17	30	2017.1	1891.7	6.22	26.67
31	2052.3	1935.3	5.70	15.39					

Table 2. June, 2020

Date	F_1	F_2	Δ, %	t, sec	Date	F_1	F_2	Δ, %	t, sec
01	2297.0	2205.9	3.97	23.05	02	1952.1	1833.3	6.09	16.67
03	2291.1	1815.4	20.76	7.52	04	2045.3	1659.5	18.86	10.29
05	1997.5	2005.2	–0.39	24.68	06	1915.3	1880.5	1.82	4.63
07	1942.8	2044.4	–5.23	6.93	08	1599.5	1321.3	17.39	5.67
09	1648.8	1299.8	21.17	3.79	10	1786.3	1535.1	14.06	5.11
11	2228.7	1903.2	14.60	32.12	12	1723.8	1507.4	12.55	2.69
13	1483.7	1228	17.23	0.66	14	1554.5	984.9	36.64	0.46
15	1195.7	1030.9	13.78	0.67	16	1104.1	873.9	20.85	0.71
17	1461.5	1123.6	23.12	0.55	18	918.9	748.1	18.59	0.67
19	1692.3	1439.8	14.92	1.71	20	1959.7	1790.9	8.61	20.54
21	1956.8	1566.8	19.93	3.37	22	2305.3	2126.6	7.75	23.25
23	1986.4	1567.8	21.07	24.78	24	1839.5	1807.7	1.73	41.31
25	1732.3	1556	10.18	7.64	26	1721.3	1742.4	–1.23	22.03
27	1921.4	1691.6	11.96	4.98	28	1679.7	1424.8	15.18	13.14
29	1831.0	1462.7	20.11	3.86	30	1963.9	1503.8	23.43	4.13

computational experiment are partially presented in Tables 1 and 2, where the following notations have been used.

— The column "Date" indicates the day of the month for which the daily task was recorded and a comparison was made between the production costs in the actual scenario and in the optimized scenario obtained using the developed LIP model.

— The columns "F_1" and "F_2" provide the objective function values of the actual and optimized scenarios, respectively — the total production costs of all heats of the daily task by resource types in monetary equivalent (in hundreds of rubles). These values are calculated for each heat, based on its processing route and taking into account the specified average costs (for example, the electrode consumption during the heating of low-alloyed steel heat in the furnace-ladle unit is 25 kg, which is equivalent to 4644.2 rubles in monetary terms).

— The column "Δ,%" indicates the difference between the values of "F_1" and "F_2", i.e. if the value in the "Δ, %" column is negative, then the functional value achieved in the optimized scenario is greater than that for the actual scenario (this is explained by the fact that some technological constraints of the model may be violated in the actual scenarios, which essentially leads to a different formulation of the problem in terms of regulatory and reference information).

— The column "t, sec" indicates the CPU time in seconds spent on finding the solution for the optimized scenario.

Similar calculations were performed for the periods from August to December 2020 and from January to April 2021. Table 3 presents the summarized results for each month of the considered period with the average value of the parameter "Δ, %", where:

— the column "Month" indicates the month of the considered period from 2020–2021;

— the column "Avg, %" provides the values of the objective function improvement in average, calculated as the arithmetic mean of all "Δ, %" values in the indicated month.

Table 3. Results of computational experiment

Month	Avg, %	Month	Avg, %	Month	Avg, %
2020, May	13.05	2020, June	13.65	2020, July	9.76
2020, August	8.81	2020, September	11.10	2020, October	18.09
2020, November	9.64	2020, December	19.16	2021, January	9.39
2021, February	13.58	2021, March	10.59	2021, April	18.15

As shown in Table 3, the developed LIP model as a mathematical ware of the optimal planning module for flow production scheduling system provides a significant improvement in the quality of objective function on real-world data — up to 19.16% on average (the maximum value in the "Avg, %" column in Table 3). The economic effect is achieved by redistributing the machines of the shop floor among the jobs, which in turn leads to the preferential selection of the main GTR for processing. In other words, in the actual scenario, the jobs assigned to alternative GTRs were processed by the main GTR in the optimized scenario. This results in a higher overall energy efficiency indicator for the daily task in the optimized scenario. It is also worth noting that not only the quantitative factor is important (when the number of jobs processed by the main GTR in the optimized scenario is greater than in the actual), but also the qualitative — if one of the two fixed jobs needs to be assigned to an alternative GTR, the priority is given to the option for which the total resource consumption is less (taking into account the grade of each job). At the same time, the computational costs are quite acceptable in terms of the operational efficiency of the system (the values in the "t, sec" column in Tables 1, 2). Thus, a high efficiency of the system implementation as a whole, and in particular, the operational scheduling module based on the developed LIP model, can be expected in the practice of operating flow production.

5. CONCLUSION

In the paper, the problem of optimal planning for flow production scheduling at the operational scheduling stage was studied using the example of the out-of-furnace department of a converter shop floor in the iron metallurgy industry. To solve this problem, a LIP model was proposed, which fully describes all technological features of the processes under consideration while allowing for flexible configuration and modification of the system constraints and objective function. The software implementation of the proposed model formed the basis of the operational scheduling module for the optimal planning system for flow production, which was used to conduct a large-scale computational experiment on real-world data. The results of the computational experiment demonstrate the high effectiveness of the proposed approach and the potential for achieving significant economic benefits from the implementation of the system in the practice of flow production operation.

Further development of the topic is primarily associated with equipping the developed model with additional functionality to improve the quality of the obtained solutions by increasing the processing time of requirements on each machine within acceptable intervals. Another functionality that also requires development and implementation in the developed model is related to the search for maximally feasible subsystems of constraints with priorities in conditions of infeasible model in its original formulation.

The continuation of the proposed approach to modeling production problems using LIP methods is the study of related problems in shop floor logistics, including transport planning and crane management. In such problems, the system of constraints in the original formulation is also often infeasible. In this regard, the planned methods for forming maximally feasible subsystems of constraints with priorities, which are intended for research and development, can also find wide and effective application.

FUNDING

The paper was supported by the Russian Science Foundation (project no. 23-21-00293).

REFERENCES

1. Lazarev, A.A. and Musatova, E.G., Integer Formulations of the Problem of Forming Railway Trains and Scheduling Their Movement *Control of Large Systems*, 2012, no. 38, pp. 161–169.

2. Gainanov, D.N., Ignatov, A.N., Naumov, A.V., and Rasskazova, V.A., On Track Procession Assignment Problem at the Railway Network Sections, *Autom. Remote Control*, 2020. vol. 81, no. 6, pp. 967–977. https://doi.org/10.1134/S0005117920060028

3. Khoo, T., *Tselochislennoe programmirovanie i potoki v setyakh* (Integer Programming and Network Flows), Moscow: Mir, 1974.

4. Ryan, D.M. and Foster, B.A., *An Integer Programming Approach to Scheduling*, in Wren, A., ed., Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling, Amsterdam: North-Holland, 1981, pp. 269–280.

5. Wagner, H.M., An Integer Linear-Programming Model for Machine Scheduling, *Naval Research Logistics Quarterly*, 1959, vol. 6, no. 2, pp. 131–140.

6. Pochet, Y. and Wolsey, L.A., *Production Planning by Mixed Integer Programming*, Switzerland: Springer Series in Operations Research & Financial Engineering, 2006.

7. Shevchenko, V.N. and Zolotykh, N.Y., *Lineinoe i tselochislennoe lineiloe programmirovanie* (Linear and Integer Linear Programming), Nizhny Novgorod: Nizhny Novgorod State University named after N.I. Lobachevsky, 2004.

8. Schrijver, A., *Teoriya lineinogo i tselochislennogo programmirovaniya* (Theory of Linear and Integer Programming), Moscow: Mir, 1991.

9. Sigal, I.Kh. and Ivanova, A.P., *Vvedenie v prikladnoe diskretnoe programmirovanie. Modeli i vychislitel'nye algoritmy* (Introduction to Applied Discrete Programming. Models and Computational Algorithms), Moscow: Fizmatlit, 2007.

10. Appa, G.M., Pitsoulis, L.S., and Paul, W.H., *Handbook on Modeling for Discrete Optimization*, Switzerland: Springer Series in Operations Research & Management Science, 2006.

11. Wolsey, L.A., *Integer Programming*, NJ: John Wiley & Sons, 2020.

12. Hu, T.C. and Kahng, A.B., *Linear and Integer Programming Made Easy*, Switzerland: Springer, 2016.

13. Kabulova, E.G., Intelligent Control of Multistage Systems in Metallurgical Production, *Modeling, Optimization, and Information Technology*, 2019, tom 7, no. 1(24), pp. 341–351. https://doi.org/10.26102/2310-6018/2019.24.1.022

14. Stolbov, V.Yu., Gitman, M.B., and Fedoseev, S.A., Management of the Process of Forming the Quality of Products of an Industrial Enterprise, *Applied Mathematics and Control*, 2016, no. 3, pp. 79–98.

15. Gainanov, D.N. and Berenov, D.A., *Algorithm for Predicting the Quality of the Product of Metallurgical Production*, CEUR Workshop Proceedings, 2017, vol. 1987, pp. 194–200.

16. Qiu, Y., Wang, L., Xu, X., Fang, X., and Pardalos, P.M., Scheduling a Realistic Hybrid Flow Shop with Stage Skipping and Adjustable Processing Time in Steel Plants, *Applied Soft Computing*, 2018, vol. 64, pp. 536–549.

17. Kong, M., Pei, J., Xu, J., Liu, X., and Pardalos, P.M., A Robust Optimization Approach for Integrated Steel Production and Batch Delivery Scheduling with Uncertain Rolling Times and Deterioration Effect, *International Journal of Production Research*, 2020, vol. 58, no. 17, pp. 5132–5154. https://doi.org/10.1080/00207543.2019.1693659

18. Long, J., Sun, Z., et al., A Robust Dynamic Scheduling Approach Based on Release Time Series Forecasting for the Steelmaking Continuous Casting Production, *Applied Soft Computing*, 2020, vol. 92, p. 106271.

19. Lazarev, A.A. and Gafarov, A.A., *Teoriya raspisanii. Zadachi i algoritmy* (Scheduling Theory. Problems and Algorithms), Moscow: Nauka, 2011.

20. Brucker, P. and Knust, S., *Complex Scheduling*, Germany: Springer-Verlag Berlin, 2006.

21. Mingozzi, A., Maniezzo, V., Ricciardelli, S., and Bianco, L., An Exact Algorithm for Project Scheduling with Resource Constraints Based on a New Mathematical Formulation, *Management Science*, 1998, vol. 44, pp. 714–729.

22. Burkov, V.N., Problems of Optimum Distribution of Resources, *Control Cybernetics*, 1972, vol. 1, no. 2, pp. 27–41.

This paper was recommended for publication by A.A. Lazarev, a member of the Editorial Board