MSC 90B80 DOI: 10.14529/mmp240307

A DECOMPOSITION APPROACH IN THE PROBLEM OF DISTRIBUTION-TYPE PLANNING WITH PRIORITY CONSTRAINTS

V.A. Rasskazova, Moscow Aviation Institute (National Research University), Moscow, Russian Federation, varvara.rasskazova@mail.ru

The paper considers the problem of distribution-type planning with priority constraints. For a given set of requirements and resources with established usage parameters, it is necessary to construct an assignment plan that satisfies a system of priority constraints. In this case, two queues of constraints on quantitative and qualitative characteristics are distinguished, respectively. At the stage of solving the problem with the first queue of constraints, a basic integer linear programming (ILP) model and a dynamic scheme for its formation are developed. Within this approach, the original problem is reduced to solving a sequence of similar problems of significantly smaller dimension, which allows to take into account the priorities of resource use directly in the construction and guarantees the convergence of the basic ILP model at the final iteration of the dynamic scheme. At the stage of implementing the second queue of constraints for the obtained basic solution, an integral criterion in the form of an upper estimate is introduced, and a modified ILP model is considered. The model modification procedure is based on the penalty function method and includes the additional equipment of the constraint system, the objective function, and the functional space by a subset of auxiliary Boolean variables. It is proved that the modified model is guaranteed to be solvable and determines the maximal feasible subsystem of constraints of the second queue for the original problem. Within the analysis of the operability and efficiency of the proposed approach, a computational experiment is conducted using real-scale data.

Keywords: scheduling theory; integer linear programming; discrete production; production planning system; decomposition approach.

Introduction

Decomposition approaches are actively researched and applied to solve various planning and resource allocation problems in conditions of uncertainty and large-scale dimensionality. The main directions are the development of new decomposition algorithms, the integration of stochastic models, and parallel computing.

In [1], a new algorithm for constructing schedules for multi-product production for large-scale problems was proposed, based on a decomposition scheme and supplemented with heuristic algorithms. Similar decomposition approaches were proposed in [2], where the planning horizon is divided into a sequence of shorter horizons. In the present paper, a decomposition approach is proposed for solving the problem of distribution-type planning with priority constraints.

Resource allocation problems are widespread in various fields, such as economics [3], organizational management [4], and production planning [5]. At the same time, such problems are often characterized by the requirement of integer variables, which significantly complicates the solution process. The extension of the Lagrangian method for the class of integer resource allocation problems was proposed in [6]. In the present paper, at each stage of solving the original problem, ILP models are considered. To reduce the computational effort of the solution search, a dynamic model scheme is proposed for construction of the ILP models.

ILP provides a wide class of methods and techniques for solving various planning and management problems, including transportation and production problems. In [7, 8], ILP models were proposed for solving applied problems of railway planning and management. In [9–11], various methods were proposed for solving planning and management problems in the metallurgical production industry. A robust approach to solving problems in steel production management was proposed in [12–14]. In the present paper, the steel production industry is also considered as an application of the obtained results.

In Section 1, the problem statement is provided. In Section 2, the stage of solving the problem with the first-priority constraints is considered, and a dynamic scheme for the ILP model construction has been developed, the solution of which determines the subset of requirements that are guaranteed to be feasible through a fixed subset of resources with adjacent priorities. In Section 3, the second-priority constraints are introduced into consideration. A modified ILP model is provided, and it is proven that its solution corresponds to the maximal feasible subsystem of constraints of the original problem. In Section 4, the results of a computational experiment using the developed approach are presented.

1. Problem Statement

Let us consider a general distribution-type planning problem with priority constraints. Let $C = \{c_i\}$, $i = \overline{1, k}$ — be the set of requirements to be executed, where for each requirement c_i the parameters ω_i, $\hat{\omega}_i \in \mathbb{R}^+$ are given, corresponding to the minimum necessary and maximum permissible volume of execution, respectively. We will assume that the order of execution of the requirements c_i is fixed and determined by the set of indices $i = \overline{1, k}$.

Denote by $T = \{t_j\}$, $j = \overline{1, n}$ the set of resources available for use. For each resource t_j, we define the parameters $\mu_j \in \mathbb{R}^+$, $\tau_j \in \mathbb{N}$ — the volume of the resource and the priority of use, respectively. The practical meaning of the priority order of resource use is that for any t', $t'' \in T$, where $\tau' > \tau''$, the resource t' must be completely exhausted before the resource t'' is used. Considering the fixed order of execution of requirements, we will assume that the resource t' is used earlier than the resource t'' if the resources t', t'' are used to execute the requirements c_{i_1}, $c_{i_2} \in C$, respectively, and $i_1 \leqslant i_2$.

For the case when the given set of resources is not sufficient (in the total volume) for the execution of the set of requirements, we introduce an additional resource t_0 with unlimited volume μ_0 and priority $\tau_0 < \min\limits_j \{\tau_j\}$. In other words, the additional resource t_0 can be used, if necessary, as a last resort, when all available resources have been completely exhausted. In particular, if the additional resource is not used, then $\mu_0 = 0$.

Similarly, if the given set of resources is excessive for the execution of the set of requirements C, an additional requirement c_0 with unlimited volume $\hat{\omega}_0$ is introduced into consideration. In this case, we will assume that the requirement c_0 is to be executed after all the requirements $c_i \in C$ have been completed to the specified volume. Thus, without loss of generality, all the excessive resources $t_j \in T$ can be assigned to execute the requirement c_0 in full and in the order established by their priority of use.

For the sake of brevity in the further presentation, let us put $C_0 = C \cup \{c_0\}$ and $T_0 = T \cup \{t_0\}$. We introduce the notation:

$$S = T_0 \times [0; +\infty) = \{(t, z) | t \in T_0, z \in [0; +\infty)\}$$

for the set of pairs (resource, volume). Then the distribution planning problem can be reduced to finding a mapping of the form

$$f\colon C_0 \longrightarrow 2^S \tag{1}$$

with constraints on the parameters of requirement execution, available volumes, and priorities of resource use. In (1), the assignment of the form $f(c_i) = \{(t_{j_1}, z_{ij_1}), \ldots, (t_{j_k}, z_{ij_k})\}$ means that for the execusion of the requirement c_i, the resources $t_{j_1}, \ldots, t_{j_k}$ are assigned in the volumes $z_{ij_1}, \ldots, z_{ij_k}$, respectively. The inverse mapping of the form $f^{-1}(t_j, z_{ij}) = \{c_i\}$ is interpreted as the requirement c_i, for the execusion of which the resource t_j is used in the volume z_{ij}.

Denote by F the set of all distinct mappings of the form (1) and introduce the criterion

$$\hat{t}(f) = \frac{1}{k} \cdot \sum_{i=1}^{k} |f(c_i)| + |f(c_0)|$$

as the sum of the average number of resources used for the execution of each requirement and the total number of not completed requirements. Then the distribution-type problem with priority constraints will take the form:

$$\hat{t}(f) \longrightarrow \min_{f \in F}. \tag{2}$$

Let us discuss the two groups of constraints associated with the quantitative and qualitative characteristics of the solution. The first-queue group of constraints is related to quantitative characteristics and includes the following constraints.

- For the execution of each requirement $c_i \in C$, no more than r resources may be used:

  $$|f(c_i)| \leqslant r \tag{3}$$

  for any $c_i \in C$ (the number of resources used for the requirement c_0 is not limited).

- Each resource $t_j \in T$ can be used for the execution of no more than s requirements:

  $$\left| \bigcup_{i=1}^{k} \{f^{-1}(t_j, z_{ij})\} \right| \leqslant s \tag{4}$$

  for any $t_j \in T$ (the number of requirements for the execution of which the resource t_0 can be used is not limited).

- The total volume of resources used in each assignment satisfies the parameters set for the corresponding requirement:

  $$\omega_i \leqslant \sum_{i=1}^{n} \{z_{ij} | (t_j, z_{ij}) \in f(c_i)\} \leqslant \hat{\omega}_i \tag{5}$$

for all $c_i \in C$, where $z_{ij} \leqslant \mu_j$ and

$$\sum_{i=1}^{k} \sum_{j=1}^{n} \{z_{ij} | (t_j, z_{ij}) \in f(c_i)\} = \mu_j \tag{6}$$

for all $t_j \in T$ (the execution volume of the requirement c_0, as well as the volume of use of the resource t_0, are not limited).

- For the execusion of each requirement, resources with the same or adjacent priorities can be used:

$$|\tau_{j_1} - \tau_{j_2}| \leqslant 1 \tag{7}$$

for all t_{j_1}, $t_{j_2} \in T$ and $c_i \in C$ such that $(t_{j_1}, z_{ij_1}), (t_{j_2}, z_{ij_2}) \in f(c_i)$.

- Resources must be used in order of priority:

$$\begin{cases} (t_j, z_{ij}) \in f(c_i) \\ \tau_j = \tau \end{cases} \Longrightarrow \{(t_j, z_{ij}) | \tau_j > \tau + 1\} \subseteq \bigcup_{k=1}^{i-1} f(c_k) \tag{8}$$

or, in the case of surplus resources:

$$\begin{cases} (t_j, z_{ij}) \in f(c_0) \\ \tau_j = \tau \end{cases} \Longrightarrow \begin{cases} \{(t_j, z_{ij}) | \tau_j > \tau\} \subseteq \bigcup\limits_{i=1}^{k} f(c_i), \\ \{(t_j, z_{ij}) | \tau_j < \tau\} \subset f(c_0), \\ \{(t_j, z_{ij}) | \tau_j = \tau\} \subseteq \bigcup\limits_{i=0}^{k} f(c_i) \end{cases} \tag{9}$$

and similarly in the case of an excessive total volume of the set of requirements:

$$\begin{cases} (t_j, z_{ij}) \in f(c_0) \\ \tau_j = \tau \end{cases} \Longrightarrow \{(t_j, z_{ij}) \in f(c_i) | \tau_j = \tau_0\} \cap f(c_{i+k}) = \varnothing \tag{10}$$

for all $k \geqslant 1$.

The constraints (7) – (10) define the specifics of the considered planning problem with priorities on the use of resources. The second-queue group of constraints defines the qualitative characteristics of the solution.

Let for each requirement $c_i \in C$ the regulatory tolerances for impurity content α be defined. Denote by α_i and $\hat{\alpha}_i$ the minimum required and maximum permissible indicators, respectively. Without loss of generality, we will assume that the regulatory tolerances for impurity content are measured in specific volume units.

Let for each resource $t_j \in T$ the volume of impurity content a_j also be defined as input data. Then the second-queue constraints can be formalized as:

$$\omega_i \cdot \alpha_i \leqslant \sum_{j=1}^{n} \{z_{ij} \cdot a_j | (t_j, z_{ij}) \in f(c_i)\} \leqslant \hat{\omega}_i \cdot \hat{\alpha}_i \tag{11}$$

for all $c_i \in C$.

To solve the problem (2) with constraints (3) – (11), a decomposition approach is proposed, within which the first-queue constraints (3) – (10) and the second-queue constraints (11) are considered sequentially.

2. ILP Model with First-Queue Constraints

Let us define for each pair of requirements $c_i \in C$, $i = \overline{1,k}$, and resources $t_j \in T$, $j = \overline{1,n}$, the value $x_{ij} \geqslant 0$ as the volume of resource t_j used to execute requirement c_i. As characteristic variables, we introduce for each x_{ij} the value $y_{ij} \in \{0,1\}$. We will consider that $y_{ij} = 1$ if the resource t_j is used in the volume $x_{ij} > 0$ to execute the requirement c_i, and $y_{ij} = 0$ otherwise. Similarly, we define $x_{i0} \geqslant 0$ and $y_{i0} \in \{0,1\}$ for the use of the additional resource t_0, as well as $x_{0j} \geqslant 0$ for the use of the additional requirement c_0. Note that the consideration of characteristic variables for c_0 is not justified, since the case of redundancy of the total resource volume is not a constraint of the problem and is not taken into account in the objective function (2). It is clear that the dimension of the set $\{x_{ij}\} \cup \{y_{ij}\}$ is determined by the initial dimensions of C and T. However, due to the constraints (7) – (10) on the priorities and order of resource use, a significant part of the variables y_{ij} will have to turn into 0. Thus, a connected problem on forming the maximal subset of requirements, for the execution of which resources with fixed adjacent priorities can be used, arises. For these purposes, a dynamic solution scheme is proposed.

Considering the introduced notation, the problem (2) with the constraints (3) – (6) can be formulated as an ILP problem of the form

$$\frac{1}{k} \cdot \sum_{i=1}^{k} \sum_{i=1}^{n} y_{ij} + \sum_{i=1}^{k} y_{i0} + \sum_{i=1}^{k} x_{i0} \longrightarrow \min_{x,y} \tag{12}$$

with the constraints

$$\begin{cases} \sum\limits_{j=1}^{n} x_{ij} + x_{i0} \geqslant \omega_i \text{ for all } i = \overline{1,k}, \\ \sum\limits_{j=1}^{n} x_{ij} + x_{i0} \leqslant \hat{\omega}_i \text{ for all } i = \overline{1,k}, \\ \sum\limits_{i=1}^{k} x_{ij} + x_{0j} = \mu_j \text{ for all } j = \overline{1,n}, \\ \sum\limits_{j=1}^{n} y_{ij} \leqslant r \text{ for all } i = \overline{1,k}, \\ \sum\limits_{i=1}^{k} y_{ij} \leqslant s \text{ for all } j = \overline{1,n}, \\ x_{ij} \leqslant y_{ij} \cdot \mu_j \text{ for all } i = \overline{1,k}, j = \overline{1,n}, \\ x_{i0} \leqslant y_{i0} \cdot \hat{\omega}_i \text{ for all } i = \overline{1,k}, \\ x_{ij} \geqslant 0 \text{ for all } i = \overline{1,k}, j = \overline{1,n}, \\ x_{i0} \geqslant 0 \text{ for all } i = \overline{1,k}, \\ x_{0j} \geqslant 0 \text{ for all } j = \overline{1,n}, \\ y_{ij} \in \{0,1\} \text{ for all } i = \overline{1,k}, j = \overline{0,n}. \end{cases} \tag{13}$$

Within the dynamic scheme, the solution of the problem (2) with the constraints (3) – (10) is formed iteratively. At each iteration, a subset of resources with adjacent priorities and the current requirement to be executed is fixed. Then the constraints (7), (8) are automatically satisfied by construction. If the total volume of the fixed resources turns out to be redundant (the additional requirement c_0 is used in the solution), then the next requirement is considered according to the established order. The process continues until a shortage of resources is achieved (the additional resource t_0 is used in the solution). The resulting subset of requirements will be maximal from the point of view of their execution using the fixed subset of resources. Then the constraints (9), (10) can be implemented at the stage of forming the functional space by excluding the variables of the form x_{0j} for all resources t_j with a higher priority.

Let $T' = \{t_j | \tau_j = \tau \text{ or } \tau_j = \tau - 1\}$ be the subset of resources with fixed adjacent priorities and available volume parameters $\{\mu'_j\}$. Let $c_N \in C$ be the current requirement to be executed, with volume parameters ω'_N, $\hat{\omega}'_N$. In particular, for the initial iteration of the solution, the value $N = 1$, the volume parameters correspond to the original parameters μ_j, ω_1, $\hat{\omega}_1$ and $\tau = \max_j \{\tau_j\}$.

Everywhere further in the description of the algorithms, the symbol $\triangleright$ denotes a comment to the main text in the corresponding line.

Algorithm 1. Dynamic scheme

Require: T', c_N, C
Ensure: C' – a subset of requirements that can be executed using the resources T'
1: $C' = \{c_N\}$
2: $N' = 1$ $\triangleright$dimension of the set C'
3: Solve the problem (12) with constraints (13) for the set of requirements C' and the set of resources T'
4: **If** $y_{i0} = 1$ for some $i \in \overline{1, N'}$ **then**
5: $C' = C' \setminus \{c_N\}$ $\triangleright$exclude from the set C' the requirement added in the previous step
6: $N = N - 1$
7: Go to step 18
8: **Else** 9: **If** $N \neq k$ **then**
10: $N = N + 1$ $\triangleright$include the next requirement in order with the original volume parameters in the set C'
11: $C' = C' \cup \{c_N\}$
12: $\omega'_N = \omega_N$
13: $\hat{\omega}'_N = \hat{\omega}_N$
14: $N' = N' + 1$
15: Go to step 3
16: **Else**
17: Go to step 18
18: **Return** C'

In Algorithm 1, the ILP problem (12) is solved multiple times based on the results of checking the condition in step 4. However, the dimensions of the problem for each call are extremely small and increase by no more than one requirement at a time. This approach allows us to consider only the significant variables of the model, which can lead to a potential speedup compared to solving the problem for the full-scale functional space and the extended system of constraints taking into account the conditions (7) – (10).

Theorem 1. *Let $T' = \{t_j | \tau_j = \tau \text{ or } \tau_j = \tau - 1\}$ and C' is a subset of requirements formed as a result of running Algorithm 1. Then for T' and C', there exists a solution to the problem* (2) *with constraints* (3) – (10).

Proof. By condition of the theorem, a subset C' is the result of running Algorithm 1. Thus, for T' and C', there exists a solution to the problem (2) with constraints (3) – (8). In this case, the constraints (3) – (6) are satisfied directly within the ILP model, taking into account (13), and the constraints (7), (8) are satisfied by the definition of the set T'.

Let us consider the solution of the problem (12). Note that according to condition 4 of Algorithm 1, $y_{i0} = 0$ for all $i \in \overline{1, N'}$ (the resource t_0 is not used). Thus, the following holds:

$$\sum_{t_j \in T'} \mu_j \geqslant \sum_{c_i \in C'} \omega_i. \tag{14}$$

<u>Case 1.</u> Let $x_{0j} = 0$ for all $j|\tau_j = \tau$ (the additional requirement c_0 does not use resources with a higher priority). Then the constraints (9), (10) are satisfied, and the theorem is proved.

<u>Case 2.</u> Let $x_{0j} > 0$ for some $j|\tau_j = \tau$. Then, in view of the relationship (14), there exists a solution of the form

$$\begin{cases} x_{i0} = 0 \text{ for all } i \in \overline{1, N'-1}, \\ x_{N'0} \geqslant 0, \\ x_{0j} = 0 \text{ for all } j|\tau_j = \tau, \end{cases} \tag{15}$$

such that the constraints (13) are satisfied. At the same time, the solution (15) also satisfies the constraints (9), (10) of the problem (2), and thus the theorem is fully proved.

□

From Theorem 1, it follows that the solution of the problem (2) with constraints (3) – (10) can be obtained as the solution to the problem (12), (13) with additional constraints of the form (15) for T' and C', constructed according to Algorithm 1. Let us describe the Algorithm 2 of general approach to solving the problem for the original T and C.

Algorithm 2. Problem with First-Queue Constraints

Require T_0, C_0

Ensure $\{f(c_i) = \{(t_j, z_{ij})\}\}$ – solution of the problem (2) with constraints (3) – (10)

1: $\mu'_j = \mu_j$ for all $j = \overline{1, n}$, and $\omega'_i = \omega_i$, $\hat{\omega}'_i = \hat{\omega}_i$ for all $i = \overline{1, k}$

2: Start with $N = 1$ and $\tau = \max_j \{\tau_j\}$

3: **While** $T \neq \varnothing$ or $C \neq \varnothing$

4: Fix c_N and $T' = \{t_j | \tau_j = \tau \text{ or } \tau_j = \tau - 1\}$ with parameters μ'_j and ω'_i, $\hat{\omega}'_i$

5: **Execute** Algorithm 1 for T', c_N and C

6: Solve the problem (12) with constraints (13), (15) for C' and T'

7: **Return** $f(c_i)$ for all $c_i \in C'$, $i = \overline{1, k}$

8: **If** $y_{N0} = 1$ **then** ▷the additional resource t_0 is used in the solution (requirement c_N is not fully executed)

9: Set $C' = C' \setminus \{c_N\}$ and adjust ω_N

10: **Else**

11: Set $N = N + 1$ ▷move to the next requirement

12: **If** $x_{0j} = 0$ for all $t_j \in T'$

13: Set $\tau = \tau - 2$

14: **Else** ▷not all resources with priority $\tau - 1$ are exhausted at the current step of the solution

15: **For all** $t_j \in T' | x_{0j} > 0$ and $x_{0j} \neq \mu'_j$ **do**

16: Adjust the available volume μ'_j
17: Set $T' = \{t_j|\tau_j = \tau - 1\}$ and $\tau = \tau - 1$
18: Set $C = C \setminus C'$ and $T = T \setminus T'$
19: **If** $T = \varnothing$ and $C \neq \varnothing$
20: Set $f(c_i) = \{(t_0, \omega_i)\}$ for all $c_i \in C$

The result of running Algorithm 2 is the solution of the problem (2) with the first-queue constraints (3) – (10). The obtained solution will be called the *basic* solution of the resource-constrained planning problem with priorities. Note that the satisfaction of the first-queue constraints is the primary goal from a practical point of view. In other words, the satisfaction of the second-queue constraints should not reduce the quantitative characteristics achieved for the basic solution. In this regard, a decomposition approach is proposed, in which the involvement of the second-queue constraints is implemented as a separate stage.

3. ILP Model with Second-Queue Constraints

Taking into account the notation adopted in (11), as well as the ILP model (12), (13), let us introduce an additional set of constraints for the technological parameter α of the form

$$\omega_i \cdot \alpha_i \leqslant \sum_{j=1}^{n} x_{ij} \cdot a_j \leqslant \hat{\omega}_i \cdot \hat{\alpha}_i \text{ for all } i = \overline{1,k}. \tag{16}$$

Note that the relations (16) can be defined similarly for any arbitrary number of technological parameters.

Let $\hat{M}_c$ be the total number of resources $t_j \in T'$ involved in the execution of requirements $c_i \in C'$ as a result of running Algorithm 2. To implement the condition related to the quantitative characteristics of the basic solution, we introduce an upper bound

$$\sum_{i=1}^{k}\sum_{j=1}^{n} y_{ij} \leqslant \hat{M}_c. \tag{17}$$

Then, the ILP model (12) with constraints (13), (15) – (17) defines the solution of the problem (2) with constraints (3) – (11) for T' and C', constructed according to Algorithm 1, and the value of $\hat{M}_c$, set according to Algorithm 2. At the same time, it is clear that the model may be infeasible for a number of natural reasons related to the initial values of a_j for the available resources $t_j \in T'$ and α_i, $\hat{\alpha}_i$ for the requirements $c_i \in C'$ to be executed.

Following the principles of the penalty function method, let us introduce Boolean variables $\zeta_i \in \{0, 1\}$ for all $i = \overline{1,k}$ and modify the ILP model (12), (13), (15) – (17) as follows:

$$\frac{1}{k} \cdot \sum_{i=1}^{k}\sum_{j=1}^{n} y_{ij} + \sum_{i=1}^{k} y_{i0} + \sum_{i=1}^{k} x_{i0} + A \cdot \sum_{i=1}^{k} \zeta_i \longrightarrow \min_{x,y,\zeta} \tag{18}$$

with constraints (13), (15), (17) and

$$\omega_i \cdot \alpha_i - A \cdot \zeta_i \leqslant \sum_{j=1}^{n} x_{ij} \cdot a_j \leqslant \hat{\omega}_i \cdot \hat{\alpha}_i + A \cdot \zeta_i \tag{19}$$

for all $i = \overline{1,k}$, where $A \in \mathbb{R}^+$ — some positive number, fixed for each instance of T', C'.

Theorem 2. *Let T' and C' be formed as a result of running Algorithm 1, and $\hat{M}_c$ be set according to Algorithm 2. Then, there exists a value $A \in \mathbb{R}^+$ such that the solution to the problem* (18) *with constraints* (13), (15), (17) *and* (19) *is guaranteed to exist for the given T', C' and $\hat{M}_c$.*

Proof. According to the conditions of the theorem, T' and C' are formed as a result of running Algorithm 1. Then, as a result of running Algorithm 2 for the given T' and C', a solution to the problem (12) with constraints (13), (15) will be constructed. Let us denote it as $\mathrm{x} = (x_{j_1 i_1}, x_{j_1 i_2}, \dots)$ and $\mathrm{y} = (y_{j_1 i_1}, y_{j_1 i_2}, \dots)$, and note that for any ζ_i, the solution x, y also satisfies the constraints (13), (15), (17) of the problem (18).

Let us set $\zeta_i = 0$ for all $i = \overline{1,k}$. If all the constraints (19) are satisfied in this case, then the theorem is proved. Consider the case when the constraints (19) are violated for some $i \in \overline{1,k}$. In this case, let us set $\zeta_i = 1$ and $A = \hat{\omega}_i \cdot \hat{\alpha}_i$. Then, the lower bound in (19) becomes < 0, and the upper bound is doubled. If the constraints (19) are still not fully satisfied (in terms of the upper bound), then the value of A can be increased accordingly.

The described procedure is repeated until the constraints (19) are satisfied for all $i = \overline{1,k}$. Thus, the theorem is fully proved.

□

The proof of Theorem 2 is built on the principles of iterative selection of the value $A \in \mathbb{R}^+$ until the constraints (19) are satisfied for all requirements $i = \overline{1,k}$. However, from the point of view of practical implementation of the ILP model (18) with constraints (13), (15), (17) and (19) for solving the problem (2) with constraints (3) – (11), such an approach turns out to be quite inflexible. In this regard, let us consider a predictive rule for a guaranteed selection of the value $A \in \mathbb{R}^+$ of the form:

$$A = \sum_{i=1}^{k} \hat{\omega}_i \cdot \hat{\alpha}_i + \sum_{j=1}^{n} \mu_j \cdot a_j, \tag{20}$$

where $\hat{\omega}_i$, $\hat{\alpha}_i$ and μ_j, a_j are given parameters of the requirements $c_i \in C'$ and resources $t_j \in T'$, respectively.

Theorem 3. *The solution to the problem* (18) *with constraints* (13), (15), (17) *and* (19), *where $A \in \mathbb{R}^+$ is defined according to the rule* (20), *corresponds to the maximal feasible subsystem (MFS) of constraints of the form* (16).

Proof. Let us denote the solution of the problem (18) as $\mathrm{x} = (x_{i_1 j_1}, x_{i_2 j_2}, ...)$, $\mathrm{y} = (y_{i_1 j_1}, y_{i_2 j_2}, ...)$ and $\zeta = (\zeta_{i_1 j_1}, \zeta_{i_2 j_2}, ...)$.

Case 1. Let the original system of constraints of the form (16) be feasible. Then, the optimal value of the functional (18) is achieved when $\zeta_i = 0$ for all $i = \overline{1,k}$, and the theorem is proved. Denote the corresponding solution as x_{opt}, y_{opt} and ζ_{opt}. In this case, it is clear that for ζ_{opt}, the condition $|\{\zeta_i = 1\}| = 0$ holds.

Case 2. Let the constraints (19) be violated for some $i \in \overline{1,k}$ for x_{opt}, y_{opt}. Following the logic of the proof of Theorem 2, let us set $\zeta_i = 1$ accordingly. Then, due to (13) and (20), the constraints (19) will be guaranteed to be satisfied for all $i = \overline{1,k}$. Denote the obtained solution as x_{opt}, y_{opt} and ζ'. For ζ', also denote $N_1 = |\{\zeta_i = 1\}|$ and $N_0 = |\{\zeta_i = 0\}|$.

Note that the number N_0 determines the dimension, and $\{\zeta_i = 0\}$ — the form of the feasible subsystem of constraints (16). In this case, an increase of the number N_0 is equivalent to a decrease of the number N_1 and leads to a violation of the constraints (19) directly by construction. At the same time, a decrease of the number N_0 (or an increase of the number N_1) is meaningless, as it leads to an increase in the functional (18) with fixed x_{opt}, y_{opt} and contradicts the condition of maximality of the feasible subsystem of constraints (16). We obtain that $N_0 = \max\limits_{\zeta_i} N$ and $N_1 = \min\limits_{\zeta_i} N$, where $N = |\{\zeta_i = 1\}|$ for an arbitrary solution ζ of the problem (18).

Thus, the solution of the form x_{opt}, y_{opt} and ζ' defines the MFS of constraints (16), and the theorem is fully proved.

□

From Theorem 3, it follows in particular that the application of the rule (20) when choosing A allows one to form the MFS of the constraints (16) in the form of x_{opt}, y_{opt} and $\zeta_i = 1$ for such $i \in \overline{1,k}$, where the corresponding constraints (19) are violated. This approach does not guarantee the optimality of the functional (18), but in a number of practical cases, it can significantly increase the speed of solving the original problem as a whole.

4. Computational Experiment

Based on the results presented in Sections 2 and 3, we will describe a general scheme for solving the distribution-type planning problem with priority constraints.

Algorithm 3. General scheme for solving the distribution-type planning problem with priority constraints

Require T_0, C_0
Ensure $\{f(c_i) = \{(t_j, z_{ij})\}\}$ – the solution of the problem (2) with constraints (3) – (11)
Set $\mu'_j = \mu_j$ for all $j = \overline{1,n}$, and $\omega'_i = \omega_i$, $\hat{\omega}'_i = \hat{\omega}_i$ for all $i = \overline{1,k}$
Start with $N = 1$ and $\tau = \max\limits_j\{\tau_j\}$
While $T \neq \varnothing$ or $C \neq \varnothing$
Fix c_N and $T' = \{t_j | \tau_j = \tau \text{ or } \tau_j = \tau - 1\}$ with parameters μ'_j and ω'_i, $\hat{\omega}'_i$
Execute Algorithm 1 for T', c_N and C
Solve the problem (12) with constraints (13), (15) for C' and T'
Establish $\hat{M}_c$
Establish A according to the rule (20) for C' and T'
Solve the problem (18) with constraints (13), (15), (17), (19) for C', T', $\hat{M}_c$ and A
Return $f(c_i)$ for all $c_i \in C'$, $i = \overline{1,k}$
Execute lines 8 – 18 of Algorithm 2
If $T = \varnothing$ and $C \neq \varnothing$ **then**
Set $f(c_i) = \{(t_0, \omega_i)\}$ for all $c_i \in C$

Algorithm 3 was implemented in Python 3.11 using the open-source PuLP library for solving ILP problems. The use of the PuLP library within the computational experiment is dictated by the low dimensionality of the arising ILP problems, which is ensured by the application of the proposed decomposition approach. In general, for solving the arising ILP problems, an efficient heuristic algorithm for solving $\mathcal{NP}$-hard combinatorial optimization problems from [15] can be used.

The computational experiment was based on the problem of planning production processes in the mixer department of the converter shop of a metallurgical enterprise.

Pig iron ladles (PIL) with molten pig iron coming from the blast furnace shop need to be poured into pig iron casting ladles (PIC) for further transportation and processing in the converter shop. Each PIC can hold up to 300 (tons) of pig iron, and each PIL — up to 100 (tons). In terms of the distribution-type planning problem, the set of PICs acts as the requirements to be executed. The resources for execution the requirements is the set of incoming PILs, where the priority of use is determined by the order of their arrival in the mixer department. The planning problem is to distribute the incoming resources among the requirements, taking into account the constraints on the resource volumes and the technological features of the implementation of the production process. In particular, for the parameters r and s of the model (18), the values $r = 2$ and $s = 4$ were chosen due to the ratio of the volumes of requirements and resources.

The test data for the computational experiment covers a one-year period and fixes the actual production scenario. The optimized scenario is formed using the developed software. The key performance indicator of the solution is the average number of PICs for which 3 PILs are used for execution. For the actual scenario, the average number of PICs using 3 PILs for execution is 40%. Consideration of this criterion is related to the fact that the use of 3 PILs in each execution ensures the timely and rhythmic return of the PILs to the blast furnace shop. As a result, an increase in the turnaround of rolling stock in the mixer department leads to an increase in the overall production quality in terms of the execution of the integrated production plan in the converter shop.

The results of the computational experiment conducted for one month within the considered period are presented in Table 1, where

• the "Day" row indicates the calendar day in the month under consideration,

• the "Number of PICs" row indicates the number of PICs to be executed (the number of requirements) in the corresponding day,

• the "Avg. number of PILs" row indicates the average number of PILs (the number of resources) used to execution each requirement in the corresponding day,

• the "Avg. number of PILs (month)" row indicates the average number of PILs used to execution each requirement in the month under consideration.

Table 1

September 2020

Day	1	2	3	4	6	7	10	11	12	15
Number of PICs	18	19	19	15	22	33	13	15	36	18
Avg. number of PILs	3.5	3.6	3.3	3.5	3.6	3.5	3.3	3.1	3.6	3.6
Day	17	18	20	21	22	23	26	27	30	
Number of PICs	38	20	19	17	17	28	2	15	10	
Avg. number of PILs	3.5	3.4	3.3	3.5	3	3.3	4	3.6	10	
Avg. number of PILs (month)	3.47									

As can be seen from Table 1, in the optimized scenario, an average of 3.47 resources are used to execute each requirement. Thus, for the month under consideration, the number of PICs using 3 PILs is 52%. A similar experiment was conducted for each month in the annual period. The overall results for the period are presented in Table 2, where the rows indicate the number of requirements for which 3 and 4 resources, respectively, are used in the optimized scenario.

Table 2

Computational Experiment

Month	4	5	6	7	8	9	10	11	12
3 PILs	62%	46%	44%	32%	37%	52%	56%	65%	61%
4 PILs	38%	54%	56%	68%	63%	48%	44%	35%	39%

According to the results presented in Table 2, the application of the developed approach leads to an increase of up to 51% on average for the number of requirements using 3 resources. In this regard, the high efficiency can be expected in other applications in the class of distribution-type problems with priority constraints. In particular, a computational experiment for arbitrary values of the parameters of model (18), as well as a comparative analysis of the results based on the performance criterion, define the directions for further testing.

Conclusion

The paper proposes a decomposition approach for solving a distribution-type planning problem with priority constraints. The proposed approach includes two stages. At the first stage, the problem with first-queue constraints related to quantitative characteristics is considered. For its solution, a basic ILP model is proposed. In order to reduce the dimension of the basic ILP model, a dynamic scheme is developed. It is proved that the resulting subset of requirements is maximal in terms of feasibility using resources with fixed adjacent priorities.

At the second stage of the decomposition approach, the second-queue constraints related to the qualitative characteristics of the solution are introduced. Based on the solution obtained at the first stage, an upper bound on the number of resources used is formed, and an additional group of Boolean variables is introduced. It is shown that for the proposed modification, there is a value of the coefficient for the additional Boolean variables such that the extended ILP model is guaranteed to be solvable. In addition, a rule for selecting this value is formulated such that the solution of the modified model determines the maximal feasible subsystem of the original planning problem with second-queue constraints.

The proposed approach is synthesized into a general solution scheme and implemented in Python 3.11. A computational experiment is conducted on an example of a distribution-type problem in the field of process planning in the mixer department of a metallurgical plant. The results of the computational experiment demonstrate the feasibility of the proposed approach and the potential to improve the objective by 10% in terms of the turnaround of the rolling stock.

Further development of the obtained results is related to testing the developed approach in the class of problems with random parameters of the basic model, as well as to analysis in terms of performance. In addition, it is of interest to consider stochastic formulations, where the qualitative characteristics of input data (requirements and resources) are random variables.

Acknowledgements. *The work was carried out with the financial support of the Russian Science Foundation (project no. 23-21-00293).*

References

1. Borisovsky P.A., Eremeev A.V. Production Scheduling of a Multi-Product Plant Using Integer Linear Programming and Evolutionary Computations. *Information Technologies*, 2015, vol. 21, no. 6, pp. 412–421.
2. Janak S.L., Floudas C.A., Kallrath J., Vormbrock N. Production Scheduling of a Large-Scale Industrial Batch Plant: Short-Term and Medium-Term Scheduling. *Industrial and Engineering Chemistry Research*, 2006, vol. 45, no. 25, pp. 8234–8252. DOI: 10.1021/ie0600588
3. Slavyanov A.S. Analysis and Practical Application of Resource Allocation Models. *Bulletin of Science and Practice*, 2018, vol. 4, no. 9, pp. 228–244.
4. Menshikh V.V., Samorokovsky A.F., Avsentev O.S. Models of Resource Allocation Optimization when Solving the Control Problems in Organizational Systems. *Journal of Physics: Conference Series*, 2018, vol. 973, no. 1, pp. 12–40. DOI: 10.1088/1742-6596/973/1/012040
5. Rezig S., Ezzeddine W., Turki S., Rezg N. Mathematical Model for Production Plan Optimization: a Case Study of Descrete Event Systems. *Mathematics*, 2020, no. 8 (6), pp. 955–971. DOI: 10.3390/math8060955
6. Ganicheva A.V., Ganichev A.V. A Method for Solving the Problem of Optimizing the Distribution of an Integer Resource. *Applied Mathematics and Control Sciences*, 2021, no. 4, pp. 42–55. DOI: 10.15593/2499-9873/2021.4.03
7. Lazarev A.A., Musatova E.G. Integer Statements of the Problem of Forming Railway Trains and Schedules of their Movement. *Management of Large Systems*, 2012, vol. 38, pp. 161–169.
8. Gainanov D.N., Ignatov A.N., Naumov A.V., Rasskazova V.A. On Track Procession Assignment Problem at the Railway Network Sections. *Automation and Remote Control*, 2020, vol. 81, no. 6, pp. 967–977. DOI: 10.1134/S0005117920060028
9. Kabulova E.G. Intelligent Management of Multi-Stage Systems of Metallurgical Production. *Modeling, Optimization, and Information Technology*, 2018, vol. 7, no. 1 (24), pp. 341–351.
10. Gitman M.B., Trusov P.V., Fedoseev S.A. On Optimization of Metal Forming with Adaptable Characteristics. *Journal of Applied Mathematics and Computing*, 2020, vol. 7, no. 2, pp. 387–396. DOI: 10.1007/BF03012200
11. Gainanov D.N., Berenov D.A. Algorithm for Predicting the Quality of the Product of Metallurgical Production. *CEUR Workshop Proceedings*, Petrovac, 2017, vol. 1987, pp. 194–200.
12. Qiu Y., Wang L., Xu X., Fang X., Pardalos P. 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.
13. Kong Min, Pei Jun, Xu Jin, Liu Xinbao, Pardalos P. 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. DOI: 10.1080/00207543.2019.1693659
14. Long Jianyu, Sun Zhenzhong, Pardalos P., Bai Yun, Zhang Shaohui, Li Chuan. A Robust Dynamic Scheduling Approach Based on Release Time Series Forecasting for the Steelmaking Continuous Casting Production. *Applied Soft Computing*, 2020, vol. 92, article ID: 106271. DOI: 10.1016/j.asoc.2020.106271

15. Gainanov D.N., Mladenovic N., Urosevic D., Rasskazova V.A. Heuristic Algorithm for Finding the Maximum Independent Set with Absolute Estimate of the Accuracy. *CEUR-WS Proceedings*, Omsk, 2018, vol. 2098, pp. 141–149.

Received May 22, 2024

УДК 519.854.3 **DOI: 10.14529/mmp240307**

ДЕКОМПОЗИЦИОННЫЙ ПОДХОД В ЗАДАЧЕ ПЛАНИРОВАНИЯ РАСПРЕДЕЛИТЕЛЬНОГО ТИПА С ПРИОРИТЕТАМИ ОГРАНИЧЕНИЙ

В.А. Рассказова, Московский авиационный институт (национальный исследовательский университет), г. Москва, Российская Федерация

В статье рассматривается задача планирования распределительного типа с приоритетами ограничений. Для заданного множества требований и ресурсов с установленными параметрами использования необходимо построить план назначений, удовлетворяющий системе приоритетных ограничений. При этом различают две очереди ограничений на количественные и качественные характеристики соответственно. На этапе решения задачи с первой очередью ограничений разрабатывается базовая модель целочисленного линейного программирования (ЦЛП) и динамическая схема ее формирования. В рамках такого подхода исходная задача сводится к решению последовательности аналогичных задач существенно меньшей размерности, что позволяет учитывать приоритеты использования ресурсов непосредственно по построению и гарантирует сходимость базовой модели ЦЛП на финальной итерации динамической схемы. На этапе реализации второй очереди ограничений для полученного базового решения вводится интегральный критерий в форме верхней оценки и рассматривается модифицированная модель ЦЛП. Процедура модификации модели опирается на метод штрафных функций и включает дооснащение системы ограничений, целевого функционала и функционального пространства подмножеством вспомогательных булевских переменных. При этом доказано, что модифицированная модель гарантировано разрешима и определяет при этом максимальную (по включению) совместную подсистему ограничений второй очереди для исходной задачи. В рамках анализа работоспособности и эффективности предложенного подхода проводится вычислительный эксперимент с использованием данных реальной размерности.

Ключевые слова: теория расписаний; целочисленное линейное программирование; дискретное производство; система планирования производственных процессов; декомпозиционный подход.

Литература

1. Борисовский, П.А. Построение расписаний многопродуктового производства с использованием целочисленного линейного программирования и эволюционных вычислений / П.А. Борисовский, А.В. Еремеев // Информационные технологии. – 2015. – Т. 21, № 6. – С. 412–421.

2. Janak, S.L. Production Scheduling of a Large-Scale Industrial Batch Plant: Short-Term and Medium-Term Scheduling / S.L. Janak, C.A. Floudas, J. Kallrath, N. Vormbrock // Industrial and Engineering Chemistry Research. – 2006. – V. 45, № 25. – P. 8234–8252.

3. Славянов, А.С. Анализ и практическое применение моделей распределения ресурсов // Бюллетень науки и практики. – 2018. – Т. 4, № 9. – С. 228–244.
4. Menshikh, V.V. Models of Resource Allocation Optimization When Solving the Control Problems in Organizational Systems / V.V. Menshikh, A.F. Samorokovsky, O.S. Avsentev // Journal of Physics: Conference Series. – 2018. – Vol. 973, № 1. – P. 12–40.
5. Rezig, S. Mathematical Model for Production Plan Optimization: a Case Study of Descrete Event Systems / S. Rezig, W. Ezzeddine, S. Turki, N. Rezg // Mathematics. – 2020. – № 8(6). – P. 955–971.
6. Ганичева, А.В. Метод решения задачи оптимизации распределения целочисленного ресурса / А.В. Ганичева, А.В. Ганичев // Прикладная математика и вопросы управления. – 2021. – № 4. – С. 42–55.
7. Lazarev, A.A. Integer Statements of the Problem of Forming Railway Trains and Schedules of Their Movement / A.A. Lazarev, E.G. Musatova // Management of Large Systems. – 2012. – V. 38. – P. 161–169.
8. Gainanov, D.N. On Track Procession Assignment Problem at the Railway Network Sections / D.N. Gainanov, A.N. Ignatov, A.V. Naumov, V.A. Rasskazova // Automation and Remote Control. – 2020. – V. 81, № 6. – P. 967–977.
9. Kabulova, E.G. Intelligent Management of Multi-Stage Systems of Metallurgical Production // Modeling, Optimization, and Information Technology. – 2018. – V. 7, № 1(24). – P. 341–351.
10. Gitman, M.B. On Optimization of Metal Forming with Adaptable Characteristics / M.B. Gitman, P.V. Trusov, S.A. Fedoseev // Journal of Applied Mathematics and Computing. – 2020. – V. 7, № 2. – P. 387–396.
11. Gainanov, D.N. Algorithm for Predicting the Quality of the Product of Metallurgical Production / D.N. Gainanov, D.A. Berenov // CEUR Workshop Proceedings. – Petrovac, 2017. – V. 1987. – P. 194–200.
12. Qiu, Y. Scheduling a Realistic Hybrid Flow Shop with Stage Skipping and Adjustable Processing Time in Steel Plants / Y. Qiu, L. Wang, X. Xu, X. Fang, P. Pardalos // Applied Soft Computing. – 2018. – V. 64. – P. 536–549.
13. Kong, Min. A Robust Optimization Approach for Integrated Steel Production and Batch Delivery Scheduling with Uncertain Rolling Times and Deterioration Effect / Min Kong, Jun Pei, Jin Xu, Xinbao Liu, P. Pardalos // International Journal of Production Research. – 2020. – V. 58, № 17. – P. 5132–5154.
14. Long, Jianyu. A Robust Dynamic Scheduling Approach Based on Release Time Series Forecasting for the Steelmaking Continuous Casting Production / Jianyu Long, Zhenzhong Sun, P. Pardalos, Yun Bai, Shaohui Zhang, Chuan Li // Applied Soft Computing. – 2020. – V. 92. – Article ID: 106271.
15. Gainanov, D.N. Heuristic Algorithm for Finding the Maximum Independent Set with Absolute Estimate of the Accuracy / D.N. Gainanov, N. Mladenovic, D. Urosevic, V.A. Rasskazova // CEUR-WS Proceedings. – Omsk, 2018. – V. 2098. – P. 141–149.

Варвара Андреевна Рассказова, кандидат физико-математических наук, кафедра «Теория вероятностей и компьютерное моделирование», Московский авиационный институт (национальный исследовательский университет) (г. Москва, Российская Федерация), varvara.rasskazova@mail.ru.

Поступила в редакцию 22 мая 2024 г.