ISSN 0005-1179, Automation and Remote Control, 2016, Vol. 77, No. 4, pp. 656–671. © Pleiades Publishing, Ltd., 2016.
Original Russian Text © A.A. Lazarev, D.I. Arkhipov, 2016, published in Avtomatika i Telemekhanika, 2016, No. 4, pp. 134–152.

INTELLECTUAL CONTROL SYSTEMS

Minimization of the Maximal Lateness for a Single Machine

A. A. Lazarev[*,**,***,****] **and D. I. Arkhipov**[*]

[*] *Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia*
[**] *National Research University Higher School of Economics, Moscow, Russia*
[***] *Lomonosov State University, Moscow, Russia*
[****] *Moscow Physical and Technical Institute, Dolgoprudnyi, Russia*
e-mail: jobmath@mail.ru, miptrafter@gmail.com

Received February 20, 2015

Abstract—Consideration was given to the classical NP-hard problem $1|r_j|L_{\max}$ of the scheduling theory. An algorithm to determine the optimal schedule of processing n jobs where the job parameters satisfy a system of linear constraints was presented. The polynomially solvable area of the problem $1|r_j|L_{\max}$ was expanded. An algorithm was described to construct a Pareto-optimal set of schedules by the criteria $L_{\max}$ and $C_{\max}$ for complexity of $O(n^3 \log n)$ operations.

DOI: 10.1134/S000511791604010X

1. INTRODUCTION

Consideration was given to the following problem of the scheduling theory: needed is to process the jobs of the set $N = \{1, \ldots, n\}$ starting from the time instant t. Interrupts of job processing and simultaneous processing of more than one job are forbidden.

For the jobs of the set N, the following notation is introduced: r_j is the *release date* of the job j, $p_j > 0$ is the processing time of the jth job d_j is the due date, $j \in N$. The *due date* is the time during which it is desirable, but not necessary, to complete job processing. We denote $r_j(t) = \max\{r_j, t\}$, $j \in N$. By the *schedule* $\pi(N,t)$ is meant the sequence of processing the jobs of the set N

$$\pi(N,t) = (K_1, \ldots, K_n)$$

beginning from the time instant t, where $K_1 \cup \cdots \cup K_n \equiv N$ and processing of the job K_1 begins at the time instant $s_1 = r_{K_1}(t)$, the rest of the job K_j $(j = 2, \ldots, n)$ being processed from the time instant $s_{K_j} = r_{K_j}(s_{K_{j-1}} + p_{K_{j-1}})$. A schedule is called *feasible* if beginning from the time instant $s_j \geqslant r_j(t)$ each job $j \in N$ is processed without interrupts over the time p_j and no two jobs are processed concurrently. The set of all feasible schedules that are constructible for the set of jobs N and the time instant t is denoted by $\Pi(N,t)$. By $C_j(\pi,t)$ we denote the time of completing processing of each job $j \in N$ under the schedule $\pi \in \Pi(N,t)$. The difference $L_j(\pi,t) = C_j(\pi,t) - d_j$ is called the *lateness* of the job j under the schedule π starting at the time instant t (Fig. 1). The maximal lateness for the jobs of the set N under the schedule π is given by

$$L_{\max}(\pi,t) = \max_{j \in N} C_j(\pi,t) - d_j.$$

The completion time of all jobs of N under the schedule π is denoted by

$$C_{\max}(\pi,t) = \max_{j \in N} C_j(\pi,t).$$

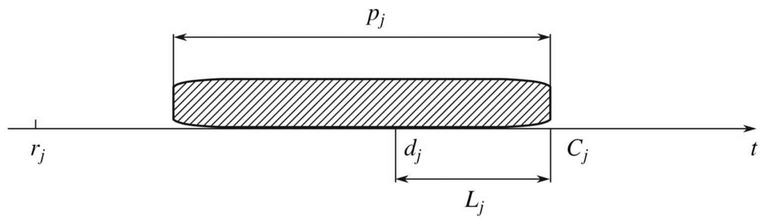

Fig. 1. Parameters of the job j.

Problem 1. For the given set of jobs N and time instant t, construct the schedule $\pi^* \in \Pi(N,t)$ for which

$$L_{\max}(\pi^*,t) = \min_{\pi\in\Pi(N,t)} L_{\max}(\pi,t).$$

In [1] this classical problem of the scheduling theory was denoted by $1|r_j|L_{\max}$. As was shown in [2], the general case of the problem $1|r_j|L_{\max}$ is NP-hard in the strong sense. Some polynomially solvable cases were specified since the time of formulating this problem. As was shown in [3], in the case of $r_j = 0,\ j \in N$, the problem is solved by the schedule where the jobs are arranged in the nondecreasing order of due dates. Such schedule is also optimal for the case where the release date and the due dates are coordinated by $r_i \leqslant r_j \Leftrightarrow d_i \leqslant d_j, \forall i,j \in N$. In the case of $d_j = d$ for all $j \in N$, the optimal schedule can also be constructed in $O(n\log n)$ operations in the nondecreasing release dates. For the case of equal processing times of the jobs $p_j = p$ for all $j \in N$, the polynomial algorithm of complexity $O(n^2\log n)$ was given in [4]. A polynomial algorithm of complexity of $O(n^2\log n)$ operations was described in [5] for a special case where for some constant A the parameters of all jobs $j \in N$ satisfy the constraints

$$d_j - p_j - A \leqslant r_j \leqslant d_j - A,\ \ \forall j \in N.$$

For the case where the job parameters satisfy the system of linear constraints

$$\begin{cases} d_1 \leqslant \cdots \leqslant d_n \\ d_1 - p_1 - r_1 \geqslant \cdots \geqslant d_n - p_n - r_n, \end{cases}$$

the polynomial algorithm of complexity of $O(n^3\log n)$ operations was presented in the book of one of the present authors [6].

2. PROPERTIES OF THE PROBLEM

We consider the case where for some real numbers $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$ the parameters of the jobs of the set N satisfy the inequality system

$$\begin{cases} d_1 \leqslant \cdots \leqslant d_n \\ d_1 - \alpha p_1 - \beta r_1 \geqslant \cdots \geqslant d_n - \alpha p_n - \beta r_n \end{cases} \tag{1}$$

and recall that t is the time instant from which the machine is available for processing the jobs. From the set N we take two jobs $f = f(N,t)$ and $s = s(N,t)$ such that

$$f(N,t) = \arg\min_{j\in N}\left\{d_j | r_j(t) = \min_{i\in N} r_i(t)\right\},$$

$$s(N,t) = \arg\min_{j\in N\setminus f}\left\{d_j | r_j(t) = \min_{i\in N\setminus f} r_i(t)\right\}.$$

If $N = \emptyset$, that is, $|N| = 0$, we assume for any real t that

$$f(\emptyset, t) = 0, \qquad s(\emptyset, t) = 0.$$

If $N = \{i\}$, that is, $|N| = 1$, we assume for any real t that

$$f(N, t) = i, \qquad s(N, t) = 0.$$

Denote by $(i \to j)_\pi$ the fact that under the schedule π the job i is processed prior to job j.

Lemma 1. *If for the job of the set N conditions (1) are satisfied for some $\alpha \in [0, 1]$ and $\beta \in [0, +\infty)$, then it is true that*

$$L_j(\pi, t) < L_f(\pi, t) \tag{2}$$

under any schedule $\pi \in \Pi(N, t)$ and for any job $j \in N \setminus \{f\}$ such that $(j \to f)_\pi$, and for any job $j \in N \setminus \{f, s\}$ such that $(j \to s)_\pi$ it is true that

$$L_j(\pi, t) < L_s(\pi, t), \tag{3}$$

where $f = f(N, t)$, $s = s(N, t)$.

Proof of Lemma 1. For all jobs j such that $(j \to f)_\pi$, the inequality

$$C_j(\pi, t) \leqslant C_f(\pi, t) - p_f$$

is satisfied. If $d_j \geqslant d_f$, we have

$$L_j(\pi, t) = C_j(\pi, t) - d_j < C_f(\pi, t) - d_f = L_f(\pi, t),$$

consequently, (2) is satisfied.

Let us consider the case where $d_j < d_f$ is true for the job $j \in N$, $(j \to f)_\pi$. From system (1) we get

$$d_j < d_f \Leftrightarrow d_j - \alpha p_j - \beta r_j \geqslant d_f - \alpha p_f - \beta r_f.$$

Then, we establish with regard for $r_j > r_f$, $\alpha \in [0, 1]$, $\beta \in [0, \infty)$, and $p > 0$ that

$$0 \leqslant \alpha p_j + (1 - \alpha) p_f + \beta(r_j - r_f) \Leftrightarrow \alpha p_f + \beta r_f \leqslant \alpha p_j + \beta r_j + p_f.$$

With allowance made for the fact that

$$d_j - \alpha p_j - \beta r_j \geqslant d_f - \alpha p_f - \beta r_f \Leftrightarrow (\alpha p_f + \beta r_f) - (\alpha p_j + \beta r_j) \geqslant d_f - d_j,$$

we establish

$$d_f \leqslant d_j + p_f.$$

Obviously, $C_j(\pi, t) \leqslant C_f(\pi, t) - p_f$. By adding the resulting inequalities, we obtain that

$$C_j(\pi, t) + d_f \leqslant C_f(\pi, t) + d_j \Leftrightarrow L_j(\pi, t) \leqslant L_f(\pi, t),$$

which proves statement (2).

Statement (3) is proved along the same lines. It suffices to notice that the job s from the set N becomes job f in the set $N \setminus \{f\}$ and one has just to replace N by $N \setminus \{f\}$.

We prove the following theorem about the properties of the jobs f and s.

Theorem 1. *Let all jobs of the subset $N' \subseteq N$ satisfy the system of inequalities (1) for some $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$. Then, for any time instant $t' \geqslant t$ and any schedule $\pi \in \Pi(N',t')$ there exists a schedule $\pi' \in \Pi(N',t')$ such that*

$$\begin{cases} L_{\max}(\pi',t') \leqslant L_{\max}(\pi,t') \\ C_{\max}(\pi',t') \leqslant C_{\max}(\pi,t') \end{cases} \tag{4}$$

and either the job $f = f(N',t')$ or $s = s(N',t')$ is satisfied first under the schedule π'. If $d_f \leqslant d_s$, then in π' the job f is processed first.

Proof of Theorem 1. Let $\pi = (\pi_1, f, \pi_2, s, \pi_3)$, where π_1, π_2, π_3 are partial subschedules of π. Consider the schedule $\pi' = (f, \pi_1, \pi_2, s, \pi_3)$. We establish from the definitions of $r_j(t)$ and $f(N,t)$ for each job $j \in N'$ that

$$r_f(t') \leqslant r_j(t').$$

Consequently,

$$C_{\max}((f,\pi_1),t') \leqslant C_{\max}((\pi_1,f),t'),$$
$$C_{\max}(\pi',t') \leqslant C_{\max}(\pi,t').$$

Therefore,

$$L_j(\pi',t') \leqslant L_j(\pi,t'), \ \ \forall j \in (\pi_2, s, \pi_3).$$

From Lemma 1 we obtain that

$$L_j(\pi',t') \leqslant L_s(\pi,t')$$

for any $j \in \{\pi_1\} \cup \{\pi_2\}$. Obviously, for f we have

$$L_f(\pi',t') \leqslant L_f(\pi,t')$$

from which it follows that

$$\begin{cases} L_{\max}(\pi',t') \leqslant L_{\max}(\pi,t') \\ C_{\max}(\pi',t') \leqslant C_{\max}(\pi,t'). \end{cases}$$

Let $\pi = (\pi_1, s, \pi_2, f, \pi_3)$, that is, the job s is executed prior to the job f. In this case, we construct the schedule $\pi' = (s, \pi_1, \pi_2, f, \pi_3)$ and repeat the proof along the same lines as above, which proves the first part of the theorem.

Assume that $d_f \leqslant d_s$ and $\pi = (\pi_1, s, \pi_2, f, \pi_3)$ and consider the schedules $\pi' = (s, \pi_1, \pi_2, f, \pi_3)$ and $\pi'' = (f, \pi_1, \pi_2, s, \pi_3)$. Then, for the schedules π, π' and π'' the inequality

$$C_{\max}((f,\pi_1,\pi_2,s),t') \leqslant C_{\max}((s,\pi_1,\pi_2,f),t')$$

is true because $r_f(t') \leqslant r_s(t')$. Consequently,

$$L_{\max}(\pi_3, C_{\max}((f,\pi_1,\pi_2,s),t')) \leqslant L_{\max}(\pi_3, C_{\max}((s,\pi_1,\pi_2,f),t')),$$

and therefore, the maximum of the objective function $L_{\max}((f,\pi_1,\pi_2,s),t')$ is reached for a job other than f. The maximum of the objective function $L_{\max}((s,\pi_1,\pi_2,f),t')$ cannot be reached for the job s because $d_f \leqslant d_s$ and $C_s((s,\pi_1,\pi_2,f),t') < C_f((s,\pi_1,\pi_2,f),t')$. Then, by Lemma 1,

$$L_{\max}((f,\pi_1,\pi_2,s),t') = L_s((f,\pi_1,\pi_2,s),t') = C_{\max}((f,\pi_1,\pi_2,s),t') - d_s$$

and

$$L_{\max}((s,\pi_1,\pi_2,f),t') = L_f((s,\pi_1,\pi_2,f),t') = C_{\max}((s,\pi_1,\pi_2,f),t') - d_f.$$

Therefore, from the fact that $d_f \leqslant d_s$ and $C_{\max}((f,\pi_1,\pi_2,s),t') \leqslant C_{\max}((s,\pi_1,\pi_2,f),t')$, we determine that

$$L_{\max}((f,\pi_1,\pi_2,s),t') \leqslant L_{\max}((s,\pi_1,\pi_2,f),t')$$

and

$$L_{\max}(\pi'', t') \leqslant L_{\max}(\pi', t'),$$

which is what we set out to prove.

The schedule $\pi' \in \Pi(N,t)$ is called *efficient* if there exists no schedule $\pi \in \Pi(N,t)$ such that the inequality system

$$\begin{cases} L_{\max}(\pi, t) \leqslant L_{\max}(\pi', t) \\ C_{\max}(\pi, t) \leqslant C_{\max}(\pi', t) \end{cases}$$

is satisfied, at least one of these inequalities being strict. Then, if for the jobs of the set N the inequality system (1) is true under certain $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$, then it follows from Theorem 1 that there exists an efficient schedule π' under which either the job $f = f(N,t)$ or $s = s(N,t)$ is executed first. Moreover, if $d_f \leqslant d_s$, then there exists an optimal schedule for which the job f is executed first.

Let $\Omega(N,t)$ be a subset of the set $\Pi(N,t)$. The schedule $\pi = (i_1, \ldots, i_n)$ belongs to $\Omega(N,t)$ if any job i_k, $k = 1, \ldots, n$, is selected from

$$f_k = f(N_{k-1}, C_{i_{k-1}}(\pi, t)) \quad \text{and} \quad s_k = s(N_{i_{k-1}}, C_{i_{k-1}}(\pi, t)),$$

where $N_{k-1} = N \setminus \{i_1, \ldots, i_{k-1}\}$, $N_0 = N$ and $C_{i_0}(\pi, t) = t$. If $d_{f_k} \leqslant d_{s_k}$, then $i_k = f_k$. If $d_{f_k} > d_{s_k}$, then either $i_k = f_k$ or $i_k = s_k$. Since at most two jobs claim for each place under the schedule, the set $\Omega(N,t)$ contains at most 2^n schedules. According to Theorem 1, it is always possible to construct an efficient schedule belonging to the set $\Omega(N,t)$ by enumerating at most 2^n variants.

Let $\omega(N,t)$ be a partial maximum-length schedule such that by considering successively the job we have $d_f \leqslant d_s$. The schedule $\omega(N,t)$ can be constructed by Algorithm 1 for any set of jobs N and time t.

Algorithm 1

Data: N, t
Result: $\omega(N,t)$
1 $N' := N$;
2 $t' := t$;
3 $f := f(N', t')$;
4 $s := s(N', t')$;
5 **if** $d_f \leq d_s$ **then**
6 | $\omega = (\omega, f)$;
7 **else**
8 | *return*(ω);
9 **end**
10 $N' := N' \setminus f$;
11 $t' := r_f(t') + p_f$;
12 **if** $N \neq \emptyset$ **then**
13 | *go to step 3;*
14 **else**
15 | *return*(ω);
16 **end**

Algorithm 1 lies in that at each run of the cycle 5–13 consideration is given to the jobs $f(N', t')$ and $s(N', t')$. If $d_f \leqslant d_s$, under the schedule ω the job $f(N', t')$ is processed from the time instant $r_f(t')$ to the time instant $r_f(t') + p_f$. The job $f(N', t')$ is eliminated from the set N', $t' := r_f(t') + p_f$

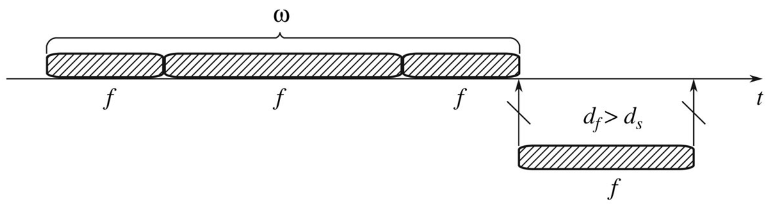

Fig. 2. Construction of the schedule $\omega(N,t)$.

is changed, and the cycle is repeated. If $d_f > d_s$, then the algorithm interrupts its operation and outputs the constructed partial schedule $\omega(N,t)$ (Fig. 2). If $d_f > d_s$, $f = f(N,t)$, $s = s(N,t)$, then $\omega(N,t) = \emptyset$.

Lemma 2. *Complexity of constructing the partial schedule $\omega(N,t)$ by Algorithm 1 for any N and t is at most $O(n \log n)$ operations.*

Proof of Lemma 2. Two jobs $f(N',t')$ and $s(N',t')$ are determined at steps 3–4 of Algorithm 1. Since the jobs are sorted out according to the release dates r_j, at most $O(\log n)$ operations are required to determined the jobs f and s. In view of the fact that the number of runs of the cycle 3–13 is restricted by the cardinality of the set N, we find that at most $O(n \log n)$ operations are required to construct the partial schedule $\omega(N,t)$.

Lemma 3. *If the jobs of the set N satisfy the conditions* (1) *for some $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$, then any schedule $\pi \in \Omega(N,t)$ begins from the partial schedule $\omega(N,t)$.*

Proof of Lemma 3. If $\omega(N,t) = \emptyset$, the condition of lemma is satisfied in view of the fact that any schedule begins from an empty schedule. If $\omega(N,t) = (i_1,\ldots,i_l)$, then $d_{f_k} \leqslant d_{s_k}$, where $f_k = f(N_{k-1}, C_{k-1}(\pi,t))$ and $s_k = s(N_{k-1}, C_{k-1}(\pi,t))$, is satisfied for any $k = 1,\ldots,l$. At the same time for $f = f(N_l, C_l(\pi,t))$ and $s = s(N_l, C_l(\pi,t))$ we have $d_f > d_s$. In view of the established relations between the due dates and definition of $\Omega(N,t)$, we get that any schedule from $\Omega(N,t)$ begins from $\omega(N,t)$, which is what we set out to prove.

We denote

$$\omega^1(N,t) = (f(N,t), \omega(N \setminus f, t'))$$

and

$$\omega^2(N,t) = (s(N,t), \omega(N \setminus s, t'')),$$

where $t' = r_f(t) + p_f$ and $t'' = r_s(t) + p_s$. We notice that as follows from the definition of $\omega^1(N,t)$ and $\omega^2(N,t)$ and Lemma 2, $O(n \log n)$ operations are also required to determine them.

Corollary. *If the jobs of the set N satisfy conditions* (1) *for some $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$, then any schedule $\pi \in \Omega(N,t)$ begins either from $\omega^1(N,t)$ or $\omega^2(N,t)$.*

3. PROBLEM OF MAKESPAN MINIMIZATION WITH RESTRICTED MAXIMAL LATENESS

Problem 2. Order the set of jobs N since the time instant t so that the maximal lateness be at most y. Needed is to establish the optimal schedule satisfying

$$\min_{\pi \in \Pi(N,t)} C_{\max}(\pi,t) | L_{\max}(\pi,t) \leqslant y.$$

The schedule satisfying the given objective function is denoted by $\Theta(N,t,y)$. If there is no such schedule $\pi \in \Pi(N,t)$, we state that $\Theta(N,t,y) = \emptyset$.

We represent an algorithm to construct the schedule $\Theta(N,t,y)$.

Algorithm 2

Data: N, t, y
Result: $\Theta(N, t, y)$
1 $\Theta := \omega(N, t)$;
2 **if** $L_{\max}(\omega(N, t), t) > y$ **then**
3   *return*$(\emptyset)$;
4 **end**
5 **while** *1* **do**
6     $N' := N \setminus \Theta$;
7     $t' := C_{\max}(\Theta, t)$;
8   **if** $N' = \emptyset$ **then**
9   *return*(Θ);
10   **end**
11   **if** $L_{\max}(\omega^1(N', t'), t') \leqslant y$ **then**
12         $\Theta := (\Theta, \omega^1(N', t'))$;
13   **else**
14   **if** $L_{\max}(\omega^2(N', t'), t') \leqslant y$ **then**
15             $\Theta := (\Theta, \omega^2(N', t'))$;
16   **else**
17   *return*$(\emptyset)$;
18   **end**
19   **end**
20 **end**

The first step of the algorithm constructs a partial schedule $\omega(N, t)$, includes it in $\Theta(N, t, y)$, and modifies $N' := N' \setminus \Theta$ and $t' := C_{\max}(\Theta, t)$. Now, the partial schedule $\omega^1(N', t')$ is constructed, and the constraint $L_{\max}(\omega^1(N', t'), t') \leqslant y$ is verified. In the case of positive result, $\omega^1(N', t')$ is added to the schedule $\Theta(N, t, y)$. Then, N' and t' are modified, and the cycle 5–20 is iterated. Otherwise, the partial schedule $\omega^2(N', t')$ is constructed, and the constraint $L_{\max}(\omega^2(N', t'), t') \leqslant y$ is verified. In the case of positive result, $\omega^2(N', t')$ is added to the schedule $\Theta(N, t, y)$, N' and t' are modified, and the procedure is repeated (Fig. 3). The algorithm aborts is all jobs of the set N are successfully included in the schedule Θ or if at some step both schedules $\omega^1(N', t')$ and $\omega^2(N', t')$ do not satisfy the constraint on the maximal lateness. In this case, the algorithm returns $\Theta(N, t, y) = \emptyset$.

Lemma 4. *Complexity of Algorithm 2 does not exceed $O(n^2 \log n)$ operations.*

Proof of Lemma 4. At steps 1, 11, and 14 of Algorithm 2 the schedules $\omega^1(N', t')$ and $\omega^2(N', t')$ are constructed using Algorithm 1. This requires at most $O(n \log n)$ operations. As the result of running cycle 5–20, at least one job is added to the schedule Θ or $\Theta = \emptyset$ is returned. Consequently, cycle 5–20 is iterated at most n times. Therefore, Algorithm 2 establishes the schedule $\Theta(N, t, y)$ at most in $O(n^2 \log n)$ operations.

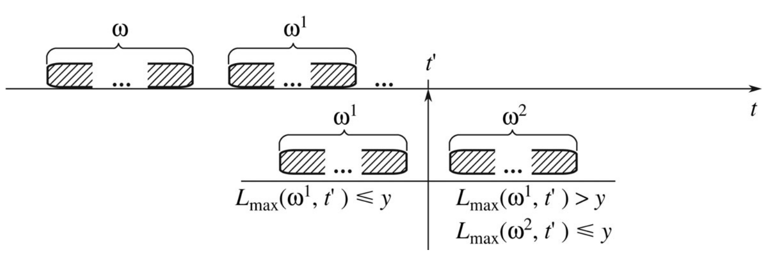

Fig. 3. Construction of the schedule Θ.

We prove a theorem about the properties of the constructed schedule $\Theta(N, t, y)$.

Theorem 2. *Let for the jobs of the set N satisfied be conditions (1) for some $\alpha \in [0, 1]$ and $\beta \in [0, +\infty)$. If the schedule $\Theta(N, t, y)$ established using Algorithm 2 is not empty, then*

$$C_{\max}(\Theta(N, t, y), t) \leqslant C_{\max}(\pi, t)$$

is satisfied for any schedule $\pi \in \Pi(N, t)$ meeting the constraint $L_{\max}(\pi, t) \leqslant y$. If $\Theta(N, t, y) = \emptyset$, then

$$L_{\max}(\pi, t) > y$$

is true for any schedule $\pi \in \Pi(N, t)$.

Proof of Theorem 2. Since conditions (1) are satisfied for any $\pi \in \Pi(N, t)$, according to Theorem 1 there exists a schedule $\pi' \in \Omega(N, t)$ such that

$$\begin{cases} L_{\max}(\pi', t) \leqslant L_{\max}(\pi, t) \\ C_{\max}(\pi', t) \leqslant C_{\max}(\pi, t). \end{cases}$$

It follows from the construction of the schedule $\Theta(N, t, y)$ that it pertains to the set $\Omega(N, t)$. Then it follows from Lemma 3 that the schedule $\Theta(N, t, y)$ begins from the partial schedule $\omega(N, t)$ denoted by $\Theta_0 = \omega(N, t)$.

The partial schedule Θ_k is obtained after k runs of cycle 5–20. At that, $N' = N \setminus \{\Theta_k\}$ and $t' = C_{\max}(\Theta_k, t)$. We assume that there exists a schedule Θ having minimal $C_{\max}$, beginning with the partial schedule Θ_k, and satisfying the constraint $L_{\max}(\Theta_k, t) \leqslant y$. Then, by Lemma 3 Θ_k can be continued by a schedule from the set $\Omega(N', t')$. Three cases are feasible here.

1. Let $\Theta_{k+1} = (\Theta_k, \omega^1(N', t'))$, that is, $L_{\max}(\omega^1(N', t'), t') \leqslant y$. Then, $\omega^1(N', t')$ is a partial schedule with the least $C_{\max}$ among all possible continuations of the schedule Θ_k satisfying $L_{\max}(\Theta_{k+1}, t) \leqslant y$.
2. If $\Theta_{k+1} = (\Theta_k, \omega^2(N', t'))$, then

$$\begin{cases} L_{\max}(\omega^1(N', t'), t') > y \\ L_{\max}(\omega^2(N', t'), t') \leqslant y, \end{cases}$$

   which follows from the fact that any schedule from the set $\Omega(N', t')$ can start either with $\omega^1(N', t')$ or with $\omega^2(N', t')$ and $L_{\max}(\omega^1(N', t'), t') > y$. As follows from steps 11–19, this case is possible only if $\omega^2(N', t')$ is a unique possible continuation of the schedule Θ_k.
3. Now we consider the case where after k runs of cycle 5–20 we have $L_{\max}(\omega^1(N', t'), t') > y$ and $L_{\max}(\omega^2(N', t'), t') > y$. It follows from the assumption that if the schedule $\Theta \in \Omega(N, t)$ exists, then it must necessarily begin with Θ_k. Additionally, for any $\pi \in \Pi(N', t')$ there always exists $\pi' \in \Omega(N', t')$ such that either

$$L_{\max}(\pi, t') \geqslant L_{\max}(\pi', t') \geqslant L_{\max}(\omega^1(N', t'), t') > y$$

   or

$$L_{\max}(\pi, t') \geqslant L_{\max}(\pi', t') \geqslant L_{\max}(\omega^2(N', t'), t') > y.$$

   Consequently, $\Theta = \emptyset$.

Therefore, the desired schedule $\Theta(N, t, y)$ is constructed at most after n runs of cycle 5–20. If case 3 arises at least once, then the schedule $\Theta(N, t, y)$ does not exist at all, which is what we set out to prove.

4. ALGORITHM TO CONSTRUCT A SET OF PARETO-OPTIMAL SCHEDULES BY THE CRITERIA C_{MAX} AND L_{MAX}

We present below an algorithm determining the Pareto-set of schedules $\Phi(N, t)$ such that $1 \leqslant |\Phi(N, t)| \leqslant n$ for any set of jobs N and time instant t. The optimal Pareto-set is constructed if condition 1 is satisfied.

Algorithm 3

Data: N, t
Result: $\Phi(N, t)$
1 $y := +\infty$;
2 $\pi^* := \omega(N, t)$;
3 $\Phi := \emptyset$;
4 $m := 0$;
5 **while** *1* **do**
6 $N' := N \setminus \pi^*$;
7 $t' := C_{\max}(\pi^*, t)$;
8 **if** $N' = \emptyset$ **then**
9 $\Phi := \Phi \cup \{\pi^*\}$;
10 $return(\Phi)$;
11 **end**
12 **if** $L_{\max}(\omega^1(N', t'), t') \leqslant L_{\max}(\pi^*, t)$ **then**
13 $\pi^* := (\pi^*, \omega^1(N', t'))$;
14 **else**
15 **if** $L_{\max}(\omega^1(N', t'), t') \leqslant y$ **then**
16 $y' := L_{\max}(\omega^1(N', t'), t')$;
17 $\Theta := \Theta(N', t', y')$;
18 **if** $\Theta = \emptyset$ **then**
19 $\pi^* := (\pi^*, \omega^1(N', t'))$;
20 **else**
21 $\pi' := (\pi^*, \Theta)$;
22 **if** $(m = 0)$ *or* $(C_{\max}(\pi'_m, t) < C_{\max}(\pi', t))$ **then**
23 $m := m + 1$;
24 $\pi'_m := \pi'$;
25 $\Phi := \Phi \cup \{\pi'_m\}$;
26 $y := L_{\max}(\pi'_m, t)$;
27 **else**
28 $\pi'_m := \pi'$;
29 **end**
30 **end**
31 **else**
32 **if** $L_{\max}(\omega^2(N', t'), t') \leqslant y$ **then**
33 $\pi^* := (\pi^*, \omega^2(N', t'))$;
34 **else**
35 $\pi^* := \pi'_m$;
36 $return(\Phi)$.
37 **end**
38 **end**
39 **end**
40 **end**

Algorithm 3 runs as follows. By Lemma 3, any schedule from $\Omega(N, t)$ that is optimal for the criterion $L_{\max}$ begins with $\omega(N, t)$. Therefore, we denote $\pi_0 = \omega(N, t)$ and consider operation of cycle 5–40. Let the partial schedule $\pi^* = \pi_k$ and the set $\Phi = \{\pi'_1, \ldots, \pi'_m\}$ be constructed after k first runs of cycle 5–40 of Algorithm 3, and let $N' = N \setminus \{\pi_k\}$ and $t' = C_{\max}(\pi_k, t)$ be the values obtained at steps 5 and 6 during the $(k+1)$st run of cycle 5–40. We consider possible continuations of the schedule π_k.

1. If $L_{\max}(\omega^1(N',t'),t') \leqslant L_{\max}(\pi_k,t)$, then carried out are the assignment $\pi^*{:=}\pi_{k+1} = (\pi_k, \omega^1(N',t'))$ and the return to step 5 after selecting the optimal continuation for the criterion $C_{\max}$ without violating the current value of the objective function $L_{\max}(\pi_{k+1},t) \leqslant L_{\max}(\pi_k,t)$.
2. $L_{\max}(\pi_k,t) < L_{\max}(\omega^1(N',t'),t') \leqslant y$ and $\Theta(N',t',L_{\max}(\omega^1(N',t'),t')) = \emptyset$. In this case, carried out are the assignment $\pi^* := \pi_{k+1} = (\pi_k, \omega^1(N',t'))$ and the return to step 5 after selecting the optimal continuation for the criterion $C_{\max}$ without violating $L_{\max}(\pi_{k+1},t) \leqslant y$.
3. $L_{\max}(\pi_k,t) < L_{\max}(\omega^1(N',t'),t') \leqslant y$ and $\Theta(N',t',L_{\max}(\omega^1(N',t'),t')) \neq \emptyset$. The assignment $\pi'{:=}(\pi^*, \Theta(N',t',L_{\max}(\omega^1(N',t'),t')))$ is carried out. Since $y' \leqslant y$, the schedule π' satisfies the constraint $L_{\max}(\pi_{k+1},t) \leqslant y$. If the value of $C_{\max}(\pi',t)$ increased as compared with $C_{\max}(\pi'_m,t)$, then the counter is incremented by $m := m+1$, inclusion of π' in the set Φ is performed, and the constraint y is modified (steps 23–26). If we have $C_{\max}(\pi',t) \leqslant C_{\max}(\pi'_m,t)$, then the schedule π'_m is replaced by π' in the set Φ (step 28). After any possible outcome, the return to step 5 is performed.
4. $L_{\max}(\omega^1(N',t'),t') > y$, $L_{\max}(\omega^2(N',t'),t') \leqslant y$. The unique possible variant of continuation π_k without violation of the constraint $L_{\max}(\pi_{k+1},t) \leqslant y$ (step 33), the assignment $\pi^* := \pi_{k+1} = (\pi_k, \omega^2(N',t'))$ is performed, followed by the passage to step 5.
5. $L_{\max}(\omega^1(N',t'),t') > y$, $L_{\max}(\omega^2(N',t'),t') > y$. It is impossible to continue the schedule π_k without violating the constraint $L_{\max}(\pi_{k+1},t) \leqslant y$. Execution of the system is interrupted (step 36). The algorithm completes operation if all jobs of the set N are included in the schedule π^* or if there is not way to continue the schedule π^* without violating the constraint y (step 36).

Lemma 5. *The complexity of Algorithm 3 is less than or equal to $O(n^3 \log n)$ operations, and the cardinality of the set $\Phi(N,t)$ does not exceed n.*

Proof of Lemma 5. Constructions of the partial schedules $\omega^1(N',t')$ and $\omega^2(N',t')$ and Θ represent the most laborious operations at running cycle 5–40. Determination of $\omega^1(N',t')$ and $\omega^2(N',t')$ requires $O(n \log n)$ s, and the schedule Θ needs $O(n^2 \log n)$ operations. Since the partial schedules $\omega^1(N',t')$ and $\omega^2(N',t')$ consist of at least one job, at least one job is added to the partial schedule π^* at each run of the cycle, and the set $\Phi(N,t)$ includes at most one schedule. Consequently, the number of runs of cycle 5–40 of Algorithm 3 is less than or equal to n. Therefore, the cardinality of the set $\Phi(N,t)$ does not exceed n and the total number of operations is less than or equal to $O(n^3 \log n)$.

Theorem 3. *Let the conditions* (1) *be satisfied for the jobs of the set N under some $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$. Then, the schedule π^* constructed by Algorithm 3 is optimal for the criterion $L_{\max}$.*

For any schedule $\pi \in \Pi(N,t)$, there exists $\pi' \in \Phi(N,t)$ such that

$$\begin{cases} L_{\max}(\pi',t) \leqslant L_{\max}(\pi,t) \\ C_{\max}(\pi',t) \leqslant C_{\max}(\pi,t) \end{cases}$$

and the set of schedules $\Phi(N,t)$ is Pareto-optimal for the criteria $L_{\max}$ and $C_{\max}$.

Proof of Theorem 3. Let us assume that there exists a schedule $\pi \in \Pi(N,t)$ not belonging to $\Phi(N,t)$ and for which at least one of the inequalities

$$C_{\max}(\pi,t) < C_{\max}(\pi',t) \tag{5}$$

or

$$L_{\max}(\pi,t) < L_{\max}(\pi',t) \tag{6}$$

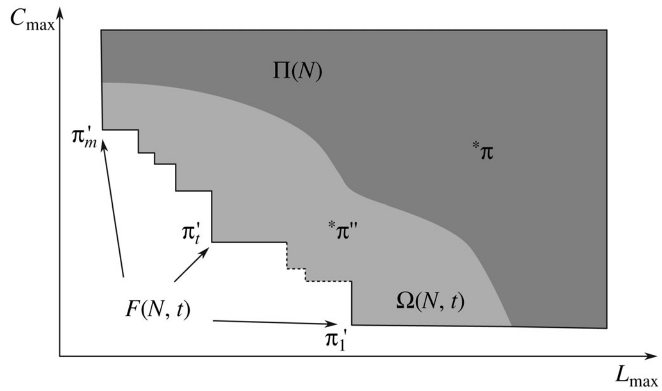

Fig. 4. Set of Pareto-optimal schedules

is satisfied for any schedule $\pi' \in \Phi(N,t)$. According to Theorem 1, there exists a schedule $\pi'' \in \Omega(N,t)$ such that

$$\begin{cases} L_{\max}(\pi'',t) \leqslant L_{\max}(\pi,t) \\ C_{\max}(\pi'',t) \leqslant C_{\max}(\pi,t). \end{cases}$$

If $\pi'' \in \Phi(N,t)$, then, obviously, none of conditions (5) and (6) can be satisfied. Consequently, $\pi'' \in \Omega(N,t) \setminus \Phi(N,t)$.

As follows from the definition of the set $\Omega(N,t)$, any schedule π'' from the set $\Omega(N,t)$ is representable as a union of the partial schedules $\pi'' = (\omega_0, \omega_1, \ldots, \omega_{k''})$, where $\omega_0 = \omega(N,t)$ and ω_i is either $\omega^1(N''_i, C''_i)$ or $\omega^2(N''_i, C''_i)$ and $N''_i = N \setminus \{\omega_0, \ldots, \omega_{i-1}\}$, $C''_i = C_{\max}((\omega_0, \ldots, \omega_{i-1}), t)$, $i = 1, \ldots, k''$.

The schedule π' is structured similarly because $\Phi(N,t) \subseteq \Omega(N,t)$, that is, $\pi' = (\omega'_0, \omega'_1, \ldots, \omega'_{k'})$, where $\omega'_0 = \omega(N,t)$ and ω'_i is either $\omega^1(N'_i, C'_i)$ or $\omega^2(N'_i, C'_i)$ and $N'_i = N \setminus \{\omega'_0, \ldots, \omega'_{i-1}\}$, $C'_i = C_{\max}((\omega'_0, \ldots, \omega'_{i-1}), t)$, $i = 1, \ldots, k'$.

Let us assume that k first partial schedules π' and π'' coincide, that is, $\omega'_i = \omega_i \ \forall i = 0, \ldots, k-1$, and $\omega'_k \neq \omega_k$. We assume that $y = L_{\max}(\omega_0, \ldots, \omega_{k-1}, t)$, $N_k = N'_k = N''_k$ and $C_k = C'_k = C''_k$ and construct the schedule $\Theta = \Theta(N_k, C_k, y)$ using Algorithm 2. If $\Theta = \emptyset$, according to Algorithm 3 we establish that $\omega'_k = \omega^1(N_k, C_k)$. Since $\omega_k \neq \omega'_k$, we obtain that $\omega_k = \omega^2(N_k, C_k)$. The condition $L_{\max}(\omega^2(N_k, C_k), C_k) \leqslant y$ cannot be satisfied because $\Theta = \emptyset$. The entire structure of Algorithm 3 is built around the idea of arranging the jobs as densely as possible until a job with the critical $L_{\max}$ occurs. Consequently, by continuing the partial schedule $\omega^1(N_k, C_k)$ we obtain

$$\begin{cases} C_{\max}(\pi',t) \leqslant C_{\max}(\pi'',t) \\ L_{\max}(\pi',t) \leqslant L_{\max}(\pi'',t). \end{cases}$$

In the case of $\Theta \neq \emptyset$, we have for the schedule $\pi' = (\omega'_0, \ldots, \omega'_k, \Theta)$ that

$$\begin{cases} C_{\max}(\pi',t) \leqslant C_{\max}(\pi'',t) \\ L_{\max}(\pi',t) = L_{\max}(\pi'',t). \end{cases}$$

Consequently, for any schedule $\pi'' \in \Omega(N,t) \setminus \Phi(N,t)$ there exists a schedule $\pi' \in \Phi(N,t)$ such that $C_{\max}(\pi',t) \leqslant C_{\max}(\pi'',t)$ and $L_{\max}(\pi',t) \leqslant L_{\max}(\pi'',t)$.

For the set of schedules $\Phi(N,t) = \{\pi'_1, \ldots, \pi'_m\}$ we have (Fig. 4)

$$\begin{cases} C_{\max}(\pi'_1,t) < C_{\max}(\pi'_2,t) < \cdots < C_{\max}(\pi'_m,t) \\ L_{\max}(\pi'_1,t) > L_{\max}(\pi'_2,t) > \cdots > L_{\max}(\pi'_m,t). \end{cases} \tag{7}$$

Consequently, $\Phi(N,t)$ is the Pareto-optimal set of schedules, and by Lemma 5 we have $|\Phi(N,t)| \leqslant n$. Therefore, we obtain that the schedule π'_1 is optimal for the criterion $C_{\max}$, whereas the schedule π'_m has the best value of the maximal lateness $L_{\max}$, which is what we set out to prove.

5. MEMBERSHIP VERIFICATION FOR AN INSTANCE OF POLYNOMIALLY SOLVABLE AREA

For an arbitrary instance of the problem $1|r_j|L_{\max}$, needed is to know whether it is possible to select α and β such that the inequality system (1) is true. For that, it is necessary and sufficient to solve the following problem.

Problem 3. Given are $3n$ real numbers $r_1, \ldots, r_n, d_1, \ldots, d_n, p_1, \ldots, p_n$. Are there real numbers $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$ such that the inequality system (1) is satisfied?

It is known that $d_1 \leqslant \cdots \leqslant d_n$ for all $i = 1, \ldots, n-1$. We make changes

$$D_i = d_{i+1} - d_i,$$
$$P_i = p_{i+1} - p_i,$$
$$R_i = r_{i+1} - r_i.$$

Then, the system of inequalities (1) is given by

$$\begin{cases} D_1 \geqslant 0 \\ \ldots \\ D_{n-1} \geqslant 0 \\ \alpha P_1 + \beta R_1 \geqslant D_1 \\ \ldots \\ \alpha P_{n-1} + \beta R_{n-1} \geqslant D_{n-1}. \end{cases} \tag{8}$$

Theorem 4. *For the set of parameters $r_1, \ldots, r_n, d_1, \ldots, d_n, p_1, \ldots, p_n$ there are $\alpha_0 \in [0,1]$ and $\beta_0 \in [0,+\infty)$ such that the inequality system (8) is satisfied if and only if there exist $\alpha_1 \in \{0,1\}$ and $\beta_1 \in [0,+\infty)$ for which system (8) is satisfied.*

Proof of Theorem 4. Let us consider possible variants of the inequalities of system (8) (Fig. 5). Let $M = \{1, \ldots, n-1\}$ be the set of indices used in system (8). Represent M as a union of subsets $M^1 \cup \cdots \cup M^7$ depending on the values of P_i and R_i in compliance with the following rules.

1. If $P_i = 0$, $R_i = 0$ are satisfied for $i \in M$, then $i \in M^1$, and the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for any values of α and β under $D_i = 0$ and not satisfied under $D_i > 0$.
2. If $P_i = 0$, $R_i \neq 0$ for are satisfied for $i \in M$, then $i \in M^2$. In this case, the inequality has the form $\beta R_i \geqslant D_i \Leftrightarrow \beta \geqslant \frac{D_i}{R_i}$. Denote $\min_{i \in M^2} \frac{D_i}{R_i}$ by $\frac{D^2}{R^2}$. Then, the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied if and only if $\beta \geqslant \frac{D^2}{R^2}$.
3. If $P_i \neq 0$, $R_i = 0$ are satisfied for $i \in M$, then $i \in M^3$. In this case, the inequality is given by $\alpha P_i \geqslant D_i \Leftrightarrow \alpha \geqslant \frac{D_i}{P_i}$. Denote $\min_{i \in M^3} \frac{D_i}{P_i}$ by $\frac{D^3}{P^3}$. Then, the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for all values of $\alpha \geqslant \frac{D^3}{R^3}$ and only for them.
4. If $P_i < 0$, $R_i < 0$ are satisfied for $i \in M$, then $i \in M^4$. In this case, solution exists if and only if $\alpha = \beta = 0$. Consequently, if $M^4 \neq \emptyset$, then $\alpha_0 = \beta_0 = 0$ are the sole possible coefficient for which (8) has a solution.
5. If $P_i > 0$, $R_i < 0$ are satisfied for $i \in M$, then $i \in M^5$ and

$$P_i + \beta R_i \geqslant \alpha P_i + \beta R_i \geqslant D_i,$$
$$\frac{D_i - P_i}{R_i} \geqslant \beta \geqslant 0.$$

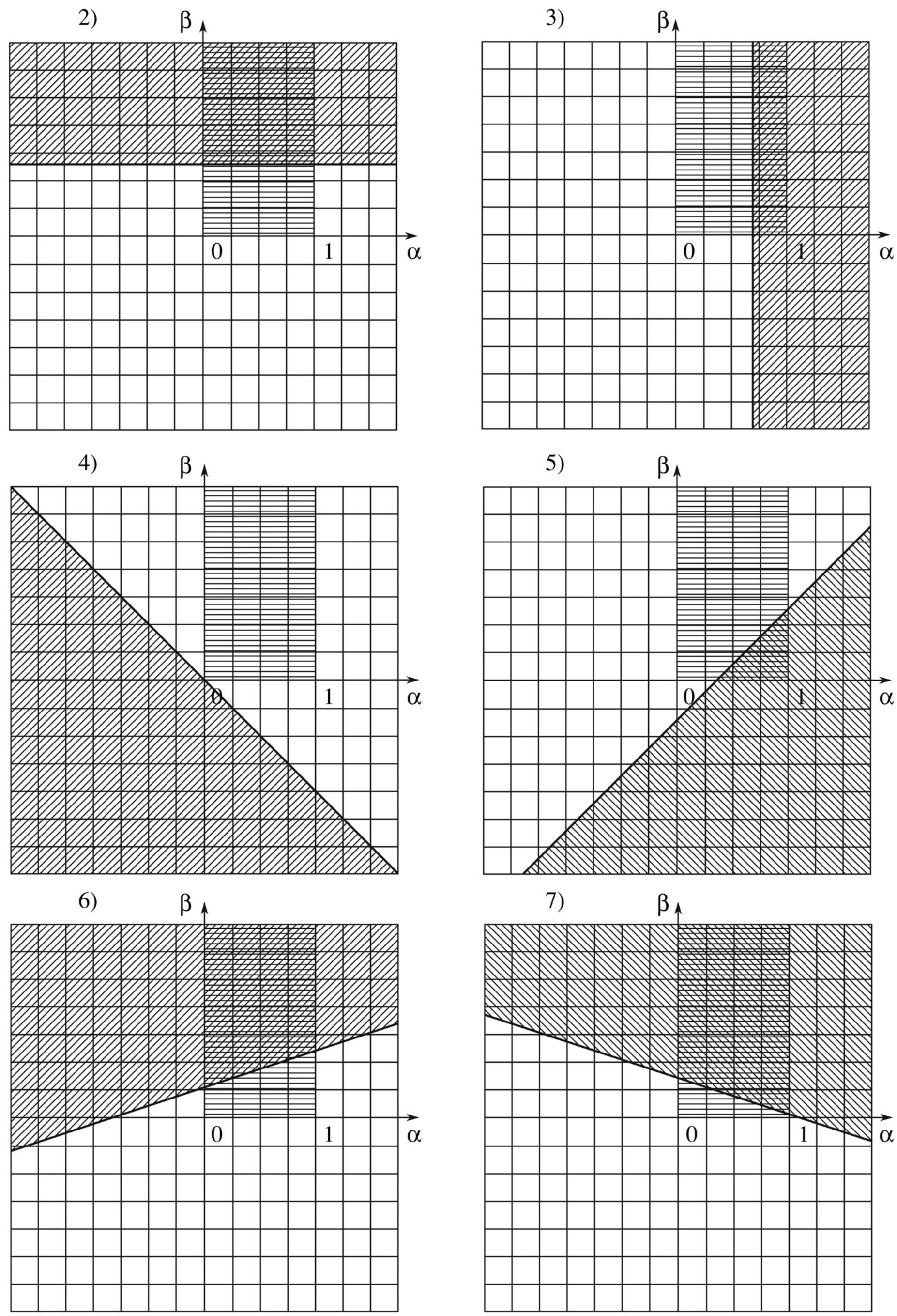

Fig. 5. Types of inequalities of system (8).

We notice that the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for $\alpha = 1$ and $\beta \leqslant \frac{D_i - P_i}{R_i}$. Denote $B = \min_{i \in M^5} \frac{D_i - P_i}{R_i}$ and notice that $B \geqslant 0$ and for any $i \in M^5$ the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for $\alpha = 1$ and $\beta \in [0, B]$.

6. If $P_i < 0$, $R_i > 0$ are satisfied for $i \in M$, then $i \in M^6$.

7. If $P_i > 0$, $R_i > 0$ are satisfied for $i \in M$, then $i \in M^7$.

We note that all possible pairs of P_i and R_i were considered, whence it follows that $M \equiv M^1 \cup \ldots \cup M^7$. We assume that there are $\alpha_0 \in [0, 1]$ and $\beta_0 \in [0, +\infty)$ such that system (8) is true.

If $M^1 \neq \emptyset$, then the inequality $\alpha_0 P_i + \beta_0 R_i \geqslant D_i \Leftrightarrow 0 \geqslant D_i$ is true for any $i \in M^1$. Consequently, this inequality is satisfied for all $\alpha \in [0,1]$ and $\beta \in [0,+\infty)$.

If $M^3 \neq \emptyset$, then for any $i \in M^3$ the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is true if and only if $\alpha \geqslant \frac{D_0^3}{P_0^3}$. Consequently, if the inequality has solution for some $\alpha_0 \in [0,1]$, then for $\alpha_1 = 1 \geqslant \alpha_0 \geqslant \frac{D_0^3}{P_0^3}$ this inequality is also true for all $i \in M^3$.

If $M^4 \neq \emptyset$, then $\alpha_0 = 0$, $\beta_0 = 0$ is a sole possible solution of system (8). Consequently, system (8) is true for $\alpha_1 = \beta_1 = 0$.

Let $M^5 \neq \emptyset$. We demonstrate that the inequality $\alpha P_i + \beta R_i \geqslant D_i$ can be satisfied for all $i \in M^2 \cup M^5 \cup M^6 \cup M^7$ under some $\alpha_0 \in [0,1]$ and $\beta_0 \in [0,+\infty)$ if and only if it is satisfied for $\alpha_1 = 1$ and $\beta_1 = B$. For $i \in M^5$ this assertion is a self-evident truth.

If $M^5 \neq \emptyset$ and $M^2 \neq \emptyset$, then $\beta_0 \geqslant \frac{D^2}{R^2}$ is satisfied. We note that since $M^5 \neq \emptyset$, then $B \geqslant \beta_0$, and consequently, the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for all $i \in M^2 \cup M^5$.

If $M^5 \neq \emptyset$ and $M^6 \neq \emptyset$, then the inequalities $\alpha_0 P_i + \beta_0 R_i - D_i \geqslant 0$ and $\alpha_0 P_j + \beta_0 R_j - D_j \geqslant 0$ are satisfied for any $i \in M^5$ and $j \in M^6$. We take $i = \arg\min_{i \in M^5} \frac{D_i - P_i}{R_i}$ and obtain

$$\frac{D_i - \alpha_0 P_i}{R_i} \geqslant \beta_0 \geqslant \frac{D_j - \alpha_0 P_j}{R_j} \Rightarrow \left(\frac{D_i}{R_i} - \frac{D_j}{R_j}\right) - \alpha_0 \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) \geqslant 0.$$

Since $R_i < 0$ and $R_j > 0$,

$$\frac{D_i}{R_i} - \frac{D_j}{R_j} < 0 \Rightarrow \alpha_0 \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) > 0,$$

consequently,

$$\left(\frac{D_i}{R_i} - \frac{D_j}{R_j}\right) - \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) \geqslant 0,$$

that is,

$$B = \frac{D_i - P_i}{R_i} \geqslant \frac{D_j - P_j}{R_j},$$
$$P_j + BR_j \geqslant D_j.$$

Therefore, we obtain that for all $j \in M^6$ the inequality $\alpha P_j + \beta R_j \geqslant D_j$ is satisfied under $\alpha_1 = 1$ and $\beta_1 = B$.

If $M^5 \neq \emptyset$ and $M^7 \neq \emptyset$, then for any $i \in M^5$ and $j \in M^7$ satisfied are the inequalities $\alpha_0 P_i + \beta_0 R_i - D_i \geqslant 0$ and $\alpha_0 P_j + \beta_0 R_j - D_j \geqslant 0$. Let $i = \arg\min_{i \in M^5} \frac{D_i - P_i}{R_i}$. We establish that

$$\frac{D_i - \alpha_0 P_i}{R_i} \geqslant \beta_0 \geqslant \frac{D_j - \alpha_0 P_j}{R_j} \Rightarrow \left(\frac{D_i}{R_i} - \frac{D_j}{R_j}\right) - \alpha_0 \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) \geqslant 0.$$

Since $0 \leqslant \alpha_0 \leqslant 1$ $\frac{P_i}{R_i} < 0$ and $\frac{P_j}{R_j} > 0$, we obtain

$$\left(\frac{D_i}{R_i} - \frac{D_j}{R_j}\right) - \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) \geqslant \left(\frac{D_i}{R_i} - \frac{D_j}{R_j}\right) - \alpha_0 \left(\frac{P_i}{R_i} - \frac{P_j}{R_j}\right) \geqslant 0.$$

Consequently,

$$B = \frac{D_i - P_i}{R_i} \geqslant \frac{D_j - P_j}{R_j},$$
$$P_j + BR_j \geqslant D_j.$$

Therefore, for all $j \in M^7$ the inequality $\alpha P_j + \beta R_j \geqslant D_j$ is satisfied under the values of $\alpha_1 = 1$ and $\beta_1 = B$.

As follows from what was proved above, if $M^5 \neq \emptyset$, then the inequality $\alpha P_i + \beta R_i \geqslant D_i$ is satisfied for all $i \in M^2 \cup M^5 \cup M^6 \cup M^7$ under $\alpha_1 = 1$ and $\beta_1 = B$. The inverse proposition is evident, it suffices to take $\alpha_0 = 1$ and $\beta_0 = B$.

If $M^5 = \emptyset$, then the aforementioned reasoning is true for $B' = \max_{i \in M^2 \cup M^6 \cup M^7} \frac{D_i - P_i}{R_i}$. Therefore, if for some $\alpha_0 \in [0, 1]$ and $\beta_0 \in [0, +\infty)$ the system of inequalities (8) is true, then:

- if $M^4 \neq \emptyset$, the system (8) is true for $\alpha_1 = 0$ and $\beta_1 = 0$;
- if $M^4 = \emptyset, M^5 \neq \emptyset$, system (8) is true for $\alpha_1 = 1$, $\beta_1 = B$;
- if $M^4 = \emptyset, M^5 = \emptyset, M \neq M^3$, system (8) is true for $\alpha_1 = 1$, $\beta_1 = B'$;
- if $M = M^3$, system (8) is true for $\alpha_1 = 1$, $\beta_1 = 0$.

We present an algorithm for determination of $\alpha_1 \in \{0, 1\}, \beta_1 \in [0, +\infty)$.

Algorithm 4

Data: $P_1, \ldots, P_n, R_1, \ldots, R_n, D_1, \ldots, D_n$
Result: α_1, β_1

if $M^4 \neq \emptyset$ **then**
  $\alpha_1 := 0, \beta_1 := 0$;
else
  if $M^5 \neq \emptyset$ **then**
    $\alpha_1 := 1, \beta_1 := \min\limits_{i \in M^5} \frac{D_i - P_i}{R_i}$;
  else
    if $M \neq M^3$ **then**
      $\alpha_1 := 1, \beta_1 := \max\limits_{i \in M^2 \cup M^6 \cup M^7} \frac{D_i - P_i}{R_i}$;
    else
      $\alpha_1 := 1, \beta_1 := 0$;
    end
  end
end
for $(i = 1, i < n, i{+}{+})$ **do**
  if $\alpha_1 P_i + \beta_1 R_i < D_i$ **then**
    return(*Do not exist* $\alpha \in [0, 1]$ *and* $\beta \in [0, +\infty)$ *for which system* (1) *is true.*);
  end
end
return(α_1, β_1).

To make sure that there are $\alpha_0 \in [0, 1]$ and $\beta_0 \in [0, +\infty)$ such that system (8) is satisfied, it suffices to verify for satisfiability all system inequalities for the determined values of α_1 and β_1. If at least one of the inequalities is not true, then it follows from the above proof that there is no $\alpha_0 \in [0, 1]$ and $\beta_0 \in [0, +\infty)$ for which the system of inequalities (8) and, consequently, system (1) are true.

Therefore, the system of inequalities (1) is true for some $\alpha_0 \in [0, 1]$ and $\beta_0 \in [0, +\infty)$ if and only if the system of inequalities (8) for α_1, β_1 determined using Algorithm 4 is true, which is what we set out to prove.

Lemma 6. *Complexity of Algorithm 4 does not exceed $O(n \log n)$ operations.*

Proof of Lemma 6. Algorithm 4 performs one sorting, two assignments, $O(n)$ clarifications of the types of inequalities, and verification of satisfiability of $O(n)$ inequalities. Sorting in terms of complexity of $O(n \log n)$ operations is the most difficult part.

6. CONCLUSIONS

Extended was the polynomially solvable area of the classical NP-hard in the strong sense problem $1|r_j|L_{\max}$. In compliance with the criteria $L_{\max}$ and $C_{\max}$, presented was the algorithm to construct the set of Pareto-optimal schedules of complexity of $O(n^3 \log n)$ operations. An algorithm was presented to determine membership of an instance to the polynomially solvable area and the parameters α and β having complexity of $O(n \log n)$ operations.

ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research, project no. 13-08-13190.

REFERENCES

1. Graham, R.L., Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G., Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey, *Ann. Discret. Math.*, 1979, vol. 5, pp. 287–326.
2. Lenstra, J.K., Rinnooy Kan, A.H.G., and Brucker, P., Complexity of Machine Scheduling Problems, *Ann. Discret. Math.*, 1977, vol. 1, pp. 343–362.
3. Jackson, J.R., *Scheduling a Production Line to Minimize Maximum Tardiness*, Manag. Sci. Res. Project. Res. Report no. 43, Los Angeles: Univ. of California, 1955.
4. Simons, B.B., A Fast Algorithm for Single Processor Scheduling, in *19 Ann. Sympos. Found. Comp. Sci.*, Ann Arbor, 1978, pp. 246–252.
5. Hoogeven, J.A., Minimizing Maximum Promptness and Maximum Lateness on a Single Machine, *Math. Oper. Res.*, 1996, vol. 21, pp. 100–114.
6. Lazarev, A.A., *Teoriya raspisanii. Otsenka absolyutnoi pogreshnosti i skhema priblizhennogo resheniya zadach teorii raspisanii* (Scheduling Theory. Estimation of the Absolute Error and a Scheme of Approximate Solution of the Problems of Scheduling Theory), Moscow: MFTI, 2008.
7. Lazarev, A.A., Arkhipov, D.I., and Karpov, I.V., Polynomially Solvable Case of the NP-hard Problem $1|r_j|L_{\max}$, in *Int. Conf. Project Management Sched.*, Tours, 2010, pp. 289–293.

This paper was recommended for publication by F.T. Aleskerov, a member of the Editorial Board