

РАСПИСАНИЯ ВЫПОЛНЕНИЯ ЗАДАНИЙ ВО ВЗАИМОСВЯЗАННЫХ ПОСЛЕДОВАТЕЛЬНО РАБОТАЮЩИХ ПРОИЗВОДСТВЕННЫХ СИСТЕМАХ

Ю.А. Зак

Аннотация. Классическая в теории расписаний задача построения последовательности выполнения заданий на одной машине в условиях наличия ограничений на времена начала и завершения выполнения заданий и учитывающая не только затраты времени на работу оборудования, но и потери на постобработку, рассмотрена для многостадийных производственных систем, состоящих из взаимосвязанной цепочки участков и цехов промышленного предприятия. Критерием оптимальности задачи является выполнение многостадийного расписания в кратчайшие сроки. Рассматриваемые в работе задачи относятся к классу NP-полных задач экспоненциальной сложности. Исследованы свойства допустимых и оптимальных последовательностей выполнения заданий. Предложены методы расчета нижней границы длины оптимального расписания и правила отсева недопустимых и неоптимальных продолжений. Разработаны алгоритмы точного и приближенного решения задачи модифицированными методами ветвей и границ. Предложенные алгоритмы проиллюстрированы числовыми примерами. Выполненные автором вычислительные эксперименты показали, что наличие системы строгих ограничений на сроки выполнения заданий при реализации предложенных в работе алгоритмов в ряде случаев существенно сокращает число рассматриваемых вариантов.

Ключевые слова: последовательность выполнения заданий, многостадийное расписание, минимальное время, эвристический алгоритм, нижняя граница критерия оптимальности, отсев неперспективных продолжений, метод ветвей и границ.

ВВЕДЕНИЕ

Определение допустимых и оптимальных последовательностей изготовления некоторого множества различных изделий, каждое из которых должно пройти последовательную обработку в K структурных подразделениях (например, участках или цехах) предприятия актуально для построения систем оперативно-календарного производства (см., например, работы [1–6]). Календарное планирование всей последовательной цепочки производств позволит повысить эффективность работы взаимосвязанной системы участков и цехов промышленного предприятия, определит конкретизацию во времени изготовления всех изделий, обеспечит выпуск различных видов продукции в установленные договорными сроки и с наилучшими технико-экономическими показателями.

Каждое структурное подразделение производства в такой постановке задачи рассматривается

как отдельная стадия. Такими стадиями обработки могут быть группы станков, многопроцессорные и многопоточные или автоматические линии, а также различные структурные единицы предприятия более низкого уровня иерархии. Каждая стадия обработки рассматривается как одна машина. В дальнейшем в качестве времени обработки изделий на машине каждой стадии — это суммарное время выполнения всех операций этого задания на этом участке, т. е. на этой стадии обработки (суммарное время обработки на всех станках или автоматических линиях этой стадии). После изготовления на технологическом оборудовании на каждой стадии производится постобработка каждого изделия, связанная с контролем, испытанием, необходимым временем прослеживания (например, с охлаждением или нагреванием), оформлением необходимой документации, транспортными потерями времени и др., что также требует затрат времени.

Технологическая схема производства, рассматриваемая в данной задаче, представлена на рис. 1.

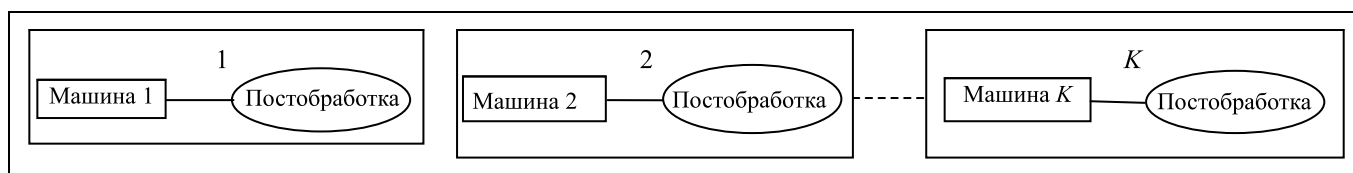


Рис. 1. Технологическая схема производства

Введем определение технологической организации производства «многостадийная обработка изделий на одной машине».

В монографиях и в периодической литературе по теории расписаний рассматривались в различных постановках и с различными критериями оптимальности математические модели, точные и приближенные методы решения классических задач построения последовательностей выполнения заданий на одной машине применительно только к одному структурному подразделению предприятия. Определенный интерес представляет учет времен, необходимых на постобработку после завершения изготовления изделий на машине. В качестве критериев оптимальности выбирается выполнение всего комплекса работ в кратчайшие сроки (см., например, работы [1, 2, 5, 6] и др.). При этом независимые друг от друга расписания для каждого отдельного участка могут оказаться совершенно не эффективными для общей многостадийной системы, включающей в себя последовательную цепочку из нескольких участков.

Задачам построения многостадийных расписаний выполнение заданий на одной машине, имеющим большое практическое значение и ярко выраженную специфику, как и задачам построения многостадийных расписаний в условиях ограничений на времена выполнения заданий, не уделялось достаточного внимания в литературе. Практически важные постановки и пути решения задач построения двухстадийных расписаний рассматривались в работах М.Х. Прилуцкого и Д.В. Седова [4], а в приложениях, когда второй стадией служит процесс сборки изделий, в работах Е.Н. Хоботова [3]. В работах автора [4, 6, 7] рассматривались постановки, математические модели, точные и приближенные методы решения некоторых классических задач в многостадийных производственных системах.

1. ПОСТАНОВКА ЗАДАЧИ. СОСТОЯНИЕ ПРОБЛЕМЫ

Технологический процесс предусматривает изготовление некоторого множества изделий на нескольких стадиях обработки. Каждая стадия рассматривается как одна единица оборудования, т. е. как одна машина. После изготовления на техно-

логическом оборудовании осуществляется постобработка каждого изделия, что также требует затрат времени. На всех стадиях обработки последовательность изготовления одинакова для всех изделий. Результат решения задачи — определение номеров временных интервалов, в которых начинается и заканчивается выполнение определенных заданий и постобработки на различных стадиях обработки. Номера этих интервалов заданы целыми числами. Поэтому времена выполнения заданий t_i^k , r_i^k , а также граничные значения d_i^k выражены в номерах и числе временных интервалов. В качестве временной единицы, в зависимости от конкретных приложений, может рассматриваться час, минута, секунда и даже день или сутки.

Изготовление изделий на оборудовании каждой из этих стадий ведется без прерываний времени выполнения. После изготовления изделия на k -й стадии и постобработки изделие поступает на следующую стадию обработки. На машине каждой стадии обработки может начаться изготовление следующего стоящего в последовательности обработки изделия непосредственно после завершения изготовления предыдущего. Начало обработки каждого изделия на следующем стоящем в последовательной цепочке участке (следующей стадии) может начаться только после постобработки его на предыдущей стадии. Заданы допустимые времена начала обработки каждого изделия на 1-й стадии обработки и граничные времена завершения выполнения этого задания на последней стадии обработки (после завершения постобработки). Необходимо найти допустимую последовательность изготовления изделий, обеспечивающую выполнение всех ограничений на допустимые времена изготовления, которая обеспечит минимальное время выполнения расписания на всех стадиях обработки.

Эффективные алгоритмы точного решения такой задачи в одностадийных системах, учитывающие потери времени на постобработку с помощью Schrage-algorithms и его модификации впервые были предложены в работе [10] и развиты в условиях различного вида ограничений в работах автора [2, 4, 11].

Задачи теории расписаний для многостадийных производственных систем, как правило, отно-



сятся к классу NP-полных задач экспоненциальной сложности. Они сложнее задач построения одностадийных расписаний и связаны с существенно большим числом переменных и ограничений. Построение математических моделей таких задач, исследование свойств их допустимых и оптимальных решений, а также алгоритмов получения эффективных и приближенных решений весьма актуально для построения систем календарного планирования производства, планирования ремонтных работ и обслуживания объектов.

Математическая формулировка задачи. Здесь и в дальнейшем под выполнением задания понимается завершение изготовления изделия на всех стадиях обработки. Введем обозначения: $i = 1, \dots, n$ — индексы обрабатываемых изделий; $k = 1, \dots, K$ — индексы различных стадий обработки; d_i^k — допустимый наиболее ранний срок начала выполнения i -го задания на k -й стадии обработки; d_i и D_i — соответственно граничное время начала и завершения выполнения i -го задания; t_i^k — время выполнения i -го задания на k -й стадии обработки; r_i^k — необходимое время постобработки после выполнения i -го задания на k -й стадии обработки; x_i^k , θ_i^k — соответственно время начала и завершения выполнения i -го задания на k -й стадии обработки; z_i^k , r_i^k и Z_i^k — соответственно время начала, длительности и завершения постобработки i -го задания на k -й стадии технологического процесса; $T_i = Z_i^K$ — время завершения всех работ многостадийного изготовления и постобработки i -го изделия; $F = \max_{i \in \bar{I}} T_i$ — время завершения всех работ многостадийного расписания, т. е. значение критерия оптимальности задачи.

Пусть построена некоторая последовательность выполнения заданий $\tilde{U} = \{u_1, u_2, \dots, u_{j-1}, u_j, \dots, u_n\}$.

Времена начала и завершения выполнения каждого из заданий и постобработки изделий на всех стадиях обработки определяются по формулам

$$\begin{aligned} x_{u_1}^1 &= d_{u_1}^1 + 1, & \theta_{u_1}^1 &= x_{u_1}^1 + t_{u_1}^1 - 1, \\ z_{u_1}^1 &= \theta_{u_1}^1 + 1, & Z_{u_1}^1 &= z_{u_1}^1 + r_{u_1}^1 - 1; \\ x_{u_1}^k &= \max(d_{u_1}^k, Z_{u_1}^{k-1}) + 1, \\ \theta_{u_1}^k &= x_{u_1}^k + t_{u_1}^k - 1, & z_{u_1}^k &= \theta_{u_1}^k + 1, \\ Z_{u_1}^k &= z_{u_1}^k + r_{u_1}^k - 1, & k &= 1, \dots, K; \end{aligned} \quad (1)$$

$$\begin{aligned} x_{u_j}^1 &= \max(d_{u_j}^1, Z_{u_{j-1}}^1) + 1, & \theta_{u_{j-1}}^1 &= x_{u_j}^1 + t_{u_j}^1 - 1, \\ z_{u_j}^1 &= \theta_{u_j}^1 + 1, & Z_{u_1}^1 &= z_{u_1}^1 + r_{u_1}^1 - 1; \\ x_{u_j}^k &= \max(d_{u_j}^k, Z_{u_{j-1}}^{k-1}) + 1, & \theta_{u_j}^k &= x_{u_j}^k + t_{u_j}^k - 1, \\ z_{u_j}^k &= \theta_{u_j}^k + 1, \end{aligned} \quad (3)$$

$$\begin{aligned} Z_{u_j}^k &= z_{u_j}^k + r_{u_j}^k - 1, \\ k &= 2, \dots, K, & j &= 1, 2, \dots, n; \end{aligned} \quad (4)$$

$$T_{u_j} = Z_{u_j}^K, \quad j = 1, \dots, n. \quad (5)$$

2. СВОЙСТВА ДОПУСТИМЫХ И ОПТИМАЛЬНЫХ РЕШЕНИЙ

Рассмотрим две подпоследовательности $\tilde{U}(s)$ и $\tilde{U}(p)$.

Правило отсева 1. Если для двух подпоследовательностей $\tilde{U}(s)$ и $\tilde{U}(p)$ справедливы условия

$$\begin{aligned} Z_{u_{m(s)}}^k &\leq Z_{u_{m(p)}}^k, & k &= 1, 2, \dots, (K-1), \\ \theta_{u_{m(s)}}^K &\leq \theta_{u_{m(p)}}^K, \end{aligned} \quad (6)$$

где $\tilde{I}^1(p)$ и $\tilde{I}^1(s)$ — подмножества заданий в подпоследовательностях $\tilde{U}(s)$ и $\tilde{U}(p)$, $u_{m(s)}$ и $u_{m(p)}$ — соответственно индексы заданий, стоящих на последнем месте в подпоследовательностях $\tilde{U}(s)$ и $\tilde{U}(p)$, $Z_{u_{m(s)}}^k$ и $Z_{u_{m(p)}}^k$ — соответственно времена завершения постобработки последнего стоящего в подпоследовательностях $\tilde{U}(s)$ и $\tilde{U}(p)$ задания на каждой стадии $k = 1, 2, \dots, (K-1)$, $\theta_{u_{m(s)}}^K$ и $\theta_{u_{m(p)}}^K$ — времена завершения обработки этих изделий на машине последней K -й стадии, то подпоследовательность $\tilde{U}(p)$ не содержит оптимальных решений и может быть отброшена как неперспективная.

Упорядочим все задания по неубыванию граничных времен их завершения, т. е. значений D_i —

$$\tilde{Q} = \{q_1, q_2, \dots, q_{l-1}, q_l, \dots, q_n \mid D_{q_{l-1}} \leq D_{q_l}, \\ l = 2, 3, \dots, n\}. \quad (7)$$

Правило отсева 2. Если для некоторой подпоследовательности $\tilde{U}(s)$ с подмножеством заданий $\tilde{I}^1(s)$, содержащей все $(l-1)$ задания, стоящие в начале последовательности \tilde{Q} , и на последнем

месте которой стоит задание $u_{m(s)}$, справедливы условия

$$\max_{1 \leq k \leq K-1} \left\{ Z_{u_{m(s)}}^k + \sum_{g=k+1}^K (t_{q_l}^g + r_{q_l}^g) \right\} > D_{q_l},$$

$$q_l \in \{\tilde{Q}/\tilde{I}^1(s)\}, \quad (8)$$

то эта подпоследовательность $\tilde{U}(s)$ не содержит допустимых планов и может быть отброшена как неперспективная.

Доказательство этого утверждения следует из условий

$$Z_{q_l}^k + \sum_{g=k+1}^K (t_{q_{l+1}}^g + r_{q_{l+1}}^g) \leq \max(Z_{q_l}^k + \theta_{q_l}^{k+1}) +$$

$$+ \sum_{g=k+1}^K [\max(Z_{q_{l+1}}^{g-1}, \theta_{q_{l+1}}^g)] + (t_{q_{l+1}}^g + r_{q_{l+1}}^g).$$

Правило отсева 3 может рассматриваться как усиление правила 2. Если выполняется хотя бы одно из системы неравенств

$$\max_{1 \leq k \leq K-1} \left\{ Z_{u_{m(s)}}^k + \sum_{q_l=(m_s+1)}^n \sum_{g=k+1}^K (t_{q_l}^g + r_{q_l}^g) \right\} > D_{q_l},$$

$$q_l \in \{\tilde{Q}/\tilde{I}^1(s)\}, \quad (9)$$

то данная подпоследовательность выполнения заданий не имеет допустимых продолжений.

Правило отсева 4. Более слабые условия проверки допустимости строящейся последовательности, но требующие меньшего объема вычислений, связаны с проверкой двух или хотя бы одной из системы неравенств (10)–(12), которые имеют вид

$$Z_{u_{m(s)}}^{K-1} + \sum_{q_l=(m_s+1)}^{m_n} (t_{q_l}^K + r_{q_l}^K) > D_{q_l},$$

$$q_l \in \{\tilde{Q}/\tilde{I}^1(s)\}, \quad (10)$$

$$\theta_{u_{m(s)}}^1 + \sum_{q_l=(m_s+1)}^{m_n} \sum_{g=1}^K (t_{q_l}^g + r_{q_l}^g) > D_{q_l},$$

$$q_l \in \{\tilde{Q}/\tilde{I}^1(s)\}, \quad (11)$$

$$\theta_{u_{m(s)}}^1 + \sum_{g=1}^K (t_{q_l}^g + r_{q_l}^g) > D_{q_l},$$

$$\theta_{u_{m(s)}}^{K-1} + (t_{q_l}^K + r_{q_l}^K) > D_{q_l}. \quad (12)$$

3. НИЖНЯЯ ГРАНИЦА ДЛИНЫ ОПТИМАЛЬНОГО РАСПИСАНИЯ

Выражения для вычисления нижней границы как в начале работы алгоритма, так и в процессе построения последовательности выполнения заданий, будут необходимы для реализации последовательных методов решения задачи (метода ветвей и границ и метода динамического программирования). Кроме того, информация о значении нижней границы длины оптимального расписания позволит оценить точность полученного решения различными алгоритмами приближенного решения задачи.

Время завершения выполнения и постобработки всех заданий многостадийной обработки изделий не может быть меньше каждой из величин

$$f^k = \min_{1 \leq i \leq n} d_i^k \sum_{i=1}^n t_i^k + \min_{1 \leq i \leq n} r_i^k, \quad k = 1, \dots, K.$$

После выполнения всех заданий с соответствующей постобработкой на k -й стадии технологического процесса необходимы еще временные затраты на всех остальных стадиях обработки, которые не могут быть меньше значения

$$\mu^l = \min_{1 \leq i \leq n} (d_i^l + t_i^l + r_i^l), \quad l = 1, \dots, (k-1),$$

$$(k+1), \dots, K, \quad l \neq k.$$

Следовательно, нижняя граница длины расписания $\vartheta(F)$ должна быть не меньше значения

$$\vartheta^k(F) = f_k + \sum_{l=1, l \neq k}^K \mu_l, \quad k = 1, \dots, K.$$

С учетом изложенного выше значение нижней границы длины многостадийного расписания вычисляется по формуле

$$\vartheta(F) = \max_{1 \leq k \leq K} \left(f^k + \sum_{l=1, l \neq k}^K \mu^l \right). \quad (13)$$

Пусть на некотором s -м шаге алгоритма построена подпоследовательность выполнения подмножества заданий $\tilde{I}^1(s) - \tilde{U}(s) = \{u_1(s), u_2(s), \dots, u_j(s), \dots, u_m(s)\}$, нижняя граница длины расписания $m \leq n$, и вычислены значения $\theta_{u_{m(s)}}^k$, $k = 1, \dots, K$. $\tilde{I}^2(s) = \{\tilde{I}/\tilde{I}^1(s)\}$ — подмножество подлежащих выполнению заданий. Тогда

$$f^k(s) = \min_{i \in \tilde{I}^2(s)} \max(d_i^k, \theta_{u_{m(s)}}^k) + \min_{\tilde{I}^2(s)} r_i^k,$$

$$k = 1, \dots, K; \quad (14)$$



$$\mu^l(s) = \min_{i \in \tilde{I}^2(s)} [\max(d_i^l, \theta_{u_{m(s)}}^l) + t_i^l + r_i^l], \quad (k+1), \dots, K, l \neq k. \quad (15)$$

Следовательно, нижняя граница длины расписания для подпоследовательности выполнения подмножества заданий вычисляется по формуле

$$\mathfrak{G}[\tilde{U}(s)] = \max_{1 \leq k \leq K} \left(f^k(s) + \sum_{l=1, l \neq k}^K \mu_l(s) \right). \quad (16)$$

В качестве еще одного выражения для вычисления нижней границы может рассматриваться формула

$$\mathfrak{G}\{F[\tilde{U}(s)]\} = \theta_{u_{m(s)}}^K + \sum_{q_l = (m_s + 1), q_l \in (\tilde{Q}/\tilde{I}^1(s))}^{m_n} t_{q_l}^K + \min_{q_l \in \{\tilde{Q}/\tilde{I}^1(s)\}} r_{q_l}^K. \quad (17)$$

4. АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ

Рассмотрим алгоритм решения задачи модифицированным методом ветвей и границ.

В процессе построения расписания строим различные варианты подпоследовательностей выполнения заданий $\tilde{U}(s), s = 1, \dots, S$. При этом, пользуясь правилами 2 или 4 (см. § 2, а также правилом 1 и формулами (6), (8)–(12)), подпоследовательности, не содержащие допустимых и оптимальных решений, исключаем из рассмотрения. Процесс решения завершается, когда построена некоторая последовательность $\tilde{U}(h)$, в которой сроки завершения выполнения всех заданий меньше или равны установленным граничным значениям, а время выполнения расписания всех работ не превышает нижней границы длины расписания всех подлежащих дальнейшему развитию подпоследовательностей.

Времена начала и завершения выполнения заданий рассчитываются по формулам (1)–(5).

Пусть на некотором шаге решения задачи построено $A(s)$ допустимых последовательностей выполнения заданий, критерии оптимальности которых соответственно равны $F_\rho[\tilde{U}(s)], \rho = 1, \dots, A(s)$,

$\bar{F}[\tilde{U}(s)] = \min_{1 \leq \rho \leq A(s)} F_\rho[\tilde{U}(s)]$ — наилучшее из построенных расписаний. В начале работы алгоритма полагаем $\bar{F}[\tilde{U}(s)] \rightarrow \infty$.

Обозначим признаком $\beta(s) = 0$ перспективные подпоследовательности $\tilde{U}(s)$, подлежащие дальнейшему развитию, $\beta(s) = 1$ и $\beta(s) = 2$ соответственно $\tilde{U}(s)$, не содержащие оптимальных и допустимых решений; $\beta(s) = 3$ — допустимое решение задачи.

Упорядочим все подлежащие выполнению задания в последовательность \tilde{Q} в соответствии с выражением (7). Алгоритм предусматривает выполнение следующих шагов.

Шаг 1. Полагаем $\tilde{U}(s) = \{q_s\}, s = 1, 2, \dots, n$. Рассчитываем значения $x_{q_s}^k, \theta_{q_s}^k, Z_{q_s}^k, k = 1, \dots, K$, по формулам (1), (2). Полагаем $\tilde{I}^1(s) = \{q_s\}, \tilde{I}^2(s) = \left\{ \frac{\tilde{I}}{q_s} \right\}$. Если выполняется неравенство

$$Z_{q_s}^K > D_{q_s},$$

хотя бы для одного значения q_s , то не существует допустимых решений задачи, и алгоритм завершает работу. В противном случае полагаем $\beta(s) = 0$, вычисляем значение нижней границы $\mathfrak{G}\{F[\tilde{U}(s)]\}$ (например, по формуле (17), положив $u_{m(s)} = q_s$). Обозначим $B(s)$ — число построенных подпоследовательностей. Переходим к шагу 2.

Шаг 2. Обозначим $\tilde{Q}(s) = \{\tilde{Q}/\tilde{I}^1(s)\}$, где $\tilde{Q}(s) = \{\bar{q}_1(s), \dots, \bar{q}_l(s), \dots, \bar{q}_p(s)\}$ — последовательность, в которой отсутствуют все индексы заданий множества $\tilde{I}^1(s)$. Среди последовательностей $\tilde{U}(s)$, у которых $\beta(s) = 0$, выбираем $\tilde{U}(s)$ с наименьшим значением нижней границы $\mathfrak{G}\{F[\tilde{U}(s)]\}$. Если таких последовательностей не существует, то допустимое расписание с наименьшим значением критерия оптимальности является решением задачи. Если построенных допустимых расписаний не существует, т. е. $\bar{F}[\tilde{U}(s)] \rightarrow \infty$, то не существует допустимых решений задачи, и алгоритм завершает работу. Если $\tilde{I}^2(s) \neq \emptyset$, то устанавливаем на последнее место в строящейся $\tilde{U}(s)$ каждое из заданий подмножества $\tilde{I}^2(s) = \{j_1(s), j_2(s), \dots, j_\mu(s)\}, \tilde{I}^1[B(s) + h] = \{\tilde{I}^1(s) \cup j_h(s)\}, \tilde{I}^2[B(s) + h] = \{\tilde{I}^2(s)/j_h(s)\},$

$$\tilde{U}[B(s) + h] = \{u_1(s), u_2(s), \dots, u(s), \dots, u_m(s), j_h(s)\}, \quad h = 1, 2, \dots, \mu.$$

Полагаем $\beta(s) = 1$. Вычисляем по формулам (2)–(5) значения $x_{j_h(s)}^k, \theta_{j_h(s)}^k, Z_{j_h(s)}^k, k = 1, \dots, K$. Полагая для $\tilde{U}[B(s) + h]$ значение $u_m(s) = j_h(s)$, проверяем выполнение неравенств (9) или (11) и (12), положив $q_l \in \{\tilde{Q}/\tilde{I}^1[B(s) + h]\}$. Если для какого-то индекса $j_h(s)$ выполняется хотя бы одно из этой системы неравенств, то полагаем $\beta[B(s) + h] = 2$.

Для подмножеств, для которых ни одно из этой системы неравенств не выполняется, полагаем $\beta[B(s) + h(s)] = 0$, вычисляем значение нижней границы по формулам (16) или (17). Увеличиваем число построенных подпоследовательностей на число μ , и переходим к шагу 3. Если $\tilde{I}^1[B(s) + h(s)] = \tilde{I}$, то переходим к шагу 5.

Шаг 3. Если для подпоследовательности $\tilde{U}[B(s) + h]$ выполняется хотя бы одно из неравенств

$$Z_{j_h(s)}^{K-1} + (t_{\gamma(s)}^K + r_{\gamma(s)}^K) > D_{\gamma(s)},$$

$$\theta_{u_{j_h(s)}}^1 + \sum_{g=1}^K (t_{\gamma(s)}^g + r_{\gamma(s)}^g) > D_{\gamma(s)},$$

то $\tilde{U}[B(s) + h]$ не содержит допустимых решений, полагаем $\beta[B(s) + h(s)] = 3$ и переходим к шагу 2. В противном случае переходим к шагу 4.

Шаг 4. Для всех вновь построенных перспективных подмножеств $\tilde{I}^1[B(s) + h(s)]$, для которых $\beta[B(s) + h(s)] = 0$, проверяем правила 1–3 перспективности продолжения со всеми ранее построенными подпоследовательностями $\tilde{U}(s)$, $s = 1, \dots, B(s)$, у которых $\beta(s) = 0$. Если будет установлено, что какое-либо из них не содержит оптимальных решений, то для этого подмножества полагаем $\beta[B(s) + h(s)] = 2$ или $\beta(s) = 2$. Переходим к шагу 2.

Шаг 5. Для построенной допустимой последовательности выполнения заданий значение критерия оптимальности равно $T_{j_h(s)} = F[B(s) + h(s)] = Z_{j_h(s)}^K$. Если выполняется неравенство $\bar{F}[\tilde{U}(s)] > F[B(s) + h(s)]$, то полагаем

$$\bar{F}[\tilde{U}(s)] = F[B(s) + h(s)].$$

Если

$$\bar{F}[\tilde{U}(s)] \leq (\min_{1 \leq s \leq [B(s) + h(s)]} \{F[\tilde{U}(s)]\}) \beta(s) = 0,$$

то получено решение задачи, и алгоритм завершает работу. В противном случае переходим к выполнению шага 2.

При полной реализации алгоритма ветвей и границ этот алгоритм является алгоритмом экспоненциальной сложности. В случае реализации односторонней схемы ветвления без последующего возвращения к вершинам дерева с меньшим значением нижней границы и в условиях достаточно большой области допустимых решений алгоритм решения в ряде случаев может иметь полиномиальную сложность.

Полиномиальным алгоритмом можно гарантировать получение только приближенного решения задачи без учета ограничений. Здесь методом вет-

вей и границ строится только одна ветвь дерева решений задачи, не обращая внимания, что вершины других ветвей содержат граничные значения критерия оптимальности лучше, чем в строящемся решении. На каждом шаге, требующем определенного объема вычислений, выбирается место в последовательности одного из заданий. Так как таких заданий N , то этот алгоритм является алгоритмом полиномиальной сложности.

5. ИЛЛЮСТРАТИВНЫЙ ПРИМЕР

Времена выполнения семи заданий на машинах трех стадий изготовления изделий, а также потери времени на постобработку на каждой из этих стадий и допустимые времена начала и завершения выполнения заданий сведены в табл. 1. Последовательность по невозрастающей временам завершения выполнения заданий (7) имеет вид $\tilde{Q} = \{7, 3, 4, 1, 6, 2, 5\}$.

Вычислим значение нижней границы длины расписания по формуле (13):

$$\theta_1^3 = 10 + 12 + 2 + 12 + 4 + 9 = 49,$$

$$\theta_2^3 = 5 + 15 + 3 + 3 + 4 + 10 = 40,$$

$$\theta_3^3 = 8 + 10 + 5 + 3 + 7 + 12 = 45,$$

$$\theta_4^3 = 0 + 8 + 7 + 8 + 5 + 13 = 41,$$

$$\theta_5^3 = 12 + 9 + 4 + 10 + 2 + 10 = 47,$$

$$\theta_6^3 = 5 + 6 + 6 + 9 + 4 + 11 = 41,$$

$$\theta_7^3 = 1 + 15 + 4 + 6 + 5 + 8 = 39.$$

$$T^3 = 9 + 10 + 12 + 13 + 10 + 11 + 8 = 73;$$

$$\vartheta(F) = \min(\theta_1^3 + (T^3 - 9 + 2); \theta_2^3 + (T^3 - 10 + 2);$$

$$\theta_3^3 + (T^3 - 12 + 2); \theta_4^3 + (T^3 - 13 + 3);$$

$$\theta_5^3 + (T^3 - 10 + 2); \theta_6^3 + (T^3 - 11 + 2);$$

$$\theta_7^3 + (T^3 - 8 + 2)) = 104.$$

Таблица 1

Временные параметры задачи

Но- мер зада- ния	d_i	Временные параметры на различных стадиях обработки $k = 1, 2, 3$						D_i
		1-я стадия		2-я стадия		3-я стадия		
		t_i^1	r_i^1	t_i^2	r_i^2	t_i^3	r_i^3	
1	10	12	2	12	4	9	3	80
2	5	15	3	3	4	10	5	102
3	8	10	5	7	3	12	4	66
4	0	8	7	8	5	13	2	72
5	12	9	4	10	2	10	6	115
6	5	6	6	9	4	11	5	93
7	1	15	4	6	5	8	4	55



Далее приведен процесс решения задачи. Для всех построенных подпоследовательностей выполнения заданий нижняя граница оптимального решения вычисляется по формулам (14), (15) или (16), (17).

$s = 1$.

Задание 1: $\theta_1^3 = 10 + 12 + 2 + 12 + 4 + 9 = 49$, $\vartheta(T_7^3) = 49 + 8 + 4 = 61 > 55$, т. е. недопустимо.

Задание 2: $\theta_2^3 = 5 + 15 + 3 + 3 + 4 + 10 = 40$, $\vartheta(\theta_7^3) = 40 + 8 + 4 = 52 < 55$; $\vartheta(\theta_3^3) = 48$, $\vartheta(T_3^3) = 48 + 12 + 4 = 64 < 68$; $\vartheta(\theta_3^3) = 6$, $\vartheta(T_4^3) = 60 + 13 + 2 = 75 > 72$, т. е. недопустимо.

Задание 3: $\theta_3^3 = 8 + 10 + 5 + 3 + 7 + 12 = 45$, $\vartheta(T_7^3) = 45 + 8 + 4 = 57 > 55$, т. е. недопустимо.

Задание 5: $\theta_5^3 = 12 + 9 + 4 + 10 + 2 + 10 = 47$, $\vartheta(T_7^3) = 47 + 8 + 4 = 59 > 55$, т. е. недопустимо.

Задание 6: $\theta_6^3 = 3 + 6 + 6 + 9 + 4 + 11 = 39$, $\vartheta(T_7^3) = 39 + 8 + 4 = 51 < 55$, $\vartheta(\theta_7^3) = 47$, $\vartheta(T_3^3) = 47 + 12 + 4 = 63 < 68$; $\vartheta(\theta_3^3) = 59$, $\vartheta(T_4^3) = 59 + 13 + 2 = 74 > 72$, т. е. недопустимо.

Аналогичные вычисления для заданий 4 и 7 показали допустимость установки их на 1-е место в строящейся последовательности. Вычислим значения нижних границ для каждого из этих вариантов: $\vartheta(F|u_1 = 4) = \theta_4^3 + 9 + 10 + 12 + 10 + 11 + 8 + 3 = 41 + 63 = 104$, $\vartheta(F|u_1 = 7) = \theta_7^3 + 9 + 10 + 12 + 13 + 10 + 11 + 2 = 38 + 67 = 105$.

Выбирая вершину с наименьшим значением нижней границы и выполняя аналогичные вычисления на каждом шаге, строим дерево решений задачи.

$s = 2$. $\tilde{U} = \{4, 7\}$. Так как $\theta_4^3 = 41$, $\vartheta(\theta_3^3|u_1 = 4) = 41 + 12 = 53$, $\vartheta(T_7^3|u_1 = 4) = 53 + 8 + 4 = 65 > 55$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{4,7\}) = 105$.

$s = 3$. $\tilde{U} = \{4, 7, 3\}$. Так как $\theta_7^3 = 49$, $\vartheta(\theta_6^3|\tilde{U} = \{4, 7, 3\}) = 49 + 11 = 60$, $\vartheta(T_1^3(\tilde{U} = \{4, 7, 1\})) = 60 + 9 + 3 = 72 < 80$, $\vartheta(T_1^3(\tilde{U} = \{4, 7, 6, 1, 3\})) = 60 + 9 + 12 + 4 = 85 > 68$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{4, 7, 3\}) = 105$.

$s = 4$. $\tilde{U} = \{4, 7, 3, 1\}$. Так как $\theta_3^3 = 61$, $\vartheta(\theta_6^3|\tilde{U} = \{4, 7, 3, 6\}) = 61 + 11 = 72$, $\vartheta(T_6^3(\tilde{U} = \{4, 7, 3, 2\})) = 72 + 5 = 77 < 90$, $\vartheta(T_1^3(\tilde{U} = \{4, 7, 2, 1\})) = 72 + 9 + 3 = 84 < 80$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{4, 7, 3, 1\}) = 108$.

$s = 5$. $\tilde{U} = \{7, 3\}$. Так как $\theta_7^3 = 38$, $\vartheta(\theta_2^3|\tilde{U} = \{7, 4\}) = 38 + 13 = 51$, $\vartheta(T_4^3(\tilde{U} = \{7, 4, 3\})) = 51 + 12 + 4 = 67 > 66$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{7, 3\}) = 107$.

$s = 6$. $\tilde{U} = \{7, 3, 4\}$. Так как $\theta_3^3 = 52$, $\vartheta(\theta_1^3|\tilde{U} = \{7, 3, 1\}) = 52 + 9 = 61$, $\vartheta(T_4^3(\tilde{U} = \{7, 3, 1, 4\})) = 61 + 13 + 2 = 76 > 72$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{7, 3, 4\}) = 109$.

$s = 7$. $\tilde{U} = \{4, 7, 3, 1, 6\}$. Так как $\theta_1^3 = 72$, $\vartheta(\theta_2^3|\tilde{U} = \{4, 7, 3, 1, 2\}) = 72 + 10 = 82$, $\vartheta(T_6^3(\tilde{U} = \{4, 7, 3, 1, 2, 6\})) = 82 + 11 + 5 = 96 > 93$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{4, 7, 3, 1, 6\}) = 109$.

$s = 8$. $\tilde{U} = \{4, 7, 3, 1, 6, 2\}$. Так как $\theta_6^3 = 83$, $\vartheta(\theta_5^3|\tilde{U} = \{4, 7, 3, 1, 6, 5\}) = 83 + 10 = 93$, $\vartheta(T_2^3(\tilde{U} = \{4, 7, 3, 1, 6, 5, 2\})) = 93 + 10 + 5 = 108 > 102$, то остальные варианты недопустимы; $\vartheta(F|\tilde{U} = \{4, 7, 3, 1, 6, 2\}) = 109$.

$s = 8$. $\tilde{U} = \{4, 7, 3, 1, 6, 2, 5\}$, $(F|\tilde{U} = \{4, 7, 3, 1, 6, 2\})$. Дерево решений задачи, построенное методом ветвей и границ описанным алгоритмом решения, приведено на рис. 2. Слева или справа от каждой вершины стоит значение нижней границы критерия оптимальности. Времена начала и завершения выполнения заданий на машинах и постобработки на различных стадиях изготовления в оптимальной последовательности их обработки приведены в табл. 2.

Таблица 2

Времена начала и завершения выполнения заданий в оптимальном решении

Последовательность выполнения заданий	Времена начала и завершения выполнения заданий								
	1-я стадия			2-я стадия			3-я стадия		
	x_i^1	θ_i^1	T_i^1	x_i^2	θ_i^2	T_i^2	x_i^3	θ_i^3	T_i^3
4	1	8	15	16	23	28	29	41	43 < 72
7	9	23	27	28	33	38	42	49	53 < 55
3	24	33	38	39	45	48	50	61	65 < 66
1	34	45	47	48	59	63	64	72	75 < 80
6	46	51	57	60	68	72	73	83	88 < 93
2	52	66	69	70	72	75	84	93	98 < 102
5	67	75	79	80	89	91	94	103	109 < 115

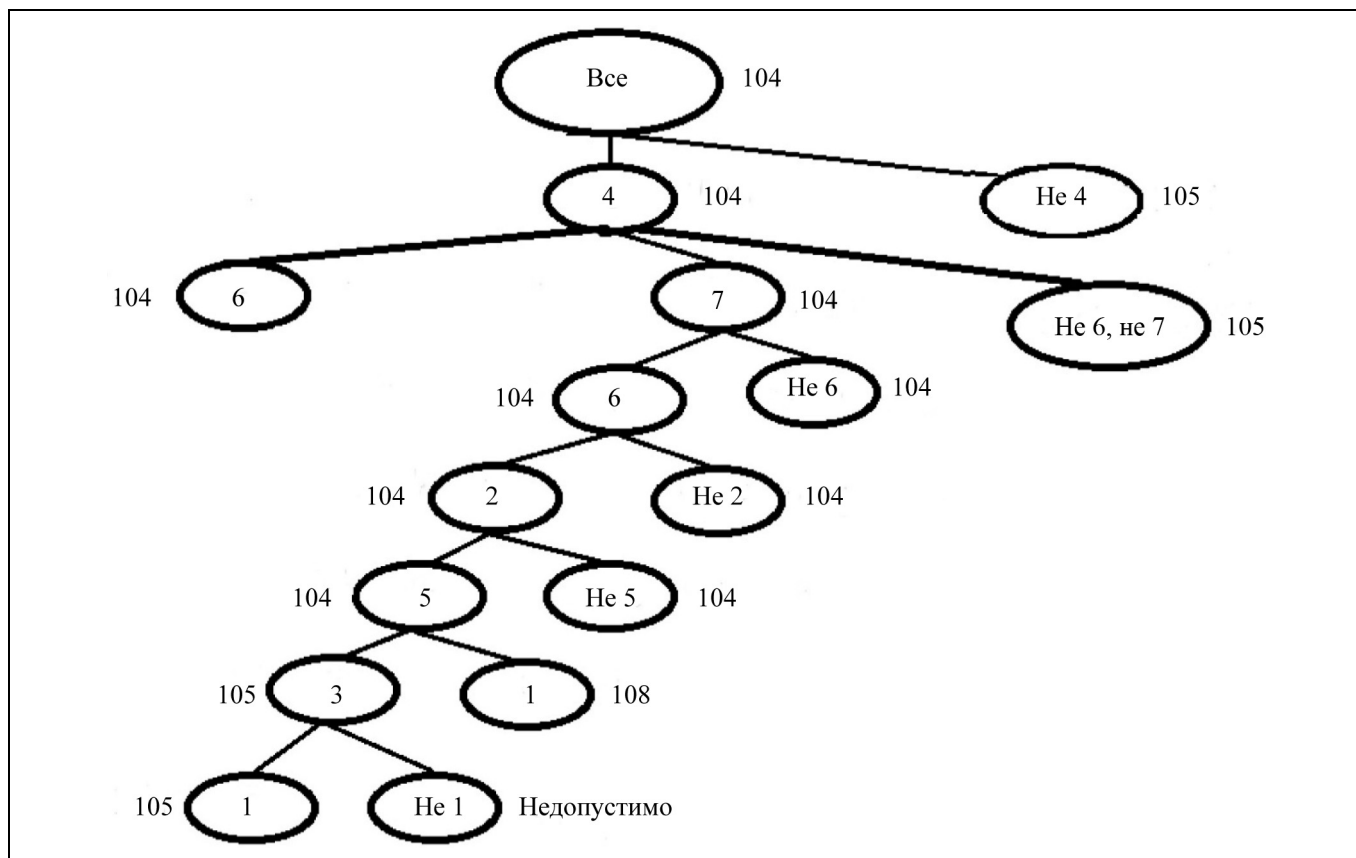


Рис. 2. Дерево решений примера

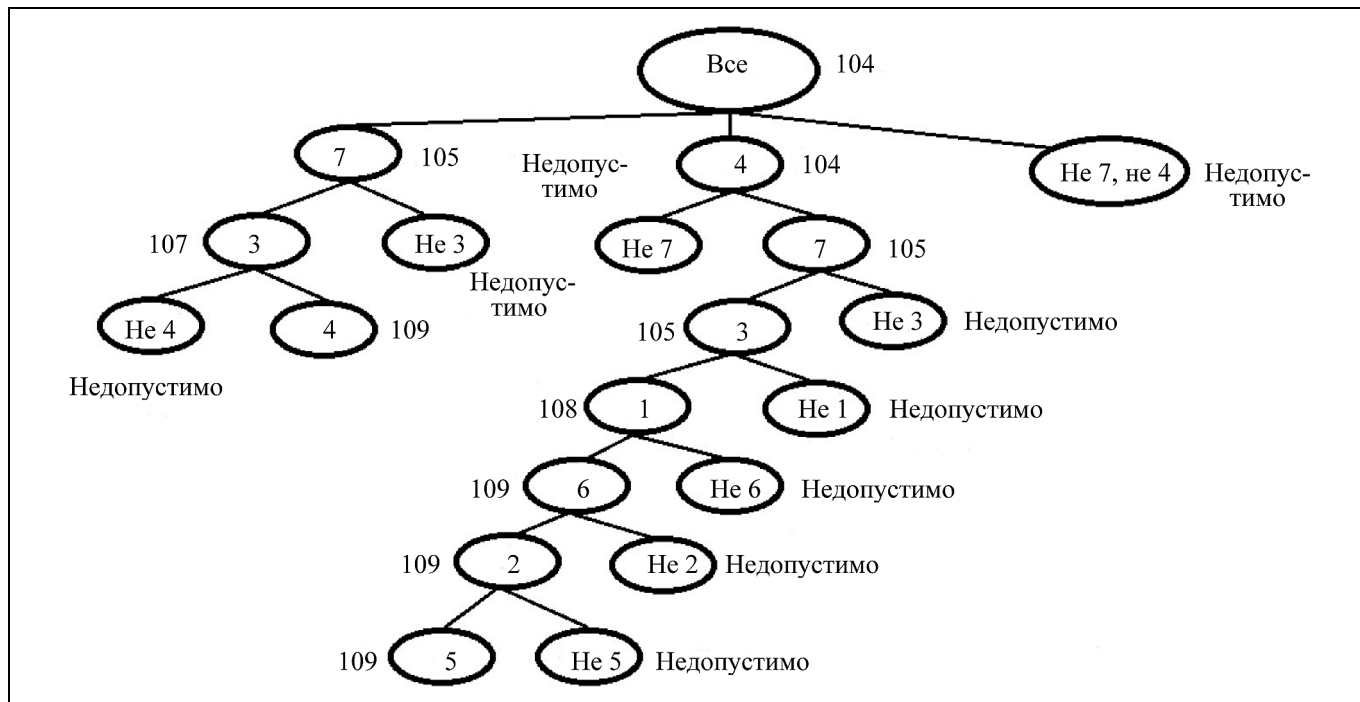


Рис. 3. Дерево приближенного решения задачи

Времена начала и завершения выполнения заданий

Последовательность выполнения заданий	Времена начала и завершения выполнения заданий								
	1-я стадия			2-я стадия			3-я стадия		
	x_i^1	θ_i^1	T_i^1	x_i^2	θ_i^2	T_i^2	x_i^3	θ_i^3	T_i^3
4	1	8	15	16	23	28	29	41	43
7	9	23	27	28	33	38	42	49	53
6	24	29	35	36	44	48	50	60	65
2	30	44	47	48	50	54	61	70	75
5	45	53	57	58	67	69	71	80	86
3	54	63	68	69	75	78	81	92	96
1	64	75	77	78	89	93	94	102	105

Рассмотрим также приближенное решение данного примера в условиях, когда ограничения на времена завершения выполнения заданий отсутствуют, т. е. $D_i \rightarrow \infty$, $i = 1, \dots, 7$. Применим одностороннюю схему ветвления, не обращая внимание на то, что в дереве решений есть вершины с меньшим значением нижней границы функции цели. Дерево решений задачи в этом случае представлено на рис. 3.

Времена начала и завершения выполнения заданий на машинах и постобработки на различных стадиях изготовления в оптимальной последовательности их обработки при отсутствии ограничений на времена завершения их выполнения приведены в табл. 3.

Значение критерия оптимальности в полученном приближенном решении очень близко к значению нижней границы длины расписания.

ЗАКЛЮЧЕНИЕ

Рассмотренные в работе постановки и методы решения задач построения расписаний работы последовательных цепочек участков и цехов взаимосвязанных производств могут найти широкое применение в системах календарного планирования мелко- и среднесерийного производства. Существенная особенность рассматриваемых задач состоит в учете затрат времени не только на работу оборудования, но и потерь времени на постобработку на каждой стадии изготовления, а также в наличии системы ограничений на сроки начала выполнения заданий на 1-й стадии обработки и завершения их выполнения на последней стадии после постобработки. В качестве критерия оптимальности принято построение последовательности вы-

полнения всех заданий системой взаимосвязанных производств в кратчайшие сроки. Сформулированные задачи обобщают известную в теории расписаний задачу построения последовательности выполнения заданий на одной машине с учетом постобработки и относятся к классу NP-полных задач экспоненциальной сложности. Алгоритмы получения точных решений таких задач применимы, в самом общем случае, только для решения задач малой размерности. В статье приведены математическая постановка, формулы выражения вычисления нижней границы значения критерия оптимальности, а также свойства допустимых последовательностей выполнения заданий и правила отсева недопустимых и неоптимальных продолжений.

Предложены алгоритмы получения точных и приближенных решений задачи на основе модифицированного метода ветвей и границ. Полученные результаты проиллюстрированы числовым примером, выполненные автором вычислительные эксперименты показали, что наличие системы строгих ограничений на сроки выполнения заданий при реализации приведенных в работе алгоритмов в ряде случаев существенно сокращает число рассматриваемых вариантов. Это позволяет при сравнительно небольшом объеме вычислений получить допустимые и даже оптимальные или очень близкие к оптимальным решения, а также на начальных этапах решения установить несовместность системы ограничений задачи.

Предложенные в работе алгоритмы применимы для практического применения в автоматизированных системах календарного планирования производств, строительства объектов, выполнения ремонтных работ и в других приложениях.

ЛИТЕРАТУРА

1. Лазарев А.А. — Теория расписаний. Методы и алгоритмы. — М. — ИПУ РАН. — 2019. — 408 с. [Lasarev, A.A. Schedule theory. Methods and algorithms. — M. — IPU RAN. — 2019. — 408 p. (In Russian)]
2. Зак Ю.А. Прикладные задачи теории расписаний и маршрутизации перевозок. — М.: URSS, 2012. — 394 с. [Zack, Yu.A. Applied Problems of the Theory of Schedules and Traffic Routing. — M., URSS, 2012. — 394 p. (In Russian)]
3. Хоботов Е.Н. О некоторых моделях и методах решения задач планирования в дискретных производствах // Автоматика и телемеханика. — 2007. — № 12. — С. 85—100. [Hobotov, E.N. On Some Models and Methods for Solving Planning problems in Discrete Production // Automation and Remote Control. — 2007. — Vol. 68, iss. 12. — P. 2172—2186.]
4. Прилуцкий М.Х., Седиков Д.В. Календарное планирование многостадийных производственных систем с взаимозаменяемым оборудованием // Вестник Нижегородского университета им. Н.И. Лобачевского. — 2014. — № 4 (1). — С. 433—437. [Priluzkij, M.H., Sedakov, D.V. Scheduling of Multistage Production Systems with Interchangeable Equipment // Bulletin of Nizhny Novgorod University im. N.I. Lobachevsky. — 2014. — No. 4 (1). — P. 433—437. (In Russian)]
5. Domschke, W., Scholl, A., Voß, S. Produktionsplanung. Ablauforganisatorische Aspekte. — Springer Verlag: Berlin — Heidelberg, 2005. — 456 s.
6. Brucker P. Scheduling Algorithms. — Springer-Verlag: Berlin — Heidelberg — New York, 1998. — 377 p.
7. Зак Ю.А. Построение двухстадийных расписаний обработки изделий на одной машине. — Системні дослідження та інформаційні технології. — 2018. — № 4. — С. 19—36. [Zack, Yu.A. Construction of two-stage product processing schedules on one machine // Systems research and information technology. — 2018. — No. 4. — P. 19—36.]
8. Зак Ю.А. Двухстадийные задачи планирования для поточной линии // Проблемы управления. — 2019. — № 6. — С. 52—62. [Zack, Yu.A. Two-stage planning tasks for the flow // Control Sciences. — 2019. — No. 6. — P. 52—62. (In Russian)]
9. Зак Ю.А. Алгоритмы приближенного решения многостадийных Flow-shop-problem // Системні дослідження та інформаційні технології. — 2019. — № 3. — С. 100—109. [Zack, Yu.A. Algorithms for the approximate solution of multistage Flow-shop-problem // Systems research and information technology. — 2019. — No. 3. — P. 100—109.]
10. Carlier, J. The one-machine sequencing problem // European Journal of Operational Research. — 1982. — Vol. 11. — P. 42—47.
11. Зак Ю.А. Построение допустимых и оптимальных расписаний выполнения работ на одной машине // Кибернетика и системный анализ. — 2012. — № 1. — С. 51—66. [Zack, Yu.A. Construction of admissible and optimal schedules of work execution on one machine // Cybernetics and Systems Analysis. — 2012. — No. 1. — P. 51—66. (In Russian)]

Статья представлена к публикации членом редколлегии А.А. Лазаревым.

Поступила в редакцию 19.02.2020, после доработки 7.07.2020.

Принята к публикации 16.07.2020.

Зак Юрий Александрович — д-р техн. наук, эксперт и консультант, г. Аахен, Германия, ✉ yuriy_zack@hotmail.com.

SCHEDULES FOR PERFORMING TASKS IN INTERCONNECTED SEQUENTIAL PRODUCTION SYSTEMS

Yu.A. Zack

✉ yuriy_zack@hotmail.com

Abstract. The problem, which is classical in scheduling theory, of constructing a sequence of tasks execution on one machine under conditions of restrictions on the start and end times of tasks execution and which takes into account not only the time spent on equipment operation, but also the losses for post-processing, is considered for multistage production systems consisting of an interconnected chain of sections and workshops of an industrial enterprise. The criterion for the optimality of the task is the implementation of the multistage schedule in the shortest possible time. The problems considered in this paper belong to the class of NP-complete problems of exponential complexity. The properties of admissible and optimal task execution sequences are investigated. Methods for calculating the lower bound for the length of the optimal schedule and the rules for rejecting inadmissible and non-optimal extensions are proposed. Algorithms for the exact and approximate solution of the problem by modified branch and bound methods have been developed. The proposed algorithms are illustrated with numerical examples. The computational experiments performed by the author have shown that the presence of a system of strict restrictions on the timing of tasks when implementing the algorithms proposed in the paper in a number of cases significantly reduces the number of options under consideration.

Keywords: sequence of task execution, multistage schedule, minimum time, heuristic algorithm, lower bound of the optimality criterion, screening out unpromising continuations, branch and bound method.