УДК 519.8
DOI: 10.20535/SRIT.2308-8893.2020.1.09

ЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ ПОСТРОЕНИЯ ЭФФЕКТИВНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ВЫПОЛНЕНИЯ ЗАДАНИЙ НА ОДНОЙ МАШИНЕ ВО ВЗАИМОСВЯЗАННЫХ ПРОИЗВОДСТВЕННЫХ СИСТЕМАХ

Ю.А. ЗАК

Аннотация. Классическая в теории расписаний задача построения последовательности выполнения заданий на одной машине, учитывающая не только затраты времени на работу оборудования, но и потери на постобработку, рассматривается для многостадийных производственных систем, состоящих из взаимосвязанной цепочки участков и цехов промышленного предприятия. В качестве критерия оптимальности рассматривается выполнение многостадийного расписания в кратчайшие сроки. Предложены методы расчета нижней границы длины оптимального расписания и эвристические алгоритмы получения приближенных решений, требующие небольших объемов вычислений. Предложенные алгоритмы иллюстрируются числовыми примерами.

Ключевые слова: последовательности выполнения заданий, многостадийные расписания, минимальное время, эвристический алгоритм, нижняя граница значения критерия оптимальности.

ВВЕДЕНИЕ

Построение расписаний в многостадийных системах имеет много практических приложений в машиностроении и приборостроении, автомобильной, электронной, деревообрабатывающей, легкой, пищевой и других отраслях промышленности, а также в организации технического и сервисного обслуживания объектов. Календарное планирование работы 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 договорами сроки и с наилучшими технико-экономическими показателями. В монографиях и большинстве публикаций по теории расписаний в периодической литературе в различных постановках и с различными критериями оптимальности рассматривались математические модели, точные и приближенные методы решения классических задач построения последовательностей выполнения заданий на одной машине применительно только к одному структурному подразделению предприятия. Наибольший интерес представляет учет времени, необходимого на постобработку после завершения изготовления изделий на машине. В качестве критериев оптимальности выбирается выполнение всего комплекса работ в кратчайшие сроки [1–5]. При этом независимые друг от друга расписания для каждого отдельного участка могут оказаться совершенно неэффективными для общей многостадийной системы, включающей последовательную цепочку, состоящую из нескольких участков. Задачам построения многостадийных расписаний выполнения заданий на одной машине, которые имеют большое практическое значение и ярко выраженную специфику, не уделялось достаточного внимания в литературе. Практически важные постановки и пути решения задач построения двухстадийных расписаний рассматривались в работах М.З. Згуровского и А.А. Павлова [3], а в случае, когда второй стадией обработки является процесс сборки изделий, — в публикациях Е.Н. Хоботова [11]. В работах автора [4, 6, 7] рассматривались постановки, математические модели, точные и приближенные методы решения некоторых классических задач в многостадийных производственных системах.

Задачи теории расписаний для многостадийных производственных систем, как правило, относятся 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 является чрезвычайно актуальным для построения систем календарного планирования производства, планирования ремонтных работ, обслуживания объектов и работы транспорта.

ПОСТАНОВКА И МАТЕМАТИЧЕСКАЯ ФОРМУЛИРОВКА ЗАДАЧИ

Технологический процесс предусматривает изготовление некоторого подмножества различных не связанных друг с другом изделий на нескольких стадиях обработки. Такими стадиями обработки могут быть стоящие в последовательной цепочке станки, многопроцессорные и многопоточные или автоматические линии, а также различные структурные единицы предприятия. В дальнейшем, как и во всех классических постановках задач построения последовательности выполнения заданий на одной машине, каждая стадия обработки рассматривается как одна машина, т.е. на каждой стадии производится обработка изделий на одной машине. Изготовление изделия на последней стадии обработки рассматривается как выполнение задания. Если даже технологией производства может предусматриваться на каждой стадии обработки изделия выполнение некоторого подмножества операций, задано суммарное время обработки на каждой стадии. После изготовления на технологическом оборудовании каждой стадии производится постобработка каждого изделия, связанная с контролем, испытанием, необходимым временем пролеживания (например, с охлаждением или нагреванием), оформлением необходимой документации, транспортными потерями времени и т.п., что наряду с необходимым временем на изготовление также требует затрат времени. На всех стадиях обработки изготовление изделий на технологическом оборудовании производится в одной и той же последовательности, и на оборудовании каждой из этих стадий ведется без прерываний. После изготовления изделия на k-й стадии и последующей постобработки изделие поступает на следующую стадию обработки. На машине каждой стадии обработки может начаться изготовление следующего в последовательности изделия непосредственно после завершения изготовления предыдущего изделия. Начало обработки каждого изделия на следующем стоящем в последовательной цепочке участке может начаться только после завершения постобработки его на предыдущей стадии. Необходимо найти последовательность изготовления изделий, которая обеспечит минимальное время выполнения расписания на всех стадиях обработки. Эффективные алгоритмы точного решения такой задачи в одностадийных системах, учитывающей потери времени на постобработку с помощью Schrage-algorithms и его модификации впервые были предложены в работе J. Carlier (1982) [8] и развиты в условиях различного вида ограничений в работах автора [2, 4, 12].

Введем следующие 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, 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 машинах k-й стадии обработки;

z_i^k, r_i^k, 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;

$F = \max\limits_{i \in I} T_i$ — время завершения всех работ многостадийного расписания, т.е. значение критерия оптимальности задачи.

Пусть построена некоторая последовательность выполнения заданий

$$\widetilde{U} = \{u_1, u_2, \dots, u_{j-1}, u_j, \dots, u_n\}.$$

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

$$x_{u_1}^1 = d_{u_1}^1 + 1,\ \theta_{u_1}^1 = x_{u_1}^1 + t_{u_1}^1 - 1,\ z_{u_1}^1 = \theta_{u_1}^1 + 1,\ Z_{u_1}^1 = z_{u_1}^1 + r_{u_1}^1 - 1; \tag{1}$$

$$x_{u_1}^k = \max(d_{u_1}, \theta_{u_1}^{k-1}) + 1,\ \theta_{u_1}^k = x_{u_1}^k + t_{u_1}^k - 1,\ z_{u_1}^k = \theta_{u_1}^k + 1,$$

$$Z_{u_1}^k = z_{u_1}^k + r_{u_1}^r - 1 \quad k = 2,\dots,K, \tag{2}$$

$$x_{u_j}^1 = \max(d_{u_j}, Z_{u_{j-1}}^K) + 1,\ x_{u_j}^k = \max(d_{u_j}, \theta_{u_j}^{k-1}) + 1,\ \theta_{u_j}^k = x_{u_j}^k + t_{u_j}^k - 1,\ z_{u_j}^k = \theta_{u_j}^k + 1,$$

$$Z_{u_j}^k = z_{u_j}^k + r_{u_j}^r - 1,\ k = 2,\dots,K;\ j = 2,\dots,n; \tag{3}$$

$$T_{u_j} = Z_{u_j}^K,\quad j = 1,\dots,n. \tag{4}$$

Необходимо найти последовательность выполнения независимых друг от друга и одинаковую на всех стадиях обработки заданий, обеспечивающую выполнение критерия оптимальности $F = \min T_{u_n}$. В качестве дополнительных ограничений могут использоваться условия

$$x_{u_1}^1 \geq d_{u_1}^1, \quad Z_{u_1}^k \leq D_{u_j}, \quad j = 1,...,n.$$

НИЖНЯЯ ГРАНИЦА ДЛИНЫ ОПТИМАЛЬНОГО РАСПИСАНИЯ

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

$$f^k = \min_{1 \leq i \leq n}\left(d_i^k + \sum_{i=1}^{n} t_i^k\right) + \min_{1 \leq i \leq n} r_i^k \cdot, \quad k = 1,...,K. \tag{5}$$

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

$$\mu^l = \min_{1 \leq i \leq n}(d_i^l + t_i^l + r_i^l), \quad l = 1,...,(k-1),(k+1),...,K, \quad l \neq k. \tag{6}$$

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

$$\vartheta^k(F) = f^k + \sum_{\substack{l=1 \\ l \neq k}}^{K} \mu^l, \quad k = 1,...,K. \tag{7}$$

Учитывая вышеизложенное, значение нижней границы длины многостадийного расписания вычисляется по формуле

$$\vartheta(F) = \max_{1 \leq k \leq K}\left(f^k + \sum_{l=1, l \neq k}^{K} \mu^l\right). \tag{8}$$

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

$$f^k(s) = \min_{i \in \widetilde{I}^2(s)}[\max(d_i^l, \theta_{u_m(s)}^l)] + \min_{i \in \widetilde{I}^2} r_i^l)], \quad k = 1,...,K;$$

$$\mu^l(s) = \min_{i \in \widetilde{I}^2(s)}[\max(d_i^l, \theta_{u_m(s)}^l)] + t_i^l + r_i^l)], \quad l = 1,...,(k-1),(k+1),...,K, \quad l \neq k.$$

Нижняя граница длины расписания для подпоследовательности выполнения подмножества заданий вычисляется по формуле

$$\vartheta[\widetilde{U}(s)] = \max_{1\le k\le K}\left\{ f^k(s) + \sum_{l=1,l\ne k}^{K} \mu^l(s)\right\}.$$

ЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ ПОЛУЧЕНИЯ ПРИБЛИЖЕННЫХ РЕШЕНИЙ

Алгоритм 1

Обозначим $\widetilde{I} = \{1,\dots,n\}$ — множество всех изделий, подлежащих обработке;

$s = 1,\dots,S = n$ — последовательные этапы выбора изделий, включаемых в обработку, т.е. выбор очередного члена последовательности, стоящего на последнем месте в строящейся подпоследовательности;

$\widetilde{I}^1(s)$, $\widetilde{I}^2(s)$ — соответственно подмножество изделий, для которых уже в процессе выполнения алгоритма определено и не определено место в последовательности обработки на s-м этапе принятия решений: $\widetilde{I}^1(s)\bigcup\widetilde{I}^2(s) = \widetilde{I}$, $\widetilde{I}^1(s)\bigcap\widetilde{I}^2(s) = \varnothing$;

$\widetilde{U}(s) = \{u_1(s), u_2(s),\dots,u_{j-1}(s), u_j(s),\dots,u_n(s)\}$, $m \le n$; — построенная подпоследовательность выполнения заданий многостадийной обработки на s-м этапе принятия решений;

$d_i^k(s), x_i^k(s), \theta_i^k(s), z_i^k(s), T_i(s) = Z_i^k(s)$ — значения выше определенных показателей на s-м этапе принятия решений.

В начале процесса, когда $s = 0$, положим $\widetilde{I}^1(s) = \varnothing$, $\widetilde{I}^2(s) = \{i = 1,2,\dots,n\}$.

Шаг 0. На этом шаге вычисляем значения параметров $d_i^k(s=0)$, $x_i^k(s=0)$, $\theta_i^k(s=0)$, $z_i^k(s=0)$, $k = 1,\dots,K$, а также $T_i(s=0) = Z_i^K(s=0)$, для всех подлежащих обработке изделий, $i = 1,2,\dots,n$ по формулам (1), (2), (4), (5).

Определяем $\theta_{u_1}^K(s=0) = \max\limits_{1\le i\le n}\theta_i^K(s=0)$. Устанавливаем изделие с индексом $i = u_1$ на 1-е место в строящейся последовательности $\widetilde{U}(s+1) = \{u_1\}$. Если значение $\theta_{u_1}^K(s=0)$ справедливо для нескольких индексов, то в качестве $i = u_1$ выбирается изделие с наименьшим значением индекса или изделие с наименьшем значением $\theta_{u_1}^{K-1}(s=0)$.

Полагаем $\widetilde{I}^2(s) = \{\widetilde{I}/u_1\}$. Вычисляем $T_{u_1} = Z_{u_1}^K$.

В дальнейшем алгоритм предусматривает выполнение следующих шагов $s = 1,\dots,(n-1)$.

Шаг s. На s-м этапе решения, когда построена подпоследовательность выполнения заданий $\widetilde{U}(s) = \{u_1, u_2,\dots,u_j\}$, для каждого из подмножества изделий выполняем вычисления следующих параметров:

$$x_{u_i}^1(s) = \max[d_{u_i}^1, Z_{u_{i-1}}^K] + 1,\ x_{u_i}^k(s) = \max[d_{u_i}^k, \theta_{u_{i-1}}^k] + 1,\ i \in \widetilde{I}^2(s),\ k = 2,\dots,K; \quad (9)$$

$$\theta_i^k(s) = x_i^k(s) + t_i^k,\ Z_i^k(s) = \theta_i^k + r_i^k,\ i \in \widetilde{I}^2(s),\ k = 1,\dots,K; \quad (10)$$

Вычисляем $\theta^K_{u_{j+1}}(s)=\min\limits_{i\in\widetilde{I}^2(s)}\theta^K_i(s)$. Если значение $\theta^{K-1}_{u_{j+1}}(s=0)$ справедливо для нескольких индексов, то в качестве $i=u_{j+1}$ из подмножества выбирается изделие с наименьшим значением индекса или изделие с наименьшем значением $\theta^{K-1}_{u_{j+1}}(s=0)$.

Определяем

$\widetilde{I}^1(s+1)=\{\widetilde{I}^1\bigcup u_{j+1}\}$, $\widetilde{I}^2(s+1)=\{\widetilde{I}^2(s)/u_{j+1}\}$, $\widetilde{U}(s+1)=\{u_1,u_2,...,u_j,u_{j+1}\}$.

Находим $T_{u_{j+1}}=Z^K_{u_{j+1}}(s)$. Если выполняются условия

$$(j+1=n),\ \widetilde{I}^2(s+1)=\varnothing,\ \ \widetilde{I}^1(s+1)=\widetilde{I}\ , \tag{11}$$

то построенная последовательность $\widetilde{U}(s+1)=\{u_1,u_2,\dots,u_j,u_{j+1}\}$ является решением задачи. Время выполнения расписания определяется выражением

$$F=\max_{1\le j\le n}T_{u_j}\ .$$

Если условия (11) не выполняются, то полагаем $s:=(s+1)$ и снова выполняем s -й шаг.

Алгоритм 2

Для решения задачи на двух машинах, что эквивалентно двухстадийной обработке на одной машине без постобрабртки и отсутствия ограничений на сроки выполнения заданий, наиболее эффективным является полиномиальный по времени алгоритм Джонсона [12, 13]. Все изделия разбиваются на 2 группы. Первая группа включает изделия, для которых $\bar{t}^1_i\le\bar{t}^2_i$, а вторая — для которых $\bar{t}^1_i>\bar{t}^2_i$. Изделия 1-й группы упорядочиваются по неубыванию времен $\bar{t}^1_i$ в последовательности $\widetilde{V}^1=\{v^1_1,v^1_2,...,v^1_{l-1},v^1_l,...,v^1_L\,|\,\bar{t}^1_{l-1}\le\bar{t}^1_l,\ \ l=1,...,L\}$, а вторые — по невозрастанию времени $\bar{t}^2_i$ в последовательности $\widetilde{V}^2=\{v^2_1,v^2_2,...,v^2_{r-1},v^2_r,...,v^2_R\,|\,\bar{t}^1_{r-1}\ge\bar{t}^2_r,\ r=2,...,R\}$. Здесь L и R — соответственно количество заданий в подмножествах $\widetilde{V}^1$ и $\widetilde{V}^2$. Обработка изделий на двух машинах осуществляется в последовательности

$$\widetilde{V}=\{v^1_1,v^1_2,...,v^1_{l-1},v^1_l,...,v^1_L,v^2_1,v^2_2,....,v^2_{r-1},v^2_r,...,v^2_R\}. \tag{12}$$

Доказано [13], что в случае 2 машин данный алгоритм обеспечивает оптимальное решение задачи и имеет полиномиальную сложность порядка $O(n^2)$ Данная эвристика может быть использована для получения приближенного решения сформулированной в данной работе задачи многостадийной обработки. При этом не принимается во внимание допустимое время начала выполнения заданий на всех стадиях обработки, кроме первой, а время постобработки на каждой стадии рассматривается как фиктивное время обработки на некоторой другой машине. Рассмотрим один из вариантов расчета значений $\bar{t}^1_i$ и $\bar{t}^2_i$.

Определим $t_i^1 := t_i^1 + d_i^1$, $t_i^k := t_i^k$, $k = 2,\ldots,K$. Пусть $m = \left|K/2\right|$, где $\left|K/2\right|$ — целая часть результата деления двух величин,

$$\bar{t}_j^1 = (d_j^1 + t_j^1 + t_j^1) + \sum_{k=2}^{m} (f_j^k + r_j^k)\,,\ \bar{t}_j^2 = \sum_{k=(m+1)}^{K} (t_j^k + r_j^k)\,.$$

Решаем задачу построения расписания для 2 машин описанным выше алгоритмом Джонсона. В качестве решения задачи выбираем вариант и соответствующую последовательность двухстадийной обработки $\widetilde{V}$. При выполнении заданий в последовательности $\widetilde{V}$, построенной в соответствии с выражением (12), времена начала и завершения выполнения заданий и постобработки на машинах всех стадий обработки определяем по формулам (1)–(5). Описанный алгоритм, как и алгоритм Джонсона, имеет полиномиальную сложность и требует только в 2 раза большего объема вычислений.

ИЛЛЮСТРАТИВНЫЙ ПРИМЕР

Параметры затрат времени на изготовление и постобработку каждого изделия на каждой стадии технологического процесса сведены в табл. 1.

Т а б л и ц а 1

Ноиер заданий	Временные параметры на различных стадиях обработки $k = 1,2,3$								
	1-я стадия			2-я стадия			2-я стадия		
	d_i^1	t_i^1	r_i^1	d_i^2	t_i^2	r_i^2	d_i^3	t_i^3	r_i^3
1	2	7	8	12	10	3	3	15	10
2	4	8	6	4	7	5	15	10	15
3	0	12	5	3	8	6	8	9	4
4	5	5	10	0	12	4	10	8	10
5	3	16	3	10	4	6	6	7	7
6	5	10	5	0	10	7	20	10	6

Нижняя граница длины многостадийного расписания, вычисленная по формулам (5)–(8), равна

$$\vartheta(F) = \max[(58+16+18),(17+50+18),(17+16+59)] = 92\,.$$

Решение примера по алгоритму 1

На каждом шаге вычисляем по формулам (9), (10) значения $i \in J^2(s)$:

$$\theta_1^3(1) = (2+7+8)+(10+3)+15 = 45\,,\quad \theta_2^3(1) = (8+9+6)+(7+5)+10 = 41\,,$$

$$\theta_3^3(1) = (0+12+5)+(8+6)+9 = 40\,,\quad \theta_4^3(1) = (5+10+15)+(2+4)+8 = 41\,,$$

$$\theta_5^3(1) = (3+16+3)+(6+10)+7 = 45\,,\quad \theta_6^3(1) = (5+10+4)+(9+7)+10 = 45\,,$$

$$\theta_{u_{j+1}}^K(s) = \min_{i\in\widetilde{I}^2(s)} \theta_i^K(s)\,,\quad \min_{i\in\widetilde{I}^2(s)} \theta_i^p(1) = \theta_1^3(1) = 40\,.$$

Устанавливаем 3-е задание на 1-е место в строящейся последовательности.

Последовательность выполнения заданий

$$\widetilde{W} = \{w_1(1),...,w_6(1) \,|\, \theta^3_{w_{i-1}(1)} \le \theta^3_{w_i(1)}\} = \{3,2,4,5,6\}.$$

Требующая небольшого объема вычислений последовательность выполнения заданий и построенная по неубыванию значений $\theta_i^2(1)$, может рассматриваться как приближенное решение задачи. Время начала и время завершения выполнения заданий на различных стадиях обработки в последовательности, построенной в оптимальном расписании, приведены в табл. 2.

Т а б л и ц а 2

Последовательность выполнения заданий	Время начала и время завершения выполнения заданий								
	1-я стадия			2-я стадия			2-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
3	1	12	17	18	25	31	32	40	44
2	13	20	26	27	33	38	41	50	55
4	21	25	35	36	47	51	52	59	69
1	26	32	40	48	57	60	61	75	85
5	33	48	51	58	61	67	76	82	89
6	49	58	63	64	73	80	83	92	98

Решение примера по алгоритму 2

Вычислим значения параметров

$$\bar{t}_j^1 = (d_j^1 + t_j^1 + r_j^1) + \sum_{k=2}^{3}(t_j^k + r_j^k), \quad \bar{t}_j^2 = (t_j^k + r_j^k), \quad j = 1,...,6,$$

$$\bar{t}^1 = (27,26,25,32,26,28), \quad \bar{t}^2 = (28,20,19,22,20,23), \quad \widetilde{V}^1 = \{1\}, \quad \widetilde{V}^2 = \{6,4,2,5,3\}.$$

Следовательно, последовательность выполнения заданий многостадийной обработки имеет вид $\widetilde{V} = \{1,6,4,2,5,3\}$. Время начала и время завершения выполнения всех заданий, а также завершения постобработки на всех стадиях обработки сведены в табл. 3.

Т а б л и ц а 3

Последовательность выполнения заданий	Время начала и время завершения выполнения заданий								
	1-я стадия			2-я стадия			2-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
1	3	9	17	18	27	30	31	45	48
6	10	19	24	28	37	44	46	55	61
4	20	24	34	38	49	53	56	63	73
2	25	32	38	50	56	61	64	73	88
5	33	48	51	57	60	66	74	80	87
3	49	60	65	66	73	79	81	89	93

Следовательно, по алгоритму 2 получено более эффективное решение. Значение критерия оптимальности очень близко к значению нижней границы.

ЗАКЛЮЧЕНИЕ

Рассмотренные в работе постановки и приближенные 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. В качестве критерия оптимальности рассматривается построение последовательности выполнения всех заданий системой взаимосвязанных производств в кратчайшие сроки. Сформулированные задачи, учитывающие не только затраты времени на работу оборудования, но и временные потери на постобработку изделий, относятся к классу NP-полных задач экспоненциальной сложности. Алгоритмы получения точных решений таких задач применимы на практике только для решения задач малой размерности. В работе приведены математическая постановка, формульные выражения вычисления нижней границы значения критерия оптимальности и 2 эвристических алгоритма решения сформулированных задач. Полученные результаты иллюстрируются числовым примером. Предложенные в работе эвристические алгоритмы получения приближенных решений этих задач применимы для практического использования в автоматизированных системах календарного планирования. Проведенные автором вычислительные эксперименты показывают, что полученное приближенное решение не превосходит значение нижней границы критерия оптимальности более, чем на 5–10%.

ЛИТЕРАТУРА

1. *Conway R.W.* Theory of Scheduling / R.W. Conway, W.L. Maxwell, L.W. Miller. — Addison-Wesley Publishing Company, 1967. — 294 p.
2. *Zack Yu.A.* Applied problems of the theory of scheduling and traffic routing [in Russian] / Yu.A. Zack. — M., URSS, 2012. — 394 p.
3. *Zgurovsky M.Z.* Decision making in networked systems with limited resources [in Russian] / M.Z. Zgurovsky, A.A. Pavlov. — K.:. Nauk. dumka, 2010. — 573 p.
4. *Zack Yu.A.* Construction of two-stage schedules of processing of products on one machine / Yu.A. Zack // System research and information technologies. — 2018. — № 4. — P. 19–36.
5. *Zack Yu.A.* Developing admissible and optimal schedules of works on one machine / Yu.A. Zack // Cybernetics and systems analysis. — № 1. — 2012.
6. *Zak Yu.A.* Two-stage planning tasks for the flow line / Yu.A. Zak // Control Sciences. — M., 2019. — № 6. — P. 52–62.
7. *Zack Yu.A.* Algorithms for approximate multi-stage Flow-Shop-Problem solution / Yu.A. Zack // System research and information technologies. — 2019. — № 3. — P. 100–109.
8. *Carlier J.* The one-machine sequencing problem / J. Carlier // European Journal of Operational Research. — 1982. — N 11. — P. 42–47.
9. *Domschke W.* Produktionsplanung. Ablauforganisatorische Aspekte / W. Domschke, A. Scholl, S. Voß. — Berlin, Heidelberg: Springer Verlag, 2005. — 456 p.
10. *Brucker P.* Scheduling Algorithms / P. Brucker // Springer-Verlag, Berlin, Heidelberg und New York, 1998. — 377 p.
11. *Sidorenko A.M.* Work planning and scheduling subject to modules and articles assembly / A.M. Sidorenko, E.N. Khobotov // Automation in industry. — 2012. — № 10. — P. 21–25.
12. *Zak Ju.A.* Properties of admissible and optimum sequences of performance of works on a single machine / Ju.A. Zak // Control Sciences. — 2012. — № 5. — P. 54–61.

Поступила 12.02.2020