УДК 519.854.2

СПЕРКАЧ М.О.

ЗАДАЧА ВИЗНАЧЕННЯ МАКСИМАЛЬНО ПІЗНЬОГО МОМЕНТУ ПОЧАТКУ ВИКОНАННЯ ЗАВДАНЬ ІЗ СПІЛЬНИМ ЖОРСТКИМ ДИРЕКТИВНИМ ТЕРМІНОМ ПАРАЛЕЛЬНИМИ ПРИСТРОЯМИ РІЗНОЇ ПРОДУКТИВНОСТІ

В статті досліджені властивості задачі календарного планування виконання завдань зі спільним жорстким директивним терміном паралельними пристроями різної продуктивності з метою максимізації моменту запуску пристроїв за умови, що усі завдання не запізнюються. Сформульована допоміжна оптимізаційна задача, за результатами якої визначені достатні умови оптимальності розкладів. На основі достатніх умов оптимальності визначено множину перестановок, які дозволяють послідовно покращувати значення критерію. Розроблена поліноміальна складова ПДС-алгоритма розв'язання задачі.

This article studies the properties of the problem of scheduling tasks with a common tough due date on parallel machines with different speeds to maximize the launch moment of the machines with the condition that all the tasks are not delayed. An auxiliary optimization problem is formulated, the results of which are the sufficient conditions for a schedule optimality. On their basis the set of permutations is defined that allow to consistently improve the criterion value. The polynomial component of PDC-algorithm for the problem solution is developed.

1. Вступ

Оперативно-календарне планування виробництва є частиною внутрішньоцехового планування та полягає в організації ритмічної роботи виробництва з випуску продукції, в заданих обсягах і у встановлені терміни при найбільш ефективному використанні всіх виробничих ресурсів підприємства. Змістом внутрішньоцехового планування є розробка оперативних планів і складання поточних графіків роботи виробничих ділянок. Для цього доцільно застосовувати моделі та методи теорії розкладів.

Стаття присвячена дослідженню однієї задачі теорії розкладів.

2. Постановка задачі

Задано множину завдань $J = \{1,2,...,n\}$ та кількість пристроїв *m*. Пристрої працюють паралельно і є взаємозамінними у тому сенсі, що кожний з пристроїв може виконувати будь-яке завдання з множини J. Пристрої відрізняються один від одного продуктивністю виконання завдань. При цьому пристрої можна впорядкувати за швидкістю виконання завдання і цей порядок однаковий для всіх завдань: для кожного пристрою i існує коефіцієнт k_i такий, що тривалість виконання завдання j на пристрої i дорівнює $k_i p_j$. «Еталонним» будемо називати пристрій з коефіцієнтом продуктивності $k = 1$. В цьому сенсі величина p_j є тривалістю виконання завдання j на еталонному пристрої. Величину k_i будемо називати *коефіцієнтом продуктивності* (якщо $k_i > 1$,то пристрій i менш продуктивний, ніж еталонний, якщо $k_i < 1$ – більш продуктивний).

Передбачається, що всі завдання множини *J* надходять одночасно та мають спільний жорсткий директивний термін *d*, процес обслуговування кожного завдання протікає без переривань до завершення обслуговування завдань. Всі пристрої працюють без переривань. Необхідно знайти максимальний момент запуску пристроїв $r_{\max}$, що дозволяє отримати допустимий розв'язок (розклад, у якому усі завдання не запізнюються) Під моментом запуску розуміється момент початку виконання множини завдань (мінімальний з моментів початку виконання завдань з множини J).

Далі, не втрачаючи загальності, будемо вважати, що виконуються такі умови: еталонним є найбільш продуктивний пристрій і у нього $k_i = 1$ (звідси витікає, що усі $k_i \geq 1$); усі p_j є цілими числами.

Задача, що розглядається належить до класу NP.

3. Дослідження задачі

Визначимо теоретично мінімальній час, за який усі пристрої могли б виконати усі завдання в об'ємі $\sum_{j=1}^{n} p_j$. Цю величину можна отримати, якщо вважати, що усі пристрої можуть паралельно виконувати кожне з завдань. Для цього m паралельних пристроїв розглядаються як один, а фактичні тривалості завдань замінюються їх так званими узагальненими значеннями, рівними:

$$\widetilde{p}_j = \frac{1}{\sum_{i=1}^{m} \frac{1}{t_{ij}}},$$

де $t_{ij} = k_i p_j$ – тривалість виконання завдання j пристроєм i ($i = \overline{1,m},\ j = \overline{1,n}$);

$\frac{1}{t_{ij}}$ – частка завдання j, яка виконується пристроєм i за одну одиницю часу;

$\sum_{i=1}^{m} \frac{1}{t_{ij}}$ – частка завдання j, яка виконуються всіма пристроями за одну одиницю часу;

$\frac{1}{\sum_{i=1}^{m} \frac{1}{t_{ij}}}$ – тривалість виконання завдання j всіма пристроями.

Тоді, мінімально можливий час, за який усі пристрої могли б виконати усі завдання, становить:

$$C^* = \sum_{j=1}^{m} \widetilde{p}_j\,, \tag{1}$$

$$C^* = \sum_{j=1}^{n} \frac{1}{\sum_{i=1}^{m} \frac{1}{k_i p_j}}. \tag{2}$$

Підійдемо до визначення величини C^* з іншого боку. Як і раніше, визначимо C^* – як теоретично мінімально можливий час, за який усі пристрої могли б виконати усі завдання в об'ємі $\sum_{j=1}^{n} p_j$, але тепер за умови, що кожне завдання виконується тільки одним пристроєм. В ідеальному випадку розклад є *рівномірним* – у ньому усі пристрої обслуговують усі свої завдання за час C^*. Очевидно, що рівномірний розклад є оптимальним. Це пояснюється наступним. Переміщення будь-якого завдання з одного пристрою на інший збільшує час завершення усіх завдань на пристрої-реципієнті. Будь-який обмін груп завдань з двох різних пристроїв також збільшує час завершення усіх завдань на одному з пристроїв.

Введемо величину $c_i^* = \frac{C^*}{k_i}$ – «ідеальну» зведену тривалість зайнятості пристрою i ($i = \overline{1,m}$). У разі рівномірного розкладу справедливо наступне:

$$\sum_{j=1}^{n} p_j = \sum_{i=1}^{m} c_i^*\,, \tag{3}$$

$$\sum_{j=1}^{n} p_j = \sum_{i=1}^{m} \frac{C^*}{k_i}\,, \tag{4}$$

$$\sum_{j=1}^{n} p_j = C^* \sum_{i=1}^{m} \frac{1}{k_i}$$

(сумарний час виконання усіх завдань в еталонних значеннях тривалостей дорівнює сумі ідеальних зведених тривалостей зайнятості пристроїв).

Звідси отримаємо значення C^*:

$$C^* = \frac{\sum_{j=1}^{n} p_j}{\sum_{i=1}^{m} \frac{1}{k_i}}, \qquad C^* = \sum_{j=1}^{n} \frac{p_j}{\sum_{i=1}^{m} \frac{1}{k_i}}.$$

Поділимо чисельник та знаменник j-го доданку на величину p_j ($j = \overline{1,n}$):

$$C^* = \sum_{j=1}^{n} \frac{1}{\sum_{i=1}^{m} \frac{1}{k_i p_j}}. \tag{5}$$

Вираз (5) співпадає з отриманим раніше виразом (2). За умови ідентичності пристроїв ($k_i = 1,\ i = \overline{1,m}$) вираз для величини C^* матиме вигляд $C^* = \frac{\sum_{j=1}^{n} p_j}{m}$ (в цьому випадку результати розрахунків C^* співпадають із результатами, отриманими при дослідженні задачі з ідентичними пристроями [1]).

Величина C^* може бути як цілим так і не цілим числом.

Алгоритм побудови початкового розкладу

Мета цього евристичного алгоритму – рівномірно навантажити пристрої з урахуванням того, що їх продуктивності різні: поточне завдання призначається на той пристрій, у якого невикористаний зведений фонд робочого часу найбільший.

КРОК 1. Перенумерувати завдання множини J за незростанням тривалостей їх виконання на еталонному пристрої.

КРОК 2. Розрахувати C^* (теоретично мінімальній час виконання пристроями усіх завдань) та $c_i^* = \frac{C^*}{k_i}$ ($i = \overline{1,m}$) – «ідеальні» зведені тривалості зайнятості пристроїв.

КРОК 3. Встановити значення величин невикористаного зведеного фонду робочого часу пристроїв: $f_i = c_i^*$, $i=\overline{1,m}$.

КРОК 4. Встановити (реальний) час зайнятості пристроїв: $C_i = 0$, $i=\overline{1,m}$.

КРОК 5. Обрати завдання $j=1$.

КРОК 6. Обрати пристрій i з максимальним невикористаним зведеним фондом робочого часу f_i (а якщо таких декілька, то обрати пристрій з найменшим значенням k_i).

КРОК 7. Призначити завдання j на пристрій i: $C_i = C_i + k_i p_j$; $f_i = f_i - p_j$.

КРОК 8. Перейти до наступного завдання: $j = j+1$. Якщо $j > n$, кінець алгоритму, в іншому випадку переходимо до **КРОКУ 6**.

Відмітимо, що наведений алгоритм за умови ідентичності пристроїв (тобто $k_i = 1$, $i=\overline{1,m}$) дає той же самий результат, що і алгоритм побудови початкового розкладу для задачі з ідентичними пристроями [1].

Розглянемо деякий допустимий розклад σ. Позначимо в цьому розкладі:

$J_i(\sigma)$ – множина завдань, що виконується пристроєм i;

$C_i(\sigma) = \sum\limits_{j\in J_i(\sigma)} k_i p_j$ – тривалість зайнятості пристрою i;

$\Delta_i(\sigma) = \max\{0;\, C_i(\sigma) - C^*\} = \max\left\{0;\, \sum\limits_{j\in J_i(\sigma)} k_i p_j - C^*\right\}$ – виступ пристрою i;

$R_i(\sigma) = \max\{0;\, C^* - C_i(\sigma)\} = \max\left\{0;\, C^* - \sum\limits_{j\in J_i(\sigma)} k_i p_j\right\}$ – резерв пристрою i ($i=\overline{1,m}$).

Позначимо через $C_i^/(\sigma) = \dfrac{C_i(\sigma)}{k_i}$ зведену тривалість зайнятості пристрою i ($i=\overline{1,m}$) в розкладі σ – час зайнятості пристрою, розрахований у еталонних тривалостях виконання завдань.

Відповідні зведені величини:

$\Delta_i^/(\sigma) = \max\{0; C_i^/(\sigma) - c_i^*\}$; $i=\overline{1,m}$;

$R_i^/(\sigma) = \max\{0; c_i^* - C_i^/(\sigma)\}$, $i=\overline{1,m}$.

Визначимо:

$I_\Delta(\sigma)$ – множина пристроїв, для яких $\Delta_i(\sigma) > 0$;

$I_R(\sigma)$ – множина пристроїв, для яких $R_i(\sigma) > 0$;

$I_0(\sigma)$ – множина пристроїв, для яких $\Delta_i(\sigma) = R_i(\sigma) = 0$.

З урахуванням того, що величина d є константою (оскільки не залежить від розкладу) критерій оптимізації вихідної задачі (максимізація моменту r запуску пристроїв)

$$r = d - (C^* + \max_{1\le i\le m} \Delta_i(\sigma)) \to \max$$

є еквівалентним критерію мінімізації загального часу виконання усіх завдань:

$$C^* + \max_{1\le i\le m} \Delta_i(\sigma) \to \min .$$

В свою чергу, з урахуванням того, що $C^* = const$, критерій спрощується і зводиться до мінімізації максимального виступу:

$$\max_{1\le i\le m} \Delta_i(\sigma) \to \min .$$

Рисунок 1 ілюструє взаємозв’язок величин, що входять до вказаних критеріїв.

Рис. 1. Взаємозв’язок величин

Для введених величин має місце наступне:

$$\sum_{j\in J_i(\sigma)} k_i p_j = C^* - R_i(\sigma) + \Delta_i(\sigma),\ i=\overline{1,m}. \qquad (6)$$

Розділимо i –те рівняння в (6) на k_i ($k_i \ne 0$):

$$\sum_{j\in J_i(\sigma)} p_j = \frac{C^*}{k_i} - \frac{R_i(\sigma)}{k_i} + \frac{\Delta_i(\sigma)}{k_i},\ i=\overline{1,m},$$

$$\sum_{j\in J_i(\sigma)} p_j = c_i^* - R_i^/(\sigma) + \Delta_i^/(\sigma),\ i=\overline{1,m}.$$

Застосовуючи методологію побудови ПДС-алгоритмів [2], згідно з якою поліноміальна складова алгоритму породжується логіко-аналітичними умовами, виконання яких гарантує оптимальність отриманого рішення, визначимо ознаки оптимальності розкладів.

4. Ознаки оптимальності розкладів

Як було вказано раніше, «ідеальним» розкладом є рівномірний розклад, тобто розклад, у якого $\sum\limits_{i=1}^{m} \Delta_i(\sigma) = \sum\limits_{i=1}^{m} R_i(\sigma) = 0$. Рівномірний розклад можливо отримати тільки тоді, коли C^* –

ціле і усі $c_i^* = \dfrac{C^*}{k_i}$ також є цілими числами (інакше не можливі рівності $\sum_{j \in J_i} p_j = c_i^*$, $i = \overline{1, m}$ – ліві частини рівностей за припущенням є сумами цілих величин).

Отже, ми визначили таку ознаку оптимальності розкладу (достатню умову оптимальності): якщо в розкладі $\sum_{j \in J_i} p_j = c_i^*$,

($\sum_{i=1}^{m} \Delta_i^{/}(\sigma) = \sum_{i=1}^{m} R_i^{/}(\sigma) = 0$), $i = \overline{1, m}$, то поточний розклад є оптимальним.

Якщо виконується умова:

$$\exists i \mid c_i^* \notin Z. \tag{7}$$

(серед величин c_i^* є не цілі), то у цьому випадку принципово неможливо побудувати рівномірний розклад. Визначимо ознаку оптимальності для цього випадку.

Позначимо:

$$\delta = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \left\lfloor \frac{C^*}{k_i} \right\rfloor = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \lfloor c_i^* \rfloor.$$

Враховуючи (3) маємо, що $\delta \ge 0$, ціле.

За умови (7) маємо, що $\delta > 0$, ціле. Визначимо, який вигляд матиме «ідеальний» розклад у цьому випадку. Іншими словами, визначимо *контур* оптимального розкладу.

Визначення. Розклад, у якому на пристрої призначені не усі завдання із множини J, будемо називати неповним.

Представимо, що ми розподілили між пристроями деяку кількість завдань (нехай вони складають множину $\overline{J} \subset J$) і отримали неповний розклад $\overline{\sigma}$, у якому:

$$\sum_{j \in \overline{J}_i} t_{ij} = k_i \lfloor c_i^* \rfloor, \; i = \overline{1, m}; \tag{8}$$

або

$$\sum_{j \in \overline{J}_i} p_j = \lfloor c_i^* \rfloor, \; i = \overline{1, m}, \tag{9}$$

тут $\overline{J_i}$ – множина завдань, які в отриманому неповному розкладі $\overline{\sigma}$ виконуються пристроєм i). Відмітимо, що залишились недорозподіленими завдання, сумарна тривалість виконання яких становить:

$$\delta = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \sum_{j \in \overline{J}_i} p_j. \tag{10}$$

З урахуванням (9), рівняння (10) має вигляд:

$$\delta = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \lfloor c_i^* \rfloor. \tag{11}$$

Припустимо, що залишилось розподілити δ завдань, еталонна тривалість кожного з яких дорівнює 1.

Отже, перед нами постає така задача: розподілити одиничні завдання в кількості δ між пристроями так, щоб максимальний з виступів повного розкладу був мінімальним. В результаті ми отримаємо розклад, у якому робота в об’ємі $\sum_{j=1}^{n} p_j$ розподілена найкращим чином (ідеальний розклад для сукупності завдань з заданою сумою тривалостей їх виконання).

Позначимо:

$$e_i = c_i^* - \sum_{j \in \overline{J}_i} p_j, \; i = \overline{1, m}; \tag{12}$$

де e_i – резерв пристрою i у неповному розкладі $\overline{\sigma}$, в якому недорозподілено δ одиничних завдань.

З урахуванням (9) маємо:

$$e_i = c_i^* - \lfloor c_i^* \rfloor, \; i = \overline{1, m}. \tag{13}$$

З (13) слідує, що $\forall \; e_i \ge 0$.

На рисунку 2 відображено контур найкращого неповного розкладу $\overline{\sigma}$, у якому на пристрої призначені завдання в об’ємі $\sum_{i=1}^{m} \lfloor c_i^* \rfloor$ (в еталонних тривалостях).

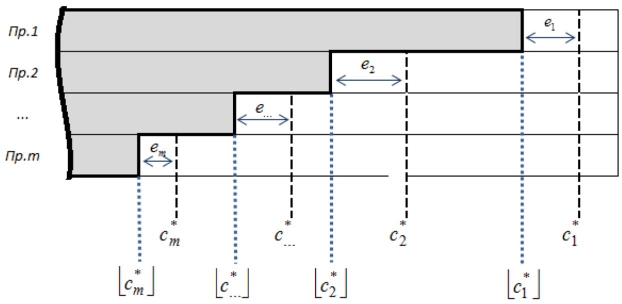

Рис. 2. Контур найкращого розкладу, у якому на пристрої призначені завдання в об’ємі $\sum_{i=1}^{m} \lfloor c_i^* \rfloor$ ***(в еталонних тривалостях)***

Просумуємо рівняння (13) по i:

$$\sum_{i=1}^{m} e_i = \sum_{i=1}^{m} c_i^* - \sum_{i=1}^{m} \lfloor c_i^* \rfloor,$$

з урахування (3) маємо:

$$\sum_{i=1}^{m} e_i = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \lfloor c_i^* \rfloor = \delta.$$

Таким чином, можна сформулювати наступну ***допоміжну оптимізаційну задачу***: необхідно δ завдань одиничної довжини розподілити між m пристроями, за умов, що пристрій i має резерв e_i, $i=1,m$ і $\sum_{i=1}^{m} e_i = \delta$ з метою мінімізації максимального з виступів (з урахуванням продуктивностей пристроїв). Математична модель цієї задачі має наступний вигляд.

Змінні: x_i – кількість «одиничних» еталонних завдань, що повинні бути призначені на пристрій i, $i=\overline{1,m}$.

Обмеження – кількість «одиничних» еталонних завдань становить δ :

$$\sum_{i=1}^{m} x_i = \delta;\ \ x_i \ge 0,\ \ i=1,m \text{, цілі.} \qquad (14)$$

Цільова функція: мінімізація максимального з виступів (з урахуванням продуктивностей пристроїв):

$$\max_i \{k_i(x_i - e_i)\} \to \min . \qquad (15)$$

Алгоритм розв'язання допоміжної оптимізаційної задачі

Дано: δ; e_i, $i=1,m$ ($\sum_{i=1}^{m} e_i = \delta$); k_i, $i=1,m$; $T_i = C^* - e_i k_i$ (тривалість виконання завдань пристроєм i в неповному ідеальному розкладі $\overline{\sigma}$), $i=1,m$.

КРОК 1 Ініціалізація**:** $x_i = 0$, $i=1,m$.

КРОК 2 Поки $\delta \ge 1$ (поки є нерозподілені одиничні завдання)

2.1 Знайти

$$\min\{T_1 + k_1; T_2 + k_2; ....; T_m + k_m\}, \qquad (16)$$

($T_i + k_i$ – тривалість виконання усіх завдань пристроєм i за умови, що на цей пристрій буде призначено поточне одиничне завдання). Нехай мінімум в (16) відповідає пристрою $i = q$.

2.2 Призначити на пристрій q одиничне завдання: $x_q = x_q + 1$, $\delta = \delta - 1$.

2.3 Перерахувати тривалість виконання усіх завдань пристроєм q: $T_q = T_q + k_q$.

КІНЕЦЬ АЛГОРИТМУ

Отже, отримали такі ознаки оптимальності (достатні умови оптимальності).

Ознака оптимальності 1

Якщо $C^* = \sum_{j=1}^{n} \frac{1}{\sum_{i=1}^{m} \frac{1}{k_i p_j}}$ – ціле та усі $c_i^* = \frac{C^*}{k_i}$ також є цілими числами – для заданого набору завдань можливо побудувати рівномірний розклад.

Якщо у розкладі σ виконується:

$$C_i^{/}(\sigma) = \sum_{j \in J_i} p_j = c_i^*,\ i=\overline{1,m},$$

$$(C_i(\sigma) = k_i \sum_{j \in J_i} p_j = k_i c_i^* = C^*,\ i=\overline{1,m})$$

або

$\sum_{i=1}^{m} \Delta_i(\sigma) = \sum_{i=1}^{m} R_i(\sigma) = 0$, то це означає, що поточний розклад є рівномірним (оптимальним).

Ознака оптимальності 2

Якщо $\delta = \sum_{j=1}^{n} p_j - \sum_{i=1}^{m} \lfloor c_i^* \rfloor > 0$

($\exists i \mid c_i^* = \frac{C^*}{k_i} \notin Z$), то у цьому випадку отримати рівномірний розклад не можливо.

Нехай x_i^*, $i=\overline{1,m}$ – оптимальний розв'язок допоміжної оптимізаційної задачі (14)-(15), $T_i^* = C^* + k_i(x_i^* - e_i)$ – тривалість виконання завдань в «ідеальному» розкладі.

Тоді розклад σ, в якому:

$$C_i^{/}(\sigma) = \lfloor c_i^* \rfloor + x_i^*,\ i=\overline{1,m}$$

або

$C_i(\sigma) = T_i^*$, $i=\overline{1,m}$, є оптимальним.

На основі ознак оптимальності розкладів визначимо множину перестановок, яка дозволить послідовно покращувати значення критерію.

5. Розробка множини перестановок

Після розв'язання допоміжної задачі отримали контур оптимального розкладу для заданої сумарної довжини завдань множини J, у якого i-й пристрій завершує свої завдання у момент:

– C^*, якщо $\delta = 0$;
– T_i^*, якщо $\delta > 0$ (рис. 3).

Введемо позначення:

$$Z_i(\sigma) = \max\{0;\ C_i(\sigma) - T_i^*\} = \max\left\{0;\ \sum_{j \in J_i(\sigma)} k_i p_j - T_i^*\right\}$$

(аналог величини $\Delta_i(\sigma)$, але в даному випадку розраховується відхилення від «ідеальної» тривалості зайнятості пристрою T_i^* в оптимальному контурі розкладу);

$$E_i(\sigma) = \max\{0;\ T_i^* - C_i(\sigma)\} = \max\left\{0;\ T_i^* - \sum_{j \in J_i(\sigma)} k_i p_j\right\}$$

(аналог величини $R_i(\sigma)$;

$I_Z(\sigma)$ – множина таких пристроїв, для яких $Z_i(\sigma) > 0$;

$I_E(\sigma)$ – множина таких пристроїв, для яких $E_i(\sigma)>0$.

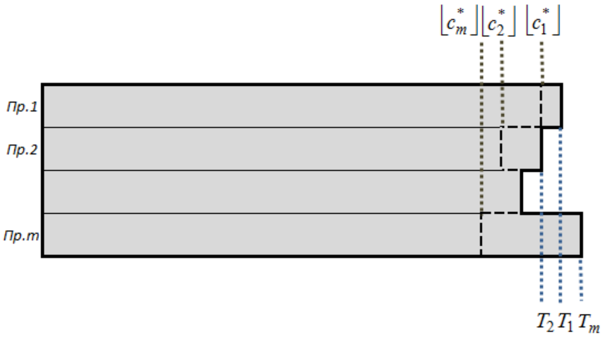

Рис. 3. Оптимальний за другою ознакою оптимальності контур розкладу

Для покращення розкладу необхідно направити зусилля на зменшення величини $\max_i Z_i(\sigma)$. Для цього пропонується використовувати обмін завдань між пристроями з множин $I_Z(\sigma)$ та $I_E(\sigma)$: коли деяка підмножина завдань з пристрою h (позначимо її як $K_h(\sigma)$, $K_h(\sigma)\subseteq J_h(\sigma)$) обмінюється з деякою підмножиною завдань з пристрою s (позначимо цю підмножину як $L_s(\sigma)$, $L_s(\sigma)\subseteq J_s(\sigma)$). Позначимо через θ різницю між сумами еталонних тривалостей завдань, які приймають участь в обміні:

$$\theta=\sum_{j\in K_h(\sigma)}p_j-\sum_{j\in L_s(\sigma)}p_j\ .$$

Далі такий обмін завданнями між підмножинами $J_h(\sigma)$ та $J_s(\sigma)$будемо називати перестановкою.

Нехай σ^1 – розклад, який отримано після виконання перестановки. Для того, щоб перестановка призвела до покращення (тобто $\sum_{i=1}^{m}Z_i(\sigma)$-$\sum_{i=1}^{m}Z_i(\sigma^1)$>0), необхідно, щоб $\theta>0$.

В роботі розроблена множина перестановок, які в залежності від умов виконання та наслідків розділено на чотири типи: $\boldsymbol{A\,k}$, $\boldsymbol{Б\,k}$, $\boldsymbol{В\,k}$ та $\boldsymbol{Г\,k}$. Перестановки усіх типів, в свою чергу, можна розділити на підтипи в залежності від кількості завдань, які приймають участь в перестановці (потужностей множин $K_h(\sigma)$ та $L_s(\sigma)$).

Таблиця 1 містить узагальнену характеристику та властивості перестановок усіх типів. Ці перестановки покладені в основу розробленої поліноміальної складової ПДС-алгоритму розв'язання задачі.

6. ПДС-алгоритм розв'язання задачі

На основі ознак оптимальності та розробленої множини перестановок побудований алгоритм визначення максимально пізнього моменту початку виконання завдань паралельними пристроями

Табл. 1. Властивості перестановок

Тип перестановки	*Умови виконання перестановки*	*Характеристики результуючого розкладу* σ^1
А k	$\theta>0$, $\theta k_h\le Z_h(\sigma)$, $\theta k_s\le E_s(\sigma)$	$Z_h(\sigma^1)=Z_h(\sigma)-\theta k_h$, $E_s(\sigma^1)=E_s(\sigma)-\theta k_s$
Б k	$\theta>0$, $\theta k_s\le Z_h(\sigma)$, $\theta k_h> E_s(\sigma)$	$Z_h(\sigma^1)=Z_h(\sigma)-\theta k_s$, $Z_s(\sigma^1)=\theta k_h-E_s(\sigma)$
В k	$\theta>0$, $\theta k_s> Z_h(\sigma)$, $\theta k_h\ge E_s(\sigma)$	$Z_h(\sigma^1)=\theta k_s-Z_h(\sigma)$, $Z_s(\sigma^1)=\theta k_h-E_s(\sigma)$
Г k	$\theta>0$, $\theta k_s< Z_h(\sigma)$, $\theta k_h< Z_h(\sigma)-Z_s(\sigma)$	$Z_h(\sigma^1)=Z_h(\sigma)-\theta k_s$, $Z_s(\sigma^1)=Z_s(\sigma)+\theta k_h$

різної продуктивності із загальним жорстким директивним терміном в допустимому розкладі.

Опис алгоритму

КРОК 1 Побудувати початковий розклад σ^0

КРОК 2 $\sigma=\sigma^0$. Визначити δ .

КРОК 3 *Перевірити виконання ознак оптимальності*

3.1 ЯКЩО $\delta=0$ і виконується перша ознака оптимальності

ТО перейти на **КРОК 6** (σ – оптимальний розклад)

ІНАКШЕ перейти на КРОК **4**

3.2 ЯКЩО $\delta>0$

ТО

3.2.1 Розв'язати допоміжну оптимізаційну задачу.

3.2.2. ЯКЩО виконується друга ознака оптимальності

ТО перейти на **КРОК 6** (σ – оптимальний розклад)

КРОК 4 Визначити пристрій h , якому відповідає максимальне значення виступу: $Z_h(\sigma)=\max\limits_{i\in I_Z(\sigma)} Z_i(\sigma)$.

КРОК 5 Для пристрою h перебираючи усі пристрої $s\in I_E(\sigma)$ виконати перестановку типу $\boldsymbol{A\,k}$, $\boldsymbol{Б\,k}$ або $\boldsymbol{В\,k}$,

ЯКЩО таких перестановок не знайшлось,

ТО

5.1 Для пристрою h перебираючи пристрої $s\in I_0(\sigma)\cup I_Z(\sigma)$ виконати перестановку типу $\boldsymbol{Г\,k}$ **.**

5.2 ЯКЩО таких перестановок не знайшлось,

ТО перейти на **КРОК 6**,

ІНАКШЕ $\sigma=\sigma^1$, перейти на **КРОК 3.**

ІНАКШЕ $\sigma=\sigma^1$, перейти на **КРОК 3.**

КРОК 6 Визначити максимально пізній момент запуску завдань на виконання: $r_{\max}=r(\sigma)=d-\max\limits_i C_i(\sigma)$.

КІНЕЦЬ АЛГОРИТМУ

Складність алгоритму складає $O(n^2W)$, де $W=\sum\limits_{i=1}^{n} p_i$. Це пояснюється наступним. На кожному кроці значення цільової функції (значення максимального з виступів) зменшується, щонайменше на 1. Значить, в гіршому випадку алгоритм зробить кількість кроків, рівне значенню суми виступів в початковому розкладі σ^0. Величина W являється дуже грубою верхньої оцінкою значення $\sum\limits_{i=1}^{m}\Delta_i(\sigma^0)$. На кожному кроці самою трудомісткою операцією є пошук допустимих перестановок завдань між двома пристроями, яка за умови $|K_h(\sigma)|=|L_s(\sigma)|=1$ зводиться до аналізу тривалостей всіх пар завдань двох обраних пристроїв, (верхня межа кількості пар завдань на пристрої дорівнює $n(n-1)/2$).

7. Висновок

Досліджені властивості задачі календарного планування виконання завдань із спільним жорстким директивним терміном паралельними пристроями різної продуктивності з метою максимізації моменту запуску пристроїв за умови, що усі завдання не запізнюються. Сформульована допоміжна оптимізаційна задача, за результатами якої визначені достатні умови оптимальності розкладів. Визначено множину перестановок, які дозволяють послідовно покращувати значення критерію. Розроблена поліноміальна складова ПДС-алгоритму розв'язання задачі.

Перелік посилань

1. Павлов О.А., Жданова О.Г., Сперкач М.О. Задача составления допустимого расписания с максимально поздним моментом запуска выполнения идентичными параллельными приборами работ с общим директивным сроком / О.А. Павлов, М.О. Сперкач, О.Г. Жданова // Вісник НТУУ "КПІ". Серія «Інформатика, управління та обчислювальна техніка». – К.: "ВЕК+", 2014. – №61 – С.93-102.
2. Згуровский, М. З. Принятие решений в сетевых системах с ограниченными ресурсами [Текст]: монография / М. З. Згуровский, А. А. Павлов. – К.: Наукова думка, 2010.– 573 с.