

## ЗАДАЧА СКЛАДАННЯ РОЗКЛАДУ ВИКОНАННЯ РОБІТ З УРАХУВАННЯМ ЇХНІХ ЧАСОВИХ ВІКОН

<sup>1</sup>Національний технічний університет України  
«Київський політехнічний інститут ім. Ігоря Сікорського»

*Розглянуто оптимізаційну задачу теорії розкладів, у якій машини є паралельними та ідентичними, а роботи, окрім тривалостей виконання, мають часові вікна, тобто часові інтервали, в межах яких робота може бути виконана, а поза межами яких — не може. Часове вікно роботи визначається моментом її надходження в систему, що в загальному випадку є ненульовим, та її директивним строком. Вважається, що всі часові вікна належать інтервалу від мінімального моменту надходження до максимального директивного строку, а кожна робота має одне і тільки одне часове вікно. Критеріями оптимізації є: максимізація кількості робіт, що виконуються, та максимізація сумарної тривалості робіт, що виконуються. Проведено аналіз подібних задач, як-от задачі планування завантаженості процесора та задачі упаковки. Указані приклади практичного застосування задачі, що розглядається. Запропоновано евристичний алгоритм розв'язання задачі, що полягає у розгляді робіт у певному порядку, пошуку для неї машини та визначення моменту початку виконання роботи. Запропоновано чотири варіації цього евристичного алгоритму, що базуються на евристичних правилах порядку розгляду робіт в залежності від параметрів робіт, таких, як тривалість виконання, ширина часового вікна, люфт (різниця ширини вікна та тривалості роботи), та відношення ширини вікна до тривалості роботи. Проведена серія експериментів, що дозволяє оцінити, яка з варіацій алгоритму є ефективною для розв'язання задачі для кожної з двох розглянутих критеріїв оптимізації. В ході проведення експериментів встановлено, що для критерію максимізації кількості робіт, що виконуються, ефективним правилом є впорядкування за шириною часового вікна, а для критерію максимізації сумарної тривалості кількості робіт, що виконуються — впорядкування за тривалістю виконання робіт.*

**Ключові слова:** теорія розкладів, паралельні машини, момент надходження, директивний строк, евристичний алгоритм.

### Вступ

Задача складання розкладу з урахуванням часових вікон належить до класу задач календарного планування. Календарне планування є процесом прийняття рішень, що має практичне застосування у багатьох галузях науки і техніки. Задачі календарного планування належать до задач теорії розкладів. Для знаходження розв'язків задач теорії розкладів використовуються як точні методи, так і наближені (евристичні) методи, що зводяться до визначення порядку призначення робіт на доступні ресурси та власне призначення робіт на машину.

*Предметом дослідження є дві задачі резервації з паралельними ідентичними машинами, у яких для кожної роботи відомі час надходження до системи та директивний строк, що формують часове вікно, та тривалість виконання роботи. Задачі відрізняються цільовими функціями (ЦФ): в першій з них максимізується кількість робіт, що виконуються, в другій — максимізується сумарна тривалість виконуваних робіт.*

*Метою дослідження є розробка алгоритму розв'язання поставлених задач та підбір параметрів, за яких досягається найбільша ефективність за умови знаходження розв'язків.*

Пінедо [1] розглядає узагальнений варіант поставленої задачі (з ЦФ максимізації зваженої кількості робіт, що виконуються) у двох варіаціях: коли ширина часового вікна дорівнює тривалості виконання роботи, та коли ширина часового вікна більша за тривалість виконання роботи. Обидві ці задачі в загальному випадку є NP-складними. Натомість алгоритми, що розглядаються у [1],

працюють лише з цілими числами у вхідних даних.

У [2]—[5] розглядаються схожі задачі з їхнім використанням у плануванні завантаженості процесорів; в ході планування вони враховують різницю між шириною часового вікна та тривалістю виконання роботи.

Також поставлена задача подібна до задач упаковки в тому сенсі, що слід якомога щільніше заповнити певний простір (у цьому випадку — часові осі, що відповідають машинам) певними об'єктами (у цьому випадку — роботами). У [6]—[9] розглядаються алгоритми одновимірної упаковки, загальна ідея яких полягає у максимізації заповнення блоків, що подібно постановці поставленої задачі. У [9]—[13] також розглядаються схожі задачі, але вони належать до інших класів задач пакування, а саме пакування з більшою кількістю вимірів (2 та більше).

На практиці ця задача може бути використана для:

– планування графіка певного заходу, в ході якого рівнозначні експерти мають обслуговувати людей, що прийшли до них на консультацію;

– планування графіка обслуговування пацієнтів у терапевтів певної поліклініки, причому кожний пацієнт може прийти лише у визначений ним період часу.

### Постановка задачі

Нехай маємо  $m$  паралельних ідентичних машин (позначаються  $M_i, i = \overline{1, m}$ ) та  $n$  робіт (позначаються  $J_j, j = \overline{1, n}$ ). Для кожної роботи відома тривалість її виконання  $p_j$  та границі часового вікна:

– момент надходження роботи у систему (момент часу, раніше за який виконання роботи не може розпочатися)  $r_j$ ;

– директивний строк (момент часу, пізніше за який не може бути завершено виконання роботи)  $d_j$ .

Водночас момент надходження та директивний строк кожної роботи обмежені мінімальним моментом надходження  $R_{\min}$  та максимальним директивним строком  $D_{\max}$  відповідно, так щоби часове вікно кожної роботи входило в інтервал  $[R_{\min}, D_{\max}]$ .

Величина  $U_j$  позначає, чи виконується робота  $J_j$ .  $U_j$  набудатиме значення 1, якщо робота  $J_j$  виконується, і набудатиме значення 0, якщо робота  $J_j$  не виконується.

Розглядаються задачі з такими ЦФ:

– максимізація кількості робіт, що виконуються  $\left( \sum_{j=1}^n U_j \right)$ ;

– максимізація сумарної тривалості виконання робіт, що виконуються  $\left( \sum_{j=1}^n p_j U_j \right)$ .

За нотацією Грехема, задача з першою ЦФ може бути записана як  $Pm|d_j, r_j| \sum_{j=1}^n U_j$ , а задача з другою ЦФ — як  $Pm|d_j, r_j| \sum_{j=1}^n p_j U_j$ .

### Евристичний алгоритм розв'язання задачі

Пропонується евристичний алгоритм для розв'язання задачі з будь-якою із запропонованих ЦФ. У цьому алгоритмі для кожної невиконаної роботи за певними правилами обчислюється її пріоритет. Ці пріоритети обчислюються на основі так званого показника гнучкості, згідно з яким вищий пріоритет надається роботам з вузьким часовим вікном та більшою тривалістю виконання.

Запропоновано такі евристичні правила обчислення показників гнучкості:

– на основі тривалості (дорівнює тривалості виконання роботи, помноженій на  $-1$ , тобто  $-p_j$ ), позначатиметься  $P$ ;

– на основі ширини вікна (дорівнює ширині часового вікна, тобто  $d_j - r_j$ ), позначатиметься  $W$ ;

– абсолютна гнучкість (дорівнює різниці ширини вікна та тривалості роботи, тобто ширина люфту, тобто  $d_j - r_j - p_j$ ), позначатиметься  $A$ ;

– відносна гнучкість (дорівнює відношенню ширини вікна до тривалості роботи, тобто  $\frac{d_j - r_j}{p_j}$ ), позначатиметься  $R$ .

Таким чином, алгоритм матиме параметр — евристичне правило обчислення показника гнучкості роботи. В ході експериментів досліджується вплив цього правила на ефективність алгоритму.

*Евристичний алгоритм розв'язання задачі* такий:

0. Обчислити показники гнучкості для кожної роботи.
1. Впорядкувати роботи за неспаданням значень показників гнучкості.
2. ЦИКЛ за роботами:
  - 2.1. Визначити порядок розгляду машин випадковим чином.
  - 2.2. ЦИКЛ за машинами (у порядку, визначеному на кроці 2.1):
    - 2.2.1. Визначити множину можливих моментів початку виконання роботи в межах часового вікна (з урахуванням того, що на інтервалі від моменту початку виконання до моменту завершення виконання роботи машина не має виконувати жодну іншу роботу).
    - 2.2.2. ЯКЩО множина порожня, ТО перейти на наступну ітерацію циклу за машинами.
    - 2.2.3. ІНАКШЕ:
      - 2.2.3.1. Згенерувати момент початку виконання роботи випадковим чином з елементів множини, визначеної на кроці 2.2.1.
      - 2.2.3.2. Призначити поточну роботу поточній машині з моментом початку виконання роботи, визначеним на кроці 2.2.3.1.
      - 2.2.3.3. ЯКЩО поточна робота призначена машині, ТО вийти з циклу за машинами.
  - 2.3. ЯКЩО поточна робота не призначена жодній машині, ТО відхилити роботу.

До переваг запропонованого алгоритму перед тим, що розглядається у [1], належить його випадковість; багатократний запуск алгоритму може покращити результати його роботи. Іншою його перевагою є можливість працювати з нецілими вхідними параметрами робіт. До недоліків можна віднести те, що в ході пошуку часового інтервалу для призначення роботи запропонований алгоритм не враховує, на який часовий інтервал претендує менше робіт.

### План експериментів

В ході експериментів генеруються задачі з однаковою кількістю машин, але різною кількістю робіт; ці задачі розв'язуються запропонованим алгоритмом з різними параметрами.

Загальний план експерименту виглядає таким чином:

1. **Для кожної** кількості робіт  $n \in \{100; 101; \dots; 1000\}$  :
  - 1.1. **Згенерувати** 10 індивідуальних задач з  $m = 4$ , визначеним  $n$ , нормально розподіленими тривалостями виконання робіт, бімодально розподіленими моментами надходження робіт та рівномірно розподіленими відношеннями ширини часових вікон до тривалостей виконання робіт.
    - 1.1.1. **Для кожної** індивідуальної задачі:
      - 1.1.1.1. **Для кожного** евристичного правила із  $\{P, W, A, R\}$  :
        - 1.1.1.1.1. **Розв'язати** задачу з заданим показником гнучкості по 10 разів.
        - 1.1.1.1.2. **Запам'ятати** найкраще з отриманих значень кожної ЦФ окремо.

2. **Обробка** отриманих результатів.

В пункті 1.1.1.1.2 для кожної індивідуальної задачі та кожної ЦФ реєструються показники ефективності:

- евристичне правило, що знайдено найкраще із зареєстрованих значень ЦФ;
- відносне відхилення від найкращого із зареєстрованих значень ЦФ для кожного з алгоритмів.

На основі кількості індивідуальних задач, в яких евристичне правило знайшло найкраще значення ЦФ, та середнього відносного відхилення від найкращого значення ЦФ, приймається висновок стосовно порівняння ефективності алгоритму за умови використання різних евристичних правил. Найкращим правилом визнається те, яке знайшло найкращий розв'язок для найбільшої кількості задач та має найменше середнє відносне відхилення від найкращого значення ЦФ.

### Результати експериментів

В ході експериментів для генерації параметрів робіт в індивідуальних задачах використовувалися такі величини:

- мінімальний момент надходження 0;
- максимальний директивний строк 480;
- усічено-нормально розподілена тривалість виконання роботи з математичним сподіванням 10 та середньоквадратичним відхиленням 1;
- неперервно рівномірно розподілене відношення ширини часового вікна до тривалості виконання роботи на відрізку [1,01; 5];
- бімодально розподілений момент надходження робіт (з рівною ймовірністю генерується усічено-нормально розподілена величина з математичним сподіванням 144 або 336, середньоквадратичне відхилення 192).

Крім цього, застосовуються такі валідації вхідних даних:

- тривалості виконання всіх робіт повинні бути більші за 1;
- часові вікна всіх робіт повинні належати інтервалу [0; 480].

В ході експерименту для задачі з ЦФ максимізації кількості виконуваних робіт, отримані результати, подані в табл. 1. Для цієї задачі алгоритм показує кращі результати з евристичним правилом W (впорядкування за шириною часового вікна).

Таблиця 1

#### Показники ефективності для ЦФ максимізації кількості виконуваних робіт

Евристичне правило	Кількість індивідуальних задач, в яких евристичне правило знайшло найкраще значення ЦФ	Середнє відносне відхилення від найкращого значення ЦФ, %
P	10 (0,11 %)	16,85
W	7700 (85,64 %)	0,18
A	1404 (15,58 %)	2,28
R	995 (11,04 %)	4,04

В ході експерименту для задачі з ЦФ максимізації сумарної тривалості виконання робіт, отримані результати, подані в табл. 2. Для цієї задачі алгоритм показує кращі результати з евристичним правилом P (впорядкування за тривалістю виконання роботи).

Таблиця 2

#### Показники ефективності для ЦФ максимізації сумарної тривалості виконання робіт

Евристичне правило	Кількість індивідуальних задач, в яких евристичне правило знайшло найкраще значення ЦФ	Середнє відносне відхилення від найкращого значення ЦФ, %
P	8241 (91,47 %)	0,14
W	117 (1,30 %)	6,69
A	126 (1,40 %)	5,49
R	657 (7,29 %)	3,83

### Висновки

В ході роботи розглянуто задачі теорії розкладів з паралельними ідентичними машинами, ненульовими моментами надходження та директивними строками:  $Pm[d_j, r_j] \sum_{j=1}^n U_j$  та  $Pm[d_j, r_j] \sum_{j=1}^n p_j U_j$ . Запропоновано евристичний алгоритм її розв'язання та чотири евристичні правила порядку розгляду робіт. Проведено серію експериментів для визначення ефективного евристичного правила. У випадку ЦФ  $\sum_{j=1}^n U_j$ , найефективнішим є евристичне правило впорядкування за шириною часового вікна, у випадку ЦФ  $\sum_{j=1}^n p_j U_j$  — евристичне правило впорядкування за тривалістю виконання роботи.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] M. L. Pinedo, *Planning and Scheduling in Manufacturing and Services*, Second Edition. New York, USA: Springer Science+Business Media, 2009, pp. 209-215.
- [2] B. B. Brandenburg, and M. Gül, "Global Scheduling Not Required: Simple, Near-Optimal Multiprocessor Real-Time

Scheduling with Semi-Partitioned Reservations,” in *2016 IEEE Real-Time Systems Symposium (RTSS)*, Porto, Portugal, 2016, pp. 99-110. <https://doi.org/10.1109/RTSS.2016.019>.

[3] L. Bo, and S. Bin, “Slack-based advance reservation for grid jobs,” in *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, Chengdu, China, 2010, pp. V3418-V3421. <https://doi.org/10.1109/ICACTE.2010.5579599>.

[4] C. Hu, J. Huai, and T. Wo, “Flexible resource reservation using slack time for service grid,” in *12th International Conference on Parallel and Distributed Systems (ICPADS'06)*, Minneapolis, MN, USA, 2006, pp. 327-324. <https://doi.org/10.1109/ICPADS.2006.49>.

[5] B. Li, Y. Li, M. He, H. Wu, and J. Yang, “Scheduling of a Relaxed Backfill Strategy with Multiple Reservations,” in *2010 International Conference on Parallel and Distributed Computing, Applications and Technologies*, Wuhan, China, 2010, pp. 311-316. <https://doi.org/10.1109/PDCAT.2010.24>.

[6] N. P. Rodrigo, W. B. Daundasekera, and A. I. Perera, “One-Dimensional Bin-Packing Problems with Branch and Bound Algorithm,” in *International Journal of Discrete Mathematics*, vol. 3, no. 2, pp. 36-40, 2018. New York, USA.

[7] K. Fleszar, and K. S. Hindi, “New heuristics for one-dimensional bin-packing,” *Computers & Operations Research*, vol. 29, no. 7, pp. 821-839, 2002, Uxbridge, United Kingdom. [https://doi.org/10.1016/S0305-0548\(00\)00082-4](https://doi.org/10.1016/S0305-0548(00)00082-4).

[8] A. C. F. Alvim, C. C. Ribeiro, F. Glover, and D. J. Aloise, “A Hybrid Improvement Heuristic for the One-Dimensional Bin Packing Problem,” *Journal of Heuristics*, vol. 10, no. 2, pp. 205-229, 2004. <https://doi.org/10.1023/B:HEUR.0000026267.44673.ed>

[9] E. G. Coffman Jr., M. R. Garey, and D. S. Johnson, “Bin packing with divisible item sizes,” *Journal of Complexity*, vol. 3, no. 4, pp. 406-428, 1987. [https://doi.org/10.1016/0885-064X\(87\)90009-4](https://doi.org/10.1016/0885-064X(87)90009-4).

[10] J. O. Berkey, and P. Y. Wang, “Two-Dimensional Finite Bin-Packing Algorithms,” *Journal of the Operational Research Society*, vol. 38, no. 5, pp. 423-429, 1987. <https://doi.org/10.1057/jors.1987.70>.

[11] A. Lodi, S. Martello, and D. Vigo, “Recent advances on two-dimensional bin packing problems,” *Discrete Applied Mathematics*, vol. 123, pp. 379-396, 2002. [https://doi.org/10.1016/S0166-218X\(01\)00347-X](https://doi.org/10.1016/S0166-218X(01)00347-X).

[12] J.-K. Kim, H. Lee-Kwang, and S. W. Yoo, “Fuzzy bin packing problem,” *Fuzzy Sets and Systems*, vol. 120, no. 3, pp. 429-434, 2001. [https://doi.org/10.1016/S0165-0114\(99\)00073-1](https://doi.org/10.1016/S0165-0114(99)00073-1).

[13] N. Bansal, J. Correa, C. Kenyon, and M. Sviridenko, “Bin Packing in Multiple Dimensions: Inapproximability Results and Approximation Schemes,” *Mathematics of Operations Research*, vol. 31, no. 1, pp. 31-49, 2006. <https://doi.org/10.1287/moor.1050.0168>.

Рекомендована кафедрою комп'ютерних систем управління ВНТУ

Стаття надійшла до редакції 19.04.2023

**Жданова Олена Григорівна** — канд. техн. наук, доцент, доцент кафедри інформаційних систем та технологій, e-mail: zhdanova.elena@hotmail.com ;

**Коваленко Владислав Вадимович** — студент факультету інформатики та обчислювальної техніки.

Національний технічний університет «Київський політехнічний інститут», Київ

**O. H. Zhdanova<sup>1</sup>**  
**V. V. Kovalenko<sup>1</sup>**

## Problem of Scheduling Jobs Considering Time Windows

<sup>1</sup>National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”

The article is devoted to the scheduling optimization problem in which the machines are parallel and identical, and jobs, apart from their processing times, also have time windows, i.e. time intervals within the limits of which the job can be performed, and outside of which the job cannot be performed. The job's time window is determined by its release date, which is non-zero in general case, and its and due date. It is considered that all time windows belong to the interval between the minimum release date and the maximum due date, and that each job has one and only one time window. The optimization criteria are: maximization of the number of jobs that are performed, and maximization of the total duration time of jobs that are performed. The analysis of the similar problems, such as problems of processor's computational load planning, and bin-packing problems, was conducted. The examples of practical usage of the considered problem were provided. The heuristic algorithm for solving the problem was proposed, and it is comprised of consideration of the job in a certain order, and search of the machine and starting time for this job. Four variations of this heuristic algorithm that are based on heuristic rules of job consideration order depending on jobs' parameters are proposed, such as processing time, time window width, slack (difference between time window width and processing time), and ratio of time window width to processing time. A series of experiments was conducted to evaluate which algorithm variation is effective for solving the problem for each of two suggested optimization criteria. In the process of experiments, it was determined that for the criteria of maximization of the number of jobs that are performed, the effective rule is based on time window width, and for the criteria of maximization of the total duration time of jobs that are performed, the effective rule is based on the duration of the job realization.

**Keywords:** scheduling, parallel machines, release date, due date, heuristic algorithm.

**Zhdanova Olena H.** — Cand. Sc. (Eng.), Associate Professor of the Chair of Information Systems and Technologies, e-mail: zhdanova.elena@hotmail.com ;

**Kovalenko Vladyslav V.** — Student of the Department of Informatics and Computer Engineering