

**Павлов В.Г.**

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

## ЗАСТОСУВАННЯ ДІАГРАМ ГАНТА ДЛЯ ІЛЮСТРАЦІЇ ПРОЦЕСУ ДІНАМІЧНОГО БАЛАНСУВАННЯ НАВАНТАЖЕННЯ ОБЧИСЛЮВАЛЬНИХ ПРИСТРОЇВ

*У статті проаналізовані математичні та графічні засоби ілюстрації статичного та динамічного балансування навантаження обчислювальних пристроїв. Традиційні способи опису не надають потрібної інформативності та наочності для ілюстрації методів і алгоритмів динамічного балансування. Особливо це необхідно у випадках, коли при виконанні балансування можуть змінюватися деякі показники. Наприклад, кількість завдань, їх складність, час додавання нових завдань, кількість обчислювальних пристроїв тощо.*

*Стрічкові діаграми, або діаграми Ганта раніше розглядалися лише як зручний і наочний засіб для відображення процесу проектного планування. Автор пропонує застосування діаграми Ганта в якості універсального засобу ілюстрації процесу динамічного балансування. Це дозволяє отримати зрозуміле графічне відображення процесу динамічного балансування. В якості показника завантаженості обчислювального пристрою застосовуються його часові характеристики; розглянути приклади побудови діаграм Ганта для відображення процесу динамічного балансування навантаження при різних випадках формування послідовності завдань, в тому числі і декількома частками з затримкою в часі.*

*Отримані діаграми дозволили провести аналіз та зрозуміти, з яких складників формується загальний час виконання всіх завдань. Мінімальне значення цього параметру є метою динамічного балансування. Діаграми також наочно надають інформацію про завантаженість кожного з обчислювальних пристроїв. Також визначені перспективи подальшого вдосконалення.*

**Ключові слова:** розподілені обчислення, балансування навантаження, динамічне балансування, стрічкова діаграма, діаграма Ганта.

**Постановка проблеми.** Обробка величезних обсягів даних, яка супроводжує вирішення багатьох задач у системах штучного інтелекту, вимагає різкого підвищення ефективності обчислень. Динамічне балансування навантаження дозволяє оптимально використовувати наявні обчислювальні пристрої. Традиційні способи опису та ілюстрації методів і алгоритмів динамічного балансування, на жаль, не надають потрібної інформативності та наочності.

Особливо це стосується випадків, коли при виконанні балансування можуть змінюватися такі показники як кількість завдань, їх складність, час додавання нових завдань, кількість обчислювальних пристроїв тощо. Тому запропоновано в якості універсального засобу, який дозволяє врахувати всі ці зміни, використовувати діаграми Ганта. Вони надають необхідну наочність процесу балансування та досить прості в побудові.

**Аналізостанніхдослідженьіпублікацій.** Використання при динамічному балансуванні показника завантаженості обчислювального пристрою

вперше було запропоновано в роботі [1]. В ній доведено, що застосування цього показника забезпечує отримання мінімального часу виконання всіх завдань за допомогою наявних обчислювальних пристроїв. При цьому враховувалося, що обчислювальні пристрої можуть мати різну продуктивність. Але в наведених прикладах не розглядалися ситуації, коли завдання надходили не одночасно, а кількість обчислювальних пристроїв змінювалася під час балансування.

Стрічкові діаграми, або діаграми Ганта були розроблені більш ніж століття тому [2], але з самого початку й дотепер розглядалися лише в якості зручного та наочного засобу для відображення процесу проектного планування. В той же час для ілюстрації процесу динамічного балансування навантаження обчислювальних пристроїв використовувалися математичні [3] або графо-аналітичні засоби, де для відображення використовувалися графи [4], стовпчасті діаграми [5], структурні схеми [6] або у вигляді черг завдань [7, 8, 9]. Однак в усіх наведених роботах здебільшого розглядаються алгоритми

або статичного балансування, або динамічного зі сталими характеристиками продуктивності обчислювальних пристроїв.

**Постановка завдання.** Під балансуванням навантаження обчислювальних пристроїв в розподілених системах традиційно розуміється розподіл наявних завдань між наявними на цей же час обчислюваними пристроями. При цьому метою балансування є рівномірне завантаження обчислювальних пристроїв. Необхідною умовою виконання балансування є незалежність обчислень в окремих завданнях або можливість декомпозиції складних задач на певну множину окремих модулів, обчислення в яких є незалежними та можуть виконуватися паралельно.

Найпростіший алгоритм Round Robin передбачає розподіл завдань «по колу», коли не враховуються ні складність завдань, ні продуктивність обчислювальних пристроїв. Зрозуміло, що ігнорування цих важливих показників не дозволяє отримати рівномірного розподілу навантаження. Більш досконалими є алгоритми зваженого розподілу: Weighted Round Robin, Least Connections, Sticky Sessions, Throttel Load Balancing та інші [10, 11]. Всі ці алгоритми дозволяють динамічне пересування (міграцію) завдань із найбільш завантажених обчислювальних пристроїв на менш завантажені. Але жодний із них не розглядає ситуації, коли:

- нові завдання надходять вже після проведення балансування;
- деякі обчислювальні пристрої припиняють вирішувати завдання після проведення балансування;
- до обчислювальних пристроїв, наявних на час проведення балансування, додаються нові, обчислювальні потужності яких потрібно врахувати.

Всі ці ситуації потребують нового підходу до відображення процесу динамічного балансування.

**Виклад основного матеріалу дослідження.** В роботі [1] розглянутий спосіб балансування на основі застосування універсального критерію, а саме *мінімальний час виконання всіх завдань на наявних обчислювальних пристроях* (далі – ОП). Необхідно розглянути приклад застосування цього критерію до диспетчеризації динамічного балансування навантаження, але для ілюстрації використовувалися таблиця. Автор проілюстрував цей же приклад за допомогою діаграми Ганта, де час вимірюється в умовних одиницях, бо в умовах масштабування будь-які визначені одиниці часу не мають значення.

### Приклад 1

У наведеному прикладі розглядається балансування 8 завдань, кожне з яких має свою відмінну складність:

$$\begin{aligned} Z_1 &= 5; Z_2 = 3; Z_3 = 2; Z_4 = 1; \\ Z_5 &= 7; Z_6 = 4; Z_7 = 8; Z_8 = 9 \end{aligned}$$

Наявні чотири ОП з такою обчислювальною потужністю:

$$P_1 = 2; P_2 = 1; P_3 = 1; P_4 = 3.$$

**Час виконання**  $i$ -того завдання на певному  $j$ -тому ОП позначено  $T_{ij}$ , який визначається:

$$T_{ij} = \frac{Z_i}{P_j}$$

Також в [1] було введено поняття «*поточний час роботи ОП*  $\tau_j$ », яке відбивало час, що минув з моменту проведення балансування, співпадав із початку роботи ОП. Завдання розподілялися по ОП згідно порядку їх надходження, тобто спочатку розподілялися завдання  $Z_1$ , потім  $Z_2, \dots, Z_N$ . Критерієм призначення завдання  $Z_i$   $j$ -тому ОП вважався найменший поточний час роботи  $\tau_j$ , який буде мати  $j$ -те ОП порівняно з усіма іншими після виконання завдання  $Z_i$ .

Діаграми Ганта у класичному вигляді спрощують роботу над проектами шляхом візуалізації виконання запланованих завдань у часі. Кожному виду робіт відводиться рядок, а окремі етапи виконання розміщуються у вигляді горизонтальних прямокутних блоків, початок і кінець яких синхронізовані в часі.

Для відображення процесу балансування модифікуємо класичну діаграму Ганта: залишимо відлік у часі, але кожний рядок буде зіставлений певному ОП, а завдання, призначені для виконання на ньому, будемо відображати у вигляді прямокутних блоків. Довжина кожного прямокутного блоку буде дорівнювати часу, який потрібен для виконання  $i$ -того завдання на певному  $j$ -тому ОП, тобто  $T_{ij}$ . Звичайно, якщо певне завдання мігрує з одного ОП до іншого, то довжина блоку буде змінюватися: зменшуватися, коли продуктивність ОП більша, або збільшуватися, коли менша.

Диспетчеризація балансування буде назначати  $i$ -те завдання  $j$ -тому ОП, коли для нього буде найменший поточний час роботи  $\tau_j$ , тобто час завершення завдання на  $j$ -тому ОП буде найменший. У графічному вигляді це буде означати, що правий кінець відповідного прямокутного блоку буде

розташований на найменшій відстані від початку діаграми. Наприклад, при розподілі першого завдання  $Z_1$  всі ОП знаходяться в режимі очікування ( $\tau_1 = \tau_2 = \dots \tau_j = \dots \tau_4 = 0$ ), тобто для його виконання буде вибрано той ОП, для якого виконується

$$\min \{T_{1j}\} \text{ для } j \in 1, 2, 3, 4$$

Діаграма Ганта для цього випадку буде виглядати так:

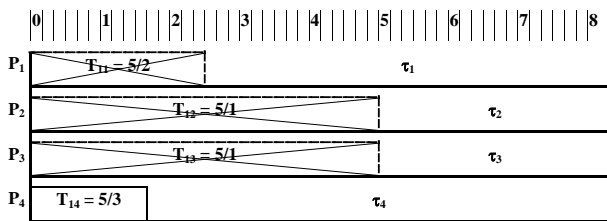


Рис. 1. Діаграма Ганта для розподілу завдання  $Z_1$

Як бачимо з побудованої діаграми Ганта, завдання  $Z_1$  буде призначено ОП  $P_4$ , оскільки час його завершення буде найменший ( $T_{14} < T_{11} < T_{12}, T_{13}$ ), що наочно виглядає як прямокутний блок найменшої довжини з усіх можливих інших варіантів розподілу. Після цього для ОП<sub>4</sub> поточний час роботи  $\tau_4$  вже буде дорівнювати  $5/3$ , для інших ОП він залишиться нульовим ( $\tau_1 = \tau_2 = \tau_3 = 0$ ).

За цими початковими умовами відбудеться розподіл завдання  $Z_2$ .

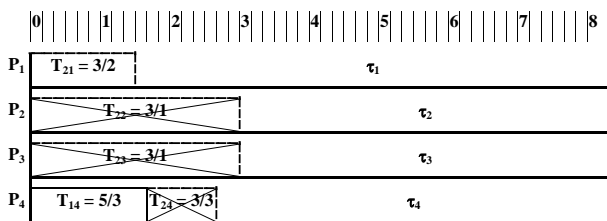


Рис. 2. Діаграма Ганта для розподілу завдання  $Z_2$

Як бачимо з діаграми  $\tau_1 = 3/2$ ,  $\tau_2 = 3$ ,  $\tau_3 = 3$ ,  $\tau_4 = 8/3$ . Оскільки з урахування розподілу завдання  $Z_2$  найменший поточний час роботи буде в ОП<sub>1</sub>, то це завдання буде призначено йому.

Подальші 6 кроків з розподілу завдань  $Z_3$ - $Z_8$  будуть проходити так само. Кінцевий результат відобразиться такою діаграмою:

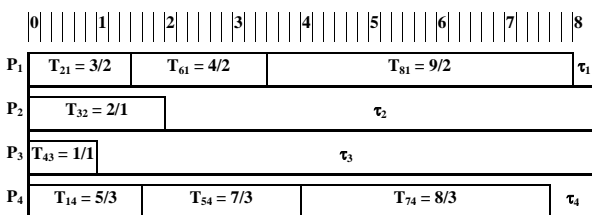


Рис. 3. Діаграма Ганта для розподілу всіх завдань

Таким чином шляхом диспетчеризації процесу балансування 8 завдань розподілені між чотирма обчислювальними пристроями так:

ОП<sub>1</sub>: виконуються завдання  $Z_2, Z_6$  та  $Z_8$ ,  $\tau_1 = 16/2 = 8$

ОП<sub>2</sub>: виконується завдання  $Z_3$ ,  $\tau_2 = 2/1 = 2$

ОП<sub>3</sub>: виконується завдання  $Z_4$ ,  $\tau_3 = 1/1 = 1$

ОП<sub>4</sub>: виконуються завдання  $Z_1, Z_5$  та  $Z_7$ ,  $\tau_4 = 20/3 = 7,67$

Було отримано той же результат, що і в [1], але процес балансування набув більшої наочності і зрозумілості. З побудованої діаграми також легко визначити, що **мінімальний час виконання всіх завдань**:  $\Theta^8 = \max \{\tau_1, \tau_2, \tau_3, \tau_4\} = 8$ . Цей результат розподілення є найкращим за умов розподілу завдань у порядку їх природнього надходження без будь-якого впорядкування.

Використання діаграми Ганта в якості наочної ілюстрації процесу балансування навантаження також дозволяє розглянути ситуацію, коли завдання для розподілу надходять не одночасно, а декількома групами.

### Приклад 2

Скорегуємо первісні умови Прикладу 1 таким чином: завдання  $Z_1$ - $Z_5$  надходять відразу, а завдання  $Z_6$ - $Z_8$  – із затримкою на 4 умовних одиниці часу. Складність завдань і потужність обчислювальних пристроїв автор залишив такою ж, як і у Прикладі 1. Зрозуміло, що балансування на першому етапі перших п'яти завдань не зазнає змін порівняно з Прикладом 1. Знизу буде додано діаграми-позначки часу надходження завдань (Рис. 4).

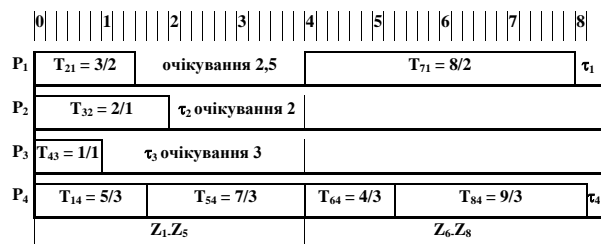


Рис. 4. Діаграма Ганта для Прикладу 2

З побудованої діаграми Ганта вбачається, що на час початку другого етапу балансування, коли додаються завдання  $Z_6$ - $Z_8$  три з чотирьох обчислювальних пристроїв, а саме ОП<sub>1</sub>, ОП<sub>2</sub> та ОП<sub>3</sub> вже закінчили виконання своїх запланованих завдань і знаходяться в режимі очікування. Тому вони можуть розглядатися в якості учасників процесу балансування. Оскільки обчислювальний пристрій ОП<sub>4</sub> зайнятий, то питання врахування його в якості учасника балансування навантаження залежить від часу його звільнення. В такому випадку ОП<sub>4</sub>

завершує виконання завдання  $Z_5$  четвертої умовної одиниці часу, а тому теж може розглядатися в якості учасника балансування навантаження. Тому диспетчеризація балансування навантаження обчислювальних пристроїв ОП<sub>1</sub>-ОП<sub>4</sub> для вирішення завдань  $Z_6$ - $Z_8$  буде відбуватися за вже описаним у Прикладі 1 алгоритмом, але результат його буде відмінним (Рис. 4):

ОП<sub>1</sub>: виконуються завдання  $Z_2, Z_8, \tau_1 = 3/2 + 2,5 + 8/2 = 16/2 = 8$

ОП<sub>2</sub>: виконується завдання  $Z_3, \tau_2 = 2/1 = 2$  (час очікування не враховується)

ОП<sub>3</sub>: виконується завдання  $Z_4, \tau_3 = 1/1 = 1$  (час очікування не враховується)

ОП<sub>4</sub>: виконуються завдання  $Z_1, Z_5, Z_6$  та  $Z_8, \tau_4 = 25/3 = 8,33$

З побудованої діаграми також визначаємо *мінімальний час виконання всіх завдань*:  $\Theta^8 = \max \{ \tau_1, \tau_2, \tau_3, \tau_4 \} = 8,33$ . Відмінність пов'язана з двома етапами балансування, кожний з яких відбувався незалежно один від одного та за різними умовами. Побудована на Рис. 4 діаграма диспетчеризації балансування навантаження обчислювальних пристроїв дає повністю зрозумілу наочність цього процесу.

**Приклад 3**

В цьому прикладі автор розглядає таку ситуацію. Другий етап балансування відбувається, коли ОП<sub>4</sub> ще зайнято виконанням попереднього завдання. Для цього зменшено час затримки другої частки завдань, а саме  $Z_6$ - $Z_8$  із 4 умовних одиниць часу до 2. Всі інші показники з Прикладу 2 залишаються без змін. Результат диспетчеризації балансування навантаження обчислювальних пристроїв можна бачити на Рис. 5.

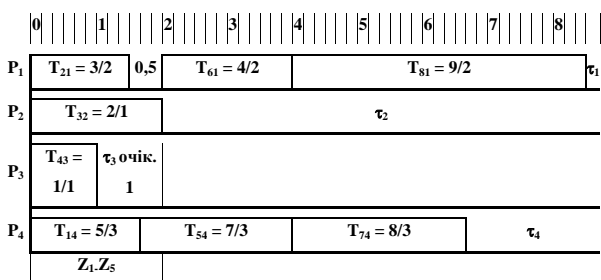


Рис. 5. Діаграма Ганта для Прикладу 3

На першому етапі балансування ніяких відмінностей порівняно з Прикладами 1 і 2 немає. Це

зрозуміло, адже умови щодо складності завдань та обчислювальних потужностей обчислювальних пристроїв теж однакові. Але оскільки час затримки другої частки завдань змінився, бачимо і відмінності порівняно з Прикладом 2.

Справа в тому, що на час другого етапу балансування, пов'язаного з надходженням завдань  $Z_6$ - $Z_8$  вільними залишалися тільки три обчислювальних пристрої: ОП<sub>1</sub>, ОП<sub>2</sub> та ОП<sub>3</sub>, серед яких і відбувалося балансування, а ОП<sub>4</sub> був зайнятий вирішенням завдання  $Z_5$  з першої частки завдань. Тому диспетчеризація відбувалася лише серед наявних вільних обчислювальних пристроїв, а саме завдання  $Z_6$  за критерієм *мінімального часу його виконання* було розподілено на ОП<sub>1</sub>. Оскільки після завершення виконання  $Z_5$  воно звільнилося, то розподіл завдань  $Z_7$  і  $Z_8$  відбувався з урахування потужностей усіх чотирьох ОП.

Визначений з побудованої діаграми *мінімальний час виконання всіх завдань*:  $\Theta^8 = \max \{ \tau_1, \tau_2, \tau_3, \tau_4 \} = 8,5$  лише на 0,5 умовних одиниць часу гірше ніж у Прикладі 1, де ніякої затримки в надходженні завдань взагалі не було. З порівняння діаграм Ганта з Прикладів 1 та 3 стає зрозумілим, що було отримано абсолютно однакові розподіли завдань між ОП, але оскільки мінімальний час  $\Theta^8$  в обох прикладах визначався саме ОП<sub>1</sub>, то режим очікування для цього обчислювального пристрою й був причиною збільшення  $\Theta^8$  на 0,5 умовних одиниць часу.

**Висновки.** У статті проаналізовано традиційні способи опису та ілюстрації методів і алгоритмів динамічного балансування, визначено їх недостатню наочність. Запропоновано використовувати діаграми Ганта в якості універсального засобу відображення процесу динамічного балансування в умовах, коли складність завдань, час їх надходження та потужність обчислювальних пристроїв не є сталими величинами.

Розглянуто послідовність диспетчеризації з використанням раніше запропонованого [1] показника завантаженості обчислювального пристрою. Приведені приклади застосування діаграми Ганта для ілюстрації процесу динамічного балансування в різних умовах.

Визначена перспективність подальшого застосування запропонованого способу відображення балансування в умовах динамічної міграції завдань через додавання або вилучення певних ОП.

**Список літератури:**

1. Павлов В.Г. Диспетчеризація динамічного балансування навантаження обчислювальних пристроїв на основі оцінювання стану поточних обчислень. *Вчені записки Таврійського національного університету*



імені В.І. Вернадського, Серія: «Технічні науки», Ч. 1, том 30, № 2, 2019, ст. 166–170. URL: [www.tech.vernadskyjournals.in.ua/journals/2019/2\\_2019/part\\_1/28.pdf](http://www.tech.vernadskyjournals.in.ua/journals/2019/2_2019/part_1/28.pdf) (дата звернення: 20. 03. 2020).

2. Gantt Henry, Clark Wallace. The Gantt chart, a working tool of management. New-York: The Ronald Press Company, 1922. 186 p. URL: <https://archive.org/details/ganttchartworkin00claruoft/page/n7/mode/2up> (дата звернення: 20. 03. 2020).

3. Mukherjee D., Borst S.C. van Leeuwen J.S.H. & Whiting P.A. Universality of Power-of- $d$  Load Balancing Schemes. *ACM SIGMETRICS Performance Evaluation Review*. 2016. Issue 44(2). P. 36–38.

4. Hu Y.F., Blake R.J., Emerson D.R. An optimal dynamic load balancing algorithm. *Concurrency – Practice and Experience*. 1998. Vol. 10. P. 467–483.

5. Бельков Д.В. Алгоритмы балансировки загрузки процессоров параллельной вычислительной системы. *Наукові праці ДонНТУ*. 2006. Випуск 106. С. 72–76.

6. Хританков А.С. Модели и алгоритмы распределения нагрузки. Алгоритмы на основе сетей СМО. *Информационные технологии и вычислительные системы*. 2009. № 2. С. 65–79.

7. Бо Тянь, Посыпкин М.А., Сигал И.Х. Балансировка нагрузки на основе оценок алгоритмической сложности подзадач. *Информационные технологии и вычислительные системы*, 2015. № 1. С. 10–18.

8. Ebada Sarhan, Atif Ghalwash, Khafagy Mohamed. Queue Weighting Load-Balancing Technique for Database Replication in Dynamic Content Web Sites. *Proceedings of the 9th WSEAS International Conference on Applied Computer Science*. 2009. P. 50–55.

9. Игнатенко Е.Г., Бессараб В.И., Терепалов В.В. Алгоритм адаптивной балансировки нагрузки в кластерных системах. *Моделювання та інформаційні технології: Зб. наук. пр.* Київ: ПІМЕ ім. Г.Є. Пухова НАН України, 2010. Вип. 58. С. 142–150.

10. Abhineet Anand, Kandi Phani Sai Sri Rohith. Analytical Study of different Load balancing algorithms. *International Journal of advanced studies in Computer Science and Engineering (JASCSE) Volume 7, Issue 1, 2018*. URL: [https://www.researchgate.net/publication/323426406\\_Analytical\\_Study\\_of\\_Different\\_Load\\_Balancing\\_Algorithms](https://www.researchgate.net/publication/323426406_Analytical_Study_of_Different_Load_Balancing_Algorithms) (дата звернення: 20.03.2020).

11. Емельянов А. Балансировка нагрузки: основные алгоритмы и методы. *blog.selectel.ru*. 2015. URL: <http://blog.selectel.ru/balansirovka-nagruzki-osnovnye-algoritmy-i-metody> (дата звернення: 20.03.2020).

12. Burkhard Remo Aslak, Meier Michael, Rodgers Peter, Smis Matthias Thomas Jelle, Stott Jonathan. Knowledge visualization: A comparative study between Project Tube Maps and Gantt Charts. *5th International Conference on Knowledge Management. Graz, Austria: University of Kent. Retrieved 17 September 2017*. P. 388–395. URL: <https://kar.kent.ac.uk/14324/1/content.pdf>.

## **Pavlov V.G. GANTT CHART APPLICATION FOR ILLUSTRATION OF PROCESS OF DYNAMIC LOAD BALANCING OF COMPUTING DEVICES**

*In the article it was investigated the mathematical and graphic methods of illustration of static and dynamic load balancing of computing devices. The traditional methods of description do not give the necessary informing and visibility for illustration of methods and algorithms of the dynamic balancing. Especially it is necessary in the case when some indexes can change at implementation of balancing. For example, they are: amount of tasks, their complication, time of receipt of new tasks, and also amount of computing devices.*

*Bar diagrams or Gantt charts were previously considered only as a comfortable and evident mean for the representation of project planning processes. It was suggested to use the horizontal bar Gantt chart as universal method of illustration of process of the dynamic balancing. It allows us to get an understandable graphical representation of the dynamic balancing process. Time descriptions of computing device are used as an index of his workload.*

*They were reviewed examples of demonstration of the dynamic load balancing with application of Gantt chart under different conditions of forming of input queue. For example, there are several parts with a delay in time. These diagrams provided an analysis and an idea of what the total time of all tasks is formed from.*

*A minimum value of this parameter is the goal of the dynamic balancing. The diagrams also provide evidently visual information of the workload of each of the computing devices. Additionally, it was considered possibilities of future improvements of the offered method of load balancing representation.*

**Key words:** distributed computing, load balancing, dynamic balancing, evaluating of loading, horizontal bar chart, Gantt chart.