https://doi.org/10.31891/2307-5732-2025-353-5
УДК 519.853

ЯХНО ВОЛОДИМИР
Київський національний університет технологій та дизайну
https://orcid.org/0000-0001-6129-4178
e-mail yaxno.vm@knutd.com.ua
ДЕМКІВСЬКА ТЕТЯНА
Київський національний університет технологій та дизайну
https://orcid.org/0000-0002-2176-163X
e -mail: demkivskiy@gmail.com
АСТІСТОВА ТЕТЯНА
Київський національний університет технологій та дизайну
https://orcid.org/0000-0002-8452-4797
e -mail: astistova@ukr.net

БАГАТОКРИТЕРІАЛЬНА ЗАДАЧА ОПЕРАТИВНОГО РЕДАГУВАННЯ РОЗКЛАДІВ

В даній роботі наведено технологію, що дозволяє оперативно редагувати календарний план виконання незалежних завдань (робіт) різними незалежними виконавцями.
Ключові слова: моделі задач оперативного планування, теорія розкладів, розклад занять

YAKHNO VOLODYMYR
DEMKIVSKA TETIANA
ASTISTOVA TETIANA
Kyiv National University of Technologies and Design

MULTICRITERION PROBLEM OF OPERATIVE EDITING OF SCHEDULES

This paper presents a technology that allows for quick editing of the calendar plan for the execution of independent tasks (works) by different independent performers. A formal model and algorithm for studying the task of editing the calendar plan for the execution of independent tasks by different independent performers (participants in the works) are considered. Performers have their own ideas about the quality of the schedule but use common resources. A partial case of the task is the schedule of classes in a higher educational institution. An algorithm is proposed that allows for improving or quick editing of the calendar plan so that all changes satisfy the requirements of all participants in the process. The task is typical and can be applied in many cases of operational dispatch planning.

The development of calendar plans for the execution of independent tasks by different independent performers should be implemented in two stages. The first stage is the formation of the initial plan, the second is editing in accordance with the wishes of individual performers. At the second stage, algorithms are needed that allow for the transformation of the known schedule in accordance with the informal wishes of the participants in the process. One of the processes that illustrates such a two-stage planning technology is the class schedule in a higher educational institution.

To implement the task, an analysis of the problems that arise during the preparation of schedules was conducted, formal models were formulated for a wide range of problems that arise, and an algorithm for improving the calendar plan for the performance of independent tasks (works) by various independent performers (participants in the works) was proposed.
Keywords: models of operational planning tasks, scheduling theory, class schedule

Стаття надійшла до редакції / Received 18.03.2025
Прийнята до друку / Accepted 14.04.2025

Постановка проблеми

Для ефективної організації роботи в організаціях та навчальних закладах, розробка календарних планів залишається важливим завданням.

Розробку таких планів для виконання незалежних завдань різними незалежними виконавцями пропонується реалізовувати в два етапи. Перший етап – формування початкового плану, другий - редагування відповідно до побажань окремих виконавців. На другому етапі необхідними є алгоритми, що дозволяють перетворювати відомий розклад відповідно до неформальних побажань учасників процесу. Одним із процесів, що ілюструє подібну двоетапну технологію планування є розклад занять в вищому учбовому закладі. В цьому конкретному випадку автоматизована технологія дозволяє реалізувати важливий і конфліктний етап узгодження планів між виконавцями. Процес узгодження планів між виконавцями є важливим етапом комп’ютерної безпеки, що базується на розподілі інформації між виконавцями. Цей етап можливий лише у випадку коли існують швидкі механізми його реалізації.

Аналіз останніх досліджень

Аналіз останніх досліджень показав, що складання розкладів охоплює широкий спектр завдань у виробництві та економіці, де необхідно визначити оптимальну послідовність або часові інтервали для виконання певного набору операцій [1, 2].

Для деяких таких задач відомі ефективні алгоритми аналізу. Наприклад, це задачі сітьового планування та найпростіші задачі визначення черговості обробки різних деталей на технологічній лінії із двох послідовних верстатів. Для всіх інших задач цього кола пропоновані алгоритми [1, 3] або є евристичними, або будуються на основі повного перебору. Великий перелік задач календарного та оперативно- диспетчерського планування містять роботи [4, 5]. Для багатьох формулювань задач календарного планування є необхідними алгоритми, що дозволяють перетворювати відомий календарний план відповідно до зміни виробничої ситуації [5]. Авторам невідомі формулювання та методи дослідження задачі редагування розкладу робіт, що задовольняють наступним умовам:

-період планування складається послідовністю фіксованих інтервалів;

-в кожному інтервалі може бути виконана деяка кількість робіт;

-кожна робота потребує деякої кількості спільних ресурсів кількість яких обмежена;

-послідовність виконання робіт не має значення – роботи є незалежними;

-виконавці різних робіт різні і мають різні вимоги до якості розкладу;

-зведених кількісних показників якості розкладу не існує.

Зрозумілий для всіх приклад задачі узгодження розкладу занять вищої школи демонструє необхідність застосування такого алгоритму.

Формулювання цілей статті

Метою роботи є визначення технології (моделі або методу дослідження), що дозволяє покращити або оперативно редагувати розклад виконання послідовності незалежних робіт (розклад занять), що задовольняє вимоги всіх учасників процесу. Задача є типовою і може бути застосована у багатьох випадках оперативно диспетчерського планування.

Виклад основного матеріалу

Завдання формування розкладу виглядає наступним чином – існує деякий плановий період розділений на фіксовані в часі інтервали. Визначений перелік робіт які необхідно виконати в плановому періоді і для кожної роботи вже відомий інтервал де вона виконана. Для кожної роботи відомий обмежений перелік ресурсів потрібних для виконання (для розкладу занять, наприклад, ресурси – викладачі, групи студентів та аудиторії). Один ресурс не може бути використаний для виконання двох робіт одночасно – роботи що не використовують спільні ресурси можуть виконуватися одночасно. Окрема група ресурсів виконавці (викладачі та студенти). Кожний виконавець має свої вимоги та критерії якості до розкладу. Формально ці вимоги формулюються у вигляді системи критеріїв тестів - тестів. Критерії визначені для кожного інтервалу і кожної роботи розкладу. Критерії визначають погіршує чи ні якість розкладу для якогось виконавця співставлення цього інтервалу планування та відповідної роботи.

Наприклад для того ж розкладу занять якість для викладачів - це кількість незайнятих пар під час робочого дня, для студентів критерії якості формулюються складніше. На першому етапі календарного планування необхідно визначити для кожної роботи інтервал в якому вона буде виконана. Вважаємо, що вже відомий якийсь варіант розкладу і для цього розкладу відома сукупність побажань виконавців, які можуть покращити розклад. Побажання визначають бажані властивості позиції календарного плану для виконання заданої роботи. Розклад можна покращити переставляючи роботи місцями. Розклад А краще розкладу Б лише в тому випадку якщо критерії якості всіх виконавців розкладу А не гірші ніж Б і існують виконавці для яких вони краще. Зміни в розкладі визначають перестановки робіт в часових інтервалах. Якщо перестановка погіршує один критерій вона не є допустимою. Не всі перестановки є допустимими але для кожної перестановки можна визначити функцію яка визначає покращення розкладу. Кількісної міри покращення розкладу не існує Взагалі це може бути відзнака - покращився розклад чи ні. Побажання учасників визначаються як побажання к позиціям плану (плановим інтервалам), що займають роботи які вони виконують.

Для розкладу занять ця постановка конкретизується наступним чином. Розклад складається на два тижні (періодичний процес, який може бути повністю описаний на цьому інтервалі) відповідно з навчальним планом. Часові інтервали – це послідовність всіх пар в плановому інтервалі, роботи - заняття, які повинні бути проведені, ресурси аудиторії та викладачі. Розклад установлює день тижня та номер пари, на якій проводиться відповідне заняття. Критерії оптимальності або вимоги до плану це, наприклад, для студентів (груп в єдиному розкладі інституту багато) - 32 години на тиждень, 4 години лекційного курсу та 6 годин занять у на день. Для кожної групи критерій окремий. Для викладачів також бажано виконувати обмеження на кількість пар в день та мінімізувати кількість вікон (незайнятих пар) в розкладі. Кількість критеріїв – це кількість груп плюс кількість викладачів.

Схвалюється, коли у розкладі поставлені підряд різні дисципліни, а лекції — на перших та/або других парах (остання вимога для першого та другого курсів). На занятті фізкультурою не повинно бути більше 12 різних груп на одній парі.

Таким чином у кожної групи студентів та кожного викладача свої уявлення про якість розкладу. Це типова багатокритеріальна задача

Прикладом частини задачі уточнення проміжного варіанта є ситуація, що описана нижче.

У списку побажань виконавців є побажання викладача Василенко. Викладач Василенко обґрунтовано просить перенести своє заняття на деяку іншу пару. Адже виявляється, що ця інша пара вже зайнята заняттям, що проводить викладач Петренко. Можливо прохання Василенко можна задовольнити, якщо знайдеться пара викладача Сидоренко, на яку можна перенести заняття Петренко, а заняття Сидоренко при цьому — на заняття Василенко. Якщо жодний з критеріїв не погіршується вважаємо що якість усього розкладу покращується і цей циклічний ланцюжок перестановок є допустимим. В нашому випадку задача оптимізації розкладу – це задача пошуку таких перестановок.

Представимо існуючий розклад у вигляді графа $G = (V, D)$, кожна вершина v_i, $i = 1, ..., n$, якого відповідає інтервалу планування - певному часовому відрізку усередині інтервалу планування (реальному заняттю). З вершиною v_i, пов'язана робота (або кілька робіт), що запланована в цьому інтервалі та ресурси, що необхідні для її виконання. Роботу та ресурси будемо позначати тим же індексом *i.* Існує сукупність алгоритмів - перевірок, що дозволяють визначити чи можна роботу інтервалу і розмістити на позицію роботи інтервалу *j* так, що якість розкладу не погіршується або навіть буде краще. В цьому випадку існує дуга $d_{i,j}$ - в даному випадку роботу інтервалу і можна розмістити на позицію роботи інтервалу *j* і жодний критерій не погіршиться. Послідовність *1* таких перестановок будемо називати припустимою або задовольняючою системі початкових критеріїв - тестів перестановкою (i, j) довжиною 1.

Підкреслимо. Якщо перша позиція перестановки не співпадає з останньою ця перестановка не може бути реалізованою. Перший інтервал планування не зайнятий будь якою роботою. Робота, що планувалась в останньому інтервалі взагалі не виконується.

Лише циклічна перестановка (перший інтервал співпадає з останнім) може використовуватись для змін розкладу. При цьому якість розкладу точно не погіршиться (а може навіть покращиться). Підкреслимо, що лише *циклічна перестановка* може бути можливою (жодний критерій не погіршується) а може покращувати розклад – хоча б один критерій покращено. Матрицю суміжності графа M будемо називати матрицею Парето оптимальних чи припустимих перестановок. Ця назва обгрунтована тим, що якщо початковий розклад є оптимальним за Парето, то будь-який розклад, отриманий шляхом застосування допустимої циклічної перестановки, також буде оптимальним за Парето.

Граф і матриця суміжності, визначені в такий спосіб, володіють такими очевидними властивостями. Будь–яка послідовність суміжних дуг графа утворить шлях. Якщо шлях є циклом, то цей цикл відповідає такій послідовності перестановок робіт, що не погіршує якості всього розкладу. Алгоритм дозволяє одночасно знаходити всі цикли на графі *G* і будувати послідовності перестановок, що утворять ці цикли. Серед цих циклів можуть бути цикли що не погіршують розклад і можуть бути цикли, що його покращують. Будь який такий цикл може бути використаний для поліпшення розкладу.

Надалі будь–який шлях на графі *G*, що відповідає ланцюжку перестановок, що не погіршують розклад, також як і просту перестановку в розкладі двох занять, будемо позначати парою індексів позицій розкладу, які вона з'єднує. Будемо записувати $(i, j) = (i_1, j_1), (i_2, j_2), ..., (i_l, j_l), ... (i_p, j)$.

Іноді ланцюжок (шлях) перестановок, що не погіршують розклад, будемо називати *багатомісною перестановкою* на відміну від простої перестановки пари занять.

Вважаємо, що шлях, складений з *p* перестановок, має *довжину p*. Назвемо шлях довжиною не більш *p припустимим*, якщо він складений із припустимих перестановок. Сукупність всіх *p* — припустимих шляхів на графі (сукупність шляхів, довжина яких не більше *p*) також можна визначити за допомогою матриці суміжності $C^p = (C_{i,j}^P)$. Кожен елемент цієї матриці $C^{P}{}_{i,j}$ дорівнює 1, якщо існує *p* припустимий шлях, що з'єднує відповідні позиції розкладу. Матриці C^P будемо називати *p припустимими*. Відзначимо, що $C^1 = M$. Пропонований алгоритм дозволяє побудувати послідовність *p* – припустимих матриць, $p = 2, ..., P$, де P — мінімальне число менше чи рівне $n – 1$, що може бути представлене у вигляді 2^n. Ці матриці будуються на підставі наступних співвідношень.

Перша матриця $C^1 = M$ визначається вихідними даними і будується на підставі безпосереднього перебору можливих варіантів. Елементи наступних матриць визначаються по індукції у такий спосіб. Нехай отримана послідовність матриць C^P $p=2, ...p^i$. Тоді матриці C^{2p^i}визначаються в такий спосіб:

$C_{j,l}^{2p^i} = 1$, якщо елемент матриці $C^{p^i} C_{j,l}^{p^i} = 1$ або існує l^l, такий, що $C_{j,l^1}^{p^i} = 1$ і $C_{l^1 l}^{p^i} = 1$.

Розглянемо приклад.

Граф *G* представляє фрагмент розкладу. Номера вершин відповідають позиціям пар у розкладі. Напрямок дуг, що з'єднують вершини, вказує на можливість переставити роботу позиції *i* на позицію *j* (наприклад, з першої на другу; з третьої на першу тощо) не погіршуючи якість бажання деякого суб'єкту розкладу.

Побудуємо матрицю C_1, що відповідає графові *G*, матрицю C_2 та матрицю C_4. Нагадаємо дуги визначають, що роботу 3 можна планувати в позиції 1 так далі. Матриця C_2 визначає, якщо роботу 3 можна планувати в позиції 1, то для роботи 1 є місце в позиції 2. Матриця C_4. Визначає, наприклад, що існує маршрут 1 -3 і 3-1. Ці перестановки можуть бути реалізовані. Всі роботи займуть необхідне місце.

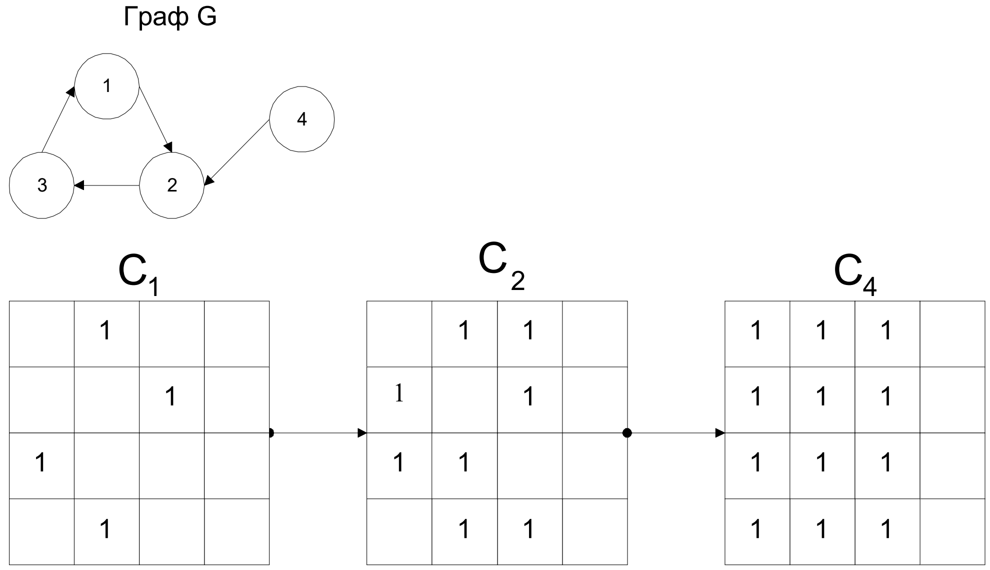

Рис.1. Послідовність матриць покращуючих шляхів для графа G

В даному прикладі немає циклічних маршрутів, довжиною більше трьох, тому далі будувати матриці непотрібно.

Перші три діагональних елемента матриці C_4 дорівнюють 1 і вони створюють цикл. Для даного фрагменту розкладу існує один цикл *(1,2), (2,3), (3, 1)*. Перестановки цього циклу не погіршують розклад. Якщо одна з пропозицій поліпшення розкладу визначається перестановкою цього циклу вона може бути реалізована.

Алгоритм побудови матриць C^P дозволяє переконатися в існуванні бажаного циклу на графі *G*, але не визначає маршрути, які складають послідовність перестановок, що відповідають цьому циклу, можна визначити в результаті виконанні з ряду операцій, які, з міркувань подоби, можна назвати зворотним ходом алгоритму.

Нехай отримана матриця C^P й елементи цієї матриці $C^P_{l_1,j_L} = 1$ і $C^p_{j_L,i} = 1$. Тоді існує цикл на графі *G*. У результаті застосування перестановок, що відповідають цьому циклу до графа *G*, заняття позиції *j* попадає на позицію i_L. Цей цикл утворений двома маршрутами — *(i_l, j_L)* і *(j_L, i_l)*. Обидва маршрути можуть бути отримані в результаті наступного ітеративного алгоритму.

На кожній ітерації алгоритму перетвориться упорядкована послідовність, складена з ненульових (рівних одиниці) елементів різних матриць C^P

$$C^{P^l}_{i_l j_l} \qquad (1)$$

Кожен елемент цієї послідовності визначає багатомісну або просту перестановку. Перестановки, що відповідають всім елементам послідовності утворять багатомісну перестановку *(i_l, j_L)*. У результаті перетворення перестановки $C^{p^l}_{i_l,l_l}$ цих двох елементів також утворять маршрут *(i_l, j_l)*. Спосіб заміни ґрунтується на властивості 2 матриць C^P і полягає в наступному. Якщо елемент $C^{p^l}_{i_l,l_l}$ матриці C^{p^i} дорівнює одиниці, то для матриці $C^{1/2p^i}$ існує такий індекс *l*, що елементи $C^{1/2p^i}_{i,l}$ і $C^{1/2p^i}_{l,j}$ рівні одиниці.

На кожній ітерації елемент $C^{p^l}_{i_1,j_1}$ заміняють двома елементами $C^{1/2p^i}_{i,l}$ і $C^{1/2p^i}_{l,j_1}$, які рівні одиниці і відповідають припустимим перестановкам меншої довжини, що не погіршують розклад.

На першій ітерації алгоритму послідовність (1) складається з одного елемента.

Обчислення припиняються, коли всі перестановки, відповідні елементам послідовності (1) стають простими. Послідовності перестановок, отримані в результаті такого процесу, можуть містити повторювані фрагменти. Ці фрагменти відповідають ситуаціям, коли елементи матриці $C^{2p^i} C^{2p^i}_{j,l} = 1$, тому що елемент матриці C^{p^i} $C^{p^i}_{j,l}$ був рівний одиниці. Повторювані фрагменти повинні бути вилучені. Індекси цієї послідовності визначають припустимий ланцюжок перестановок, що не погіршують. розклад.

Необхідно два етапи для розробка календарних планів для виконання незалежних завдань різними незалежними виконавцями. Перший етап – формування початкового плану, другий - редагування відповідно до побажань окремих виконавців. Запропонована технологія дозволяє на другому етапі, для узгодження неформальних побажань учасників процесу, побудувати перелік всіх можливих змін календарного плану які не погіршують критерії оптимальності жодного учасника процесу. Цей перелік містить зміни плану, що дозволяють перетворювати відомий розклад відповідно до побажань виконавців. Безумовно, якщо такі перетворення існують. Необхідність такої двоетапної технології планування вирішує задача формування розкладу занять в вищому учбовому закладі. В цьому конкретному випадку автоматизована технологія дозволяє реалізувати важливий і конфліктний етап узгодження планів між виконавцями

Висновки

Для вирішення поставленої задачі було проведено аналіз проблем, що виникають під час складання розкладів, сформульовані формальні моделі для широкого кола проблем, що виникають, і запропонований алгоритм покращення календарного плану виконання незалежних завдань (робіт) різними незалежними виконавцями (учасниками робіт).

Запропонована постановка задачі складання розкладів та модель для багатокритеріальної оптимізації розкладів. Алгоритм оптимізації розкладів (узгодження планів для незалежних учасників) формулюється як задача знаходження покращуючих циклів на графі. Запропонована двох-етапна технологія планування (другий етап узгодження планів між учасниками) може бути застосований для виконання незалежних робіт, послідовність виконання яких не має значення. В перспективі алгоритм може бути узагальнений для робіт що мають часткову упорядкованість.

Література

1. Hamdy A. Taha. (2017). Operations Research: An Introduction Pearson Education Limited. P. 833. URL: http://zalamsyah.staff.unja.ac.id/wp-content/uploads/sites/286/2019/11/9-Operations-Research-An-Introduction-10th-Ed.-Hamdy-A-Taha.pdf

2. Ronald L. Rardin (2017) Optimizations in operations research, Pearson Higer educations, P. 1127:URL: https://industri.fatek.unpatti.ac.id/wp-content/uploads/2019/03/173-Optimization-in-Operations-Research-Ronald-L.-Rardin-Edisi-2-2015.pdf

3. F. Werner, L. Burtneva, Y. Sotokov. (2018), Algotitms for Scheduling Problems . – MDPI, P. 192 URL: https://books.google.com.ua/books?hl=uk&lr=&id=Da1qDwAAQBAJ&oi=fnd&pg=PR7&dq=related:xrREKSP4fp0J:scholar.google.com/&ots=9SEhWKyEPz&sig=NcoH5QHB3CLL7uufHdzhuMjzsbo&redir_esc=y#v=onepage&q&f=false

4. Pinedo, M.L. (2012) Scheduling: Theory, Algorithms, and Systems. Springer, NY, USA. P. 672. URL: https://www.google.com.ua/books/edition/Scheduling/QRiDnuXSnVwC?hl=ru&gbpv=1&dq=inauthor:%22Michael+L.+Pinedo%22&printsec=frontcover

5. Low, C., Li, R. K., Wu, G. H., & Huang, C. L. (2015). Minimizing the sum of absolute deviations under a common due date for a single-machine scheduling problem with availability constraints. Journal of Industrial and Production Engineering, 32(3), 204-217.2017. URL: https://doi.org/10.1080/21681015.2015.1031196

References

1. Hamdy A. Taha. (2017). Operations Research: An Introduction Pearson Education Limited. P. 833. URL: http://zalamsyah.staff.unja.ac.id/wp-content/uploads/sites/286/2019/11/9-Operations-Research-An-Introduction-10th-Ed.-Hamdy-A-Taha.pdf

2. Ronald L. Rardin (2017) Optimizations in operations research, Pearson Higer educations, P. 1127 URL: https://industri.fatek.unpatti.ac.id/wp-content/uploads/2019/03/173-Optimization-in-Operations-Research-Ronald-L.-Rardin-Edisi-2-2015.pdf

3. F. Werner, L. Burtneva, Y. Sotokov. (2018), Algotitms for Scheduling Problems . – MDPI, P. 192 URL: https://books.google.com.ua/books?hl=uk&lr=&id=Da1qDwAAQBAJ&oi=fnd&pg=PR7&dq=related:xrREKSP4fp0J:scholar.google.com/&ots=9SEhWKyEPz&sig=NcoH5QHB3CLL7uufHdzhuMjzsbo&redir_esc=y#v=onepage&q&f=false

4. Pinedo, M.L. (2012) Scheduling: Theory, Algorithms, and Systems. Springer, NY, USA. P. 672. URL: https://www.google.com.ua/books/edition/Scheduling/QRiDnuXSnVwC?hl=ru&gbpv=1&dq=inauthor:%22Michael+L.+Pinedo%22&printsec=frontcover

5. Low, C., Li, R. K., Wu, G. H., & Huang, C. L. (2015). Minimizing the sum of absolute deviations under a common due date for a single-machine scheduling problem with availability constraints. Journal of Industrial and Production Engineering, 32(3), 204-217.2017. URL: https://doi.org/10.1080/21681015.2015.1031196