

УДК 004.4

DOI: 10.18372/2073-4751.85.21121

Русанова О. В., к.т.н.,
orcid.org/0000-0003-0145-3012,
olga.rusanova.v@gmail.com,

Корочкін О. В., к.т.н.,
orcid.org/0000-0002-6569-5849,
avcora@gmail.com,

Кучеренко У. В.,
orcid.org/0009-0001-6359-7145
ulianasurg7@gmail.com,

Шевело О. П.,
orcid.org/0009-0001-9938-2760
alex.shevelo@gmail.com

СИСТЕМА ОПТИМІЗАЦІЇ РОЗКЛАДУ СПРАВ З УРАХУВАННЯМ УПОДОБАНЬ КОРИСТУВАЧА

НТУУ «Київський політехнічний інститут імені Ігоря Сікорського»

Вступ

Організація завдань у сучасному світі є надзвичайно складним і багатогранним процесом. Кількість справ, які необхідно виконати щодня, постійно зростає, і людина змушена розподіляти свій час між роботою, навчанням, особистими справами та відпочинком. При цьому у більшості користувачів відсутні дієві інструменти, які дозволяють не лише структурувати перелік задач, а й ефективно оптимізувати їх порядок виконання.

Багато сучасних додатків - Google Calendar, Microsoft To Do, Asana, Trello - справді допомагають упорядковувати перелік завдань. Проте вони фактично зводяться до створення "списку справ" без глибшого аналізу. У результаті користувач стикається з проблемою: навіть маючи всі задачі перед очима, він змушений самостійно вирішувати, у якій послідовності краще їх виконати, як відреагувати на зміни, які завдання можуть бути перенесені, а які - ні.

Ще однією важливою проблемою є відсутність адаптивності. Більшість інструментів не вміють перебудувати розклад у разі непередбачуваних змін. Наприклад, якщо одна зустріч скасована, то користувачеві доводиться вручну змінювати увесь розклад, що має додатковий час і створює ризик помилок. У масштабах великих компаній це призводить до значних втрат робочого часу.

Крім того, варто враховувати і психологічний аспект. Велика кількість завдань, що накопичуються без чіткого плану виконання, може викликати стрес, відчуття перенавантаження та навіть зниження мотивації. За дослідженнями Американської асоціації психологів (APA), близько 45% працівників зазначають, що основним джерелом стресу на роботі є саме неефективна організація завдань і розкладу [1]. Це ще раз доводить необхідність створення інтелектуальних систем, які б знімали з людини "організаційне навантаження" і дозволяли зосередитися безпосередньо на виконанні роботи.

Сучасні системи управління завданнями та календарного планування надають широкий спектр можливостей для організації робочого часу, координації командних процесів та підвищення ефективності індивідуальної діяльності. Проте, попри значний розвиток технологій, ці інструменти не завжди задовольняють потреби користувачів, особливо в контексті індивідуальної оптимізації розкладу з урахуванням їхніх особистих уподобань і пріоритетів.

Календарні сервіси.

Найпоширенішими інструментами для організації часу є календарні застосунки, такі як Google Calendar чи Microsoft Outlook Calendar. Вони дозволяють створювати події, виставляти нагадування, здійснювати синхронізацію між

пристроями та ділитися розкладом із колегами. Проте, незважаючи на високий рівень інтеграції з іншими сервісами, більшість календарних систем не пропонують механізмів автоматичної оптимізації завдань. Користувач має вручну визначати час виконання справ, що стає незручним у випадках великої кількості змінних подій або необхідності швидкого перепланування.

Системи управління завданнями. Trello, Asana, Todoist орієнтовані на візуалізацію завдань, управління робочими процесами та командну співпрацю. Вони добре підходять для проектного менеджменту, але здебільшого зосереджені на статичному представленні задач: користувач самостійно розподіляє їх за пріоритетами і термінами виконання. У роботі [2] запропонована система управління проектами з використанням автоматичного розподілу задач проектів між співробітниками без урахування динаміки зміни умов. Автоматична перебудова розкладу при зміні умов або врахування особистих уподобань користувача відсутні, що знижує ефективність у динамічних середовищах.

Мобільні планувальники. Багато мобільних застосунків для організації часу пропонують базові можливості нагадувань, чек-листів і віджетів. Їхніми перевагами є простота використання та доступність, проте функціональність часто обмежується лише базовим відстеженням виконання завдань. Вони не мають потужних алгоритмів аналітики чи оптимізації, а отже, не здатні гнучко адаптувати розклад до змін у пріоритетах чи нових обставин.

Таблиця 1

| Функціональність | Google Calendar | MS Outlook Calendar | Todoist | Trello | Розробка |
|-----------------------------|-----------------|---------------------|---------|--------|----------|
| Створення подій | Так | Так | Так | Так | Так |
| Пріоритети задач | Обмежено | Частково | Так | Так | Так |
| Автоматичне планування | Ні | Ні | Ні | Ні | Так |
| Адаптивне перепланування | Ні | Ні | Ні | Ні | Так |
| Врахування фіксованих подій | Так | Так | Так | Так | Так |
| Врахування плаваючих подій | Частково | Частково | Так | Так | Так |
| Оптимізація розкладу | Ні | Ні | Ні | Ні | Так |
| Локальна оптимізація | Ні | Ні | Ні | Ні | Так |
| Інтелектуальні алгоритми | Ні | Ні | Ні | Ні | Так |

Таким чином, хоча існуючі технології значно полегшують процес правління завданнями, вони здебільшого залишають користувача “сам на сам” із необхідністю оптимізації власного часу. Це створює передумови для розвитку нових систем, орієнтованих на автоматичну побудову та адаптацію розкладу з урахуванням пріоритетів і вподобань користувача.

Постановка задачі

Завдання даної роботи полягає в підвищенні ефективності управління особистим та корпоративним часом за допомогою створення інноваційної системи оптимізації розкладу справ з урахуванням індивідуальних уподобань користувача. Така система має забезпечувати автоматизоване планування завдань, динамічне перепланування при змінах та можливість інтеграції з іншими сервісами для формування максимально зручного та адаптивного розкладу.

Розробка спрямована на вдосконалення існуючих методів планування шляхом поєднання евристичних алгоритмів (жадібного підходу та локального пошуку) з урахуванням користувацьких пріоритетів і специфіки завдань (фіксованих чи змінних). В результаті система повинна не лише створювати “календар справ”, а й пропонувати оптимізовані сценарії їх виконання, підвищуючи продуктивність та комфорт користувача. У таблиці 1 наведені основні характеристики аналогів і переваги запропонованої системи.

Мета

Мета роботи полягає у підвищенні ефективності процесу планування завдань шляхом створення веб-застосунку, який реалізує алгоритм комбінованої оптимізації та забезпечує адаптивне перепланування з урахуванням вказаних опцій користувача.

Основна частина

Для побудови інтелектуальної системи оптимізації розкладу, яка враховує обмеження, продуктивність користувача та його індивідуальні пріоритети, необхідно проаналізувати найпоширеніші методи оптимізації, які будуть найдоцільнішими для задач типу динамічного календарного планування.

Жадібні алгоритми (Greedy Algorithm) - це клас евристичних

$$f(A_i) = \frac{\min}{t \in T} \{t \mid t \geq t_{min} \wedge t + d_i \leq t_{max} \wedge t \notin S\} \quad (1)$$

де A_i - поточне завдання; d_i - його тривалість; S - множина зайнятих інтервалів.

Алгоритм швидко генерує початковий розклад, проте не гарантує глобальної оптимізації, оскільки не "повертається" до попередніх кроків. Перевагами даного алгоритму є висока швидкодія, мінімальне використання ресурсів, простота реалізації. В той же час, даний алгоритм може призвести до конфліктів між завданнями та не враховувати зміни продуктивності користувача протягом доби.

Отже, жадібний підхід ідеально підходить для початкового етапу формування розкладу в системі, оскільки дозволяє швидко створити базову структуру, яка згодом може бути покращена іншими методами.

Методи локального пошуку (Local Search) - це евристичні алгоритми вдосконалення, які працюють із вже існуючим розкладом, поступово покращуючи його шляхом локальних змін.

Принцип дії полягає у переході від поточного рішення S до одного з його

методів, які ґрунтуються на послідовному прийнятті локально оптимальних рішень. На кожному кроці алгоритм вибирає дію, що здається найвигіднішою в даний момент, не враховуючи майбутніх наслідків [3].

У задачі розкладу це означає, що система за наступним алгоритмом:

1. Сортує усі завдання за пріоритетом p_i ;
2. Для кожного завдання вибирає найближчий вільний часу у межах допустимого проміжку;
3. Додає його до розкладу, позначаючи інтервал як зайнятий.

Математична модель жадібного алгоритму виглядає як (1):

сусідніх станів S' , якщо останній покращує певну функцію якості (або цільову функцію). Процес повторюється доти, поки не буде досягнуто стабільного стану (локального мінімуму).

Цільова функція для задачі розкладу має такий вигляд (2):

$$F = \alpha C + \beta P + \gamma D, \quad (2)$$

де: C - кількість конфліктів між завданнями; P - сумарне відхилення від продуктивних годин; D - сумарна затримка виконання щодо бажаного часу; α, β, γ - вагові коефіцієнти, що задають пріоритет критеріям.

Алгоритм поступового змінює послідовність або час завдань, доки F не досягне мінімального значення. Алгоритм працює за наступною послідовністю:

1. Вибір сусіднього стану S' (перестановка або зсув завдання).
2. Перерахунок функції $F(S')$.
3. Якщо $F(S') < F(S)$, прийняти новий стан.
4. Повторювати до стабілізації або досягнення граничної кількості ітерацій.

Перевагами даного алгоритму є підвищення якості розкладу, зменшення

кількості конфліктів, можливість враховувати продуктивні інтервали користувача. Недоліком може бути підвищена обчислювальна складність та можливість “застрягти” у локальних мінімумах.

Можна зробити висновок, що такий метод доцільно використовувати після жадібного етапу - для точного балансування задач та покращення ефективності графіка користувача без надмірного навантаження на систему.

Адаптивне перерозподілення завдань (Adaptive Rescheduling).

Адаптивні алгоритми оптимізації розкладу призначені для реактивної адаптації системи до змін умов або параметрів користувача [4]. Їх мета - не створювати розклад “з нуля”, а оновлювати лише його змінену частину, що дозволяє зберегти стабільність та економити ресурси.

Наприклад, якщо користувач додає нове завдання або змінює час сну, алгоритм виконує лише часткову перебудову розкладу, не змінюючи інші його елементи.

Даний алгоритм має таку математичну постановку (3):

$$\min_{S'} \{F(S') - F(S)\}, \text{де } S' \subset S \quad (3)$$

Тут S' - множина завдань, які потрібно перебудувати, а $F(S)$ - поточна функція якості розкладу.

До переваг даного алгоритму відносяться гнучкість та швидкість реакції, мінімальна обчислювальна вартість, можливість оновлення у реальному часі. До можливих недоліків відносяться потреба в складній логіці синхронізації та ризик накопичення дрібних помилок при надто частих змінах.

Тому адаптивне перерозподілення необхідне для динамічного оновлення календаря користувача без повного перерахунку, що робить систему стабільною при постійному редагуванні завдань.

Жадібний алгоритм вирізняється високою швидкістю та простотою

реалізації, що робить його ефективним для початкового формування розкладу. Проте, оскільки він приймає лише локально оптимальні рішення, результат часто виявляється далеким від глобального оптимуму - особливо у випадках, коли потрібно враховувати складні обмеження, такі як інтервали сну користувача чи варіації продуктивності протягом доби.

Метод локального пошуку, навпаки, дозволяє покращити вже існуючий розклад, зменшити кількість конфліктів і підвищити узгодженість із персональними пріоритетами. Однак, цей метод вимагає значно більших обчислювальних ресурсів і не підходить для моментального реагування на зміни. Таким чином, локальний пошук є доцільним на етапі вдосконалення базового рішення, створеного жадібним алгоритмом.

У свою чергу, адаптивне перерозподілення завдань забезпечує гнучкість і динамічну стабільність системи. Воно дозволяє оновлювати розклад частково, без необхідності повного перерахунку, що ж критично важливим для користувача, який регулярно додає або змінює події. Проте самостійне застосування цього методу не забезпечить якісного плану - його доцільно використовувати як підтримувальний механізм для оперативного реагування на зміни в уже оптимізованому графіку.

Саме тому логічним рішенням стало поєднання цих трьох методів у комбінований алгоритм, який використовує переваги кожного з них у різних етапах роботи системи.

Основна ідея такого поєднання полягає у послідовному виконанні трьох етапів оптимізації, кожен із яких відповідає за певну функціональну частину процесу.

1. Етап формування базового розкладу

На першому етапі система отримує вхідні дані: список завдань

користувача з їх пріоритетами p_i , тривалістю d_i та часовими обмеженнями; періоди сну та активності користувача; фіксовані (незмінні) події календаря.

Жадібний алгоритм формує базовий розклад, послідовно розподіляючи завдання за принципом “від найважливішого до найменш важливого” у вільні часові інтервали. Якщо знайдений інтервал задовольняє всі обмеження користувача (не перетинається з іншими завданнями, не потрапляє в час сну чи зниження продуктивності), завдання розміщується у ньому.

Результатом є швидко побудований, але ще не оптимізований графік, який слугуватиме вхідними даними для наступного етапу.

2. Етап вдосконалення

Після побудови початкового розкладу запускається етап локального вдосконалення, який виконує аналіз отриманого графіка за допомогою цільової функції якості. Алгоритм виконує ітераційний пошук кращого рішення, змінюючи місця завдань або їх часові межі. Якщо нова конфігурація зменшує значення функції, то вона приймається як новий поточний стан.

Процес триває доти, доки не буде досягнуто стабільного розкладу, де подальші зміни не покращують результат.

Цей етап виконує роль “тонкого налаштування”, підвищуючи узгодженість графіка з реальними умовами користувача.

3. Етап адаптації

На заключному етапі система переходить у режим динамічного оновлення. Коли користувач додає нову задачу або змінює параметри сну чи пріоритетів, повний перерахунок усього розкладу не проводиться. Замість цього застосовується адаптивний підхід, при якому перебудовується лише локальна частина розкладу, що була порушена зміною.

Алгоритм визначає зону впливу зміни (4):

$$\Delta S = \{t_i \mid t_i \in [t_{new} - \delta, t_{new} + \delta]\} \quad (4)$$

і виконує перебудову тільки для ΔS , залишаючи решту графіка незмінною.

Такий підхід дозволяє забезпечити стабільність і швидке реагування системи навіть при частих оновленнях даних, що є критично важливим для користувача в умовах реального часу. Поєднання трьох методів дозволяє забезпечити комплексне розв'язання задачі оптимізації розкладу. У таблиці 2 наведені характеристики використання окремих методів та комбінованого підходу.

Таблиця 2

| Характеристика | Жадібний алгоритм | Локальний пошук | Адаптивне перерозподілення | Комбінований алгоритм |
|----------------------------|------------------------------|-----------------------|----------------------------|----------------------------|
| Швидкість виконання | Висока | Середня | Висока | Висока |
| Точність рішення | Низька | Висока | Середня | Висока |
| Гнучкість системи | Низька | Середня | Висока | Висока |
| Реакція на зміни | Слабка | Середня | Висока | Висока |
| Обчислювальна ефективність | Висока | Середня | Висока | Оптимальна |
| Використання у системі | Формування початкового плану | Оптимізація структури | Реакція на зміни | Повний цикл роботи системи |

Отже, перевагами такого підходу є:

Підвищення швидкодії системи за рахунок жадібного етапу; Покращення точності завдяки локальному пошуку; Гнучкість і стабільність через адаптивний перерозподіл при динамічних змінах; Відсутність конфліктів між завданнями навіть при значній кількості записів; Персоналізація розкладу - система враховує індивідуальні часові патерни користувача.

Після формування початкового розкладу система переходить до етапу локальної оптимізації. Локальний пошук дозволяє покращити якість розкладу, зменшивши накладання завдань та підвищивши ефективність використання часу.

Даний комбінований алгоритм поєднує найсильніші сторони кожного з розглянутих методів. Такий підхід забезпечує баланс між продуктивністю, точністю та гнучкістю, що є ключовими вимогами для сучасних систем планування.

Висновки

У роботі було проведено аналіз сучасних підходів до планування часу та організації особистих і робочих розкладів, а також розглянуто функціональні можливості популярних календарних сервісів та систем управління завданнями. На основі цього аналізу було встановлено, що більшість наявних інструментів не пропонують автоматизованого формування оптимального графіка, не враховують індивідуальних пріоритетів користувача та не забезпечують адаптивного перерозподілу завдань у разі змін у розкладі.

У рамках дослідження було запропоновано підхід до автоматизації побудови розкладу, який поєднує жадібні алгоритми формування початкового графіка, методи локального пошуку для

покращення розміщення подій та адаптивні механізми перебудови календаря при динамічних змінах. Така комбінація дозволила створити систему, здатну швидко й ефективно генерувати оптимізований розклад, мінімізувати накладання завдань, зменшувати простоті та зберігати пріоритети користувача в умовах обмежень реального життя.

Запропоновані алгоритми враховують різні типи задач, їх фіксовані та плаваючі властивості, пріоритети, часові межі та цільові групи активностей, що робить створений механізм ближчим до реальних сценаріїв використання, ніж традиційні, переважно ручні підходи. Результати експериментального тестування підтвердили високу точність і стабільність роботи системи, її здатність адаптуватися до змін без втрати структури та її перевагу за швидкістю та ефективністю над популярними аналогами.

На основі розроблених алгоритмів створено веб-застосунок, який забезпечує інтерактивну роботу з календарем, автоматичну оптимізацію розкладу та зручну взаємодію з користувачем. Завдяки модульній архітектурі система може бути розширена для корпоративного використання, групового планування, інтеграції з іншими сервісами та подальшої автоматизації процесів управління часом.

Література

1. Workplace Stress Survey Report [Електронний ресурс] / American Psychological Association. – 2022. – Режим доступу до ресурсу: <https://www.apa.org/>.
2. Спосіб управління проектами на базі оцінок STORY POINTS / О.В.Русанова О.В., О.В.Корочкін, А.В.Ачілов// Проблеми інформатизації та управління-2024.-№1(77)-С.96-103.
3. Kleinberg J., Tardos É. Algorithm Design. — Pearson, 2014.
4. IBM Research. Adaptive Scheduling Optimization in Dynamic Environments. — IBM Technical Report, 2022

**Русанова О.В., Корочкін О.В., Кучеренко У. В., Шевело О. П.
СИСТЕМА ОПТИМІЗАЦІЇ РОЗКЛАДУ СПРАВ З УРАХУВАННЯМ УПОДОБАНЬ
КОРИСТУВАЧА**

Робота присвячена розробці методів автоматизованого формування особистих та робочих розкладів, у яких завдання мають різні пріоритети, часові обмеження та типи виконання. У дослідженні було проведено огляд сучасних календарних сервісів та систем управління задачами, виявлено їх ключові обмеження та запропоновано новий підхід до побудови оптимізованого розкладу. Розроблений метод поєднує жадібні алгоритми формування початкового графіка, локальні методи вдосконалення та адаптивне перерозподілення завдань, що забезпечує мінімізацію накладань, ефективно використання часу користувача та стабільність графіка при його зміні.

Запропонований підхід дозволяє враховувати індивідуальні пріоритети, наявність фіксованих подій, важливість завдань, можливі часові вікна та поведінкові патерни користувача. Завдяки цьому формування розкладу наближається до реальних умов, а сама система здатна підвищити ефективність планування, зменшити втрати часу та забезпечити високу точність адаптації при динамічних змінах у графіку.

Ключові слова: *планування задач, автоматизація розкладу, оптимізація часу, жадібні алгоритми, локальний пошук, адаптивне перерозподілення, календарні системи.*

**Rusanova O. V., Korochkin O.V., Kucherenko U. V, Shevelo O. P.
SYSTEM FOR OPTIMIZING THE SCHEDULE OF CASES TAKING INTO
ACCOUNT USER PREFERENCES**

The work is devoted to the development of methods for automated formation of personal and work schedules in which tasks have different priorities, time constraints, and types of execution. The study reviewed modern calendar services and task management systems, identified their key limitations, and proposed a new approach to building an optimized schedule. The developed method combines greedy algorithms for forming the initial schedule, local improvement methods, and adaptive task redistribution, which ensures minimization of overlaps, efficient use of user time, and stability of the schedule when it changes.

The proposed approach allows for individual priorities, fixed events, task importance, possible time windows, and user behavior patterns to be taken into account. As a result, the schedule formation is closer to real conditions, and the system itself is capable of increasing planning efficiency, reducing time losses, and ensuring high accuracy of adaptation in case of dynamic changes in the schedule.

Keywords: *task planning, schedule automation, time optimization, greedy algorithms, local search, adaptive redistribution, calendar systems.*

Стаття подана до редакції: 18/02/2026

Стаття прийнята до опублікування: 27/02/2026

Стаття опублікована: 27/04/2026

Стаття поширюється на умовах ліцензії CC BY 4.0