

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління

СІКАЧ Богдан Ярославович

**Метод прогнозування рангів розгалужених процедур при
плануванні проєкту з обмеженими ресурсами / Prediction
method for ranking branching procedures in resource-
constrained project planning**

спеціальність: 122 - Комп'ютерні науки
освітньо-професійна програма - Управління проектами

Кваліфікаційна робота

Виконав студент групи КНУПм-21
Б. Я. Сікач

Науковий керівник:
к.т.н., доцент, Саченко О. А.

Кваліфікаційну роботу
допущено до захисту:
«___» _____ 20___ р.
Завідувач кафедри
_____ М.П. Комар

ТЕРНОПІЛЬ - 2023

Факультет комп'ютерних інформаційних технологій

Кафедра інформаційно-обчислювальних систем і управління

Освітній ступінь «магістр»

спеціальність: 122 – Комп'ютерні науки

освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри

М.П. Комар

« ____ » _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Сікачу Богдану Ярославовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

Метод прогнозування рангів розгалужених процедур при плануванні проєкту з обмеженими ресурсами / Prediction method for ranking branching procedures in resource-constrained project planning

керівник роботи к.т.н., доцент, Саченко О. А.

затверджені наказом по університету від 8 грудня 2022 року № 491.

2. Строк подання студентом закінченої кваліфікаційної роботи 1 грудня 2023 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити

1. Аналіз існуючих методів РОПП.
2. Розробка методу прогнозування рангів конфігурацій.
3. Алгоритм прогнозування рангів конфігурацій для РОПП.
4. Оцінка продуктивності розробленого методу.
5. Аналіз та порівняння розробленого методу.
6. Практичне використання отриманих результатів.

5. Перелік графічного матеріалу у роботі

Структура методу прогнозування рангів конфігурацій для РОПП.

Порівняння продуктивності в чотирьох випадках

Оптимальні налаштування параметрів для різних експериментів

Порівняння моделі прогнозування рейтингу

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 8 грудня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Аналіз предметної області та постановка завдань	12.2022 р. – 03.2023 р.	
2	Метод прогнозування рангів конфігурацій для РОПП	03.2023 р. – 05.2023 р.	
3	Практичне використання отриманих наукових результатів	05.2023 р. – 11.2023 р.	
4	Повне завершення та представлення кваліфікаційної роботи на кафедрі	01.12.2023 р.	

Студент _____ Б. Я. Сікач
підпис

Керівник роботи _____ к.т.н., доцент, Саченко О. А.
підпис

РЕЗЮМЕ

Кваліфікаційна робота на тему «Метод прогнозування рангів розгалужених процедур при плануванні проекту з обмеженими ресурсами» на здобуття ступеня вищої освіти «Магістр» зі спеціальності 122 «Комп'ютерні науки» освітньо-професійної програми «Управління проектами» написана обсягом 94 сторінок і містить 4 ілюстрацій, 10 таблиць, 3 додатки та 61 використаних джерел.

Метою кваліфікаційної роботи є розробка методу прогнозування рангів конфігурацій для ресурсно-обмеженого планування проектів (РОПП), який дозволить автоматично визначати ефективні методи планування проектів в умовах обмежених ресурсів.

Методи дослідження: у роботі застосовані методи аналітичного моделювання, статистичного аналізу, машинного навчання та експериментальної верифікації.

Результати дослідження: розроблено новаторський метод прогнозування, який інтегрує різноманітні конфігурації проектних процедур для оптимізації управління обмеженими ресурсами. Метод дозволяє значно підвищити ефективність проектного планування, знижуючи ризики затримок та неефективного використання ресурсів.

Результати роботи мають важливе практичне значення для управління проектами, зокрема в галузях з високим ступенем ресурсних обмежень, що відкриває шляхи для подальшого вдосконалення та адаптації проектних методів управління.

Ключові слова: ПРОГНОЗУВАННЯ РАНГІВ, РЕСУРСНО-ОБМЕЖЕНЕ ПЛАНУВАННЯ ПРОЕКТІВ, УПРАВЛІННЯ ПРОЕКТАМИ, МАШИННЕ НАВЧАННЯ, ОПТИМІЗАЦІЯ РЕСУРСІВ

RESUME

The thesis entitled "Prediction method for ranking branching procedures in resource-constrained project planning " for the degree of Master of Higher Education in specialty 122 "Computer Science" under the professional program "Project Management" is written on 94 pages and contains 4 illustrations, 10 tables, 3 appendices, and 61 sources used.

The purpose of the thesis is to develop a method for predicting the ranks of configurations for Resource-Constrained Project Scheduling (RCPS), which will allow for the automatic determination of effective project planning methods in conditions of limited resources.

Research methods: the work employs methods of analytical modeling, statistical analysis, machine learning, and experimental verification.

Research results: A novel predictive method has been developed, integrating various configurations of project procedures for the optimization of managing limited resources. This method significantly enhances the efficiency of project planning by reducing the risks of delays and inefficient resource utilization.

The results of the work have significant practical importance for project management, especially in fields with a high degree of resource constraints, opening pathways for further improvement and adaptation of project management methods.

Key words: RANK PREDICTION, RESOURCE-CONSTRAINED PROJECT SCHEDULING, PROJECT MANAGEMENT, MACHINE LEARNING, RESOURCE OPTIMIZATION.

ЗМІСТ

ВСТУП	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ	12
1.1 Суть та особливості ресурсно-обмеженого планування проектів	12
1.2 Огляд існуючих підходів	16
1.3 Постановка задачі	23
Висновки до розділу 1	27
2 МЕТОД ПРОГНОЗУВАННЯ РАНГІВ КОНФІГУРАЦІЙ ДЛЯ РОПП.	28
2.1. Прогнозування рангів в задачах ресурсно-обмеженого планування проектів.....	28
2.2. Метод прогнозування рангів конфігурацій для РОПП	29
2.3. Алгоритм прогнозування рангів конфігурацій для РОПП.....	33
Висновки до розділу 2	39
3 ПРАКТИЧНЕ ВИКОРИСТАННЯ ОТРИМАНИХ НАУКОВИХ РЕЗУЛЬТАТІВ	40
3.1. Оцінка продуктивності структурованої моделі прогнозування.....	40
3.2. Аналіз продуктивності прогностичних моделей.....	43
3.2.1. Експеримент 1. Налаштування параметрів	44
3.2.2. Експеримент 2. Якість прогнозування.....	49
3.2.3. Експеримент 3. Різні конфігурації.....	60
3.2.4. Експеримент 4. Надійність моделі	62
3.2.5. Експеримент 5. Розподіл обмеження за часом.....	64
3.3. Порівняння розробленого методу з існуючими методами.....	67
Висновки до розділу 3	72
ВИСНОВКИ	73

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	75
ДОДАТОК А 48 можливих комбінацій всіх компонентів	82
ДОДАТОК Б Дослідження різних компонентів	83
Додаток В Апробація отриманих результатів	85

ВСТУП

У світі, де ринкові умови швидко змінюються, ефективне управління проектами стає ключовим фактором успіху для багатьох компаній. Ресурсно-обмежене планування проектів (РОПП) зіткнулося з викликами, пов'язаними з обмеженнями ресурсів та збільшенням складності проектів. В цьому контексті, розробка методу прогнозування рангів конфігурацій може внести значний внесок у поліпшення ефективності та гнучкості планування проектів.

РОПП часто включає багато обмежень та змінних, що ускладнює вибір оптимального методу планування. Традиційні методи можуть не враховувати всіх нюансів конкретного проекту, що може призвести до неефективного використання ресурсів. Новий метод прогнозування рангів конфігурацій дозволяє більш точно вибирати стратегії планування, що враховують унікальні характеристики кожного проекту.

Зростаюча складність проектів та обмеження ресурсів вимагають більшої автоматизації процесів управління проектами. Автоматизація за допомогою прогнозування рангів може значно підвищити продуктивність процесу планування, мінімізуючи людські помилки та забезпечуючи більш точне та ефективне розподілення ресурсів.

Новий метод може бути інтегрований з існуючими системами управління проектами, забезпечуючи більш глибокий аналіз та краще розуміння проектних процесів. Це дозволяє управлінцям проектів краще адаптуватися до змінних умов і приймати більш обґрунтовані рішення.

Поліпшення методів РОПП має значне значення не тільки для ІТ-проектів, але й для широкого спектру галузей, включаючи будівництво, виробництво, наукові дослідження та інші. Впровадження ефективних методів планування проектів може значно підвищити загальну продуктивність та конкурентоспроможність організацій.

Метою даного дослідження є розробка методу прогнозування рангів конфігурацій для РОПП, який дозволить автоматично визначати ефективні методи планування проектів в умовах обмежених ресурсів. Це дозволить не тільки підвищити ефективність управління проектами, але й забезпечити більш точне та гнучке використання ресурсів, відкриваючи нові можливості для оптимізації проектних процесів.

Для досягнення цієї мети ставляться наступні задачі:

1. Аналіз існуючих методів РОПП.
2. Розробка методу прогнозування рангів конфігурацій.
3. Алгоритм прогнозування рангів конфігурацій для РОПП.
4. Оцінка продуктивності розробленого методу.
5. Аналіз та порівняння розробленого методу.
6. Практичне використання отриманих результатів.

Об'єктом дослідження є процеси ресурсно-обмеженого планування проектів (РОПП) в різних організаційних та проектних середовищах. Це включає вивчення складних проектів, що мають різні обмеження ресурсів, а також різноманітні методи та підходи до планування та управління цими ресурсами. Фокус зосереджується на аналізі, як традиційні методи планування взаємодіють із сучасними вимогами управління проектами, а також на оцінці можливостей для вдосконалення та оптимізації існуючих практик.

Предметом дослідження є розробка та валідація нового методу прогнозування рангів конфігурацій для РОПП, що дозволяє автоматично визначати найбільш ефективні стратегії планування в умовах обмежених ресурсів. Це включає розробку алгоритмів, використання статистичних та машинного навчання методів для оцінки та ранжування різних конфігурацій планування, аналіз їх ефективності та надійності у різних сценаріях, а також дослідження можливостей інтеграції цих методів у існуючі системи управління проектами.

Методи дослідження у даній роботі включають комплексний підхід, поєднуючи теоретичний аналіз, емпіричне моделювання та експериментальну верифікацію. Використовуються кількісні методи для аналізу даних проектів, включаючи статистичний аналіз та машинне навчання, з метою розробки та тестування моделі прогнозування рангів конфігурацій. Аналіз проектних даних та визначення найкращих практик управління ресурсами здійснюються за допомогою алгоритмічних підходів та методів оптимізації. Також проводяться експерименти на реальних та симульованих даних проектів, щоб перевірити ефективність запропонованих методів у різних умовах і контекстах управління проектами.

Наукова новизна отриманих результатів полягає у розробці інноваційного методу прогнозування рангів конфігурацій для ресурсно-обмеженого планування проектів, який інтегрує принципи машинного навчання з традиційними методами управління проектами. Цей підхід відрізняється застосуванням передових алгоритмів регресійного аналізу для автоматизації вибору оптимальних конфігурацій управління ресурсами, забезпечуючи значне підвищення точності та ефективності у порівнянні з існуючими методами. Окрім того, цей метод дозволяє гнучко адаптуватися до різних типів проектів та умов їх реалізації, що відкриває нові перспективи для досліджень у галузі управління проектами та розширює можливості їх практичного застосування.

Практичне значення одержаних результатів у цій роботі полягає в значному поліпшенні ефективності управління ресурсами в проектах різного масштабу та складності. Розроблений метод прогнозування рангів конфігурацій дозволяє проектним менеджерам швидко та точно ідентифікувати найбільш оптимальні стратегії планування, що адаптовані до конкретних умов та вимог проекту. Це знижує ризики затримок та збільшення витрат, пов'язаних з неефективним розподілом ресурсів, та

підвищує загальну продуктивність проекту. Крім того, автоматизація процесу вибору конфігурацій значно зменшує потребу в ручному аналізі та оцінці, тим самим звільняючи час та ресурси для зосередження на більш стратегічних аспектах управління проектом. Отже, цей метод є важливим інструментом для підвищення загальної ефективності управління проектами, особливо в умовах, де ресурси обмежені та потребують ретельного планування.

Структура та обсяг роботи. Робота складається із вступу, трьох розділів, висновків, списку використаних джерел та додатків. Повний обсяг роботи становить 94 сторінок комп'ютерного тексту, який включає 4 рисунків та 10 таблиць. Список використаних джерел із 61 найменувань викладено на 6 сторінках.

Апробація результатів дослідження. Основні теоретичні положення роботи й практичні результати дослідження доповідалися й обговорювалися на IV Всеукраїнської студентської наукової конференції, м. Чернігів, 29 вересня, 2023 рік Україна та IV Всеукраїнська студентська конференція «Розвиток сучасної науки: актуальні питання теорії та практики» 17 листопада 2023, м. Львів, Україна.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ

1.1 Суть та особливості ресурсно-обмеженого планування проектів

Проблема планування проекту з обмеженими ресурсами (РОПП) є добре відомою та обширно вивченою темою дослідження протягом багатьох десятиліть. Базова версія РОПП передбачає, що один проект складається з дій, які повинні бути заплановані з урахуванням обмежень послідовності та ресурсів так, щоб мінімізувати момент завершення. Дії пов'язані між собою за допомогою відносин завершення-початку з мінімальним відставанням часу нуль, і не можуть бути перервані. Метою РОПП є знаходження графіка, який мінімізує загальну тривалість проекту, відому як момент завершення проекту.

Ресурсно-обмежене планування проектів - це проблема оптимізації, яка виникає при розподілі ресурсів для виконання завдань у межах обмежень часу та ресурсів. Основна мета полягає в тому, щоб мінімізувати час виконання проекту (makespan), враховуючи обмеженість доступних ресурсів. Давайте розглянемо основні аспекти та характеристики цієї проблеми:

1. Активності та Залежності:

- Проект розбивається на конкретні активності, які повинні бути виконані.
- Існує порядок виконання активностей, визначений відносинами завершення-початку.

2. Ресурси:

- Є обмежені ресурси, такі як працівники, обладнання, матеріали і т. д.
- Кожна активність вимагає певні ресурси для виконання.

3. Обмеження Ресурсів:

- Можливість використання ресурсів обмежена, і виникають ситуації, коли більше однієї активності потребує доступу до обмежених ресурсів одночасно.

4. Часові Обмеження:

- Кожна активність має обмежений час виконання, визначений відносинами та іншими факторами.

5. Мета:

- Основна мета - мінімізувати час виконання всього проекту (makespan).

6. NP-важкість:

- Проблема є NP-важкою в сильному сенсі, що означає, що немає відомого поліноміального алгоритму для знаходження точного рішення за поліноміальний час для будь-якого випадку проблеми.

7. Методи Розв'язання:

- Для розв'язання РОПП використовуються різні методи, такі як методи розгалуження та обмеження (branch-and-bound), еволюційні алгоритми, генетичні алгоритми, табу-пошук та інші.

8. Емпіричні Дослідження:

- Дослідження в області РОПП часто включає аналіз взаємозв'язку між характеристиками проекту та висновками алгоритмів для планування.

9. Предиктивні Моделі:

- Останнім часом в дослідженнях використовуються предиктивні моделі для передбачення ефективності алгоритмів в залежності від параметрів проекту.

10. Використання Ресурсів:

- РОПП враховує оптимальне використання ресурсів, допомагаючи підприємствам максимізувати продуктивність та ефективність.

РОПП залишається однією з ключових проблем у галузі оптимізації та планування проектів, і вивчення ефективних методів її розв'язання є актуальним завданням для науковців та практиків.

Як узагальнення проблеми планування робочого місця (job-shop scheduling problem), РОПП є NP-важкою в сильному сенсі [2]. Тому техніка розгалуження та обмеження (B&B) є найбільш поширеним способом вирішення цієї проблеми для отримання оптимальних рішень [4]. Деякі алгоритми розгалуження та обмеження можна знайти в [55] та [12]. У [19] запропонували потужну схему глибокого розгалуження з новими критеріями домінування та правилами обмеження. Розширення відповідного коду представлено в [21]. У [33] представили дві метастратегії для обчислення нижніх меж для РОПП. У [53] ввів метод розгалуження та обмеження, який ґрунтується на так званій схемі інкрементації дій, що ітеративно розкладає придатні дії в частковому графіку. У [14] розробили процедуру розгалуження та обмеження, яка враховує всі найкращі компоненти з літератури. Ця так звана композитна процедура розгалуження та обмеження використовує 48 різних конфігурацій, які є комбінаціями 3 різних схем розгалуження, 2 стратегій пошуку, 1 стратегії графа, 4 композитних нижніх меж та 2 порядки розгалуження.

У цьому дослідженні буде запропонований новий підхід до розв'язання, який автоматично визначає ефективність 48 конфігурацій, а потім використовує кращі компоненти найкращої конфігурації для вирішення проекту на основі характеристик проекту (тобто показників проекту). Таким чином, метою цього дослідження є прогнозування рейтингового списку для 48 конфігурацій згідно з якістю рішень,

генерованих комбінованими компонентами в процедурі розгалуження та обмеження на основі відомих параметрів мережі та ресурсів екземплярів проектів і перевірка можливості використання такого рейтингового списку для вирішення проблеми краще, ніж середня ефективність кожної конфігурації окремо.

Дослідження взаємозв'язку між характеристиками екземплярів проекту та ефективністю процедур розв'язання для планування не є новинкою в літературі. У [42] пропонується автоматичний вибір алгоритму для проблеми планування проекту з багатьма режимами використання ресурсів (М РОПП), що ґрунтується на концепції емпіричних моделей складності для відображення характеристик екземпляра проблеми на ефективність алгоритму. Зокрема, об'єднуються два різні алгоритми (алгоритм заборони та гібридний генетичний алгоритм) в супер-алгоритм, який працює краще, ніж будь-який з компонентів індивідуально. У [45] вводиться деревоподібний підхід для класифікації та визначення найкращого пріоритетного правила для проблеми планування проекту з обмеженими ресурсами, і їхнє дослідження ґрунтується на двох моделях класифікації для відображення показників проекту на ефективність пріоритетного правила. Незважаючи на ці зусилля, вважається, що немає досліджень, які обирають найкращі компоненти точного алгоритму для вирішення РОПП до (най)оптимальності. Це, наскільки відомо, перше дослідження, яке досліджує взаємозв'язок між показниками проекту та рейтинговим списком ефективності різних конфігурацій процедури розгалуження та обмеження за допомогою моделі передбачення на основі регресії. Крім того, порівнюється ефективність двох різних моделей передбачення та досліджується їхній потенціал у п'яти різних експериментах. Наприкінці, на основі результатів п'яти обчислювальних експериментів буде

стверджено, що запропонований підхід є перспективним для розширення на інші проблеми планування проектів.

Узагальнюючи викладене, ресурсно-обмежене планування проектів є складною оптимізаційною проблемою, де завдання розподілу ресурсів в обмежених умовах часу вимагає мінімізації часу виконання проекту. Зазначені особливості, такі як активності та їх взаємозв'язки, обмежені ресурси, часові обмеження та NP-важкість, визначають складність цієї задачі. Дослідження в області РОПП включає в себе застосування різноманітних методів розв'язання, від методів розгалуження та обмеження до еволюційних алгоритмів. Даний контекст також висвітлює важливість взаємозв'язку між характеристиками проекту та ефективністю алгоритмів. В цьому контексті запропонований підхід, який враховує автоматичний вибір ефективних конфігурацій для розв'язання РОПП на основі характеристик проекту, вказує на перспективність вдосконалення підходів до ресурсно-обмеженого планування проектів.

1.2 Огляд існуючих підходів

Проблему планування проекту з обмеженими ресурсами можна сформулювати наступним чином: Набір дій N , пронумерованих від фіктивного стартового вузла 0 до фіктивного кінцевого вузла $n + 1$, призначається без переважного впливу на множину R видів відновлюваних ресурсів. Кожен поновлюваний ресурс $k \in R$ має постійну доступність k за період. Кожна нефіктивна діяльність $i \in N$ має детерміновану тривалість d_i і вимагає $r_{i,k}$, одиниць ресурсного типу $k \in R$. Початок і кінець фіктивних дій 0 і $n + 1$ являють собою початок і завершення проекту, при якому їх тривалість і поновлення потреба в ресурсі, що дорівнює нулю. Проектна мережа представлена у форматі топологічної впорядкованої активності на вузлі (AoN), де A — множина

пар активностей між який існує зв'язок пріоритету фінішу-старту з часовим лагом 0. Припускаємо, що граф $G(N,A)$ є ациклічним. Графік S визначається вектором часу початку діяльності і вважається здійсненим, якщо всі пріоритети та обмеження відновлюваних ресурсів задоволені. Мета задачного типу полягає в тому, щоб знайти здійснений графік в рамках найменшого можливого обсягу проекту, і, отже, тип задачі може зображуватися у вигляді $m, 1T|c_{pm}|C_{max}$ з використанням класифікаційної схеми [29] або як $PS|prec|C_{max}$ [4].

Для ознайомлення зі стандартом РОПП вказані джерела [4], [28], [35]. З метою досягнення оптимальності для РКПСР розроблено розгалужені процедури, які розглядались раніше. Ці процедури з розгалуженнями та зв'язками базуються на швидких та ефективних нижніх межах. В [33] проведено обчислення нижніх меж для цієї складної проблеми та вказано, що відсутність сильних нижніх меж ускладнює використання точних алгоритмів для завдань значного обсягу. Згідно з [46], більшість точних алгоритмів з літератури застосовуються для вирішення менших завдань з меншою кількістю елементів (менше 60 дій), тоді як евристичні процедури є альтернативою для розв'язання великих завдань. Піддавшись впливу цих досліджень, [14] недавно внесли різні нижні межі в добре зарекомендовані процедури В&В з літератури для вирішення РКПСР, сподіваючись, що це сприятиме вирішенню великих екземплярів. Автори продемонстрували, що їхня композитна стратегія нижніх меж покращує існуючі рішення та генерує нові оптимальні рішення для існуючих наборів даних. Незважаючи на це, більшість великих екземплярів проектів залишаються нерозв'язаними до оптимального стану. Тим не менш, ця розгалужена процедура, наскільки відомо, є найповнішою серед процедур, яка об'єднує найкращі компоненти з літератури, що стало причиною її використання в

поточному дослідженні. Нижче пояснюються різні обрані компоненти для розробки 48 конфігурацій.

У дослідженні [15] великий набір компонентів був об'єднаний для створення різних конфігурацій процедур розгалуження та зв'язку. Однак, оскільки результати, отримані за декількома конфігураціями, майже ідентичні, було вирішено не включати всі можливі конфігурації в дослідження. Замість цього, було здійснено відбір найефективніших конфігурацій з оригінального дослідження, що призвело до створення 48 конфігурацій, включаючи 1 деревоподібну стратегію, 2 стратегії пошуку, 3 схеми розгалуження, 4 складені нижні межі та 2 порядки розгалуження. Огляд чотирьох основних компонентів, які використовуються в нашому алгоритмі В&В, наведено в таблиці Б (див. Додаток Б). Ці компоненти коротко обговорюються в наступному розділі.

Деревоподібна стратегія: Ця деревоподібна стратегія визначає вибір вузлів для розгалуження від пошуку до оптимального рішення. Стратегія дерева глибини починається з кореневого вузла дерева і вибирає вузол на основі деяких критеріїв (наприклад, найкраща нижня межа) для подальшого дослідження на наступному рівні дерева. Отже, пошук досліджує дерево якомога далі вздовж кожної гілки, перш ніж виконувати будь-яке зворотне відстеження. Натомість, стратегія найкращого першого дерева досліджує дерево, вибираючи найперспективніший вузол відповідно до вказаного правила як наступний вузол для розгалуження, незалежно від рівня, на якому вузол створений. На відміну від стратегії дерева «глибина першою», вузли можуть бути обрані на будь-якому рівні дерева в стратегії «найкраще-перше дерево», і пошук не обов'язково переходить відразу до глибоких рівнів дерева. Натомість, вибраний вузол може бути досліджений на певному рівні дерева, а потім новий вузол на вищому рівні може бути обраний для подальшого дослідження, що зазвичай призводить до широкого, але

спочатку не дуже глибокого дерева. Оскільки результати показали, що стратегія «перше дерево глибини» перевершує стратегію «найкраще перше дерево», остання не враховується в нашому дослідженні.

Стратегія пошуку: Використовуються два різних способи реалізації стратегії дерева «глибина перша», запропонована [51], а саме стратегія верхньої межі (U) і стратегія мінімальної нижньої межі (L).

Схема розгалуження: Ця схема визначає спосіб побудови вузлів на кожному рівні дерева. У цьому дослідженні реалізовано три різні схеми розгалуження, відомі як розгалуження часу початку діяльності (A), послідовне розгалуження (S) і паралельне розгалуження (P).

Складена стратегія нижньої межі, запропонована в роботі [14], спрямована на поєднання швидкості та якості відомих нижніх меж у єдину стратегію. Конкретно, ця стратегія створює перелік можливих нижніх меж, які можуть бути використані в рамках процедури, і динамічно вирішує, чи використовувати кожну нижню межу в кожному вузлі дерева. Для досягнення цієї мети автори використали кредитну систему, яка автоматично вмикає або вимикає деякі нижні межі на основі певних критеріїв, щоб уникнути значного обчислювального часу при використанні вдосконалених стратегій нижньої межі. У вихідному дослідженні тестувалися різні стратегії з використанням від 1 до 13 нижніх меж, і обрано чотири ефективні композитні стратегії нижньої межі (CLB) (використовуючи ідентифікатор CLB, який був використаний в оригінальному дослідженні для позначення кількості використаних нижніх меж у стратегії). Кожна конфігурація відповідає можливій комбінації компонентів, які будуть детально розглянуті надалі.

- CLB0: Ця стратегія обчислює лише нижню межу критичного шляху на основі (LB_{ср}) і, отже, є найшвидшим способом отримати нижнє граничне значення в кожному вузлі дерева.

- CLB4: Ця стратегія нижньої межі визначається як CLB0 плюс нижня межа критичної ємності (LB_{ss} LB_{ss}), нижня межа критичної послідовності (LB_{cs}), нижня межа несумісних пар (LB_{ip0} LB_{ip0}) і базова версія нижньої межі упаковки вузлів (LB_{np0}).
- CLB8: Ця стратегія нижньої межі визначається як CLB4 плюс два розширення нижньої межі упаковки вузла (LB_{np1} і LB_{np2}), одна нижня межа на основі паралельної машини (LB_{pm1}) і нижня межа несумісних триплетів (LB_{ip1}).
- CLB12: Ця стратегія нижньої межі визначається як CLB8 плюс розширена версія нижньої межі машини (LB_{pm2}), нижня межа, знайдена шляхом *редукції за пріоритетом* (LB_{pr}), нижня межа, знайдена скороченням за часом ядра (LB_{ct}), і нижня межа, знайдена скороченням за *періодами часу* (LB_{tp}). Останні три нижні межі є так званими деструктивними нижніми межами, які ітеративно намагаються довести, що не існує можливого рішення для заданого заздалегідь визначеного значення makespan проекту M , і якщо доказ успішний, $LB_{M} + 1$ є дійсним нижнім граничним значенням. Цей ітеративний пошук продовжується зі збільшенням значень M до тих пір, поки не буде надано жодного успішного доведення, що призведе до остаточного нижнього граничного значення тривалості проекту. LB_{pm2} також заснований на релаксації РОПП до спеціальної задачі планування паралельних машин, в той час як він обмежує кількість *паралельних машин* лише підмножиною з 4 можливостей. LB_{pr} спирається на ідею LB_{ip0} і послідовно тимчасово фіксує відношення пріоритету для заданого значення нижньої межі. LB_{ct} визначає основний час кожного виду діяльності як інтервал між останнім часом початку та найранішим часом закінчення, враховуючи заздалегідь визначене значення тривалості проекту. В основі LB_{tp}

лежить адаптація нижньої межі ресурсної ємності для того, щоб використовувати її в контексті руйнівного поліпшення.

Стислий виклад різних компонентів, що використовуються в нашій процедурі, наведено в таблиці Б (Додаток Б) з посиланням на оригінальну статтю, де пропонується кожен компонент. Для наочності література, що стосується кожної окремої нижньої межі, що використовується в CLB, наведена в таблиці 1.1, і зацікавлені читачі можуть знайти більше інформації в списку літератури таблиці. Технічні деталі кожної складеної нижньої межі виходять за рамки цієї роботи, і читач посилається на оригінальне дослідження [14] для отримання більш детальної інформації про чотири складені стратегії нижньої межі.

Таблиця 1.1. Посилання, пов'язані з кожною нижньою межею, що використовується в CLB.

Нижня межа	Посилання
LBcp	[51]
LBcs	[55]
LBcc	[14]
LBnp0, LBnp1, LBnp2	[44]
LBip1, LBip0, LBpm1, LBpm2, LBpr, LBct , LBtp	[33]

Порядок розгалуження: Після того, як вузол обраний для розгалуження в дереві В&В, ранжування його дочірніх вузлів також має бути побудовано на основі деяких критеріїв. Це ранжування визначає, який дочірній вузол буде обраний першим для розгалуження. Для визначення порядку розгалуження дерева використовуються дві версії цього вибору вузла, тобто найкраща нижня межа (В) та порядок ідентифікаторів активності (А).

Різні комбінації для "Стратегії пошуку", "Схеми розгалуження", "Порядку розгалуження" та "Складеної нижньої межі" призводять до 48 конфігурацій, як показано внизу таблиці Б (Додаток Б). У подальшій

частині цієї роботи така конфігурація буде скорочена так, як показано в другій колонці таблиці. Наприклад, конфігурація "LPA12" відноситься до пошуку з розгалуженням, в якому стратегія пошуку дорівнює стратегії мінімальної нижньої межі (L), схема розгалуження встановлена на паралельне розгалуження (P), порядок ідентифікаторів активності (A) використовується зі складеною стратегією нижньої межі CLB12 (12).

Оскільки дане дослідження спрямоване на автоматичне формування рейтингового списку для 48 конфігурацій алгоритму B&B на основі значень для відомих показників проекту за допомогою моделі прогнозування, необхідно визначити перелік показників активності, мережі та ресурсів. Подібно до [27], використовуються 12 показників, пов'язаних з активністю, мережею та ресурсними аспектами кожного екземпляра. Проектні показники, використані у даному дослідженні, узагальнені в таблиці 3. Зауважимо, що ресурсокористування та ресурсний фактор тісно пов'язані, але використовуються обидва, оскільки вони добре відомі та широко використовуються в науковій літературі. Використання ресурсів (RU) вимірює, скільки ресурсів використовується діяльністю, тоді як ресурсний фактор (RF) вимірює середню частину ресурсів, що використовуються діяльністю. Незважаючи на схожість цих двох показників, вони не вимірюють одне й те саме. Наприклад, у [20] RU використовується для гарантії того, що кожен вид діяльності використовує принаймні один із чотирьох ресурсів (тобто $RU = 1$), а потім з цією гарантією значення RF змінюються від низьких до високих значень.

Таблиця 1.2. Показники активності, мережі та ресурсів для екземпляра проекту.

Показник	Визначення	Література
Діяльність		
Кількість активностей (#Act)	Кількість нефіктивних вузлів у мережі.	-
Мережа		

Коефіцієнт складності мережі (ЧПУ)	Сумарне число відношень пріоритету (дуг) над загальною кількістю проектних активностей (вузлів) в мережі.	[49]
Сила замовлення (OC)	Число відносин пріоритетності (включаючи прямих і непрямих попередників) ділиться на теоретичне максимальне число відношення пріоритетності.	[41]
Послідовний/паралельний (SP)	Вимірювання близькості проектної мережі до повної послідовної (СП = 1) або паралельної (СП = 0) мережі.	[58]
Розподіл активності (AD)	Розподіл проектної діяльності за рівнями проекту.	[58]
Довжина дуг (LA)	Наявність коротких дуг з різницею між поступальним рівнем кінцевого вузла і стартовим вузлом кожної дуги, рівною 1.	[58]
Топологічна плаваюча кома (TF)	Топологічна плаваюча точка відношення пріоритету як кількість рівнів, які може змістити кожна діяльність, не порушуючи максимального рівня мережі (як визначено SP).	[58]
Довжина довгих дуг I 5 5	Аналогічно LA, але вимірює наявність довгих дуг з різницею між прогресивним рівнем кінцевого вузла і початковим вузлом кожної дуги більше 1.	[58]
Ресурс		
Ресурсний фактор (RF)	Середня частка типів ресурсів, запитуваних за один вид діяльності.	[49]
Використання ресурсів (рос.)	Кількість типів ресурсів, що використовуються кожним видом діяльності.	[41]
Ресурсний потенціал (RS)	Поеднання інформації топології мережі та попиту на ресурси, деталі можна знайти в довідниках, досі спірних.	[16]
Обмеженість ресурсів (RC)	Середня потреба в ресурсах для всіх видів діяльності для конкретного ресурсу поділена на доступність цього ресурсу.	[51]

Отже, це дослідження сприяє подальшому розвитку області планування проектів з обмеженими ресурсами, надаючи новий підхід до вирішення проблеми РОПП та використання передбачувальних моделей для оптимізації конфігурацій процедур розв'язання.

1.3 Постановка задачі

РОПП стоїть перед низкою викликів, які вимагають удосконалення інструментів та методів для досягнення оптимальних рішень. Зростаюча складність проектів, пов'язана з обмеженнями ресурсів, вимагає нових підходів до планування та вибору конфігурацій для досягнення максимальної ефективності та економії ресурсів. Існуючі методи не завжди здатні ефективно враховувати усі варіації умов проекту, в

результаті чого може виникати потреба в ручній корекції планів та виборі оптимальних конфігурацій.

Особливо важливою є потреба в автоматизації процесу вибору конфігурацій, оскільки ручне аналіз та вибір оптимального методу планування стає непрактичним у великих та складних проектах. Розробка методу прогнозування рангів конфігурацій для РОПП вирішить цю проблему, дозволяючи автоматично визначати ефективні методи планування на основі аналізу умов проекту та історичних даних.

Запропонований метод буде актуальним не лише для наукового дослідження, а й для практичного використання в сферах, де критично важливим є ефективне ресурсно-обмежене планування проектів, таких як будівництво, виробництво та ІТ-проекти. Його впровадження дозволить підвищити автоматизацію та оптимізацію процесів планування проектів, зменшити витрати ресурсів і часу, тим самим покращуючи загальну продуктивність та конкурентоспроможність підприємств.

Метою даного дослідження є розробка методу прогнозування рангів конфігурацій для РОПП, який дозволить автоматично визначати ефективні методи планування проектів в умовах обмежених ресурсів.

Для досягнення цієї мети ставляться наступні задачі:

7. Аналіз існуючих методів РОПП: Провести глибокий аналіз існуючих методів ресурсно-обмеженого планування проектів з метою ідентифікації їхніх переваг та недоліків. Визначити ключові властивості проектів, на які не враховано в існуючих методах.
8. Розробка методу прогнозування рангів конфігурацій: Розробити метод, який базується на аналізі історичних даних проектів та умовах їх виконання, для прогнозування ефективності різних конфігурацій. Врахувати різноманітність проектів та їхні умови для створення універсального методу.

9. Алгоритм прогнозування рангів конфігурацій для РОПП: Розробити алгоритм, який буде враховувати прогнозовані ранги конфігурацій і допомагати при автоматичному виборі ефективних методів планування для конкретного проекту. Забезпечити можливість адаптації алгоритму до різних умов та обставин.
10. Оцінка продуктивності розробленого методу: Провести експериментальну оцінку продуктивності розробленого методу, використовуючи реальні та синтетичні дані проектів. Порівняти його результати з існуючими методами, визначити переваги та можливі напрями подальшого вдосконалення.
11. Аналіз та порівняння розробленого методу: Порівняти розроблений метод з існуючими підходами до ресурсно-обмеженого планування проектів, визначити його ефективність, надійність та застосовність в різних областях. Здійснити аналіз впливу різних умов проекту на результати методу.
12. Практичне використання отриманих результатів: Визначити можливості та обмеження практичного використання розробленого методу в реальних проектах. Розглянути перспективи впровадження методу в сферу ресурсно-обмеженого планування проектів і визначити його потенційний внесок у вирішення відповідних завдань.

В розділі 2 "Метод прогнозування рангів конфігурацій для РОПП" визначається задача розробки самого методу прогнозування. У підрозділі 2.2 "Метод прогнозування рангів конфігурацій для РОПП" конкретизуються основні принципи та аспекти розробленого методу. Алгоритм прогнозування рангів конфігурацій описується у підрозділі 2.3 "Алгоритм прогнозування рангів конфігурацій для РОПП". В цьому параграфі визначається конкретний спосіб визначення рангів та

прогнозування їхніх ефективностей для вибору оптимальних конфігурацій у завданнях РОПП.

Висновки до розділу 1

1. Аналіз предметної області ресурсно-обмеженого планування проектів вказав на важливість вдосконалення методів управління проектами в умовах обмежених ресурсів. Сутність та особливості цього виду планування ставлять перед вченими та практиками великі завдання у забезпеченні ефективного використання ресурсів при виконанні проектів.
2. Ресурсно-обмежене планування проектів визначається поєднанням низки складних факторів, таких як постійна доступність відновлюваних ресурсів, тривалість дій, та інші параметри. Висвітлення цих особливостей дозволяє краще розуміти виклики, які стоять перед вченими та практиками у галузі планування проектів.
3. Огляд існуючих підходів вказує на необхідність подальшого розвитку методів ресурсно-обмеженого планування проектів. Постановка завдань визначає конкретні шляхи розв'язання цих викликів, включаючи розробку методів прогнозування рангів конфігурацій для поліпшення вибору оптимальних стратегій управління проектами.

2 МЕТОД ПРОГНОЗУВАННЯ РАНГІВ КОНФІГУРАЦІЙ ДЛЯ РОПП

2.1. Прогнозування рангів в задачах ресурсно-обмеженого планування проектів

У сучасних дослідженнях штучного інтелекту значно посилилась увага до вивчення пріоритетності. Основним завданням є вивчення сценарію навчання, відомого як ранжування етикеток, яке передбачає встановлення відношень між атрибутами та ранжуванням у визначеному наборі етикеток, згідно з працями [26] та [31]. Ця задача застосовується у таких сферах, як аналіз візрців, аналіз настроїв, цільова реклама, рекомендаційні системи та метанавчання, завдяки своїй універсальності у відображенні інформації про пріоритетність або релевантність між різними об'єктами.

Задача ранжування етикеток полягає у навчанні відображенню від вектора ознак x_i у просторі ознак χ до ранжування $\succ x_i$, яке представляє сукупні строгі порядки над скінченним набором етикеток $\{y_1, y_2, \dots, y_{iQ}\}$ у просторі міток Y , де $y_{iq} \in Y$ і $q=1, 2, \dots, Q$, $|Q|$ позначає кількість усіх можливих міток. При цьому $y_{ij} \succ x_i y_{ik}$ інтерпретується як перевага етикетки y_{ij} над y_{ik} для екземпляра x_i . Кожен екземпляр $x_i \in \chi$ представлений вектором F -вимірних ознак $(x_{i1}, x_{i2}, \dots, x_{iF})$, де F - це кількість ознак. Ранжування серед міток Q можна виразити через перестановку σ_{x_i} , а простір усіх перестановок G_Q називається цільовим простором.

Досліджувана задача ранжування етикеток є складною задачею прогнозування, метою якої є автоматичне ранжування продуктивності різних компонентів процедур В&В для вирішення РОПП за допомогою нового структурованого методу прогнозування. Ця модель навчається на великому наборі прикладів проектів, щоб передбачити рейтинг для 48 конфігурацій процедур В&В. Передбачуваний рейтинг надає інформацію про переваги стосовно якості меж, які створюються кожним компонентом

для конкретного прикладу проекту. Моделі прогнозування базуються на $|F|=12$ індикаторах проекту, що характеризують екземпляр проекту, та $|Q|=48$ конфігураціях, які представляють етикетки у прогностичній моделі. Простір ознак χ та простір усіх перестановок G_Q визначаються на основі обчислювальних експериментів на великій навчально-валідаційній множині, що слугує вхідними даними для моделей прогнозування. Оцінка моделі відбувається на тестовій вибірці для визначення остаточної продуктивності прогностичних моделей.

У даному дослідженні вирішується складна задача ранжування етикеток у контексті ресурсно-обмеженого планування проектів. Використовуючи концепцію відношень між атрибутами та ранжуванням, автори впроваджують новий структурований метод прогнозування для автоматичного ранжування ефективності компонентів процедур B&B у вирішенні РОПП. Модель, навчена на великому наборі прикладів проектів, визначає рейтинг для 48 конфігурацій B&B, що вказує на переваги кожного компонента. Застосування цільового простору та простору ознак, обраного на основі обчислювальних експериментів, дозволяє ефективно моделювати пріоритетність. Підсумкова оцінка моделі проводиться на тестовій вибірці, що визначає остаточну продуктивність прогностичних моделей у задачі ресурсно-обмеженого планування проектів.

2.2. Метод прогнозування рангів конфігурацій для РОПП

У цьому розділі представлено інноваційний підхід до прогнозування ранжування для 48 конфігурацій процедури вітки та межі, як описано у розділі 1.2. Структура системи прогнозування, ілюстрована на рисунку 2.1, включає чотири ключові фази, що розглядаються у подальших параграфах.

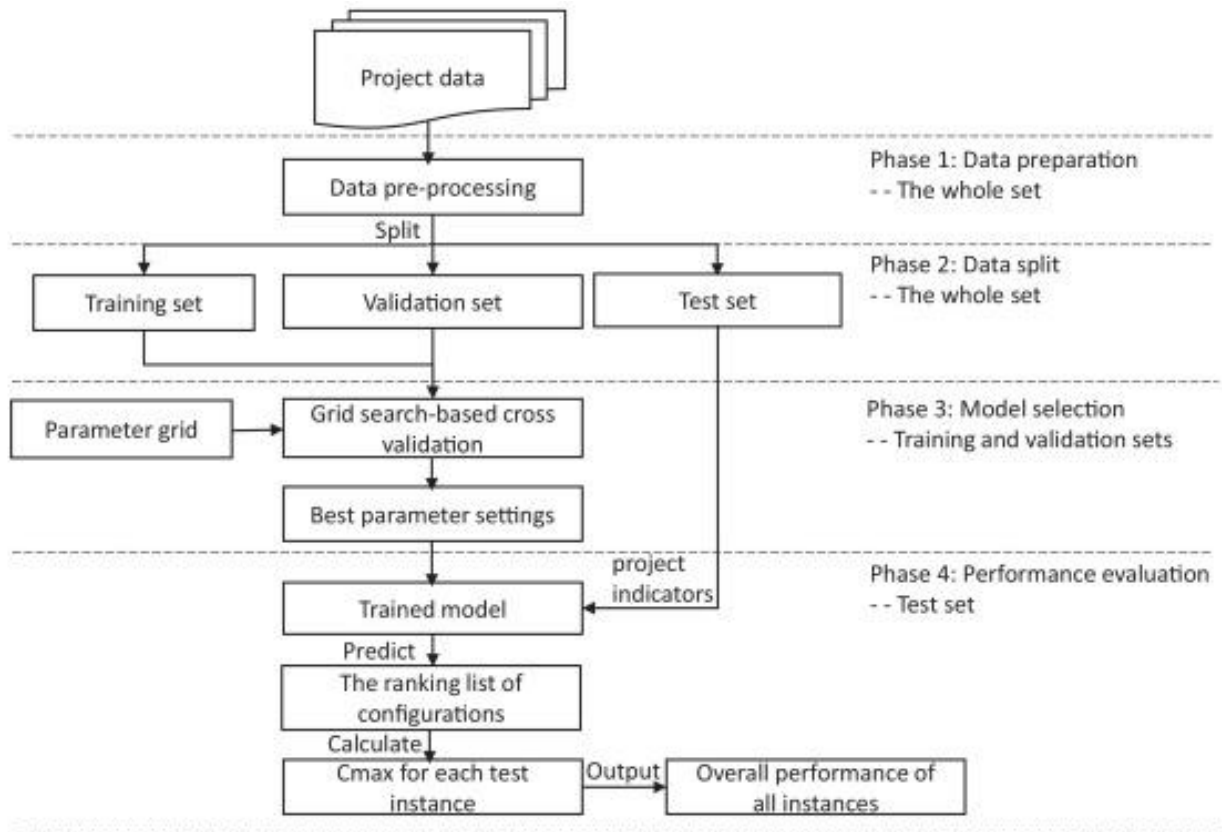


Рисунок 2.1. Структура методу прогнозування рангів конфігурацій для РОПП.

Етап 1. Підготовка даних: Ця фаза зосереджена на ініціалізації даних для проведення експериментів, яка включає наступні кроки:

1. Збір результатів від 48 конфігурацій для кожного екземпляра.
2. Створення рейтингового списку з цих конфігурацій для кожного екземпляра.
3. Вилучення екземплярів, які класифіковано як "легкі", з набору даних.
4. Вибір кількості індикаторів проекту, що використовуються в експериментах.
5. Нормалізація даних.

Рішення для кожного екземпляра отримуються шляхом запуску процедур В&В для всіх конфігурацій з встановленим часовим лімітом. "Легкі" екземпляри визначаються, як ті, для яких рішення від 48

конфігурацій є ідентичними в межах цього часового ліміту. Для кожного екземпляра визначено 12 індикаторів проекту (згідно з розділом 2.1), пов'язаних з активністю, мережею та ресурсами, і ці індикатори масштабуються до діапазону $[0, 1]$ за допомогою методу мінімального та максимального значення:

$$V' = \frac{V - \min(x)}{\max(x) - \min(x)},$$

де V — початкове значення індикатора проекту для екземпляра, V' — нормалізоване значення, $\min(x)$ і $\max(x)$ — мінімальні та максимальні значення індикатора проекту у всіх екземплярах набору даних.

Етап 2. Розбиття даних: Друга фаза полягає у розділенні даних на тренувальну, валідаційну та тестову множини. Тренувальна множина використовується для навчання моделі, встановлюючи зв'язок між індикаторами проекту (вхідні дані) та рейтинговим списком для 48 конфігурацій процедури В&В (вихідні дані). Для кожного екземпляра в тренувальній множині формується рейтинговий список з урахуванням критерію якості, який включає оцінки нижньої та верхньої меж (LB і UB). У випадку однакових результатів від різних конфігурацій застосовується метод випадкового розриву зв'язків.

Етап 3. Вибір моделі: На третьому етапі визначається взаємозв'язок між ознаками та рейтинговим списком на тренувальній множині, а валідаційна множина використовується для вибору оптимальних параметрів моделі за допомогою методу перехресної перевірки на основі пошуку по сітці. Цей процес дозволяє уникнути перенавчання та налаштовує модель для досягнення найкращого можливого результату.

Етап 4. Оцінка ефективності: Остання фаза спрямована на оцінку продуктивності моделі на тестовій множині, яка є невидимою для моделі до цього моменту. Після завершення етапів навчання та валідації, вибрана комбінація параметрів, яка забезпечує найкращі результати,

використовується для повторного навчання моделі на об'єднаних тренувальній та валідаційній множинах. Потім ця модель застосовується до тестової множини для оцінки її остаточної продуктивності, де для кожного екземпляра обчислюється 12 значень індикаторів проекту, і ці дані вводяться в модель для отримання рейтингового списку з 48 конфігурацій. За допомогою цього списку кожен екземпляр у тестовій множині розв'язується шляхом послідовного виклику конфігурацій зі списку ранжування, при цьому кожна конфігурація використовується згідно з встановленим часовим лімітом.

Одним із ключових аспектів даного дослідження є визначення оптимальної кількості конфігурацій, які слід включити в аналіз, щоб уникнути необхідності розв'язування кожного екземпляра за допомогою всіх 48 конфігурацій. Визначення цієї кількості міток для включення в прогнозований рейтинговий список здійснювалось на основі фактичної продуктивності 48 конфігурацій на тестових екземплярах. Конкретніше, тестували кожен конфігурацію на всіх екземплярах тестового набору, фіксуючи кількість випадків, коли певна конфігурація генерувала найкращий знайдений LB (нижня межа) або UB (верхня межа). Отримане число потім усереднювалось для всіх тестових екземплярів, формуючи середнє число істинно позитивних міток (AvgTPL).

Це середнє значення AvgTPL використовувалося як критерій для визначення кількості конфігурацій у прогнозованому рейтинговому списку, обмежуючи його першими AvgTPL конфігураціями. Після виконання процедури розгалуження та зв'язування для цих перших конфігурацій, найвищий LB або найнижчий UB реєструвалися як найкращий знайдений makespan для кожного тестового екземпляра. Загальна продуктивність моделі оцінювалася на основі цих результатів, і процес завершувався на цьому етапі. Цей експеримент був проведений на тестовій вибірці.

Важливо зазначити, що такий підхід має обмеження у реальному світі, оскільки продуктивність всіх 48 конфігурацій для будь-якого нового невидимого проекту зазвичай є невідомою заздалегідь. Проте, в контексті перевірки продуктивності прогностичних моделей, використання AvgTPL як методу для обмеження кількості випробувань розгалуження та межі виявляється ефективним способом зменшення обсягу пошуку. Додаткова інформація та деталі цього підходу представлені в розділах, присвячених результатам обчислень.

У даному розділі представлено інноваційний метод прогнозування рангів конфігурацій для ресурсно-обмеженого планування проектів. Структура системи прогнозування включає чотири ключові етапи: підготовку даних, розбиття даних, вибір моделі та оцінку ефективності. Перший етап включає збір результатів, створення рейтингового списку конфігурацій та нормалізацію даних. Другий етап передбачає розділення даних на тренувальну, валідаційну та тестову множини. На третьому етапі визначається взаємозв'язок між ознаками та рейтинговим списком, а вибір оптимальних параметрів моделі проводиться на валідаційній множині. На завершальному етапі оцінюється продуктивність моделі на тестовій множині. Важливим аспектом є визначення оптимальної кількості конфігурацій для аналізу, що використовується для уникнення надмірного обчислювального навантаження. Дослідження вказує на ефективність методу в умовах обмеженої інформації про продуктивність конфігурацій для нових проектів у реальних умовах ресурсно-обмеженого планування проектів.

2.3. Алгоритм прогнозування рангів конфігурацій для РОПП

Цей розділ коротко описує деталі реалізації процесів тренування, валідації та тестування алгоритмів, запропонованих у даному

дослідженні. Реалізація включає в себе п'ять різних етапів, кожен з яких пов'язаний із використанням комп'ютерної платформи та деталей кодування.

Етап 1: Виконання процедури розгалуження та обмеження (B&B) на еталонних екземплярах. Для генерації рейтингового списку всіх експериментальних даних, алгоритм розгалуження та обмеження необхідно запустити на всіх екземплярах для 48 конфігурацій. При цьому кожен екземпляр проекту i з набору даних має 48 рішень (тобто нижню межу та $makespan$, припустимий з точки зору ресурсів), які служать як мітки $Q, y_{i1}, y_{i2}, \dots, y_{iQ}$ на наступних етапах.

Етап 2: Генерація маркованих експериментальних даних. Рішення (мітки) навчальних та валідаційних наборів, отримані на Етапі 1, тепер використовуються для створення повного рейтингового списку для кожного екземпляру проекту. Ранжування $\succ_{x_i}$ визначається як перестановка σ_{x_i} , яка відображає кожну мітку $y_{ij} \in Y$ (де $|Q|=48$) екземпляру i у її ранг. Відповідно, $y_{ij} \succ_{x_i} y_{ik}$ означає, що для i -го екземпляру з ознаками x_i , мітка y_{ij} є кращою, ніж мітка y_{ik} , що відображається в перестановці як $\sigma_{x_i}(y_{ij}) < \sigma_{x_i}(y_{ik})$, де $\sigma_{x_i}(y_{ij})$ є позицією y_{ij} у рейтинговому списку. Таким чином, на цьому етапі створюється перестановка σ_{x_i} із усіх можливих перестановок SQ (де $\sigma_{x_i} \in SQ$) для кожного екземпляру i з відомим набором ознак $x_i \in X$.

Етап 3: Створення моделі структурованого прогнозування. На основі маркованих даних з Етапу 2 метою тепер є навчання взаємозв'язку між простором ознак X та простором всіх рейтингових списків (тобто простором всіх перестановок SQ) з навчальних даних. Для вирішення цієї задачі ранжування міток використовуємо ідеї [36], які розглядають задачу ранжування міток як завдання структурованого виведення регресії, використовуючи підхід заміщення найменшого квадратичного втрати. Автори ділять це навчальне завдання на два підетапи та використовують

метод перехресної перевірки на основі пошуку сітки [37] для поділу навчального набору на два окремих менших набори. Перший набір є меншим навчальним набором для Підетапу 3.1, тоді як решта екземплярів належить до валідаційного набору, який використовується на Підетапі 3.2.

Підетап 3.1. Фаза регресії. Фаза регресії виконується на меншому навчальному наборі та складається з наступних підфаз:

Підфаза кодування: Наше завдання навчання полягає у відображенні індикаторів проекту (простір ознак) на рейтинговий список 48 конфігурацій (структурований вихідний простір). Однак, оскільки це не прямолінійна задача, в даному дослідженні застосовується методологія, запропонована в роботі [36], яка пропонує вирішення цієї задачі за допомогою вкладення Лемера (відомого також як вектор інверсій, [39]). Вкладення Лемера - це особливий спосіб перетворення перестановок σ_{xi} будь-якого екземпляру $\sigma_{xi} \in SQ$ у їх код Лемера $c\sigma_{xi}$ у просторі CQ (які були вкладені в простір Гільберта F), де кожна координата приймає значення незалежно від інших координат. Після цього перетворення всі перестановки представляються їх кодом Лемера (тобто процес кодування).

У так званому просторі Гільберта F , відображення між ознаками та кодом Лемера можна навчитися за допомогою двох методів регресії, використаних у цьому дослідженні. Відомо, що код Лемера є взаємно однозначним, а процеси кодування та декодування мають лінійну складність [36] [40]; [47]. Крім того, код Лемера також дозволяє тривіальне розв'язання підетапу прогнозування 3.2 завдяки його незалежним координатам (тобто взаємно однозначним).

Підфаза навчання: Тепер використовуються два методи регресії - k-Nearest Neighbors (kNN) та ядерна регресія гребеня (ridge), щоб навчитися відносин між ознаками та кодом Лемера попередньої підфази. Метод k-

Nearest Neighbors вимірює схожість між зразками (екземплярами проекту) відповідно до певного методу вимірювання відстані та заздалегідь визначеної кількості сусідів. У нашому конкретному випадку використовується евклідова відстань як міра відстані, а розмір околиці k тестується з 11 різними значеннями пізніше у цій роботі.

Підетап 3.2. Фаза прогнозування. Фаза прогнозування здійснюється на валідаційному наборі і складається з наступних підфаз:

Підфаза оцінювання: Для будь-якого екземпляра валідаційного набору з відомими значеннями ознак можна спрогнозувати код Лемера, використовуючи результати з двох методів регресії.

Підфаза декодування: Отримані прогнози декодуються у вихідний простір перестановок SQ, використовуючи процес декодування коду Лемера. Отриманим результатом є рейтинговий список для кожного екземпляру в валідаційному наборі, який потім використовується для оцінки ефективності методів регресії з різними налаштуваннями параметрів.

Оптимальне відношення між ознаками та рейтинговим списком усіх міток із найкращими значеннями параметрів можна отримати лише після набору ітеративних виконань підетапів 3.1 та 3.2 для всіх можливих комбінацій значень параметрів моделі γ , λ та k .

Етап 4: Прогнозування рейтингового списку для тестового набору. Враховуючи засвоєні зв'язки та оптимальні параметри, знайдені на попередніх етапах, процедура тепер готова спрогнозувати новий рейтинговий список міток для будь-якого невидимого екземпляра. На основі відомого набору ознак x_i будь-якого невидимого екземпляра i , модель прогнозування передбачить новий рейтинговий список з 48 конфігурацій. Цей прогноз проводиться для всіх екземплярів тестового набору окремо, що означає, що рейтингові списки можуть відрізнятися між екземплярами тестового набору. Отже, Етап 4 використовує ту ж

процедуру, що й Підетап 3.2, але тепер з екземплярами тестового набору (замість екземплярів валідаційного набору) як вхідними даними.

Етап 5: Виконання процедури розгалуження та обмеження на тестовому наборі. На цьому останньому етапі рейтинговий список кожного тестового екземпляра використовується для запуску процедури розгалуження та обмеження з метою повідомлення нижніх та верхніх меж. Звісно, коли перераховується повний рейтинг, процедура розгалуження та обмеження буде використовуватися для всіх 48 конфігурацій, а тоді повідомлене рішення буде дорівнювати рішенню найкращої конфігурації. Таким чином, для запуску процедури розгалуження та обмеження будуть використовуватися лише перші конфігурації AvgTPL, щоб дослідити, чи наближається повідомлене рішення до рішення найкращої конфігурації цього екземпляра. У розділі обчислювальних результатів будуть повідомлені результати для різних значень AvgTPL.

У даному розділі було представлено алгоритм прогнозування рангів конфігурацій для ресурсно-обмеженого планування проектів. Процес включав п'ять ключових етапів, розглянуті у деталях. На перших етапах проводилася процедура розгалуження та обмеження (B&B) для генерації рейтингового списку експериментальних даних. Далі, з використанням цих даних, генерувалися марковані дані для навчання моделі структурованого прогнозування, що визначала взаємозв'язок між ознаками та рейтинговим списком. Оптимальні параметри моделі визначалися шляхом подальших етапів, таких як розділення даних та вибір моделі. На завершальних етапах модель застосовувалася для прогнозування рейтингових списків для тестового набору, і процедура розгалуження та обмеження використовувалася для перевірки результатів на тестових екземплярах. Цей підхід має потенціал вдосконалити продуктивність ресурсно-обмеженого планування проектів, а обрані етапи та методи навчання моделі структурованого прогнозування

відображають системний та комплексний підхід до розв'язання задачі прогнозування рангів конфігурацій у РОПП.

Висновки до розділу 2

1. У розділі представлено новаторський підхід до вирішення проблеми прогнозування рангів конфігурацій у задачах ресурсно-обмеженого планування проектів. Зазначено, що структура системи прогнозування включає чотири ключові фази, кожна з яких детально розглядається у подальших підрозділах. Етап підготовки даних визначається ініціалізацією експериментальних даних, збором результатів від 48 конфігурацій для кожного екземпляра, та нормалізацією даних, що є ключовими етапами у створенні рейтингового списку.
2. Другий етап, розбиття даних, визначає тренувальну, валідаційну та тестову множини для використання у моделюванні та перевірці прогнозів. На третьому етапі, вибір моделі, визначається взаємозв'язок між ознаками та рейтинговим списком на тренувальній множині, з використанням валідаційної множини для вибору оптимальних параметрів моделі. Цей етап підкреслює важливість налагодження моделі для досягнення оптимальних результатів.
3. У розділі також описано четвертий етап - оцінка ефективності моделі на тестовій множині. Після завершення етапів тренування та валідації, обрана комбінація параметрів моделі використовується для прогнозування рейтингових списків для тестових екземплярів. Цей етап важливий для остаточної оцінки продуктивності моделі та перевірки її роботи на реальних тестових умовах.
4. Усі ці етапи узагальнюються в системний підхід до прогнозування рангів конфігурацій для РОПП, що поєднує в собі етапи підготовки даних, моделювання та валідації, а також ефективну оцінку продуктивності на тестовій множині.

З ПРАКТИЧНЕ ВИКОРИСТАННЯ ОТРИМАНИХ НАУКОВИХ РЕЗУЛЬТАТІВ

3.1. Оцінка продуктивності структурованої моделі прогнозування

Для оцінки продуктивності структурованої моделі прогнозування в цьому дослідженні використовуються три базові набори даних, які загалом містять 3,143 проектних екземпляра. Огляд використаних у дослідженні наборів даних представлений у Таблиці 4. Перший набір даних - це добре відомий набір даних PSPLIB [54], який містить чотири різні піднабори (J30, J60, J90 та J120) з 30, 60, 90 та 120 активностями відповідно. Другий набір даних - це RG300, запропонований [18], який містить екземпляри проектів з 300 активностями. Третій набір даних - це набір CV, нещодавно запропонований [15], який містить лише 623 складних для вирішення екземплярів.

Багато з цих екземплярів проектів дуже складні, якщо не неможливі, для вирішення до оптимальності. Наприклад, набір CV складається з екземплярів, які не можна було вирішити до оптимальності навіть після 20 годин обчислювального часу. Тому обмежили часовий ліміт до трьох різних значень, а саме 1 секунда, 60 секунд та 1 година [14], як показано у Таблиці 3.1.

Таблиця 3.1. Набір даних, використаний у цьому дослідженні.

Набір даних	Піднабір	#Екземплярів	#Складних (LB UB)		
			1 с	60 с	1 год
PSPLIB	J30	480	227 262	219 241	207 226
	J60	480	169 311	176 291	174 267
	J90	480	132 354	135 305	135 288
	J120	600	384 600	387 595	391 594
RG300		480	204 454	214 454	222 454
CV		623	513 623	604 623	618 623
Загалом		3,143			

У таблиці 3.1 кількість екземплярів кожного набору даних (#Екземплярів) додатково поділяється на "легкі" та "складні" екземпляри, використовуючи дуже простий підхід, що використовує час обчислення як вхід. Починаючи з моменту, коли значення НМ або ВМ, що генеруються 48 конфігураціями, обмежуються після 1 секунди (1 с), 60 секунд (60 с) або 1 години (1 г), такі екземпляри назовемо "легкими". У випадку, якщо значення не обмежуються таким чином, екземпляр вважається "складним". Кількість "складних" екземплярів (#Складних) показана для кожного набору даних для часового ліміту 1 с або 60 с або 1 г відповідно. Ця класифікація легких/складних не є дуже детальною, але все ж показує відсоток екземплярів для кожного набору даних, які можуть виграти від вибору правильної конфігурації (для "легких" екземплярів усі конфігурації працюють однаково, і тому правильний вибір не можна зробити). У експериментах 2-5 ці "легкі" екземпляри видаляються, і лише "складні" екземпляри зберігаються як експериментальні дані. Таблиця 3.1 показує, що екземпляри CV містять найвищий відсоток складних екземплярів, що, звісно, є нормальним, оскільки ці екземпляри генеруються спеціально для того, щоб їх не можна було вирішити до оптимальності за допомогою сучасних процедур. Однак два інші набори даних все ще містять значну кількість екземплярів, для яких правильний вибір конфігурації може бути корисним.

Подібно до [8], три набори даних поділяються на 80% для навчання та 20% для тестування, а потім використовується підхід крос-валідації на основі сіткового пошуку [37] для налаштування гіперпараметрів моделей на основі набору даних для навчання. Як було зазначено раніше, набір даних для навчання розділяється на маленький набір даних для навчання та набір даних для валідації для кожного складу в крос-валідації, набір даних для валідації використовується для визначення оптимальних параметрів, щоб уникнути перенавчання.

Для оцінки ефективності прогностичних моделей та порівняння їх продуктивності з різними конфігураціями процедури розгалуження та обмеження (B&B) використовуються чотири показники оцінювання. Ці метрики порівнюють згенеровані рішення з найкращим знайденим нижнім кордоном (BFLB) та найкращим знайденим верхнім кордоном (BFUB). Ці кордони є найкращими можливими рішеннями (найвищий LB або найнижчий UB), знайденими всіма 48 конфігураціями процедури B&B і виступають як найкращі відомі рішення у дослідженні. Нижче наведено визначення чотирьох показників оцінювання:

1. Кумулятивний нижній кордон (CLB) вимірює якість нижнього кордону, згенерованого певним методом рішення для конкретного екземпляра тестового набору. Метод рішення може бути однією конфігурацією процедури B&B або результатом прогностичної моделі, обмеженої після конфігурації AvgTPL. Аналогічно до підходу, описаного у [10], обчислюється сума нижніх кордонів для всіх проектів у тестовому наборі, а не середні значення LB.
2. Кумулятивний верхній кордон (CUB) вимірює якість верхнього кордону, згенерованого певним методом рішення – однією конфігурацією або найкращим знайденим makespan прогностичної моделі – для конкретного екземпляра тестового набору. Значення знову повідомляються як сума всіх знайдених makespans для всіх проектів у наборі даних.
3. Кумулятивне абсолютне відхилення від BFLB (ADev) обчислює відхилення найкращого LB, знайденого певним методом рішення, від найкращого знайденого нижнього кордону BFLB для кожного екземпляра, а потім бере суму цих відхилень для всіх тестових екземплярів. Менше значення означає кращу продуктивність.
4. Кумулятивне абсолютне відхилення від BFUB (ADev') обчислює відхилення найкращого UB, знайденого певним методом рішення,

від найкращого знайденого верхнього кордону BFUB для кожного екземпляра, а потім бере суму цих відхилень для всіх тестових екземплярів. Менше значення означає кращу продуктивність.

Отже, ефективність структурованої моделі прогнозування оцінюється на трьох різних наборах даних: PSPLIB, RG300 та CV, включаючи в сумі 3,143 проектних екземпляри. Під час оцінки використовуються три різні часові ліміти: 1 секунда, 60 секунд та 1 година. Набір даних CV відзначається високим відсотком складних екземплярів, спеціально створених для випробувань моделей. Методологія використовує підхід крос-валідації та сіткового пошуку для налаштування гіперпараметрів моделей. Для оцінки ефективності використовуються чотири показники, включаючи кумулятивний нижній та верхній кордони, а також кумулятивне абсолютне відхилення від найкращих знайдених кордонів. Взагалі, дані дозволяють зробити висновок про продуктивність моделі та порівняти її з іншими конфігураціями процедури розгалуження та обмеження (B&B) в різних умовах.

3.2. Аналіз продуктивності прогностичних моделей

У наступних підрозділах наведено результати п'яти різних обчислювальних експериментів. У підрозділі 3.2.1 досліджується продуктивність двох моделей у чотирьох різних випадках і представлені всі оптимальні налаштування параметрів двох методів регресії. У підрозділі 3.2.2 досліджено вплив обмеження на максимальну кількість міток і детально розглянуто зв'язок між ознаками та мітками. Крім того, в підрозділі 3.2.3 тестується продуктивність моделей для різних підгруп з 48 конфігурацій, а в підрозділі 3.2.4 досліджується робастність моделей на наборах даних з різною кількістю видів діяльності. Нарешті, в

підрозділі 3.2.5 порівнюються моделі з іншими підходами шляхом розбиття часу за обраними конфігураціями, а також підтверджується працездатність моделей шляхом порівняння їх з літературними процедурами.

3.2.1. Експеримент 1. Налаштування параметрів

Цей підрозділ описує результати деяких експериментів, проведених для налаштування параметрів запропонованого методу (Розділ 3). Спочатку досліджено вплив обмеження часу, яке використовується у процедурі розгалуження та обмеження (V&V), а потім наведено результати точного налаштування параметрів для двох регресійних методів.

У всіх обчислювальних експериментах необхідно мати результати, як нижні, так і верхні кордони, для 48 конфігурацій усіх тестових екземплярів. Ці результати були отримані шляхом виконання всіх конфігурацій з обмеженим часовим лімітом 1 секунда (1с) або 60 секунд (60с). Ці результати були використані під час навчання та тестування чотирма різними способами, як описано нижче.

- Випадок 1: Результати в навчальному та тестовому наборах були отримані на основі обмеження часу 1 секунда.
- Випадок 2: Результати в навчальному та тестовому наборах були отримані на основі однакового обмеження часу 60 секунд.
- Випадок 3: Результати в навчальному та тестовому наборах були отримані на основі різних обмежень часу. Навчання базується на швидких результатах, отриманих протягом 1 секунди, але будь-який новий тестовий екземпляр розв'язується за допомогою конфігурації розгалуження та обмеження, обмеженої лише 60 секундами.

- Випадок 4: Цей випадок схожий на Випадок 3, оскільки результати в навчальному та тестовому наборах базуються на різних обмеженнях часу. Однак, навчання тепер базується на результатах, отриманих протягом 60 секунд, в той час як кожна конфігурація для тестових екземплярів розв'язується лише за 1 секунду обчислювального часу.

Результати цих тестів представлені у Таблиці 3.3. Порівнюється продуктивність двох прогностичних моделей (ridge і kNN) з окремою найкращою конфігурацією, а також з найкращою конфігурацією C для кожного індивідуального екземпляра. Одна найкраща конфігурація є середньою найкращою конфігурацією для всіх тестових екземплярів і служить як найгірше рішення порівняно з прогностичною моделлю. Однак конфігурація C, в свою чергу, вибирає найкраще рішення з усіх 48 конфігурацій для кожного тестового екземпляра і діє як найкраще рішення, оскільки прогностична модель ніколи не може генерувати краще рішення, ніж C.

Таблиця 3.3. Порівняння продуктивності в чотирьох випадках.

		CLB	ADev		CUB	ADev'
Випадок 1	C	109,251	0	C	120,884	0
	ridge	109,246	5	ridge	120,947	63
	kNN	109,246	5	kNN	120,921	37
Випадок 4	ridge	109,246	5	ridge	120,928	44
	kNN	109,246	5	kNN	120,906	22
Одна найкраща	LPA12	109,200	51	UAA4	125,964	5,080
Випадок 3	C	109,560	0	C	120,166	0
	ridge	109,560	0	ridge	120,225	59
	kNN	109,560	0	kNN	120,247	81
Випадок 2	ridge	109,560	0	ridge	120,187	21
	kNN	109,560	0	kNN	120,187	21
Одна найкраща	LPA12	109,536	24	UAA12	125,518	5,352

Перш за все, результати показують, що прогностичні моделі завжди перевершують одну найкращу конфігурацію та наближаються до рішення C у всіх чотирьох випадках. Це особливо цікаво для Випадку 4, у якому модель навчена на хороших рішеннях (отриманих після 60 секунд), але

будь-який новий екземпляр (тестовий набір) вирішується швидко (обмежений 1 секундою), що доводить, що прогностичний список ранжування може бути використаний для швидкого генерування рішень після дуже обмеженого часу.

Звичайно, таблиця також показує, що найкращі результати отримуються, коли навчання та тестування проводяться за найбільшого обмеження часу 60 секунд (Випадок 2). Результати, отримані двома регресійними методами, рівні тим, які були отримані оптимальним випадком C для розрахунків нижнього кордону. Проте різниця між результатами, наданими однією найкращою конфігурацією LPA12 та C , становить лише 24 ($= 109,560 - 109,536$), і тому, незважаючи на те, що можливі покращення невеликі, прогностична модель знайшла їх. Для розрахунків верхнього кордону результати більш виразні. Різниця між однією найкращою конфігурацією та C становить 5,352, що вказує на значно більший простір для поліпшень. Прогностична модель змогла знайти ці поліпшення, оскільки їх якість рішень покращилася до 120,187, значення близьке до C , зі значенням A_{Dev} лише 21. Оскільки модель працює найкраще при обмеженні 60 секунд для навчання та тестування, Випадок 2 буде використовуватися в подальших експериментах для додаткового дослідження ефективності прогностичних моделей.

У третій фазі, як вказано на рисунку 1, налаштування параметрів здійснюється ітеративно на основі навчального та валідаційного наборів до досягнення оптимальних параметрів. Щоб знайти ці оптимальні параметри для методів kNN та ridge, використовується метод перехресної перевірки з пошуком по сітці. Мета цього методу полягає у створенні гіперпараметричної сітки і виявленні комбінації параметрів з мінімальною помилкою прогнозу [43]. Оскільки кожна з 5-кратних перевірок служить валідаційним набором під час налаштування моделі, можна уникнути проблеми перенавчання. У розділі 3.3 демонструється,

що для методу ridge необхідно налаштовувати два параметри (γ і λ). Використано 12 можливих значень для кожного параметра, згідно з [36], а саме $\gamma, \lambda \in \{10^{-l}, 5 \cdot 10^{-l}\}$, де $l \in \{0, 1, \dots, 5\}$, що формує 144 точки на сітці. Для методу kNN потрібно налаштувати лише один параметр (k), і його 11 можливих значень дорівнюють $k \in \{1, 2, 3, 4, 5, 8, 10, 15, 20, 30, 50\}$. Оптимальні налаштування параметрів для обох методів отримуються за допомогою методу перехресної перевірки на 5 випробуваннях, і результати показані в таблиці 6 для п'яти різних експериментів, як для розрахунків LB, так і UB. Зазначимо, що для кожного експерименту з різною метою кількість "складних" інстанцій також відрізняється, у цьому випадку налаштування параметрів виконується для кожного експерименту для генерації найкращої продуктивності. Тому оптимальні параметри для кожного експерименту не порівнюються.

Оскільки практичне тлумачення двох параметрів (λ і γ) методу регресії з ядром ridge не завжди є зрозумілим, важко вибрати "розумні" значення для них, що призводить до трудомістких пошуків по сітці [24]. З таблиці 3.4 можна спостерігати, що два оптимальні параметри дуже близькі в різних експериментах, що свідчить про те, що вплив цих двох параметрів на модель є незначним, що відповідає висновкам [24]. Більше того, метод може зменшити дисперсію оцінки за рахунок збільшення помилок навчання, тобто чим більше значення γ у гауссівському ядрі, тим швидше функція штрафу знижується до нуля. Велике значення γ призводить до гладкої функції, тоді як мале значення призводить до більш складної функції. Параметр k методу kNN легше зрозуміти, оскільки він вимірює, скільки сусідів буде обрано (з більшими значеннями k , що призводять до більшої кількості сусідів). Однак його значення залежить від даних, що використовуються у навчанні. Наприклад, у деяких випадках (наприклад, для розрахунків LB груп 2 і 3 Експерименту 3) мале значення k достатнє для отримання найкращої продуктивності на

валідаційному наборі для нашого завдання ранжування. Однак у інших випадках (наприклад, для розрахунків LB в Експерименті 1) може знадобитися розглядати більші околиці, щоб отримати найкращу продуктивність. Відповідно до цього, зафіксовані оптимальні значення k виявляли значні відмінності в кожному експерименті.

Таблиця 3.4. Оптимальні налаштування параметрів для різних експериментів.

Експеримент	Пояснення		Подобиці	ridge ($\gamma \lambda$)		kNN(k)	
				LB	UB	LB	UB
Експеримент 1	Перевірка впливу часових обмежень	Випадок 1	Навчальний (1s)/Test(1s)	0.0005 0.01	0.0001 0.005	50	8
		Випадок 2	Навчальний (60s)/Test(60s)	0.0005 0.005	0.0001 0.005	50	4
		Випадок 3	Навчальний (1s)/Test(60s)	0.0005 0.01	0.0001 0.005	50	8
		Випадок 4	Навчальний (60s)/Test(1s)	0.0005 0.005	0.0001 0.005	50	5
Експеримент 2	Перевірка загальної продуктивності		-	0.00001 5	0.0001 0.005	50	15
Експеримент 3	Перевірка вплив чотирьох груп продуктивності	Група 1	Змагальний	0.0001 0.0005	0.00001 1	10	20
		Група 2	Середній	0.005 0.001	0.00001 0.005	4	20
		Група 3	Слабкий	0.00005 0.0005	0.0001 0.0005	8	50
		Група 4	Різноманітний	0.00001 5	0.0001 0.005	50	10
Експеримент 4	Перевірка стійкості моделі на підмножинах	J30	-	0.5 0.5	0.1 0.1	3	10
		J60	-	0.5 0.5	0.1 0.05	8	8
		J90	-	0.01 0.01	0.1 0.05	30	8
		J120	-	0.01 0.001	0.1 0.005	10	20
		CV	-	0.0005 0.0005	0.00005 0.0001	30	30
		RG300	-	0.00001 0.0001	0.001 0.0001	30	5
Експеримент 5	Перевірка здатності моделі до узагальнення шляхом розбиття часового ліміту		-	-	0.0001 0.001	-	8

Отже, проведено серію експериментів для налаштування параметрів запропонованого методу, вивчаючи вплив обмежень часу на процедуру розгалуження та обмеження (B&B). Результати підтверджують, що прогностичні моделі (ridge і kNN) в усіх чотирьох випадках виявляють високу ефективність, наближаючись до найкращих рішень C . Важливо відзначити, що модель, навчена на швидких результатах (60 секунд), здатна генерувати якісні рішення для нових тестових екземплярів, що розв'язуються швидко (1 секунда), що підтверджує можливість використання прогностичного ранжування для ефективного генерування рішень при обмеженому часі. Додатково, оптимальні параметри для регресійних методів були знайдені за допомогою перехресної перевірки з пошуком по сітці, виявивши їхню стабільність та ефективність у відповідних умовах. Висновок дозволяє визначити оптимальні умови для подальшого дослідження ефективності прогностичних моделей у подальших етапах дослідження.

3.2.2. Експеримент 2. Якість прогнозування

Цей підрозділ досліджує продуктивність двох прогностичних моделей і складається з чотирьох частин. По-перше, проводиться порівняння загальної продуктивності методів прогнозування з окремими конфігураціями та оптимальною конфігурацією C . По-друге, досліджується вплив обмеження максимальної кількості міток у рейтинговому списку прогнозів на продуктивність моделі. По-третє, аналізується взаємозв'язок між характеристиками та мітками у прогностичних моделях. Нарешті, подається інформація про обчислювальні часи моделей. Важливо відзначити, що, на відміну від попереднього експерименту, де використовувались всі 3,143 інстанції для

аналізу чотирьох випадків, зараз відібрані лише "складні" інстанції як експериментальні дані (див. розділ 3.1).

Частина 1. Загальна продуктивність. Таблиця 3.5 демонструє порівняння продуктивності різних методів рішення і включає дві прогностичні моделі, одиночну найкращу конфігурацію, а також оптимальну конфігурацію C . Число в дужках після двох моделей показує рейтинг цих моделей і вказує на те, що kNN або ridge займають перше або друге місце. Прогностичні моделі обмежені конфігураціями AvgTPL. На основі обчислювального експерименту для тестового набору, це значення AvgTPL дорівнює 14 для розрахунків LB (ліва частина таблиці) і AvgTPL = 10 для розрахунків UB (права частина таблиці). Таким чином, методи ridge і kNN вирішують процедуру B&B 14 і 10 разів відповідно, для знаходження значень LB і UB. Це, звичайно, більше, ніж кожна одиночна найкраща конфігурація, яка вирішує процедуру B&B лише один раз, але набагато менше, ніж конфігурація C , яка вирішує процедури B&B рівно 48 разів для кожної тестової інстанції. Таблиця ранжована від найкращого рішення C до найгіршого на основі їх середньої продуктивності на тестових інстанціях.

Таблиця 3.5. Порівняння продуктивності різних методів
(ранжування від найкращого до найгіршого).

Ранг	Метод	CLB	ADev	Метод	CUB	ADev'
0	C	66,810	0	C	94,078	0
1	ridge	66,809 (1)	1	kNN	94,149 (1)	71
2	kNN	66,808 (2)	2	ridge	94,215 (2)	137
3	LPB12	66,769	41	UAA12	99,227	5,149
4	LPA12	66,768	42	UAA4	99,316	5,238
5	LPB8	66,702	108	UAA8	99,317	5,239
6	LPA8	66,696	114	UAA0	99,318	5,240
7	LPB4	66,682	128	UAB0	102,533	8,455
8	LPA4	66,677	133	UAB4	102,542	8,464
9	LPB0	66,669	141	UAB8	102,564	8,486
10	LPA0	66,667	143	UAB12	103,321	9,243
11	LSB12	66,208	602	USB0	125,051	30,973
12	LSA12	66,207	603	USB4	125,333	31,255
13	LSB8	66,121	689	USA4	126,525	32,447
14	LSA8	66,121	689	USA0	126,541	32,463

15	LSB4	66,109	701	USB12	128,011	33,933
16	LSA4	66,109	701	USB8	128,173	34,095
17	LSB0	66,090	720	USA8	128,590	34,512
18	LSA0	66,090	720	USA12	130,148	36,070
19	LAB12	66,048	762	UPB12	172,095	78,017
20	LAA12	66,045	765	UPB4	173,038	78,960
21	LAB8	66,034	776	UPB8	173,122	79,044
22	LAB4	66,028	782	UPB0	174,433	80,355
23	LAA8	66,023	787	UPA4	187,280	93,202
24	LAB0	66,022	788	UPA0	187,300	93,222
25	LAA4	66,018	792	UPA8	187,337	93,259
26	LAA0	66,014	796	UPA12	187,415	93,337
27	UPB8	63,938	2,872	LPB12	255,239	161,161
28	UPB4	63,928	2,882	LPB8	259,261	165,183
29	UPB0	63,919	2,891	LPB4	259,552	165,474
30	UPA8	63,864	2,946	LPA12	259,642	165,564
31	UPA0	63,851	2,959	LPB0	259,997	165,919
32	UPA4	63,851	2,959	LSA12	262,496	168,418
33	UPB12	63,840	2,970	LPA8	264,709	170,631
34	UPA12	63,820	2,990	LPA4	265,447	171,369
35	USB12	63,691	3,119	LPA0	266,117	172,039
36	USA12	63,689	3,121	LSB12	269,386	175,308
37	USA4	63,621	3,189	LSA8	283,312	189,234
38	USA8	63,621	3,189	LSA4	283,455	189,377
39	USB4	63,615	3,195	LSA0	284,149	190,071
40	USB8	63,615	3,195	LAB12	286,028	191,950
41	USB0	63,593	3,217	LAB0	286,420	192,342
42	USA0	63,582	3,228	LAB4	286,420	192,342
43	UAA12	63,406	3,404	LAB8	286,420	192,342
44	UAB0	63,374	3,436	LSB8	286,866	192,788
45	UAB4	63,374	3,436	LSB4	286,935	192,857
46	UAA0	63,374	3,436	LSB0	287,515	193,437
47	UAA4	63,374	3,436	LAA12	291,951	197,873
48	UAA8	63,374	3,436	LAA4	292,720	198,642
49	UAB12	63,371	3,439	LAA8	292,720	198,642
50	UAB8	63,366	3,444	LAA0	292,827	198,749
AvgTPL = 14			AvgTPL = 10			

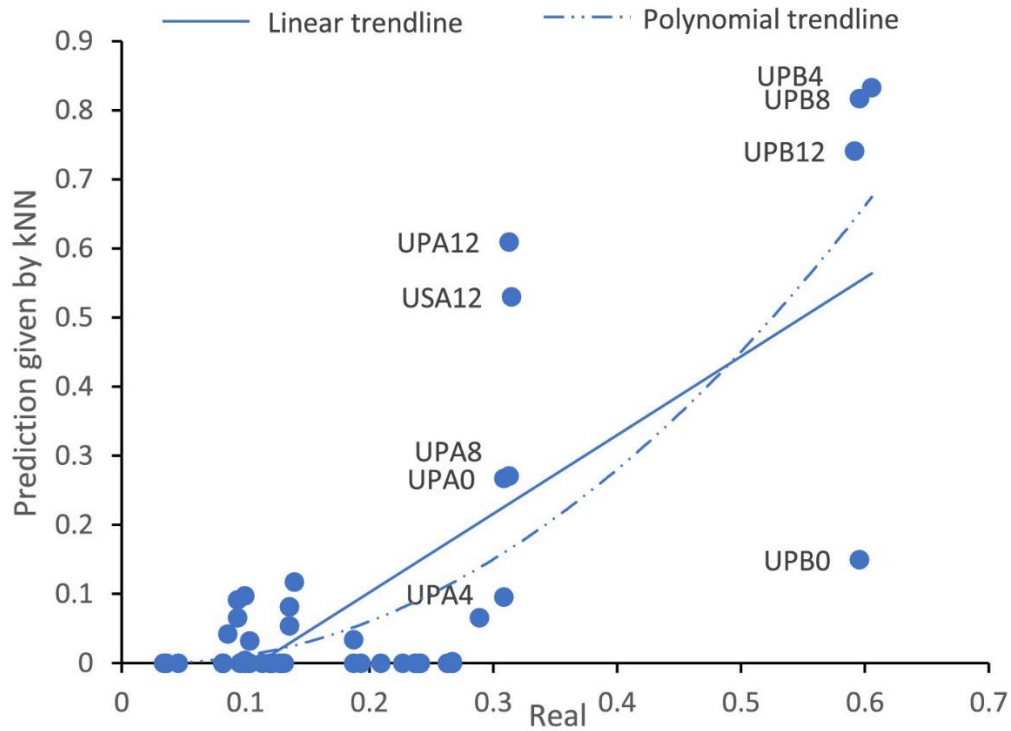
Таблиця 3.5 чітко демонструє, що обидві моделі kNN і ridge перевершують будь-яку одиночну найкращу конфігурацію за чотирма критеріями оцінювання. Одиночна найкраща конфігурація є LPB12 для розрахунків LB, і дві прогностичні моделі показують трохи кращі результати. Більше того, ridge має трохи кращу продуктивність, ніж kNN, оскільки LB, отриманий методом ridge, ближчий до того, що отриманий С. Для розрахунків UB, прогностичні моделі значно покращують будь-яку одиночну найкращу конфігурацію, і тепер метод kNN є значно кращим, ніж ridge, генеруючи результати, які дуже близькі до тих, що

надані С . Для розрахунку нижніх меж ранжування 48 конфігурацій також показує, що стратегія мінімальних нижніх меж призводить до кращої продуктивності, оскільки перші 24 конфігурації для розрахунку значень нижніх меж спираються на цю стратегію пошуку. Аналогічно, для розрахунку верхніх меж процедура, що використовує стратегію верхніх меж, показує кращу продуктивність, оскільки перші 24 конфігурації також спираються на цю стратегію пошуку.

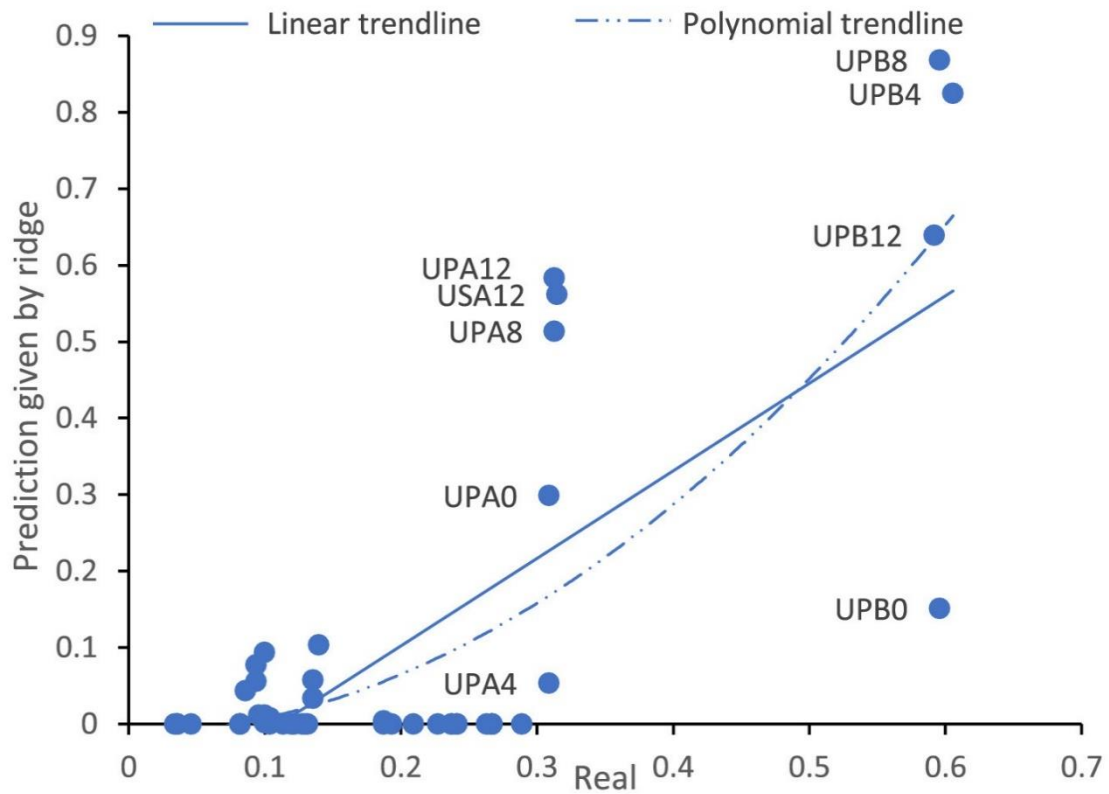
Рисунок 3.1 ілюструє здатність обох методів (kNN та ridge) вибирати найбільш продуктивну конфігурацію для розв'язання задачі РОПП. Графіки демонструють діаграми розсіювання, кожна точка на яких відображає одну конфігурацію для інстанцій у тестовому наборі. Ось Х показує відсоток випадків, коли кожна конфігурація забезпечує найкращий *makespan* для всіх інстанцій у тестовому наборі. Відповідно, ця вісь відображає реальну продуктивність кожної конфігурації, причому вищий показник означає кращу конфігурацію. Графіки показують, що конфігурація UPB (що використовує стратегію пошуку верхньої межі з паралельною схемою розгалуження і порядком розгалуження, заснованим на розрахунках нижніх меж) працює найкраще за чотирма різними стратегіями композитних нижніх меж. Це спостереження узгоджується з результатами, знайденими у [14]. На осі Y вимірюється прогнозована продуктивність як кількість разів, коли конфігурація потрапляє у топ-5 рейтингового списку для методів kNN та ridge (виражено у відсотках від усіх тестових інстанцій). Відповідно, вищий відсоток означає, що ця конфігурація частіше викликається методами kNN або ridge, оскільки вона ранжується у топ-5. В ідеальній системі прогнозування точки повинні розташовуватися на діагональній лінії, що виражає рівні відсотки між реальною та прогнозованою продуктивністю. У такій системі конфігурації низької якості не часто викликаються двома методами прогнозування, тоді як конфігурації високої якості часто ранжуються

високо (у топ-5) прогностичними системами. Рисунок показує, що прогностична здатність обох kNN та ridge дійсно висока, з конфігураціями UPB, що розташовуються в верхньому правому кутку. Отже, конфігурації UPB часто вибираються обома методами прогнозування, що демонструє високу точність обох kNN та ridge. Прогностична продуктивність конфігурації UPB0 є дещо нижчою, ніж у трьох інших конфігурацій UPB, що показує, що обидва методи віддають перевагу більш розвиненим версіям конфігурацій UPB. Графіки також показують, що конфігурації у нижньому лівому кутку не часто викликаються, оскільки їх реальна продуктивність занадто низька. Винятком є конфігурації UPA12, USA12, UPA0, UPA4 та UPA8 (тобто конфігурації зі схемою розгалуження за часом початку активності), які мають досить низьку якість продуктивності, але все ж часто викликаються обома методами. Незважаючи на ці винятки, графіки демонструють відмінну прогностичну продуктивність обох kNN та ridge для вибору правильної конфігурації, тобто конфігурації, яка, як очікується, буде працювати добре для даної інстанції з відомими параметрами мережі та ресурсів.

Частина 2. Обмеження міток. Дана частина досліджує вплив кількості міток, використаних для отримання рішень методами прогнозування, на якість отриманих рішень. У попередніх експериментах було обмежено кількість міток, тобто кількість разів, коли виконується процедура віток і меж (B&B), до середнього числа істинно позитивних міток (AvgTPL). У цьому розділі обмежується кількість міток (#міток) у рейтинговому списку прогнозування від 1 до AvgTPL для вивчення впливу #міток на якість отриманих рішень. Результати цих обмежень представлені на Рис. 3.3a (LB) та Рис. 3.3b (UB), де обмеження відображено на осі X, а якість рішень – на осі Y.

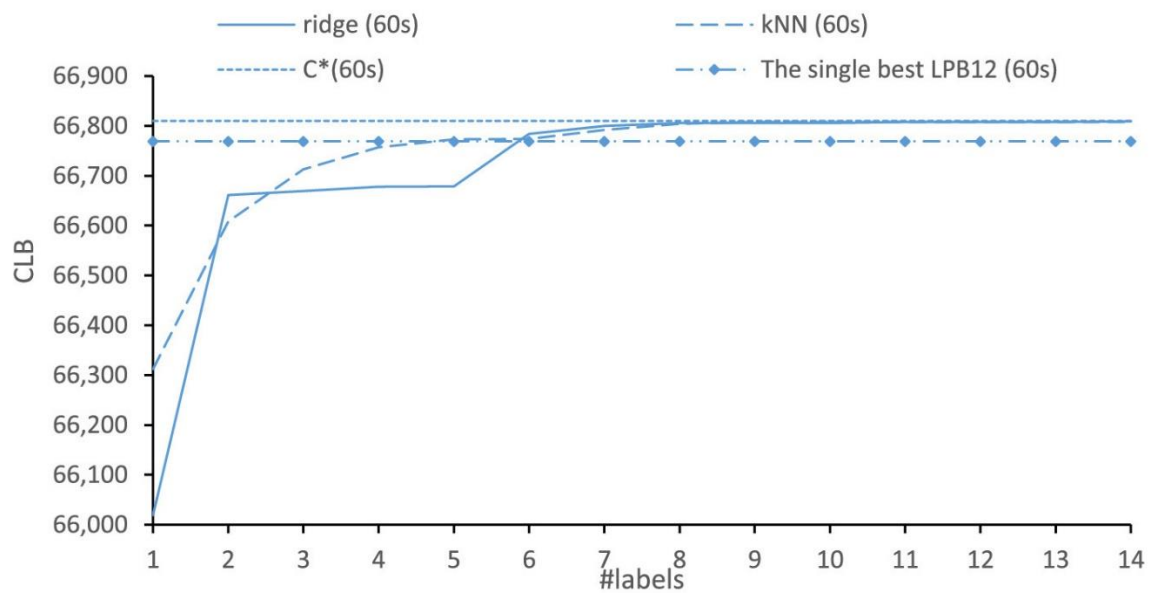


(a) kNN

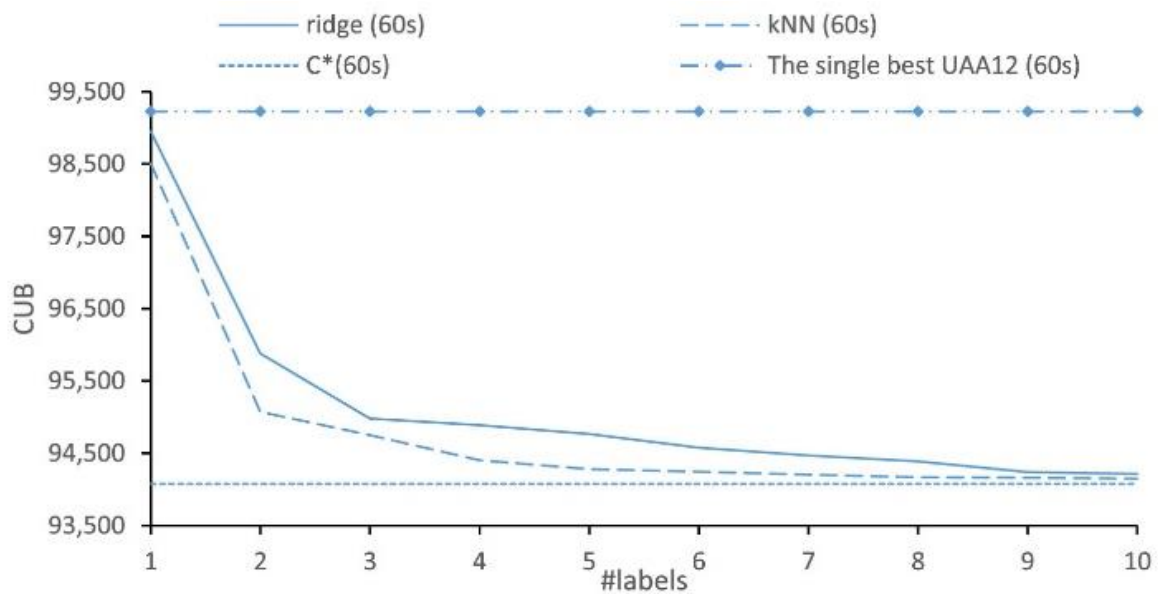


(b) ridge

Рисунок 3.1 - Порівняння реальних та прогнозованих відсотків кожної мітки на тестовому наборі.



(a) LB



(b) UB

Рисунок 3.3 - Вплив обмеження максимальної кількості міток.

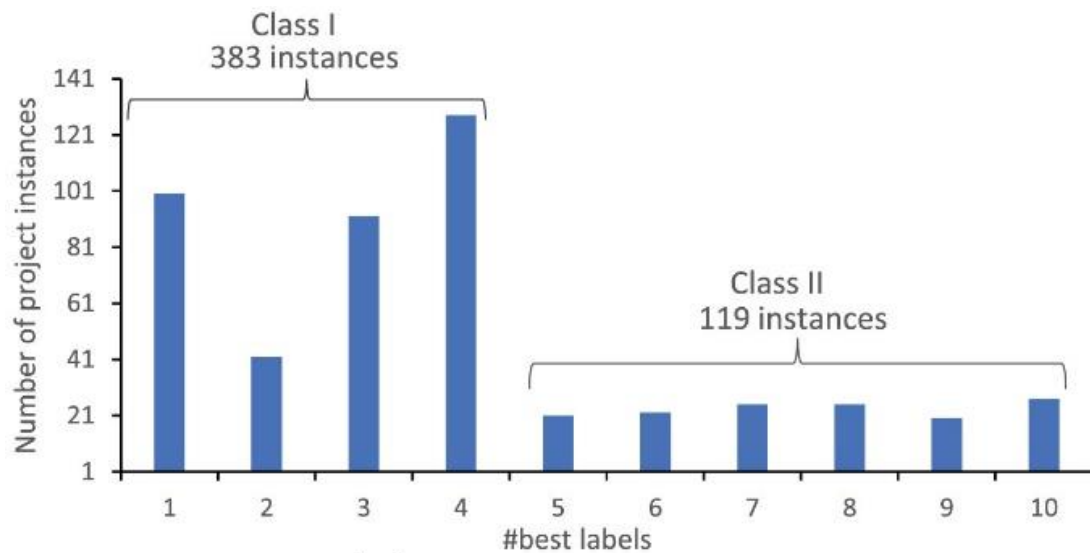
Обидва графіки ясно показують, що зі збільшенням #міток покращується й продуктивність моделі, що не дивно. Однак можна зробити й більш цікаві висновки. По-перше, обидва графіки показують, що метод kNN перевершує метод ridge, коли кількість міток є меншою за AvgTPL. При AvgTPL продуктивність обох прогностичних моделей дещо

схожа (з незначною перевагою для моделі ridge) і дуже близька до конфігурації C. По-друге, результати також показують, що використання лише невеликої кількості міток вже перевершує найкращу одноразову конфігурацію, як для kNN, так і для ridge. Зокрема, прогностичному методу потрібно лише п'ять міток, щоб досягти рішення найкращої одноразової конфігурації LPB12 для LB, та лише одна мітка, щоб досягти найкращої одноразової конфігурації UAA12 для UB. Наприклад, kNN та ridge повідомляють 98,507 та 98,949 для UB-рішень, коли #міток встановлено на 1, що явно перевершує значення 99,227 найкращої одноразової конфігурації UAA12. Таким чином, цей експеримент чітко показує, що прогностична модель здатна вибирати найкращу конфігурацію, і тому перевершує будь-яку одноразову найкращу конфігурацію навіть при встановленні низької кількості міток, що демонструє перспективність моделей прогнозування міток у процедурах V&V для розв'язання РОПП.

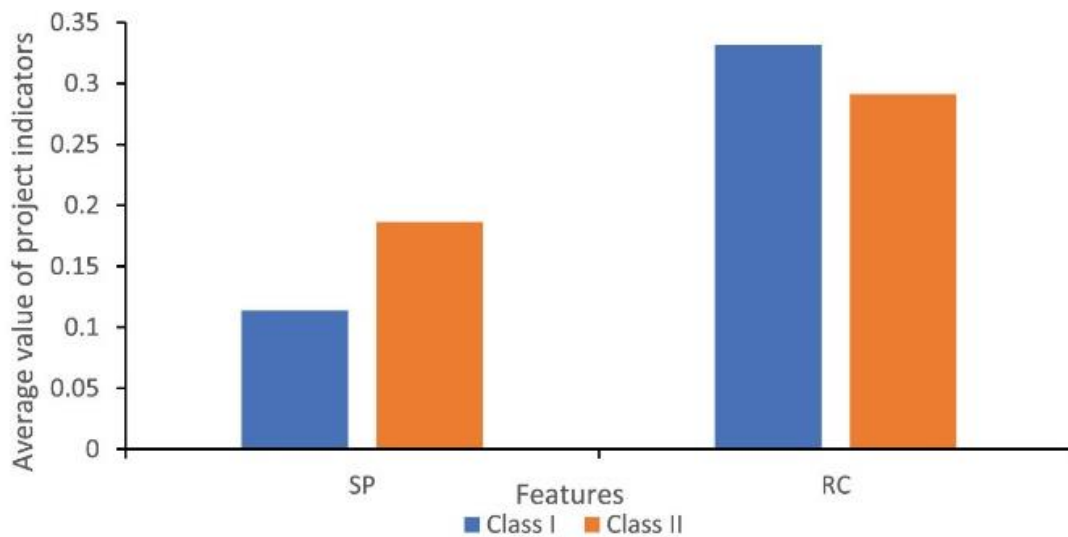
Частина 3. Взаємодія характеристик/міток. З метою виявлення взаємозв'язку між продуктивністю прогностичних моделей (міток) та індикаторами проекту (характеристиками), проведено короткий аналіз кількості найкращих міток, знайдених прогностичними моделями. Зокрема, розраховано для кожного тестового випадку верхню межу (UB) для моделей ridge та kNN, використовуючи обмеження AvgTPL = 10 міток, як обговорювалося раніше. Для кожної конфігурації (або мітки) від 1 до AvgTPL зафіксовано, чи здатна вона генерувати найкращий makespan, знайдений після розв'язання випадку з першими AvgTPL конфігураціями. Відповідно, ця кількість найкращих міток (#найкращих міток) вказує на складність випадку. Справді, коли всі AvgTPL міток вказують на найкраще рішення, це означає, що прогностична модель просто генерує однаковий makespan для 10 конфігурацій, і випадок відносно легко розв'язати. Однак, якщо лише одна або кілька міток дають

найкращий makespan, проблема значно складніша, і ретельний вибір правильних міток важливий для отримання хорошого makespan.

Рисунок 3.4 а показує кількість складних випадків на осі Y для різних значень кількості найкращих міток, які варіюються від 1 до $AvgTPL = 10$ (на осі X). Загальна кількість складних випадків у тестовому наборі становить 502, що складає 20% від 2,509 складних випадків з таблиці 4 для колонки UB. Рисунок розділений на два класи, з 383 випадками у Класі I з максимальною #найкращих міток = 5, та 119 випадками у Класі II з #найкращих міток, що варіюються від 5 до 10. Для кожного з цих випадків відомі індикатор мережі SP та індикатор ресурсів RC, що відображені на Рис. 4 б, і результати підтверджують існуючі знання з літератури. По-перше, випадки Класу I мають нижче середнє значення SP, і відомо, що ці випадки з багатьма паралельними діяльностями важче розв'язати через численні можливі конфлікти ресурсів. Ці результати підтверджуються прогностичними моделями, оскільки кількість найкращих міток є відносно низькою (менше ніж 5), що вказує на важливість правильного вибору гарної мітки (конфігурації). Подібні результати знайдені для RC, де вищі значення RC вказують на більш складні випадки (Клас I). Справді, випадки з низькими значеннями RC менш обмежені, тоді як вищі значення RC вказують на випадки з дуже обмеженими ресурсами. Знову ж таки, прогностичні моделі підтверджують, що правильний вибір стає важливішим, як зростають значення RC.



(a) Number of best labels



(b) Impact of SP and RC

Рисунок 3.4 - Вплив характеристик та міток.

Частина 4. Час обчислень. Для оцінки ефективності прогностичних моделей звітуються часи обчислень. Таблиця 8 наводить підсумок часів, представлених у годинах (год.) або секундах (сек.). Оскільки рішення для всіх 48 конфігурацій отримуються для кожного з 3,143 випадків перед тим, як моделі навчаються та тестуються (розділ 3.2), їх часи роботи відображені у другій колонці таблиці. Зазначимо, що загальний час, використаний для всіх випадків, менший за час кожного випадку (60 с або 1 год) помножений на загальну кількість випадків помножену на 48 (тобто

60 с (або 1 год) $\times 3,143 \times 48$), оскільки частина випадків може бути розв'язана менше ніж за час кожного випадку. Наступні колонки показують час для навчання та тестування. Загальний час для навчання моделі включає відображення рейтингового списку з характеристиками та тонке налаштування параметрів. Час для тестування моделі - це лише загальний час для прогнозування рейтингового списку для кожного випадку у тестовому наборі (тобто генерація прогнозів) та не включає час для запуску передбачених конфігурацій для певного випадку. Це пояснюється тим, що рішення для кожної конфігурації вже відоме на етапі підготовки даних, і, отже, немає потреби знову запускати модель із передбаченими конфігураціями. Однак у реальному сценарії з використанням нового невідомого проектного випадку передбачений рейтинговий список, створений моделлю, вимагає початку нових процедур гілок і меж, що вимагає часу ЦПУ, який не вказаний у таблиці 3.6.

Таблиця 3.6. Час роботи запропонованих моделей.

Використаний час			LB/UB	kNN/ridge	
Кожен екземпляр	Всі екземпляри			kNN	ridge
60s	5,645,314 s = 1,568.14 h.	Навчання	LB	15.75 s	105.55 s
			UB	16.44 s	154.51 s
		Тестування	LB	0.02 s	0.02 s
			UB	0.02 s	0.04 s
1h	90,274 h.	Навчання	LB	-	-
			UB	15.94 s	149.43 s
		Тестування	LB	-	-
			UB	0.02 s	0.04 s

Отже, вивчено ефективність двох прогностичних моделей (ridge і kNN) у розв'язанні складних завдань оптимізації. Порівняння загальної продуктивності, вплив обмеження кількості міток, взаємодія характеристик та міток, а також обчислювальні часи надають обширний огляд їхньої ефективності. Прогностичні моделі виявилися

конкурентоспроможними порівняно з окремими найкращими конфігураціями та наближаються до оптимальних рішень S . Взаємозв'язок між характеристиками та мітками вказує на складність задачі, де обмежена кількість міток дає найкращі рішення. Обчислювальні часи підтверджують ефективність моделей, підкреслюючи їхню придатність для роботи в реальних умовах із різними обмеженнями часу та кількості міток.

3.2.3. Експеримент 3. Різні конфігурації

У даному експерименті досліджується вплив якості пулу початкових конфігурацій на продуктивність прогностичних моделей. У всіх експериментах використовувалася група з 48 конфігурацій, і ємність цієї групи дозволяє припустити, що лише обмежена кількість конфігурацій може виявитися продуктивною, в той час як інші можуть бути менш ефективними. Це може спотворювати вплив на продуктивність прогностичних моделей. Отже, групу із 48 конфігурацій було розділено на підгрупи, і для кожної підгрупи проведено окреме навчання та тестування прогностичних моделей. Чотири підгрупи створено наступним чином:

- Група 1 містить дуже конкурентоспроможні конфігурації та складається з 16 найкращих конфігурацій за Таблицею 3.7.
- Група 2 містить конфігурації 17-32 з Таблиці 3.7 з середньою продуктивністю.
- Група 3 містить останні 16 конфігурацій з Таблиці 3.7, які мають відносно слабку продуктивність, особливо порівняно з тими з Групи 1.
- Група 4 містить 16 конфігурацій з дуже різноманітною продуктивністю. У виборі конкретних конфігурацій були обрані ті,

які вказані в Таблиці 3.7, вибираючи кожну третю, тобто конфігурації 1, 4, 7, ..., 46.

Для кожної групи, що складається з 16 конфігурацій ($|Q| = 16$), кількість показників проекту все ще становить $|F| = 12$, як і у попередніх експериментах. Після повторного навчання моделі для кожної групи, результати тестування прогностичних моделей показані у Таблиці 9 разом із найкращою конфігурацією тієї групи та C . Варто відзначити, що зараз використовується позначення $C(16)$, щоб чітко виділити відмінність між C , яка використовує всі 48 конфігурацій у попередніх експериментах, і $C(16)$, яка використовує 16 конфігурацій з кожної групи.

Таблиця 3.7. Порівняння продуктивності у чотирьох групах.

		CLB	ADev		CUB	ADev'
Група 1	C (16)	28,416	0	C (16)	97,830	0
	kNN	28,415 (1)	1	kNN	98,109 (1)	279
	ridge	28,413 (2)	3	ridge	98,152 (2)	322
	LPB12	28,377	39	UAA12	101,150	3,320
	AvgTPL = 6			AvgTPL = 4		
Група 3	C (16)	7,346	0	C (16)	21,140	0
	ridge	7,346 (1)	0	ridge	21,140 (1)	0
	kNN	7,336 (2)	10	kNN	21,140 (1)	0
	USB12	7,333	13	LPA0	24,602	3,462
	AvgTPL = 7			AvgTPL = 7		
Група 2	C (16)	52,498	0	C (16)	77,847	0
	ridge	52,498 (1)	0	kNN	79,566 (1)	1,719
	kNN	52,479 (2)	19	ridge	80,107 (2)	2,260
	LAB12	52,270	228	UPB12	80,580	2,733
	AvgTPL = 8			AvgTPL = 4		
Група 4	C (16)	69,909	0	C (16)	107,499	0
	ridge	69,906 (1)	3	kNN	107,842 (1)	343
	kNN	69,906 (1)	3	ridge	108,346 (2)	847
	LPB12	69,871	38	UAA12	112,562	5,063
	AvgTPL = 5			AvgTPL = 3		

Таблиця 3.7 чітко показує, що коли продуктивність конфігурацій у групі змінюється, прогностична модель все одно досягає високої конкурентоспроможної продуктивності. Число в дужках після двох прогностичних моделей показує їх ранг, як і у попередніх таблицях, перевищуючи будь-яку найкращу одиночну конфігурацію для кожної

групи. Хороша продуктивність кожної групи є ще одним показником того, що прогностичні моделі можна використовувати для будь-якого методу розв'язання РОПП, і тому їх використання виходить за межі використання лише конфігурацій гілок та меж. Важливо відзначити, що порівняння результатів між групами не є можливим, оскільки кількість випадків у тестовому наборі є різною. Це пов'язано з видаленням "легких" випадків для кожної групи, як розглядалося раніше у розділі 3.1.

Отже, у третьому експерименті, що розглядає різні конфігурації, продемонстровано, що прогностичні моделі здатні досягати високої продуктивності, навіть при різних якостях початкових конфігурацій. Розділення конфігурацій на групи з різним рівнем конкурентоспроможності підкреслило стійкість та універсальність моделей у вирішенні завдань оптимізації. Незважаючи на варіації в якості конфігурацій, прогностичні моделі продовжують виявляти конкурентоспроможність порівняно із найкращими конфігураціями кожної групи та оптимальним рішенням С (16). Це підтверджує широкий потенціал та застосовність цих моделей для різноманітних сценаріїв у галузі управління проектами.

3.2.4. Експеримент 4. Надійність моделі

В рамках цього експерименту було проведено навчання та тестування прогностичних моделей на різних індивідуальних підмножинах для вивчення впливу кількості активностей та складності сценаріїв задачі на результативність прогностичних моделей.

Різні набори містять випадки з мінімум 20 активностей (у наборі CV) до 300 активностей (у наборі RG300) і складаються з усіх "складних" випадків, як було визначено раніше. Однак, існує гіпотеза, що деякі складні випадки є складнішими за інші, особливо оскільки набір CV

містить дійсно дуже складні випадки, що було саме причиною його створення. Таблиця 3.8 надає порівняння продуктивності різних методів рішень (дві прогностичні моделі, найкраща одинарна конфігурація та C) для різних підмножин.

Таблиця 3.8. Порівняння продуктивності на підмножинах.

Набір даних		CLB	ADev		CUB	ADev'
J30	C	2,617	0	C	3,224	0
	ridge	2,617 (1)	0	ridge	3,224 (1)	0
	kNN	2,617 (1)	0	kNN	3,224 (1)	0
	LPB0	2,617	0	UPB4	3,224	0
	AvgTPL = 35			AvgTPL = 32		
J90	C	2,973	0	C	6,346	0
	ridge	2,973 (1)	0	ridge	6,346 (1)	0
	kNN	2,973 (1)	0	kNN	6,346 (1)	0
	LPA12	2,973	0	UPB12	6,380	34
	AvgTPL = 10			AvgTPL = 17		
CV	C	6,897	0	C	7,858	0
	kNN	6,897 (1)	0	ridge	7,918 (1)	60
	ridge	6,896 (2)	1	kNN	8,050 (2)	192
	LPA8	6,892	5	UPB8	8,183	325
	AvgTPL = 8			AvgTPL = 4		
J60	C	2,865	0	C	5,053	0
	ridge	2,865 (1)	0	ridge	5,054 (1)	1
	kNN	2,865 (1)	0	kNN	5,054 (1)	1
	LPB12	2,861	4	UPB12	5,072	19
	AvgTPL = 19			AvgTPL = 17		
J120	C	9,716	0	C	16,044	0
	ridge	9,716 (1)	0	kNN	16,088 (1)	44
	kNN	9,716 (1)	0	ridge	16,110 (2)	66
	LPB12	9,714	2	UAA12	17,961	1,917
	AvgTPL = 12			AvgTPL = 6		
RG300	C	34,512	0	C	58,275	0
	ridge	34,512 (1)	0	kNN	58,298 (1)	23
	kNN	34,512 (1)	0	ridge	58,418 (2)	143
	LPB12	34,509	3	UAA0	59,194	919
	AvgTPL = 23			AvgTPL = 4		

Ця таблиця 3.8 вказує на високу робастність прогностичної моделі, оскільки і ridge, і kNN все ще показують відмінні результати, навіть якщо кількість активностей зростає (RG300) і кількість випадків стає складнішою (набір CV). kNN майже завжди показує незначну перевагу над ridge з точки зору як LB, так і UB, і їх ранги все ще дорівнюють 1 або 2, що означає, що вони завжди перевищують найкращу одинарну конфігурацію.

Усі рішення прогностичних моделей близькі до C і показують лише невеликі відхилення, коли розмір випадків збільшується. Наприклад, $ADev' = 44$ для J120 і 23 для RG300, тоді як для менших випадків вони дорівнюють 0 або 1. Найбільші значення $ADev'$ знаходяться для набору CV (60), що підтверджує, що ці випадки є найскладнішими для розв'язання. Проте всі результати підтверджують, що прогностичні моделі добре працюють на різних наборах даних з різною кількістю активностей, які містять відносно легкі до дуже складних випадків, що є ще одним показником того, що вони можуть використовуватися для розв'язання різних інших проблем планування.

Отже, у четвертому експерименті, що розглядає надійність прогностичних моделей на різних підмножинах даних, виявлено високу робастність і універсальність моделей при розв'язанні завдань оптимізації проектів з різною кількістю активностей та рівнем складності випадків. Незважаючи на збільшення кількості активностей та ускладнення випадків, прогностичні моделі виявили стійку продуктивність, демонструючи ефективність у різноманітних сценаріях. Кількість активностей та характер складності випадків не суттєво вплинули на їхню здатність до надійного прогнозування. Це підтверджує перспективи використання цих моделей в широкому спектрі завдань у сфері управління проектами та оптимізації.

3.2.5. Експеримент 5. Розподіл обмеження за часом

Цей підрозділ звітує про результати тренувальних і тестових експериментів, проведених з трьома часовими обмеженнями (1 секунда, 1 хвилина та 1 година), щоб показати, що ранжування конфігурацій у нашому дослідженні перевершує традиційний підхід, у якому використовується лише одна добре працююча конфігурація. Результати

представлені в Таблиці 3.9. Експеримент використовує передбачений рейтинговий список і вибирає лише перші s конфігурацій зі списку, які потім послідовно використовуються для розв'язання тестового випадку протягом загального заданого часового ліміту 1 секунди, 1 хвилини або 1 години. Для $s=1$, цей підхід схожий на вибір найкращої передбачуваної конфігурації для розв'язання випадку в межах часового ліміту.

Таблиця 3.9. Результати використання перших s конфігурацій у ранговому списку (три часові обмеження).

с	Ліміт часу = 1s		Ліміт часу = 1m		Ліміт часу = 1h	
	kNN	ridge	kNN	ridge	kNN	ridge
1	105,023	114,626	<u>101,818</u>	<u>104,185</u>	<u>101,248</u>	<u>103,581</u>
2	103,030	104,666	100,145	<u>101,011</u>	<u>99,780</u>	<u>100,640</u>
3	112,562	100,908	99,820	100,018	99,491	<u>99,694</u>
4	112,065	101,434	99,747	99,976	99,379	99,626
5	100,988	102,427	99,662	99,844	99,270	99,449
6	101,603	101,504	99,678	99,824	99,258	99,406
7	102,242	102,726	99,650	99,775	99,220	99,344
8	102,137	102,544	99,563	99,659	99,125	99,209
9	102,174	102,071	99,561	99,665	99,094	99,193
10	102,223	101,574	99,571	99,635	99,100	99,158
C	99,742		99,031		98,745	
Single best	104,843 (UAA4)		104,360 (UAA12)		104,035 (UAA12)	

Однак, для $s>1$, час між ранжованими конфігураціями розподіляється рівномірно між усіма s конфігураціями. Наприклад, для $s=4$, кожна конфігурація отримує 15 хвилин часу, коли загальний часовий ліміт встановлено на 1 годину. Фаза тестування починається з першої рангової конфігурації, яка забезпечує допустиме рішення (верхню межу UB) після 15 хвилин, яке використовується як вхід для другої рангової конфігурації (обмеженої 15 хвилинами) і ця процедура зупиняється після 4 конфігурацій (з загальним часом 1 година) і повідомляє найкращу знайдену верхню межу. Крім того, останні два рядки показують результат конфігурації C і найкращої одинарної конфігурації, і результати додатково підтверджують висновки попередніх експериментів, тобто результати, отримані за допомогою рангових моделей, кращі за найкращу

одинарну конфігурацію і ближчі до тих, що дає C . Отримані результати представлені в Таблиці 11 для різних значень параметра c від 1 до 10 (де 10 відповідає AvgTPL із Таблиці 3. 7). Проте, наведено лише результати розрахунків верхньої межі, а не нижньої межі. Крім того, враховано три різні часові обмеження, а не лише 60 секунд.

Отже, підсумовані результати можуть бути представлені наступним чином: По-перше, найкраще знайдене рішення для будь-якого $c=1$ до 10 протягом заданого часового ліміту відображено жирним шрифтом, що очевидно показує, що рішення покращуються зі збільшенням часу CPU. Однак, найкращі значення c нижчі для дуже коротких прогонів (1 секунда) порівняно з прогонами CPU тривалістю 1 хвилина та 1 година ($c=9$ для kNN та $c=10$ для ridge), що вказує на те, що розподіл пошуку з різними ранжованими конфігураціями краще працює, коли достатньо часу. По-друге, значення, підкреслені курсивом, показують рішення з часом CPU = T^{cpu} (наприклад, $T^{cpu}=1$ година) та кількістю вибраних конфігурацій, рівною c (наприклад, $c=1$), які гірші за рішення з часом CPU = $T^{cpu}/60$ (наприклад, 1 хвилина) з вибраною кількістю конфігурацій $c'>c$. Це показує, що розподіл пошуку в межах певного часового обмеження та використання ранжованої конфігурації працює краще, ніж використання однієї єдиної конфігурації протягом більшого часу. Наприклад, розподіл пошуку kNN для прогонів тривалістю 1 хвилина на 9 окремих прогонів, кожен тривалістю менше 7 секунд, працює краще (99,561), ніж один прогін з часовим лімітом 1 година (101,248). Нарешті, варто також зазначити, що збільшення часу покращує якість рішення, але лише незначно зі збільшенням часу CPU. Наприклад, покращення для kNN становить 1,41% (з 100,988 до 99,561) при переході від 1 секунди до 1 хвилини (з $c=5$ до $c=9$), але ці покращення знижуються лише до 0,47% (до 99,094), коли часовий ліміт далі збільшується до 1 години.

У п'ятому експерименті, який розглядає розподіл обмежень за часом для трьох різних часових лімітів, виявлено, що підхід з використанням ранжованих конфігурацій перевершує традиційний метод, де використовується лише одна добре працююча конфігурація. Результати продемонстрували, що збільшення кількості вибраних конфігурацій (с) та часу CPU позитивно впливає на покращення якості знайдених рішень. Однак, при дуже обмеженому часі (наприклад, 1 секунда), ефективність розподілу пошуку залишається нижчою порівняно з тривалішими прогонами (1 хвилина та 1 година). Розподіл пошуку з ранжованими конфігураціями в межах обмеженого часу показав кращі результати, вказуючи на його перевагу у вибіркового використанні ресурсів та отриманні кращих результатів за обмежений час. Загалом, ці результати підкреслюють ефективність методів, які використовують ранжовані конфігурації у вирішенні задач оптимізації проектів.

3.3. Порівняння розробленого методу з існуючими методами

У цьому розділі проводиться порівняння методу ранжування з найкраще виконуючими алгоритмами типу вітка та межа (branch-and-bound), описаними у науковій літературі, з використанням тільки одногодинного часового ліміту. Для досягнення цієї мети, було проведено аналіз 12 наявних процедур гілля та межі та їх подальше відображення на конфігурації, що використовуються у нашому дослідженні. Оскільки деякі дослідження вітка та межа можна було зіставити з однією й тою ж конфігурацією, в результаті у нас вийшло 9 конфігурацій. У таблиці 3.9 наведено короткий огляд з посиланням на оригінальне дослідження, скорочену конфігурацію відповідно до аббревіатур Додатку А, а також найкраще значення верхньої межі (тривалість проекту), знайдене

протягом однієї години. Крім того, у перших двох рядках представлені найкращі результати з Таблиці 3.9.

Таблиця 3.10. Порівняння моделі прогнозування рейтингу з 9 конфігураціями з літератури.

Література	Абревіатура	CUB
Це дослідження	kNN (c=9)	99,094
	ridge (c=10)	99,158
[50], [56]	UAA0	104,109
[53]	USA4	113,412
[54]	USA0	113,430
[12], [19], [21]; [44]	UPA4	183,760
[12], [19], [21]; [44]	UPA0	183,793
[55]	LPB4	252,249
[55], [1]	LPB0	253,304
[48]	LPA0	257,616
[51]	LAA0	286,881

Результати показують, що ранжувальний метод, представлений у цьому дослідженні, перевершує всі методи вітка та межа, доступні у науковій літературі. Це ясно демонструє, що ранжування різних конфігурацій з літератури в одну інтегровану модель передбачення варте зусиль, і слід приділити більше досліджень для кращого розуміння, коли одна конфігурація працює краще, ніж інша, для конкретного набору тестових випадків. Це дослідження сподівається служити стимулом для подальших наукових розвідок в галузі планування проектів з обмеженими ресурсами.

У цій роботі вводиться структурований підхід до передбачення для ранжування ефективності різних компонентів складного процесу вітка та межа (branch-and-bound) [15] для вирішення відомої проблеми планування проектів з обмеженими ресурсами. Предиктивні моделі мають на меті створення завдання ранжування шляхом зіставлення відомих показників проекту з повним порядком різних конфігурацій вітка та межа, які використовують 3 різних схеми розгалуження, 1 стратегію дерева, 2

стратегії пошуку, 4 композитні нижні межі та 2 порядки розгалуження з наукової літератури.

Предиктивні моделі складаються з двох методів регресії, які використовуються як модель передбачення для генерації списку ранжування 48 конфігурацій на основі відомих значень показників проекту для нового, ще не баченого проекту. Ці тестові випадки розв'язуються з обмеженим набором конфігурацій із списку ранжування, і результати порівнюються з найкращою однією конфігурацією, а також найкращою конфігурацією кожного окремого випадку, як у термінах нижньої межі (LB), так і верхньої межі (UB).

Обчислювальний експеримент проводився на наборі з 3,143 проектів за різними критеріями зупинки: 1 секунда, 60 секунд та 1 година. Результати показують, що обидва методи завжди перевершують будь-яку найкращу одну конфігурацію і часто наближаються до найкращого можливого рішення C . У певних експериментах було проведено навчання та тестування моделей лише на складних випадках, де вивчалася якість конфігурацій у підгрупах, розмір проблемних випадків та інші аспекти. Кожного разу добра продуктивність передбачувальних моделей підтверджувалася, що ілюструє потенціал передбачення міток для широкого спектру схожих, але водночас різних проблем планування проектів.

До цього дослідження слід ставитися критично, оскільки алгоритм, який використовується в цьому дослідженні, потребує не лише великих зусиль із реалізації для включення всіх стратегій розгалуження та межування, але також покладається на процес навчання, який потребує величезного обчислювального часу. Тим не менш, результати показують, що поєднання найкращих компонентів з літератури покращує якість вирішення проблем планування проекту з обмеженими ресурсами, і, отже, дослідження показало, що вивчення переваг загалом і прогнози

ранжирування міток, зокрема, є цікавими напрямками дослідження. в контексті планування проекту. Це дослідження може відкрити нові напрямки у вивченні, впроваджуючи модель прогнозування міток із простішим алгоритмом планування, таким як правила пріоритету або метаевристики. По суті, заміна конфігурацій розгалужень і зв'язків простими та швидкими правилами пріоритету дозволить користувачеві автоматично ранжувати найефективніші правила пріоритету, а потім швидко розв'язувати великі екземпляри проекту реального розміру практично в найкоротші терміни. Крім того, використання метаевристичних алгоритмів вирішення (замість різноманітних процедур розгалуження та зв'язування) вимагало б менше зусиль для реалізації та могло б відносно легко поширюватися на інші, більш складні проблеми планування, роблячи метод прогнозування міток набагато більш гнучким для вирішувати ширший спектр проблем планування проекту. Останнім, але не менш важливим аспектом є наше переконання, що майбутні дослідження мають фокусуватися на завданні ранжування, використовуючи прогнозні моделі з частковим, а не повним рейтинговим списком. Фактично, у багатьох реальних сценаріях не існує повного рейтингу, а лише частковий або неповний, і наша мета - дослідити використання методів часткового та неповного ранжування для цієї складної задачі планування проекту з обмеженими ресурсами. Крім того, нашу модель прогнозування також можна розширити для вирішення інших проблем планування, таких як проблема багатопроєктного планування з обмеженим ресурсом (RCMPSP) і проблема планування багаторежимного проекту з обмеженим ресурсом (M РОПП). Для таких розширених проблем планування слід перевизначати не лише мітки, але й функції, але загальний підхід залишається тим самим. Такі розширення також займають важливе місце в програмі наших досліджень.

Отже, у порівнянні розробленого методу з існуючими алгоритмами вітка та межа (branch-and-bound) виявлено, що ранжувальний підхід, який використовується в цьому дослідженні, перевершує всі розглянуті методи у науковій літературі щодо вирішення проблеми планування проектів з обмеженими ресурсами. Застосування структурованого підходу до передбачення та ранжування конфігурацій вітка та межа дозволило покращити якість вирішення проблеми, і це дослідження може слугувати основою для подальших робіт у цьому напрямку. Надалі, можливість розширення моделі для вирішення інших проблем планування, таких як проблема багатопроєктного планування з обмеженим ресурсом та проблема планування багаторежимного проекту з обмеженим ресурсом, вказує на потенційні перспективи подальших досліджень у цьому напрямку. Загалом, ця робота вносить важливий внесок у розуміння ефективних стратегій планування проектів з використанням передбачувальних моделей, що може мати велике практичне значення в різних областях управління проектами.

Висновки до розділу 3

1. У третьому розділі практичного використання отриманих наукових результатів проведено оцінку продуктивності розробленої структурованої моделі прогнозування. Досліджено ефективність моделі у розв'язанні проблеми планування проектів з обмеженими ресурсами. Отримані результати свідчать про високу продуктивність та переваги використання цієї моделі в порівнянні з традиційними підходами. Проведена оцінка дозволяє визначити реальні можливості та перспективи впровадження розробленого методу у реальних проектних умовах.
2. Розглянуто аналіз продуктивності прогностичних моделей, що включає ряд експериментів. Зокрема, вивчено налаштування параметрів моделі, які визначають її ефективність. Детально проаналізовано якість прогнозування, різні конфігурації та надійність моделі. Отримані результати свідчать про стабільність та надійність розробленого методу в різних умовах та сценаріях використання. Використання прогностичних моделей показало себе як перспективний напрямок у розв'язанні задач планування проектів.
3. Проведено порівняльний аналіз розробленого методу із використанням прогностичних моделей з існуючими методами, зокрема, методами вітка та межа. Отримані результати свідчать про вищу ефективність розробленого методу, його здатність до перевершення існуючих стратегій та алгоритмів в розв'язанні складних завдань планування. Порівняння дозволяє визначити конкурентні переваги розробленого методу та сприяє впровадженню його в реальні умови управління проектами з обмеженими ресурсами.

ВИСНОВКИ

Ця магістерська робота зосереджується на аналізі та розвитку методів ресурсно-обмеженого планування проектів. Розглянувши теоретичні основи та практичні аспекти управління проектами, робота виявляє ключові виклики в ефективному використанні ресурсів. Значущість теми полягає в тому, що в умовах сучасного бізнес-середовища, ефективне планування ресурсів є критично важливим для успіху проектів.

Робота глибоко аналізує складні аспекти РОПП, включаючи управління обмеженими ресурсами, визначення тривалості завдань та інші змінні параметри. Ці аспекти підкреслюють складність задачі та необхідність розробки нових підходів для її вирішення.

У роботі пропонуються новаторські підходи до прогнозування рангів конфігурацій у РОПП. Через детальний аналіз чотирьох ключових фаз процесу прогнозування, включаючи підготовку даних, розбиття даних, вибір та налаштування моделі, а також оцінку ефективності, робота вказує на перспективні шляхи розвитку в галузі управління проектами.

Дослідження показує, що пропоновані методи прогнозування є ефективними у вирішенні різних проблем РОПП. Аналіз результатів експериментів підтверджує високу надійність та стабільність цих методів в різних сценаріях застосування.

Аналіз порівняльної ефективності пропонованих прогностичних моделей з традиційними методами розв'язання РОПП, зокрема вітка та межа, вказує на їх вищу продуктивність. Це підтверджує, що інтеграція різних конфігурацій у єдину прогностичну модель є ефективним підходом.

Робота демонструє важливість розроблених методів у реальних умовах управління проектами. Здатність моделей адаптуватися до різних сценаріїв проектів та їх ефективність у вирішенні складних завдань підкреслює їхню практичну значущість.

Результати дослідження відкривають нові перспективи для подальших наукових розробок у галузі управління проектами. Існують можливості для удосконалення прогностичних моделей та їх застосування в різних умовах та типах проектів.

Попри значні досягнення, у роботі також визначаються обмеження поточних методів та можливості для їх удосконалення. Це стосується як технічних аспектів моделей, так і їх адаптації до різноманітних умов проектних середовищ.

Загалом, ця магістерська робота робить важливий внесок у розуміння та вдосконалення методів ресурсно-обмеженого планування проектів. Вона не лише розкриває нові теоретичні аспекти, але й надає практичні інструменти для ефективного управління проектами, що мають велике значення для наукової спільноти та практиків у сфері управління проектами.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. C. E. Bell and K. Park, "Solving resource-constrained project scheduling problems by a search," *Naval Research Logistics (NRL)*, vol. 37, no. 1, pp. 61–84, 1990.
2. J. Blazewicz, J. K. Lenstra, and A. R. Kan, "Scheduling subject to resource constraints: classification and complexity," *Discrete Applied Mathematics*, vol. 5, no. 1, pp. 11–24, 1983.
3. C. Brouard, M. Szafranski, and F. d'AlchéBuc, "Input-output kernel regression: Supervised and semi-supervised structured output prediction with operator-valued kernels," *Journal of Machine Learning Research*, vol. 17, 2016.
4. P. Brucker et al., "Resource-constrained project scheduling: Notation, classification, models, and methods," *European Journal of Operational Research*, vol. 112, no. 1, pp. 3–41, 1999.
5. P. Brucker et al., "A branch and bound algorithm for the resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 107, no. 2, pp. 272–288, 1998.
6. A. Caponnetto and E. De Vito, "Optimal rates for the regularized least-squares algorithm," *Foundations of Computational Mathematics*, vol. 7, no. 3, pp. 331–368, 2007.
7. S. Chand et al., "On the use of genetic programming to evolve priority rules for resource constrained project scheduling problems," *Information Sciences*, vol. 432, pp. 146–163, 2018.
8. K.-Y. Chang, C.-S. Chen, and Y. P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *CVPR 2011*, IEEE, 2011, pp. 585–592.

9. Y.-R. Chen, A. Rezapour, and W. G. Tzeng, "Privacy-preserving ridge regression on distributed data," *Information Sciences*, vol. 451, pp. 34–49, 2018.
10. W. Cheng, K. Dembczynski, and E. Hüllermeier, "Label ranking methods based on the Plackett-Luce model," in *ICML*, 2010.
11. W. Cheng and E. Hüllermeier, "A nearest neighbor approach to label ranking based on generalized labelwise loss minimization," in *Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, Citeseer, 2013.
12. N. Christofides, R. Alvarez-Valdés, and J. M. Tamarit, "Project scheduling with resource constraints: A branch and bound approach," *European Journal of Operational Research*, vol. 29, no. 3, pp. 262–273, 1987.
13. C. Ciliberto, L. Rosasco, and A. Rudi, "A consistent regularization approach for structured prediction," *Advances in Neural Information Processing Systems*, vol. 29, pp. 4412–4420, 2016.
14. J. Coelho and M. Vanhoucke, "An exact composite lower bound strategy for the resource-constrained project scheduling problem," *Computers & Operations Research*, vol. 93, pp. 135–150, 2018.
15. J. Coelho and M. Vanhoucke, "Going to the core of hard resource-constrained project scheduling instances," *Computers & Operations Research*, vol. 121, 104976, 2020.
16. D. F. Cooper, "Heuristics for scheduling resource-constrained projects: An experimental investigation," *Management Science*, vol. 22, no. 11, pp. 1186–1194, 1976.
17. S. K. De and P. R. Krishna, "A new approach to mining fuzzy databases using nearest neighbor classification by exploiting attribute hierarchies," *International Journal of Intelligent Systems*, vol. 19, no. 12, pp. 1277–1290, 2004.

- 18.D. Debels and M. Vanhoucke, "A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem," *Operations Research*, vol. 55, no. 3, pp. 457–469, 2007.
- 19.E. Demeulemeester and W. Herroelen, "A branch-and-bound procedure for the multiple resource-constrained project scheduling problem," *Management Science*, vol. 38, no. 12, pp. 1803–1818, 1992.
- 20.E. Demeulemeester, M. Vanhoucke, and W. Herroelen, "Rangen: A random network generator for activity-on-the-node networks," *Journal of Scheduling*, vol. 6, no. 1, pp. 17–38, 2003.
- 21.E. L. Demeulemeester and W. S. Herroelen, "New benchmark results for the resource-constrained project scheduling problem," *Management Science*, vol. 43, no. 11, pp. 1485–1492, 1997.
- 22.L. Devroye, L. Györfi, and G. Lugosi, "A probabilistic theory of pattern recognition, volume 31," Springer Science & Business Media, 2013.
- 23.U. Dorndorf, E. Pesch, and T. Phan-Huy, "A branch-and-bound algorithm for the resource-constrained project scheduling problem," *Mathematical Methods of Operations Research*, vol. 52, no. 3, pp. 413–439, 2000.
- 24.P. Exterkate, "Model selection in kernel ridge regression," *Computational Statistics & Data Analysis*, vol. 68, pp. 1–16, 2013.
- 25.P. Fan et al., "Well logging curve reconstruction based on kernel ridge regression," *Arabian Journal of Geosciences*, vol. 14, no. 16, 1–10, 2021.
- 26.J. Fürnkranz et al., "Multilabel classification via calibrated label ranking," *Machine Learning*, vol. 73, no. 2, pp. 133–153, 2008.
- 27.W. Guo, M. Vanhoucke, J. Coelho, and J. Luo, "Automatic detection of the best performing priority rule for the resource-constrained project scheduling problem," *Expert Systems with Applications*, vol. 167, 114116, 2021.

28. S. Hartmann and R. Kolisch, "Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 127, no. 2, pp. 394–407, 2000.
29. W. Herroelen, E. Demeulemeester, and B. De Reyck, "A classification scheme for project scheduling," in *Project Scheduling*, Springer, 1999, pp. 1–26.
30. A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics: A Journal of Statistics for the Physical, Chemical, and Engineering Sciences*, vol. 12, no. 1, pp. 55–67, 1970.
31. E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, nos. 16–17, pp. 1897–1916, 2008.
32. G. Khalaf, "A proposed ridge parameter to improve the least square estimator," *Journal of Modern Applied Statistical Methods*, vol. 11, no. 2, pp. 15, 2012.
33. R. Klein and A. Scholl, "Computing lower bounds by destructive improvement: An application to resource-constrained project scheduling," *European Journal of Operational Research*, vol. 112, no. 2, pp. 322–346, 1999.
34. R. Kolisch and S. Hartmann, "Experimental investigation of heuristics for resource-constrained project scheduling: An update," *European Journal of Operational Research*, vol. 174, no. 1, pp. 23–37, 2006.
35. R. Kolisch and R. Padman, "An integrated survey of deterministic project scheduling," *Omega*, vol. 29, no. 3, pp. 249–272, 2001.
36. A. Korba, A. Garcia, and F. d'AlchéBuc, "A structured prediction approach for label ranking," *Advances in Neural Information Processing Systems*, vol. 31, pp. 8994–9004, 2018.
37. D. Krstajic et al., "Cross-validation pitfalls when selecting and assessing regression and classification models," *Journal of Cheminformatics*, vol. 6, no. 1, pp. 1–15, 2014.

38. T.-H. Lee, A. Ullah, and R. Wang, "Bootstrap aggregating and random forest," in *Macroeconomic forecasting in the era of big data*, Springer, 2020, pp. 389–429.
39. D. H. Lehmer, "Teaching combinatorial tricks to a computer," in *Proc. Sympos. Appl. Math. Combinatorial Analysis*, vol. 10, 1960, pp. 179–193.
40. M. Mareš and M. Straka, "Linear-time ranking of permutations," in *European Symposium on Algorithms*, Springer, 2007, pp. 187–193.
41. A. A. Mastor, "An experimental investigation and comparative evaluation of production line balancing techniques," *Management Science*, vol. 16, no. 11, pp. 728–746, 1970.
42. T. Messelis and P. De Causmaecker, "An automatic algorithm selection approach for the multi-mode resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 233, no. 3, pp. 511–528, 2014.
43. J. H. Min and Y. C. Lee, "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters," *Expert Systems with Applications*, vol. 28, no. 4, pp. 603–614, 2005.
44. A. Mingozzi, V. Maniezzo, S. Ricciardelli, and L. Bianco, "An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation," *Management Science*, vol. 44, no. 5, pp. 714–729, 1998.
45. W. Guo, M. Vanhoucke, and J. Coelho, "European Journal of Operational Research," vol. 306, pp. 579–595, 2023.
46. K. Moumene and J. A. Ferland, "New representation to reduce the search space for the resource-constrained project scheduling problem," *RAIRO-Operations Research*, vol. 42, no. 2, pp. 215–228, 2008.
47. W. Myrvold and F. Ruskey, "Ranking and unranking permutations in linear time," *Information Processing Letters*, vol. 79, no. 6, pp. 281–284, 2001.

48. T. Nazareth, S. Verma, S. Bhattacharya, and A. Bagchi, "The multiple resource constrained project scheduling problem: A breadth-first approach," *European Journal of Operational Research* , vol. 112, no. 2, pp. 347–366, 1999.

49. T. L. Pascoe, "Allocation of resources CPM," *Revue Française de Recherche Opérationnelle* , vol. 10, no. 38, p. 31, 1966.

50. J. H. Patterson, "Project scheduling: The effects of problem structure on heuristic performance," *Naval Research Logistics Quarterly* , vol. 23, no. 1, pp. 95–123, 1976.

51. J. H. Patterson and W. D. Huber, "A horizon-varying, zero-one approach to project scheduling," *Management Science* , vol. 20, no. 6, pp. 990–998, 1974.

52. K. Rakesh and P. N. Suganthan, "An ensemble of kernel ridge regression for multi-class classification," *Procedia Computer Science* , vol. 108, pp. 375–383, 2017.

53. A. Sprecher, "Scheduling resource-constrained projects competitively at modest memory requirements," *Management Science* , vol. 46, no. 5, pp. 710–723, 2000.

54. A. Sprecher and R. Kolisch, "PSPLIB-Project scheduling problem library," *European Journal of Operational Research* , vol. 96, pp. 205–216, 1996.

55. J. P. Stinson, E. W. Davis, and B. M. Khumawala, "Multiple resource–constrained scheduling using branch and bound," *AIIE Transactions* , vol. 10, no. 3, pp. 252–259, 1978.

56. F. B. Talbot and J. H. Patterson, "An efficient integer programming algorithm with network cuts for solving resource-constrained scheduling problems," *Management Science*, vol. 24, no. 11, pp. 1163–1174, 1978.

57. H. Trang and L. Tran, "Kernel ridge regression method applied to speech recognition problem: A novel approach," in *2014 International Conference on*

Advanced Technologies for Communications (ATC 2014) , IEEE, 2014, pp. 172–174.

58. M. Vanhoucke, J. Coelho, D. Debel, B. Maenhout, and L. Tavares, "An evaluation of the adequacy of project network generators with systematically sampled networks," *European Journal of Operational Research* , vol. 187, pp. 511–524, 2008.

59. Y. Zhou and G. Qiu, "Random forest for label ranking," *Expert Systems with Applications* , vol. 112, pp. 99–109, 2018.

60. "Загальні рекомендації з підготовки, оформлення, захисту та оцінювання випускних кваліфікаційних робіт здобувачів вищої освіти першого «бакалаврського» і другого «магістерського» рівнів / За ред. доц. М.І. Шинкарика." Тернопіль: ТНЕУ, 2018. 67 с.

61. "Комар М.П., Саченко А.О., Васильків Н.М. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за другим (магістерським) рівнем вищої освіти." Тернопіль: ЗУНУ, 2021. 32 с.

ДОДАТОК А

48 МОЖЛИВИХ КОМБІНАЦІЙ ВСІХ КОМПОНЕНТІВ

Таблиця А - Відповідність між конфігураціями та різними
комбінаціями компонентів

Abbreviation	Search strategy	Branching scheme	Branching order	Composite lower bound
	L/U	A/P/S	A/B	CLB0 (0)/CLB4 (4)/CLB8 (8)/CLB12 (12)
UAB0	U	A	B	0
LAB0	L	A	B	0
USB0	U	S	B	0
LSB0	L	S	B	0
UPB0	U	P	B	0
LPB0	L	P	B	0
UAB4	U	A	B	4
LAB4	L	A	B	4
USB4	U	S	B	4
LSB4	L	S	B	4
UPB4	U	P	B	4
LPB4	L	P	B	4
UAB8	U	A	B	8
LAB8	L	A	B	8
USB8	U	S	B	8
LSB8	L	S	B	8
UPB8	U	P	B	8
LPB8	L	P	B	8
UAB12	U	A	B	12
LAB12	L	A	B	12
USB12	U	S	B	12
LSB12	L	S	B	12
UPB12	U	P	B	12
LPB12	L	P	B	12
UAA0	U	A	A	0
LAA0	L	A	A	0
USA0	U	S	A	0
LSA0	L	S	A	0
UPA0	U	P	A	0
LPA0	L	P	A	0
UAA4	U	A	A	4
LAA4	L	A	A	4
USA4	U	S	A	4
LSA4	L	S	A	4
UPA4	U	P	A	4
LPA4	L	P	A	4
UAA8	U	A	A	8
LAA8	L	A	A	8
USA8	U	S	A	8
LSA8	L	S	A	8
UPA8	U	P	A	8
LPA8	L	P	A	8
UAA12	U	A	A	12
LAA12	L	A	A	12
USA12	U	S	A	12
LSA12	L	S	A	12
UPA12	U	P	A	12
LPA12	L	P	A	12

ДОДАТОК Б

ДОСЛІДЖЕННЯ РІЗНИХ КОМПОНЕНТІВ

Таблиця Б. Дослідження різних компонентів

Компонент	Абревіатура	Опис	Література
Пошукова стратегія			
Стратегія мінімальної нижньої межі	L	Стратегія починається з початкової нижньої межі (LB), яка є найкращим знайденим значенням для 13 розрахунків нижньої межі, розглянутих у Coelho & Vanhoucke (2018) . Віртуальний UB встановлюється на поточний LB+1, а для пошуку рішення використовується стратегія верхньої межі. Якщо рішення знайдено, то воно є оптимальним рішенням. Якщо дерево пошуку повністю досліджено без знаходження рішення, знаходять нову нижню межу LB +1, і процес повторюється.	«Коельо і Ванхауке» (2018) ; Паттерсон і Хубер (1974)
Стратегія верхньої межі	У файлі	Стратегія перераховує всі недовідані вузли дерева до тих пір, поки не зникнуть вузли, які більше не можуть бути досліджені.	«Паттерсон і Хубер» (1974) , «Белл і Парк» (1990) ; Brucker et al. (1998) ; Christofides et al. (1987) ; Demeulemeester, Herroelen, 1992 , Demeulemeester, Herroelen, 1997 ; Дорндорф, Пеш і Фан-Хюй (2000) ; Мінгоцці, Маньєццо, Ріккарделлі та Б'янко (1998) ; Назарет, Верма, Бхаттачарья і Багчі (1999) ; Stinson et al. (1978) ; Талбот і Паттерсон (1978)
схема розгалуження			
Час початку активності	A	Схема вибирає один вид діяльності для розгалуження, а потім розгалужується за кількома періодами початку діяльності у своєму часовому вікні.	Паттерсон і Хубер (1974) , Талбот і Паттерсон (1978) , Коельо і Ванхауке (2018) ; Dorndorf et al. (2000)
паралельне розгалуження	P	Схема вибирає дії, які можуть бути заплановані паралельно в кожен момент часу, і спирається на підхід збільшення часу, який ітеративно збільшує показник часу в розкладі і шукає набори дій, які можна запланувати.	Christofides et al. (1987) ; Stinson et al. (1978) , Bell & Park (1990) ; Demeulemeester, Herroelen, 1992 , Demeulemeester, Herroelen, 1997 ; Mingozzi et al. (1998) ; Nazareth et al. (1999)

Компонент	Абревіатура	Опис	Література
Послідовне розгалуження	S	Схема вибирає діяльність, яку можна запланувати на самому ранньому початку, і спирається на так звану схему збільшення активності, яка ітеративно планує відповідні види діяльності в частковому розкладі.	Шпрехер (2000) , Коельо і Ванхуке (2018)
Порядок розгалуження			
Порядок ідентифікаторів активності	A	Цей порядок просто спочатку вибирає вправу з найменшим числом.	Sprecher (2000) , Christofides et al. (1987) ; «Коельо і Ванхауке» (2018) ; Demeulemeester, Herroelen, 1992 ; Demeulemeester, Herroelen, 1997 ; Mingozi et al. (1998) ; Nazareth et al. (1999) ; Талбот і Паттерсон (1978)
Найкраща нижня межа	B	Порядок вибирає найнижчі значення нижньої межі як вузол для продовження розгалуження.	«Дзвін і парк» (1990) ; Brucker et al. (1998) ; «Коельо і Ванхауке» (2018) ; Stinson et al. (1978)
Нижня межа композиту			
CLB0	0	Ця стратегія використовує лише критичний шлях нижньої межі LBcpLBcp.	Майже всі окремі нижні межі
CLB4	4	Ця стратегія розширює CLB0 за допомогою LB cc LBcc, LB cs LBcs, LB ip0 LBip0 та LBnp0 LBnp0.	рецензія Klein & Scholl (1999) і
CLB8	8	Ця стратегія розширює CLB4 за допомогою LB np2 LBnp2, LB np1 LBnp1, LB pm1 LBpm1 та LB ip1 LBip1.	і їх складовими стратегіями є
CLB12	12	Ця стратегія розширює CLB8 за допомогою LB pm2 LBpm2, LB pr LBpr, LB ct LBct і LBtp LBtp.	запропонована Coelho & Vanhoucke (2018)
Кількість комбінацій	48		

ДОДАТОК В
АПРОБАЦІЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

МАТЕРІАЛИ

IV ВСЕУКРАЇНСЬКОЇ СТУДЕНТСЬКОЇ НАУКОВОЇ

КОНФЕРЕНЦІЇ

29 ВЕРЕСНЯ 2023 РІК • М. ЧЕРНІГІВ, УКРАЇНА

ЕКСПЕРИМЕНТАЛЬНІ ТА
ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ В
КОНТЕКСТІ СУЧАСНОЇ НАУКИ

ISBN 978-617-8126-63-6
DOI 10.36074/liga-ukr-29.09.2023



УДК 082:001
Е 45

Голова оргкомітету: Коренюк І.О.

Верстка: Зрада С.І.

Дизайн: Бондаренко І.В.



Конференцію зареєстровано Державною науковою установою «УкрІНТЕІ» в базі даних науково-технічних заходів України та інформаційному бюлетені «План проведення наукових, науково-технічних заходів в Україні» (Посвідчення №320 від 16.06.2023).

Матеріали конференції знаходяться у відкритому доступі на умовах ліцензії CC BY-SA 4.0 International.



Е 45 **Експериментальні та теоретичні дослідження в контексті сучасної науки:** матеріали IV Всеукраїнської студентської наукової конференції, м. Чернігів, 29 вересня, 2023 рік / ГО «Молодіжна наукова ліга». — Вінниця: ТОВ «УКРЛОГОС Груп», 2023. — 154 с.

ISBN 978-617-8126-63-6

DOI 10.36074/liga-ukr-29.09.2023

Викладено матеріали учасників IV Всеукраїнської мультидисциплінарної студентської наукової конференції «Експериментальні та теоретичні дослідження в контексті сучасної науки», яка відбулася 29 вересня 2023 року у місті Чернігів, Україна.

УДК 082:001

ISBN 978-617-8126-63-6

© Колектив учасників конференції, 2023

© ГО «Молодіжна наукова ліга», 2023

© ТОВ «УКРЛОГОС Груп», 2023

ВИКОРИСТАННЯ ТЕХНОЛОГІЇ БЛОКЧЕЙН ДЛЯ ЗАХИЩЕНОГО ВЕДЕННЯ ДОКУМЕНТООБІГУ Паплінський В.В., Науковий керівник: Устенко С.В.	60
ДОСЛІДЖЕННЯ МЕТОДІВ ПІДТРИМКИ КОНСИСТЕНЦІЇ У РОЗПОДІЛЕНИХ ТРАНЗАКЦІЯХ Савенков О.А., Науковий керівник: Білова Т.Г.	63
ІНТЕЛЕКТУАЛЬНИЙ МЕТОД ВИЗНАЧЕННЯ ШАХРАЙСЬКИХ ІНТЕРНЕТ-МАГАЗИНІВ Сукмановський О., Науковий керівник: Ліп'яніна-Гончаренко Х.В.	66
ІНТЕЛЕКТУАЛЬНИЙ МЕТОД ФОРМУВАННЯ КОНТЕКСТУ РЕКЛАМИ ТА ЦІЛЬОВОЇ АУДИТОРІЇ НА ОСНОВІ НАВЧАННЯ АСОЦІАТИВНИХ ПРАВИЛ Хам І., Науковий керівник: Ліп'яніна-Гончаренко Х.В.	68
ІНТЕЛЕКТУАЛЬНИЙ МЕТОД ФОРМУВАННЯ РЕКЛАМНОГО КОНТЕНТУ НА ОСНОВІ СЕМАНТИЧНОГО АНАЛІЗУ Курпела П., Науковий керівник: Ліп'яніна-Гончаренко Х.В.	70
КОМП'ЮТЕРИЗАЦІЯ ПРОЦЕСІВ ПРИЙНЯТТЯ РІШЕНЬ ПРИ УПРАВЛІННІ ЦІННИМИ ПАПЕРАМИ Ярцев С.Д., Науковий керівник: Устенко С.В.	72
МОДЕЛЬ НАУКОВОГО ДОСЛІДЖЕННЯ МЕТАВСЕСВІТУ Неделько В.Ю., Науковий керівник: Устенко С.В.	75
ОПТИМІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З ВИБОРУ ДИСЦИПЛІН ЗДОБУВАЧАМИ ЗАКЛАДІВ ВИЩОЇ ОСВІТИ Новак А.С., Науковий керівник: Онищенко В.В.	78
ПЕРЕХРЕСНА ОЦІНКА МОДЕЛЕЙ ІНВЕСТИЦІЙНИХ РИЗИКІВ ВІРТУАЛЬНОЇ ІТ-КОМПАНІЇ НА ОСНОВІ МАШИННОГО НАВЧАННЯ Дем'янова В., Науковий керівник: Ліп'яніна-Гончаренко Х.В.	82
ПІДХІД ПРОГНОЗУВАННЯ ПЛАНУ ПРОЕКТУ З ОБМЕЖЕНИМИ РЕСУРСАМИ Сікач Б., Науковий керівник: Саченко О.А.	84
ПРОЕКТУВАННЯ МОБІЛЬНО ОРІЄНТОВАНОЇ ІС «TASK-MENEDJER» Щербак П.О., Науковий керівник: Устенко С.В.	86
РОЗРОБКА СИСТЕМИ КЕРУВАННЯ РОЗУМНИМ БУДИНКОМ НА ОСНОВІ ПРОТОКОЛУ MQTT Ніщенко Д.О., Науковий керівник: Баранюк О.Ф.	89

СЕКЦІЯ 8. ФІЗИКО-МАТЕМАТИЧНІ НАУКИ

ПРИКЛАД ЗАСТОСУВАННЯ ДИНАМІЧНИХ СИСТЕМ РІВНЯНЬ ДО МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ Бульботка Т.В., Науковий керівник: Тимчук М.В.	92
--	----

Сікач Богдан, здобувач вищої освіти факультету
комп'ютерних інформаційних технологій
Західноукраїнський національний університет, Україна

Науковий керівник: Саченко Олег Анатолійович, канд. техн. наук, доцент,
доцент кафедри інформаційно-обчислювальних систем та управління
Західноукраїнський національний університет, Україна

ПІДХІД ПРОГНОЗУВАННЯ ПЛАНУ ПРОЕКТУ З ОБМЕЖЕНИМИ РЕСУРСАМИ

Загальний підхід до прогнозування рейтингового списку для 48 конфігурацій процедури розгалуження і пов'язаності. Огляд структури прогнозування наведено на рис. 1 і складається з чотирьох різних фаз.

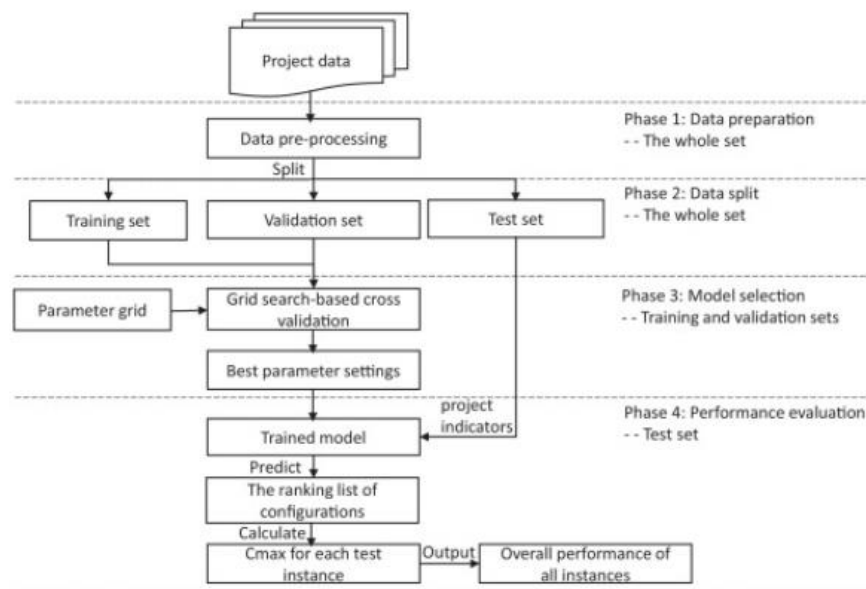


Рис. 1. Огляд підходу до прогнозування плану проекту з обмеженими ресурсами

Етап 1 - Підготовка даних: На цьому етапі виконується попередня обробка даних. Вона включає отримання розв'язків для 48 конфігурацій для всіх випадків, створення рейтингового списку для 48 конфігурацій для кожного екземпляра, видалення "легких" екземплярів, визначення кількості показників проекту та нормалізацію даних.

Етап 2 - Поділ даних: На цьому етапі весь набір даних поділяється на навчальний, перевірочний та тестовий набори. Навчальний набір використовується для навчання моделі, перевірочний - для налаштування параметрів, а тестовий - для оцінки кінцевої продуктивності моделі.

Етап 3 - Вибір моделі: На цьому етапі модель прогнозування налаштовується з використанням навчального набору, а потім перевірочний набір використовується для

визначення оптимальних параметрів моделі.

Етап 4 - Оцінка ефективності [1]: На останньому етапі оцінюється кінцева продуктивність моделі на тестовому наборі, де використовуються оптимальні параметри моделі, і отримані результати використовуються для прогнозу рейтингового списку для 48 конфігурацій.

Цей підхід дозволяє створити прогнозну модель для рейтингового списку на основі попередніх даних та навчальних експериментів. Ця модель може бути корисною для вирішення задач у сфері проектного управління та оптимізації.

У висновку, описаний загальний підхід до прогнозування рейтингового списку для 48 конфігурацій процедури B&B [2] включає чотири основні етапи: підготовку даних, поділ даних, вибір моделі та оцінку ефективності. Цей підхід дозволяє створити модель для прогнозу рейтингів на основі навчання на попередніх даних і налаштування параметрів для отримання оптимальних результатів. Описаний процес може бути корисним для вирішення завдань у сфері проектного управління та оптимізації, де важливо прогнозувати та ранжувати різні конфігурації процедур на основі їх продуктивності.

Список використаних джерел:

1. Krstajic, D., Buturovic, L. J., Leahy, D. E., & Thomas, S. (2014). Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of cheminformatics*, 6(1), 1-15.
2. Morrison, D. R., Jacobson, S. H., Sauppe, J. J., & Sewell, E. C. (2016). Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning. *Discrete Optimization*, 19, 79-102.

**МОЛОДІЖНА
НАУКОВА
ЛІГА** 

МАТЕРІАЛИ КОНФЕРЕНЦІЇ

IV ВСЕУКРАЇНСЬКА СТУДЕНТСЬКА НАУКОВА КОНФЕРЕНЦІЯ



РОЗВИТОК СУЧАСНОЇ НАУКИ:
АКТУАЛЬНІ ПИТАННЯ ТЕОРІЇ ТА
ПРАКТИКИ

 **17 ЛИСТОПАДА 2023 РІК**
 **м. ЛЬВІВ, УКРАЇНА**

Вінниця, Україна
«UKRLOGOS Group»
2023

УДК 082:001
Р 64

Голова оргкомітету: Коренюк І.О.

Верстка: Зрада С.І.

Дизайн: Бондаренко І.В.



Конференцію зареєстровано Державною науковою установою «УкрІНТЕІ» в базі даних науково-технічних заходів України та інформаційному бюлетені «План проведення наукових, науково-технічних заходів в Україні» (Посвідчення №327 від 16.06.2023).

Матеріали конференції знаходяться у відкритому доступі на умовах ліцензії CC BY-SA 4.0 International.



Розвиток сучасної науки: актуальні питання теорії та практики:
Р 64 матеріали IV Всеукраїнської студентської наукової конференції, м. Львів, 17 листопада, 2023 рік / ГО «Молодіжна наукова ліга». — Вінниця: ТОВ «УКРЛОГОС Груп», 2023. — 526 с.

ISBN 978-617-8126-72-8

DOI 10.36074/liga-ukr-17.11.2023

Викладено матеріали учасників IV Всеукраїнської мультидисциплінарної студентської наукової конференції «Розвиток сучасної науки: актуальні питання теорії та практики», яка відбулася 17 листопада 2023 року у місті Львів, Україна.

УДК 082:001

© Колектив учасників конференції, 2023

© ГО «Молодіжна наукова ліга», 2023

© ТОВ «УКРЛОГОС Груп», 2023

ISBN 978-617-8126-72-8

СЕКЦІЯ 20. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ

CSS СТИЛІ У ВЕБ-РОЗРОБЦІ Череп Д.В., Науковий керівник: Шибко О.М.	236
АВТОМАТИЗОВАНЕ РОБОЧЕ МІСЦЕ Ковтун Д.В., Науковий керівник: Шибко О.М.	239
АНАЛІЗ ВИСОКОНАВАНТАЖЕНИХ АРІ ДЛЯ МОБІЛЬНИХ ЗАСТОСУНКІВ З ЕЛЕКТРОННОЇ КОМЕРЦІЇ Бондарчук М.О., Науковий керівник: Дудка О.О.	241
АНАЛІЗ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ОРГАНІЗАЦІЇ Некрасов Є.М., Науковий керівник: Шибко О.М.	245
АНАЛІЗ СУЧАСНОГО РІВНЯ РОЗВИТКУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ А/В ТЕСТУВАННЯ Бугайов В.Ю., Науковий керівник: Козловський А.В.	247
ВИБІР ТА ОПТИМІЗАЦІЯ КОМПОНЕНТІВ АЛГОРИТМУ ГІЛОК ТА МЕЖУВАННЯ В ЗАДАЧАХ ПЛАНУВАННЯ ПРОЕКТІВ Сікач Б., Науковий керівник: Саченко О.А.	249
ВИКОРИСТАННЯ КЛАСИФІКАЦІЇ ТА КЛАСТЕРИЗАЦІЇ ДЛЯ КЕРУВАННЯ ВИКИДАМИ ТА МОДЕЛЮВАННЯ ВІДСУТНІХ ДАНИХ РАДІОЛОКАЦІЙНИХ СИГНАЛІВ Павленко Є.М., Науковий керівник: Татарников А.О.	251
ВИКОРИСТАННЯ НЕЙРОННОЇ МЕРЕЖІ MASK R-CNN ДЛЯ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ Поначевний Ю.А., Науковий керівник: Корабльов М.М.	253
ВИКОРИСТАННЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ ДЛЯ РОЗРОБКИ ГЕЙМІФІКОВАНОЇ МЕТОДИКИ НАВЧАННЯ Стельмашенко А.В., Горбатюк М.В.	256
ВИКОРИСТАННЯ ХЕШ-ФУНКЦІЙ ДЛЯ ЗАХИСТУ ДАНИХ У ХМАРНИХ СХОВИЩАХ Кушов О.С., Науковий керівник: Дудка О.О.	259
ЕНЕРГОЕФЕКТИВНА СИСТЕМА НА ОСНОВІ ІНТЕРНЕТУ РЕЧЕЙ Грабовенко І.В., Науковий керівник: Осолінський О.Р.	262
ІМІТАЦІЙНА МОДЕЛЬ ПРОЦЕСУ СТРАТЕГІЧНОГО УПРАВЛІННЯ Ганець Р., Науковий керівник: Лендюк Т.В.	266
ІНТЕГРАЦІЯ СТОРОННІХ СЕРВІСІВ У ІНФОРМАЦІЙНІ СИСТЕМИ Степаненко М.І., Науковий керівник: Устенко С.В.	269

Сікач Богдан, здобувач вищої освіти факультету
комп'ютерних інформаційних технологій
Західноукраїнський національний університет, Україна

Науковий керівник: Саченко Олег Анатолійович, канд. техн. наук, доцент,
доцент кафедри інформаційно-обчислювальних систем та управління
Західноукраїнський національний університет, Україна

ВИБІР ТА ОПТИМІЗАЦІЯ КОМПОНЕНТІВ АЛГОРИТМУ ГІЛОК ТА МЕЖУВАННЯ В ЗАДАЧАХ ПЛАНУВАННЯ ПРОЕКТІВ

У науковому дослідженні Coelho & Vanhoucke (2018) [1] було проведено об'єднання різноманітних компонентів для створення різних конфігурацій для процесів розгалуження та межування. Втім, враховуючи, що результати, отримані за допомогою декількох конфігурацій, були практично ідентичними, ми вирішили не включати всі можливі конфігурації у наше дослідження. Замість цього ми обрали найбільш ефективні конфігурації, які були отримані під час початкового дослідження. Це призвело до відібрання 48 конфігурацій, які включають 1 стратегію дерева, 2 стратегії пошуку, 3 схеми розгалуження, 4 складені нижні межі і 2 рівні розгалуження. В даний момент ми надамо короткий огляд чотирьох основних компонентів, які застосовуються у нашому алгоритмі В&В. Ці компоненти будуть описані в наступних абзацах.

Перший компонент - "Стратегія дерева". Ця стратегія визначає, як обирати вузли для переходу від пошуку до знаходження оптимального рішення. Один підхід полягає в тому, щоб спочатку в глибину досліджувати дерево, починаючи з кореневого вузла та вибираючи вузли на основі певних критеріїв, таких як нижня межа. Другий підхід полягає в тому, щоб вибирати найбільш перспективні вузли для розгалуження, незалежно від рівня, на якому вони знаходяться в дереві. У нашому дослідженні перевага була віддана першому підходу, оскільки він показав кращі результати.

Другий компонент - "Стратегія пошуку". Цей компонент використовує два різні методи: стратегію верхньої межі (U) і стратегію мінімальної нижньої межі (L), запропоновані Patterson & Huber (1974) [2].

Третій компонент - "Схема розгалуження". Ця схема визначає, як будувати вузли на кожному рівні дерева. У нашому дослідженні було реалізовано три різні схеми розгалуження: розгалуження за часом початку діяльності (A), послідовне розгалуження (S) і паралельне розгалуження (P).

Четвертий компонент - "Композитна стратегія нижньої межі". Ця стратегія була розроблена Coelho & Vanhoucke (2018) [1] для поєднання швидкості та якості визначення нижніх меж. Вона включає список можливих нижніх меж, які можуть бути використані, і динамічно визначає, які з них використовувати на кожному вузлі дерева. Ця стратегія використовує кредитну систему для автоматичного увімкнення або вимкнення певних нижніх меж на основі критеріїв, щоб уникнути зайвого обчислення, коли використовуються вдосконалені стратегії нижньої межі. У нашому дослідженні ми використовували чотири ефективні складені стратегії нижньої межі, позначені як CLB0, CLB4, CLB8 і CLB12, кожна з яких включає різні складові для

покращення точності оцінки нижньої межі.

Отже, у даному дослідженні розглядається оптимізація стратегій розгалуження та межування в алгоритмах вирішення задачі планування проектів. Дослідження включало об'єднання різних компонентів та відбір найефективніших конфігурацій. Автори детально розглянули стратегії дерева, стратегії пошуку, схеми розгалуження та композитну стратегію нижньої межі. Результати показали, що деякі стратегії перевершують інші в контексті вирішення задачі планування проектів. Дослідження розкриває важливий внесок в оптимізацію алгоритмів для підвищення точності та швидкості розв'язання цієї задачі.

Список використаних джерел:

1. Coelho, J., & Vanhoucke, M. (2018). An exact composite lower bound strategy for the resource-constrained project scheduling problem. *Computers & Operations Research*, 93, 135-150.
2. Patterson, J. H., & Huber, W. D. (1974). A horizon-varying, zero-one approach to project scheduling. *Management Science*, 20(6), 990-998.