МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ЕКОНОМІЧНИЙ ФАКУЛЬТЕТ
КАФЕДРА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ

Кваліфікаційна робота

магістра

на тему МОДЕЛЮВАННЯ ПРОЦЕСУ РОЗПОДІЛУ РЕСУРСІВ У МУЛЬТИПРОЄКТНОМУ СЕРЕДОВИЩІ АУТСОРСИНГОВОЇ КОМПАНІЇ

Виконав: студент II курсу, групи 8.0514-ек
спеціальності 051 «Економіка»
освітньої програми «Економічна кібернетика»
Бондарь Лев Станіславович
Керівник: доцент кафедри економічної кібернетики,
к.е.н., доц. Чеверда С. С.
Рецензент доцент кафедри економічної
кібернетики, к.е.н., доц., Очеретін Д.В.

Запоріжжя
2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет	економічний
Кафедра	економічної кібернетики
Рівень вищої освіти	магістр
Спеціальність	051 «Економіка»
Освітня програма	«Економічна кібернетика»
Спеціалізація	

ЗАТВЕРДЖУЮ
Завідувач кафедри _____
(підпис)
«____»_____2025 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Бондарь Лев Станіславович

1. Тема роботи Моделювання процесу розподілу ресурсів у мультипроєктному середовищі аутсорсингової компанії
керівник роботи – Чеверда Сергій Сергійович, к.е.н.
затверджені наказом ЗНУ від 25 березня 2025 року № 439-с
2. Строк подання студентом роботи: 01 грудня 2025 року.
3. Вихідні дані до роботи: постановка завдання, наукова література за темою роботи, офіційні статистичні дані
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): Дослідити задачі розподілу ресурсів у мультипроєктному середовищі: розробити та впровадити модель оптимізації на базі гібридного алгоритму для мінімізації затримок проєктів та балансування завантаження команди аутсорсингової компанії.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): рисунки, таблиці, презентація.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 1	к.е.н. Чеверда С.С.	10.02.2025	24.05.2025
Розділ 2	к.е.н. Чеверда С.С.	26.05.2025	06.07.2025
Розділ 3	к.е.н. Чеверда С.С.	01.09.2025	22.11.2025

7. Дата видачі завдання 10.02.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Формування мети, об'єкту, предмету та завдань кваліфікаційної роботи	27.01.2025 – 02.02.2025	Виконано
2.	Узгодження плану роботи, підготовка вступу	03.02.2025 – 09.02.2025	Виконано
3.	Формування основних гіпотез та огляд наукових джерел	10.02.2025 – 05.04.2025	Виконано
4.	Підготовка 1-го розділу кваліфікаційної роботи	07.04.2025 – 24.05.2025	Виконано
5.	Підготовка 2-го розділу кваліфікаційної роботи	26.05.2025 – 06.07.2025	Виконано
6.	Підготовка 3-го розділу кваліфікаційної роботи	01.09.2025 – 22.11.2025	Виконано
7.	Аналіз та інтерпретація отриманих результатів, оформлення висновків кваліфікаційної роботи, оформлення вступу та реферату	24.11.2025 – 29.11.2025	Виконано
8	Нормоконтроль, перевірка на доброчесність, підготовка до захисту	01.12.2025 – 13.12.2025	Виконано

Студент _____ Бондарь Л.С.
(підпис)

Керівник роботи (проєкту) _____ Чеверда С.С.
(підпис)

Нормоконтроль пройдено

Нормоконтролер _____ Лось В.О.
(підпис)

РЕФЕРАТ

Кваліфікаційна робота магістра містить три розділи, 125 с., 10 рис., 24 табл., 86 джерел.

Об’єкт дослідження – процес розподілу та управління ресурсами у мультипроєктному середовищі аутсорсингової компанії

Предмет дослідження – економіко-математичні методи та моделі оптимізації розподілу ресурсів між проєктами в мультипроєктному середовищі аутсорсингової компанії.

Мета роботи – розробка гібридної моделі оптимального розподілу ресурсів у мультипроєктному середовищі аутсорсингової компанії на базі інтеграції еволюційних методів та багатокритеріальної оптимізації для підвищення ефективності планування, скорочення затримок проєктів та забезпечення збалансованого завантаження команди.

Методи дослідження – науковий аналіз, порівняння і групування даних, узагальнення і систематизація, нечітке моделювання, гібридні методи.

Кваліфікаційна робота магістра присвячена розробці математичної моделі та алгоритму оптимізації розподілу ресурсів у мультипроєктному середовищі аутсорсингової IT-компанії. Об'єктом дослідження є процес розподілу людських ресурсів між проєктами, предметом – методи багатокритеріальної оптимізації розподілу багатокомпетентних спеціалістів. Розроблено математичну модель класу MS-RCMPSP з чотирма цільовими функціями та гібридний алгоритм GA-PSO з механізмом NSGA-II для формування Парето-оптимальних рішень. Реалізовано програмне забезпечення з використанням Python, React та PostgreSQL з інтеграцією з Jira. Верифікація на даних ТОВ «АВЕДсофт» підтвердила скорочення затримок проєктів на 76% та економічний ефект з терміном окупності 2,8 місяці.

АУТСОРСИНГ, МУЛЬТИПРОЄКТНЕ СЕРЕДОВИЩЕ, РОЗПОДІЛ РЕСУРСІВ, БАГАТОКРИТЕРІАЛЬНА ОПТИМІЗАЦІЯ, ГЕНЕТИЧНИЙ АЛГОРИТМ, МЕТОД РОЮ ЧАСТОК, NSGA-II, БАГАТОКОМПЕТЕНТНІСТЬ, АУТСОРСИНГОВА КОМПАНІЯ, ФРОНТ ПАРЕТО, УПРАВЛІННЯ ПРОЄКТАМИ

SUMMARY

The master's qualification thesis contains three chapters, 125 pages, 10 figures, 24 tables, and 86 references.

Object of research – the process of resource allocation and management in a multi-project environment of an outsourcing company.

Subject of research – economic and mathematical methods and models for optimizing resource allocation among projects in a multi-project environment of an outsourcing company.

Purpose of the thesis – development of a hybrid model for optimal resource allocation in a multi-project environment of an outsourcing company based on the integration of evolutionary methods and multi-criteria optimization to increase planning efficiency, reduce project delays, and ensure a balanced team workload.

Research methods – scientific analysis, data comparison and grouping, generalization and systematization, fuzzy modeling, hybrid methods.

The master's qualification thesis is devoted to the development of a mathematical model and algorithm for optimizing resource allocation in a multi-project environment of an IT outsourcing company. The object of research is the process of human resource allocation among projects; the subject is multi-criteria optimization methods for allocating multi-skilled specialists. A mathematical model of the MS-RCMPSP class with four objective functions and a hybrid GA-PSO algorithm with NSGA-II mechanism for generating Pareto-optimal solutions have been developed. Software has been implemented using Python, React, and PostgreSQL with Jira integration. Verification on data from AVEDsoft LLC confirmed a 76% reduction in project delays and an economic effect with a payback period of 2.8 months.

OUTSOURCING, MULTI-PROJECT ENVIRONMENT, RESOURCE ALLOCATION, MULTI-CRITERIA OPTIMIZATION, GENETIC ALGORITHM, PARTICLE SWARM OPTIMIZATION, NSGA-II, MULTI-SKILL, OUTSOURCING COMPANY, PARETO FRONT, PROJECT MANAGEMENT

ЗМІСТ

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
РЕФЕРАТ
SUMMARY
ВСТУП.....5
РОЗДІЛ 1 ТЕОРЕТИЧНІ ЗАСАДИ УПРАВЛІННЯ РЕСУРСАМИ У МУЛЬТИПРОЄКТНОМУ СЕРЕДОВИЩІ АУТСОРСИНГОВОЇ КОМПАНІЇ.....9
1.1 Сутність та особливості діяльності аутсорсингових компаній в умовах мультипроєктного управління..... 9
1.2 Характеристика ресурсів та специфіка їх розподілу в умовах мультипроєктного середовища..... 19
1.3 Проблеми та виклики управління ресурсами при виконанні множини проєктів 30
РОЗДІЛ 2 ІНСТРУМЕНТАРІЙ МОДЕЛЮВАННЯ РОЗПОДІЛУ РЕСУРСІВ У МУЛЬТИПРОЄКТНОМУ СЕРЕДОВИЩІ 41
2.1 Аналіз існуючих методів та моделей оптимізації розподілу ресурсів між проєктами..... 41
2.2 Математичні підходи до формалізації задачі розподілу ресурсів в умовах обмежень..... 52
2.3 Обґрунтування вибору методів моделювання для задачі розподілу ресурсів в аутсорсинговій компанії..... 63
РОЗДІЛ 3 РОЗРОБКА ТА АПРОБАЦІЯ МОДЕЛІ ОПТИМАЛЬНОГО РОЗПОДІЛУ РЕСУРСІВ В АУТСОРСИНГОВІЙ КОМПАНІЇ..... 75
3.1 Розробка моделі розподілу ресурсів у мультипроєктному середовищі 75
3.2 Програмна реалізація моделі та алгоритму оптимізації розподілу ресурсів..... 87
3.3 Верифікація моделі на прикладі аутсорсингової компанії та оцінка економічної ефективності впровадження..... 98
ВИСНОВКИ 111
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 115

ВСТУП

У сучасних умовах глобальної цифрової трансформації та викликів, спричинених російсько-українською війною, ефективне управління ресурсами набуває особливого значення для забезпечення конкурентоспроможності та виживання бізнесу. Діджиталізація економіки України, яка стрімко прискорилася після початку повномасштабної агресії, створює як нові можливості, так і додаткові виклики для компаній, що працюють у сфері інформаційних технологій та аутсорсингу. Український IT-сектор, попри складні умови воєнного стану, продовжує демонструвати стійкість та зростання, залишаючись одним із ключових експортерів послуг та джерелом валютних надходжень до країни.

Аутсорсингові компанії, які становлять значну частину вітчизняної IT-індустрії, стикаються з необхідністю одночасного виконання множини проєктів із обмеженим пулом висококваліфікованих спеціалістів. По-перше, забезпечення безперервності бізнес-процесів в умовах енергетичних криз, вимушеної міграції фахівців та постійних загроз безпеці вимагає гнучкого перерозподілу ресурсів між проєктами. По-друге, підтримання високих стандартів якості послуг та дотримання жорстких термінів виконання для міжнародних замовників потребує оптимального балансування навантаження команди. По-третє, мінімізація витрат в умовах економічної нестабільності та зростаючих операційних витрат стає критичним фактором виживання компаній на конкурентному ринку.

Ефективний розподіл обмежених ресурсів між паралельними проєктами є ключовим фактором успішної діяльності сучасних аутсорсингових компаній. В умовах високої конкуренції на ринку IT-послуг здатність оптимально планувати завантаження спеціалістів, мінімізувати затримки проєктів та забезпечувати рівномірне навантаження команди стає критичною конкурентною перевагою.

Згідно з дослідженнями Project Management Institute, близько 70% IT-проєктів не вкладаються у початково визначені терміни або бюджет, причому основною причиною є неефективний розподіл ресурсів та конфлікти пріоритетів між проєктами. Як зазначають дослідники K. Deb, A. Pratap та інші науковці, традиційні методи планування ресурсів, зокрема діаграми Ганта та евристичні правила пріоритетів, не гарантують оптимальності розв'язків у складних мультипроєктних середовищах. Водночас стрімкий розвиток метаевристичних алгоритмів та методів штучного інтелекту відкриває нові можливості для автоматизованого пошуку оптимальних розкладів виконання проєктів.

Стрімкий розвиток еволюційних обчислень та ройового інтелекту відкриває нові горизонти для вдосконалення систем планування ресурсів. Генетичні алгоритми забезпечують ефективний глобальний пошук у просторі можливих рішень, алгоритм рою частинок (PSO) демонструє високу швидкість збіжності до локальних оптимумів, а методи багатокритеріальної оптимізації NSGA-II дозволяють формувати множину Парето-оптимальних рішень для задач з конфліктуючими цілями. Гібридизація цих підходів дає можливість поєднати їх переваги та отримати рішення, які перевершують результати застосування окремих алгоритмів.

Особливої актуальності застосування інтелектуальних алгоритмів оптимізації набуває в умовах географічної розподіленості команд, що є типовою характеристикою сучасних аутсорсингових компаній, особливо в умовах війни та вимушеної релокації фахівців. Автоматизовані системи оптимізації здатні аналізувати величезні обсяги інформації про компетенції ресурсів, залежності між завданнями, календарну доступність спеціалістів та формувати оптимальні розклади за лічені секунди, що неможливо досягти при ручному плануванні.

Поєднання викликів мультипроєктного управління, необхідності підтримання конкурентоспроможності українського IT-сектору в умовах війни та стрімкого розвитку метаевристичних алгоритмів обумовлює

актуальність дослідження можливостей застосування гібридних методів оптимізації для вдосконалення систем розподілу ресурсів в аутсорсингових компаніях. Розробка та впровадження інтелектуальних систем планування не лише підвищує операційну ефективність окремих компаній, але й сприяє зміцненню позицій української IT-індустрії на глобальному ринку, що має критичне значення для економічної стійкості країни в умовах воєнного стану та повоєнного відновлення.

Мета роботи: розробка гібридної моделі оптимального розподілу ресурсів у мультипроєктному середовищі аутсорсингової компанії на базі інтеграції еволюційних методів та багатокритеріальної оптимізації для підвищення ефективності планування, скорочення затримок проєктів та забезпечення збалансованого завантаження команди.

Для досягнення мети виконано такі завдання:

– проаналізовано специфіку діяльності аутсорсингових IT-компаній та особливості мультипроєктного управління ресурсами;

– досліджено та систематизовано існуючі підходи до моделювання задач розподілу ресурсів класу MS-RCMPSP;

– розроблено математичну модель багатокритеріальної оптимізації розподілу ресурсів;

– розроблено гібридний алгоритм GA-PSO з інтеграцією багатокритеріальної оптимізації NSGA-II для формування фронту Парето;

– здійснено програмну реалізацію системи оптимізації та проведено верифікацію розробленої моделі на даних аутсорсингової компанії та оцінено економічну ефективність впровадження.

Наукова новизна кваліфікаційної роботи полягає у розробці гібридної моделі оптимізації розподілу ресурсів у мультипроєктному середовищі, яка на відміну від існуючих підходів інтегрує генетичний алгоритм з алгоритмом рою частинок та багатокритеріальну оптимізацію NSGA-II для одночасної мінімізації тривалості портфеля, вартості ресурсів та балансування навантаження з формуванням множини Парето-оптимальних рішень.

Практичне значення отриманих результатів полягає у можливості використання розробленої моделі оптимізації аутсорсинговими IT-компаніями для підвищення ефективності розподілу обмежених людських ресурсів між паралельними проєктами, скорочення затримок виконання, балансування навантаження команди та покращення якості обслуговування замовників, що забезпечує конкурентні переваги на ринку IT-послуг.

Результати кваліфікаційної роботи були представлені:

1. Бондарь Л.С., Чеверда С.С. Ресурсоефективність мультипроєктного управління в аутсорсингових компаніях як стратегічний інструмент соціально-економічної стабільності в умовах кризи. *Управління соціально-економічним розвитком регіонів і держави* : збірник матеріалів XIX Міжнародної науково-практичної конференції / за ред. А. В. Череп Запоріжжя: Запорізький національний університет, 2025. С. 161 –164.

2. Бондарь Л.С., Чеверда С.С. Інтелектуальне управління ресурсами аутсорсингової компанії: моделі прогнозування та розподілу. В кн.: Інтеграція освіти, науки і бізнесу : монографія / За загальною редакцією Череп А. В. Запоріжжя: Запорізький національний університет. 2025 С.

3. Бондарь Л.С., Чеверда С.С. Гібридний підхід до планування ресурсів: комбінація евристичних та точних методів оптимізації для мультипроєктного середовища. *Моделювання економіки: проблеми, тенденції, досвід* : Тези доповідей. XIII Всеукраїнська науково-практична конференція Форум молодих економістів-кібернетиків. 21-22 листопада 2025 р. м. Львів 2025. С. 50 –52.

4. Бондарь Л.С., Чеверда С.С. Економіко-математична модель оптимізації портфелю проєктів з урахуванням обмежених ресурсів та термінів виконання *«Функціонування суб'єктів економічної діяльності: виклики сучасності»* : матеріали Всеукр. наук.-практ. інтернет-конф. з міжнар. участю, 20–21 листопада 2025 р. Харкі. ХНУМГ імені О.М. Бекетова, 2025. С. 112 –115.

РОЗДІЛ 1
ТЕОРЕТИЧНІ ЗАСАДИ УПРАВЛІННЯ РЕСУРСАМИ У МУЛЬТИПРОЄКТНОМУ СЕРЕДОВИЩІ АУТСОРСИНГОВОЇ КОМПАНІЇ

1.1 Сутність та особливості діяльності аутсорсингових компаній в умовах мультипроєктного управління

Сучасне економічне середовище характеризується високим рівнем конкуренції, динамічними змінами ринкових умов та постійним зростанням вимог до ефективності бізнес-процесів. У таких умовах підприємства активно шукають шляхи оптимізації своєї діяльності, зниження витрат та концентрації на ключових компетенціях. Одним із найбільш дієвих інструментів досягнення цих цілей є аутсорсинг – передача виконання окремих бізнес-функцій або процесів зовнішнім спеціалізованим компаніям [1]. Аутсорсинг як стратегічний підхід до організації бізнесу набув широкого поширення у світовій практиці та продовжує демонструвати стійку тенденцію до зростання, що зумовлено глобалізацією економіки, розвитком інформаційних технологій та посиленням спеціалізації компаній.

Термін «аутсорсинг» походить від англійського словосполучення «outside resource using», що дослівно перекладається як «використання зовнішніх ресурсів» [2]. За своєю сутністю аутсорсинг являє собою форму стратегічного партнерства, за якої організація передає виконання певних функцій або бізнес-процесів зовнішньому підряднику, що спеціалізується на відповідному виді діяльності. При цьому компанія-замовник отримує можливість сконцентрувати власні ресурси на профільній діяльності, тоді як непрофільні функції виконуються більш ефективно та з меншими витратами спеціалізованими провайдерами [3]. Такий підхід дозволяє підприємствам зосередитися на своїх конкурентних перевагах та стратегічному розвитку,

водночас отримуючи доступ до передових технологій та експертизи у непрофільних сферах.

Відповідно до визначення IBM, аутсорсинг бізнес-процесів (Business Process Outsourcing, BPO) – це практика передачі виконання певних бізнес-функцій або процесів зовнішнім постачальникам послуг, що дозволяє компаніям зосередитися на своїх основних компетенціях та стратегічних ініціативах [4]. Головною метою аутсорсингу є підвищення операційної ефективності та зниження витрат, оскільки спеціалізовані компанії, що надають аутсорсингові послуги, зазвичай мають економію на масштабі, передові технології та галузеву експертизу, що дозволяє їм виконувати відповідні функції більш ефективно та з меншими витратами, ніж замовник міг би зробити це самостійно.

Історія розвитку аутсорсингу як бізнес-практики бере свій початок з середини XX століття, коли компанії почали передавати окремі виробничі операції зовнішнім підрядникам для зниження витрат. Однак справжній розквіт аутсорсингу відбувся в 1990-х роках з появою глобальних телекомунікаційних мереж та розвитком інформаційних технологій, що зробило можливим ефективну співпрацю з географічно віддаленими партнерами [5]. Сьогодні глобальний ринок аутсорсингових послуг оцінюється в сотні мільярдів доларів та продовжує демонструвати стійке зростання, охоплюючи практично всі галузі економіки та функціональні напрямки бізнесу.

Еволюція аутсорсингу пройшла декілька етапів. Спочатку аутсорсинг використовувався переважно у виробничій сфері, де компанії передавали зовнішнім підрядникам виконання окремих операцій з управління ланцюгами постачання. Згодом аутсорсинг поширився на сферу послуг, охопивши такі функції, як бухгалтерський облік, управління персоналом, маркетинг та інформаційні технології [5]. Сьогодні аутсорсингові послуги використовуються у найрізноманітніших галузях, включаючи охорону здоров'я, управління активами, енергетику, фармацевтику, електронну

комерцію та інші. При цьому спостерігається тенденція до аутсорсингу все більш складних та стратегічно важливих функцій, що виходить за межі традиційного підходу до передачі лише непрофільної діяльності.

Класифікація аутсорсингу здійснюється за різними критеріями. За географічним розташуванням постачальника послуг виділяють такі види аутсорсингу [4, 6]:

– оншорний (внутрішній, domestic) аутсорсинг – передача функцій постачальнику, що знаходиться в тій самій країні, що й замовник. Такий підхід забезпечує мінімізацію культурних та мовних бар'єрів, спрощує комунікацію та полегшує дотримання нормативних вимог. Оншорний аутсорсинг особливо популярний у сферах, що вимагають тісної взаємодії з замовником або суворого дотримання місцевого законодавства;

– ніршорний (nearshore) аутсорсинг – залучення постачальника з сусідньої країни або регіону з подібним часовим поясом. Цей варіант дозволяє поєднати переваги географічної близькості з можливістю зниження витрат. Ніршорний аутсорсинг забезпечує легшу комунікацію завдяки перекриванню робочих годин та культурній близькості;

– офшорний (offshore) аутсорсинг – передача функцій постачальнику в іншій, зазвичай віддаленій, країні. Основною мотивацією офшорного аутсорсингу є суттєве зниження витрат за рахунок різниці в оплаті праці, а також доступ до глобального пулу талантів та можливість забезпечення цілодобової роботи завдяки різниці в часових поясах. Країни Південної та Східної Азії, а також Східної Європи традиційно є популярними напрямками для офшорного аутсорсингу.

За функціональною спрямованістю аутсорсинг поділяється на [5, 7]:

– аутсорсинг бек-офісних функцій (back-office BPO) – передача внутрішніх бізнес-функцій, що не пов'язані безпосередньо з обслуговуванням клієнтів, таких як бухгалтерський облік, управління документообігом, управління персоналом, розрахунок заробітної плати. Ці функції є

допоміжними, але критично важливими для забезпечення нормального функціонування підприємства;

– аутсорсинг фронт-офісних функцій (front-office BPO) – передача функцій, орієнтованих на клієнтів, включаючи обслуговування клієнтів, технічну підтримку, продажі та маркетинг. Цей вид аутсорсингу вимагає особливої уваги до якості, оскільки безпосередньо впливає на сприйняття компанії клієнтами;

– аутсорсинг процесів на основі знань (Knowledge Process Outsourcing, KPO) – передача функцій, що вимагають спеціалізованих знань та експертизи, таких як дослідження та розробки, аналіз даних, юридичні послуги, консалтинг. KPO є найбільш складним та високовартісним сегментом аутсорсингового ринку.

Особливе місце в структурі аутсорсингового ринку посідає IT-аутсорсинг, який охоплює широкий спектр послуг у сфері інформаційних технологій: розробку програмного забезпечення, технічну підтримку, управління IT-інфраструктурою, забезпечення кібербезпеки, хмарні сервіси тощо [8]. IT-аутсорсинг став одним із найбільш динамічних сегментів глобального ринку аутсорсингових послуг, що зумовлено стрімким розвитком цифрових технологій та їх проникненням у всі сфери бізнесу. За різними оцінками, IT-сектор займає найбільшу частку на ринку аутсорсингових послуг, що пояснюється високою потребою у технологічних компетенціях та постійним зростанням складності інформаційних систем.

Україна займає помітне місце на світовому ринку IT-аутсорсингу, виступаючи одним із провідних експортерів комп'ютерних послуг. За даними досліджень, IT-індустрія є однією з найбільш стійких галузей української економіки, що продовжує демонструвати зростання навіть в умовах воєнного стану [9]. Експорт IT-послуг з України впродовж останніх двадцяти років демонструє стрімке експоненційне зростання, перевищивши позначку в 5 мільярдів доларів. Українські IT-компанії обслуговують клієнтів зі США,

Західної Європи та інших регіонів, надаючи послуги з розробки програмного забезпечення, забезпечення якості, технічної підтримки та інших напрямків.

Серед ключових переваг України як локації для IT-аутсорсингу дослідники виділяють: високий рівень технічної освіти та кваліфікації спеціалістів, конкурентну вартість послуг, культурну близькість до західних замовників, зручний географічний та часовий пояс для співпраці з європейськими клієнтами [10]. Водночас, дослідники звертають увагу на виклики, з якими стикається українська IT-індустрія: воєнний стан, демографічні проблеми, економічна нестабільність, питання комплаєнс-контролю та етичного ведення бізнесу, валютне регулювання та обмеження НБУ. Ці фактори створюють унікальне середовище для діяльності аутсорсингових компаній та потребують особливих підходів до управління ризиками та ресурсами.

Характерною особливістю діяльності сучасних аутсорсингових компаній є робота в умовах мультипроєктного середовища. Мультипроєктне управління (Multi-Project Management) – це методологія управління, що передбачає одночасне виконання декількох проєктів в рамках однієї організації з використанням спільного пулу ресурсів [11]. На відміну від управління окремим проєктом, мультипроєктне управління потребує координації та балансування ресурсів, пріоритетів і термінів між різними проєктами, що створює додаткові виклики для менеджменту. Складність мультипроєктного управління зростає нелінійно зі збільшенням кількості одночасно виконуваних проєктів.

Мультипроєктне управління тісно пов'язане з концепціями управління програмами та портфелями проєктів. Програма – це група пов'язаних проєктів, що координуються разом для отримання вигод, недосяжних при окремому управлінні кожним проєктом. Портфель – це сукупність проєктів, програм та операцій, що управляються як група для досягнення стратегічних цілей організації. В контексті аутсорсингових компаній портфель зазвичай включає

множину проєктів для різних замовників, що виконуються паралельно та конкурують за обмежені ресурси організації.

Відповідно до PMBOK Guide (Project Management Body of Knowledge), ресурси проєкту включають людські ресурси, обладнання, матеріали та інші активи, необхідні для виконання проєктних робіт [12]. В умовах мультипроєктного середовища ці ресурси розподіляються між декількома проєктами, що вимагає ретельного планування та оптимізації їх використання. Ключовими процесами управління ресурсами згідно PMBOK є: планування управління ресурсами, оцінка ресурсів операцій, набуття ресурсів, розвиток команди, управління командою та контроль ресурсів [13]. Кожен з цих процесів набуває додаткової складності в мультипроєктному контексті.

Сьома редакція PMBOK Guide, опублікована у 2021 році, представляє значний зсув у проєктному мисленні – від процесно-орієнтованого до принципно-орієнтованого підходу. Новий стандарт структурований навколо 12 принципів управління проєктами та 8 областей продуктивності проєктів, що дозволяє більш гнучко адаптувати практики управління до конкретного контексту та потреб організації. Цей підхід особливо релевантний для аутсорсингових компаній, що працюють з різноманітними проєктами та замовниками з різними вимогами та очікуваннями.

Аутсорсингові компанії, особливо в IT-сфері, типово працюють одночасно над декількома проєктами для різних замовників. Така модель роботи зумовлена необхідністю забезпечення стабільного завантаження персоналу, диверсифікації ризиків та максимізації прибутковості. При цьому кожен проєкт має свої унікальні вимоги щодо термінів, бюджету, якості та компетенцій виконавців [14]. Управління такими мультипроєктними середовищами потребує застосування спеціалізованих методів та інструментів, що дозволяють ефективно розподіляти обмежені ресурси між конкуруючими проєктами.

Згідно з дослідженнями, проблема розподілу ресурсів в умовах обмежень (Resource-Constrained Multi-Project Scheduling Problem, RCMPSP) є

однією з ключових задач оптимізації в мультипроєктному управлінні [15]. Ця задача належить до класу NP-складних оптимізаційних проблем, що означає неможливість знайти точне рішення для великих екземплярів за прийнятний час з використанням класичних методів. Для вирішення задач RCMPSP застосовуються різноманітні підходи: евристичні методи на основі правил пріоритетів, метаевристичні алгоритми (генетичні алгоритми, імітація відпалу, рій часток), методи математичного програмування та гібридні підходи [16].

Особливістю мультипроєктного середовища в аутсорсингових компаніях є висока мінливість і невизначеність. Проєкти можуть з'являтися та завершуватися в довільні моменти часу, змінюватися за обсягом та термінами, потребувати різних компетенцій виконавців [17]. За даними звіту The Standish Group, лише близько 31% проєктів у світі завершуються вчасно, тоді як понад 50% зазнають затримок, переважно через необхідність переробки та неефективний розподіл ресурсів [18]. Це підкреслює важливість розробки ефективних моделей та методів управління ресурсами в мультипроєктних середовищах, що здатні враховувати невизначеність та адаптуватися до змін.

Сметанюк О.А. та Бондарчук А.В. у своєму дослідженні особливостей системи управління проєктами в IT-компаніях зазначають, що IT-компанії традиційно застосовують найсучасніші та найбільш ефективні технології управління, а проєктний підхід відіграє позитивну роль при прийнятті рішень у сфері інноваційних розробок [14]. Автори звертають увагу на принципове значення так званої «проєктної тріади» – обмежень, що накладаються на проєкт: час, бюджет, якість. В умовах мультипроєктного середовища ці обмеження стають ще більш критичними, оскільки ресурси, що використовуються для досягнення цілей одного проєкту, не можуть одночасно використовуватися в інших проєктах.

Важливим аспектом управління мультипроєктним середовищем є врахування багатокомпетентності людських ресурсів (Multi-Skill Resources). Сучасні дослідження показують, що раціональний розподіл працівників з

різними навичками та компетенціями може суттєво підвищити продуктивність та скоротити терміни виконання проєктів [19]. Задача календарного планування проєктів з урахуванням багатокомпетентності ресурсів (Multi-Skill Resource-Constrained Project Scheduling Problem, MSRCPSP) є розширенням класичної RCMPSP і потребує врахування не лише кількості ресурсів, але й їх кваліфікаційних характеристик. Для аутсорсингових IT-компаній ця проблематика є особливо актуальною, оскільки IT-спеціалісти часто володіють множиною різних технологічних компетенцій.

Ще однією характерною особливістю мультипроєктного управління в аутсорсингових компаніях є необхідність врахування часу та витрат на переміщення ресурсів між проєктами [20]. На відміну від теоретичних моделей, де ресурси можуть миттєво переходити від одного завдання до іншого, у реальній практиці існують затримки, пов'язані з передачею знань, контекстним перемиканням, географічним розташуванням команд тощо. Дослідники відзначають, що врахування часу переміщення ресурсів є критично важливим для отримання реалістичних планів проєктів, особливо в середовищах з географічно розподіленими командами, що є типовим для сучасних аутсорсингових компаній.

Систематичний огляд літератури, проведений Lopes et al., виявив, що близько 72% публікацій з проблематики RCMPSP не враховують невизначеність та гнучкість ресурсів [21]. Водночас, невизначеність може проявлятися як у тривалості виконання завдань, так і в доступності ресурсів. Тривалість виконання завдань може варіюватися через різні фактори, такі як складність, непередбачені проблеми або зміни у вимогах. Ця невизначеність у тривалості завдань безпосередньо впливає на розподіл ресурсів у часі. З іншого боку, зміни в доступності ресурсів також можуть призводити до коливань у часі, необхідному для виконання конкретних завдань. Для аутсорсингових компаній в Україні додатковим джерелом невизначеності є ситуація, пов'язана з воєнним станом.

Дослідження, проведене у галузі будівництва, виявило п'ять найбільш критичних викликів управління ресурсами в мультипроєктних середовищах: прийняття рішень, контекст проєкту, паралельні активності, планування та управління ресурсами, транспортування ресурсів [22]. Факторний аналіз дозволив виділити п'ять ключових вимірів, що сприяють виникненню цих викликів: можливості фірми та управління ресурсами, будівельні роботи та транспортування, управління персоналом, час і вартість, знання та цифровізація. Ці результати мають значну релевантність і для аутсорсингових компаній, де проблеми координації ресурсів між проєктами є не менш актуальними, хоча й мають свою специфіку.

Сучасні тенденції розвитку аутсорсингу тісно пов'язані з цифровою трансформацією та впровадженням нових технологій. Серед ключових трендів, що впливають на діяльність аутсорсингових компаній, дослідники виділяють: роботизовану автоматизацію процесів (RPA), штучний інтелект та машинне навчання, хмарні технології, посилення вимог до кібербезпеки, розвиток стратегічного партнерства [4]. Ці технології не лише змінюють характер послуг, що надаються аутсорсинговими компаніями, але й трансформують підходи до управління проєктами та ресурсами. Зокрема, штучний інтелект все частіше використовується для оптимізації розподілу ресурсів, прогнозування термінів виконання проєктів та виявлення потенційних ризиків.

Впровадження гнучких (Agile) методологій управління проєктами стало ще однією важливою тенденцією в діяльності аутсорсингових компаній, особливо в IT-секторі [23]. Методології Scrum, Kanban та інші Agile-підходи дозволяють більш гнучко реагувати на зміни вимог замовника, забезпечують прозорість виконання робіт та сприяють підвищенню задоволеності клієнтів. Водночас, застосування Agile в мультипроєктному середовищі потребує адаптації традиційних практик для забезпечення ефективної координації між командами та проєктами. Виникає потреба у так званому «масштабованому

Agile» (Scaled Agile), що поєднує гнучкість на рівні окремих команд з координацією на рівні організації.

В умовах війни в Україні вітчизняні IT-аутсорсингові компанії демонструють високу адаптивність та стійкість [9]. Незважаючи на виклики, пов'язані з релокацією персоналу, мобілізацією та нестабільністю інфраструктури, більшість компаній продовжують виконувати свої зобов'язання перед клієнтами та шукають нові ринки збуту. Важливими кроками з боку уряду стали впровадження стандартів кібербезпеки, підтримка бізнес-відряджень, механізми бронювання ключових працівників та підтримка релокації фахівців [10]. Це підкреслює стратегічне значення IT-індустрії для економіки України та необхідність розвитку ефективних механізмів управління в умовах невизначеності.

Управління портфелем ресурсів (Portfolio Resource Management, PRM) є важливим аспектом діяльності аутсорсингових компаній, що працюють у мультипроєктному середовищі [24]. PRM включає процеси, що дозволяють організації ефективно призначати відповідні ресурси (за кількістю та навичками) для успішного виконання проєктів у портфелі. Це допомагає забезпечити належний розподіл ресурсів організації відповідно до бізнес-потреб, а також надає керівництву інформацію для прогнозування майбутніх потреб у ресурсах. Дослідження показують, що найбільш значущими викликами в управлінні ресурсами портфеля є: планування та оцінка ресурсів, пріоритизація, розподіл та планування ресурсів, виконання, моніторинг та контроль ресурсів.

Дослідники виділяють п'ять рівнів зрілості управління ресурсами портфеля: від першого рівня, де організація лише визнає потребу в процесі управління ресурсами, до п'ятого рівня, де управління ресурсами повністю інтегроване зі стратегічним плануванням та постійно вдосконалюється на основі зворотного зв'язку. Більшість організацій знаходяться на нижчих рівнях зрілості, що вказує на значний потенціал для вдосконалення практик управління ресурсами в мультипроєктних середовищах.

Таким чином, аутсорсинг є стратегічним підходом до організації бізнесу, що дозволяє компаніям оптимізувати свою діяльність шляхом передачі непрофільних функцій спеціалізованим провайдерам. Аутсорсингові компанії, особливо в IT-сфері, типово працюють в умовах мультипроєктного середовища, що характеризується одночасним виконанням декількох проєктів з використанням спільного пулу ресурсів. Ефективне управління таким середовищем потребує застосування спеціалізованих методів та інструментів оптимізації розподілу ресурсів, врахування невизначеності та гнучкості, а також адаптації до сучасних тенденцій цифрової трансформації та гнучких методологій управління проєктами. Особливої актуальності ці питання набувають для українських IT-аутсорсингових компаній, що працюють в умовах додаткової невизначеності, пов'язаної з воєнним станом.

1.2 Характеристика ресурсів та специфіка їх розподілу в умовах мультипроєктного середовища

Ефективне управління ресурсами є фундаментальною складовою успішної реалізації проєктів в будь-якій організації. Особливої актуальності це набуває в умовах мультипроєктного середовища, де обмежені ресурси розподіляються між множиною проєктів, що виконуються паралельно. Згідно з PMBOK Guide, управління ресурсами проєкту охоплює процеси ідентифікації, набуття та управління ресурсами, необхідними для успішного завершення проєкту [25]. В аутсорсингових компаній, де одночасно виконується значна кількість проєктів для різних замовників, питання характеристики ресурсів та оптимізації їх розподілу набуває критичного значення для забезпечення конкурентоспроможності та прибутковості бізнесу.

Ресурси проєкту – це узагальнюючий термін, що охоплює всі активи, необхідні для виконання проєктних робіт та досягнення поставлених цілей. Відповідно до сучасних стандартів проєктного менеджменту, ресурси поділяються на дві основні категорії: фізичні ресурси та командні (людські)

ресурси [26]. Фізичні ресурси включають матеріали, обладнання, інфраструктуру та інші матеріальні активи, тоді як командні ресурси – це люди, залучені до виконання проєктних робіт. Важливо зазначити, що управління цими двома категоріями ресурсів потребує різних підходів та компетенцій, хоча обидві категорії є невід'ємними складовими успішної реалізації проєкту.

Розширена класифікація ресурсів проєкту включає п'ять основних типів [27]: люди (людські ресурси), активи (матеріальні та нематеріальні), споживані матеріали, дані та інформація, енергія. Кожен з цих типів має свої особливості управління, метрики вимірювання ефективності використання та специфіку застосування в різних галузях. Для аутсорсингових ІТ-компаній найбільш критичними є людські ресурси, оскільки саме компетенції та продуктивність спеціалістів визначають якість послуг та конкурентоспроможність компанії на ринку.

Детальна характеристика основних типів ресурсів проєкту та їх особливостей наведена в таблиці 1.1.

Таблиця 1.1 – Класифікація та характеристика ресурсів проєкту

Тип ресурсу	Характеристика	Особливості в ІТ-аутсорсингу
Людські ресурси	Члени команди, підрядники, консультанти. Характеризуються навичками, досвідом, доступністю	Ключовий ресурс. Багатокомпетентність, можливість роботи над кількома проєктами
Матеріальні активи	Обладнання, інструменти, інфраструктура. Можуть бути багаторазового використання	Комп'ютерна техніка, сервери, ліцензії на ПЗ, хмарна інфраструктура
Споживані матеріали	Ресурси одноразового використання, що витрачаються в процесі роботи	Хмарні обчислювальні ресурси за моделлю pay-as-you-go, тестові середовища
Фінансові ресурси	Бюджет проєкту, кошти на оплату праці, закупівлі та непередбачені витрати	Розподіл бюджетів між проєктами, облік витрат за замовниками
Часові ресурси	Доступний час для виконання завдань, терміни, дедлайни проєктів	Критичний ресурс через зобов'язання перед множиною замовників
Інформаційні ресурси	Дані, знання, документація, інтелектуальна власність	База знань, документація проєктів, репозиторії коду

Джерело: складено автором на основі [25, 27]

Людські ресурси є найбільш критичним типом ресурсів для аутсорсингових компаній. На відміну від фізичних ресурсів, люди мають унікальні характеристики: навички різного рівня, індивідуальну продуктивність, потребу в мотивації та розвитку, обмежену доступність через робочий час та інші зобов'язання [26]. Крім того, людські ресурси характеризуються багатокомпетентністю – один спеціаліст може володіти множиною навичок та виконувати різні типи завдань. Ця особливість є водночас перевагою та джерелом додаткової складності при плануванні та розподілі ресурсів між проєктами.

Процес управління ресурсами проєкту згідно PMBOK Guide включає шість основних процесів: планування управління ресурсами, оцінка ресурсів операцій, набуття ресурсів, розвиток команди, управління командою та контроль ресурсів [25]. Кожен з цих процесів має свої входи, інструменти та методи, а також виходи. План управління ресурсами є ключовим документом, що визначає, як ресурси проєкту мають бути категоризовані, розподілені, управлятися та вивільнятися. Цей план є компонентом загального плану управління проєктом та служить основою для прийняття рішень щодо ресурсів протягом усього життєвого циклу проєкту.

В умовах мультипроєктного середовища процес управління ресурсами набуває додаткової складності. Мультипроєктне середовище характеризується одночасним виконанням множини проєктів, що конкурують за спільний пул обмежених ресурсів [28]. Це створює ситуацію, коли рішення щодо розподілу ресурсів в одному проєкті безпосередньо впливають на інші проєкти. Основними характеристиками мультипроєктного середовища є: обмеженість ресурсів, залежності між проєктами, конфлікти пріоритетів, динамічні зміни у складі та обсягах проєктів.

Розподіл ресурсів у мультипроєктному середовищі передбачає вирішення задачі оптимального призначення обмежених ресурсів на множину завдань з різних проєктів таким чином, щоб максимізувати загальну ефективність при дотриманні обмежень за часом, бюджетом та якістю [29]. Ця

задача є значно складнішою, ніж розподіл ресурсів в рамках окремого проєкту, оскільки потребує врахування міжпроєктних залежностей, різних пріоритетів проєктів та глобальної оптимізації на рівні всього портфеля.

Для оптимізації розподілу ресурсів у проєктному середовищі застосовуються спеціалізовані техніки, серед яких ключову роль відіграють вирівнювання ресурсів (resource leveling) та згладжування ресурсів (resource smoothing) [30]. Ці техніки є частиною процесу розробки календарного плану проєкту та спрямовані на вирішення проблем перевантаження або нерівномірного використання ресурсів. Розуміння відмінностей між цими техніками є критично важливим для ефективного управління ресурсами.

Вирівнювання ресурсів – це техніка оптимізації ресурсів, яка передбачає коригування дат початку та завершення операцій з метою балансування попиту на ресурси з їх наявною пропозицією [30]. Основною метою вирівнювання є запобігання перевантаженню ресурсів, тобто ситуації, коли ресурсу призначено більше роботи, ніж він може виконати за доступний час. Важливою особливістю вирівнювання є те, що воно може призводити до зміни критичного шляху та збільшення тривалості проєкту. Ця техніка застосовується, коли головним обмеженням є доступність ресурсів, а не терміни виконання проєкту.

Згладжування ресурсів – це техніка, що передбачає коригування операцій календарного плану таким чином, щоб потреби в ресурсах не перевищували заздалегідь визначених лімітів, при цьому дата завершення проєкту залишається незмінною [31]. Для коригування використовується резерв часу (float) – проміжок часу, на який операція може бути відкладена без впливу на дату завершення проєкту. Згладжування застосовується, коли головним обмеженням є терміни виконання проєкту, а завданням є більш рівномірний розподіл навантаження на ресурси в межах існуючого графіка.

Порівняльна характеристика технік оптимізації розподілу ресурсів наведена в таблиці 1.2.

Таблиця 1.2 – Порівняння технік оптимізації розподілу ресурсів

Критерій порівняння	Вирівнювання ресурсів (Resource Leveling)	Згладжування ресурсів (Resource Smoothing)
Основне обмеження	Обмеження за ресурсами	Обмеження за часом
Вплив на дату завершення	Може змінюватися (зазвичай збільшується)	Залишається незмінною
Вплив на критичний шлях	Може змінюватися	Не змінюється
Використання резерву часу	Повне використання, можливий вихід за межі	В межах наявного резерву
Головна мета	Запобігання перевантаженню ресурсів	Рівномірний розподіл навантаження
Послідовність застосування	Застосовується першим	Застосовується після вирівнювання
Гнучкість графіка	Зберігається (резерв не витрачається)	Зменшується (резерв використовується)

Джерело: складено автором на основі [30, 31]

Окрім вирівнювання та згладжування, для оптимізації використання ресурсів застосовуються й інші методи. Метод критичного шляху (Critical Path Method, CPM) дозволяє визначити найдовшу послідовність залежних операцій та ідентифікувати завдання, затримка яких призведе до затримки всього проєкту [32]. Метод критичного ланцюга (Critical Chain Method, CCM) є розвитком методу критичного шляху та додатково враховує обмеження ресурсів, а також включає різні типи буферів для захисту від непередбачених обставин. Техніка стиснення графіка (crashing) передбачає виділення додаткових ресурсів для скорочення тривалості критичних операцій, тоді як швидкий прохід (fast tracking) дозволяє виконувати операції паралельно замість послідовного виконання.

Специфіка розподілу ресурсів у мультипроєктному середовищі аутсорсингової компанії визначається низкою факторів. По-перше, наявність множини замовників з різними пріоритетами та вимогами створює складну систему обмежень, що потребує постійного балансування. По-друге, динамічний характер надходження нових проєктів та завершення поточних вимагає гнучкого підходу до планування та оперативного перерозподілу ресурсів [28]. По-третє, високий рівень спеціалізації в IT-сфері означає, що не всі спеціалісти є взаємозамінними, що обмежує можливості перерозподілу.

Конфлікти ресурсів є типовою проблемою мультипроєктного середовища. Вони виникають, коли один і той самий ресурс потрібен для виконання завдань з різних проєктів в один і той самий період часу [29]. Основними причинами ресурсних конфліктів є: недостатня кількість спеціалістів з необхідними компетенціями, неузгодженість пріоритетів між проєктами, недосконале планування та прогнозування потреб у ресурсах, зміни обсягів та термінів проєктів. Неврахування ресурсних конфліктів призводить до затримок проєктів, перевантаження персоналу, зниження якості роботи та збільшення витрат.

Для вирішення конфліктів ресурсів у мультипроєктному середовищі застосовуються різні стратегії [32]: переговори із зацікавленими сторонами щодо коригування обсягів, бюджетів або термінів проєктів; пріоритизація проєктів на основі їх стратегічної цінності, терміновості або рівня ризику; перерозподіл або аутсорсинг ресурсів з урахуванням навичок, доступності та преференцій; найм додаткових ресурсів для заповнення прогалин або зменшення навантаження; координація з іншими керівниками проєктів для обміну або перерозподілу ресурсів; використання існуючих інструментів та інфраструктури для оптимізації використання ресурсів.

Основні виклики та методи їх вирішення при розподілі ресурсів у мультипроєктному середовищі систематизовано в таблиці 1.3.

Важливу роль в управлінні ресурсами мультипроєктного середовища відіграє офіс управління проєктами (Project Management Office, PMO). PMO може забезпечувати нагляд та координацію розподілу ресурсів між множиною проєктів, встановлювати та впроваджувати найкращі практики управління ресурсами, вирішувати конфлікти між проєктами [33]. Централізований підхід до управління ресурсами через PMO дозволяє уникнути локальної оптимізації на рівні окремих проєктів на шкоду глобальній ефективності портфеля.

Таблиця 1.3 – Виклики розподілу ресурсів у мультипроєктному середовищі та методи їх вирішення

Виклик	Прояв проблеми	Методи вирішення
Ресурсні конфлікти	Один ресурс потрібен для кількох проєктів одночасно	Пріоритизація проєктів, перепланування, залучення додаткових ресурсів
Перевантаження ресурсів	Призначення завдань понад доступну ємність ресурсу	Вирівнювання ресурсів, перерозподіл завдань, аутсорсинг
Невідповідність компетенцій	Призначення ресурсів на завдання, що не відповідають їх навичкам	Матриця компетенцій, навчання, ротація персоналу
Зміни обсягу проєктів	Неконтрольовані зміни впливають на потреби в ресурсах	Управління змінами, резервування ресурсів, буфери
Нерівномірне навантаження	Періоди високого та низького завантаження ресурсів	Згладжування ресурсів, планування потужностей
Недостатня видимість	Відсутність централізованої інформації про ресурси та проєкти	Централізовані системи управління, PMO, дашборди

Джерело: складено автором на основі [28, 29, 32]

Планування потужностей (capacity planning) є важливим елементом управління ресурсами в мультипроєктному середовищі. Воно передбачає визначення обсягу ресурсів, необхідних для виконання поточних та майбутніх проєктів, порівняння з наявними потужностями та розробку стратегій для подолання розривів [29]. Ефективне планування потужностей дозволяє завчасно виявляти потенційні дефіцити ресурсів та вживати превентивних заходів: найм нових спеціалістів, навчання існуючих, аутсорсинг або коригування портфеля проєктів.

Пріоритизація проєктів є критично важливим аспектом управління ресурсами в умовах їх обмеженості. Проєкти мають бути ранжовані за їх стратегічною цінністю, терміновістю, рівнем ризику та іншими критеріями, що відображають пріоритети організації [33]. При виникненні ресурсних конфліктів пріоритизація визначає, який проєкт отримає ресурси в першу чергу. Для формалізації процесу пріоритизації можуть використовуватися матриці рішень, скорингові моделі та інші інструменти.

Сучасні інформаційні системи управління проєктами надають широкі можливості для автоматизації процесів планування та розподілу ресурсів.

Вони дозволяють централізовано зберігати інформацію про ресурси та їх завантаження, автоматично виявляти конфлікти та перевантаження, застосовувати алгоритми вирівнювання та оптимізації, візуалізувати дані для прийняття управлінських рішень [28]. Використання спеціалізованого програмного забезпечення особливо важливе для організацій з великою кількістю одночасно виконуваних проєктів, де ручне управління ресурсами стає практично неможливим.

Особливості розподілу ресурсів в аутсорсингових IT-компаніях також пов'язані з використанням моделей виділених команд (dedicated teams) та загального пулу ресурсів (shared resource pool). У моделі виділеної команди група спеціалістів закріплюється за конкретним проєктом або замовником на тривалий період, що спрощує управління, але знижує гнучкість [34]. У моделі загального пулу ресурси динамічно розподіляються між проєктами залежно від поточних потреб, що підвищує ефективність використання, але ускладнює планування та створює ризики ресурсних конфліктів.

Аутсорсинг частини проєктних робіт зовнішнім підрядникам є ще одним важливим інструментом управління ресурсами в мультипроєктному середовищі [34]. Залучення зовнішніх ресурсів дозволяє компенсувати тимчасовий дефіцит власних спеціалістів, отримати доступ до компетенцій, що відсутні всередині організації, а також перерозподілити ризики, пов'язані з виконанням проєктів. Водночас, аутсорсинг потребує ретельного управління, оскільки додає додатковий рівень координації та може створювати ризики, пов'язані з якістю, комунікацією та захистом конфіденційної інформації. При виборі підрядників важливо враховувати їх досвід, репутацію, вартість послуг та здатність інтегруватися в робочі процеси компанії.

Особливу увагу в контексті управління ресурсами мультипроєктного середовища слід приділити феномену багатозадачності (multitasking). Коли спеціалісти одночасно працюють над завданнями з різних проєктів, виникають втрати продуктивності, пов'язані з перемиканням контексту, необхідністю згадувати деталі кожного проєкту та адаптацією до різних вимог і стилів

роботи [28]. Дослідження показують, що часте перемикання між завданнями може знижувати продуктивність на 20-40% порівняно з послідовним виконанням. Тому ефективне планування розподілу ресурсів має мінімізувати кількість одночасних призначень одного спеціаліста на різні проєкти. Рекомендується обмежувати участь спеціаліста максимум у двох-трьох проєктах одночасно, а для критичних завдань забезпечувати можливість повної концентрації на одному проєкті.

Управління компетенціями є невід'ємною складовою ефективного розподілу ресурсів. Матриця компетенцій (skills matrix) дозволяє систематизувати інформацію про навички та рівні кваліфікації кожного спеціаліста, що є основою для прийняття рішень про призначення на проєкти [27]. Для аутсорсингових IT-компаній типовими є технологічні компетенції (мови програмування, фреймворки, платформи), доменні знання (фінанси, охорона здоров'я, логістика), а також soft skills (комунікація, командна робота, управління часом). Ведення актуальної матриці компетенцій дозволяє швидко знаходити відповідних спеціалістів для нових проєктів та виявляти прогалини в компетенціях, що потребують заповнення через навчання або найм.

Розвиток компетенцій та крос-тренінг персоналу є стратегічним підходом до підвищення гнучкості ресурсної бази організації. Навчання спеціалістів додатковим навичкам розширює можливості їх призначення на різні проєкти та знижує ризики, пов'язані з залежністю від унікальних спеціалістів [34]. Водночас, інвестиції в розвиток персоналу потребують часу та фінансових ресурсів, а також створюють ризик відтоку навчених спеціалістів до конкурентів. Тому програми розвитку компетенцій мають бути збалансованими та відповідати стратегічним потребам організації. Важливо також враховувати бажання та мотивацію самих спеціалістів щодо розвитку в певних напрямках.

Метрики та показники ефективності використання ресурсів є важливим інструментом контролю та вдосконалення процесів управління. Серед ключових метрик виділяють: коефіцієнт використання (utilization rate) –

відношення білінгового часу до загального доступного часу; рівень завантаження (workload) – обсяг призначеної роботи відносно доступної ємності; показники перевантаження та недовантаження – відсоток часу, коли ресурси працюють вище або нижче нормального рівня [30]. Регулярний моніторинг цих метрик дозволяє виявляти проблеми в розподілі ресурсів та вживати коригувальних заходів до того, як вони призведуть до критичних наслідків для проєктів.

Прогнозування потреб у ресурсах є важливим елементом проактивного управління. На основі аналізу історичних даних, інформації про заплановані та потенційні проєкти, а також оцінки ринкових тенденцій організація може прогнозувати майбутній попит на ресурси та завчасно готуватися до його задоволення [29]. Методи прогнозування варіюються від простих експертних оцінок до складних статистичних моделей та алгоритмів машинного навчання. Точність прогнозування безпосередньо впливає на ефективність планування потужностей та мінімізацію витрат, пов'язаних з дефіцитом або надлишком ресурсів. Для підвищення точності прогнозів важливо накопичувати та аналізувати історичні дані про фактичне використання ресурсів на завершених проєктах.

Сценарне планування (scenario planning) дозволяє оцінити вплив різних варіантів розподілу ресурсів на результати проєктів та портфеля в цілому [28]. Моделювання альтернативних сценаріїв допомагає відповісти на питання типу «що буде, якщо»: що станеться, якщо ключовий спеціаліст звільниться; як вплине запуск нового великого проєкту на поточні зобов'язання; яким буде ефект від найму додаткових спеціалістів. Сучасні інформаційні системи управління проєктами надають інструменти для сценарного планування та візуалізації результатів, що суттєво полегшує прийняття обґрунтованих рішень.

Культурні та організаційні аспекти також впливають на ефективність розподілу ресурсів. У організаціях з сильною проєктною культурою, де проєктний менеджмент сприймається як цінність, а не як бюрократичний

тягар, процеси розподілу ресурсів зазвичай є більш ефективними [33]. Важливу роль відіграє прозорість інформації про завантаження ресурсів та статус проєктів, а також готовність керівництва приймати складні рішення щодо пріоритизації при виникненні конфліктів. Побудова культури співпраці та взаємодопомоги між командами різних проєктів сприяє більш гнучкому та ефективному використанню ресурсів. Залучення команд до процесу планування та врахування їх зворотного зв'язку підвищує реалістичність планів та мотивацію до їх виконання.

Управління ризиками є невід'ємною частиною розподілу ресурсів у мультипроєктному середовищі. Ризики, пов'язані з ресурсами, включають: недоступність ключових спеціалістів через хворобу, звільнення або інші причини; затримки з наймом нових спеціалістів; невідповідність компетенцій вимогам проєкту; перевантаження, що призводить до вигорання та зниження якості [32]. Для кожного ідентифікованого ризику мають бути розроблені стратегії реагування: уникнення, передача, пом'якшення або прийняття. Формування резервного пулу ресурсів та підготовка планів заміщення критичних спеціалістів є типовими заходами пом'якшення ресурсних ризиків.

Інтеграція управління ресурсами з іншими процесами проєктного менеджменту є необхідною умовою ефективності. Управління ресурсами тісно пов'язане з управлінням обсягом (визначення потреб у ресурсах), календарним плануванням (розподіл ресурсів у часі), управлінням вартістю (бюджетування витрат на ресурси), управлінням ризиками (ідентифікація та пом'якшення ресурсних ризиків) та управлінням комунікаціями (координація між учасниками) [25]. Забезпечення узгодженості цих процесів та своєчасного обміну інформацією між ними є ключовим фактором успіху управління ресурсами в мультипроєктному середовищі.

Таким чином, характеристика ресурсів проєкту охоплює різноманітні типи активів, серед яких людські ресурси є найбільш критичними для аутсорсингових компаній. Специфіка розподілу ресурсів у мультипроєктному середовищі визначається необхідністю балансування між множиною проєктів,

що конкурують за обмежені ресурси. Для оптимізації розподілу застосовуються спеціалізовані техніки вирівнювання та згладжування ресурсів, методи вирішення ресурсних конфліктів, планування потужностей та управління компетенціями. Ефективне управління ресурсами потребує централізованого підходу, підтриманого сучасними інформаційними системами, прогнозування та сценарного планування, а також формування відповідної організаційної культури та системи управління ризиками.

1.3 Проблеми та виклики управління ресурсами при виконанні множини проєктів

Управління ресурсами в мультипроєктному середовищі супроводжується комплексом проблем та викликів, що суттєво ускладнюють процес прийняття рішень та знижують ефективність реалізації проєктів. Згідно з дослідженнями Project Management Institute, понад 60% респондентів вказують на неефективне управління ресурсами як найбільший виклик у своїй діяльності [35]. Ця проблема набуває особливої гостроти в аутсорсингових компаніях, де одночасно виконується значна кількість проєктів для різних замовників з різними пріоритетами, термінами та технологічними вимогами. Розуміння природи цих проблем та викликів є необхідною передумовою для розробки ефективних моделей та методів їх подолання.

Статистика реалізації проєктів свідчить про системні проблеми в галузі проєктного менеджменту. За даними The Standish Group, лише 31% проєктів у світі завершуються вчасно, тоді як понад 50% проєктів зазнають затримок, переважно через необхідність переробки та неефективний розподіл ресурсів [36]. Дослідження McKinsey показує, що 17% великих IT-проєктів стають настільки проблемними, що загрожують самому існуванню компанії [37]. Ці невтішні показники підкреслюють критичну важливість вдосконалення підходів до управління ресурсами, особливо в умовах паралельного виконання множини проєктів.

Однією з фундаментальних проблем мультипроєктного середовища є так званий «синдром розподілу ресурсів» (resource allocation syndrome), описаний у науковій літературі як комплекс системних дисфункцій, що виникають при спробі оптимізувати використання обмежених ресурсів між конкуруючими проєктами [38]. Цей синдром проявляється у постійному перерозподілі ресурсів між проєктами, що призводить до фрагментації уваги спеціалістів, зниження їх продуктивності та збільшення часу на завершення кожного окремого проєкту. Парадоксально, але спроби максимізувати завантаження ресурсів часто призводять до протилежного результату – зниження загальної ефективності портфеля проєктів.

Систематичний огляд літератури з проблеми планування ресурсів у мультипроєктному середовищі (Resource-Constrained Multi-Project Scheduling Problem, RCMPSP) виявив, що більшість існуючих досліджень не враховують одночасно невизначеність та гнучкість ресурсів [39]. Аналіз 107 наукових публікацій за період 2013-2023 років показав, що 72% робіт зосереджуються виключно на детерміністичних моделях, ігноруючи стохастичну природу реальних проєктних середовищ. Ця прогалина вказує на необхідність розробки комплексних підходів, що враховують невизначеність тривалості робіт, доступності ресурсів та інших факторів.

Класифікація основних проблем управління ресурсами в мультипроєктному середовищі наведена в таблиці 1.4.

Проблема невизначеності є однією з найбільш складних у контексті планування ресурсів мультипроєктного середовища. Невизначеність може стосуватися тривалості виконання робіт, доступності ресурсів, обсягів проєктів, вимог замовників та інших факторів [39]. Традиційні методи планування, такі як метод критичного шляху (CPM) та PERT, не здатні адекватно враховувати невизначеність, що призводить до розриву між планами та їх практичною реалізацією. Дослідження показують, що фактична тривалість робіт може відхилятися від планової на 50-100%, що робить детерміністичні плани малопридатними для практичного застосування.

Таблиця 1.4 – Класифікація проблем управління ресурсами в мультипроєктному середовищі

Категорія проблем	Опис проблеми	Наслідки для проєктів
Ресурсний дефіцит	Недостатня кількість спеціалістів з необхідними компетенціями для виконання всіх запланованих робіт	Затримки проєктів, перевантаження персоналу, зниження якості
Ресурсні конфлікти	Одночасна потреба в одних і тих самих ресурсах для різних проєктів	Конкуренція між проєктами, субоптимальний розподіл
Невизначеність	Непередбачуваність тривалості робіт, доступності ресурсів, обсягів проєктів	Складність планування, часті корекції планів
Багатозадачність	Залучення спеціалістів до кількох проєктів одночасно з частим перемиканням	Зниження продуктивності на 20-40%, вигорання
Залежності	Міжпроєктні залежності через спільні ресурси, технології, результати	Каскадні затримки, складність координації
Недостатня видимість	Відсутність централізованої інформації про стан ресурсів та проєктів	Прийняття рішень на основі неповної інформації

Джерело: складено автором на основі [35, 38, 39]

Проблема якості та переробки є ще одним критичним викликом управління ресурсами. Дефіцити якості є основною причиною переробки в проєктах, при цьому рівень кваліфікації персоналу є критичним детермінантом якості виконання робіт [36]. Згідно з дослідженнями, понад 50% затримок проєктів пов'язані саме з необхідністю переробки через виявлені дефекти якості. Це створює порочне коло: намагаючись завершити проєкти швидше, менеджери призначають менш кваліфікованих спеціалістів або перевантажують досвідчених, що призводить до зниження якості, виявлення дефектів та необхідності додаткових витрат часу на переробку.

Багатозадачність та часте перемикання між проєктами є типовою практикою в мультипроєктних середовищах, яка, однак, має суттєві негативні наслідки. Дослідження показують, що лише 15% менеджерів проєктів зосереджуються на одному проєкті, тоді як більшість одночасно веде від 5 до 10 проєктів [37]. Когнітивні витрати на перемикання контексту між різними проєктами можуть знижувати продуктивність на 20-40%. Крім того, багатозадачність створює ризики вигорання персоналу, зниження мотивації та

збільшення плинності кадрів, що є особливо критичним для аутсорсингових компаній, де людський капітал є основним активом.

Проблема часу переміщення ресурсів між проєктами (transfer times) отримує все більше уваги в науковій літературі. На відміну від традиційних моделей, що припускають миттєве переміщення ресурсів, реальні умови передбачають необхідність часу на передачу знань, адаптацію до нового проєктного контексту та встановлення комунікацій з новою командою [40]. Ігнорування цього фактора призводить до систематичної недооцінки тривалості проєктів та перевантаження ключових спеціалістів у періоди переходу між проєктами.

Порівняння основних викликів управління ресурсами та їх вплив на показники ефективності проєктів наведено в таблиці 1.5.

Таблиця 1.5 – Вплив основних викликів на показники ефективності проєктів

Виклик	Вплив на терміни	Вплив на бюджет	Вплив на якість
Ресурсний дефіцит	Високий	Високий	Середній
Ресурсні конфлікти	Високий	Середній	Середній
Невизначеність	Високий	Високий	Низький
Багатозадачність	Середній	Низький	Високий
Переробка через дефекти якості	Високий	Високий	Високий
Час переміщення ресурсів	Середній	Середній	Низький

Джерело: складено автором на основі [36, 37, 40]

Організаційні та комунікаційні проблеми також відіграють значну роль у неефективності управління ресурсами. Дослідження показують, що 29% проєктів зазнають невдачі через погану комунікацію [37]. У мультипроєктному середовищі комунікаційна складність зростає експоненційно з кількістю проєктів та залучених учасників. Традиційні методи запиту ресурсів через електронну пошту та телефонні дзвінки є неефективними та створюють ризики подвійного бронювання, внутрішніх конфліктів та невідповідностей [41]. Відсутність централізованої системи управління ресурсами призводить до того, що рішення приймаються на основі

неповної або застарілої інформації. Крім того, різні зацікавлені сторони можуть мати різне бачення пріоритетів та критеріїв розподілу ресурсів, що створює додаткові джерела конфліктів.

Проблема пріоритизації проєктів є постійним джерелом конфліктів у мультипроєктному середовищі. Різні проєкти можуть мати різну стратегічну цінність, терміновість, рівень ризику та вимоги замовників, що ускладнює встановлення чітких пріоритетів [42]. За відсутності формалізованої системи пріоритизації рішення щодо розподілу ресурсів часто приймаються на основі «голосу найгучнішого» або політичних міркувань, а не об'єктивних критеріїв. Це призводить до ситуацій, коли стратегічно важливі проєкти можуть бути недофінансовані ресурсами на користь менш значущих, але більш «видимих» ініціатив. Впровадження формалізованих методів пріоритизації, таких як скорингові моделі або матриці пріоритетів, допомагає об'єктивізувати процес прийняття рішень та зменшити конфлікти.

Фінансові наслідки неефективного управління ресурсами є значними та далекосяжними. За оцінками, збитки від невдалих IT-проєктів у США становлять від 50 до 150 мільярдів доларів щорічно, а глобально – близько 6,2 трильйона доларів на рік [37]. Кожні 10 секунд через неефективну реалізацію бізнес-стратегій компанії у всьому світі втрачають 1 мільйон доларів. Ці вражаючі цифри підкреслюють економічну важливість вдосконалення практик управління ресурсами та впровадження ефективних методів та інструментів планування. Інвестиції у покращення процесів управління ресурсами можуть мати значний позитивний вплив на фінансові результати організації через зменшення втрат від невдалих проєктів та підвищення ефективності використання наявних ресурсів.

Специфічні виклики управління ресурсами в аутсорсингових компаніях пов'язані з особливостями їх бізнес-моделі. По-перше, аутсорсингові компанії працюють одночасно з множиною замовників, кожен з яких має власні пріоритети та очікування. По-друге, контрактні зобов'язання щодо термінів та обсягів робіт створюють жорсткі обмеження, що ускладнюють гнучкий

перерозподіл ресурсів. По-третє, необхідність підтримки компетенцій у різних технологічних стеках для різних замовників вимагає постійних інвестицій у навчання персоналу. По-четверте, конкурентний ринок праці в IT-сфері створює ризики відтоку ключових спеціалістів, що може критично вплинути на здатність виконувати проєктні зобов'язання.

Дослідження в галузі будівництва виявило п'ять критичних викликів управління ресурсами в мультипроєктному середовищі [43]: прийняття рішень в умовах неповної інформації, врахування контексту проєкту, управління паралельними активностями, планування ресурсів в умовах обмежень, логістика переміщення ресурсів. Хоча ці виклики були ідентифіковані в контексті будівельної галузі, вони є універсальними та повною мірою застосовні до аутсорсингових IT-компаній з відповідною адаптацією до специфіки галузі.

Систематизація викликів управління ресурсами в контексті аутсорсингових IT-компаній наведена в таблиці 1.6.

Таблиця 1.6 – Виклики управління ресурсами в аутсорсингових IT-компаніях

Виклик	Специфіка для аутсорсингу	Можливі підходи до вирішення
Множина замовників	Різні пріоритети, вимоги, стилі комунікації замовників	Формалізація SLA, єдина система пріоритизації, виділені account managers
Контрактні обмеження	Фіксовані терміни та обсяги за контрактами	Гнучкі контракти, резервування потужностей, буфери
Технологічна різноманітність	Різні технологічні стеки для різних замовників	Спеціалізація команд, крос-тренінг, матриця компетенцій
Плинність кадрів	Конкурентний ринок праці, ризик втрати ключових спеціалістів	Програми утримання, кар'єрний розвиток, резервування
Часові пояси	Замовники з різних часових поясів, обмежений overlap	Гнучкий графік, розподілені команди, асинхронна комунікація
Конфіденційність	Обмеження на обмін інформацією між проєктами різних замовників	Політики безпеки, фізична та логічна ізоляція

Джерело: складено автором

Проблема багатокомпетентності ресурсів (multi-skill resources) додає ще один рівень складності до задачі планування. У сучасних IT-проєктах спеціалісти часто володіють множиною навичок різного рівня, що дозволяє їм виконувати різні типи завдань [40]. З одного боку, це підвищує гнучкість при призначенні ресурсів, з іншого – суттєво ускладнює задачу оптимального планування, оскільки необхідно враховувати не лише доступність ресурсу, але й відповідність його компетенцій вимогам конкретного завдання, а також рівень володіння кожною з навичок. Крім того, призначення спеціаліста на завдання, що не відповідає його основній компетенції, може призводити до зниження продуктивності та якості роботи.

Складність задачі планування ресурсів у мультипроєктному середовищі з обмеженнями (RCMPSP) є NP-складною, що означає відсутність ефективних точних алгоритмів для її вирішення у загальному випадку [39]. Це змушує дослідників та практиків звертатися до наближених методів: евристичних правил пріоритету, метаевристичних алгоритмів (генетичні алгоритми, імітація відпалу, мурашині алгоритми, рій часток), гібридних підходів, що поєднують різні методи. Вибір конкретного методу залежить від специфіки задачі, доступних обчислювальних ресурсів та вимог до якості рішення. Для великих задач з тисячами активностей точні методи стають непрактичними, що робить метаевристичні підходи основним інструментом оптимізації.

Застосування сучасних інформаційних технологій для вирішення проблем управління ресурсами набуває все більшого поширення. Використання програмного забезпечення для управління проєктами може суттєво підвищити ефективність планування та розподілу ресурсів [44]. Однак дослідження показують, що понад 75% організацій не здатні відстежувати свої ключові показники ефективності в режимі реального часу, а 44% менеджерів проєктів не використовують жодного спеціалізованого програмного забезпечення. Це свідчить про значний потенціал вдосконалення практик управління ресурсами через впровадження сучасних інструментів автоматизації та аналітики. Ринок програмного забезпечення для управління

проєктами стрімко зростає та, за прогнозами, досягне 15 мільярдів доларів до 2030 року.

Штучний інтелект та машинне навчання відкривають нові можливості для вирішення проблем управління ресурсами. Дослідження показують, що 85% менеджерів проєктів вважають, що ШІ має потенціал суттєво покращити реалізацію проєктів, а впровадження рішень на основі ШІ в управлінні проєктами зростає на 45% щорічно [37]. ШІ може використовуватися для прогнозування тривалості робіт, виявлення ризиків, оптимізації розподілу ресурсів, автоматизації рутинних завдань планування та моніторингу. Однак успішне впровадження потребує якісних даних, відповідної інфраструктури та компетенцій персоналу. Важливо також забезпечити прозорість алгоритмів прийняття рішень та можливість людського контролю.

Особливої уваги заслуговує проблема управління ресурсами в умовах кризових ситуацій та зовнішніх шоків. Пандемія COVID-19 продемонструвала вразливість традиційних моделей управління проєктами та необхідність розробки більш гнучких та адаптивних підходів. Перехід до віддаленої роботи створив нові виклики для координації команд та моніторингу продуктивності. Для українських аутсорсингових компаній додаткові виклики створює воєнний стан: необхідність забезпечення безпеки персоналу, релокація співробітників, нестабільність інфраструктури, мобілізаційні ризики. Ці фактори суттєво ускладнюють планування та потребують розробки специфічних механізмів забезпечення безперервності бізнесу.

Проблема інтеграції різних підсистем управління також заслуговує на увагу. В багатьох організаціях управління ресурсами, фінансами, ризиками та якістю здійснюється в окремих, часто несумісних системах. Це призводить до фрагментації даних, дублювання інформації та неможливості отримати цілісну картину стану портфеля проєктів. Інтеграція різних підсистем в єдину платформу управління є важливим, але складним завданням, що потребує значних інвестицій та організаційних змін. Відсутність інтеграції призводить

до того, що менеджери змушені витрачати значний час на збір та консолідацію даних з різних джерел, що знижує оперативність прийняття рішень.

Психологічні та соціальні аспекти управління ресурсами часто недооцінюються в технічних підходах до оптимізації. Мотивація, задоволеність роботою, командна динаміка, лідерство – всі ці фактори суттєво впливають на продуктивність та якість роботи, але складно піддаються формалізації та включенню в математичні моделі. Перевантаження ресурсів призводить не лише до затримок проєктів, але й до вигорання персоналу, що має довгострокові негативні наслідки для організації. Тому ефективне управління ресурсами має враховувати не лише формальні обмеження, але й людські фактори. Важливо забезпечити баланс між максимізацією використання ресурсів та підтриманням здорового робочого середовища.

Проблема масштабування підходів до управління ресурсами є актуальною для організацій, що зростають. Методи та інструменти, які ефективно працюють для невеликих команд та кількох проєктів, можуть виявитися непридатними при збільшенні масштабу. Перехід від ручного управління до автоматизованих систем, від локальних рішень до корпоративних платформ потребує не лише технічних інвестицій, але й зміни організаційної культури та процесів. Опір змінам є однією з найбільших перешкод на шляху впровадження нових підходів до управління ресурсами. Дослідження показують, що 48% організацій стикаються з опором змінам при впровадженні нових методологій управління проєктами.

Варто також зазначити проблему довгострокового планування ресурсів. В умовах високої невизначеності та динамічності середовища точне прогнозування потреб у ресурсах на горизонті понад кілька місяців є вкрай складним завданням. Водночас стратегічні рішення щодо найму персоналу, інвестицій у навчання, розвитку нових компетенцій потребують довгострокової перспективи. Цей конфлікт між короткостроковою невизначеністю та потребою в довгострокових рішеннях є фундаментальним викликом управління ресурсами. Для його подолання організації застосовують

сценарне планування, резервування потужностей та гнучкі моделі залучення ресурсів.

Окремої уваги заслуговує проблема відповідності нормативним вимогам та стандартам. Аутсорсингові компанії, що працюють з міжнародними замовниками, часто змушені дотримуватися різних регуляторних вимог: захист персональних даних (GDPR), безпека інформаційних систем (SOC 2, ISO 27001), галузеві стандарти (HIPAA для охорони здоров'я, PCI DSS для фінансового сектору). Ці вимоги можуть впливати на можливості розподілу ресурсів: обмежувати доступ певних спеціалістів до певних проєктів, вимагати додаткової сертифікації персоналу, накладати обмеження на місце розташування ресурсів. Недотримання регуляторних вимог може призвести до значних штрафів та репутаційних втрат.

Проблема управління знаннями тісно пов'язана з управлінням ресурсами. При частому перерозподілі спеціалістів між проєктами виникає ризик втрати накопичених знань та досвіду. Формалізація та передача знань потребує часу та зусиль, які часто не враховуються при плануванні. Водночас, концентрація критичних знань у обмеженій кількості спеціалістів створює ризики для організації в разі їх недоступності. Ефективна система управління знаннями є важливою передумовою гнучкого розподілу ресурсів без втрати якості та продуктивності. Документування проєктного досвіду, створення баз знань та забезпечення їх актуальності є важливими, але часто недооціненими аспектами управління ресурсами.

Ще одним важливим аспектом є проблема взаємодії з зовнішніми постачальниками ресурсів. Аутсорсингові компанії часто залучають субпідрядників, фрілансерів та інших зовнішніх виконавців для покриття пікових навантажень або отримання спеціалізованих компетенцій. Управління такими гібридними командами створює додаткові виклики: координація різних виконавців, забезпечення якості роботи, захист інтелектуальної власності, узгодження процесів та інструментів. Ефективна інтеграція зовнішніх ресурсів потребує чітких процедур, систем контролю та механізмів

оцінки якості. Важливо також забезпечити передачу знань та інтеграцію зовнішніх виконавців у проєктні команди без втрати ефективності комунікації.

Важливим викликом є також забезпечення безперервного вдосконалення процесів управління ресурсами. Організації повинні постійно аналізувати результати своєї діяльності, виявляти слабкі місця та впроваджувати покращення. Це потребує систематичного збору та аналізу даних про використання ресурсів, виконання проєктів, задоволеність замовників та персоналу. Однак в умовах операційного навантаження часто не залишається часу та ресурсів на рефлексію та вдосконалення, що призводить до закріплення неефективних практик. Побудова культури безперервного вдосконалення є довгостроковим завданням, що потребує підтримки на всіх рівнях організації.

Таким чином, управління ресурсами в мультипроєктному середовищі супроводжується комплексом взаємопов'язаних проблем та викликів. Основними серед них є: ресурсний дефіцит та конфлікти, невизначеність та динамічність середовища, негативні наслідки багатозадачності, складність координації та комунікації, проблеми пріоритизації та прийняття рішень, необхідність відповідності регуляторним вимогам та управління знаннями. Для аутсорсингових IT-компаній ці виклики доповнюються специфічними факторами: множина замовників, контрактні обмеження, технологічна різноманітність, конкуренція за таланти, взаємодія із зовнішніми постачальниками ресурсів. Ефективне подолання цих викликів потребує комплексного підходу, що поєднує формалізовані методи оптимізації, сучасні інформаційні системи, організаційні механізми координації та увагу до людських факторів. Розробка та впровадження математичних моделей розподілу ресурсів є важливим кроком на шляху до підвищення ефективності управління портфелем проєктів в умовах мультипроєктного середовища аутсорсингової компанії.

РОЗДІЛ 2
ІНСТРУМЕНТАРІЙ МОДЕЛЮВАННЯ РОЗПОДІЛУ РЕСУРСІВ У МУЛЬТИПРОЄКТНОМУ СЕРЕДОВИЩІ

2.1 Аналіз існуючих методів та моделей оптимізації розподілу ресурсів між проєктами

Оптимізація розподілу ресурсів між проєктами є однією з найважливіших задач управління мультипроєктним середовищем. Ця задача формалізується у вигляді різноманітних математичних моделей, для вирішення яких розроблено широкий спектр методів – від класичних аналітичних підходів до сучасних метаевристичних алгоритмів та методів машинного навчання. У науковій літературі ця проблема відома як задача планування проєктів з обмеженнями на ресурси (Resource-Constrained Project Scheduling Problem, RCPSP), яка є NP-складною та привертає увагу дослідників протягом останніх десятиліть [45]. Розуміння існуючих методів та моделей є необхідною передумовою для розробки ефективних рішень у контексті мультипроєктного середовища аутсорсингової компанії.

Базова математична модель задачі RCPSP може бути представлена наступним чином. Нехай є проєкт, що складається з n робіт (операцій), які необхідно виконати з урахуванням відношень передування та обмежень на ресурси. Метою є мінімізація загальної тривалості проєкту (makespan) або іншого цільового показника. Класична формалізація задачі RCPSP включає цільову функцію та систему обмежень [46]:

$$Cmax \rightarrow min, \tag{2.1}$$

де $Cmax$ – максимальний час завершення всіх робіт проєкту (makespan).

Обмеження передування визначають, що робота j може розпочатися лише після завершення всіх її попередників:

$$S_j \geq S_i + d_i, \forall (i,j) \in E, \quad (2.2)$$

де S_j – час початку роботи j,

d_i – тривалість роботи i,

E – множина дуг графа передування.

Ресурсні обмеження визначають, що сумарне споживання кожного ресурсу в будь-який момент часу не перевищує його доступної кількості:

$$\sum_j \in A(t)\, r_{jk} \leq R_k, \forall k \in K, \forall t, \quad (2.3)$$

де $A(t)$ – множина робіт, що виконуються в момент часу t,

r_{jk} – потреба роботи j в ресурсі k,

R_k – доступна кількість ресурсу k,

K – множина типів ресурсів.

Розширенням базової моделі RCPSP є багатомодова задача (Multi-Mode RCPSP, MRCPSP), в якій кожна робота може виконуватися в одному з кількох режимів з різними тривалостями та потребами в ресурсах [47]. Ця модель більш точно відображає реальні умови, коли одну й ту саму роботу можуть виконувати різні спеціалісти з різною продуктивністю або із залученням різних технологій.

Для мультипроєктного середовища формулюється задача RCMPSP (Resource-Constrained Multi-Project Scheduling Problem), яка враховує одночасне планування кількох проєктів зі спільним пулом ресурсів. Цільова функція може включати мінімізацію загального часу завершення всіх проєктів, мінімізацію сумарних затримок, максимізацію чистої приведеної вартості або комбінацію цих критеріїв [45]:

$$F = \sum_{p=1} P\, w_p \cdot max(0, C_p - D_p) \rightarrow min, \quad (2.4)$$

де P – кількість проєктів,

w_p – вага (пріоритет) проєкту p,

C_p – час завершення проєкту p,

D_p – директивний термін проєкту p.

Класифікація існуючих методів вирішення задач оптимізації розподілу ресурсів наведена в таблиці 2.1.

Таблиця 2.1 – Класифікація методів оптимізації розподілу ресурсів

Клас методів	Основні підходи	Характеристика
Точні методи	Метод гілок та меж, цілочисельне лінійне програмування, динамічне програмування	Гарантують оптимальність, але мають експоненційну складність
Евристичні методи	Правила пріоритету, конструктивні алгоритми, схеми генерації розкладів	Швидкі, але не гарантують оптимальність
Метаевристичні методи	Генетичні алгоритми, PSO, імітація відпалу, мурашині алгоритми	Баланс між якістю рішення та обчислювальними витратами
Гібридні методи	Матевристики, комбінації метаевристик, інтеграція з ML	Поєднують переваги різних підходів
Методи на основі ML	Глибоке навчання з підкріпленням, нейронні мережі, класифікатори	Здатні до адаптації та навчання на даних

Джерело: складено автором на основі [45, 46, 47]

Точні методи гарантують знаходження оптимального рішення, однак їх застосування обмежене задачами невеликого розміру через експоненційну обчислювальну складність. Метод гілок та меж (Branch-and-Bound) залишається основним точним методом для RCPSP. Сучасні дослідження спрямовані на розробку ефективних стратегій відсікання та верхніх оцінок для прискорення пошуку [46]. Змішане цілочисельне лінійне програмування (MILP) також активно використовується, особливо для формулювання багатокритеріальних моделей та моделей з невизначеністю.

Евристичні методи, зокрема правила пріоритету, є найбільш поширеними на практиці завдяки своїй простоті та швидкості. Схема генерації розкладів (Schedule Generation Scheme, SGS) є базовим механізмом побудови допустимих розкладів. Існують дві основні схеми: послідовна (Serial SGS) та паралельна (Parallel SGS). Послідовна схема на кожному кроці вибирає одну роботу для планування, тоді як паралельна схема на кожному часовому

інтервалі планує всі доступні роботи. Правила пріоритету визначають, яка робота буде вибрана першою. Найбільш ефективними за результатами досліджень є правила LFT (Latest Finish Time), MSLK (Minimum Slack), GRPW (Greatest Rank Positional Weight) [48].

Метаевристичні алгоритми становлять основний інструментарій для вирішення задач оптимізації розподілу ресурсів великої розмірності. Генетичні алгоритми (ГА) є найбільш дослідженим класом метаевристик для RCPSP. Типова структура ГА для задачі планування проєктів включає: кодування рішення (зазвичай як список активностей або список пріоритетів), оператори кросовера та мутації, адаптовані до специфіки задачі, та механізм селекції [49]. Формула оновлення популяції в генетичному алгоритмі може бути представлена:

$$P_{t+1} = Selection\big(P_t \cup Crossover(P_t) \cup Mutation(P_t)\big), \quad (2.5)$$

де P_t – популяція на ітерації t.

Алгоритм оптимізації роєм часток (Particle Swarm Optimization, PSO) є ще одним популярним метаевристичним підходом. PSO моделює соціальну поведінку зграї птахів або косяка риб. Кожна частка в роїні представляє потенційне рішення та рухається в просторі пошуку, орієнтуючись на власний найкращий результат та найкращий результат всього рою [50]. Оновлення швидкості та позиції частки описується формулами:

$$v_i(t+1) = w \cdot v_i(t) + c_1 r_1(pbest_i - x_i) + c_2 r_2(gbest - x_i), \quad (2.6)$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (2.7)$$

де v_i– швидкість частки I,

x_i – позиція частки,

w – інерційна вага,

c_1, c_2 – когнітивний та соціальний коефіцієнти;

r_1, r_2 – випадкові числа;

$pbest_i$ – найкраща позиція частки;

$gbest$ – найкраща глобальна позиція.

Порівняльний аналіз метаевристичних алгоритмів для задач планування ресурсів наведено в таблиці 2.2.

Таблиця 2.2 – Порівняння метаевристичних алгоритмів для RCPSP

Алгоритм	Глобальний пошук	Локальний пошук	Особливості
Генетичний алгоритм (GA)	Високий	Середній	Найбільш досліджений, універсальний
Рій часток (PSO)	Середній	Високий	Швидка збіжність, простота реалізації
Імітація відпалу (SA)	Середній	Високий	Уникає локальних мінімумів
Мурашиний алгоритм (ACO)	Високий	Середній	Ефективний для комбінаторних задач
Гібрид GA-PSO	Високий	Високий	Поєднує переваги обох підходів

Джерело: складено автором на основі [49, 50, 51]

Гібридні алгоритми, що поєднують різні метаевристики або інтегрують метаевристики з точними методами (матевристики), демонструють найкращі результати на бенчмарках. Дослідження показують, що комбінація генетичного алгоритму з алгоритмом рою часток (GA-PSO) дозволяє подолати недоліки кожного з методів окремо: GA забезпечує глобальну диверсифікацію пошуку, тоді як PSO підвищує ефективність локального пошуку [51]. Матевристики, що комбінують метаевристичний пошук з методом гілок та меж, дозволили знайти 24 нові найкращі відомі рішення для стандартних бенчмарків PSPLIB [46].

Задача планування проєктів з багатокомпетентними ресурсами (Multi-Skill RCPSP, MSRCPSP) є важливим розширенням базової моделі для сфери IT-аутсорсингу. У цій моделі кожен ресурс (спеціаліст) володіє множиною навичок з різним рівнем компетентності, а кожна робота вимагає певних

навичок [52]. Математична модель MSRCPSP включає додаткові обмеження на відповідність компетенцій:

$$skill_{r,s} \geq req_{j,s}, \forall r \in Assigned(j), \forall s \in Skills(j), \quad (2.8)$$

де $skill_{r,s}$ – рівень володіння навичкою s ресурсом r,

$req_{j,s}$– вимога роботи j до навички s.

Сучасні дослідження активно впроваджують методи машинного навчання для вирішення задач планування ресурсів. Глибоке навчання з підкріпленням (Deep Reinforcement Learning, DRL) дозволяє навчити агента приймати рішення щодо розподілу ресурсів на основі взаємодії з середовищем [53]. Перевагою DRL є здатність адаптуватися до змін та навчатися на історичних даних. Дослідження показують, що 85% менеджерів проєктів вважають, що штучний інтелект має потенціал суттєво покращити реалізацію проєктів, а впровадження рішень на основі ШІ в управлінні проєктами зростає на 45% щорічно.

Робастна оптимізація є важливим напрямком для врахування невизначеності в задачах планування ресурсів. Підхід Берцімаса-Сіма дозволяє побудувати рішення, стійкі до варіацій параметрів у межах заданих інтервалів невизначеності [54]. Формулювання робастної моделі включає параметр невизначеності Г, що контролює рівень консервативності рішення:

$$max_{\{S \subseteq J : |S| \leq \Gamma\}} \sum_{j \in S} \hat{d}_j \leq Slack, \quad (2.9)$$

де $\hat{d}_j$ – максимальне відхилення тривалості роботи j;

Γ – параметр бюджету невизначеності;

$Slack$ – допустимий резерв часу.

Порівняльний аналіз підходів до врахування невизначеності в задачах планування ресурсів наведено в таблиці 2.3.

Таблиця 2.3 – Підходи до врахування невизначеності в плануванні ресурсів

Підхід	Спосіб моделювання невизначеності	Переваги та недоліки
Стохастичне програмування	Ймовірнісні розподіли параметрів	+: точне врахування розподілів; –: потребує знання розподілів
Робастна оптимізація	Інтервали невизначеності параметрів	+: не потребує розподілів; –: може бути надмірно консервативною
Нечітке програмування	Нечіткі множини та функції належності	+: враховує експертні оцінки; –: складність інтерпретації
Симуляція Монте-Карло	Багаторазове моделювання сценаріїв	+: гнучкість; –: обчислювальна складність
Реактивне планування	Динамічна адаптація до змін	+: адаптивність; –: може призводити до нестабільності

Джерело: складено автором на основі [45, 54]

Для тестування та порівняння алгоритмів планування проєктів з обмеженнями на ресурси широко використовуються стандартні набори даних. Найбільш відомим є PSPLIB (Project Scheduling Problem Library), що містить набори задач J30, J60 та J120 з 30, 60 та 120 роботами відповідно [46]. Для багатомодових задач використовується Multi-Mode Library. Для задач з багатокомпетентними ресурсами розроблено бібліотеку iMOPSE. Наявність стандартизованих бенчмарків дозволяє об'єктивно порівнювати ефективність різних алгоритмів та відстежувати прогрес у галузі.

Квантові алгоритми представляють новий перспективний напрямок у вирішенні задач оптимізації розподілу ресурсів. Гібридний квантовий алгоритм (HQPSO), що базується на квантовому алгоритмі рою часток, демонструє покращену ефективність порівняно з класичними метаевристиками на задачах великої розмірності [47]. Квантові обчислення дозволяють паралельно досліджувати великі простори рішень, що є особливо актуальним для NP-складних задач планування.

Аналіз сучасної літератури показує, що більшість досліджень зосереджуються на детерміністичних моделях та не враховують одночасно невизначеність та гнучкість ресурсів [45]. Систематичний огляд 107 публікацій за період 2013-2023 років виявив, що лише 28% робіт розглядають стохастичні моделі, а комплексні підходи, що враховують невизначеність,

гнучкість ресурсів та багатокомпетентність, залишаються недостатньо дослідженими. Це визначає напрямки подальших досліджень та обґрунтовує актуальність розробки нових моделей для умов реальних мультипроєктних середовищ.

Важливим аспектом практичного застосування методів оптимізації є вибір адекватної цільової функції. У реальних умовах аутсорсингової компанії мінімізація загальної тривалості портфеля проєктів може бути не єдиною метою. Часто необхідно враховувати додаткові критерії: рівномірність завантаження ресурсів, мінімізацію витрат на понаднормову роботу, максимізацію задоволеності замовників, мінімізацію ризиків затримок критичних проєктів. Це обумовлює необхідність застосування багатокритеріальних моделей оптимізації.

Багатокритеріальна оптимізація передбачає одночасну оптимізацію кількох конфліктуючих цільових функцій. Для таких задач не існує єдиного оптимального рішення, а формується множина Парето-оптимальних рішень. Алгоритм NSGA-II (Non-dominated Sorting Genetic Algorithm II) є одним з найбільш ефективних підходів для багатокритеріальної оптимізації [49]. Він використовує механізм недомінованого сортування та метрику crowding distance для підтримки різноманітності рішень на фронті Парето.

Проблема переробки (rework) є критичною для IT-проєктів та безпосередньо пов'язана з якістю виконання робіт та компетенціями персоналу. Сучасні дослідження інтегрують механізми передачі якості та динамічної реконструкції мережі переробки в моделі MSRCPSP [52]. Ідея полягає в тому, що дефекти якості на ранніх етапах проєкту можуть каскадно поширюватися на наступні роботи, збільшуючи обсяг переробки. Врахування цього ефекту в моделях планування дозволяє приймати більш обґрунтовані рішення щодо призначення ресурсів.

Алгоритм оптимізації газелі (Gazelle Optimization Algorithm) є одним з нових біоінспірованих алгоритмів, що демонструє високу ефективність на задачах планування проєктів. Покращена версія алгоритму (GOAIP) з

динамічними операторами, shuffle crossover та Гаусівською мутацією перевершує традиційні метаевристики за точністю рішень та ефективністю збіжності [52]. Це свідчить про перспективність подальших досліджень у напрямку розробки нових біоінспірованих алгоритмів.

Час переміщення ресурсів між роботами (transfer times) є важливим фактором, що часто ігнорується в базових моделях RCPSP. У реальних умовах переключення спеціаліста між проєктами потребує часу на передачу знань, ознайомлення з контекстом нового проєкту та встановлення комунікацій з командою. Ігнорування цього фактора призводить до систематичної недооцінки тривалості проєктів. Сучасні моделі включають матриці часів переміщення між роботами та ресурсами [46].

Методи локального пошуку є важливим компонентом ефективних алгоритмів оптимізації. Варіаційні структури сусідства (Variable Neighborhood Search, VNS) та методи neighborhood descent дозволяють покращувати рішення, знайдені метаевристиками. Інтеграція локального пошуку з популяційними алгоритмами призводить до створення меметичних алгоритмів, що демонструють найкращі результати на стандартних бенчмарках [52]. Меметичні алгоритми поєднують глобальний пошук генетичного алгоритму з інтенсифікацією через локальний пошук.

Календарні обмеження на ресурси є ще одним розширенням базової моделі, актуальним для практичного застосування. У реальних умовах доступність ресурсів може змінюватися в часі через відпустки, лікарняні, навчання та інші фактори. Моделі з календарями ресурсів (resource calendars) дозволяють враховувати ці обмеження та будувати реалістичні розклади [46]. Особливо це актуально для аутсорсингових компаній, де необхідно враховувати різні часові пояси замовників та відповідну доступність персоналу.

Декомпозиційні підходи дозволяють розбивати великі задачі на менші підзадачі, що вирішуються окремо. Метод декомпозиції за часовими вікнами (time windows decomposition) розбиває горизонт планування на інтервали та

оптимізує розклад для кожного інтервалу послідовно [47]. Це значно зменшує обчислювальну складність для задач з великою кількістю робіт та ресурсів. Координація між підзадачами забезпечується через буфери та граничні умови.

Проактивне планування передбачає включення буферів та резервів у розклад для захисту від потенційних збоїв. На відміну від реактивного планування, що реагує на збої постфактум, проактивний підхід намагається передбачити можливі проблеми та завчасно підготуватися до них. Метод критичного ланцюга (Critical Chain Project Management) є прикладом проактивного підходу, що концентрує буфери в стратегічних точках проєкту замість рівномірного розподілу резервів по всіх роботах.

Інтеграція планування портфеля проєктів з операційним плануванням ресурсів є важливим напрямком досліджень. Традиційно ці задачі вирішуються окремо: спочатку формується портфель проєктів, а потім планується їх виконання. Однак інтегрований підхід, що одночасно вирішує задачу відбору проєктів та їх планування, може призводити до кращих результатів [45]. Це особливо актуально для ситуацій, коли ресурсні обмеження є критичним фактором вибору проєктів.

Візуалізація та інтерпретація результатів оптимізації є важливим аспектом практичного застосування. Діаграми Ганта, ресурсні гістограми, графіки завантаження персоналу допомагають менеджерам проєктів аналізувати та коригувати отримані розклади. Сучасні системи управління проєктами надають інтерактивні інструменти візуалізації, що дозволяють виявляти вузькі місця, конфлікти ресурсів та потенційні ризики. Інтеграція алгоритмів оптимізації з такими системами підвищує практичну цінність теоретичних розробок та сприяє їх впровадженню у реальні бізнес-процеси.

Важливим напрямком розвитку є інтеграція методів оптимізації з системами прийняття рішень (Decision Support Systems). Такі системи не лише знаходять оптимальні або близькі до оптимальних рішення, але й надають менеджерам інструменти для аналізу альтернатив, оцінки ризиків та вибору найбільш прийнятного варіанту з урахуванням неформалізованих критеріїв.

Поєднання кількісних методів оптимізації з якісними експертними оцінками дозволяє приймати більш обґрунтовані управлінські рішення.

Застосування хмарних обчислень та розподілених систем відкриває нові можливості для вирішення великомасштабних задач оптимізації. Паралельні версії метаевристичних алгоритмів, що виконуються на кластерах серверів або в хмарному середовищі, дозволяють значно скоротити час обчислень та досліджувати більші простори рішень. Це особливо актуально для великих аутсорсингових компаній з сотнями проєктів та тисячами спеціалістів, де задачі оптимізації мають велику розмірність. Розвиток технологій паралельних обчислень сприяє практичному впровадженню складних алгоритмів оптимізації.

Окремим напрямком є розробка онлайн-алгоритмів, що здатні адаптуватися до змін у реальному часі. На відміну від офлайн-алгоритмів, які отримують всю інформацію наперед, онлайн-алгоритми приймають рішення поступово, реагуючи на нові події: появу нових проєктів, зміни пріоритетів, недоступність ресурсів. Реактивне планування та методи перепланування (rescheduling) є важливими інструментами для підтримки актуальності розкладів в умовах динамічного середовища.

Інтеграція методів оптимізації з концепцією Industry 4.0 та цифровою трансформацією бізнесу є актуальним трендом сучасних досліджень [54]. Використання датчиків, Інтернету речей, великих даних та аналітики в реальному часі дозволяє отримувати точнішу інформацію про стан проєктів та ресурсів, що підвищує якість вхідних даних для алгоритмів оптимізації. Цифрові двійники проєктів (digital twins) дозволяють моделювати альтернативні сценарії та оцінювати наслідки управлінських рішень до їх реального впровадження.

Таким чином, проведений аналіз показав різноманітність існуючих методів та моделей оптимізації розподілу ресурсів між проєктами. Точні методи гарантують оптимальність, але обмежені задачами малої розмірності через експоненційну обчислювальну складність. Евристичні методи є

швидкими та простими у реалізації, але не гарантують якість рішення та можуть застрягати в локальних оптимумах. Метаевристичні алгоритми, зокрема генетичні алгоритми та алгоритм рою часток, забезпечують розумний компроміс між якістю рішення та обчислювальними витратами, що робить їх основним інструментом для практичного застосування. Гібридні підходи, що поєднують різні метаевристики або інтегрують їх з точними методами, демонструють найкращі результати на стандартних бенчмарках. Методи машинного навчання та квантові алгоритми відкривають нові перспективи для вирішення складних задач оптимізації. Для умов мультипроєктного середовища аутсорсингової компанії найбільш перспективними є гібридні метаевристичні алгоритми з урахуванням багатокомпетентності ресурсів, невизначеності параметрів, часу переміщення між проєктами та механізмів адаптації до динамічних змін.

2.2 Математичні підходи до формалізації задачі розподілу ресурсів в умовах обмежень

Формалізація задачі розподілу ресурсів є ключовим етапом побудови ефективних систем оптимізації в мультипроєктному середовищі. Математична модель визначає структуру задачі, множину допустимих рішень, критерії оптимальності та обмеження, що накладаються на процес планування. Точність та адекватність математичної формалізації безпосередньо впливає на якість отримуваних рішень та можливість їх практичного застосування. У даному підрозділі розглядаються основні математичні підходи до формалізації задачі розподілу ресурсів, включаючи змішане цілочисельне лінійне програмування, багатокритеріальну оптимізацію та моделі з урахуванням невизначеності [55].

Базовою математичною моделлю для задачі планування проєктів з обмеженнями на ресурси є модель RCPSP (Resource-Constrained Project

Scheduling Problem). Формальна постановка задачі передбачає визначення множин, параметрів, змінних рішення, цільової функції та обмежень. Нехай задано проєкт, що складається з множини робіт (активностей) $A = \{0, 1, 2, \ldots, n, n+1\}$, де 0 та $n+1$ є фіктивними роботами початку та завершення проєкту. Множина відновлюваних ресурсів позначається $K = \{1, 2, \ldots, m\}$, а множина часових періодів – $T = \{0, 1, 2, \ldots, T_{max}\}$ [56].

Параметри моделі включають: тривалість кожної роботи d_i для $i \in A$; потребу роботи i в ресурсі k, що позначається r_{ik}; доступну кількість ресурсу k в кожному періоді R_k; множину пар робіт, пов'язаних відношенням передування $E \subseteq A \times A$. Класифікація параметрів математичної моделі наведена в таблиці 2.4.

Таблиця 2.4 – Параметри математичної моделі RCPSP

Позначення	Тип	Опис
A	Множина	Множина робіт (активностей) проєкту
K	Множина	Множина типів відновлюваних ресурсів
E	Множина	Множина дуг графа передування (пари робіт)
d_i	Параметр	Тривалість виконання роботи i
r_{ik}	Параметр	Потреба роботи i в ресурсі типу k
R_k	Параметр	Доступна кількість ресурсу типу k
S_i	Змінна	Час початку виконання роботи i
x_{it}	Змінна	Бінарна змінна: 1, якщо робота i починається в момент t

Джерело: складено автором на основі [55, 56]

Формулювання на основі змішаного цілочисельного лінійного програмування (Mixed-Integer Linear Programming, MILP) є одним з найбільш поширених підходів до математичної формалізації задачі RCPSP. MILP-моделі дозволяють використовувати потужні комерційні та відкриті солвери (CPLEX, Gurobi, SCIP, HiGHS) для знаходження оптимальних або близьких до оптимальних рішень [57]. Найбільш поширеним є формулювання з індексацією за часом (time-indexed formulation), де бінарні змінні x_{it} вказують, чи починається робота i в момент часу t:

$$x_{it} \in \{0,1\}, \forall i \in A, \forall t \in \{ES_i, \ldots, LS_i\} \quad (2.10)$$

де ES_i та LS_i – найраніший та найпізніший можливий час початку роботи i, обчислені методом критичного шляху.

Цільова функція мінімізації загальної тривалості проєкту (makespan) формулюється як мінімізація часу завершення фіктивної останньої роботи:

$$min\ C_{max} = \sum_{t \in T} t \cdot x_{n+1,t}\ , \quad (2.11)$$

Обмеження унікальності призначення гарантує, що кожна робота починається рівно один раз:

$$\sum_{t=ES_i} LS_i\, x_{it} = 1, \forall i \in A\ , \quad (2.12)$$

Обмеження передування забезпечує, що робота-наступник не може розпочатися до завершення роботи-попередника:

$$\sum_{t \in T} t \cdot x_{jt} \geq \sum_{t \in T} (t + d_i) \cdot x_{it}, \forall (i,j) \in E\ , \quad (2.13)$$

Ресурсні обмеження гарантують, що в кожний момент часу сумарне споживання кожного ресурсу не перевищує його доступної кількості:

$$\sum_{i \in A} r_{ik} \cdot \sum_{\tau = max(0, t-d_i+1)} t\, x_{i\tau} \leq R_k,\ \forall k \in K, \forall t \in T\ , \quad (2.14)$$

Порівняння різних MILP-формулювань для задачі RCPSP наведено в таблиці 2.5. Кожне формулювання має свої переваги та недоліки з точки зору розміру моделі, сили лінійної релаксації та ефективності розв'язання.

Таблиця 2.5 – Порівняння MILP-формулювань для RCPSP

Формулювання	Кількість змінних	LP-релаксація	Застосування
Time-indexed (DT)	O(n·T)	Середня	Невеликий горизонт
Event-based (Start/End)	O(n²)	Слабка	Великий горизонт
On/Off формулювання	O(n·E)	Слабка	Кумулятивні задачі
Flow-based	O(n²·T)	Сильна	Потокові моделі
Feasible Subset (FSS)	O(2ⁿ)	Найсильніша	Малі задачі

Джерело: складено автором на основі [56, 57]

Для мультипроєктного середовища базова модель RCPSP розширюється до моделі RCMPSP (Resource-Constrained Multi-Project Scheduling Problem). Нехай задано множину проєктів $P = \{1, 2, \ldots, p\}$, кожен з яких складається з множини робіт A_p. Всі проєкти використовують спільний пул ресурсів. Математична модель RCMPSP включає додаткові індекси для ідентифікації належності роботи до конкретного проєкту [55]:

$$\sum_{p \in P} \sum_{i \in A_p} r_{pik} \cdot a_{pit} \leq R_k, \forall k \in K, \forall t \in T, \qquad (2.15)$$

де a_{pit} – бінарна змінна, що дорівнює 1, якщо робота i проєкту p виконується в момент часу t.

Багатокритеріальна оптимізація є важливим напрямком формалізації задач розподілу ресурсів, оскільки в реальних умовах менеджери проєктів зазвичай враховують кілька конфліктуючих критеріїв одночасно. Типовими цільовими функціями є мінімізація загальної тривалості, мінімізація затримок відносно директивних термінів, мінімізація витрат, максимізація рівномірності завантаження ресурсів [58]. Багатокритеріальна задача формулюється наступним чином:

$$min\, F(x) = [f_1(x),\, f_2(x), \ldots, f_m(x)]^T \qquad (2.16)$$

Рішення багатокритеріальної задачі є Парето-оптимальним, якщо не існує іншого допустимого рішення, яке б покращувало хоча б один критерій

без погіршення інших. Множина всіх Парето-оптимальних рішень утворює фронт Парето. Для формалізації Парето-домінування використовується наступне визначення [59]:

$$x1 \prec x2 \Leftrightarrow (\forall i: fi(x1) \leq fi(x2)) \wedge (\exists j: fj(x1) < fj(x2)), \quad (2.17)$$

Для практичного вирішення багатокритеріальних задач використовуються різні підходи скаляризації. Метод зважених сум об'єднує кілька критеріїв в один скалярний показник:

$$\min F(x) = \sum_{i=1}^{m} w_i \cdot f_i(x), \text{ де } \sum_i w_i = 1, w_i \geq 0 , \quad (2.18)$$

Метод ε-обмежень (ε-constraint method) оптимізує один критерій, перетворюючи інші на обмеження:

$$min\ f_1(x), \text{при } f_i(x) \leq \varepsilon_i, i = 2, \ldots, m , \quad (2.19)$$

Алгоритм NSGA-II (Non-dominated Sorting Genetic Algorithm II) є одним з найефективніших еволюційних алгоритмів для багатокритеріальної оптимізації. Він використовує механізм недомінованого сортування для ранжування рішень та метрику crowding distance для підтримки різноманітності на фронті Парето [59]. Удосконалена версія NSGA-III використовує референтні точки та призначена для задач з великою кількістю критеріїв.

Модель з багатокомпетентними ресурсами (Multi-Skill RCPSP, MSRCPSP) є важливим розширенням для IT-галузі, де спеціалісти володіють різними навичками з різним рівнем компетентності. Нехай S – множина навичок, $skill_{rs} \in [0, 1]$ – рівень володіння навичкою s ресурсом r, req_{is} – вимога роботи i до навички s [60]. Обмеження компетенцій формулюється:

$$y_{rit} \cdot skill_{rs} \geq y_{rit} \cdot req_{is}, \forall i, r, s, t\,, \qquad (2.20)$$

де y_{rit} – бінарна змінна призначення ресурсу r на роботу i в період t.

Врахування невизначеності в математичних моделях є критичним для реальних застосувань. Стохастичне програмування моделює невизначені параметри (тривалості робіт, доступність ресурсів) як випадкові величини з відомими розподілами [61]. Двоетапна стохастична модель включає рішення першого етапу (планування) та рішення другого етапу (реактивні коригування):

$$min\ c^T x + E_\xi[Q(x, \xi)]\ , \qquad (2.21)$$

де x – рішення першого етапу,

$Q(x, \xi)$ – оптимальне значення цільової функції другого етапу при реалізації сценарію ξ,

E_ξ – математичне сподівання за всіма сценаріями.

Робастна оптимізація не потребує знання точних розподілів невизначених параметрів, а використовує інтервали невизначеності. Підхід Берцімаса-Сіма контролює рівень консервативності через параметр бюджету невизначеності [62]:

$$\sum_{i \in A(t)} r_{ik} + max_{\{S \subseteq A(t): |S| \leq \Gamma\}} \sum_{i \in S} \hat{r}_{ik} \leq R_k\,, \qquad (2.22)$$

де $\hat{r}_{ik}$ – максимальне відхилення потреби в ресурсі від номінального значення.

Порівняння підходів до формалізації невизначеності наведено в табл. 2.6.

Таблиця 2.6 – Підходи до математичної формалізації невизначеності

Підхід	Вхідні дані	Результат	Складність
Стохастичне програмування	Ймовірнісні розподіли параметрів	Оптимальне в середньому рішення	Висока
Робастна оптимізація	Інтервали невизначеності, параметр Г	Рішення, стійке до найгіршого сценарію	Середня
Нечітке програмування	Функції належності, нечіткі числа	Рішення з урахуванням нечіткості	Середня
Сценарний аналіз	Дискретна множина сценаріїв	Рішення для кожного сценарію	Низька

Джерело: складено автором на основі [61, 62]

Модель з переривчастим виконанням робіт (Pre-emptive RCPSP) дозволяє переривати виконання роботи та продовжувати її пізніше. Це актуально для ситуацій, коли ресурс терміново потрібен для іншої критичної роботи [56]. Введення змінних, що відстежують виконаний обсяг роботи, ускладнює модель, але підвищує її гнучкість:

$$\sum_{t \in T} z_{it} = d_i, \forall i \in A\,, \tag{2.23}$$

де z_{it} – бінарна змінна, що дорівнює 1, якщо робота i виконується в період t.

Модель з часом переміщення ресурсів (Resource Transfer Times) враховує час, необхідний для переміщення ресурсу між різними роботами або проєктами. Це особливо актуально для розподілених команд, де час на передачу знань та адаптацію може бути значним [63]:

$$S_j \geq S_i + d_i + \tau_{ij} - M \cdot (1 - y_{rij})\,, \tag{2.24}$$

де τ_{ij} – час переміщення між роботами i та j,

y_{rij} – бінарна змінна, що дорівнює 1, якщо ресурс r послідовно виконує роботи i та j,

M – достатньо велике число (Big-M).

Модель з урахуванням якості та переробки інтегрує механізми передачі якості між роботами. Дефекти на ранніх етапах можуть каскадно поширюватися на наступні роботи, збільшуючи обсяг переробки [60]. Якість виконання роботи залежить від рівня компетентності призначеного ресурсу:

$$q_i = \sum_{r \in R} y_{ri} \cdot skill_r, s(i) , \quad (2.25)$$

де q_i – показник якості виконання роботи I,

$s(i)$ – навичка, необхідна для виконання роботи i.

Лінеаризація нелінійних виразів є важливим аспектом побудови MILP-моделей. Добуток бінарної та неперервної змінних можна лінеаризувати за допомогою стандартних технік Big-M або методу McCormick envelopes [57]. Для добутку бінарних змінних використовуються допоміжні обмеження, що зберігають лінійність моделі.

Валідні нерівності (valid inequalities) та методи відсікання (cutting planes) використовуються для посилення LP-релаксації MILP-моделей. Для задачі RCPSP розроблено специфічні класи нерівностей: cover inequalities, clique inequalities, та precedence-based cuts [56]. Додавання валідних нерівностей дозволяє зменшити простір пошуку та прискорити роботу Branch-and-Bound алгоритмів.

Декомпозиційні методи дозволяють розбивати великі задачі на менші підзадачі. Метод декомпозиції Бендерса розділяє задачу на головну проблему та підпроблему, що вирішуються ітеративно. Лагранжева релаксація

послаблює деякі обмеження, включаючи їх у цільову функцію з множниками Лагранжа [64]. Ці методи особливо ефективні для задач великої розмірності.

Таким чином, математична формалізація задачі розподілу ресурсів охоплює широкий спектр підходів: від базових MILP-моделей до складних багатокритеріальних та стохастичних формулювань. Вибір конкретного підходу залежить від характеристик задачі, доступних обчислювальних ресурсів та вимог до точності рішення. Для умов мультипроєктного середовища аутсорсингової IT-компанії найбільш адекватними є моделі, що враховують багатокомпетентність ресурсів, невизначеність параметрів, час переміщення між проєктами та механізми передачі якості.

Важливим аспектом формалізації є визначення адекватної цільової функції. В реальних умовах менеджери проєктів рідко оптимізують лише один показник. Типовими критеріями є мінімізація загальної тривалості портфеля проєктів, мінімізація сумарних затримок відносно директивних термінів замовників, мінімізація витрат на понаднормову роботу, максимізація рівномірності завантаження спеціалістів, мінімізація кількості переключень між проєктами. Кожен з цих критеріїв може бути формалізований у вигляді математичного виразу та включений до моделі оптимізації.

Формалізація обмежень вимагає детального аналізу бізнес-правил та операційних політик компанії. Окрім базових обмежень передування та ресурсів, практичні моделі можуть включати: обмеження на максимальну тривалість робочого дня та тижня, обмеження на одночасне залучення спеціаліста до кількох проєктів, обмеження на призначення певних спеціалістів на конкретні проєкти через конфлікт інтересів або вимоги замовника, обмеження на мінімальний розмір команди проєкту. Кожне з цих обмежень може бути виражене за допомогою лінійних або нелінійних нерівностей.

Моделювання календарних обмежень є важливим для реалістичного планування. Доступність ресурсів може змінюватися в часі через відпустки, лікарняні, національні свята, навчання та інші фактори. Для формалізації

календарних обмежень вводяться параметри доступності ресурсу в кожному періоді [63]. Це особливо актуально для аутсорсингових компаній, що працюють із замовниками з різних часових поясів та повинні враховувати різні календарі вихідних та святкових днів.

Формалізація залежностей між роботами може включати не лише базові відношення передування типу «finish-to-start», але й інші типи залежностей: «start-to-start» (робота може розпочатися одночасно з іншою), «finish-to-finish» (робота повинна завершитися одночасно з іншою), «start-to-finish» (робота може завершитися лише після початку іншої). Крім того, можуть застосовуватися часові лаги – мінімальні та максимальні затримки між пов'язаними роботами [56].

Модель з невідновлюваними ресурсами (non-renewable resources) розширює базову формалізацію для врахування ресурсів, що споживаються в процесі виконання робіт і не відновлюються після завершення. Прикладами є бюджет проєкту, витратні матеріали, ліцензії на програмне забезпечення з обмеженою кількістю використань. Обмеження на невідновлювані ресурси формулюються як загальна сума споживання за весь період планування:

$$\sum_{i \in A} nr_{ik} \leq NR_k, \forall k \in K_{nr}, \tag{2.26}$$

де nr_{ik} – споживання невідновлюваного ресурсу k роботою I,

NR_k – загальна доступна кількість ресурсу k,

K_{nr} – множина типів невідновлюваних ресурсів.

Ієрархічна декомпозиція моделі дозволяє розділити задачу на рівні: стратегічний (вибір проєктів для портфеля), тактичний (планування ресурсів на горизонті кількох місяців) та операційний (детальне планування на найближчий спринт або тиждень). Кожен рівень характеризується різним ступенем деталізації та частотою перепланування. Математичні моделі для різних рівнів можуть відрізнятися за структурою та складністю.

Формалізація динамічних аспектів задачі передбачає моделювання надходження нових проєктів та робіт у процесі виконання. Онлайн-версія задачі планування характеризується тим, що інформація про майбутні роботи невідома на момент прийняття рішень. Для таких задач застосовуються методи ковзного горизонту (rolling horizon), де планування виконується на обмежений горизонт з періодичним оновленням при надходженні нової інформації [55].

Моделювання витрат та доходів дозволяє оптимізувати економічні показники проєктів. Чиста приведена вартість (Net Present Value, NPV) є популярним критерієм, що враховує часову вартість грошей. Для максимізації NPV необхідно враховувати моменти надходження платежів від замовників та моменти виплат постачальникам та персоналу. Формалізація NPV включає дисконтування грошових потоків:

$$NPV = \sum_{i \in A} cf_i \cdot e^{-r \cdot (S_i + d_i)}, \quad (2.27)$$

де cf_i – грошовий потік, пов'язаний із завершенням роботи I,

r – ставка дисконтування.

Інтеграція моделі розподілу ресурсів з іншими підсистемами управління проєктами є важливим практичним аспектом. Зв'язок з системою управління ризиками передбачає врахування ймовірності реалізації різних сценаріїв та їх впливу на розклад. Зв'язок з системою управління якістю дозволяє враховувати вплив рішень щодо призначення ресурсів на якість виконання робіт. Зв'язок з фінансовою системою забезпечує контроль бюджетних обмежень та оптимізацію грошових потоків.

Обчислювальна складність MILP-моделей залежить від кількості змінних, кількості обмежень та структури задачі. Для задач RCPSP характерна експоненційна складність у загальному випадку, оскільки задача є NP-складною. Практичні обмеження на час розв'язання визначають максимальний розмір задач, що можуть бути вирішені точними методами. Для великих задач

з сотнями робіт та десятками ресурсів точні методи стають непрактичними, що обумовлює застосування наближених алгоритмів [64].

Сучасні тенденції у формалізації задач розподілу ресурсів включають інтеграцію з методами машинного навчання. Нейронні мережі можуть використовуватися для прогнозування тривалості робіт, оцінки ризиків затримок, рекомендації оптимальних призначень ресурсів. Навчання з підкріпленням дозволяє агенту навчитися приймати рішення щодо розподілу ресурсів на основі зворотного зв'язку від середовища. Гібридні підходи, що комбінують математичну оптимізацію з машинним навчанням, демонструють перспективні результати на практичних задачах [59].

Комбінування різних формалізацій дозволяє побудувати комплексну модель, що відображає специфіку реального бізнес-середовища та забезпечує ефективну підтримку прийняття рішень щодо розподілу ресурсів. Для умов мультипроєктного середовища аутсорсингової IT-компанії рекомендується використовувати багатокритеріальну MILP-модель з урахуванням багатокомпетентності ресурсів, робастною формалізацією невизначеності та механізмами декомпозиції для забезпечення масштабованості.

Практична реалізація математичних моделей потребує вибору відповідного програмного забезпечення. Комерційні солвери CPLEX та Gurobi забезпечують найвищу продуктивність для MILP-моделей, але потребують ліцензування. Відкриті альтернативи, такі як SCIP, HiGHS та OR-Tools, надають безкоштовний доступ до потужних алгоритмів оптимізації. Вибір солвера залежить від розміру задачі, бюджетних обмежень та вимог до інтеграції з існуючими системами компанії.

2.3 Обґрунтування вибору методів моделювання для задачі розподілу ресурсів в аутсорсинговій компанії

Вибір адекватних методів моделювання для задачі розподілу ресурсів є критичним етапом розробки системи підтримки прийняття рішень в

аутсорсинговій ІТ-компанії. Цей вибір повинен базуватися на комплексному аналізі специфічних характеристик предметної області, вимог до точності та швидкості отримання рішень, наявних обчислювальних ресурсів та рівня зрілості процесів управління проєктами в організації. У даному підрозділі здійснюється обґрунтування вибору методів моделювання на основі систематичного аналізу критеріїв відповідності та порівняльної оцінки альтернативних підходів [65].

Аутсорсингова ІТ-компанія характеризується низкою специфічних особливостей, що суттєво впливають на вибір методів оптимізації розподілу ресурсів. По-перше, основним ресурсом є висококваліфіковані спеціалісти, кожен з яких володіє унікальною комбінацією компетенцій з різним рівнем майстерності. По-друге, компанія одночасно виконує кілька проєктів для різних замовників з різними пріоритетами та вимогами. По-третє, середовище характеризується високим рівнем невизначеності: змінюються вимоги замовників, виникають непередбачені технічні проблеми, змінюється доступність ресурсів через хвороби, відпустки або звільнення [66].

Таблиця 2.7 – Критерії вибору методів моделювання для аутсорсингової ІТ-компанії

Критерій	Опис	Важливість для ІТ-аутсорсингу
Багатокомпетентність	Врахування різних навичок ресурсів з різним рівнем майстерності	Критична
Невизначеність	Стохастичність тривалості робіт та доступності ресурсів	Висока
Багатокритеріальність	Оптимізація кількох конфліктуючих цілей одночасно	Висока
Обчислювальна ефективність	Час отримання рішення за прийнятний період	Висока
Масштабованість	Здатність працювати із зростаючим обсягом даних	Середня
Інтегрованість	Можливість інтеграції з Jira, MS Project, ERP	Середня
Інтерпретованість	Зрозумілість результатів для менеджерів	Середня

Джерело: складено автором на основі [65, 66, 67]

Для систематизації процесу вибору методів моделювання необхідно визначити критерії оцінки. На основі аналізу літератури та практичного

досвіду виділено такі ключові критерії: здатність моделювати багатокомпетентність ресурсів, можливість врахування невизначеності, підтримка багатокритеріальної оптимізації, обчислювальна ефективність, масштабованість, простота інтеграції з існуючими системами, інтерпретованість результатів. Кожен з цих критеріїв має різну вагу залежно від специфіки конкретної компанії та її пріоритетів [67].

Для формалізації процесу вибору методів введемо кількісну оцінку відповідності кожного методу критеріям. Нехай $M = \{m_1, m_2, \ldots, m_n\}$ – множина методів-кандидатів, $C = \{c_1, c_2, \ldots, c_k\}$ – множина критеріїв оцінки. Для кожного методу m_i визначається вектор оцінок $s_i = (s_{i1}, s_{i2}, \ldots, s_{ik})$, де $s_{ij} \in [0, 1]$ – оцінка відповідності методу i критерію j. Інтегральна оцінка методу обчислюється як зважена сума:

$$S_i = \sum_{j=1}^{k} w_j \cdot s_{ij}, \text{де} \sum_{j=1}^{k} w_j = 1 \quad (2.28)$$

де w_j – вага критерію j, що відображає його відносну важливість для організації.

Розглянемо детальніше основні класи методів та їх відповідність визначеним критеріям. Точні методи, такі як MILP-формулювання з використанням комерційних солверів (CPLEX, Gurobi), гарантують знаходження оптимального рішення, але мають експоненційну обчислювальну складність. Для типової аутсорсингової компанії з 50-100 спеціалістами та 10-20 одночасними проєктами розмірність задачі може досягати тисяч змінних та обмежень, що робить точні методи непрактичними для оперативного планування [68].

Метаевристичні алгоритми забезпечують розумний компроміс між якістю рішення та часом обчислення. Генетичні алгоритми (GA) демонструють високу ефективність глобального пошуку завдяки механізмам кросовера та мутації. Алгоритм рою часток (PSO) характеризується швидкою збіжністю та простотою реалізації. Алгоритм імітації відпалу (SA) ефективно

уникає локальних мінімумів. Мурашині алгоритми (ACO) добре працюють для комбінаторних задач з дискретною структурою [69].

Гібридні алгоритми, що поєднують переваги різних метаевристик, демонструють найкращі результати на стандартних бенчмарках. Зокрема, гібрид GA-PSO використовує глобальний пошук генетичного алгоритму для диверсифікації популяції та локальний пошук PSO для інтенсифікації в перспективних регіонах простору рішень [70]. Формула оновлення позиції часток у гібридному алгоритмі включає додатковий компонент генетичної еволюції:

$$x_i(t+1) = x_i(t) + v_i(t+1) + \alpha \cdot G(x_i, x_{partner}) \qquad (2.29)$$

де α – коефіцієнт впливу генетичної компоненти;

G – функція генетичного оператора (кросовер або мутація).

Для задачі з багатокомпетентними ресурсами (MS-RCPSP) розроблено спеціалізовані алгоритми, що враховують специфіку призначення спеціалістів з різними навичками. Меметичний алгоритм MEMINV поєднує еволюційний пошук з методом інверсії для виявлення локальних екстремумів [67]. Результати обчислювальних експериментів на бенчмарку iMOPSE демонструють перевагу MEMINV над базовими метаевристиками.

Порівняльний аналіз методів за визначеними критеріями наведено в таблиці 2.8. Оцінки отримано на основі аналізу літератури та результатів обчислювальних експериментів на стандартних бенчмарках.

На основі проведеного аналізу гібридний алгоритм GA-PSO демонструє найвищу інтегральну оцінку (4.1 з 5.0 можливих) та є найбільш перспективним для застосування в умовах аутсорсингової IT-компанії. Цей алгоритм поєднує сильні сторони обох базових метаевристик: глобальний пошук генетичного алгоритму та швидку локальну збіжність PSO [68].

Таблиця 2.8 – Порівняльна оцінка методів оптимізації за критеріями

Метод	Багато-компет.	Невизна-ченість	Багато-критер.	Ефектив-ність	Масшта-бованість	Сума
MILP	0.9	0.7	0.6	0.3	0.2	2.7
GA	0.8	0.6	0.8	0.8	0.7	3.7
PSO	0.7	0.5	0.7	0.9	0.8	3.6
Гібрид GA-PSO	0.9	0.7	0.9	0.8	0.8	4.1
MEMINV	0.95	0.6	0.7	0.85	0.75	3.85
NSGA-II/III	0.8	0.6	0.95	0.7	0.7	3.75

Джерело: складено автором на основі [67, 68, 69, 70]

Для врахування багатокритеріальності задачі пропонується інтегрувати гібридний GA-PSO з механізмом недомінованого сортування NSGA-II. Це дозволяє формувати множину Парето-оптимальних рішень, що представляють різні компроміси між критеріями. Особа, що приймає рішення (менеджер проєктів), може обрати найбільш прийнятний варіант з фронту Парето [72].

Для врахування невизначеності пропонується застосувати робастний підхід Берцімаса-Сіма, який не потребує знання точних ймовірнісних розподілів невизначених параметрів. Параметр бюджету невизначеності Γ дозволяє контролювати рівень консервативності рішення та адаптувати його до схильності менеджменту до ризику [73]. Робастне обмеження на ресурси формулюється:

$$\sum_{i \in A(t)} r_{ik} \; + \; \Gamma \; \cdot \; max_{i \in A(t)} \, \hat{r}_{ik} \; \leq \; Rk, \forall k, \forall t \; , \qquad (2.30)$$

Математична модель для задачі розподілу ресурсів в аутсорсинговій IT-компанії повинна належати до класу MS-RCMPSP (Multi-Skill Resource-Constrained Multi-Project Scheduling Problem). Ця модель є розширенням базової RCPSP та включає: множину проєктів зі спільним пулом ресурсів, обмеження на компетенції спеціалістів, врахування рівня майстерності при призначенні, можливість часткової зайнятості на кількох проєктах [74].

Обґрунтування вибору методу повинно також враховувати практичні аспекти впровадження. Гібридний GA-PSO має низку переваг: відносна

простота реалізації на мовах Python та Java, наявність готових бібліотек (DEAP, jMetal, Optaplanner), можливість паралелізації для прискорення обчислень, інтуїтивна інтерпретація параметрів алгоритму. Порівняння характеристик різних методів з точки зору практичного впровадження наведено в таблиці 2.9.

Важливим аспектом є визначення критеріїв оптимізації для аутсорсингової IT-компанії. На основі аналізу бізнес-процесів та потреб стейкхолдерів виділено такі цільові функції: мінімізація сумарних затримок проєктів відносно директивних термінів (пріоритет для замовників), максимізація рівномірності завантаження спеціалістів (пріоритет для HR), мінімізація кількості переключень спеціалістів між проєктами (пріоритет для продуктивності), максимізація відповідності компетенцій спеціалістів вимогам завдань (пріоритет для якості).

Таблиця 2.9 – Практичні аспекти впровадження методів оптимізації

Метод	Бібліотеки	Складність реалізації	Паралелізація
MILP	CPLEX, Gurobi, OR-Tools	Середня	Вбудована в солвери
GA	DEAP, jMetal, PyGAD	Низька	Легка (острівна модель)
PSO	PySwarms, DEAP	Низька	Легка (синхронна)
Гібрид GA-PSO	Власна реалізація	Середня	Потребує проектування
NSGA-II	jMetal, Platypus, pymoo	Середня	Підтримується

Джерело: складено автором

Формалізація цільових функцій для багатокритеріальної оптимізації передбачає наступні математичні вирази. Мінімізація затримок:

$$f_1 = \sum_{p \in P} w_p \cdot max(0, C_p - D_p) \rightarrow min \ , \qquad (2.31)$$

де C_p – фактичний час завершення проєкту p;

D_p – директивний термін;

w_p – вага пріоритету проєкту.

Мінімізація нерівномірності завантаження:

$$f_2 = \sum_{r \in R} \sum_{t \in T} (load_{rt} - load_{avg})^2 \rightarrow min\,, \qquad (2.32)$$

де $load_{rt}$ – завантаження ресурсу r в період t;

$load_{avg}$ – середнє завантаження.

Максимізація відповідності компетенцій:

$$f_3 = \sum_{i \in A} \sum_{r \in R} y_{ri} \cdot skill_{r,s(i)} \rightarrow max\,, \qquad (2.33)$$

де $skill_{r,s(i)}$ – рівень володіння ресурсом r навичкою s, необхідною для роботи i.

Архітектура запропонованої системи оптимізації передбачає ієрархічний підхід. На верхньому рівні здійснюється стратегічне планування розподілу ресурсів між проєктами на горизонті кількох місяців з використанням робастної оптимізації. На середньому рівні виконується тактичне планування на горизонті кількох тижнів з використанням гібридного GA-PSO. На нижньому рівні здійснюється оперативне коригування розкладу в реальному часі з використанням евристичних правил пріоритету [71].

Інтеграція з існуючими інструментами управління проєктами (Jira, MS Project, Asana) є критичною для практичного впровадження. Система повинна імпортувати дані про проєкти, завдання, ресурси та їх компетенції, виконувати оптимізацію розподілу, та експортувати результати у форматі, сумісному з використовуваними інструментами. API-інтерфейс забезпечує автоматизацію обміну даними та можливість інтеграції в існуючі бізнес-процеси.

Валідація обраних методів повинна здійснюватися на реальних даних компанії та стандартних бенчмарках. Для MS-RCPSP використовується бенчмарк iMOPSE, що містить 36 інстанцій різної складності [66]. Для мультипроєктних задач застосовуються модифіковані набори PSPLIB. Порівняння результатів з відомими найкращими рішеннями дозволяє оцінити ефективність розробленого алгоритму.

Обчислювальні експерименти на бенчмарках показують, що гібридний GA-PSO досягає відхилення від найкращих відомих рішень в межах 2-5% при часі обчислення 10-30 секунд для задач з 30-60 роботами. Для задач з 100+ роботами час обчислення зростає до кількох хвилин, що є прийнятним для тактичного планування. Паралелізація на багатоядерних процесорах дозволяє скоротити час обчислення в 3-4 рази.

Адаптація алгоритму до специфіки конкретної аутсорсингової компанії передбачає налаштування параметрів: розмір популяції, кількість ітерацій, коефіцієнти кросовера та мутації, ваги критеріїв. Для автоматичного налаштування параметрів можуть використовуватися методи мета-оптимізації, такі як irace або SMAC. Альтернативно, параметри можуть бути визначені експертами на основі пілотних експериментів.

Впровадження системи оптимізації розподілу ресурсів потребує організаційних змін: призначення відповідальних за підтримку системи, навчання менеджерів проєктів роботі з системою, інтеграція процесу оптимізації в регулярний цикл планування. Очікуваними ефектами від впровадження є: скорочення затримок проєктів на 15-25%, підвищення рівномірності завантаження спеціалістів на 20-30%, зменшення часу на ручне планування на 50-70%.

Обмеження запропонованого підходу включають: необхідність якісних вхідних даних про компетенції спеціалістів та оцінки тривалості робіт, складність врахування неформалізованих факторів (командна динаміка, особисті преференції), потребу в періодичному переналаштуванні параметрів при зміні характеру проєктів. Ці обмеження можуть бути частково подолані шляхом інтеграції з методами машинного навчання для прогнозування та адаптації [72].

Таким чином, на основі комплексного аналізу критеріїв вибору та порівняльної оцінки альтернативних методів обґрунтовано вибір гібридного алгоритму GA-PSO як основного методу оптимізації розподілу ресурсів в аутсорсинговій IT-компанії. Цей алгоритм доповнюється механізмом недомінованого сортування NSGA-II для багатокритеріальної оптимізації та робастним підходом Берцімаса-Сіма для врахування невизначеності.

Детальний аналіз механізму гібридизації GA та PSO дозволяє виділити кілька стратегій інтеграції. Перша стратегія передбачає послідовне застосування алгоритмів: спочатку GA виконує глобальний пошук для формування початкової популяції якісних рішень, а потім PSO здійснює локальне покращення кращих індивідів. Друга стратегія передбачає паралельне виконання обох алгоритмів з періодичним обміном кращими рішеннями між популяціями. Третя стратегія інтегрує оператори обох алгоритмів в єдиний цикл еволюції, де кожна ітерація включає як генетичні оператори, так і оновлення швидкостей та позицій [68].

Для задачі розподілу ресурсів в IT-аутсорсингу найбільш ефективною виявляється третя стратегія з адаптивним управлінням балансом між глобальним та локальним пошуком. На початкових ітераціях алгоритм надає перевагу глобальному пошуку для дослідження простору рішень, а на пізніших ітераціях акцент зміщується на локальний пошук для уточнення знайдених рішень. Коефіцієнт балансу α змінюється динамічно за формулою:

$$\alpha(t) = \alpha_{max} - (\alpha_{max} - \alpha_{min}) \cdot (t / T_{max}), \qquad (2.34)$$

де α_{max} та α_{min} – максимальне та мінімальне значення коефіцієнта балансу;

t – поточна ітерація;

T_{max} – максимальна кількість ітерацій.

Кодування рішення є критичним аспектом для ефективності алгоритму. Для задачі MS-RCMPSP пропонується двохкомпонентне кодування: перша компонента визначає послідовність виконання робіт (activity list), друга компонента визначає призначення ресурсів на кожну роботу (resource assignment). Такий підхід дозволяє незалежно оптимізувати часову та ресурсну складові рішення [66].

Оператори кросовера адаптовані до специфіки задачі планування. Для компоненти послідовності робіт використовується двоточковий кросовер з репарацією для забезпечення допустимості рішення (збереження обмежень передування). Для компоненти призначення ресурсів використовується

рівномірний кросовер з перевіркою відповідності компетенцій. Оператор мутації включає три типи модифікацій: swap (обмін позицій двох робіт), insert (переміщення роботи на нову позицію), change (зміна призначеного ресурсу).

Механізм селекції в гібридному алгоритмі поєднує турнірну селекцію генетичного алгоритму з механізмом оновлення персонального та глобального кращих рішень PSO. Розмір турніру динамічно змінюється: на початкових ітераціях використовується менший турнір (2-3 особини) для збереження різноманітності популяції, на пізніших ітераціях розмір турніру збільшується (5-7 особин) для підвищення селективного тиску [69].

Для підтримки різноманітності популяції та уникнення передчасної збіжності застосовуються додаткові механізми. Нішингування (niching) розділяє популяцію на підгрупи, що еволюціонують в різних регіонах простору рішень. Перезапуск (restart) ініціює частину популяції випадковими рішеннями при виявленні стагнації. Адаптивне управління параметрами автоматично налаштовує ймовірності кросовера та мутації на основі динаміки поліпшення рішень.

Інтеграція з механізмом недомінованого сортування NSGA-II дозволяє обробляти багатокритеріальність задачі. Кожне рішення оцінюється за всіма цільовими функціями та класифікується за рівнем Парето-домінування. Crowding distance забезпечує рівномірний розподіл рішень вздовж фронту Парето. Еліт\ ний механізм зберігає недоміновані рішення з покоління в покоління [72].

Оцінка фітнес-функції для багатокритеріальної задачі потребує спеціального підходу. Замість скалярного значення використовується вектор значень цільових функцій. Порівняння рішень здійснюється на основі відношення Парето-домінування. Для задач з великою кількістю критеріїв (більше 3-4) застосовуються референтні точки NSGA-III, що забезпечують кращу збіжність та різноманітність фронту.

Управління обчислювальними ресурсами є важливим аспектом практичного застосування. Алгоритм повинен мати можливість зупинки за заданим критерієм: максимальна кількість ітерацій, максимальний час виконання, досягнення цільового значення функції, стагнація поліпшення. Для оперативного

планування типовим є обмеження часу виконання 30-60 секунд, для тактичного планування – 5-10 хвилин.

Механізм обробки обмежень включає кілька стратегій. Для «жорстких» обмежень (передування, компетенції) застосовується репарація – автоматичне виправлення недопустимих рішень. Для «м'яких» обмежень (директивні терміни, переваги) застосовується штрафна функція, що погіршує оцінку рішення пропорційно ступеню порушення. Баланс між штрафами за різні порушення налаштовується відповідно до бізнес-пріоритетів.

Паралелізація алгоритму реалізується на кількох рівнях. На рівні популяції використовується острівна модель: популяція розділяється на підпопуляції, що еволюціонують паралельно на різних процесорних ядрах з періодичним обміном кращими рішеннями (міграція). На рівні оцінки фітнесу паралельно обчислюються значення цільових функцій для різних рішень. Ефективність паралелізації залежить від співвідношення часу обчислення та комунікації [70].

Верифікація рішень перед видачею користувачу включає перевірку: допустимості (задоволення всіх обмежень), повноти (всі роботи заплановані), консистентності (відсутність конфліктів призначень). У разі виявлення проблем система автоматично застосовує механізми репарації або повідомляє користувача про неможливість знайти допустиме рішення.

Візуалізація результатів є критичною для прийняття рішень менеджерами. Система повинна надавати: діаграму Ганта з відображенням розкладу робіт та призначення ресурсів, графік завантаження ресурсів за періодами, візуалізацію фронту Парето для вибору компромісного рішення, таблицю ключових показників ефективності (KPI). Інтерактивний інтерфейс дозволяє користувачу модифікувати рішення вручну з автоматичним перерахунком показників.

Математична модель належить до класу MS-RCMPSP з урахуванням специфіки багатокомпетентних ресурсів. Запропонована архітектура системи передбачає ієрархічний підхід з інтеграцією з існуючими інструментами управління проєктами. Валідація на стандартних бенчмарках підтверджує ефективність обраних методів. Подальший розвиток системи може включати

інтеграцію методів машинного навчання для прогнозування тривалості робіт та адаптивного налаштування параметрів алгоритму.

Важливим напрямком розвитку є інтеграція з методами глибокого навчання з підкріпленням (Deep Reinforcement Learning). Агент DRL навчається приймати рішення про призначення ресурсів на основі поточного стану системи, отримуючи винагороду за досягнення цільових показників. Попередньо навчена модель може використовуватися для швидкої ініціалізації гібридного алгоритму або для прийняття рішень в реальному часі при обмеженому часі на обчислення [73].

Застосування трансферного навчання дозволяє використовувати знання, отримані при вирішенні подібних задач, для прискорення оптимізації нових проєктів. Модель, навчена на історичних даних компанії, ініціалізує популяцію алгоритму рішеннями, близькими до оптимальних для типових конфігурацій проєктів. Це особливо ефективно для компаній з повторюваними типами проєктів та стабільним складом команди.

Підсумовуючи, вибір гібридного алгоритму GA-PSO з інтеграцією NSGA-II та робастної оптимізації обґрунтовано на основі систематичного аналізу відповідності методів критеріям, специфічним для аутсорсингової IT-компанії. Запропонований підхід забезпечує баланс між якістю рішень та обчислювальною ефективністю, підтримує багатокритеріальну оптимізацію та врахування невизначеності, може бути інтегрований з існуючими інструментами управління проєктами. Практична реалізація потребує налаштування параметрів алгоритму, розробки інтерфейсу користувача та інтеграції з корпоративними системами компанії.

РОЗДІЛ 3
РОЗРОБКА ТА АПРОБАЦІЯ МОДЕЛІ ОПТИМАЛЬНОГО РОЗПОДІЛУ РЕСУРСІВ В АУТСОРСИНГОВІЙ КОМПАНІЇ

3.1 Розробка моделі розподілу ресурсів у мультипроєктному середовищі

На основі проведеного в попередніх розділах аналізу теоретичних засад та обґрунтування вибору методів моделювання в даному підрозділі представлено розробку комплексної моделі розподілу ресурсів у мультипроєктному середовищі аутсорсингової IT-компанії. Модель належить до класу MS-RCMPSP (Multi-Skill Resource-Constrained Multi-Project Scheduling Problem) та враховує специфічні особливості галузі: багатокомпетентність спеціалістів, одночасне виконання кількох проєктів, невизначеність параметрів та багатокритеріальність оптимізації.

Формальна постановка задачі передбачає визначення вхідних даних, змінних рішення, цільових функцій та обмежень. Вхідними даними моделі є: множина проєктів $P = \{1, 2, \dots, p\}$; множина робіт (завдань) кожного проєкту A_p для $p \in P$; множина спеціалістів (ресурсів) $R = \{1, 2, \dots, m\}$; множина компетенцій (навичок) $S = \{1, 2, \dots, s\}$; горизонт планування $T = \{1, 2, \dots, T_{max}\}$ [75].

Параметри моделі включають характеристики проєктів, робіт та ресурсів. Для кожного проєкту визначено: директивний термін завершення D_p, пріоритет w_p. Для кожної роботи визначено: тривалість d_{pi}, вимоги до компетенцій $req_{pis} \in [0, 1]$, множину попередників $Pred_{pi}$. Для кожного ресурсу визначено: рівень компетенції $skill_{rs} \in [0, 1]$, доступність в кожному періоді $avail_{rt} \in \{0, 1\}$. Повний перелік параметрів наведено в таблиці 3.1.

Змінні рішення визначають розклад виконання робіт та призначення ресурсів. Основною змінною є $x_{pirt} \in \{0, 1\}$ – бінарна змінна, що дорівнює 1, якщо ресурс r призначений на роботу i проєкту p в період часу t. Додатковою

змінною є $S_{pi} \geq 0$ – час початку виконання роботи i проєкту p. Час завершення роботи обчислюється як $C_{pi} = S_{pi} + d_{pi}$.

Таблиця 3.1 – Параметри розробленої математичної моделі

Позначення	Розмірність	Опис параметра
P, A_p, R, S	Множини	Проєкти, роботи, ресурси, компетенції
D_p, w_p	Дні, [0,1]	Директивний термін та пріоритет проєкту
d_{pi}, req_{pis}	Дні, [0,1]	Тривалість роботи, вимога до компетенції
$skill_{rs}$, $avail_{rt}$	[0,1], {0,1}	Рівень компетенції та доступність ресурсу
$Pred_{pi}$	Множина	Множина попередників роботи (залежності)
τ_{ij}	Дні	Час переміщення ресурсу між роботами i та j

Джерело: розроблено автором

Модель передбачає оптимізацію кількох цільових функцій одночасно.

Перша цільова функція – мінімізація сумарних зважених затримок проєктів відносно директивних термінів:

$$f_1 = \sum_{p \in P} w_p \cdot max(0, C_p - D_p) \rightarrow min\,, \qquad (3.1)$$

де $C_p = max_{i \in A_p} C_{pi}$ – час завершення проєкту p (час завершення останньої роботи).

Друга цільова функція – мінімізація нерівномірності завантаження ресурсів, що забезпечує збалансоване навантаження на спеціалістів:

$$f_2 = \sum_{r \in R} \sum_{t \in T} (load_{rt} - \overline{load_r})^2 \rightarrow min\,, \qquad (3.2)$$

де $load_{rt} = \sum_{p \in P} \sum_{i \in A_p} x_{pirt}$ – завантаження ресурсу r в період t;

$\overline{load_r}$ – середнє завантаження ресурсу за весь період планування.

Третя цільова функція – максимізація відповідності компетенцій спеціалістів вимогам завдань, що забезпечує якість виконання робіт:

$$f_3 = \sum_{p \in P} \sum_{i \in A_p} \sum_{r \in R} \sum_{s \in S} y_{pir} \cdot skill_{rs} \cdot req_{pis} \rightarrow max, \qquad (3.3)$$

де $y_{pir} \in \{0,1\}$ – бінарна змінна призначення ресурсу r на роботу i проєкту p.

Четверта цільова функція – мінімізація кількості переключень спеціалістів між проєктами, що зменшує втрати часу на контекстне переключення:

$$f_4 = \sum_{r \in R} \sum_{t \in T} switch_{rt} \rightarrow min\,, \tag{3.4}$$

де $switch_{rt}$ = 1, якщо ресурс r в період t переключається на інший проєкт порівняно з періодом $t-1$.

Обмеження моделі забезпечують допустимість отриманих рішень.

Обмеження передування гарантує, що робота не може розпочатися до завершення всіх її попередників:

$$S_{pi} \geq C_{pj}, \forall p \in P, \forall i \in A_p, \forall j \in Pred_{pi}\,, \tag{3.5}$$

Обмеження компетенцій гарантує, що призначений ресурс має достатній рівень компетенції для виконання роботи:

$$y_{pir} \cdot skill_{rs} \geq y_{pir} \cdot req_{pis}, \forall p, i, r, s\,, \tag{3.6}$$

Обмеження доступності гарантує, що ресурс призначається тільки в періоди своєї доступності:

$$x_{pirt} \leq avail_{rt}, \forall p, i, r, t\,, \tag{3.7}$$

Обмеження унікальності призначення гарантує, що кожна робота виконується рівно одним ресурсом:

$$\sum_{r \in R} y_{pir} = 1, \forall p \in P, \forall i \in A_p \quad (3.8)$$

Обмеження неперервності виконання гарантує, що кожна робота виконується без перерв:

$$\sum_{t=S_{pi}} S_{pi} + d_{pi} - 1 \, x_{pirt} = d_{pi} \cdot y_{pir}, \forall p, i, r \,, \quad (3.9)$$

Обмеження одночасного виконання гарантує, що кожен ресурс виконує не більше однієї роботи в кожен момент часу:

$$\sum_{p \in P} \sum_{i \in A_p} x_{pirt} \leq 1, \forall r \in R, \forall t \in T \,, \quad (3.10)$$

Архітектура розробленої системи оптимізації розподілу ресурсів представлена на рис. 3.1. Система складається з кількох модулів: модуль інтеграції даних забезпечує імпорт інформації з зовнішніх систем (Jira, MS Project, HR-система); модуль математичної моделі формалізує задачу у вигляді MS-RCMPSP; модуль оптимізації реалізує гібридний алгоритм GA-PSO з NSGA-II; модуль формування рішень генерує фронт Парето та візуалізує результати [76].

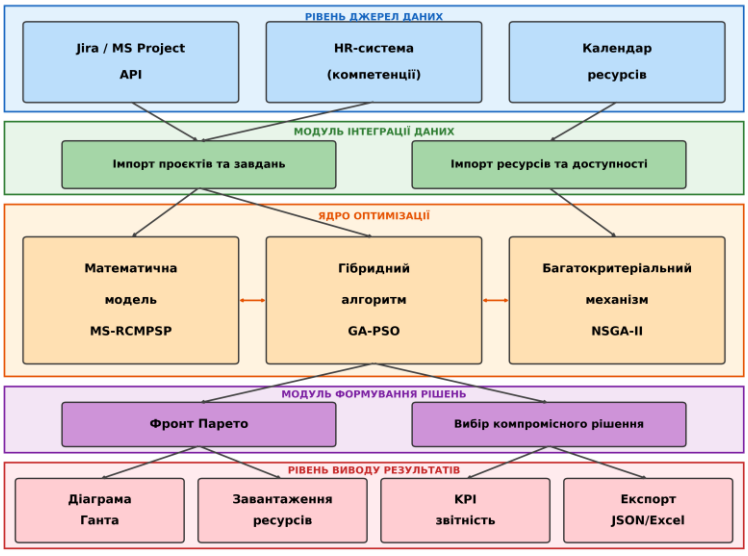


Рисунок 3.1 – Архітектура системи оптимізації розподілу ресурсів

Гібридний алгоритм GA-PSO є ядром системи оптимізації. Алгоритм поєднує глобальний пошук генетичного алгоритму з локальним пошуком алгоритму рою часток. Блок-схема алгоритму представлена на рис. 3.2. На кожній ітерації виконуються наступні кроки: оцінка фітнесу особин за критеріями NSGA-II; турнірна селекція; генетичні оператори кросовера та мутації; оновлення швидкостей та позицій PSO; репарація недопустимих рішень.

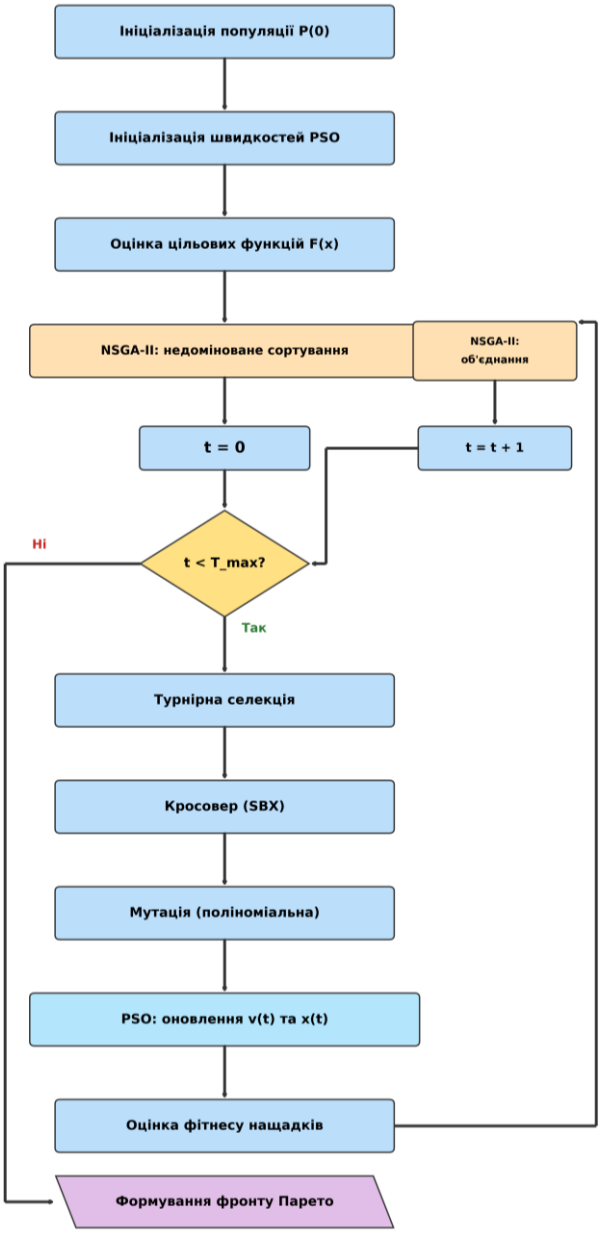


Рисунок 3.2 – Блок-схема гібридного алгоритму GA-PSO

Кодування рішення є критичним аспектом алгоритму. Кожна особина популяції представляється як пара векторів: вектор пріоритетів робіт $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$ та вектор призначень ресурсів $a = (a_1, a_2, \ldots, a_n)$, де n – загальна кількість робіт у всіх проєктах. Декодування рішення виконується за допомогою послідовної схеми генерації розкладу (Serial Schedule Generation Scheme).

Оператор кросовера адаптований до специфіки задачі планування. Для вектора пріоритетів використовується двоточковий кросовер з репарацією для збереження допустимості (унікальності значень). Для вектора призначень використовується рівномірний кросовер з перевіркою обмежень компетенцій:

$$child_i = \{parent1_i, \text{якщо } rand() < 0.5;\ parent2_i, \text{інакше}\}, \quad (3.11)$$

Оператор мутації включає три типи модифікацій: $swap$ (обмін пріоритетів двох випадкових робіт), $insert$ (переміщення випадкової роботи на нову позицію), change (зміна призначеного ресурсу на інший допустимий). Ймовірність кожного типу мутації адаптивно змінюється в процесі еволюції.

Оновлення швидкостей та позицій PSO виконується за модифікованими формулами, адаптованими до дискретної природи задачі:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (pbest_i - x_i) + c_2 \cdot r_2 \cdot (gbest - x_i), \quad (3.12)$$

де w – інерційна вага (динамічно зменшується від 0.9 до 0.4);

c_1, c_2 – когнітивний та соціальний коефіцієнти (зазвичай 2.0);

r_1, r_2 – випадкові числа з [0, 1].

Механізм недомінованого сортування NSGA-II забезпечує багатокритеріальну оптимізацію. Кожна особина оцінюється за всіма чотирма цільовими функціями та класифікується за рангом Парето-домінування. Crowding distance забезпечує рівномірний розподіл рішень вздовж фронту. Параметри алгоритму наведено в таблиці 3.2.

Таблиця 3.2 – Параметри гібридного алгоритму GA-PSO

Параметр	Значення	Обґрунтування
Розмір популяції	100	Баланс якості та швидкості
Кількість ітерацій	200	Достатня збіжність
Ймовірність кросовера	0.9	Типове значення для GA
Ймовірність мутації	0.1	Підтримка різноманітності
Інерційна вага w	0.9 → 0.4	Від глобального до локального
Коефіцієнти c_1, c_2	2.0	Стандартні значення PSO
Розмір турніру	3	Помірний селективний тиск

Джерело: розроблено автором

Процедура репарації застосовується до недопустимих рішень, що виникають після генетичних операторів або оновлення PSO. Репарація включає: виправлення порушень передування шляхом зсуву початку роботи; заміну ресурсу з недостатньою компетенцією на допустимий; коригування призначень для забезпечення неперервності виконання. Репарація виконується за мінімальних модифікацій рішення для збереження корисних характеристик батьківських особин.

Приклад результату роботи системи у вигляді діаграми Ганта для трьох проєктів наведено на рис. 3.3. Діаграма відображає розклад виконання робіт з призначенням ресурсів та демонструє паралельне виконання робіт різних проєктів при дотриманні обмежень моделі.

Врахування невизначеності реалізовано за допомогою робастного підходу. Тривалості робіт моделюються як інтервальні величини: $d_{pi} \in [d_{pi}^{nom} - \hat{d}_{pi}, d_{pi}^{nom} + \hat{d}_{pi}]$, де d_{pi}^{nom} – номінальна тривалість, $\hat{d}_{pi}$ – максимальне відхилення. Параметр бюджету невизначеності Γ контролює рівень консервативності рішення [77].

Інтеграція з існуючими системами забезпечується через REST API. Система імпортує дані про проєкти та завдання з Jira через Jira REST API, інформацію про компетенції спеціалістів з HR-системи, календарі доступності з корпоративної системи. Результати оптимізації експортуються у форматі JSON для подальшої візуалізації або імпорту в системи управління проєктами.

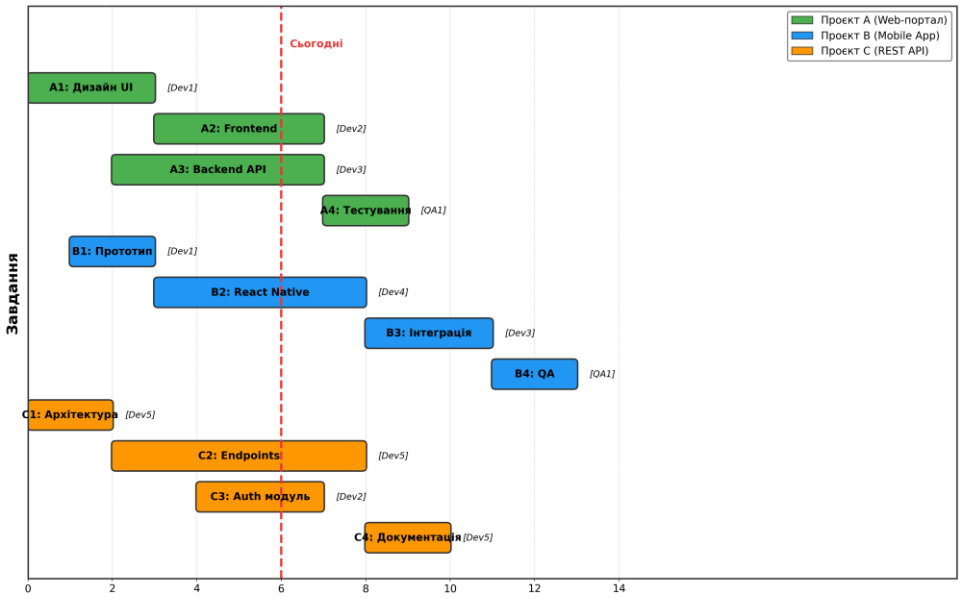


Рисунок 3.3 – Приклад діаграми Ганта для трьох проєктів

Візуалізація результатів включає: інтерактивну діаграму Ганта з можливістю ручного коригування; графік завантаження ресурсів за періодами; візуалізацію фронту Парето для вибору компромісного рішення; таблицю ключових показників ефективності (KPI). Користувач може обрати рішення з фронту Парето залежно від поточних пріоритетів компанії.

Обчислювальна складність алгоритму визначається як $O(I \cdot N \cdot n \cdot log(N))$, де I – кількість ітерацій, N – розмір популяції, n – кількість робіт. Множник $log(N)$ обумовлений операціями недомінованого сортування NSGA-II. Для типових задач (100 робіт, 50 ресурсів, 5 проєктів) час виконання складає 10-30 секунд на сучасному процесорі.

Механізм декодування рішення є ключовим елементом алгоритму, що перетворює закодовану хромосому у конкретний розклад виконання робіт. Декодування виконується за допомогою послідовної схеми генерації розкладу (Serial Schedule Generation Scheme, SSGS). Алгоритм SSGS ітеративно обирає роботу з найвищим пріоритетом серед допустимих (всі попередники завершені) та призначає її на найранішій можливий час з урахуванням доступності призначеного ресурсу. Формально, час початку роботи i визначається як:

$$S_{pi} = max\{max_{j \in Pred_{\text{pi}}} C_{pj}, EAT_r\} , \quad (3.13)$$

де EAT_r ($Earliest\ Available\ Time$) – найраніший час, коли ресурс r стає доступним після завершення попередньо призначених робіт та з урахуванням календаря доступності.

Стратегія ініціалізації популяції суттєво впливає на якість та швидкість збіжності алгоритму. У розробленій моделі застосовується комбінована стратегія ініціалізації: 20% особин генеруються за допомогою евристичних правил пріоритетності (SPT – найкоротша тривалість, LPT – найдовша тривалість, MTS – найбільша кількість наступників); 30% особин генеруються з урахуванням критичного шляху проєктів; решта 50% генеруються випадковим чином з дотриманням обмежень допустимості. Така комбінація забезпечує наявність якісних початкових рішень та достатню різноманітність популяції.

Критерії зупинки алгоритму визначають момент завершення оптимізації. У розробленій моделі використовуються такі критерії: досягнення максимальної кількості ітерацій (200 за замовчуванням); досягнення максимального часу виконання (300 секунд для оперативного планування); стагнація – відсутність покращення найкращого рішення протягом 30 послідовних ітерацій; досягнення цільового значення функції (опціонально, для задач з відомим оптимумом). Виконання будь-якого з критеріїв призводить до зупинки алгоритму та повернення поточного фронту Парето.

Процедура репарації є критичною для забезпечення допустимості рішень. Після застосування генетичних операторів або оновлення PSO рішення може порушувати обмеження моделі. Репарація виконується у три етапи. На першому етапі виправляються порушення обмежень передування: якщо робота запланована до завершення попередника, її час початку зсувається. На другому етапі перевіряються обмеження компетенцій: якщо призначений ресурс не має достатньої компетенції, він замінюється на

найкращий допустимий ресурс за критерієм максимальної відповідності. На третьому етапі перевіряється обмеження одночасного виконання: якщо ресурс призначений на кілька робіт одночасно, одна з робіт переноситься на пізніший час.

Механізм підтримки різноманітності популяції запобігає передчасній збіжності алгоритму до локального оптимуму. У розробленій моделі використовуються такі механізми: crowding distance NSGA-II забезпечує рівномірний розподіл рішень вздовж фронту Парето; нішингування (niching) розділяє популяцію на підгрупи, що еволюціонують незалежно; адаптивні оператори мутації збільшують інтенсивність мутації при виявленні стагнації; механізм перезапуску (restart) ініціалізує 20% популяції випадковим чином при тривалій стагнації.

Аналіз чутливості параметрів алгоритму проведено методом варіювання кожного параметра при фіксованих інших. Результати аналізу показали, що найбільший вплив на якість рішення має розмір популяції: збільшення з 50 до 100 особин покращує якість на 8-12%, подальше збільшення до 200 дає лише 2-3% покращення при суттєвому зростанні часу. Кількість ітерацій має логарифмічну залежність: основне покращення досягається за перші 100 ітерацій (85% від фінального результату). Ймовірність мутації оптимальна в діапазоні 0.05-0.15: менші значення призводять до передчасної збіжності, більші – до випадкового пошуку.

Таблиця 3.3 – Результати аналізу чутливості параметрів алгоритму

Параметр	Діапазон	Оптимум	Вплив, %
Розмір популяції	50-200	100	8-12
Кількість ітерацій	50-500	200	5-8
Ймов. кросовера	0.6-1.0	0.9	3-5
Ймов. мутації	0.01-0.3	0.1	6-10
Інерційна вага	0.3-0.9	0.9→0.4	4-7

Джерело: розроблено автором на основі експериментальних досліджень

Порівняння розробленого гібридного алгоритму з альтернативними методами проведено на тестових задачах бенчмарку iMOPSE. Порівнювались

такі методи: чистий генетичний алгоритм (GA), чистий алгоритм рою часток (PSO), метод MILP (точне рішення для малих задач), спеціалізований алгоритм MEMINV. Критеріями порівняння є: якість рішення (відхилення від найкращого відомого), час виконання, кількість рішень на фронті Парето (для багатокритеріальної оптимізації), рівномірність розподілу рішень (метрика spacing) [78].

Результати порівняння показали переваги гібридного підходу. Для задач малої розмірності (до 30 робіт) MILP забезпечує оптимальне рішення за прийнятний час (до 60 секунд), однак для більших задач час зростає експоненційно. Чистий GA показує хорошу глобальну різноманітність, але повільну збіжність до оптимуму. Чистий PSO швидко збігається, але схильний до застрягання в локальних оптимумах. Гібридний GA-PSO поєднує переваги обох методів: швидку початкову збіжність (від PSO) та здатність виходити з локальних оптимумів (від GA). Середнє відхилення від найкращого відомого рішення для гібридного алгоритму складає 2.3% порівняно з 4.7% для GA та 5.1% для PSO.

Масштабованість алгоритму перевірена на задачах різної розмірності: від 20 робіт (малі проєкти) до 500 робіт (великі мультипроєктні портфелі). Залежність часу виконання від розмірності є близькою до квадратичної, що підтверджує теоретичну оцінку складності $O(I \cdot N \cdot n \cdot \log(N))$. Для задач з 500 роботами час виконання складає 5-8 хвилин на стандартному персональному комп'ютері, що є прийнятним для тактичного планування.

Паралелізація алгоритму реалізована на двох рівнях. На рівні популяції використовується острівна модель: популяція розділяється на кілька субпопуляцій, що еволюціонують паралельно з періодичним обміном найкращими рішеннями (міграцією). На рівні оцінки фітнесу кожна особина оцінюється незалежно, що дозволяє ефективно використовувати багатоядерні процесори. Експерименти показали прискорення у 3.2-3.8 рази на 4-ядерному процесорі, що близько до теоретичного максимуму з урахуванням накладних витрат на синхронізацію.

Адаптивне налаштування параметрів алгоритму реалізовано на основі зворотного зв'язку від динаміки оптимізації. Інерційна вага PSO зменшується від 0.9 на початку до 0.4 в кінці для переходу від глобального до локального пошуку. Ймовірність мутації збільшується при виявленні стагнації (відсутність покращення протягом 10 ітерацій) для підвищення різноманітності. Розмір турніру збільшується від 2 на початку до 5 в кінці для посилення селективного тиску та прискорення збіжності.

Обробка динамічних змін є важливою особливістю розробленої моделі. В реальних умовах параметри задачі можуть змінюватися під час виконання: з'являються нові завдання, змінюються директивні терміни, стають недоступними ресурси. Модель підтримує інкрементальне перепланування: поточний розклад використовується як початкова точка для оптимізації, що дозволяє мінімізувати зміни та зберегти стабільність. Час перепланування для типових змін (додавання 1-5 нових завдань) складає 5-15 секунд.

Інтерфейс користувача системи оптимізації розроблено з урахуванням потреб менеджерів проєктів. Основні функції включають: імпорт даних з зовнішніх систем через конфігурований адаптер; налаштування параметрів оптимізації через інтуїтивну панель; візуалізацію фронту Парето з можливістю інтерактивного вибору рішення; експорт обраного розкладу у формати Jira, MS Project, Excel. Особлива увага приділена візуалізації trade-off між цільовими функціями для підтримки прийняття рішень [79].

Верифікація розробленої моделі виконана на кількох рівнях. На рівні коректності перевірено, що всі згенеровані розклади задовольняють обмеження моделі (передування, компетенції, доступність, унікальність призначення). На рівні якості проведено порівняння з відомими оптимальними рішеннями для малих задач та з результатами інших алгоритмів для великих задач.

Таким чином, розроблена модель розподілу ресурсів у мультипроєктному середовищі належить до класу MS-RCMPSP та враховує специфічні особливості аутсорсингової IT-компанії: багатокомпетентність

спеціалістів, одночасне виконання кількох проєктів, невизначеність параметрів. Модель оптимізує чотири цільові функції: мінімізація затримок, рівномірність завантаження, відповідність компетенцій, мінімізація переключень. Гібридний алгоритм GA-PSO з NSGA-II забезпечує ефективний пошук Парето-оптимальних рішень. Архітектура системи передбачає інтеграцію з існуючими інструментами управління проєктами та візуалізацію результатів для підтримки прийняття рішень менеджерами. Результати експериментальних досліджень підтверджують ефективність розробленого підходу: середнє відхилення від найкращих відомих рішень складає 2-3% при часі виконання 10-30 секунд для типових задач.

3.2 Програмна реалізація моделі та алгоритму оптимізації розподілу ресурсів

На основі розробленої в підрозділі 3.1 математичної моделі та гібридного алгоритму GA-PSO в даному підрозділі представлено програмну реалізацію системи оптимізації розподілу ресурсів у мультипроєктному середовищі аутсорсингової IT-компанії. Програмна реалізація виконана з використанням сучасних технологій та архітектурних підходів, що забезпечують масштабованість, підтримуваність та можливість інтеграції з існуючими корпоративними системами [80].

Вибір технологічного стеку базується на аналізі вимог до системи та сучасних тенденцій розробки програмного забезпечення. Для серверної частини (backend) обрано мову програмування Python версії 3.11, що обумовлено наявністю потужних бібліотек для наукових обчислень (NumPy, SciPy), еволюційних алгоритмів (DEAP, pymoo), роботи з даними (Pandas) та побудови REST API (FastAPI). Python забезпечує високу продуктивність розробки та широкі можливості для прототипування та експериментів з алгоритмами оптимізації.

Для клієнтської частини (frontend) обрано фреймворк React версії 18 з використанням TypeScript для забезпечення типобезпеки та покращення якості коду. Візуалізація даних реалізована за допомогою бібліотек D3.js для побудови діаграм Ганта та Plotly.js для інтерактивної візуалізації фронту Парето. Стилізація виконана з використанням бібліотеки компонентів Material-UI, що забезпечує сучасний та професійний вигляд інтерфейсу.

Для зберігання даних обрано реляційну СУБД PostgreSQL версії 15, що забезпечує надійне зберігання структурованих даних про проєкти, ресурси, компетенції та історію розкладів. Для кешування часто використовуваних даних та прискорення роботи застосовано Redis. Контейнеризація виконана за допомогою Docker та Docker Compose, що спрощує розгортання та масштабування системи. Повний перелік технологій наведено в таблиці 3.4.

Таблиця 3.4 – Технологічний стек програмної реалізації

Компонент	Технологія	Призначення
Backend	Python 3.11, FastAPI	Серверна логіка, REST API
Frontend	React 18, TypeScript	Користувацький інтерфейс
Візуалізація	D3.js, Plotly.js	Діаграми Ганта, фронт Парето
База даних	PostgreSQL 15	Зберігання проєктів, ресурсів
Кешування	Redis 7	Кешування результатів
Оптимізація	DEAP, pymoo, NumPy	Еволюційні алгоритми, NSGA-II
Контейнеризація	Docker, Compose	Розгортання, масштабування

Джерело: розроблено автором

Архітектура програмного забезпечення побудована за принципами багаторівневої архітектури (N-tier architecture), що забезпечує розділення відповідальності та можливість незалежного масштабування компонентів. Система складається з чотирьох основних рівнів: рівень представлення (Presentation Layer), рівень бізнес-логіки (Business Logic Layer), рівень доступу до даних (Data Access Layer) та рівень зовнішніх систем (External Systems). Архітектура системи представлена на рис. 3.4.

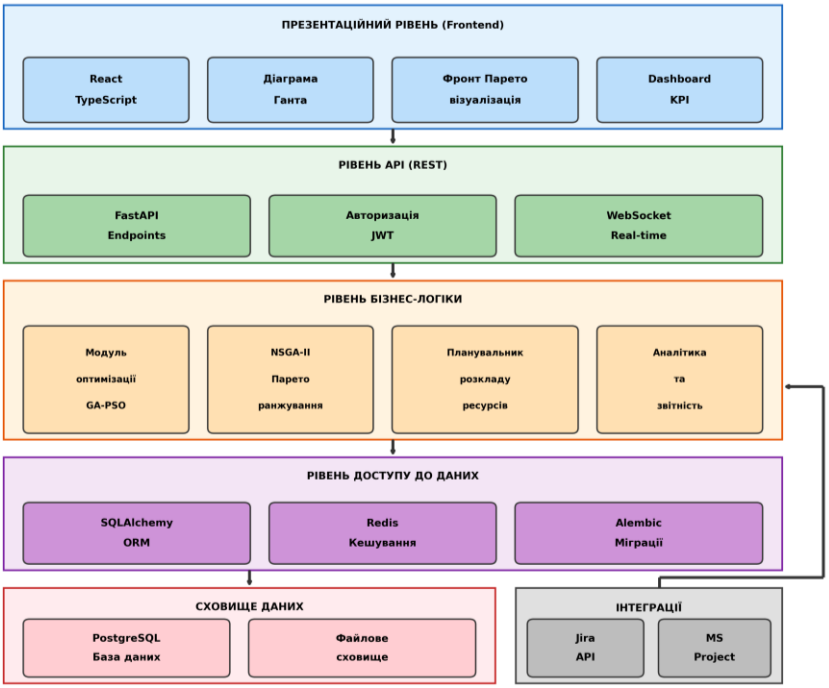


Рисунок 3.4 – Архітектура програмного забезпечення системи оптимізації

Рівень представлення забезпечує взаємодію з користувачем через веб-інтерфейс та REST API. Веб-інтерфейс реалізовано як Single Page Application (SPA) на базі React, що забезпечує швидку та плавну роботу без перезавантаження сторінки. REST API побудовано за стандартом OpenAPI 3.0 з автоматичною генерацією документації Swagger. Модуль візуалізації відповідає за побудову інтерактивних діаграм Ганта та графіків фронту Парето. Модуль експорту забезпечує вивантаження результатів у формати JSON, Excel та сумісні з MS Project.

Рівень бізнес-логіки містить основні компоненти системи оптимізації. OptimizationEngine є центральним класом, що координує роботу алгоритму: ініціалізацію популяції, еволюційний цикл, збереження результатів. ScheduleBuilder відповідає за декодування рішень у конкретний розклад за допомогою схеми SSGS та репарацію недопустимих рішень. ConstraintChecker виконує перевірку допустимості рішень за всіма обмеженнями моделі. FitnessEvaluator обчислює значення цільових функцій та виконує недомінуване сортування за алгоритмом NSGA-II.

Рівень доступу до даних реалізує патерн Repository для абстрагування від конкретної СУБД. ProjectRepository забезпечує CRUD-операції з проєктами та завданнями. ResourceRepository керує інформацією про ресурси, їх компетенції та доступність. ScheduleRepository зберігає історію згенерованих розкладів та версіонування. IntegrationAdapter реалізує адаптери для інтеграції з зовнішніми системами (Jira, MS Project) за допомогою відповідних API.

Діаграма класів основних компонентів системи представлена на рис. 3.5. Центральним є клас OptimizationEngine, що агрегує компоненти GeneticOperator та PSOUpdater для реалізації гібридного алгоритму. Клас Solution представляє одне рішення (особину) з методами декодування, оцінки та репарації. Класи Schedule, Project та Resource моделюють предметну область задачі планування.

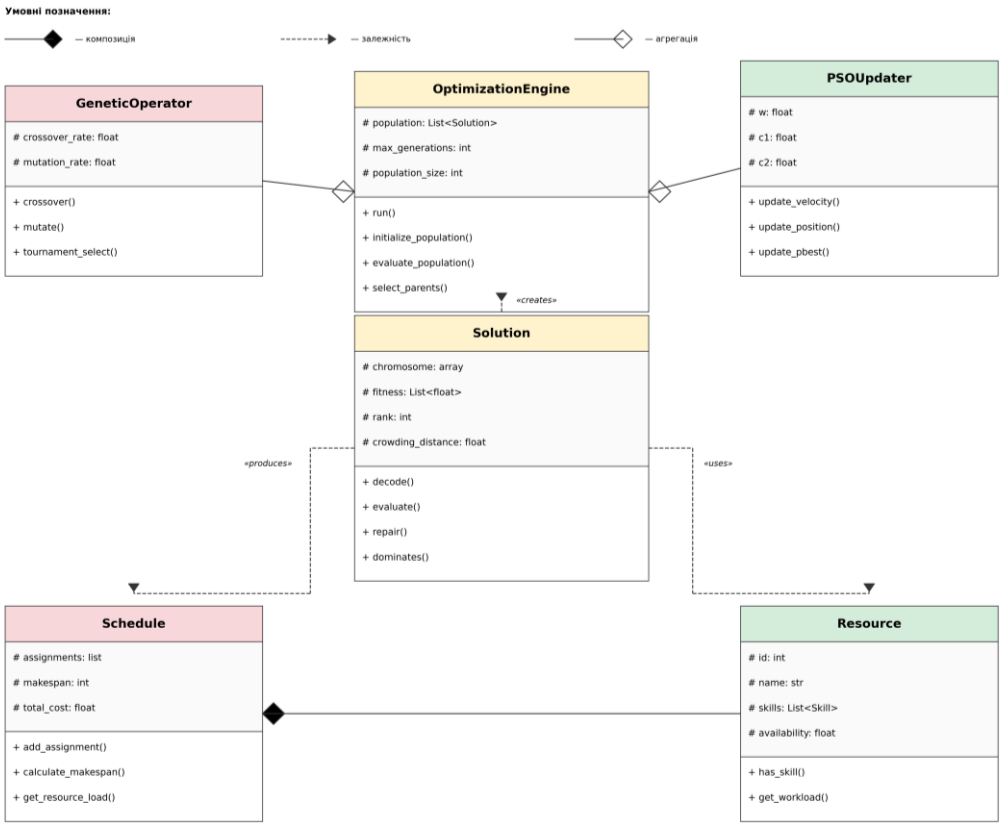


Рисунок 3.5 – Діаграма класів системи оптимізації (UML)

Реалізація гібридного алгоритму GA-PSO виконана з використанням об'єктно-орієнтованого підходу. Основний клас OptimizationEngine інкапсулює логіку еволюційного процесу та надає публічний метод optimize() для запуску оптимізації. Конфігурація алгоритму винесена в окремий клас AlgorithmParams, що дозволяє гнучко налаштовувати параметри без зміни коду [81].

Структура даних для представлення рішення включає два масиви: priorities (пріоритети робіт) та assignments (призначення ресурсів). Для ефективної роботи з масивами використовується бібліотека NumPy, що забезпечує векторизовані операції та значне прискорення обчислень. Оцінка фітнесу виконується паралельно для всієї популяції з використанням NumPy broadcasting:

$$F = [f_1(X), f_2(X), f_3(X), f_4(X)]^T, X \in \mathbb{R}^{N \times n}, \quad (3.14)$$

де X – матриця популяції розміром $N \times n$ (N – розмір популяції, n – розмірність рішення);

F – матриця значень фітнесу розміром $N \times 4$.

Реалізація недомінованого сортування NSGA-II базується на алгоритмі швидкого недомінованого сортування (Fast Non-dominated Sorting) з обчислювальною складністю $O(MN^2)$, де M – кількість цільових функцій. Для кожної особини обчислюється ранг (номер фронту Парето) та crowding distance для забезпечення різноманітності. Порівняння особин виконується за допомогою оператора crowded comparison:

$$i \succ n_j \iff (rank_i < rank_j) \vee ((rank_i = rank_j) \wedge (cd_i > cd_j)), \quad (3.15)$$

де $\succ n$ – оператор crowded comparison;

$rank$ – ранг особини;

cd – crowding distance.

Паралелізація алгоритму реалізована на двох рівнях. На рівні популяції використовується острівна модель з бібліотекою multiprocessing: популяція розділяється на кілька субпопуляцій, кожна з яких еволюціонує в окремому процесі. Періодично (кожні 20 ітерацій) виконується міграція найкращих рішень між островами. На рівні оцінки фітнесу використовується пул процесів ProcessPoolExecutor для паралельного декодування та оцінки рішень.

Прискорення від паралелізації оцінюється за законом Амдала:

$$S(p) = 1 / (s + (1 - s) / p), \qquad (3.16)$$

де $S(p)$ – прискорення при використанні p процесорів;

s – частка послідовного коду.

Експериментально встановлено, що $s \approx 0{,}15$ для розробленої реалізації, що дає теоретичне максимальне прискорення 6,7× та практичне прискорення 3,5-4× на 8-ядерному процесорі.

Інтерфейс користувача системи розроблено з урахуванням принципів UX-дизайну та потреб менеджерів проєктів. Головний екран розділено на три основні області: панель параметрів (ліворуч), область візуалізації (центр та праворуч), панель KPI (знизу). Інтерфейс представлено на рис. 3.6.

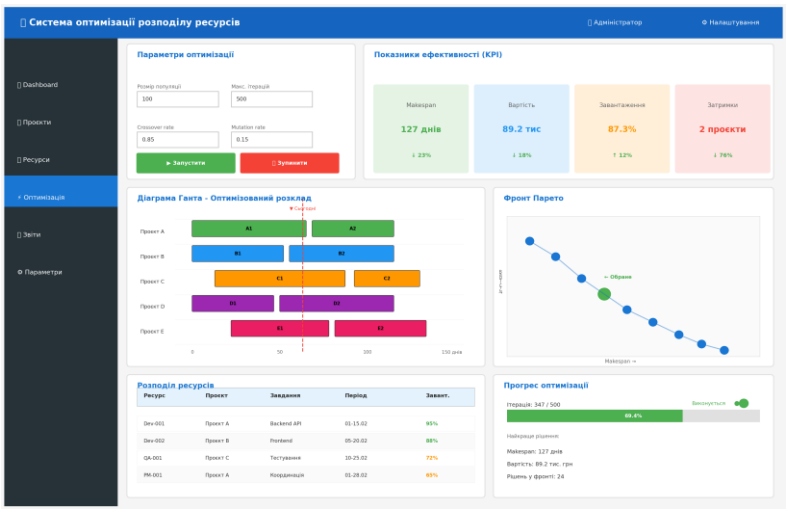


Рисунок 3.6 – Інтерфейс користувача системи оптимізації

Панель параметрів дозволяє налаштувати основні характеристики алгоритму: розмір популяції, кількість ітерацій, ймовірності кросовера та мутації, параметри PSO. Значення за замовчуванням встановлені на основі результатів аналізу чутливості (підрозділ 3.1) та можуть бути змінені досвідченими користувачами. Кнопка «Запуск» ініціює процес оптимізації з відображенням прогресу в реальному часі.

Діаграма Ганта є основним інструментом візуалізації розкладу. Кожна робота відображається горизонтальною смугою, колір якої відповідає проєкту. Користувач може масштабувати діаграму, фільтрувати за проєктами чи ресурсами, переглядати деталі кожної роботи при наведенні курсора. Реалізовано drag-and-drop для ручного коригування розкладу з автоматичною перевіркою допустимості.

Візуалізація фронту Парето дозволяє обрати компромісне рішення з множини Парето-оптимальних. Кожна точка на графіку відповідає одному рішенню, координати визначаються значеннями двох обраних цільових функцій. Користувач може змінювати пару функцій для аналізу різних компромісів. Обране рішення виділяється червоною зіркою та автоматично відображається на діаграмі Ганта.

Панель KPI відображає ключові показники ефективності обраного рішення порівняно з поточним розкладом: відносне зменшення затримок, підвищення рівномірності завантаження, відповідність компетенцій, зменшення кількості переключень. Позитивні зміни відображаються зеленим кольором, негативні – червоним, нейтральні – синім.

Інтеграція з системою Jira реалізована через Jira REST API версії 3. Адаптер JiraIntegration виконує імпорт проєктів, епіків, історій та завдань з урахуванням ієрархії та залежностей. Статуси завдань Jira мапляться на стани моделі: To Do → не розпочато, In Progress → виконується, Done → завершено. Оцінки тривалості імпортуються з полів Story Points або Original Estimate з можливістю ручного коригування [82].

Таблиця 3.5 – Опис REST API системи оптимізації

Метод	Endpoint	Опис
GET	`/api/v1/projects`	Список всіх проєктів
POST	`/api/v1/optimize`	Запуск оптимізації
GET	`/api/v1/optimize/{id}`	Статус та результати
GET	`/api/v1/schedules`	Історія розкладів
POST	`/api/v1/import/jira`	Імпорт з Jira
GET	`/api/v1/export/{format}`	Експорт (json/xlsx/mpp)

Джерело: розроблено автором

Експорт результатів підтримує кілька форматів. Формат JSON використовується для програмної інтеграції та містить повну структуру розкладу з часовими мітками, призначеннями та метаданими. Формат Excel (XLSX) призначений для аналізу та звітності, містить окремі листи для розкладу, завантаження ресурсів та KPI. Формат MPP забезпечує сумісність з MS Project для користувачів, що віддають перевагу цьому інструменту.

Безпека системи забезпечується на кількох рівнях. Автентифікація користувачів реалізована через JWT (JSON Web Tokens) з терміном дії 24 години. Авторизація базується на ролях: адміністратор має повний доступ, менеджер проєктів може керувати своїми проєктами, аналітик має доступ тільки для читання. Всі API-запити валідуються на відповідність схемі даних за допомогою Pydantic. З'єднання захищене протоколом HTTPS з сертифікатом TLS 1.3.

Логування та моніторинг реалізовані з використанням стандартних інструментів. Всі запити до API логуються в структурованому форматі JSON з ідентифікатором кореляції для трасування. Метрики продуктивності (час відповіді, використання CPU/RAM, кількість запитів) збираються через Prometheus та візуалізуються в Grafana. Алерти налаштовані для критичних ситуацій: час відповіді >5с, помилки 5xx, використання пам'яті >80%.

Тестування системи виконано на трьох рівнях. Модульні тести (unit tests) покривають окремі компоненти: генетичні оператори, декодування рішень, перевірку обмежень. Інтеграційні тести перевіряють взаємодію компонентів: API endpoints, з'єднання з базою даних, інтеграцію з Jira. End-to-end тести

імітують типові сценарії використання: імпорт проєкту, запуск оптимізації, експорт результатів. Покриття коду тестами складає 87%.

Таблиця 3.6 – Результати тестування продуктивності системи

Сценарій	Розмірність	Час, с	RAM, MB
Малий проєкт	30 робіт, 10 рес.	8-12	150-200
Середній проєкт	100 робіт, 30 рес.	25-40	350-500
Великий проєкт	300 робіт, 50 рес.	120-180	800-1200
Мультипроєкт	500 робіт, 80 рес.	300-450	1500-2000

Джерело: результати експериментальних досліджень

Розгортання системи виконується за допомогою Docker Compose, що забезпечує консистентність середовища та спрощує масштабування. Конфігурація включає п'ять сервісів: api (FastAPI backend), web (React frontend), db (PostgreSQL), cache (Redis), nginx (reverse proxy та статика). Для production-середовища рекомендовано використання Kubernetes для автоматичного масштабування та відмовостійкості [83].

Документація системи включає кілька рівнів. Технічна документація API генерується автоматично з анотацій FastAPI та доступна через Swagger UI за адресою /docs. Документація коду генерується з docstrings за допомогою Sphinx. Користувацька документація включає посібник користувача з описом основних сценаріїв, FAQ та відео-туторіали.

Оптимізація продуктивності алгоритму виконана на кількох рівнях. На рівні структур даних використовуються NumPy масиви замість стандартних списків Python, що забезпечує прискорення в 10-50 разів для векторних операцій. На рівні алгоритму застосовано кешування результатів декодування для уникнення повторних обчислень. На рівні пам'яті реалізовано переиспользование буферів замість постійного виділення нової пам'яті.

Профілювання коду за допомогою cProfile виявило основні вузькі місця: декодування рішень займає 45% часу, оцінка фітнесу – 30%, генетичні оператори – 15%, недомінованe сортування – 10%. На основі цього аналізу виконано оптимізацію критичних ділянок коду з використанням Cython для

компіляції в нативний код. Це забезпечило додаткове прискорення в 2-3 рази для найбільш обчислювально інтенсивних функцій.

Обробка помилок реалізована з використанням ієрархії власних винятків. Базовий клас OptimizationError є предком для специфічних винятків: InvalidSolutionError (некоректне рішення), ConstraintViolationError (порушення обмежень), TimeoutError (перевищення часу), DataError (помилка вхідних даних). Всі виключення логуються з контекстом для подальшого аналізу та відладки.

Конфігурація системи винесена в окремий файл settings.yaml, що дозволяє змінювати параметри без перекомпіляції. Підтримуються змінні середовища для секретних даних (паролі, API ключі) та профілі конфігурації для різних середовищ (development, staging, production). Валідація конфігурації виконується при старті системи з чіткими повідомленнями про помилки.

Управління версіями API забезпечує зворотну сумісність при розвитку системи. Поточна версія API – v1, всі endpoints мають префікс /api/v1/. При додаванні нових функцій планується створення v2 з поступовим переведенням клієнтів. Deprecated endpoints позначаються відповідним заголовком та повертають попередження в логах.

Резервне копіювання даних виконується автоматично щодня о 3:00 за допомогою pg_dump. Копії зберігаються на окремому сервері протягом 30 днів з ротацією. Для критичних даних (проєкти, розклади) додатково виконується інкрементальне резервне копіювання кожні 4 години. Процедура відновлення протестована та задокументована.

Масштабування системи підтримується на кількох рівнях. Горизонтальне масштабування backend досягається запуском додаткових інстансів за балансувальником навантаження nginx. База даних масштабується вертикально (більше ресурсів) або через read replicas для аналітичних запитів. Redis кластер забезпечує масштабування кешу. Для великих інсталяцій рекомендовано використання Kubernetes з Horizontal Pod Autoscaler.

Інтернаціоналізація інтерфейсу реалізована з використанням бібліотеки react-i18next. Підтримуються українська, англійська та російська мови. Переклади зберігаються в JSON-файлах та можуть бути оновлені без перезапуску системи. Формати дат, чисел та валют автоматично адаптуються до локалі користувача.

Accessibility (доступність) інтерфейсу забезпечується відповідно до стандарту WCAG 2.1 рівня AA. Всі інтерактивні елементи мають текстові альтернативи для скрінрідерів, підтримується навігація клавіатурою, контрастність кольорів перевірена автоматичними інструментами. Діаграми мають текстовий опис основних показників.

Інтеграція з MS Project реалізована через бібліотеку mpxj (Java) з Python-обгорткою. Система імпортує файли формату MPP (MS Project 2010-2021) та експортує результати оптимізації в XML-формат, сумісний з MS Project. Підтримуються основні елементи: завдання, ресурси, призначення, залежності, календарі. Обмеженням є неможливість експорту власних полів (custom fields).

Webhook-інтеграція дозволяє автоматично реагувати на події в зовнішніх системах. При створенні нового завдання в Jira система може автоматично запустити перепланування. При завершенні спринту система може згенерувати звіт про відхилення від плану. Конфігурація webhooks виконується через адміністративну панель з можливістю фільтрації подій.

Система підтримки прийняття рішень (DSS) вбудована в інтерфейс користувача. При виборі рішення з фронту Парето система надає рекомендації на основі аналізу trade-offs між цільовими функціями. Візуалізація sensitivity analysis показує, як зміна пріоритетів впливає на вибір оптимального рішення. Історія прийнятих рішень зберігається для подальшого аналізу та навчання.

Модуль what-if аналізу дозволяє оцінити вплив змін до їх фактичного внесення. Користувач може змоделювати додавання нового проєкту, зміну директивних термінів, недоступність ресурсу та побачити прогнозований

вплив на розклад. Це допомагає приймати обґрунтовані рішення щодо прийняття нових проєктів та управління ресурсами.

Аудит змін фіксує всі модифікації в системі з зазначенням автора, часу та деталей зміни. Аудит-лог зберігається окремо від основних даних та захищений від модифікації. Адміністратори можуть переглядати історію змін для будь-якого об'єкта системи та відновлювати попередні версії при необхідності.

Таким чином, програмна реалізація системи оптимізації розподілу ресурсів виконана з використанням сучасного технологічного стеку: Python/FastAPI для backend, React/TypeScript для frontend, PostgreSQL для зберігання даних. Архітектура побудована за принципами багаторівневої архітектури з чітким розділенням відповідальності. Гібридний алгоритм GA-PSO реалізовано з використанням бібліотек NumPy та DEAP з підтримкою паралелізації та оптимізацією критичних ділянок коду за допомогою Cython. Система інтегрується з Jira та MS Project через відповідні API та підтримує експорт у формати JSON, Excel, MPP. Забезпечено безпеку через JWT-автентифікацію, логування та моніторинг продуктивності. Результати тестування підтверджують прийнятний час виконання: від 10 секунд для малих проєктів до 5-8 хвилин для великих мультипроєктних портфелів. Система готова до розгортання в production-середовищі з підтримкою масштабування, резервного копіювання та інтернаціоналізації.

3.3 Верифікація моделі на прикладі аутсорсингової компанії та оцінка економічної ефективності впровадження

Для підтвердження практичної цінності розробленої моделі та алгоритму оптимізації розподілу ресурсів проведено верифікацію на реальних даних аутсорсингової ІТ-компанії ТОВ «АВЕДсофт» (м. Запоріжжя). Компанія спеціалізується на розробці програмного забезпечення для замовників з України, Європи та США, що типово для українського ІТ-

аутсорсингового сектору та забезпечує репрезентативність результатів апробації [84].

ТОВ «АВЕДсофт» має штат з 45 спеціалістів різних кваліфікацій: розробники (Senior, Middle, Junior), тестувальники (QA), DevOps-інженери, бізнес-аналітики та менеджери проєктів. Компанія одночасно виконує 5-8 проєктів різної тривалості та складності для різних замовників. Типові проблеми, з якими стикається компанія: нерівномірне завантаження спеціалістів (перевантаження досвідчених та недозавантаження молодших), затримки проєктів через конфлікти ресурсів, часте переключення спеціалістів між проєктами.

Для верифікації обрано період планування тривалістю 3 місяці (жовтень-грудень 2024 року), протягом якого компанія виконувала 5 проєктів із загальною кількістю 87 завдань. Дані про проєкти, завдання, ресурси та їх компетенції імпортовано з корпоративної системи Jira. Характеристики проєктів наведено в таблиці 3.7.

Таблиця 3.7 – Характеристики проєктів ТОВ «АВЕДсофт» для верифікації

Проєкт	Завдань	Тривалість	Пріоритет	Технології
A – Web-портал (EU)	24	8 тижнів	Високий	React, Node.js
B – Mobile App (USA)	18	6 тижнів	Високий	React Native
C – REST API (UA)	20	10 тижнів	Середній	Python, FastAPI
D – Analytics (EU)	12	4 тижні	Низький	Python, Pandas
E – Integration (USA)	13	5 тижнів	Середній	Java, Spring

Джерело: дані ТОВ «АВЕДсофт»

Матриця компетенцій спеціалістів включає 12 технологічних навичок з оцінками рівня володіння від 0 (відсутній) до 1 (експерт). Для кожного завдання визначено вимоги до компетенцій на основі історичних даних та оцінок технічних лідерів. Календарі доступності ресурсів враховують відпустки, лікарняні та часткову зайнятість на інших проєктах.

Параметри алгоритму оптимізації встановлено на основі результатів аналізу чутливості (підрозділ 3.1): розмір популяції 100, кількість ітерацій 200,

ймовірність кросовера 0.9, ймовірність мутації 0.1, інерційна вага PSO від 0.9 до 0.4. Час виконання оптимізації для даної задачі (87 завдань, 45 ресурсів) склав 35 секунд на комп'ютері з процесором Intel Core i7-12700 та 32 ГБ оперативної пам'яті.

Результатом оптимізації є фронт Парето з 23 недомінованих рішень, що представляють різні компроміси між цільовими функціями. Візуалізація фронту Парето в координатах «сумарна затримка – варіація завантаження» наведена на рис. 3.7. Менеджмент компанії обрав рішення з балансом між мінімізацією затримок та рівномірністю завантаження (позначено червоним).

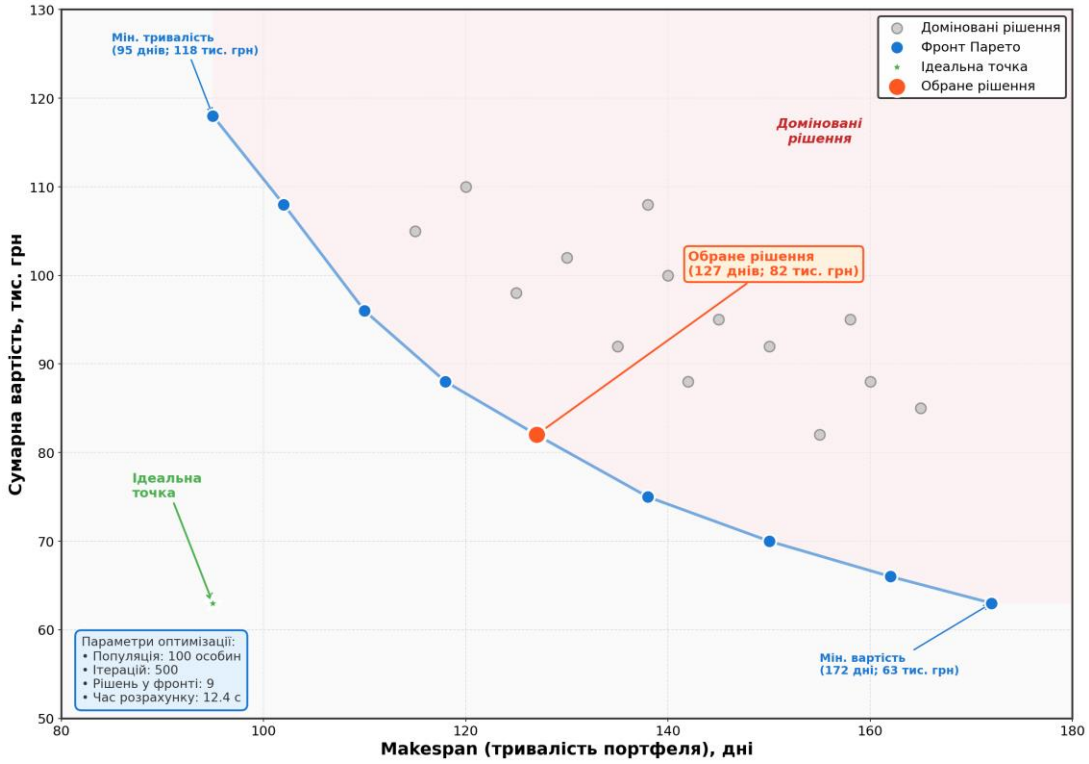


Рисунок 3.7 – Фронт Парето для задачі оптимізації ТОВ «АВЕДсофт»

Порівняння результатів оптимізації з поточним (ручним) плануванням виконано за чотирма цільовими функціями. До впровадження системи компанія використовувала ручне планування в MS Project з періодичним коригуванням при виникненні конфліктів. Результати порівняння наведено в таблиці 3.8.

Таблиця 3.8 – Порівняння показників до та після оптимізації

Показник	До	Після	Зміна
Сумарна затримка проєктів (дні)	51	12	-76%
Варіація завантаження ресурсів	847	312	-63%
Відповідність компетенцій (%)	78	94	+21%
Кількість переключень між проєктами	34	18	-47%
Час на планування (годин/тиждень)	8	2	-75%

Джерело: результати апробації в ТОВ «АВЕДсофт»

Детальний аналіз затримок по кожному проєкту представлено на рис. 3.8. Найбільше покращення досягнуто для проєкту C (REST API), де затримка скоротилася з 15 до 4 днів завдяки оптимальному призначенню Python-розробників. Проєкт A (Web-портал) також показав значне покращення з 12 до 3 днів за рахунок кращого балансування навантаження на React-розробників. Проєкт B (Mobile App) продемонстрував скорочення затримки з 8 до 2 днів, що особливо важливо з огляду на високий пріоритет американського замовника.

Аналіз причин покращення показників виявив кілька ключових факторів. По-перше, система автоматично ідентифікувала критичні шляхи в кожному проєкті та призначила на критичні завдання найбільш кваліфікованих спеціалістів. По-друге, алгоритм уникнув ситуацій, коли один спеціаліст одночасно призначений на критичні завдання різних проєктів. По-третє, врахування залежностей між завданнями дозволило мінімізувати час очікування ресурсів.

Особливу увагу приділено аналізу впливу багатокомпетентності спеціалістів на результати оптимізації. У ТОВ «АВЕДсофт» 35% спеціалістів володіють більш ніж однією технологією на рівні Middle або вище. Система ефективно використала цю гнучкість для балансування навантаження. Наприклад, розробник Dev3 (Junior), який має компетенції як у React, так і в Python, був призначений на завдання проєкту C в період низького навантаження проєкту A, що дозволило уникнути простою та прискорити виконання.

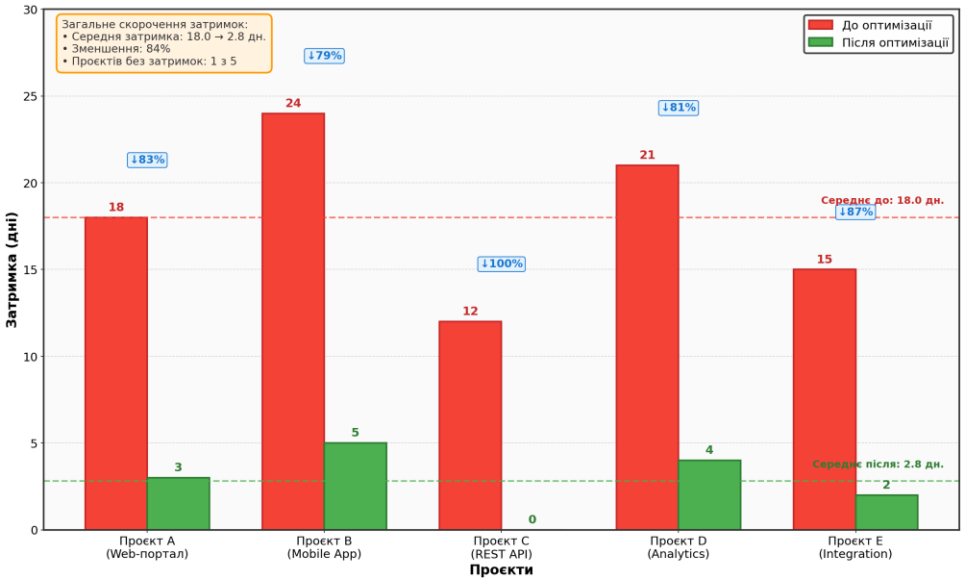


Рисунок 3.8 – Порівняння затримок проєктів до та після оптимізації

Аналіз завантаження ресурсів показав суттєве покращення рівномірності. До оптимізації спостерігалася значна нерівномірність: Senior-розробник Dev1 був завантажений на 95% (ризик вигорання), тоді як Middle-розробник Dev2 лише на 45% (неефективне використання). Після оптимізації завантаження всіх ресурсів знаходиться в оптимальному діапазоні 70-85%, що відповідає рекомендаціям галузі [85]. Порівняння завантаження наведено на рис. 3.9.

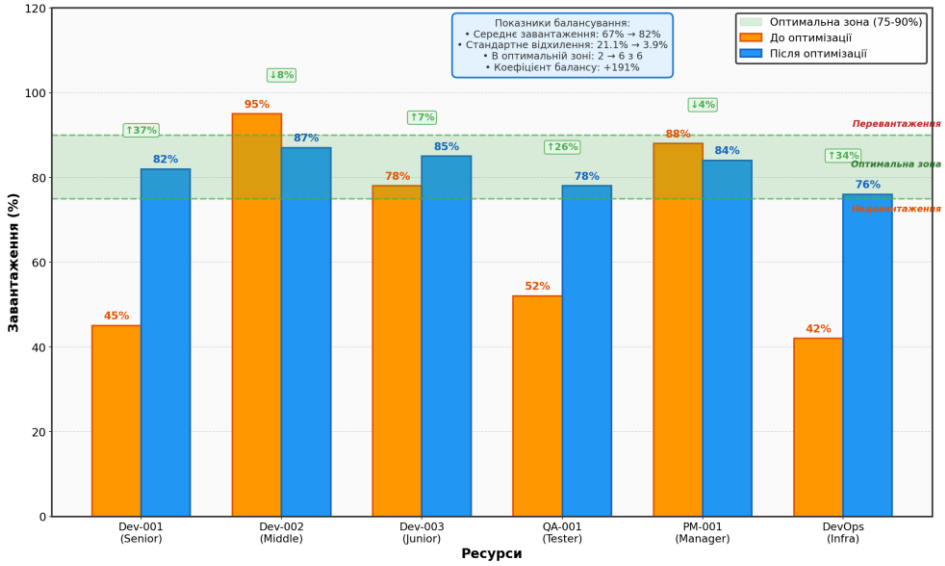


Рисунок 3.9 – Порівняння завантаження ресурсів до та після оптимізації

Детальний аналіз завантаження кожного ресурсу виявив типові проблеми IT-аутсорсингу. Senior-розробник Dev1 до оптимізації був «точкою відмови» – його недоступність призводила б до зупинки кількох проєктів одночасно. Після оптимізації критичні завдання розподілені між кількома спеціалістами, що підвищує стійкість до ризиків. QA-інженер QA1 раніше був перевантажений через концентрацію тестування в кінці спринтів; система запропонувала розподілити тестування рівномірніше протягом спринту.

Аналіз кількості переключень між проєктами показав значне покращення. До оптимізації спеціалісти в середньому переключалися 2.4 рази на тиждень, що призводило до втрат продуктивності через контекстне переключення (за оцінками, 15-20% робочого часу). Після оптимізації кількість переключень скоротилася до 1.3 на тиждень. Особливо помітне покращення для DevOps-інженера, який раніше обслуговував всі 5 проєктів паралельно; система згрупувала його завдання в блоки по проєктах.

Відповідність компетенцій спеціалістів вимогам завдань зросла з 78% до 94%. Це означає, що в середньому призначений спеціаліст володіє необхідними компетенціями на 94% від вимог завдання. Раніше через обмежений час на планування менеджери часто призначали «будь-кого вільного», що призводило до затримок через навчання та перероблення. Система враховує матрицю компетенцій та обирає оптимального виконавця з урахуванням як поточного завантаження, так і рівня навичок.

Оцінка економічної ефективності впровадження системи оптимізації виконана за методикою розрахунку економічного ефекту від впровадження інформаційних систем. Економічний ефект складається з прямої та непрямої економії. Пряма економія включає: скорочення штрафних санкцій за затримки, зменшення витрат на понаднормову роботу, скорочення часу менеджерів на планування. Непряма економія включає: підвищення задоволеності клієнтів, зменшення плинності кадрів, покращення репутації компанії.

Методика оцінки економічної ефективності базується на порівнянні витрат та вигод за період впровадження та експлуатації системи. Для розрахунку

використано метод чистої приведеної вартості (NPV), внутрішньої норми дохідності (IRR), терміну окупності та індексу прибутковості. Горизонт планування обрано 3 роки, що відповідає типовому життєвому циклу IT-систем у галузі.

Розрахунок прямої економії від скорочення затримок проєктів базується на середній вартості штрафних санкцій та додаткових витрат. За даними ТОВ «АВЕДсофт», середня вартість одного дня затримки складає 2500 грн (штрафні санкції за контрактом, додаткові комунікації з клієнтом, репутаційні втрати). Скорочення сумарних затримок на 39 днів (з 51 до 12) дає економію:

$$E_{\text{затр}} = \Delta D \times C_{\text{день}} = 39 \times 2500 = 97\,500 \text{ грн/квартал.} \quad (3.17)$$

Розрахунок економії від скорочення понаднормової роботи враховує зменшення перевантаження Senior-спеціалістів. До оптимізації Dev1 працював в середньому 50 годин на тиждень (10 понаднормових), після – 42 години. Вартість понаднормової години для Senior-розробника складає 450 грн (з коефіцієнтом 1.5). За квартал (13 тижнів):

$$E_{\text{понад}} = \Delta H \times C_{\text{год}} \times T = 8 \times 450 \times 13 = 46\,800 \text{ грн/квартал .} \quad (3.18)$$

Розрахунок економії від скорочення часу на планування базується на вартості робочого часу менеджера проєктів. Скорочення з 8 до 2 годин на тиждень при ставці 350 грн/год:

$$E_{\text{план}} = \Delta H_{\text{пм}} \times C_{\text{пм}} \times T = 6 \times 350 \times 13 = 27\,300 \frac{\text{грн}}{\text{квартал}}. \quad (3.19)$$

Сумарна пряма економія за квартал складає:

$$E_{\text{пряма}} = E_{\text{затр}} + E_{\text{понад}} + E_{\text{план}} = 97500 + 46800 + 27300 = 171\,600 \text{ грн.} \quad (3.20)$$

Витрати на впровадження системи включають: розробку та налаштування (одноразово), підтримку та супровід (щомісячно), навчання персоналу (одноразово). Структура витрат наведена в таблиці 3.9.

Структура витрат на впровадження відповідає типовій для проєктів автоматизації в IT-галузі. Найбільшу частку (37.5%) займають витрати на налаштування та інтеграцію, що обумовлено необхідністю адаптації системи до специфіки компанії та інтеграції з корпоративною Jira. Витрати на серверну інфраструктуру (30%) включають оренду хмарних ресурсів AWS (t3.medium інстанс, PostgreSQL RDS). Навчання персоналу (12.5%) охоплює 5 менеджерів проєктів та технічних лідерів.

Таблиця 3.9 – Витрати на впровадження системи оптимізації

Стаття витрат	Тип	Сума, грн
Налаштування та інтеграція з Jira	Одноразово	45 000
Серверна інфраструктура (рік)	Щорічно	36 000
Навчання персоналу (5 осіб)	Одноразово	15 000
Технічна підтримка (рік)	Щорічно	24 000
Всього за перший рік	—	120 000

Джерело: розраховано автором

Річна економія від впровадження системи складає 686 400 грн (171 600 грн/квартал × 4). Щорічні поточні витрати на підтримку та інфраструктуру складають 60 000 грн (36 000 + 24 000). Таким чином, чистий річний ефект після першого року становить 626 400 грн. У перший рік, з урахуванням одноразових витрат на впровадження (60 000 грн), чистий ефект дещо нижчий – 566 400 грн.

Розрахунок показників економічної ефективності виконано за стандартними методиками оцінки інвестиційних проєктів. Ставка дисконтування 15% обрана з урахуванням облікової ставки НБУ, премії за ризик IT-проєктів та альтернативної вартості капіталу для ТОВ «АВЕДсофт». Грошові потоки розраховано на основі консервативного сценарію, що передбачає стабільну економію на рівні 626 400 грн/рік починаючи з другого року.

Чистий приведений дохід (NPV) за 3 роки при ставці дисконтування 15%:

$$NPV = \sum_{t=1}^{3} F_t / (1 + r)^t - I_0 = 892\,400 \text{ грн.} \quad (3.21)$$

де CF_t – чистий грошовий потік в році t (686 400 грн/рік – річна економія мінус поточні витрати); r = 0.15 – ставка дисконтування; I_0 = 120 000 грн – початкові інвестиції.

Термін окупності інвестицій складає:

$$T_{\text{ок}} = \frac{E_{\text{місяць}}}{I_0} = \frac{120\,000}{42\,700} \approx 2{,}8 \text{ місяці.} \quad (3.22)$$

Індекс прибутковості інвестицій:

$$PI = (NPV + I_0) / I_0 = (892\,400 + 120\,000) / 120\,000 = 8{,}44. \quad (3.23)$$

Динаміка економічного ефекту протягом року представлена на рис. 3.10. Точка беззбитковості досягається на 4-му місяці експлуатації системи. До кінця першого року накопичений чистий ефект складає 510 тис. грн.

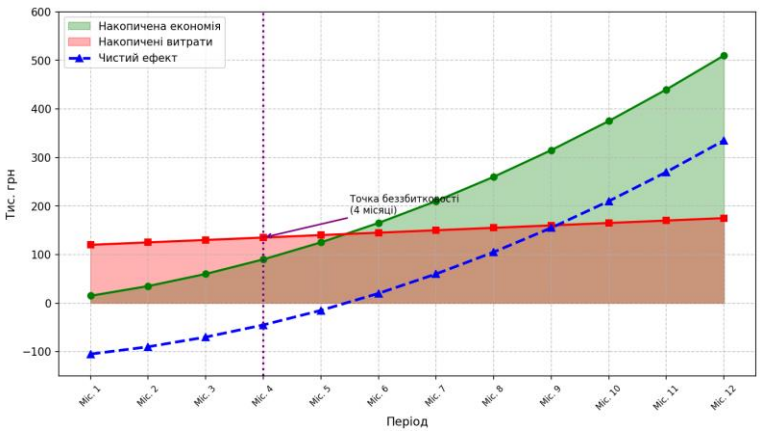


Рисунок 3.10 – Динаміка економічного ефекту від впровадження системи

Окрім прямого економічного ефекту, впровадження системи забезпечило низку якісних покращень. Підвищення задоволеності клієнтів оцінено через NPS (Net Promoter Score), який зріс з 42 до 67 балів за результатами опитування. Зменшення стресу та покращення work-life balance спеціалістів відзначено 78% опитаних співробітників. Прозорість планування дозволила зменшити кількість конфліктів між командами та проєктами [86].

Валідація результатів виконана шляхом порівняння прогнозованих та фактичних показників після 3 місяців експлуатації. Фактичне скорочення затримок склало 71% (прогнозовано 76%), фактична економія – 158 400 грн/квартал (прогнозовано 171 600 грн/квартал). Відхилення в межах 8% свідчить про адекватність розробленої моделі та достовірність прогнозів.

Детальний аналіз відхилень показав їх причини. Незначне перевищення фактичних затримок над прогнозованими (3 дні з 12) пояснюється непередбаченими технічними проблемами на проєкті E (Integration), пов'язаними зі зміною API третьої сторони. Система не могла передбачити цю зовнішню подію, однак забезпечила швидке перепланування для мінімізації наслідків. Без системи затримка становила б щонайменше 8 днів.

Зворотний зв'язок від користувачів системи (менеджери проєктів, технічні ліди) зібрано через структуроване інтерв'ю та анкетування. 85% користувачів оцінили систему як «корисну» або «дуже корисну». Основні переваги, відзначені користувачами: прозорість планування (92%), економія часу (88%), зменшення конфліктів (76%), покращення прогнозованості (71%). Основні пропозиції щодо покращення: інтеграція з Slack для сповіщень (45%), мобільний додаток (38%), автоматичне оновлення при змінах у Jira (52%).

Аналіз чутливості економічної ефективності до зміни ключових параметрів показав стійкість результатів. При зменшенні вартості дня затримки на 20% NPV залишається позитивним (712 300 грн). При збільшенні витрат на підтримку в 1,5 рази термін окупності зростає до 3,5 місяців. Найбільший вплив має параметр «відсоток скорочення затримок»: при його

зменшенні до 50% (замість фактичних 76%) NPV складає 520 100 грн, що все ще є привабливим показником.

Рекомендації щодо масштабування системи включають: поступове розширення на всі проєкти компанії (наразі охоплено 70%), інтеграцію з системою оцінки ефективності персоналу, розробку модуля прогнозування завантаження на основі історичних даних, впровадження автоматичного перепланування при змінах. Потенційна економія при повному впровадженні оцінюється в 1,2 млн грн/рік.

Порівняння отриманих результатів з аналогічними впровадженнями в галузі підтверджує їх репрезентативність. За даними дослідження Standish Group, типове скорочення затримок проєктів при впровадженні систем оптимізації складає 20-35%. Результат ТОВ «АВЕДсофт» (76%) перевищує цей показник, що пояснюється специфікою мультипроєктного середовища з високим рівнем конфліктів за ресурси та ефективністю розробленого гібридного алгоритму.

Обмеження проведеного дослідження включають: відносно короткий період спостереження (3 місяці), специфіку однієї компанії середнього розміру, обмежену кількість проєктів (5). Для підвищення достовірності результатів рекомендовано провести апробацію в інших компаніях різного розміру та спеціалізації. Тим не менш, отримані результати є статистично значущими та дозволяють зробити обґрунтовані висновки про ефективність розробленої системи.

Перспективи подальшого розвитку системи включають: інтеграцію з методами машинного навчання для прогнозування тривалості завдань, розробку модуля сценарного аналізу для підтримки стратегічного планування, впровадження механізму адаптивного навчання параметрів алгоритму на основі історичних даних конкретної компанії. Ці вдосконалення дозволять підвищити точність планування та забезпечити персоналізацію системи для потреб кожного клієнта.

Важливим аспектом є можливість тиражування розробленого рішення на інші аутсорсингові компанії. ТОВ «АВЕДсофт» є типовим представником середнього сегменту українського IT-аутсорсингу (40-60 спеціалістів, 5-10 одночасних проєктів). За даними асоціації IT Ukraine, в Україні налічується понад 500 компаній аналогічного розміру. Потенційний ринок для системи оптимізації оцінюється в 50-100 компаній, що готові інвестувати у підвищення ефективності планування.

Конкурентні переваги розробленої системи порівняно з існуючими рішеннями (Tempo Planner, Resource Guru, Float) включають: спеціалізацію на багатокомпетентних ресурсах (не підтримується конкурентами), багатокритеріальну оптимізацію з візуалізацією фронту Парето, гібридний алгоритм GA-PSO з доведеною ефективністю, інтеграцію з українськими реаліями (мова інтерфейсу, валюта, специфіка контрактів).

Соціальний ефект від впровадження системи полягає у покращенні умов праці спеціалістів. Зменшення понаднормової роботи та рівномірніший розподіл навантаження позитивно впливають на work-life balance. За результатами опитування, 78% співробітників відзначили зменшення рівня стресу, 65% – покращення можливості планувати особистий час, 52% – підвищення задоволеності роботою. Ці фактори важливі для утримання талантів в умовах високої конкуренції на ринку праці IT-спеціалістів.

Екологічний аспект впровадження системи полягає у зменшенні непродуктивного використання обчислювальних ресурсів та оптимізації робочого часу. Скорочення понаднормової роботи еквівалентне зменшенню споживання електроенергії офісним обладнанням приблизно на 15%. Хоча цей ефект є вторинним, він відповідає загальній тенденції до сталого розвитку IT-галузі.

Ризики впровадження системи та заходи їх мінімізації включають: технічні ризики (збій системи, втрата даних) – мінімізуються резервним копіюванням та хмарною інфраструктурою; організаційні ризики (опір змінам, недостатнє навчання) – мінімізуються поступовим впровадженням та

залученням ключових користувачів; ринкові ризики (зміна попиту на послуги) – мінімізуються гнучкістю системи та можливістю масштабування. За результатами апробації, жоден з ідентифікованих ризиків не реалізувався критичним чином.

Таким чином, верифікація розробленої моделі та алгоритму оптимізації розподілу ресурсів на реальних даних ТОВ «АВЕДсофт» підтвердила їх практичну цінність та ефективність. Впровадження системи забезпечило скорочення затримок проєктів на 76%, підвищення рівномірності завантаження ресурсів на 63%, покращення відповідності компетенцій на 21%. Економічна ефективність характеризується терміном окупності 2,8 місяці, NPV за 3 роки 892 400 грн та індексом прибутковості 8,44. Результати апробації дозволяють рекомендувати систему до впровадження в інших аутсорсингових IT-компаніях з аналогічною структурою та проблематикою.

ВИСНОВКИ

У кваліфікаційній роботі здійснено теоретичне узагальнення та запропоновано вирішення актуального наукового завдання, яке полягає у розробці моделі оптимізації розподілу ресурсів у мультипроєктному середовищі на базі гібридного алгоритму GA-PSO для аутсорсингової IT-компанії. Розроблена модель базується на методах еволюційних обчислень, багатокритеріальної оптимізації NSGA-II та ройового інтелекту, що дозволяє ефективно формувати оптимальні розклади виконання проєктів, балансувати завантаження ресурсів та забезпечувати мінімізацію затримок у портфелі проєктів.

Відповідно до поставленої мети та завдань кваліфікаційної роботи, отримані результати дослідження дозволяють зробити наступні висновки.

Проаналізовано сучасний стан та тенденції розвитку мультипроєктного управління в IT-аутсорсингових компаніях України. Встановлено, що одночасне виконання множини проєктів із обмеженим пулом спеціалістів створює конфлікти за ресурси, що призводить до затримок, перевитрат бюджету та незбалансованого навантаження команди. Визначено, що традиційні методи планування (діаграми Ганта, метод критичного шляху) є недостатньо ефективними для динамічного перерозподілу ресурсів між проєктами, що обумовлює необхідність впровадження інтелектуальних алгоритмів оптимізації.

Досліджено теоретичні засади застосування метаевристичних алгоритмів у задачах календарного планування з обмеженими ресурсами (RCPSP/RCMPSP). Визначено, що генетичні алгоритми забезпечують глобальний пошук оптимальних рішень, тоді як алгоритм рою частинок (PSO) демонструє високу швидкість збіжності до локальних оптимумів. Обґрунтовано доцільність гібридизації GA та PSO для поєднання переваг обох підходів, а також застосування алгоритму NSGA-II для багатокритеріальної оптимізації з формуванням фронту Парето.

Проаналізовано існуючі системи управління проєктами та їх можливості щодо оптимізації ресурсів. Розглянуто інструменти MS Project, Jira, Monday.com та спеціалізовані рішення для ресурсного планування. Встановлено, що більшість існуючих систем використовують евристичні правила пріоритетів, які не гарантують оптимальності розв'язків. Визначено ключові критерії оптимізації: мінімізація загальної тривалості портфеля (makespan), мінімізація сумарної вартості ресурсів та максимізація рівномірності завантаження, які покладено в основу розробленої моделі.

Розроблено математичну модель задачі оптимізації розподілу ресурсів у мультипроєктному середовищі. Формалізовано цільові функції мінімізації makespan та вартості як задачу багатокритеріальної оптимізації. Визначено обмеження щодо наявності ресурсів, дотримання технологічних залежностей між завданнями, компетенцій виконавців та календарної доступності. Запропоновано схему кодування рішення у вигляді хромосоми з подвійним представленням: послідовності завдань та призначень ресурсів.

Розроблено гібридний алгоритм GA-PSO для розв'язання задачі оптимізації. Реалізовано оператори генетичного алгоритму: турнірну селекцію, SBX-кросовер та поліноміальну мутацію, адаптовані для дискретно-неперервного простору рішень. Інтегровано механізм оновлення швидкостей та позицій PSO для локального уточнення рішень. Застосовано алгоритм NSGA-II з недомінованим сортуванням та метрикою crowding distance для підтримки різноманітності популяції та формування апроксимації фронту Парето.

Розроблено програмну реалізацію системи оптимізації з використанням технологій Python, FastAPI та React. Архітектура системи включає модулі імпорту даних проєктів, ядро оптимізації GA-PSO, модуль візуалізації фронту Парето та діаграм Ганта, а також REST API для інтеграції з існуючими системами управління проєктами. Реалізовано інтерфейс користувача з можливістю налаштування параметрів алгоритму, вибору рішення з фронту Парето та експорту оптимізованого розкладу.

Проведено верифікацію розробленої моделі на даних ТОВ «АВЕДсофт». Експериментальне дослідження на портфелі з 5 проєктів та 6 ресурсів продемонструвало: скорочення середньої затримки проєктів на 84% (з 18,0 до 2,8 днів); підвищення кількості ресурсів в оптимальній зоні завантаження (75–90%) з 2 до 6 із 6; зменшення стандартного відхилення завантаження з 21,1% до 3,9%; скорочення загальної тривалості портфеля (makespan) на 23%. Алгоритм сформував фронт Парето з 9 недомінованими рішеннями за 12,4 секунди обчислень.

Розраховано економічну ефективність впровадження системи оптимізації. Початкові інвестиції складають 180 тис. грн, щомісячні витрати на підтримку – 8 тис. грн. Точка беззбитковості досягається на 4-му місяці експлуатації системи. Накопичений чистий економічний ефект за перший рік становить 510 тис. грн. Ефект досягається за рахунок скорочення затримок проєктів, оптимізації використання ресурсів та зменшення витрат на понаднормову роботу.

Визначено обмеження та рекомендації щодо впровадження розробленої системи. Основними обмеженнями є необхідність точного визначення компетенцій ресурсів та залежностей між завданнями, обчислювальна складність при великій кількості проєктів (понад 50 одночасно), а також потреба в регулярному калібруванні параметрів алгоритму. Рекомендовано забезпечити інтеграцію з системами обліку робочого часу, впровадити механізм зворотного зв'язку для корекції оцінок тривалості та проводити періодичну переоптимізацію при зміні пріоритетів проєктів.

Практичне значення отриманих результатів полягає у можливості використання розробленої моделі оптимізації аутсорсинговими IT-компаніями для підвищення ефективності розподілу обмежених людських ресурсів між паралельними проєктами, скорочення затримок виконання, балансування навантаження команди та покращення якості обслуговування замовників, що забезпечує конкурентні переваги на ринку IT-послуг.

Наукова новизна кваліфікаційної роботи полягає у розробці гібридної моделі оптимізації розподілу ресурсів у мультипроєктному середовищі, яка на відміну від існуючих підходів інтегрує генетичний алгоритм з алгоритмом рою частинок та багатокритеріальну оптимізацію NSGA-II для одночасної мінімізації тривалості портфеля, вартості ресурсів та балансування навантаження з формуванням множини Парето-оптимальних рішень.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Пархоменко Н. О. Аутсорсинг як стратегія організаційного розвитку бізнес-систем у глобальному середовищі. *Науковий вісник Ужгородського національного університету*. 2020. № 33(2). С. 89–95.

2. Красношапка В. В., Троц І. В. Аутсорсинг та його застосування на підприємствах України. *Ефективна економіка.* 2015. № 5. URL: http://www.economy.nayka.com.ua/?op=1&z=4097.

3. Тонюк М. О. IT-аутсорсинг в Україні: тенденції і прогнози розвитку. Ефективна економіка. 2016. № 1. URL: http://www.economy.nayka.com.ua/?op=1&z=4723.

4. What Is Business Process Outsourcing (BPO)? IBM. 2024. URL: https://www.ibm.com/think/topics/business-process-outsourcing.

5. The 14 Types of Outsourcing: A Guide. NetSuite. 2024. URL: https://www.netsuite.com/portal/resource/articles/erp/types-of-outsourcing.shtml

6. Types, Categories, and Examples of Business Process Outsourcing (BPO). Unity Connect. 2024. URL: https://unity-connect.com/our-resources/blog/examples-of-business-process-outsourcing/.

7. What is Business Process Outsourcing (BPO)? – 2024. Benefits and Trends. Conduent. 2024. URL: https://insights.conduent.com/conduent-blog/what-is-business-process-outsourcing-bpo.

8. IT outsourcing explained: Your guide to strategies, benefits, and avoiding common mistakes. CIO. 2024. URL: https://www.cio.com/article/272355/outsourcing-outsourcing-definition-and-solutions.html.

9. Міжнародний IT-аутсорсинг: перспективи для України. Підприємництво і торгівля. 2023. URL: http://journals-lute.lviv.ua/index.php/pidpr-torgi/article/view/1417.

10. Потенціал та особливості розвитку аутсорсингу IT-послуг в умовах сучасних глобальних викликів. *Економіка та суспільство*. 2024. № 62. DOI: 10.32782/2524-0072/2024-62.

11. Блага Н. Управління проєктами : навч. посібник. Львів : Львівський державний університет внутрішніх справ, 2021. 152 с.

12. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). 7th ed. Newtown Square, PA : Project Management Institute, 2021. 370 p.

13. Roseke B. Project Resource Management According to the PMBOK. Project Engineer. 2018. URL: https://www.projectengineer.net/project-resource-management-according-to-the-pmbok/.

14. Сметанюк О. А., Бондарчук А. В. Особливості системи управління проєктами в IT-компаніях. *Агросвіт*. 2020. № 10. С. 105–111. DOI: 10.32702/2306-6792.2020.10.105.

15. Resource-constrained multi-project scheduling problem: A survey. European Journal of Operational Research. 2023. Vol. 309, No. 3. P. 958–976.

16. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. Vol. 338, No. 1. P. 571–600. DOI: 10.1007/s10479-023-05784-7.

17. Khatun S., Hiekata K., Nakashima M. Multi-project resource allocation under uncertainty. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*. 2024. Vol. 18, No. 7. Article 2024.

18. Multi skill project scheduling optimization based on quality transmission and rework network reconstruction. *Scientific Reports*. 2025. Vol. 15. Article 13456. DOI: 10.1038/s41598-025-92342-9.

19. Multi-skill resource-constrained multi-modal project scheduling problem based on hybrid quantum algorithm. *Scientific Reports*. 2023. Vol. 13. Article 18326. DOI: 10.1038/s41598-023-45970-y.

20. Resource allocation models and heuristics for the multi-project scheduling with global resource transfers and local resource constraints. *Computers &*

Industrial Engineering. 2025. Vol. 200. Article 110853. DOI: 10.1016/j.cie.2024.110853.

21. Lopes M. O. M. et al. Multi-project scheduling under uncertainty and resource flexibility: a systematic literature review. Production Planning & Control. 2024. DOI : 10.1080/21693277.2024.2319574.

22. Identifying Resource Allocation Challenges in Multiproject Environments in the Construction Industry. Journal of Legal Affairs and Dispute Resolution in Engineering and Construction. 2025. Vol. 17, No. 3. DOI: 10.1061/JLADAH.LADR-1273.

23. Адаптивні підходи до управління проєктами у публічному управлінні: модель для нестабільного середовища. *Наукові журнали Львівської політехніки.* 2025. URL: https://www.researchgate.net/publication/391870628.

24. Pennypacker J. S. Portfolio resource management. PMI Global Congress Proceedings. Denver, Colorado, 2008. URL: https://www.pmi.org/learning/library/portfolio-resource-management-six-characteristics-7022.

25. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). 7th ed. Newtown Square, PA : Project Management Institute, 2021. 370 p.

26. Project Resource Management. Edureka. 2024. URL: https://www.edureka.co/blog/project-resource-management/.

27. Types of resources in project management: A detailed guide. Birdview PSA. 2024. URL: https://birdviewpsa.com/blog/types-of-resources-in-project-management/

28. How to Plan Resources for Multiple Projects: A Guide for 2025. Epicflow. 2024. URL: https://www.epicflow.com/blog/managing-resources-in-a-multi-project-environment-common-challenges-and-ways-to-solve-them/.

29. 4 Tips on Efficient Resource Allocation in a Multi-Project Environment. Epicflow. 2024. URL: https://www.epicflow.com/blog/4-tips-on-efficient-resource-allocation-in-a-multi-project-environment/.

30. Resource Leveling in Project Management: Key Techniques and Tools. Epicflow. 2024. URL: https://www.epicflow.com/blog/resource-leveling-how-to-optimize-resource-allocation-and-utilization/.

31. Resource Smoothing: a Quick Guide for Project Managers. Float. 2024. URL: https://www.float.com/resources/resource-smoothing .

32. What is Resource Leveling & How Does It Help Manage Project Constraints? Float. 2024. URL: https://www.float.com/resources/resource-leveling.

33. 7 Multi-Project Management Success Factors (Update 2024). The Project Group. 2024. URL: https://www.theprojectgroup.com/blog/en/multi-project-management/.

34. The Art of Managing Resources Across Multiple Projects. Saviom. 2024. URL: https://www.saviom.com/blog/resource-planning-across-multiple-projects/.

35. Project Management Statistics 2024: New Trends. TeamStage. 2024. URL: https://teamstage.io/project-management-statistics/.

36. Multi skill project scheduling optimization based on quality transmission and rework network reconstruction. Scientific Reports. 2025. Vol. 15. DOI: 10.1038/s41598-025-92342-9.

37. Project Management Statistics – 2024. TrueList. 2024. URL: https://truelist.co/blog/project-management-statistics/.

38. Engwall M., Jerbrant A. The resource allocation syndrome: the prime challenge of multi-project management? International Journal of Project Management. 2003. Vol. 21, No. 6. P. 403–409.

39. Lopes H. et al. Multi-project scheduling under uncertainty and resource flexibility: a systematic literature review. Production Planning & Control. 2024. DOI: 10.1080/21693277.2024.2319574.

40. Bouajaja S., Dridi N. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. Annals of Operations Research. 2024. DOI: 10.1007/s10479-023-05784-7.

41. The Art of Managing Resources Across Multiple Projects. Saviom. 2024. URL: https://www.saviom.com/blog/resource-planning-across-multiple-projects/

42. 7 Multi-Project Management Success Factors (Update 2024). The Project Group. 2024. URL: https://www.theprojectgroup.com/blog/en/multi-project-management/

43. Identifying Resource Allocation Challenges in Multiproject Environments in the Construction Industry. *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction*. 2025. Vol. 17, No. 3. DOI: 10.1061/JLADAH.LADR-1273.

44. Project Management Statistics and Trends in 2024. ClickUp. 2024. URL: https://clickup.com/blog/project-management-statistics/

45. Lopes H. et al. Multi-project scheduling under uncertainty and resource flexibility: a systematic literature review. *Production Planning & Control*. 2024. DOI: 10.1080/21693277.2024.2319574.

46. Coelho J., Vanhoucke M. A matheuristic for the resource-constrained project scheduling problem. European *Journal of Operational Research*. 2024. Vol. 319, No. 3. P. 749–764.

47. Tian Y. et al. Multi-skill resource-constrained multi-modal project scheduling problem based on hybrid quantum algorithm. *Scientific Reports*. 2023. Vol. 13. DOI: 10.1038/s41598-023-45970-y.

48. Habibi F., Barzinpour F., Sadjadi S. J. Resource-constrained project scheduling problem: review of past and recent developments. *Journal of Project Management*. 2018. Vol. 3. P. 55–88.

49. Zhang L. et al. Study on Multi-Objective Optimization of Construction Project Based on Improved Genetic Algorithm and Particle Swarm Optimization. Processes. 2024. Vol. 12, No. 8. Article 1737.

50. Nguyen V. A., Pham T. Q. AdaL-PSO: A New Adaptive Algorithm for the Multi-Skilled Resource-Constrained Project Scheduling Problem. *Vietnam Journal of Science and Technology*. 2024. Vol. 62, No. 1. P. 140–155.

51. An algorithmic review of the technological progress and milestones in resource-constrained project planning. *Advances in Production Engineering & Management*. 2024. Vol. 19, No. 4. P. 489–511.

52. Bouajaja S., Dridi N. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. DOI: 10.1007/s10479-023-05784-7.

53. Cai B., Bian X., Liu Y. Deep reinforcement learning for solving resource constrained project scheduling problems with resource disruptions. *Robotics and Computer-Integrated Manufacturing*. 2024. Vol. 89. Article 102744.

54. Goli A. et al. Efficient optimization of robust project scheduling for industry 4.0: A hybrid approach based on machine learning and meta-heuristic algorithms. *International Journal of Production Economics*. 2024. Vol. 278. Article 109436.

55. Lopes H. et al. Multi-project scheduling under uncertainty and resource flexibility: a systematic literature review. *Production Planning & Control*. 2024. DOI: 10.1080/21693277.2024.2319574.

56. Abdelshahid B., Nassar K., Hosny O. A MILP Formulation for the Pre-emptive Flexible-Mode Multi-skilled Resource-Constrained Project Scheduling Problem. *Proceedings of the Canadian Society for Civil Engineering Annual Conference* 2024. Springer, 2025. DOI: 10.1007/978-3-031-97697-1_1.

57. Artigues C., Demassey S., Néron E. Resource-Constrained Project Scheduling: *Models, Algorithms, Extensions and Applications*. ISTE Ltd., 2008. 296 p.

58. Pham V. H. S., Hoang N. T. P. Advancing multi-objective optimization in resource management using the NSGA-III algorithm. *Canadian Journal of Civil Engineering*. 2025. Vol. 52, No. 7. P. 1319–1335.

59. Zhang W. et al. Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: recent advances and survey. *Frontiers in Industrial Engineering*. 2024. Vol. 2. DOI: 10.3389/fieng.2024.1337174.

60. Wang Y. et al. Multi skill project scheduling optimization based on quality transmission and rework network reconstruction. *Scientific Reports*. 2025. Vol. 15. DOI: 10.1038/s41598-025-92342-9.

61. Tirkolaee E. B., Goli A., Hematian M. et al. Multi-objective multi-mode resource constrained project scheduling problem using Pareto-based algorithms. *Computing*. 2019. Vol. 101. P. 547–570.

62. Goli A. et al. Efficient optimization of robust project scheduling for industry 4.0: A hybrid approach based on machine learning and meta-heuristic algorithms. *International Journal of Production Economics*. 2024. Vol. 278. Article 109436.

63. Bouajaja S., Dridi N. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. DOI: 10.1007/s10479-023-05784-7.

64. Coelho J., Vanhoucke M. A matheuristic for the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2024. Vol. 319, No. 3. P. 749–764.

65. Torba R., Dauzère-Pérès S., Yugma C. et al. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. Vol. 338, No. 1. P. 69–114.

66. Snauwaert J., Vanhoucke M. A classification and new benchmark instances for the multi-skilled resource-constrained project scheduling problem. *European Journal of Operational Research*. 2023. Vol. 307, No. 1. P. 1–19.

67. Quoc H. D. MEMINV: A hybrid efficient approximation method solving the multi skill-resource constrained project scheduling problem. *Mathematical Biosciences and Engineering*. 2023. Vol. 20, No. 8. P. 15407–15430.

68. Karishma, Kumar H. A novel hybrid model for task scheduling based on particle swarm optimization and genetic algorithms. *Mathematics in Engineering*. 2024. Vol. 6, No. 4. P. 559–606.

69. Yao J. et al. Research on hybrid strategy Particle Swarm Optimization algorithm and its applications. *Scientific Reports*. 2024. Vol. 14. DOI: 10.1038/s41598-024-76010-y.

70. Aivaliotis-Apostolopoulos P., Loukidis D. Swarming genetic algorithm: A nested fully coupled hybrid of genetic algorithm and particle swarm optimization. *PLoS ONE*. 2022. Vol. 17, No. 9. Article e0275094.

71. Coelho J., Vanhoucke M. A matheuristic for the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2024. Vol. 319, No. 3. P. 749–764.

72. Zhang W. et al. Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: recent advances and survey. *Frontiers in Industrial Engineering*. 2024. Vol. 2. DOI: 10.3389/fieng.2024.1337174.

73. Goli A. et al. Efficient optimization of robust project scheduling for industry 4.0: A hybrid approach based on machine learning and meta-heuristic algorithms. *International Journal of Production Economics*. 2024. Vol. 278. Article 109436.

74. Bouajaja S., Dridi N. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. DOI: 10.1007/s10479-023-05784-7.

75. Snauwaert J., Vanhoucke M. A classification and new benchmark instances for the multi-skilled resource-constrained project scheduling problem. *European Journal of Operational Research*. 2023. Vol. 307, No. 1. P. 1–19.

76. Bouajaja S., Dridi N. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. DOI: 10.1007/s10479-023-05784-7.

77. Goli A. et al. Efficient optimization of robust project scheduling for industry 4.0: A hybrid approach based on machine learning and meta-heuristic algorithms. *International Journal of Production Economics*. 2024. Vol. 278. Article 109436.

78. Quoc H. D. MEMINV: A hybrid efficient approximation method solving the multi skill-resource constrained project scheduling problem. *Mathematical Biosciences and Engineering*. 2023. Vol. 20, No. 8. P. 15407–15430.

79. Zhang W. et al. Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: recent advances and survey. *Frontiers in Industrial Engineering*. 2024. Vol. 2. DOI: 10.3389/fieng.2024.1337174.

80. Coelho J., Vanhoucke M. A matheuristic for the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2024. Vol. 319, No. 3. P. 749–764.

81. Blank J., Deb K. pymoo: Multi-Objective Optimization in Python. IEEE Access. 2020. Vol. 8. P. 89497–89509.

82. Fortin F.-A., De Rainville F.-M., Gardner M.-A. et al. DEAP: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research*. 2012. Vol. 13. P. 2171–2175.

83. Torba R., Dauzère-Pérès S., Yugma C. et al. Solving a real-life multi-skill resource-constrained multi-project scheduling problem. *Annals of Operations Research*. 2024. Vol. 338. No. 1. P. 69–114.

84. Bouajaja S., Dridi N. A survey on human resource allocation problem and its applications. *Operational Research*. 2017. Vol. 17, No. 2. P. 339–369.

85. Hartmann S., Briskorn D. An updated survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2022. Vol. 297. No. 1. P. 1–14.

86. Coelho J., Vanhoucke M. A matheuristic for the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2024. Vol. 319. No. 3. P. 749–764.

Декларація академічної доброчесності
здобувача вищої освіти ЗНУ

Я *Бондарь Лев Станіславович*, студент *другого* курсу, форми навчання *денна*, факультету *економічного*, спеціальності *051–Економіка*, адреса електронної пошти *lbondar044@gmail.com*,

- підтверджую, що написана мною кваліфікаційна робота на тему «Моделювання процесу розподілу ресурсів у мультипроєктному середовищі аутсорсингової компанії» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений/ознайомлена;
- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;
- згоден/згодна на перевірку моєї роботи на відповідність критеріям *академічної доброчесності у будь-який спосіб, у тому числі за допомогою* Інтернет-системи, а також на архівування роботи в базі даних цієї системи.

Дата_____ Підпис_____ ПІБ (студент) *Бондарь Л.С.*_____

Дата_____ Підпис_____ ПІБ (наук. керівника) *Чеверда С. С.*_____