КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Рукопис

МАХНО МИХАЙЛО ФЕДОРОВИЧ

УДК 519.874:519.852:519.816

ДИСЕРТАЦІЯ

МОДЕЛІ ТА МЕТОДИ РОЗВ’ЯЗАННЯ НЕЧІТКИХ ЗАДАЧ ОПТИМАЛЬНОГО РОЗПОДІЛУ ЧАСОВОГО РЕСУРСУ

01.05.04 – системний аналіз і теорія оптимальних рішень

Подається на здобуття наукового ступеню кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

____
(підпис, ініціали та прізвище здобувача)

Науковий керівник **Івохін Євген Вікторович**, доктор фізико-математических наук, професор

Київ – 2017

АНОТАЦІЯ

Махно М. Ф. Моделі та методи розв'язання нечітких задач оптимального розподілу часового ресурсу. – Рукопис.

Дисертація на здобуття наукового ступеня кандидата технічних наук із спеціальності 01.05.04 – Київський національний університет імені Тараса Шевченка МОН України, Київський національний університет імені Тараса Шевченка МОН України, Київ, 2017.

Підготовка дисертаційної роботи здійснювалась на кафедрі системного аналізу і теорії прийняття рішень факультету комп'ютерних наук та кібернетики Київського національного університету імені Тараса Шевченка.

Дисертаційна робота присвячена розробці та практичному впровадженню методів і алгоритмів розв'язування задач розподілу часового ресурсу при складанні розкладів виконання сукупності робіт з нечітко заданими ресурсними обмеженнями та термінами виконання, формалізації підходів для моделювання поведінки динамічних процесів з урахуванням зміни темпів плину часу.

Пошук розв'язків задач оптимізації суттєво залежить від природи параметрів, що присутні у математичній моделі. В багатьох випадках середовище вносить порушення в соціальні та економічні фактори задач. Окремої уваги заслуговують оптимізаційні задачі лінійного програмування, що сформульовані з елементами неточності та невизначеності. Проблемі пошуку розв'язків нечітких задач лінійного програмування з урахуванням величини функції належності нечітких параметрів моделі, розробці гнучких і достатньо простих обчислювальних методів, що враховують степінь точності у визначенні параметрів, присвячено роботи Л.А.Заде, Р.Р.Ягера, А.Дюбуа, А.Прада, С.А.Орловського, H.J. Zimmermann'a, А.Н.Борисова, Ю.П.Зайченка та інших.

Дослідження задач впорядковування сукупності робіт, що виконуються з урахуванням обмеженості ресурсів (задач теорії розкладів) знайшли своє відображення у роботах Дж.Адамса, Р.Л.Грехема, Х.Фишера, Е.Г.Коффмана, Б.Гиффлера, Дж. Томпсона, С.В.Севастьянова, Я.М. Шафранского та ін.

Необхідно звернути увагу, що в задачах розподілу ресурсів та теорії

розкладів не враховується динаміка зміни обсягів ресурсу. Тому подальша розробка методів та алгоритмів розв’язування задач розподілу часового ресурсу при складанні розкладів виконання сукупності робіт з нечітко заданими ресурсними обмеженнями та термінами виконання, формалізація підходів для моделювання динамічних процесів з урахуванням зміни темпів плину часу визначили мету дисертаційного дослідження.

У дисертації проведено дослідження та розробка моделей, методів і алгоритмів для розв’язання оптимізаційних задач часового розподілу з обмеженнями на ресурси, заданими у вигляді нечітких трикутних чисел. В роботі наведено класифікацію задач теорії розкладів залежно від різних характеристик, викладено математичні постановки оптимізаційних задач розподілу часового ресурсу, сформульовано загальну постановку задачі складання розкладу в термінах лінійного цілочисельного програмування, наведено постановку та спосіб розв’язання задачі оптимальної рекомбінації виконання заданої сукупності робіт. Сформульовано нечітку задачу про рюкзак як засіб розподілу нечітко визначеного часового ресурсу. Запропоновано нову схему реалізації жадібного алгоритму для розв’язування задач розподілу та рекомбінації.

Наведено основні положення про нечіткі величини та способи їх формалізації. Розглянуто принцип формування нечіткого розв’язку Белмана-Заде для нечітких задач лінійного програмування. Проведено огляд властивостей та методів розв’язування чітких оптимізаційних задач розподілу ресурсів, визначено методику їх розв’язання як задач складання розкладу.

При розв’язанні задач, в яких виникає необхідність в реалізації розподілу часового ресурсу, запропоновано враховувати нерівномірність плину часу при виконанні сукупності робіт. Формалізація зміни темпів витрат часу проводиться з урахуванням суб’єктивного оцінювання швидкості відліку часових інтервалів одиничної довжини. Визначено поняття структурованих нечітких чисел на основі побудови агрегатних функцій належності спеціальних нечітких множин.

Виділяючи інтервали з різною швидкістю плину часу, пропонується врахувати нечіткий характер термінів виконання кожної з робіт. Розв'язано задачу часового розподілу за умов нечітко заданого часового ресурсу. Запропоновано розв'язання лінійної оптимізаційної задачі з нечітко заданими технологічними коефіцієнтами, що залежать від швидкості плтну часу. Розглянуто нечітку задачу про рюкзак з нечітко заданими інтервалами налаштувань.

На основі методу найменших відхилень запропоновано спосіб розв'язання задачі розподілу часового ресурсу при виконанні заданої кількості робіт. Запропоновано схему реалізації жадібного алгоритму для розв'язування задач часового розподілу та рекомбінації. Реалізовано підходи до застосування економічних стратегій при розв'язанні задач розподілу часових ресурсів. Розподіл ресурсу часу проводиться за пропорційним принципом та на основі методів обмеженого вирівнювання досягнень і обмеженого вирівнювання втрат.

На основі сильної та слабкої зв'язності обмежень лінійних задач оптимізації розроблено метод перетворення області допустимих розв'язків з урахуванням близькості обмежень. Запропоновано методи розв'язання нечітких задач лінійного програмування на основі перетворення області допустимих розв'язків та за наявності систем альтернативних обмежень.

Побудовано гібридну динамічну модель процесу обробки сукупності завдань з використанням єдиного наявного обчислювального пристрою і без врахування диспетчеризації порядку виконання. Робота пристрою описується за допомогою моделі функціонування нейроподібного елементу, що має декілька входів і один вихід. Його стан подається скалярною величиною завантаженості, яка називається потенціалом. При цьому, на обробку кожної задачі виділяється необхідний часовий ресурс, який є пропорційним складності задачі, що виконується у даний момент. Вважається, що обробка задачі у часі приводить до зменшення складності завдання зі швидкістю, обернено пропорційною загальній складності усіх задач. Сформульовано відповідну систему диференціальних рівнянь, отримано твердження про фундаментальну матрицю її розв’язків.

Розглянуто приклади практичного розв’язування задач оптимального розподілу часового ресурсу. Проведено комп’ютерне моделювання динаміки розподілу часового ресурсу при імітації процесів тестування. Розроблено систему для проведення дистанційного online-тестування для визначення рівнів підготовки учасників тестувань та моделювання їх поведінки. Описано її можливості та вимоги для використання. Розглянуто задачу формування розкладу занять академічних груп та викладачів вищого навчального закладу у вигляді задачі дискретної оптимізації за умов обмеженості часових ресурсів і заданого аудиторного фонду. Запропоновано та впроваджено програмну реалізацію способу формування розкладу з урахуванням наявних обсягів аудиторного фонду.

Ключові слова: чіткі та нечіткі задачі лінійного програмування, часові обмеження, нечіткі трикутні числа, задача розподілу часового ресурсу, нечіткий відлік часу, складання розкладу, тестування.

ANNOTATION

Makhno M.F. Models and methods for solving fuzzy problems of time resource optimal distribution. – Manuscript.

The dissertation for a degree of candidate of technical sciences, speciality 01.05.04 – system analysis and optimal decision theory. – Taras Shevchenko National University of Kyiv MES of Ukraine, Taras Shevchenko National University of Kyiv MES of Ukraine, Kyiv, 2017.

The preparation of the dissertation was carried out at Department of System Analysis and Decision Making Theory of Computer Science and Cybernetics of Taras Shevchenko National University of Kyiv.

The thesis is devoted to the development and practical application of methods and algorithms for solving fuzzy linear programming problems with resource constraints. The ways of solving problems have been analyzed, the optimization

tasks for finding solutions in fuzzy tasks of time distribution with constraints and production costs, given in the form of fuzzy triangular numbers have been formulated.

The first section of the dissertation describes results of investigating fuzzy linear programming problems. Formulations the Knapsack problem as well as scheduling and selection of applications are studied and analyzed in detail. The main concepts of fuzzy quantities and methods of their formalization are given. The principle of the formation of the fuzzy Bellman-Zadeh solution for fuzzy linear programming problems has been exposed. A review of the properties and methods of solving precise optimization problems of resources allocation is made, and the methodology of their solution as the tasks of scheduling is defined.

In the second section, the problems of the optimal allocation of time resource are analyzed, the classification of scheduling theory problems is given. The mathematical formulations of optimizing problems of allocation in terms of integer linear programming are analyzed, the problem of timetable compilation for the performance of a given number of jobs on one machine is exposed, the method of solving the problem of optimal recombination during the accomplishment of a set of works are described.

On the basis of the method of minimum deviations, is proposed the way of solving the problem of distributing the time resource during the accomplishment of specified amount of works. A scheme for implementing a greedy algorithm for solving distribution and recombination problems is proposed. The approaches to implementing economical strategies while solving problems of distribution were described.

The third section is focused in investigating models and methods of solving fuzzy problems of optimal distribution of time resources. The fuzzy Knapsack problem is formulated as a way of distributing fuzzy determined time resource. The problem of timetable compilation with fuzz stated readjustment intervals has been described. Particular attention is paid to the theory of scheduling with fuzzy time measurement. The definition of structured fuzzy numbers for describing the measurement of time counting is made.

The last chapter of the thesis describes the use of the proposed methods for developing a computer system for organizing and testing individuals for which estimate the level to preparation exam on the basis of a given set of tests.

Key words: fuzzy and crisp linear programming tasks, time limited resource, fuzzy triangle numbers, the task of time resource distribution, fuzzy countdown, scheduling, testing.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ПО ТЕМІ ДІСЕРТАЦІЇ

1. Махно М.Ф. Про гібридну модель динаміки процесу обробки сукупності завдань / М.Ф.Махно // Вісник КНУ імені Тараса Шевченка. Серія: Кибернетика. – 2015. – №1(15). – С.22-27.
2. Івохін Є.В. Розробка засобів адаптивного тестування та автоматичного оцінювання знань / Є.В.Івохін, М.Ф.Махно // Вісник Київського університету ім. Тараса Шевченка. Серія: Фізико-математичні науки. – 2015. – №1. – С.130-133.
3. Івохін Є.В. Про один підхід до розв'язання нечіткої задачі рекомбінації / Є.В.Івохін, М.Ф.Махно // Вісник КНУ імені Тараса Шевченка. Серія: Фізико-математичні науки. – 2016. – №2. – С.94-97.
4. Івохін Є.В. Про наближений розв'язок чіткої та нечіткої задачі лінійного програмування з системою обмежень великої розмірності / Є.В.Івохін, В.О.Навродський, М.Ф.Махно //Вісник КНУ імені Тараса Шевченка. Серія: Фізико-математичні науки. – 2016. – №3. – С.69-72.
5. Ивохин Е.В. О подходе к построению структурированных нечетких множеств и их использовании для описания нечеткого отсчета времени/ Е.В.Ивохин, М.Ф.Махно // Проблемы управления и информатики. – 2017. – №5. – С.147-156.
6. Івохін Є.В. Про одну гібридну модель задачі теорії розкладів / Є.В.Івохін, М.Ф.Махно // Сборник трудов Міжнародной научной конференции

имени Т. А.Таран “Интеллектуальний анализ информации” (IAI-2016), Київ, 18-20 травня 2016. – С.90-96.

7. Івохін Є.В. Про підхід до формалізації процесу тестування на основі гібридної моделі задачі розкладів / Є.В.Івохін, М.Ф.Махно // Матеріали VII міжнар. конф. ["Сучасні проблеми математичного моделювання, прогнозування та оптимізації"], (Камя’нець-Подільський національний університет імені Івана Огієнка, березень 2016). - Камя’нець-Подільський, 2016. - С. 85.
8. Makhno M. On one hybrid model of a schedule theory problem/ M.Makhno, D.Apanasenko//Abstracts XXVII International Conference [“Problems of descision making under uncertainties”], (May 23-27, 2016, Tbilisi-Batumi, Georgia). Tbilisi-Batumi, 2016. – P.108.
9. Івохін Є.В. Про один підхід до розв’язання нечіткої задачі теорії розкладів / Є.В.Івохін, М.Ф.Махно // Тези VIII міжнар. школи-сем. [«Теорія прийняття рішень»], (Ужгорож, 26.09-01.10.2016). - Ужгорож, 2016. – С.129.
10. Івохін Є.В. Про спосіб перетворення області допустимих розв’язків в чітких та нечітких задачах лінійного програмування/ Є.В.Івохін, М.Ф.Махно// Матер. IV міжнар. наук.-техн. конф. [“Обчислювальний інтелект” (ОІ-2017)], (Київ, 16-18 травня 2017р.). - Київ, 2017. – С.118-119.

ЗМІСТ

ВСТУП

Актуальність теми. Одним з найважливіших класів систем управління є клас систем, які складаються з великої кількості підсистем, що взаємодіють між собою, інтереси яких не завжди узгоджуються або навіть можуть бути суперечливими, а ефективність функціонування оцінюється за допомогою сукупності якісних і кількісних показників. Розробка і практичне впровадження методів найбільш ефективного управління організаційними системами відносяться до наукових досліджень в області дослідження операцій.

Багато проблем оптимізації можна розглядати у вигляді задач лінійного програмування (ЗЛП), де всі цільові функції і обмеження, що визначають множину допустимих рішень, є лінійними. Розв’язок задачі оптимізації на заданій множині допустимих рішень характеризується однозначною залежністю із значенням лінійної цільової функції і за наявності єдиного критерію оптимальності легко може бути знайдено за допомогою різних методів [1]. Необхідно відзначити, що величезний внесок у розвиток сучасного математичного апарату і створення багатьох напрямів дослідження операцій внесли як зарубіжні (Р.Белман, Г.Данциг, Г.Кун, Т.Сааті, А.Кофман, Р.Форд, та ін.), так і вітчизняні (Л.В.Канторович, Б.В.Гнеденко, В.С.Михалевич, Ю.М.Ермол’єв, Н.З.Шор та ін.) вчені.

Пошук розв’язків задач оптимізації суттєво залежить від природи параметрів, присутніх в моделі. У багатьох випадках значення параметрів можуть бути відомі частково, з деякою мірою точності або не можуть бути чисельно формалізовані традиційними способами. В цьому випадку в дослідженні операцій був розроблений підхід, що враховує різні ступені задоволення отриманим рішенням. У ряді задач даний підхід забезпечує кращий розв’язок, чим традиційна оптимізація [2].

Окремої уваги серед інших заслуговують оптимізаційні задачі лінійного програмування, сформульовані з елементами неточності і невизначеності. До них відносяться задачі нечіткого лінійного програмування (НЛП), параметри в яких можуть бути описані кількісно тільки за допомогою нечітких множин. Проблеми пошуку розв’язків нечітких задач лінійного програмування з урахуванням величини функції приналежності нечітких параметрів моделі, розробка гнучких і достатньо простих обчислювальних методів, що враховують ступінь точності визначення параметрів, розглядаються, наприклад, у [2-5].

Необхідно звернути увагу на те, що формалізація неточностей в завданні параметрів може бути проведена на основі підходів, запропонованих у теорії нечітких числових множин. У цьому випадку модель параметризується таким чином, щоб розв’язок задачі задовольняв бажаному рівню значень функції належності. Така модель НЛП, як правило, є більш гнучкого по відношенню до умов реальної задачі і простою в обчислювальному плані [6].

Першим і найбільш значущим стимулом до математичної формалізації нечіткості вважається робота Заде [7]. Сучасний рівень розвитку його ідеї включає численні спроби застосування теорії нечітких множин як інструмент для адекватного математичного опису проблем формалізації неточності [8-11]. За останній період були подолані деякі проблеми формалізації реальних задач, створені деякі важливі принципи в теорії нечітких множин та її застосувань. Окремі результати, присвячені пошуку рішень нечітких задач оптимізації можна знайти в [12,13]. Величезна увага приділяється розробці методів і підходів для вирішення деяких спеціальних задач шляхом залучення так званих «м'яких обчислень» [14].

Нечіткі методи активно застосовуються в задачах формування та підтримки прийняття рішень, зокрема багатоцільового (багатокритеріального) прийняття рішень та пошуку рішень в ієрархічних оптимізаційних задачах [6, 15-22]. Широка дослідницька робота, пов'язана з нечітким прийняттям рішень, включає також розробку додатків теорії нечітких множин в теорії управління і дослідженні операцій [4]. Деякі характерні результати приведені в [5, 8, 9, 10, 23].

Багато прикладних проблем пов'язано з розподілом обмежених ресурсів в ієрархічних системах [24-30]. Це, наприклад, може бути задача збалансованого розподілу навантаження в однорідній мережі [25], розподіл робіт для паралельних комп'ютерів [26], розподіл ресурсів в процесі переговорів [27] і т.д. Основна проблема при цьому полягає у формалізації проблеми у вигляді багатокритеріальних багатоіндексних задач оптимізації [28-30]. В цьому випадку вважається, що елементи процесу та їх відносини задовольняють умовам обмеженості ресурсів, що впливає на об'єми ресурсів, які циркулюють у системі. Прикладами таких задач розподіл навантаження на канали передачі даних Internet-провайдерів [28], планування виробництва [30] та ін.

Задачі впорядковування робіт, що виконуються машинами, при використанні деяких ресурсів є задачами теорії розкладів (ТР). Результати розв’язання подібних задач мають вартісний характер або визначаються іншою величиною. Дослідження задач ТР, а також методів і алгоритмів їх вирішення дозволяє отримати оптимальні або близькі до оптимальних результати.

Теорії розкладів присвячені роботи зарубіжних і вітчизняних учених Дж.Адамса, Р.Л.Грехема [31], Х.Фишера [32], Е.Г.Коффмана, Б.Гиффлера [33], Дж. Томпсона, С.В.Севастьянова [34], Я.М.Шафранского та ін.

У ТР історично склалися два напрямки: один називається мережевим або календарним плануванням (Project Scheduling (PS)) [31-34], другий - власне теорія розкладів (Machine Scheduling (MS)) (див. рис.1). На початковому етапі досліджень вони суттєво розрізнялися у формулюваннях задач: у задачах мережевого планування, як правило присутні ресурсні обмеження і не йдеться мова про якісь машини; у теорії розкладів, навпаки, обов'язково є машини (що є специфічним видом ресурсів), є роботи, що складаються, взагалі кажучи, з декількох операцій (на відміну від робіт, що розглядаються в мережевому плануванні), і, як правило, немає інших ресурсних обмежень, хоча можуть бути присутніми досить складні обмеження специфічного вигляду. Проте, останнім часом ці напрямки все більше зближуються один з іншим.

В результаті маємо загальний висновок: обидві області ТР вирішують практично одне і те ж коло задач, але при цьому користуються різними моделями і різними методами розв’язання [34].

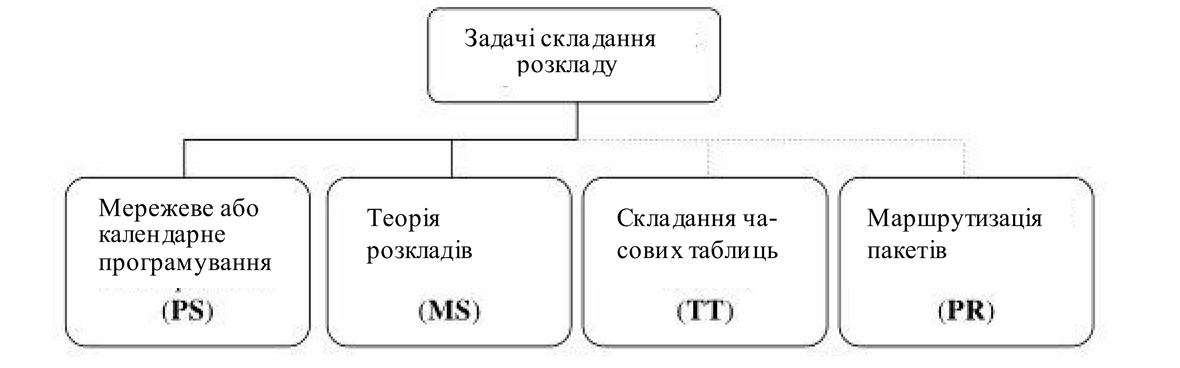


Рис.1. Основні класи задач теорії складання розкладів

Крім мережевого планування (PS) і теорії розкладів (MS), задачі на складання розкладів розглядаються в таких галузях дослідження операцій як складання тимчасових таблиць (Time tabling (TT)) і маршрутизація пакетів (Packet routing (PR)) [31-34]. До задач маршрутизації пакетів (PR) відносяться задачі складання розкладу занять у навчальному закладі, тижневих (циклічних) розкладів польотів літаків, тижневих або добових розкладів руху залізничного транспорту і т.п. Для всіх цих задач характерним є те, що інтервал часу, в рамках якого слід розподілити заданий набір робіт, відомий наперед. Таким чином, тут не розглядаються завдання на швидкодію (тобто на мінімум тривалості реалізації розкладу), а використовуються інші критерії оцінки якості розкладу, що будується.

Маршрутизація пакетів (PR) виникла у зв'язку з необхідністю в оптимізації потоків інформації, що курсують в мережах зв'язку (у локальних мережах або в глобальній мережі Інтернет). Для задач з цієї області також характерні специфічні критерії оптимальності, що не розглядаються в рамках класичної теорії розкладів.

Результати дослідження процесів функціонування різних технічних і соціальних систем приводять до висновку про необхідність і важливість рішення оптимізаційних задач, в рамках яких виникає питання про ефективний розподіл і порядок використання часових ресурсів. Такі задачі традиційно виникають, наприклад, в процесі розв’язування і оцінювання тестових завдань при проведенні різних заходів навчально-методичного характеру в різних учбових закладах.

В задачах розподілу ресурсів часто спостерігається ситуація, коли є повний набір робіт (дій), які необхідно виконати, а наявних ресурсів для виконання кожної роботи найкращим чином не вистачає. Для заданих множини робіт та обсягів наявних ресурсів потрібно розподілити ресурси поміж роботами таким чином, щоб максимізувати певний критерій ефективності (наприклад, максимізувати прибуток виробництва за умов реалізації повного переліку дій). До таких задач відносяться задача про рюкзак [35], задача календарного планування [36] та інші. В математичних моделях оптимізаційних задач серед інших розглядається критерій мінімізації загальної тривалості робіт, тобто часового інтервалу між моментами початку першого етапу робіт до завершення останнього етапу робіт.

Необхідно відзначити, що «часовий» характер задач розподілу ресурсів виділяє їх в особливу категорію, що відноситься до теорії розкладів. При цьому специфіка обліку часу істотно відрізняє їх від «об'ємних» економічних задач. Якщо в останніх потрібно відповісти на питання, що та скільки виробляти, то в задачах теорії розкладу необхідно визначити, коли і в якій послідовності виконувати роботи. Ця відмінність у специфіці задач визначає відмінність в методах і можливостях їх розв’язання.

Пошук оптимального або близького до оптимального розкладу здійснюється за допомогою методів математичного програмування. При цьому необхідно звернути увагу на те, що у математичних моделях задач оптимізації система обмежень визначає область допустимих розв’язків на основі сукупності умов, які повинні виконуватись одночасно. В задачах теорії розкладів ряд умов мають виконуватися альтернативно: або одна операція передує іншій, або навпаки.

Ряд робіт (див., наприклад, [37]) відрізняються тим, що в них розглядаються не окремі операції, а технології, що представляють собою набір операцій, в результаті дії яких може бути отриманий той чи інший продукт. Для виконання технології необхідна наявність деякої сукупності машин, що працюють одночасно, іншими словами, має місце групування машин за технологіями (в [37, 38] такі технології називаються багатопроцесорними роботами). На практиці також часто виникають задачі теорії розкладів, де для виконання операцій потрібно дотримуватися різного роду обмежень на ресурси. Розв’язання цих задач в рамках декомпозиційного підходу [39], як правило, здійснюється поетапно. При цьому, на кожному з етапів розв’язується своя сукупність завдань, а формування остаточного розкладу їх виконання складає суть задач календарного планування [40].

Серед робіт, присвячених алгоритмам розв’язування задач математичного програмування та оцінці їх складності, виділимо, наприклад, роботи [41-43]. Ефективні алгоритми розв’язання задач оптимізації розроблені для випадків, коли параметри моделей відомі апріорі. Проте, на практиці достатньо часто розглядаються ситуації, в яких ці параметри не можуть бути задані точно або є невідомими із-за специфіки окремих неконтрольованих чинників [44].

Багато досліджень зв’язано з розв’язуванням транспортної задачі (ТЗ). У роботі Bellman і Zadeh [23] запропонована концепція прийняття рішення в нечітких умовах для транспортної задачі з неточними параметрами. Стаття Lai і Hwang [45] присвячена ситуації, в якій всі параметри моделі ТЗ є нечіткими. У 1979 році Isermann [46] розробив алгоритм рішення ТЗ, який знаходить її ефективні рішення. Ringuest і Rinks [47] запропонували дві ітераційні схеми рішення лінійних багатокритеріальних транспортних задач. S.Chanas і D.Kuchta [48] розробили підхід, заснований на інтервальному визначенні неточно заданих коефіцієнтів. Tien Fuling [49] застосував метод інтерактивного нечіткого багатоцільового лінійного програмування для розв’язання задачі транспортного планування.

Існують дослідження, що розглядають зведення нечітких транспортних задач до традиційних ТЗ [50-52]. R.N.Gasimov і K.Yenilmez [53] досліджували транспортні задачі з нечіткими величинами запитів і пропозицій, вирішуючи їх шляхом зведення до параметричних моделей математичного програмування з урахуванням критерію Белмана і Заде.

Нові арифметичні операції над трапецієвидними (трикутними) нечіткими числами [52] дали можливість використовувати нечіткі числа для формалізації нечітких задач лінійного програмування. Цей підхід спростив формалізацію і рішення оптимізаційних задач, величини ресурсів в яких задаються нечіткими трикутними числами [54]. Окрім цього, залучення методики порівняння важливості критеріїв в задачах вибору дозволило узагальнити даний підхід на випадок різної важливості обмежень ЗЛП.

Серед нечітких задач оптимізації, яким дослідники приділяють достатньо багато уваги, необхідно відмітити задачі цілочисельного лінійного програмування [55-57]. До таких задач відносяться задачі оптимального розподілу ресурсів в комп'ютерних і енергетичних мережах [58, 59], задача про призначення [60, 61]. При цьому, потрібно підкреслити високу обчислювальну складність таких задач.

У науковій літературі багато публікацій присвячено огляду результатів, пов’язаних з аналізом складності та побудовою наближених алгоритмів розв’язання згаданих задач. Більшість задач складання розкладів є NP-повними задачами [62]. Лише деякі постановки з частинними умовами можуть бути розв’язані за поліноміальний час. Детальний огляд задач теорії розкладів з аналізом їх обчислювальної складності наведено у [63].

Наближені алгоритми дозволяють знаходити допустимі розв’язки з гарантованою оцінкою точності за поліноміальний час [62]. Чисельні алгоритми, що базуються на реалізації евристичних підходів, також знаходять допустимі розв’язки за прийнятний час, але, як правило, не мають апріорної оцінки похибки [63].

Дана дисертаційна робота присвячена розробці та практичному застосуванню методів і алгоритмів розв’язування нечітких задач складання розкладу виконання сукупності завдань з урахуванням змін у часових ресурсних обмеженнях. У роботі розглянуті класи задач розподілу ресурсів як задач складання розкладів, наведені основні положення про нечіткі величини і способи їх формалізації. Використовуючи принцип формування нечіткого розв’язку

Белмана-Заде в нечітких задачах лінійного програмування, розглянуті різні варіанти нечітких моделей ЗЛП (з нечіткими технологічними коефіцієнтами та ресурсними обмеженнями). Проаналізовані способи та виписані оптимізаційні задачі для знаходження нечітких рішень.

Наведено огляд результатів досліджень чітких і нечітких задач розподілу ресурсів при виконанні заданої сукупності робіт, сформульована модель розподілу часового ресурсу, дослідження якої проводиться у дисертаційній роботі. Розглянуто математичні постановки оптимізаційних задач розподілу часового ресурсу за умов відсутності та наявності термінів переналагодження робіт.

Проведено детальний аналіз задач нечіткого розподілу часового ресурсу як задач лінійного програмування з параметричними обмеженнями у вигляді нечітких трикутних чисел. Використання понять сильної і слабкої зв'язності обмежень задачі оптимізації дозволило розробити метод перетворення області допустимих розв'язків, що враховує близькість обмежень. Запропоновано схему розв'язання нечітких ЗЛП за наявності систем альтернативних обмежень. Розроблено схеми жадібних алгоритмів для розв'язування задач рекомбінації.

Формалізовано та досліджено нечіткі задачі складання розкладу виконання робіт з урахуванням впливу динаміки плину часу. Визначено спосіб побудови нечітких множини, що описують швидкість зміни часу. Запропоновано гібридну модель динаміки часового ресурсу за умов виконання/ невиконання окремих робіт, отримано твердження про загальний розв'язок цієї моделі, змодельовано динаміку функціонування схеми виконання сукупності робіт з врахуванням швидкості плину часу.

Розроблену методику використано при складанні та проведенні процесів навчання та тестування осіб, що проходять освітню перепідготовку за умов обмеженості часу та аудиторного форду. Запропоновані методи можуть бути використані при розв'язанні різних нечітких задач складання розкладів виконання заданої кількості робіт з нечітко заданими часовими ресурсними обмеженнями.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційна робота є складовою частиною наукових робіт, які ведуться на кафедрі системного аналізу і теорії прийняття рішень і в науково-дослідному секторі "Проблем системного аналізу" Київського національного університету імені Тараса Шевченка. Дослідження виконувалися в рамках науково-дослідної теми №16БФ015-02 "Розробка нових математичних методів системного аналізу і теорії оптимальних рішень та їх застосування" (державний номер реєстрації 0116U002529, термін виконання 2016-2018г.г., в рамках програми "Інформатизація суспільства") і науково-дослідної теми «Розробка і впровадження інформаційної та організаційної системи заходів по забезпеченню інноваційної спрямованості науково-дослідних робіт в Київському національному університеті імені Тараса Шевченка», НДР № 08БП 013-01 (за напрямом Підпрограми "Інформаційні технології в науці та навчальному процесі").

Мета і задачі дослідження. Метою роботи є розробка методів та алгоритмів розв'язування задач розподілу часового ресурсу при складанні розкладів виконання сукупності робіт з нечітко заданими ресурсними обмеженнями та термінами виконання, формалізації підходів для моделювання поведінки динамічних процесів з урахуванням зміни темпів плину часу (на прикладі процесу тестування осіб, що проходять перевірку рівня підготовки на основі заданої сукупності тестів).

Задачі дослідження полягають в наступному:

– провести системний аналіз сучасного стану досліджень чітких та нечітких задач складання часового розкладу виконання сукупності робіт;

– розробити методи для чіткої та нечіткій оптимізації задач розподілу часового ресурсу;

– провести моделювання динаміки процесів і систем, функціонування яких враховує різні темпи плину часу;

– розробити підхід для імітації процесу тестування з урахуванням різної складності завдань та рівня підготовки, створити комп'ютерну систему автоматизації процесу тестування.

Об'єктом дослідження є процеси знаходження розв'язків та оцінювання кількісних характеристик в системах оптимізації розподілу часових ресурсів**.**

Предмет дослідження – математичні моделі, методи та інструментальні засоби для врахування характеристик параметрів плину часу в динамічних процесах, зв'язаних з оптимальним використанням часових ресурсів.

Методи дослідження. При розв'язанні поставлених задач були використані методи системного аналізу, методи теорії оптимізації та теорії нечітких множин для побудови математичних моделей задач лінійного програмування з нечіткими ресурсними обмеженнями; методи дослідження гібридних систем для розв'язання задачі моделювання динаміки процесів з урахуванням зміни темпів плину часу.

Для подання нечітких даних при розв'язанні задачі розподілу нечітко визначених часових ресурсів використано спосіб формалізації нечітких чисел у вигляді нечітких трикутних чисел.

Наукова новизна отриманих результатів. У процесі розв'язання поставлених задач отримано нові наукові результати, які полягають в наступному:

1. на основі системного аналізу досліджень чітких та нечітких задач математичного програмування обґрунтовано доцільність і необхідність розвитку методів оптимізації для розв'язання лінійних оптимізаційних задач з нечіткими ресурсними обмеженнями;
2. вперше сформульовано задачу розподілу часових ресурсів як задачу складання розкладу виконання сукупності робіт;
3. на основі понять сильної і слабкої зв'язності обмежень задачі оптимізації вдосконалено метод перетворення області допустимих рішень на основі оцінки близькості обмежень;

4. запропоновано нову схему розв’язання нечітких задач лінійного програмування за наявності систем альтернативних обмежень;
5. вперше змодельовано динаміку процесів з урахуванням зміни темпів плину часу;
6. запропоновано нову схему жадібного алгоритму для розв’язання задачі рекомбінації;
7. розроблено нові математичні моделі процесів тестування та оптимального розподілу часових ресурсів;
8. розроблено комп’ютерну систему для організації та проведення процесів тестування осіб, що проходять перевірку рівня підготовки на основі заданої сукупності тестів.

Обґрунтованість та достовірність отриманих результатів підтверджується коректністю постановок задач, строгим доведенням теорем, узгодженістю отриманих аналітичних результатів з даними чисельного експерименту.

Практичне значення отриманих результатів полягає в тому, що в дисертаційній роботі досліджено та розв’язано ряд практичних задач складання розкладів виконання сукупності робіт з нечітко заданими часовими параметрами. Запропоновані методи та підходи створено на основі використання нечітких трикутних чисел. Ці результати можуть бути використані при створенні ефективних програмних систем для аналізу та моделювання процесів розподілу часового ресурсу в нечітких умовах, для підтримки прийняття рішень при дослідженні процесів і систем, динаміка поведінки яких враховує різні темпи плину часу. Результати дисертаційного дослідження використовувались у навчальному процесі на факультеті кібернетики Київського національного університету імені Тараса Шевченка при підготовці лекційних та спеціальних курсів «Сучасні інформаційні технології», «Методи прийняття рішень в умовах нечіткості» та «Методи дослідження нечітких динамічних систем».

Особистий внесок здобувача. Дисертація є самостійною науковою працею, в якій висвітлені власні ідеї та розробки автора, що дозволили досягти поставленої мети. Наукові положення, пропозиції та рекомендації, що виносяться на захист, отримані здобувачем самостійно. У спільних роботах із науковим керівником Є.В.Івохіним, вибір методів дослідження та доведення основних результатів виконано автором, особистий внесок якого полягає у наступному: у роботі [2] наведено опис математичних і програмних засобів адаптивного тестування та автоматичного оцінювання знань, у роботі [3] встановлено умови евристичного алгоритму розв'язання нечіткої задачі рекомбінації, у роботі [4] отримано умови сильної залежності обмежень в лінійних задачах оптимізації, наведено результати проведених обчислень, що підтверджують застосування методики скорочення обмежень, у роботі [5] запропоновано метод розподілу часового ресурсу із змінними темпами плину часу.

Апробація результатів дисертації. Матеріали дисертаційної роботи доповідалися та обговорювалися на наукових конференціях та семінарах: XVII Міжнародній конференції "Dynamic System Modeling and Stability Investigation" (Київ, травень, 2015); VII міжнародній науково-практичній конференції «Сучасні проблеми математичного моделювання, прогнозування та оптимізації» (Кам'янець-Подільський, квітень, 2016); міжнародній школі-семінарі «Теорії прийняття рішень» (Ужгород, вересень, 2016); XXVII Міжнародній науковій конференції "Problems of Decision Making under Uncertainties" (PDMU-2016), (Тбілісі-Батумі, Грузія, травень, 2016); Міжнародній науковій конференції "Интеллектуальний анализ информации" (IAI-2016) (Київ, травень, 2016); Міжнародній науково-практичній конференції «Обчислювальний інтелект» (Черкаси, травень, 2015; Київ, травень, 2017); на наукових семінарах факультету кібернетики Київського національного університету імені Тараса Шевченка.

Публікації. Основні наукові твердження, висновки і результати дисертаційної роботи опубліковані в 10 наукових працях. З них – 6 наукових статей, у тому числі 5 у фахових виданнях [1-5], 1 стаття у виданні, яке входить до наукометричної бази даних [5], 4 – праці та тези наукових конференцій [7-10].

РОЗДІЛ 1
ОСНОВНІ МАТЕМАТИЧНІ МОДЕЛІ ЧІТКИХ ТА НЕЧІТКИХ ЗАДАЧ ОПТИМАЛЬНОГО ПЛАНУВАННЯ І РОЗПОДІЛУ РЕСУРСІВ

Оптимізаційні задачі математичного програмування виникають в процесі розробки і практичного впровадження методів ефективного управління різними організаційними системами і є частиною досліджень по дослідженню операцій.

Характерними особливостями дослідження операцій є системний підхід до поставленої проблеми і аналіз проблемної області. Він полягає у всебічному дослідженні і оцінці чинників конкретної проблеми, а також їх впливи на критерії функціонування системи в цілому [64].

З іншого боку, слід зазначити, що при вирішенні кожної проблеми можуть виникати нові завдання. Тому ще однією важливою особливістю дослідження операцій є аналіз проблемної області, що полягає в глибокому і постійному вивченні суті проблем, що забезпечує послідовність процесу уточнення і вдосконалення кожного класу оптимізаційних завдань.

Важливою характеристикою завдань дослідження операцій є прагнення знайти оптимальне рішення (принцип оптимальності). На практиці існує ряд обмежень, що не дозволяють знайти таке рішення. У цих випадках ставиться питання про знаходження не оптимальних, але достатньо хороших (ефективних або компромісних) розв’язків, що задовольняють постановці задачі. Часто доводиться шукати компроміс між ефективністю розв’язків і витратами на їх пошук. Серйозні складнощі виникають при розв’язуванні оптимізаційних задач в умовах неповної інформації, а також у випадках, коли істотну роль грають випадкові чинники. Але потрібно відзначити, що головним результатом дослідження операцій є те, що він дає інструмент для пошуку таких розв’язків.

Сформулюємо умови деяких оптимізаційних задач математичного програмування. У кожній з них потрібно серед множини можливих варіантів вибрати той, для якого значення заданої цільової функції досягає екстремального значення.

1.1. Чіткі задачі оптимального планування і розподілу ресурсів та їх властивості

За своїм змістом і постановкам множину задач дослідження операцій можна розбити на класи, найпоширенішими серед яких є:

- задача управління запасами;
- задача розподілу ресурсів;
- задача ремонту і заміни устаткування;
- задача масового обслуговування;
- задача календарного планування і теорії розкладів;
- задача транспортного типу (вибору маршрутів перевезень);
- задача мережевого планування і управління;
- задача розміщення об'єктів;
- комбіновані задачі.

Більшість завдань, які вирішуються методами дослідження операцій, можна сформулювати у вигляді

$$\max F(x_1, x_2, ..., x_n) \tag{1.1}$$

при обмеженнях

$$\begin{aligned} &g_1(x_1, x_2, ..., x_n) \le b_1; \\ &g_2(x_1, x_2, ..., x_n) \le b_2; \\ &\dots\dots\dots\dots\dots\dots \\ &g_m(x_1, x_2, ..., x_n) \le b_m; \end{aligned} \tag{1.2}$$

де $F(x_1, x_2, ..., x_n)$ - цільова функція або критерій ефективності (наприклад, прибуток від виробництва деяких видів продукції, вартість перевезень і т.д.); $X = \{x_1, x_2, ..., x_n\}$ - змінні параметри; $g_1(x_1, x_2, ..., x_n)$, …, $g_m(x_1, x_2, ..., x_n)$ - функції, що визначають обмеження на наявні ресурси.

Серед різних розділів математичного програмування найбільш розвиненим і розробленим є лінійне програмування (ЛП).

Не дивлячись на вимоги лінійності функцій критеріїв і обмежень, до задач лінійного програмування відносять багато задач теорії розкладів, управління ресурсами, транспортні задачі та інші. Розглянемо деякі з них, моделі яких будуть використані і узагальнені в дисертаційному дослідженні.

■ Математичні моделі задач дискретного програмування.

Множина задач дослідження операцій описується математичними моделями дискретного програмування.

Розглянемо загальну задачу максимізації:

$$\max f(x_1, x_2, ..., x_n), \tag{1.3}$$

при обмеженнях

$$\begin{aligned} & g_1(x_1, x_2, ..., x_n) \le 0; \\ & \dots \\ & g_m(x_1, x_2, ..., x_n) \le 0; \end{aligned} \tag{1.4}$$

$$X = (x_1, x_2, ..., x_n) \in D, \tag{1.5}$$

де D - деяка множина.

Якщо множина D кінцева або що перераховує, то умова (1.40) є умовою дискретності і задача (1.3)-(1.5) – це задача дискретного програмування.

Найчастіше умова дискретності розподіляється по окремих змінних у вигляді $x_j \in D_j$, $j = 1,2,...,n$. Якщо вводиться обмеження, що x_j - ціле число, $j = 1,2,...,n$, то отримуємо задачу цілочисельного програмування (ЦП), а якщо вводиться обмеження, $j = 1,2,...,n$, - то задачу булього програмування.

У задачах дискретного програмування область допустимих рішень є неопуклою і незв'язною. Пошук рішень таких задач пов'язаний із значними складнощами. Для знаходження рішень використовуються спеціальні методи, які діляться на такі групи:

1) точні методи (метод відсікань Гоморі, метод віток і меж, метод послідовного аналізу і відсіву варіантів, адитивний метод і ін.);

2) наближені методи (метод локальної оптимізації, методи випадкового пошуку, евристичні методи).

Задачі дискретного лінійного програмування є задачами лінійного програмування, в яких є додаткові обмеження, які перетворюють область допустимих рішень на неопуклу або незв'язну.

Розглянемо деякі випадки незв'язності і неопуклості області.

1. Припустимо, що деяка змінна x_k, $1 \le k \le n$, задачі ЛП має обмеження зверху і обмеження знизу у вигляді:

$$\text{або } x_k \le a \quad \text{або } x_k \ge b, \tag{1.6}$$

де $0 \le a < b \le T_k$, T_k - деяка константа.

Введемо цілочисельну змінну y_k, що приймає значення 0 і 1. Ця додаткова змінна не включається в цільову функцію, а обмеження для k-ої змінної перепишуться у вигляді:

$$x_k - b + by_k \ge 0;$$
$$-x_k + ay_k + T_k(1 - y_k) \ge 0; \; y_k = \{0,1\}. \tag{1.7}$$

Система обмежень (1.7) еквівалентна системі (1.6), а дана задача при введенні додаткової змінної стає задачею дискретного програмування.

У загальному випадку, припустимо, що в завданні математичного програмування з допустимою областю рішень $G(x_1, x_2, ..., x_n)$ є альтернативні обмеження:

$$\begin{aligned} &\text{або } h(x_1, x_2, ..., x_n) \ge 0, \\ &\text{або } k(x_1, x_2, ..., x_n) \ge 0, \end{aligned} \tag{1.8}$$

де, $k(x_1, x_2, ..., x_n)$ - задані функції. Допустимо, що відомі нижні межі функцій, $k(x_1, x_2, ..., x_n)$ на області, позначені відповідно $k_{\min}$. Введемо додаткову змінну $y = 0$ либо 1 і розглянемо систему нерівностей

$$\begin{aligned} &h(x_1, x_2, ..., x_n) - h_{\min} y \ge 0, \\ &k(x_1, x_2, ..., x_n) - k_{\min}(1 - y) \ge 0 \end{aligned}. \tag{1.9}$$

Ця система еквівалентна альтернативному обмеженню (1.8), а початкова задача стає задачею дискретного програмування.

Окремо розглянемо випадок, в якому дозволяється використовувати K з N заданих обмежень ($K \le N$). Без обмеження загальності, вважатимемо, що ці N обмежень записані у вигляді

$$
\begin{aligned}
& g_1(x_1, x_2, ..., x_n) \le b_1, \\
& g_2(x_1, x_2, ..., x_n) \le b_2, \\
& \quad ...... \\
& g_N(x_1, x_2, ..., x_n) \le b_N.
\end{aligned}
\tag{1.10}
$$

Введемо N штучних змінних $y_i \ge 0, y_i \le 1, y_i \in Z, i = \overline{1, N}$, і задамо деяке число $M >> 0$. Використовуючи нескладні викладення, отримаємо еквівалентний запис обмежень у вигляді системи нерівностей

$$
\begin{aligned}
& g_1(x_1, x_2, ..., x_n) \le b_1 + y_1 M, \\
& g_2(x_1, x_2, ..., x_n) \le b_2 + y_2 M, \\
& \quad ...... \\
& g_N(x_1, x_2, ..., x_n) \le b_N + y_N M, \\
& \sum_{i=1}^{N} y_i = N - K, y_i \ge 0, y_i \le 1, y_i \in Z, i = \overline{1, N}.
\end{aligned}
\tag{1.11}
$$

Оскільки обмеження на y_i, $i = \overline{1, N}$, гарантують, що K з цих штучних змінних будуть нульовими, а інші рівні 1, K початкових обмежень залишаться незмінними, а інші будуть виключені з розгляду за рахунок ефекту усунення.

Важливим класом задач дискретного програмування є задача бульового програмування:

$$
\sum_{j=1}^{n} c_j x_j \to \max \tag{1.12}
$$

при обмеженнях

$$
\sum_{j=1}^{n} a_{ij} x_j \le b_i \quad i = 1, 2, ..., m, \tag{1.13}
$$

$$
x_j = \{0,1\} \quad j = 1, 2, ..., n. \tag{1.14}
$$

До цього класу завдань відносяться різні задачі дослідження операцій, такі як задача розміщення виробництв, задача про призначення, задача синтезу структури транспортних і інформаційних мереж і ін.

1.1.1. Задача про рюкзак

За структурою математичної моделі задачі дискретного програмування діляться на декілька класів. Однією з найбільш відомих серед них є задача про рюкзак.

Зміст задачі полягає у наступному. Припускається, що є набір предметів, у кожного з яких відомо два параметри - об'єм і цінність. Предмети необхідно покласти у рюкзак, визначеної місткості. При цьому, завдання полягає в тому, щоб зібрати рюкзак з максимальною цінністю предметів усередині, дотримуючись при цьому обмеження на об'єм рюкзака [1].

Нехай задано об'єм рюкзаку V і список з n предметів, кожен з яких має об'єм v_j і вартість c_j, $j=\overline{1,n}$, при чому всі об'єми і всі вартості задаються додатними числами. Математична постановка задачі полягає у визначенні кількості кожного з предметів, узятих в рюкзак, щоб при цьому сумарна цінність рюкзака була максимальною з урахуванням обмеження на об'єм рюкзака.

Позначимо через x_j - кількість предметів j-го типу, $j=\overline{1,n}$, в рюкзаку. В задачі про рюкзак, яка називається задачею з необмеженим рюкзаком, кожен предмет може бути вибраний необмежену кількість разів. Очевидно, що ці величини будуть цілими, а математична модель задачі, яка полягає у визначенні набору предметів з найбільшою загальною цінністю, записується у вигляді:

$$\sum_{j=1}^{n} c_j x_j \to \max \tag{1.15}$$

$$\sum_{j=1}^{n} v_j x_j \le V, \tag{1.16}$$

$$x_j \in Z_+, \ j=1,2,...,n. \tag{1.17}$$

У бульовій постановці оптимізаційної задачі про рюкзак введемо логічні змінні x_j, $j=1,...,n$, для визначення наявності предмета j в наборі:

$$x_j = \begin{cases} 1, \text{\textit{якщо предмет з номером} } j \text{ \textit{розміщено в рюкзаці}}, \\ 0, \text{\textit{інакше}}, \end{cases} \quad j=1,...,n.$$

Загальний об'єм заповненого рюкзака обмежується величиною його місткості, що формально записується у вигляді нерівності (1.16) за умов використання булевих змінних

$$x_j \in \{0,1\},\ j = \overline{1,n}. \qquad (1.18)$$

а максимізації сумарної цінності предметів у рюкзаку відповідає цільова функція (1.15).

1.1.2. Задача складання розкладу

Припустимо, що є n робіт і m машин. Для кожної роботи заданий порядок виконання на машинах, тобто робота j спочатку виконується на машині з номером $j(1)$, потім на машині з номером $j(2)$ і т.д. В кожний момент часу машина може виконати не більш однієї роботи, кожна робота виконується не більше, як на одній машині. Роботи не перериваються. Відома тривалість pij виконання роботи j на машині i. Необхідно виконати всі роботи та мінімізувати суму моментів часу завершення всіх робіт.

Введемо цілочисельні змінні t_{ij}, що визначають час початку виконання роботи j на машині i. Використаємо mnn бульових змінних, щоб виключити умови перетину робіт при виконанні на одній машині:

$$x_{ijk} = \begin{cases} 1, \textit{якщо робота } j \textit{ передує роботі } k \textit{ на машині } i, \mathrm{j} < \mathrm{k}, \\ 0, \textit{у протилежному випадку} \end{cases}.$$

При цьому виникають дві взаємно виключні умови. Якщо робота j передує роботі k на машині i, то час початку роботи k повинен відбутися не раніше часу завершення роботи j. Маємо

$$t_{ik} \geq t_{ij} + p_{ij}, \text{ якщо } x_{ijk} = 1, \text{ і } t_{ij} \geq t_{ik} + p_{ik}, \text{ якщо } x_{ijk} = 0.$$

Нескладно перевірити, що ці умови можна переписати за допомогою наступних нерівностей:

$$t_{ik} \geq t_{ij} + p_{ij} - W(1 - x_{ijk}),$$

$$t_{ij} \geq t_{ik} + p_{ik} - Wx_{ijk},\ i = \overline{1,m},\ j,k = \overline{1,n},$$

де W — досить велике додатне число.

Кожна робота складається з операцій, що виконуються на різних машинах, і (r+1)-а операція роботи j не може початися, поки не буде завершена попередня r-я операція. Це означає, що

$$t_{j(r+1)j} \geq t_{j(r)j} + p_{j(r)j}.$$

Отже, остаточно, математична модель має такий вигляд. Цільова функція - мінімальна сума моментів часу завершення всіх робіт:

$$\min \sum_{j=1}^{n} (t_{j(m)j} + p_{j(m)j}) \quad (1.19)$$

при обмеженнях:

$$\begin{aligned} & t_{j(r+1)j} \geq t_{j(r)j} + p_{j(r)j}, \; r = \overline{1, m-1}, \; j = \overline{1, n}; \\ & t_{ik} \geq t_{ij} + p_{ij} - W(1 - x_{ijk}), \; i = \overline{1, m}, \; j, k = \overline{1, n}; \\ & t_{ij} \geq t_{ik} + p_{ik} - W x_{ijk}, \; i = \overline{1, m}, \; j, k = \overline{1, n}; \\ & t_{ij} \geq 0, \; i = \overline{1, m}, \; j = \overline{1, n}; \\ & x_{ijk} \in \{0,1\}, \; i = \overline{1, m}, \; j, k = \overline{1, n}. \end{aligned} \quad (1.20)$$

Нескладно помітити, що у цільовій функції (1.19) величина $\sum_{j=1}^{n} p_{j(m)j}$ є постійною і не залежить від порядку виконання робіт. Отже, оптимальний розв'язок не зміниться, якщо цільову функцію (1.19) замінити на $\min \sum_{j=1}^{n} t_{j(m)j}$.

1.1.3. Задача про вибір заявок

Нехай дано n заявок на проведення занять в одній аудиторії. Два різних заняття не можуть перекриватись за часом. В кожній заявці вказані початок і кінець заняття. Різні заявки можуть перетинатись, і тоді можна задовольнити тільки одну з них. Необхідно набрати максимальну кількість сумісних заявок [65].

Нехай $\alpha_1, ..., \alpha_n$ - моменти початку, а $\beta_1, ..., \beta_n$ - моменти закінчення занять з номерами від 1 до n. Тоді $[\alpha_i, \beta_i)$ - проміжок часу проведення заняття з номером i. Необхідно вибрати максимальне число занять так, щоб відповідні їм проміжки не перетинались (моменти початку одного заняття можуть збігатись з моментами початку наступного за ним).

1.2. Загальна постановка нечітких задач лінійного програмування

Для опису неточності математичної моделі найбільш розповсюдженим є стохастичний підхід, згідно з яким на множині параметрів задається імовірнісний розподіл, який інтерпретується наступним чином: імовірність події дорівнює частоті її появи в (нескінченній) послідовності незалежних випробувань. Таким чином, для адекватного застосування стохастичних принципів при моделюванні систем необхідно, щоб величина, що спостерігається, була результатом усереднення незалежних випадкових величин. Якщо експеримент принципово не може бути повторений багато разів, при стохастичному підході зазвичай пропонується інтерпретація результатів моделювання шляхом “уявного експерименту”, результати якого вже не зв’язані безпосередньо з реальними даними. Опис таких подій в термінах імовірностей вже є неприроднім. Неадекватність імовірнісних моделей проявляється при описі думки експерта: в різні моменти часу експерт може приймати різні рішення при одних і тих же умовах. Для таких систем природно використовувати нечітких підхід, при якому неточність задання параметрів моделі описується в термінах теорії нечітких множин.

1.2.1. Нечіткі величини і нечіткі числа.

В останній час теорія нечітких множин перетворилась в детально вивчену область з широким спектром задач практичного характера. Термін «нечітка величина» вперше був використаний Кофманом у [66], а після цього з’явився в роботах [67] і [68].

Одним з способів опису нечітких величин є підхід, запропонований Наміасом [68] і в якому використовується поняття простору можливостей.

Означення 1.1. [69] Нечітка величина визначається як функція з простору можливостей $(\theta, \mathrm{P}(\theta), Pos)$ у множину дійсних чисел R^1, де θ - деяка непорожня множина, $Pos\{A\}$ - величина міри можливості для довільного $A \in \mathrm{P}(\theta)$, $\mathrm{P}(\theta)$- множина усіх підмножин для θ.

Нечіткі величини можуть бути визначені різними іншими способами. У прикладних задачах для формалізації нечіткої величини з метою збільшення конструктивності використовуються означення, які еквівалентні класичному означенню *1.1* нечіткої множини. Розглянемо в якості універсальної множини X множину дійсних чисел, тобто $X = R^1$.

Означення 1.13. [70] Нечітким числом називається впорядкована пара функцій $(u(r), v(r))$, $r \in [0,1]$, які задовольняють наступним умовам:

1. $u(r)$ обмежена, неперервна зліва, неспадна функція на $[0,1]$;
2. $v(r)$ обмежена, неперервна зліва, незростаюча функція на $[0,1]$;
3. $u(r) \le v(r)$, $r \in [0,1]$.

При цьому, довільне чітке число a подається у вигляді нечіткого числа, у якого $u(r) = v(r) = a$, $r \in [0,1]$, і для будь-яких двох нечітких чисел $x = (u_1(r), v_1(r))$, $y = (u_2(r), v_2(r))$ і $\lambda \in R$ можна визначити арифметичні операції і відношення у вигляді

- $x = y$, якщо і $v_1(r) = v_2(r)$ $r \in [0,1]$;
- $x + y = (u_1(r) + u_2(r), v_1(r) + v_2(r))$, $r \in [0,1]$;
- $x - y = (u_1(r) - v_2(r), v_1(r) - u_2(r))$, $r \in [0,1]$;
- $\lambda x = \begin{cases} (\lambda u_1(r), \lambda v_1(r)), \lambda > 0, \\ (\lambda v_1(r), \lambda u_1(r)), \lambda < 0. \end{cases}$

Далі вважаємо, що будь-яка множина, що належить сукупності нечітких множин $K_X(\tilde{A})$ універсальної множини $X = R^1$, є нормальною і опуклою.

Означення 1.2. [52] Нечітким трапецеїдальним числом $\tilde{A}$ називається впорядкована четвірка чисел *(a,b,c,d)*, $a \le b \le c \le d$, що визначають функцію належності $\mu_{\tilde{A}}(x)$ вигляду:

1. $\mu_{\tilde{A}}(x)=\dfrac{x-a}{b-a}, x\in[a,b];$

2. $\mu_{\tilde{A}}(x)=1, x\in[b,c];$ (1.21)

3. $\mu_{\tilde{A}}(x)=\dfrac{d-x}{d-c}, x\in[c,d];$

4. $\mu_{\tilde{A}}(x)=0, x\notin[a,d]$.

Якщо покласти $b=c$, то отримаємо нечітке число, яке називається трикутним нечітким числом (триплетом).

Означення 1.3. [52] Нечітким трикутним числом $\tilde{A}$ називається впорядкована трійка чисел *(a,b,c)*, $a\le b\le c$, що визначають функцію належності $\mu_{\tilde{A}}(x)$ вигляду:

1. $\mu_{\tilde{A}}(x)=\dfrac{x-a}{b-a}, x\in[a,b];$

2. $\mu_{\tilde{A}}(x)=\dfrac{c-x}{c-b}, x\in[b,c];$ (1.22)

3. $\mu_{\tilde{A}}(x)=0, x\notin[a,c]$.

Відзначимо, що нечітке трикутне число *(a, b, c)*, $a\le b\le c$, є нечітким числом з функціями $v(r)=(b-c)r+c,\ u(r)=(b-a)r+a,\ r\in[0,1]$.

Нечітке трикутне число вигляду *(a, b, b)*, що називається лівим нечітким трикутним числом, визначається функцією належності вигляду

$$\mu_{\tilde{A}}(x)=0, x<a;\ \ \mu_{\tilde{A}}(x)=\frac{x-a}{b-a}, x\in[a,b]; \mu_{\tilde{A}}(x)=1, x>b, \tag{1.23}$$

а нечітке трикутне число вигляду *(b, b, c)*, що називається правим нечітким трикутним числом, - функцією належності

$$\mu_{\tilde{A}}(x)=1, x<b; \mu_{\tilde{A}}(x)=\frac{c-x}{c-b}, x\in[b,c];\ \mu_{\tilde{A}}(x)=0, x>c. \tag{1.24}$$

Обробка нечітких величин пов'язана з побудовою і застосуванням бінарних відношень. Найчастіше використовується поняття нечіткого бінарного відношення з простору X у простір Y. Тому у подальшому викладенні даного дослідження під нечітким бінарним відношенням будемо розуміти нечітку множину $\widetilde{R}$, яку визначено на декартовому добутку $X \times Y$, з функцією належності $\mu_{\widetilde{R}} : X \times Y \to [0,1]$.

У випадку, коли множини X і Y співпадають, нечітку множину з функцією належності $\mu_{\widetilde{R}} : X \times X \to [0,1]$ називають нечітким відношенням $\widetilde{R}$ на множині X.

Відзначимо одну властивість множині рівня довільних нечітких трикутних чисел. Множина α–рівня для нечіткого трикутного числа $\widetilde{A}$, поданого у вигляді триплету *(a, b, c)*, є відрізком вигляду $[a_\alpha, c_\alpha] \subseteq [a, c]$, де $a_\alpha : \mu_{\widetilde{A}}(a_\alpha) = \alpha$ і $c_\alpha : \mu_{\widetilde{A}}(c_\alpha) = \alpha$. При цьому, множина α–рівня для нечіткого лівого трикутного числа $\widetilde{A}$ вигляду *(a, b, b)* задається відрізком $[a_\alpha, b]$, де $a_\alpha : \mu_{\widetilde{A}}(a_\alpha) = \alpha$, а множина α–рівня для нечіткого правого трикутного числа $\widetilde{A}$ вигляду *(b, b, c)* – відрізком вигляду $[b, c_\alpha]$, де $c_\alpha : \mu_{\widetilde{A}}(c_\alpha) = \alpha$.

Чіткі значення a_α і c_α, отримані з множини заданого α–рівня для нечітких трикутних лівих і правих чисел $\widetilde{A}$, залежать від величини діапазону носія і форми функції належності нечіткого числа $\widetilde{A}$. При цьому, справедливі представлення $a_\alpha = a + \alpha(b - a)$ і $c_\alpha = c - \alpha(c - b)$ для $\forall \alpha \in [0,1]$.

1.2.2. Принцип формулювання нечітких задач лінійного програмування.

Перед тим, як розробляти конкретну модель лінійного програмування в нечітких умовах, повинно бути зрозуміло, що на відміну від класичної задачі лінійного програмування модель "нечіткого лінійного програмування" є неоднозначно визначеним типом моделі, і що можливі численні варіанти її реалізації в залежності від передумов або особливостей реальної ситуації, які мають бути змодельовані.

Розглянемо з початку базову модель задач "нечіткого лінійного програмування". У цьому випадку будемо вважати, що особа, що приймає рішення, може встановити для значень цільової функції моделі (1.3)-(1.4) лінійного вигляду рівень надійності Z (деякий бажаний рівень), який необхідно досягти. Припустимо також, що кожне з обмежень формується на основі даних з нечіткої множини. Отримуємо загальний вигляд нечіткої ЗЛП: знайти вектор x такий, що

$$c^T x \tilde{\geq} Z\,, \tag{1.25}$$

$$Ax \tilde{\leq} b\,,\ x \geq 0\,. \tag{1.26}$$

Тут знаки “$\tilde{\leq}$” застосовується для позначення нечіткого варіанту нестрогого відношення “$\leq$” і може бути лінгвістично інтерпретований як "по суті менше або дорівнює". Знак “$\tilde{\geq}$”, відповідно, позначає фазифіковану версію відношення “$\geq$”, що лінгвістично може бути сформульовано як "істотніше, ніж більше або дорівнює". Формально нечіткі відношення “$\tilde{\leq}$” і “$\tilde{\geq}$” можна розглядати як часткові випадки нечітких відношень відповідно “нестрого менше” і “нестрого більше” на множині $X = R^1$.

Цільова функція в даному випадку може бути сформульована у вигляді мінімізації величини функції цілі у ЗЛП (1.3) для розгляду Z в якості верхньої границі значень оптимального розв’язку. Очевидно, що формат умов (1.24)-(1.25) повністю симетричний відносно цільової функції та обмежень. При цьому бажано, щоб задача зберегла вигляд задачі лінійного програмування.

Зробивши заміни $\begin{pmatrix} -c \\ A \end{pmatrix} = B$, $\begin{pmatrix} -Z \\ b \end{pmatrix} = p$, перепишемо обмеження (1.25)-(1.26) у вигляді:

$$Bx \tilde{\leq} p\,,\ x \geq 0\,. \tag{1.27}$$

Припустимо, що для опису Z і b задано нечіткі множини у вигляді нечітких чисел. З урахуванням невизначеності в формалізації величини Z і вектору b права частина система нерівностей (1.26) представляє собою вектор, елементами якого можуть розглядатися деякі значення носіїв для заданих нечітких множин. При цьому, кожна з (m + 1) нерівностей (1.27) при визначених значеннях вектора p має множину розв’язків, що характеризується відповідною величиною функції належності, кожну з яких позначимо через

$\mu_i(x), i=\overline{1,m+1}$. Не обмежуючи загальності, будемо вважати, що всі функції належності монотонно зростають. Тоді при довільних $\alpha\in[0,1]$ значення $\mu_i(x)=\alpha$, $i=\overline{1,m+1}$, можна інтерпретувати як величини ступеня надійності, з якими вектор x задовольняє чітким нерівностям $\sum_{j=1}^{n} B_{ij}x_j \le \tilde{p}_i$, $i=\overline{1,m+1}$ (тут B_{ij} - елементи матриці B, а $\tilde{p}_i$ - значення, отримані на основі множин заданого α–рівня для нечітких елементів вектору p). Зважаючи на означення розв'язку ЗЛП, функція належності нечіткої множини «розв'язку» $\tilde{P}$ нечіткої моделі (1.27) буде задаватися у вигляді

$$\mu_{\tilde{P}}(x)=\min_{i=\overline{1,m+1}}\mu_i(x). \tag{1.28}$$

Якщо припустити, що особа, яка приймає рішення зацикавлена не у нечіткій множині розв'язків, а у чіткому "оптимальному" розв'язку початкової задачі, то можна запропонувати «покращити розв'язок" у (1.28) за допомогою розв'язування задачі нелінійного програмування

$$\max_{x\ge 0}\min_{i=\overline{1,m+1}}\mu_i(x)=\max_{x\ge 0}\mu_{\tilde{P}}(x). \tag{1.29}$$

Далі необхідно уточнити вигляд функцій належності $\mu_i(x)$, $i=\overline{1,m+1}$. Величини $\mu_i(x)$ повинні бути рівними 0, якщо обмеження (а також і цільова функція) сильно порушені, і дорівнює 1, якщо вони повністю задоволені (в звичному чіткому сенсі); і $\mu_i(x)$ повинні монотонно зростати від 0 до 1, тобто:

$$\mu_i(x)=\begin{cases}1, & \sum_{j=1}^{n} B_{ij}x_j \le p_i,\\ \in[0,1], & p_i<\sum_{j=1}^{n} B_{ij}x_j \le p_i+d_i,\ i=\overline{1,m+1}.\\ 0, & \sum_{j=1}^{n} B_{ij}x_j > p_i+d_i,\end{cases} \tag{1.30}$$

Використовуючи найпростіший вигляд функцій належності, припустимо, що вони будуть лінійними зростаючими на відповідних "інтервалах допустимості" $[p_i, p_i+d_i]$, $i=\overline{1,m+1}$:

$$\mu_i(x) = \begin{cases} 1, & \sum\limits_{j=1}^{n} B_{ij} x_j \le p_i, \\ 1 - (\sum\limits_{j=1}^{n} B_{ij} x_j - p_i)/d_i, & p_i < \sum\limits_{j=1}^{n} B_{ij} x_j \le p_i + d_i, \quad i = \overline{1, m+1}. \\ 0, & \sum\limits_{j=1}^{n} B_{ij} x_j > p_i + d_i, \end{cases} \quad (1.31)$$

Значення d_i, $i = \overline{1, m+1}$, є суб'єктивно заданими величинами допустимих порушень в обмеженнях і цільовій функції. Підставляючи (1.31) в (1.29), після нескладних перетворень [71] і з урахуванням деяких додаткових припущень, отримаємо критерій вибору оптимального розв'язку

$$\max_{x \ge 0} \min_{i=1,m+1} \left(1 - (\sum_{j=1}^{n} B_{ij} x_j - p_i)/d_i \right). \quad (1.32)$$

Введемо нову змінну $\lambda \in [0,1]$, яка відповідає рівню функції належності нечіткої множини «розв'язку» $\tilde{P}$ (1.27) нечіткої моделі (1.25)-(1.26). Тоді ми приходимо до оптимізаційної моделі

$$\max_{x \ge 0} \lambda \quad (1.33)$$

за умови

$$\lambda d_i + \sum_{j=1}^{n} B_{ij} x_j \le p_i + d_i, \quad i = \overline{1, m+1}, \ x \ge 0. \quad (1.34)$$

Якщо записати оптимальний розв'язок задачі (1.33)-(1.34) у вигляді вектору (λ, x^O), то x^O буде розв'язком задачі максимізації (1.29) для моделі (1.25)-(1.26) за припущенням, що функція належності має вигляд (1.31).

Цей оптимальний розв'язок може бути знайдений шляхом розв'язування стандартної (чіткої) ЗЛП з додатковою змінною і додатковим обмеженням по відношенню до моделі (1.27). Дана схема дозволяє розглядати запропонований підхід як достатньо конструктивний алгоритмічно і досить ефективний з обчислювальної точки зору.

Далі розглянемо виробничу ситуацію з невизначеністю. Припустимо, що деяка виробнича фірма планує випуск різних виробів $x_1, ..., x_n$ на поточний період. Позначимо через c_j очікуваний прибуток на одиницю реалізованої продукції типу $j, j=\overline{1,n}$. Для виготовлення будь-якого з виробів використовуються ресурси $b_1,...,b_m$– виробничі потужності фірми, причому питомі витрати *i*–го ресурсу, $i=\overline{1,m}$, при виробництві одиниці продукції типу *j*, $j=\overline{1,n}$ складають a_{ij} одиниць. Необхідно знайти такий раціональний план випуску виробів кожного типу, що забезпечує максимальний прибуток фірми. Математична модель даної задачі при фіксованих відомих значеннях параметрів c_j, a_{ij}, $j=\overline{1,n}$ $,i=\overline{1,m}$, аналогічна стандартній задачі лінійного програмування виду:

$$\max \ \sum_{j=1}^{n} c_j x_j \tag{1.35}$$

при обмеженнях

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i=\overline{1,m}, \tag{1.36}$$

$$x \ge 0; x \in R^n.$$

Якщо значення параметрів задачі є випадковими величинами з відомими функціями розподілу $F(c_j)$, $F(a_{ij})$, то її можна розв’язати методами стохастичного програмування. Однак, на практиці ці параметри часто невідомі і для них можна визначити лише інтервали можливих значень. Задачу такого типу називають задачею з багатозначними коефіцієнтами. В її рамках не має сенсу говорити про максимізацію цільової функції, тому що значення цієї функції – не числа, а множини чисел. Як вихід, необхідно з’ясувати, яке відношення переваги на множині альтернатив породжує ця функція, а потім визначити, які вироби потрібно вважати раціональними в розумінні даного відношення переваги.

Наступним етапом на шляху деталізації та уточнення розглянутої моделі є опис параметрів задачі у формі нечітких чисел. В модель вноситься додаткова інформація у вигляді функцій належності нечітких величин. Ці функції можна розглядати як спосіб наближеного відображення експертом неформалізованого уявлення про реальну величину конкретного параметра. Значення функцій належності – це вагові коефіцієнти, які експерти приписують різним можливим значенням кожного параметра задачі.

Після такого уточнення можна перейти до постановки задачі нечіткого математичного програмування [79]. Задано лінійну модель вигляду

$$\sum_{j=1}^{n} \tilde{c}_j x_j \to \max \ , \tag{1.37}$$

в якій значення коефіцієнтів $\tilde{c}_j$ задано нечітко у формі нечітких підмножин заданих універсальних множин. Крім цього, задано обмеження

$$\sum_{j=1}^{n} \tilde{a}_{ij} x_j \le \tilde{b}_i \ , \quad i=\overline{1,m}, \ \ x_j \ge 0, \ \ j=\overline{1,n} \ , \tag{1.38}$$

причому значення коефіцієнтів $\tilde{a}_{ij}$, $\tilde{b}_i$ також описано у формі відповідних нечітких множин. Необхідно здійснити раціональний вибір розв’язку $x \in R^n$, який в деякому розумінні максимізує задану нечітко лінійну форму (1.37).

Назвемо таку постановку нечіткої задачі оптимізації задачею лінійного програмування з нечіткими параметрами і розглянемо окремі її варіанти.

1.2.3. Задачі лінійного програмування з нечіткими ресурсними обмеженнями у правій частині.

Розглянемо далі задачу лінійного програмування з функцією цілі (1.35) з нечіткими обмеженнями на ресурси у вигляді

$$\sum_{j=1}^{n} a_{ij} x_j \le \tilde{b}_i \ , \quad i=\overline{1,m}, \ \ x \ge 0; \ \ x \in R^n , \tag{1.39}$$

де праві частини обмежень (1.39) задаються у формі нечітких правих трикутних чисел $\tilde{b}_i = (b_i, b_i, b_i + b_i^0)$, $b_i^0 \ge 0$, $i=\overline{1,m}$, з відповідними функціями належності

$$\mu_{b_i}(s) = \begin{cases} 1, \quad s \le b_i, \\ (b_i + b_i^0 - s)/b_i^0, b_i < s \le b_i + b_i^0, \ \ i=\overline{1,m}. \\ 0, \quad s > b_i + b_i^0, \end{cases} \tag{1.40}$$

Тут допустимі відхилення $b_i^0 \geq 0$, $i=\overline{1,m}$, визначають величини граничних змін ресурсів моделі.

Таке формулювання не обмежує загальності вигляду оптимізаційних задач з нечіткими обмеженнями [52]. Дійсно, можна розглядати задачу лінійного програмування з нечіткими ресурсами у формі задачі оптимізації функції цілі (1.35) за наявності системи змішаних обмежень

$$\begin{aligned} &\sum_{j=1}^{n} a_{ij} x_j \geq \tilde{b}_i, \quad i=\overline{1,m_1}, \\ &\sum_{j=1}^{n} a_{ij} x_j \leq \tilde{b}_i, \quad i=\overline{m_1+1,m_2}, \\ &\sum_{j=1}^{n} a_{ij} x_j = \tilde{b}_i, \quad i=\overline{m_2+1,m}, \end{aligned} \tag{1.41}$$

де праві частини перших m_1 обмежень задаються лівими нечіткими трикутними числами $\tilde{b}_i=(b_i-b_i^0, b_i, b_i)$, $b_i^0 \geq 0$, $i=\overline{1,m_1}$, праві частини наступної групи обмежень – правими нечіткими трикутними числами $\tilde{b}_i=(b_i, b_i, b_i+b_i^0)$, $b_i^0 \geq 0$, $i=\overline{m_1+1,m_2}$, а праві частини останніх $m\text{-}m_2$ обмежень – нечіткими трикутними числами $\tilde{b}_i=(b_i-b_i^l, b_i, b_i+b_i^r)$, з допустимими відхиленнями $0 \leq b_i^l \leq b_i$, $b_i^r \geq 0$, $i=\overline{m_2+1,m}$.

Ця модель ЗЛП може бути переписана у формі (1.35), (1.39) шляхом заміни перших m_1 обмежень на обмеження $\sum_{j=1}^{n}(-a_{ij})x_j \leq -\tilde{b}_i$, $\tilde{b}_i=(-b_i,-b_i,-b_i+b_i^0)$, $i=\overline{1,m_1}$, а останніх $m\text{-}m_2$ обмежень – на систему обмежень $\sum_{j=1}^{n}(-a_{ij})x_j \leq -\tilde{b}_i$, $\tilde{b}_i=(-b_i,-b_i,-b_i+b_i^l)$; $\sum_{j=1}^{n} a_{ij} x_j \leq \tilde{b}_i$, $\tilde{b}_i=(b_i, b_i, b_i+b_i^r)$; $i=\overline{m_2+1,m}$. Таким чином, будемо вважати, що загальний вигляд задачі лінійного програмування з нечіткими ресурсними обмеженнями у правій частині задається моделлю (1.35), (1.39).

Оптимізаційна задача, що розглядається, може бути розв’язана як задача параметричного лінійного програмування [72]. Такий спосіб є універсальним, не завжди враховуючим специфіку поставленої задачі.

Використаємо підхід, викладений вище. Як і у попередніх випадках, для розв’язання ЗЛП (1.35), (1.39) необхідно провести її дефазифікацію. Для цього обчислимо оптимальні значення рівнів цільової функції Z_l і Z_u шляхом розв’язування двох задач лінійного програмування:

$$Z_l = \max_x \sum_{j=1}^{n} c_j x_j \tag{1.42}$$

за умови

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i\,, \;\; i = \overline{1,m},\; x \ge 0;\, x \in R^n, \tag{1.43}$$

та

$$Z_u = \max_x \sum_{j=1}^{n} c_j x_j \tag{1.44}$$

за умови

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i + b_i^0\,, \;\; i = \overline{1,m},\; x \ge 0;\, x \in R^n. \tag{1.45}$$

Нехай далі $L = \min(Z_l, Z_u)$, $U = \max(Z_l, Z_u)$. Нечітка множина оптимальних розв’язків ЗЛП (1.35), (1.39), що задана у R^n (позначимо його $\tilde{G}$), описується функцією належності вигляду

$$\mu_{\tilde{G}}(x) = \begin{cases} 1, & \sum_{j=1}^{n} c_j x_j \le L, \\ (U - \sum_{j=1}^{n} c_j x_j)/(U - L), & L < \sum_{j=1}^{n} c_j x_j < U, \\ 0, & \sum_{j=1}^{n} c_j x_j \ge U, \end{cases} \tag{1.46}$$

а нечіткі множини кожного обмеження (позначаємо їх $\tilde{F}_i$, $i = \overline{1,m}$) з (1.39) визначаються функціями належності

$$\mu_{\tilde{F}_i}(x)=\begin{cases}1, & \sum_{j=1}^{n} a_{ij}x_j \le b_i,\\ (b_i+b_i^0-\sum_{j=1}^{n} a_{ij}x_j)/b_i^0, & b_i<\sum_{j=1}^{n} a_{ij}x_j<b_i+b_i^0, \quad i=\overline{1,m}.\\ 0, & \sum_{j=1}^{n} a_{ij}x_j \ge b_i+b_i^0,\end{cases} \quad (1.47)$$

На основі означення нечіткого розв'язку Белмана-Заде [73], нечітка задача лінійного програмування (1.35), (1.39) запишеться у формі оптимізаційної задачі наступного вигляду: знайти значення параметра $\lambda \in [0,1]$, яке є розв'язком задачі лінійного програмування

$$\max_{x} \lambda \quad (1.48)$$

за наявності обмежень

$$\begin{aligned} &\mu_{\tilde{G}}(x) \ge \lambda,\\ &\mu_{\tilde{F}_i}(x) \ge \lambda, \quad i=\overline{1,m}.\\ &x \ge 0. \end{aligned} \quad (1.49)$$

Підставляючи (1.46) і (1.47) у (1.49), запишемо остаточний вигляд оптимізаційної задачі

$$\max_{x} \lambda$$

$$\begin{aligned} &\lambda(U-L)+\sum_{j=1}^{n} c_j x_j - U \le 0,\\ &\sum_{j=1}^{n} a_{ij}x_j \le b_i+b_i^0-\lambda b_i^0, \, i=\overline{1,m},\\ &x \ge 0, \, 0 \le \lambda \le 1, \end{aligned} \quad (1.50)$$

яка є класичною задачею ЛП.

1.2.4. Задачі лінійного програмування з нечіткими технологічними коефіцієнтами

Цікавим різновидом оптимізаційних задач зі змінним обсягом ресурсів є задача нечіткого лінійного програмування з нечіткими технологічними коефіцієнтами

$$\max_{x} \sum_{j=1}^{n} c_j x_j \tag{1.51}$$

за умови

$$\sum_{j=1}^{n} \tilde{a}_{ij} x_j \le b_i \,,\ i = \overline{1,m}, \tag{1.52}$$

$$x \ge 0;\ x \in R^n .$$

Тут $\tilde{a}_{ij}, i = \overline{1,m},\ j = \overline{1,n}$, - технологічні коефіцієнти задачі, які задаються у вигляді правих нечітких чисел $(a_{ij}, a_{ij}, a_{ij} + d_{ij}), i = \overline{1,m},\ j = \overline{1,n}$, з наступними функціями належності:

$$\mu_{a_{ij}}(s) = \begin{cases} 1, \quad s \le a_{ij}, \\ (a_{ij} + d_{ij} - s)/d_{ij}, a_{ij} < s < a_{ij} + d_{ij}, i = \overline{1,m}\ j = \overline{1,n}, \\ 0, \quad s \ge a_{ij} + d_{ij}, \end{cases} \tag{1.53}$$

де $d_{ij} > 0\ i = \overline{1,m}\ j = \overline{1,n}$. Для застосування схеми дефазифікації задачі необхідно обчислити оптимальні значення рівнів цільової функції Z_l і Z_u шляхом розв’язання двох задач лінійного програмування:

$$Z_l = \max_{x} \sum_{j=1}^{n} c_j x_j \tag{1.54}$$

за умови

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i \,,\ i = \overline{1,m}\ x \ge 0;\ x \in R^n, \tag{1.55}$$

та

$$Z_u = \max_{x} \sum_{j=1}^{n} c_j x_j \tag{1.56}$$

за умови

$$\sum_{j=1}^{n} (a_{ij} + d_{ij}) x_j \le b_i \quad i = \overline{1,m}\ x \ge 0;\ x \in R^n . \tag{1.57}$$

Припускаємо, що обидві задачі мають розв’язок. При цьому, значення цільової функції початкової задачі (1.51) лежатиме між значеннями Z_l і Z_u за умови, що значення технологічних коефіцієнтів розміщені між a_{ij} та $a_{ij}+d_{ij}$, $i=\overline{1,m}$, $j=\overline{1,n}$.

Нехай $L=\min(Z_l,Z_u)$, $U=\max(Z_l,Z_u)$. Як і в наведеному вище випадку, отримані значення L і U будуть відповідно нижньою і верхньою межами оптимальних значень цільової функції.

Нечітка множина оптимальних значень (позначимо її $\tilde{G}$), задана в R^n, описується функцією належності вигляду

$$\mu_{\tilde{G}}(x)=\begin{cases}1, & \sum_{j=1}^{n} c_j x_j < L,\\ (U-\sum_{j=1}^{n} c_j x_j)/(U-L), & L\le \sum_{j=1}^{n} c_j x_j < U,\\ 0, & \sum_{j=1}^{n} c_j x_j \ge U,\end{cases} \tag{1.58}$$

Нечітка множина кожного обмеження (позначимо їх $\tilde{F}_i$, $i=\overline{1,m}$) з (1.52) визначаються функціями належності

$$\mu_{\tilde{F}_i}(x)=\begin{cases}1, & \sum_{j=1}^{n} a_{ij}x_j \le b_i,\\ (\sum_{j=1}^{n}(a_{ij}+d_{ij})x_j-b_i)/\sum_{j=1}^{n} d_{ij}x_j, & \sum_{j=1}^{n} a_{ij}x_j < b_i < \sum_{j=1}^{n}(a_{ij}+d_{ij})x_j,\ i=\overline{1,m}.\\ 0, & \sum_{j=1}^{n}(a_{ij}+d_{ij})x_j \le b_i,\end{cases} \tag{1.59}$$

Використовуючи означення нечіткого розв’язку в розумінні Белмана-Заде [73], нечітка задача лінійного програмування (1.51)-(1.52) перепишеться у формі оптимізаційної задачі:

$$\max_x \lambda \tag{1.60}$$

за умови

$$\begin{aligned}&\mu_{\tilde{G}}(x)\ge\lambda,\\ &\mu_{\tilde{F}_i}(x)\ge\lambda,\quad i=\overline{1,m}.\\ &x\ge 0.\end{aligned} \tag{1.61}$$

Підставляючи (1.58) і (1.59) в (1.61), запишемо остаточний вигляд оптимізаційної задачі

$$\max_x \lambda$$

$$\begin{aligned} &\lambda(U-L)+\sum_{j=1}^{n} c_j x_j - U \le 0, \\ &\sum_{j=1}^{n} (a_{ij}+\lambda d_{ij}) x_j \le b_i,\ i=\overline{1,m}, \\ &x \ge 0,\ 0 \le \lambda \le 1. \end{aligned} \tag{1.62}$$

Необхідно відзначити, що обмеження задачі (1.62) містять добуток λx_j, $j=\overline{1,n}$, і тому не є опуклими. В цьому випадку, для знаходження розв’язку задачі (1.62) необхідно застосовувати методи, призначені для пошуку рішень неопуклих оптимізаційних задач.

1.3. Висновки по першому розділу.

У першому розділі розглянуто класи задач лінійного програмування як основних задач дослідження операцій. Детально викладено зміст задач про рюкзак, про складання розкладу та про вибір заявок. Наведено основні положення про нечіткі величини та способи їх формалізації. Викладено принцип формування нечіткого розв’язку Белмана-Заде нечітких задач лінійного програмування, розглянуто різні варіанти нечітких моделей ЗЛП (з нечіткими технологічними коефіцієнтами та ресурсними обмеженнями). Записано нечіткі оптимізаційні задачі та проаналізовано способи для знаходження нечітких розв’язків.

Проведено огляд властивостей та результатів аналізу методів розв’язання оптимізаційних задач планування, розміщення та розподілу ресурсів. Визначено методику розв’язування чітких і нечітких задач розподілу часового ресурсу як задач складання розкладу, яка застосовується далі у дисертаційній роботі.

РОЗДІЛ 2
МОДЕЛІ ТА МЕТОДИ РОЗВ'ЯЗУВАННЯ ЧІТКИХ ЗАДАЧ ОПТИМАЛЬНОГО РОЗПОДІЛУ ЧАСОВОГО РЕСУРСУ ЯК ЗАДАЧ СКЛАДАННЯ РОЗКЛАДУ

Задачі впорядковування робіт, що виконуються машинами, при використанні деяких ресурсів є задачами теорії розкладів (ТР). Дослідження задач ТР, а також методів і алгоритмів їх вирішення дозволяє отримати оптимальні або близькі до оптимальних результати.

2.1. Задачі оптимального розподілу часового ресурсу як задачі складання розкладів

Основними термінами теорії розкладів є наступні поняття: операція, робота, машина, фізична природа яких байдужа для даної теорії [34]. Істотними є тільки їх загальні властивості, незалежні від прикладного змісту.

Задача теорії розкладів вважається заданою, якщо визначені ті роботи і операції, що підлягають виконанню; кількість і типи машин, що виконують операції; порядок проходження машин; критерії оцінки отриманих розкладів.

Під моделлю в рамках теорії розкладів розуміється будь-яка математична модель керованого дискретного технологічного процесу, в якій визначено множину допустимих розкладів цього процесу [34]. Якщо при цьому визначено критерій, відповідно до якого одні розклади більш важливі, ніж інші, і ставиться задача визначити найоптимальнішій за даним критерієм розклад, то можна говорити про задачу ТР. В рамках однієї і тієї ж технологічної моделі можна ставити різні задачі, вибираючи різні критерії оптимальності. Причому ці задачі можуть бути різними за складністю їх розв'язання - як точного, так і наближеного.

Чітко обкреслити коло задач, якими повинна займатися ТР, неможливо. Проте, в самій ТР виділяють моделі статичні (у яких вибір розкладу ухвалюється за наявності повної інформації, відомої до початку вирішення задачі) і динамічні (в яких процес вирішення затягується на невизначений термін або розклад будується за частинами, відповідно до надходження або не надходження нової інформації.

Розглянемо приклад задачі складання розкладів. Є множина різних машин (наприклад, це можуть бути верстати у виробничому цеху на заводі, комп'ютерні робочі станції), які здійснюють деякі операції в роботах (виконують певні технічні операції з деталями, проводять цикл розрахунків на певній робочій станції). Відповідно, є сукупність технічних операцій або циклів розрахунків. Кожна робота складається з операцій, які мають певний встановлений порядок. Для кожної машини задано ті операції, які необхідно провести, але порядок не встановлений. Існують деякі обмеження: кількість машин для кожної роботи встановлено; одна машина у довільний момент часу може виконувати лише одну роботу. Термін виконання кожної конкретної операції зафіксований. Необхідно мінімізувати час виконання всіх операцій.

У даному дослідженні під поняттям задач теорії складання розкладів (ТР) розуміються тільки задачі класу MS (Machine Scheduling, теорія розкладів у вузькому сенсі).

2.1.1. Класифікація задач теорії розкладів залежно від характеристик машин, робіт, цільової функції.

У класичній класифікації задач складання розкладів [31,34] будь-яка задача може бути записана таким чином: $\alpha\,|\,\beta\,|\,\gamma$, де α - характеристики машин (табл. 2.1); β - характеристики робіт (табл. 2.2); γ - цільова функція задачі (табл. 2.3). Роботи складаються з операцій: $J_i = \{O_{i_1}, O_{i_2}, ..., O_{i_n}\}, i = \overline{1,n}$, n - кількість робіт. Операція O_{i_j} вимагає p_{ij} часу і може виконуватися на одній з машин множини $\mu_{ij} \subseteq \{M_1, ..., M_k\}$. Якщо $|\,\mu_{ij}\,| = 1, \forall i, j$, то отримуємо модель з приписами. Якщо $|\mu_{ij}| = k$, $\forall i, j$, то отримуємо модель з паралельними машинами.

Поле α складається з 2 частин $\alpha = \alpha_1\alpha_2$, де α_1 - характеристики машин. У цільових функціях через c_i позначимо час закінчення роботи J_i, $i = \overline{1,n}$. Тоді можна розглядати два типи цільових функцій, що мінімізуються:

$$f(c) = \max_i f_i(c_i), \qquad ()$$

$$f(c) = \sum_{i=1}^{n} f_i(c_i), \qquad ()$$

де $f_i(\cdot)$, $i = \overline{1,n}$, - деякі функції.

За умов задання послідовності виконання робіт і відсутності обмежень на ресурси вважається, що така задача має регулярний критерій якості, який полягає у найшвидшому завершенні робіт. Відзначають, що дана задача має оптимальну підструктуру. Зміст наявності такої підструктури полягає у припущенні, що для того, щоб щонайшвидше виконати *n* робіт, необхідно за мінімальний час обробити *n-1* роботу і вибрати мінімальний час для обробки *n*-ї роботи. Наявність обмежень на ресурси та відсутність обмежень на послідовність виконання робіт приводять до задач з нерегулярним критерієм. У цьому випадку довести, що оптимальне рішення задачі конструюється з оптимальних рішень допоміжних підзадач неможливо.

Запропонована класифікація теорії розкладів (див. табл.2.1, 2.2, 2.3) є далеко не вичерпною, оскільки постійно виникають нові задачі, що не вкладаються у наведені формулювання.

Особливості завдань ТР залежно від характеристик машин, робіт, цільової функції приведені на рис. 2.1. У лівій частині малюнка представлені можливі цільові функції ($C_{\max}$ - час закінчення всіх робіт, $L_{\max}$- запізнювання щодо директивних термінів, $\sum_{i=1}^{n} w_i c_i$ - зважена сума закінчення робіт), посередині – характеристики машин (SM - одна машина, PM - паралельні машини, JSP - робочий цех, FSP - потокова лінія, OSP - відкрита лінія), а справа характеристики робіт (s-batch, p-batch — розбиття робіт на групи, pmtn — дозволено переривання, prec — умова передування).

Таблиця 2.1

Значення α	Характеристики машин
$\alpha_1 = \varnothing$	Для кожної роботи задана машина для її виконання
$\alpha_1 = P$	Машини паралельні і однакові $p_{ij} = p_i$.
$\alpha_1 = R$	Машини паралельні, тривалість виконання работ довільна, але $p_{ij} = p_i / s_{ij}$
Якщо $\alpha_1 \in \{\varnothing, P, Q, R\}$	То $n_i = 1, \forall j$, тобто кожна робота складається рівно з однієї операції
$\alpha_1 = J$ (робочий цех)	У кожної операції своя машина $\lvert \mu_{ij} \rvert = 1$ і лінійний порядок виконання операцій $O_{i_1} \to O_{i_2} \to \ldots \to O_{i_{n_i}}$
$\alpha_1 = F$ (потокова лінія)	Машини впорядковані $M_1, \ldots, M_k$ і кожна робота проходить всі машини в цьому порядку, $n_i = k$ і $\mu_{ij} = M_j$, $\forall i$
$\alpha_1 = O$ (відкрита лінія)	Кожна робота складається з k операцій ($n_i = k$), але $\mu_{ij} = \{M_1, \ldots, M_k\}$ і на множині операцій немає умов передування
$\alpha_1 = X$ (змішаний цикл)	Поєднання J і O
$\alpha_1 = G$ (загальний випадок)	Довільний порядок передування на операціях (як в календарному плануванні (PS))
Якщо $\alpha_1 \in \{J, F, O, X, G\}$	То $n_i \geq 1$, тобто у кожної роботи може бути декілька операцій

Таблиця 2.2

Значення β	Характеристики робіт
$\beta_1 = pmtn$	Вирішуються переривання
$\beta_2 = prec$	Умови передування на множині робіт (ланцюги, дерева, мережі)
$\beta_3 = r_i$	Час надходження на обслуговування
$\beta_4 \in \{p_{ij} = 1; p_{ij} \in \{0,1\};$ $p_{ij} = p_{ij}(t),...\}$	Уточнення для часу виконання операцій
$\beta_5 = d_i$	Директивні терміни закінчення робіт
$\beta_6 = p - batching (s - batching)$	Роботи розбиваються на групи, і у кожній групі береться максимум (сума) тривалостей виконання робіт

Таблиця 2.3

Цільова функція	Формулювання цільової функції
$C_{\max} = \max\limits_{i=\overline{1,n}} c_i$	Час закінчення всіх робіт
$L_{\max} = \max\limits_{i=\overline{1,n}} (c_i - d_i)$	Запізнювання щодо директивних термінів
$D_{\max} = \max\limits_{i=\overline{1,n}} \lvert c_i - d_i \rvert$	Відхилення від директивних термінів
$F_{\max} = \max\limits_{i=\overline{1,n}} (\max\{0; d_i - c_i\})$	Випередження директивних термінів
$\sum\limits_{i=1}^{n} w_i c_i$	Зважена сума закінчення робіт

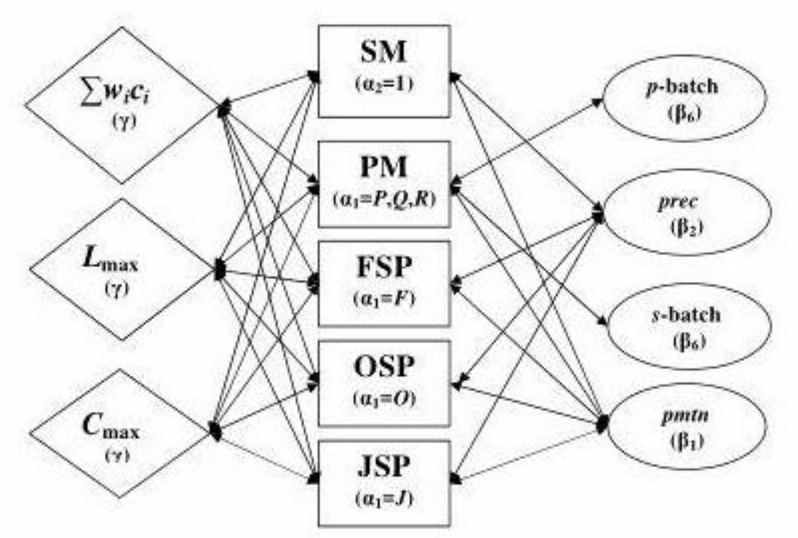


Рис. 2.1. Особливості задач ТР у залежності від характеристик машин, робіт, цільової функції

Задачі типу JSP згідно даної класифікації розглядається як задачі з дозволеним перериванням і цільовою функцією - часом закінчення всіх робіт. Загальна задача JSP розмірністю *nxk*, розв’язком якої є мінімізація тривалості розкладу, задається символами $J \mid pmtn \mid C_{\max}$ і може бути описана множиною робіт $\{J_i\}, i=\overline{1,n}$, які проводяться на множині *k* машин $\{M_r\},\ r=\overline{1,k}$.

Дана задача може бути сформульована з урахуванням наступних умов і обмежень: кожна робота повинна здійснюватися на конкретній машині згідно технологічної послідовності; кожна машина може у довільний момент часу здійснювати лише одну роботу; здійснення роботи J_j на машині M_r називається операцією O_{jr}, операція O_{jr} тривалістю p_{jr} передбачає неперервне використання машини M_r протягом всього виконання операції; стартовий час і час виконання операції O_{jr} задаються як s_{jr} і c_{jr}, відповідно; розклад - це множина моментів часу виконання для кожної операції $\{C_{jr}\}, j=\overline{1,n}, r=\overline{1,k}$, які задовольняють даним вимогам; час, необхідний для виконання всіх робіт називається тривалістю розкладу, він визначається як $C_{\max}$, тобто $C_{\max} = \max\limits_{j=\overline{1,n};r=\overline{1,k}} C_{jr}$.

Технологічна послідовність машин може бути різною для кожної роботи згідно першої умови і послідовність робіт, здійснюваних на машинах, може бути також різною для кожної машини. Задана технологічна послідовність для кожної роботи може бути задана за допомогою матриці $\{T_{js}\}$, у якій $T_{js} = r$ відноситься до s-ої операції O_{jr} роботи J_i на машині M_r. Необхідно оптимізувати задачу розкладу по знаходженню мінімального значення $C_{\max}$.

2.1.2. Математичні постановки оптимізаційних задач розподілу часового ресурсу.

Розглянемо *m* виконавців (машин), що володіють різною продуктивністю. Припустимо, що задано сукупність робіт, які згруповано в *p* типів (без обмеження можна вважати задано *p* робіт). Кожен виконавець $k \in \{1,...m\}$ може виконувати будь-яку роботу, але для виконання роботи *i*-го типу виконавці можуть витрачати різний час. Цей факт враховується технологічною матрицею

$$T = \begin{pmatrix} t_{11} & t_{12} & \dots & t_{1m} \\ t_{21} & t_{22} & \dots & t_{2m} \\ \dots & \dots & \dots & \dots \\ t_{p1} & t_{p2} & \dots & t_{pm} \end{pmatrix} \tag{2.1}$$

де t_{ik} – час виконання роботи типу *i* виконавцем *k*. Необхідно розподілити *p* робіт по *m* виконавцям (скласти план робіт) так, щоб загальний час виконання робіт, що обчислюється з початку роботи першого виконавця і закінчується за фактом звільнення всіх виконавців, був мінімальним. Відповідно до нотації, описаної в [37], сформульовану задачу позначимо у вигляді *Rm*||*Cmax*.

У класичному випадку, де $t_{ik} \in R$, задачу можна звести до задачі дискретного програмування наступного вигляду

$$\max_k \sum_{i=1}^{p} c_{ik} t_{ik} \to \min$$

$$\sum_{k=1}^{m} c_{ik} = n_i, \; i = \overline{1,p}, \tag{2.2}$$

$$c_{ik} \ge 0, i = \overline{1,p}, k = \overline{1,m},$$

де c_{ik} – кількість призначених робіт типу i для виконавця k, n_i – загальна (сумарна) кількість робіт типу i. Сукупність обмежень в задачі (2.2) гарантує, що всі роботи будуть залучені до розкладу. Розв'язок задачі (2.2) формується у вигляді матриці $C=\{c_{ik}\}_{i=\overline{1,p};k=\overline{1,m}}$.

Розглянемо математичні моделі оптимізаційних задач, що виникають при визначенні черговості виконання завдань з урахуванням або без урахування будь-яких додаткових обмежень і цільових функцій, які пов'язані з процесом обробки завдань.

Припустимо, що є множина $Z=\{1,2,...N\}$ з N задач (робіт або завдань, що підлягають виконанню), кожна з яких характеризується тривалістю рішення $T_i \in R_+$, $i=\overline{1,N}$. Будемо вважати, що виконання кожного завдання оцінюється відповідною кількістю балів B_i, $i=\overline{1,N}$. Можливо, що для перемикання з одного завдання на інше необхідний часовий проміжок (так званий час переналагодження) π_{ij}, $i=\overline{1,N}$ $j=\overline{1,N}$. $i \neq j$. Передбачається, що запасу часу $\bar{T}$ недостатньо для виконання всіх завдань, тобто $\sum_{i=1}^{N} T_i \leq \bar{T}$, виконання завдань проводиться одним виконавцем (людиною, машиною, арифметико-логічним пристроєм).

Задача максимально ефективного використання часу $\bar{T}$ розглядається як одна з класичних задач розподілу (резервування) ресурсів заданого об'єму за множиною категорій (робіт). Постановка такої задачі полягає в знаходженні плану витрат наявного ресурсу (у нашому випадку таким ресурсом є час $\bar{T}$) на виконання деякої сукупності з N завдань, при якому сумарне (ітогове) використання ресурсу є оптимальним.

Для визначення порядку виконання робіт, проведення яких забезпечує максимальну часову ефективність, можна сформулювати задачу у вигляді оптимізаційної задачі про рюкзак спеціального типу. Розглядатимемо розв'язання задачі оптимізації ітогової оцінки у часі.

Визначення найкращого підсумкового часу проведення сукупності робіт будемо здійснювати шляхом знаходження перестановки $p=(p_1,..., p_m)$, $m \le N$, $p_i \in \overline{1,N}$, $i=\overline{1,m}$, $p_i \ne p_j, i \ne j$, що визначає такий порядок виконання робіт, за яким досягається оптимальне значення в задачі оптимізації вигляду

$$W(p)=\sum_{i=1}^{m} t_{p_i} x_{p_i}(s_{p_i}) \to \max_{p=(p_1,...,p_m)}, \tag{2.3}$$

$$\sum_{i=1}^{k} t_{p_i} x_{p_i}(s_{p_i}) \le \overline{T}, \; k=\overline{1,m}, \tag{2.4}$$

де через $x_{p_i}(s)=\begin{cases} 1, \; s_{p_{i-1}} < s \le s_{p_i}, \\ 0, \; інакше, \end{cases}$, $i=\overline{1,m}$, позначено бульові змінні, одиничне значення кожної з яких відповідає часу виконання роботи p_i, $i=\overline{1,N}$, протягом часового інтервалу $s_{p_{i-1}} < s \le s_{p_i}$, $i=\overline{1,m}$, а нульові – в усіх інших випадках; $s_{p_i}=\sum_{j=0}^{i} t_{p_j}$, $i=\overline{1,m}$, $t_{p_0}=0$, $s_{p_0}=0$, $s_{p_m} \le \overline{T}$; $t_{p_i} \in \{T_1,...,T_N\}$, $i=\overline{1,m}$, та $\neg\exists k$: $t_{p_i}=T_k$, $t_{p_j}=T_k$, $p_i \ne p_j, i \ne j$, а підсумкова оцінка визначатиметься оптимальним значенням $W^o(p)$ цільової функції (2.3).

У такій постановці вважається, що пошук перестановки $p=(p_1,..., p_m)$ здійснюється у вигляді послідовності етапів, на кожному з яких обирається та виконується одна з робіт, що залишаються нерозв'язаними. Перехід від одного етапу до іншого характеризується також модифікацією обмеження: для $k=1$ воно має вигляд $t_{p_1} x_{p_1}(s_{p_1}) \le \overline{T}$, для $k=2$: $t_{p_2} x_{p_2}(s_{p_2}) \le \overline{T} - t_{p_1}$, …, для $k=m$: $t_{p_m} x_{p_m}(s_{p_m}) \le \overline{T} - \sum_{i=0}^{m-1} t_{p_i}$.

Традиційний підхід до відшукання оптимального значення функції $W(p)$ вигляду (2.3) з урахуванням обмеження (2.4) зводиться до повного перебору різних можливих перестановок $p=(p_1,..., p_m)$ з метою визначення такої з них, на якій досягається оптимум задачі

$$\sum_{i=1}^{m} t_{p_i} x_{p_i} \to \max_{p} \tag{2.5}$$

за умови виконання обмеження

$$\sum_{i=1}^{m} t_{p_i} x_{p_i} \le \overline{T} \ . \qquad (2.6)$$

Для випадку визначення робіт, виконання яких забезпечує максимальну бальну оцінку, можна сформулювати задачу у вигляді оптимізаційної задачі про рюкзак. Як і у попередньому випадку, позначимо через $x_i \in \{0,1\}$, $i = \overline{1,N}$, - бульові змінні, одиничні значення яких відповідають виконанню завдання, а нульові – його невиконанню. Тоді задача оптимізації підсумкової оцінки при виконанні сукупності робіт, має вигляд

$$W = \sum_{i=1}^{m} B_{p_i} x_{p_i} \to \max_{p} \qquad (2.7)$$

$$\sum_{i=1}^{m} T_{p_i} x_{p_i} \le \overline{T} \ , \qquad (2.8)$$

а підсумкова оцінка визначатиметься оптимальним значенням W^o цільової функції (2.7).

2.1.3. Формулювання загальної задачі складання розкладу в термінах лінійного цілочисельного програмування.

Нехай маємо систему з n робіт і m машин. Кожна робота складається з g_i операцій, $i = \overline{1,n}$. Кожній операції приписано три індекси:

i - номер роботи, що містить цю операцію;

j - номер операції всередині роботи, $j = \overline{1, g_i}$;

k - номер машини, на якій операція повинна виконуватися.

Обмеження на час і порядок виконання операцій машинами такі:

- кожна машина виконує одночасно не більше однієї операції;
- операції виконуються у зазначеній послідовності;
- ніякі дві операції, що відносяться до однієї роботи, не виконуються одночасно.

Будемо вважати, що кожна робота вимагає в точності одного виконання на кожній з машин.

Нехай t_{ik} - тривалість виконання роботи i машиною k;

$$r_{ijk} = \begin{cases} 1, \textit{якщо операція } j \textit{ роботи } i \textit{ виконується машиною } k; \\ 0, \textit{в іншому випадку} \end{cases}$$

$\underline{t}_{ik}$ - момент початку виконання роботи i машиною k (дорівнює часу початку виконання відповідної операції роботи i машиною k).

<u>Група обмежень 1</u>. З того, що кожна машина у довільний момент часу може виконувати не більше однієї роботи, випливає, що для кожної пари робіт I та J справедлива лише одна з нерівностей:

$\underline{t}_{Ik} - \underline{t}_{Jk} \geq t_{Jk}$ - виконанню роботи I передує виконання роботи J,

або (2.9)

$\underline{t}_{Jk} - \underline{t}_{Ik} \geq t_{Ik}$ - виконанню роботи J передує виконання роботи I.

Обмеження (2.9) типу «або-або» не можна описати в термінах традиційних задач лінійного програмування і потребує введення цілочисельних змінних.

Позначимо $Y_{IJk} = \begin{cases} 1, \textit{якщо робота } I \textit{ передує роботі } J \textit{ на машині } k; \\ 0, \textit{в іншому випадку} \end{cases}$

Тепер сформульовані вище обмеження типу «або-або» можна записати у вигляді двох умов, кожна з яких має бути виконана:

$$(M + t_{Jk})Y_{IJk} + (\underline{t}_{Ik} - \underline{t}_{Jk}) \geq t_{Jk} \qquad (2.10)$$

$$(M + t_{Ik})(1 - Y_{IJk}) + (\underline{t}_{Jk} - \underline{t}_{Ik}) \geq t_{Ik} \qquad (2.11)$$

де $M = \sum_{i}\sum_{k} t_{ik}$ - досить велика константа, обрана таким чином, щоб виконувалася тільки одна з двох умов: $Y_{IJk} = 0$ або $Y_{IJk} = 1$.

Нехай I передує J, тобто $\underline{t}_{Ik} < \underline{t}_{Ik}$ і $Y_{IJk} = 1$, тоді (2.11) в точності збігається з другою умовою «або-або» в (2.9), а умова (2.10) завдяки великому параметру M перетворюється в надлишкове обмеження, що не суперечить всій системі в цілому.

Група обмежень 2. Забезпечує дотримання обмежень на порядок виконання операцій.

Зауважимо, що $\sum_k r_{ijk} \underline{t}_{ik}$ - момент початку виконання операції j роботи i.

Тоді для всіх операцій кожної роботи, повинна мати місце нерівність

$$\sum_k r_{ijk}(\underline{t}_{ik} + t_{ik}) \le \sum_k r_{i(j+1)k} \underline{t}_{ik} \tag{2.12}$$

Цільові функції можуть бути різними. Так, мінімізація сумарного часу завершення робіт рівносильна мінімізації суми моментів початку виконання останніх операцій всіх робіт:

$$\sum_i \sum_k r_{imk} \underline{t}_{ik} \to \min .$$

При мінімізації максимального часу завершення робіт додається обмеження виду

$$\sum_k r_{imk}(\underline{t}_{ik} + t_{ik}) \le T_{\max}, i = \overline{1, n},$$

де $T_{\max}$ - змінна, яку потрібно мінімізувати.

При застосуванні методів математичного програмування для вирішення задач теорії розкладів необхідно враховувати *NP*-повну складність розв'язання задачі, що неминуче призводить до експоненціальності величини часу отримання оптимального розв'язку.

2.1.4. Задача складання розкладу виконання заданої кількості робіт для однієї машини.

Припустимо, що кожна робота складається тільки з однієї операції. В цьому випадку множину робіт можна розбити на групи в залежності від виду операції і кожна машина, що виконує певну операцію, не залежить від інших. Отже, можна обмежитися складанням розкладу тільки для однієї машини і підмножиною робіт, яка на ній виконується.

Вважатимемо, що

* кількість робіт є скінченою і відома наперед, всі роботи повинні бути виконані;

* машини використовуються тільки для виконання даних робіт, що вони завжди доступні і не виходять з ладу;

* роботи поступають в систему одночасно, так що при складанні розкладу процес обслуговування може початися з будь-якою з них;

* виконання кожної з робіт відбувається або без налагодження машини, або налагодження не залежить від попередньої роботи. У останньому випадку тривалість налагодження перед будь-якою роботою залежить лише від самої роботи і цю тривалість можна приєднати до тривалості виконуваної роботи.

Для подальшої роботи, приймемо наступні позначення:

$m=1$, тобто є одна машина;

$g_i=1,\ i=\overline{1,n}$, оскільки за припущенням кожна робота складається з однієї операції;

$r_i=0,\ \ i=\overline{1,n}$, тобто всі роботи отримуються одночасно, тому, без обмеження загальності, початок відліку на часовій вісі можна сумістити з моментом надходження робіт;

$t_{i1}=t_1,\ \ i=\overline{1,n}$, - тривалість роботи обирається довільною величиною, що задається наперед і незалежною від розкладу. Оскільки кожна робота складається з однієї операції, то другий індекс опускається;

$a_i=d_i,\ \ i=\overline{1,n}$, - допустима тривалість проходження співпадає з плановим терміном, оскільки $d_i=r_i+a_i=0+a_i$;

$W_{i1}=W_i,\ i=\overline{1,n}$, - тривалість очікування роботою i початку її виконання;

$T_i=F_i=t_i+W_i,\ \ i=\overline{1,n}$, - момент закінчення роботи дорівнює тривалості виконання, оскільки відлік часу починається з моменту надходження роботи.

Результати впорядкування для однієї машини використовуються у випадках:

* коли складні технічні (технологічні) комплекси функціонують *як одне ціле*;

* коли з сукупності операцій кожної роботи одна (що виконується на певній машині) є *домінантною*; у таких ситуаціях другорядними операціями можна нехтувати і розглядати систему такою, що складається з однієї машини, яка здійснює цю домінантну операцію;

* коли одна з машин тимчасово стає настільки *вузьким місцем* в системі, що в основному визначає ефективність роботи усієї системи. Для цієї машини розклад складається окремо і незалежно від інших машин.

У загальному випадку корисними можуть виявитися розклади, що допускають переривання (коли робота, що виконується, переривається до її завершення і робота знімається з машини), і простої, що штучно вводяться (коли машина простоює за наявності роботи, що чекає на виконання). Відповідно до твердження [37] для системи n/1 такі розклади можна не розглядати.

Отже, пошук оптимального розкладу має проводиться в класі перестановочних розкладів, тобто розкладів, які повністю визначаються порядком виконання робіт. У разі n робіт маємо $n!$ можливих перестановок їх номерів. Задаючи перестановку і тривалість кожної роботи, можна обчислити значення W_i, $i=\overline{1,n}$, а потім знайти решту властивостей розкладу.

Позначимо через $p_i=l$, $i=\overline{1,n}$, порядковий номер роботи з номером l на i-у етапі виконання перестановочного розкладу. Наприклад, p_1 означає номер роботи, яка виконується першою. Нехай також, t_{p_i}, $i=\overline{1,n}$, визначає тривалість роботи, що виконується на p_i-му етапі (на етапі з номером p_i, $i=\overline{1,n}$).

Зауваження. Нескладно помітити, що для перестановочних розкладів в системі $n/1$ максимальна тривалість виконання дорівнює сумі тривалостей n робіт і однакова для всіх $n!$ можливих впорядкувань. У складніших системах максимальна тривалість проходження часто обирається у якості основного критерію оцінки якості розкладів.

Розглянемо порядок виконання робіт такий, що: $t_{p_1} \le t_{p_2} \le \ldots \le t_{p_n}$.

Такий алгоритм називатимемо *впорядкуванням за критерієм мінімуму тривалості робіт* (shortest processing time sequencing) або, скорочено, SPT. Цей важливий алгоритм в різних варіантах неодноразово зустрічається в ТР.

Оптимальність впорядкування SPT встановлюється наступною теоремою.

Теорема [37]. Середній час перебування робіт в системі n/1 мінімально, якщо після впорядкування тривалості робіт не убувають:

$$t_{p_1} \le t_{p_2} \le \ldots \le t_{p_n},$$

і максимально, якщо після впорядкування тривалості робіт не зростають:

$$t_{p_1} \ge t_{p_2} \ge \ldots \ge t_{p_n}.$$

2.1.5. Формулювання задачі оптимальної рекомбінації робіт.

Якщо враховувати час переналагодження при перемиканні з однієї роботи на іншу, стає важливим порядок виконання робіт. Тоді необхідно розглядати наступні випадки: 1) задається час, що обмежує виконання всіх завдань; 2) час виконання завдань не враховується.

Нехай задана множина робіт $Z = \{1,\ldots,N\}$, що мають бути виконані на єдиній машині. Кожна робота $i \in V$ характеризується тривалістю $t_i \in R_+$, $i = \overline{1,N}$, в межах якої виконання роботи не переривається. Якщо машина, на якій виконуються роботи, переходить від виконання однієї роботи на іншу, то необхідно виконати переналагодження. Будемо вважати, що задано тривалість переналагодження $\pi_{ij} \in R_+$ з роботи *i* на роботу *j*, для усіх $i, j \in V$, $i \ne j$, $i, j = \overline{1,N}$.

У випадку, коли задано час $\overline{T}$, що обмежує виконання всіх завдань, розглядаються задачі про максимальне використання часу $\overline{T}$ і про оптимізацію підсумкової бальної оцінки. Облік інтервалів переналагодження ускладнює отримувані задачі, що приводить до необхідності використання методів комбінаторної оптимізації. Наприклад, неважко відмітити, що розв'язком задачі оптимального використання часу $\overline{T}$ буде перестановка $p = (p_1,\ldots,p_m)$, $m \le N$,

$p_i \in \overline{1,N}$, $i=\overline{1,m}$, що визначає порядок виконання робіт і яка є розв'язком оптимізаційної задачі, аналогічної (2.5)-(2.6):

$$W=\sum_{i=1}^{m-1}(t_{p_i}+\pi_{p_i p_{i+1}})x_{p_i p_{i+1}}+t_{p_m}\to\max_p \tag{2.13}$$

$$\sum_{i=1}^{m-1}(t_{p_i}+\pi_{p_i p_{i+1}})x_{p_i p_{i+1}}+t_{p_m}\le\overline{T}\ , \tag{2.14}$$

$x_{ij}\in\{0,1\}$ - змінні, що характеризують порядок виконання робіт (при одиничному значенні робота i виконується безпосередньо перед роботою j, в решті випадків - 0), $i,j=\overline{1,N}, i\neq j$.

Задача оптимізації підсумкової оцінки при виконанні сукупності робіт, аналогічна оптимізаційній задачі про рюкзак (2.7) -(2.8), матиме вигляд

$$W=\sum_{i=1}^{m}B_{p_i}y_{p_i}\to\max_p \tag{2.15}$$

$$\sum_{i=1}^{m-1}(t_{p_i}+\pi_{p_i p_{i+1}})x_{p_i p_{i+1}}+t_{p_m}\le\overline{T}\ , \tag{2.16}$$

де $y_i=\begin{cases}1, якщо\, x_{ij}=1,\\ 0, інакше,\end{cases}$, $i=\overline{1,N}$, а підсумкова оцінка визначається за сукупністю виконаних робіт.

Зупинимося детальніше на випадку, коли час виконання завдань не враховується. Вимога повного відпрацювання всіх завдань означає обов'язкове виділення часу на їх виконання і, отже, в оптимізації часу завершення всіх робіт необхідно враховувати лише тривалість переналагоджень. Необхідно скласти розклад, що мінімізує час завершення усіх робіт. Така задача називається задачею оптимальної рекомбінації.

Позначивши через $p=(p_1,...,p_N)$, $p_k\in\overline{1,N}$, $k=\overline{1,N}$, перестановку, що визначає порядок виконання робіт (значення p_i визначає роботу, яка виконується i-ю), отримаємо задачу пошуку перестановки, за якою мінімізується сумарна тривалість переналагоджень

$$l(\pi) = \sum_{i=1}^{N-1} \pi_{p_i p_{i+1}} . \qquad (2.17)$$

2.2. Методи розв'язання задач оптимального розподілу часового ресурсу

2.2.1. Властивості методів розв'язання оптимізаційних задач.

■ NP - повна складність розв'язання оптимізаційних задач.

При застосуванні методів математичного програмування для вирішення оптимізаційних задач комбінаторного типу необхідно враховувати NP-повну складність розв'язання задачі, що неминуче призводить до експоненційної оцінки величини часу отримання оптимального розв'язку.

Складність розв'язування оптимізаційних задач, як правило, суттєво залежить від їх розмірності n. Зрозуміло, що в задачі про рюкзак n - це число предметів, в задачі комівояжера n - число міст. Від цього параметра залежить час роботи $T(n)$ алгоритмів розв'язання задач, який задається деякою функцією $f(x)$ або її оцінками.

Вважається, що $f(x) = O(p(x))$, якщо існують константа $c > 0$ та величина x_0 такі, що для всіх $x > x_0$ виконується:

$$f(x) \le c \cdot p(x)$$

Функцію $p(x)$ називають оцінкою зверху для функції $f(x)$. Аналогічно можна ввести і нижню оцінку. Будемо говорити, що $f(x) = \Omega(q(x))$, якщо існують константа $c > 0$ і величина x_0, що для всіх $x > x_0$ виконується:

$$f(x) \ge c \cdot q(x) .$$

Функція $f(x)$ може бути поліномом n-го ступеня, експонентою e^x, факторіалом $x!$ і т.п. При великих значеннях x величина факторіалу зростає швидше за експоненту, значення експоненти - швидше за будь-який поліном, поліном більшого ступеня - швидше за поліном меншого ступеня. При цьому, важливою є оцінка поведінки функції $f(x)$ при великих значеннях x. Якщо задача має експоненційну складність, точний розв'язок її можливий лише для невеликих значень x. Тому найбільший прикладний інтерес має знаходження поліноміальних алгоритмів її розв'язання або доведення того, що для даної задачі таких алгоритмів немає. Якщо існує поліноміальний алгоритм, який розв'язує одну із задач деякого класу, і будь-яка інша задача цього класу може бути перетворена в ту, що розв'язується, за поліноміальний час, то вважається, що для цієї задачі існує алгоритм заданої складності. Таким чином, на відміну від задач класу P, для яких може бути знайдений поліноміальний алгоритм (ступінь поліному довільна - задача зі складністю $n^{1000000}$ поліноміальна!), та задач класу E, котрі не розв'язуються менш ніж за експоненційний час, формулюється клас NP – задач, що розв'язуються "недетермінованими машинами Тьюрінга" за поліноміальний час.

Якщо розв'язування задачі з класу NP може бути зведене до розв'язання задачі з конкретного класу, то такий клас задач називається NP-складним (тобто ці задачі не менш складні, ніж NP). За умови, що отримані задачі також належать класу NP, говорять про NP-повні задачі. Отже, задача називається NP-повною, якщо вона належить класу NP і будь-яка інша задача з NP зводиться до неї за поліноміальний час.

NP - повна складність розв'язування задачі є вагомим аргументом при обґрунтуванні необхідності побудови наближених або евристичних алгоритмів її розв'язку, важливості використання схем спрямованого перебору варіантів, а також при обґрунтуванні необхідності дослідження окремих випадків задачі.

До найбільш широко відомих прийомів скорочення перебору відносяться прийоми, побудовані на неповному переборі. Ці прийоми базуються на побудові «часткових рішень», представлених у вигляді дерева пошуку та застосуванні методів побудови оцінок, що дозволяють відсікати безперспективні часткові рішення. Однак, навіть досконалі прийоми скорочення перебору не дозволяють звільнитися від експоненційної складності розв'язання сформульованих задач теорії розкладів.

■ Методи локального пошуку (жадібні алгоритми).

Для розв'язку багатьох задач є прості і швидкі алгоритми, що дозволяють отримати наближений розв'язок задачі. Один з варіантів реалізації наближених алгоритмів розв'язання задачі оптимізації може бути створений на основі застосування жадібного підходу. Схема жадібних алгоритмів ґрунтується на введенні додаткових умов, які й визначають порядок вибору розв'язку. Жадібний алгоритм робить на кожному кроці локально оптимальний вибір і надалі цей вибір не відміняється.

Розглянемо приклад застосування жадібного алгоритму для вирішення задачі про вибір заявок (див. п.1.1.3). Будемо вважати, що заняття впорядковані за часом їх закінчення: $\beta_1 \leq ... \leq \beta_n$.

Ідея алгоритму полягає у наступному. Оберемо першу заявку (для неї буде мінімальним час закінчення заняття). Потім переглядаємо послідовно всі заявки, починаючи з другої, і обираємо ті, у яких час початку не менше, ніж час закінчення останньої обраної заявки.

Іншими словами, задача оптимального вибору заявок розбивається на сукупність підзадач, розв'язки яких на кожному кроці формують кінцевий розв'язок вихідної задачі.

Таким чином, остаточно маємо, що оптимальний розв'язок містить оптимальні розв'язки усіх підзадач. При цьому відмітимо, що запропонований алгоритм має поліноміальну часову складність роботи.

Звернемо увагу на іншу властивість жадібного вибору. На кожному кроці розглянутого алгоритму необхідно дотримуватися певної стратегії - серед відповідних заявок обираємо ту, час закінчення заняття якої найменший. Можна довести, що саме такий спосіб збільшення числа взятих заявок на кожному кроці призводить до отримання оптимального розв'язку. Іншими словами, при таких "жадібних" відборах не відсікаються можливі шляхи для пошуку оптимального розв'язку.

Жадібні алгоритми застосовуються для розв’язання багатьох прикладних задач і можуть бути основою різних евристичних алгоритмів, в тому числі алгоритмів розв'язання задач лінійного розкрою, про рюкзак, розкладу та інших.

■ Евристичні методи.

Розглянемо алгоритми, що дають наближені (неточні) розв’язки задач. В основі таких евристичних алгоритмів лежать ідеї, побудовані на інтуїції, життєвому досвіді. Для задач оптимізації вони можуть давати далеко не оптимальні розв’язки. Отже, якщо потрібно знайти точний розв’язок задачі, ці методи не застосовуються. В той самий час їх просто зрозуміти та легко запрограмувати. Евристичні алгоритми зазвичай мають невелику складність обчислень і часто дозволяють отримати близькі до оптимальних розв’язки. Тому їх часто застосовують до розв’язування складних задач.

2.2.2. Застосування методу динамічного програмування для розв’язання задач теорії розкладів.

Для розв’язання деяких класичних задач теорії розкладів може застосовуватися метод динамічного програмування.

■ Застосування методу для складання розкладу виконання робіт за наявності регулярного критерію.

Розглянемо задачу складання розкладу виконання послідовності дій (операцій, робіт) на прикладі простого конвеєра.

Припустимо, що є дві виробничі лінії, на кожній з яких передбачено n робочих місць, пронумерованих від 1 до n. На кожній лінії на робочих місцях з однаковими номерами виконується один і той же тип операції. Проте, час виконання одних і тих самих операцій на різних лініях a_{ij} (час виконання операції i на лінії j) відрізняється. Часом переходу від одного робочого місця до іншого, якщо цей перехід виконується в межах однієї лінії, можна нехтувати.

Проте, якщо наступна операція повинна бути виконана на іншій лінії, необхідно витратити час t_{ij}, щоб перемістити виріб з лінії i на сусідню лінію після проходження операції j.

Завдання полягає в тому, щоб визначити, які робочі місця повинні бути вибрані на першій лінії, а які - на другій, щоб мінімізувати повний час, витрачений на виготовлення виробу.

Позначимо робоче місце j на лінії i через S_{ij}, а час обробки деталі на робочому місці S_{ij}, через a_{ij}.

Очевидно, що дана задача має регулярний критерій якості, який полягає у найшвидшому завершенні виготовлення виробу. Зрозуміло також, що дана задача має *оптимальну підструктуру.* Дійсно, адже для того, щоб щонайшвидше виготовити деталь на n лініях, необхідно за мінімальний час обробити її на n-1 лінії і вибрати мінімальний час для обробки на n-й лінії.

Для визначення оптимальної структури розв'язку необхідно відзначити, що якщо j-я операція повинна бути виконана на перший лінії, то справедливе одне з наступних тверджень:

- цей шлях проходить через робоче місце S_{1j-1}, після чого виріб потрапляє на робоче місце S_{1j};
- цей шлях проходить через робоче місце S_{2j-1}, після чого виріб перекидається з другої лінії на першу, а потім потрапляє на робоче місце $S_{1j.}$

Аналогічні висновки можна зробити для найшвидшого шляху, що проходить через робоче місце S_{2j}:

- цей шлях проходить через робоче місце S_{2j-1}, після чого виріб потрапляє на робоче місце S_{2j};
- цей шлях проходить через робоче місце S_{1j-1}, після чого виріб перекидається з першої лінії на другу, а потім потрапляє на робоче місце S_{2j}.

Якщо позначити через $f_i(j)$ мінімально можливий час, протягом якого виріб проходить від стартової позиції до робочого місця S_{ij}, то для величин $f_1(j)$ і $f_2(j)$ можна виписати наступні співвідношення:

$$f_1(j) = \min(f_1(j-1) + a_{1j},\ f_2(j-1) + t_{2j} + a_{1j}); \quad (2.18)$$

$$f_2(j) = \min(f_2(j-1) + a_{2j},\ f_1(j-1) + t_{1j} + a_{2j}). \quad (2.19)$$

Таким чином, для того, щоб визначити оптимальне робоче місце для *i*-ї операції, необхідно визначити оптимальне робоче місце для *i-1*-ї операції, а потім, використовуючи формули (2.18) і (2.19), визначити, яку саме з двох ліній використовувати на даному етапі.

Остаточно отримуємо, що при розв’язанні задачі виникає *n* допоміжних задачі і на кожному етапі необхідно перевірити лише два варіанти. Тому в даному випадку вибір динамічного програмування як метод розв’язання виправданий.

За наявності регулярного критерію дана задача має також такі властивості:

- відсутність обмежень на ресурси;
- мінімальна кількість можливих етапів виконання роботи.

Єдині обмеження, що мають місце в даній задачі, - це обмеження на послідовність виконання робіт.

Ці дві властивості є важливими для застосування методу динамічного програмування, оскільки перше з них робить можливим використання «висхідного напрямку» при вирішенні задачі, а друге - зменшує складність задачі. Дійсно, за наявності жорстких обмежень рух у висхідному напрямі при розв’язуванні задачі може привести до тупикової ситуації, оскільки вирішення деякої підзадачі на першому етапі ще не гарантує можливість подальшого просування у цьому напрямку. З іншого боку, обмеження на послідовність виконання робіт приводять лише до зменшення кількості варіантів, що перебираються. Все це є аргументами на користь застосування методу динамічного програмування.

■ <u>Загальне завдання складання розкладів з нерегулярним критерієм.</u>

Сформулюємо задачу складання розкладу у більш загальному вигляді, припускаючи, що оптимальний розв'язок у ній не може бути побудований на основі оптимальних рішень допоміжних підзадач.

Розглянемо m машин, які виконують n робіт. Задано час виконання кожної роботи на кожній машині. Припустимо, що є обмеження на ресурси (наприклад, на час роботи машин і т.д.). Необхідно скласти розклад роботи машин з урахуванням деякого нерегулярного критерію. До завдань такого класу відносяться завдання, в яких всі роботи повинні бути проведені у відведений час. При цьому швидкість проведення даних робіт ніяк не позначається на якості розкладу. Більш того, у ряді випадків рівномірний розподіл робіт у відведений відрізок часу є більш важливим, ніж «швидке» проведення або завершення всіх робіт.

Особливістю таких завдань є:

- наявність обмежень на ресурси;
- відсутність обмежень на послідовність виконання робіт;
- велика розмірність задачі (велике число робіт і машин);
- наявність нерегулярного критерію.

У попередньому випадку вже згадувалося, що відсутність обмежень на ресурси і відносно невелике число варіантів вибору роблять вибір методу динамічного програмування найбільш ефективним для розв'язання. У реальних задачах складання розкладу ці умови зустрічаються достатньо рідко. Навпаки, найбільшу складність вирішенню задач складання розкладів надає наявність досить жорстких обмежень і велика розмірність задачі. Відсутність обмежень на послідовність виконання робіт в значній мірі збільшує кількість варіантів, що перебираються. На відміну від попередньої задачі будь-яка перестановка двох робіт буде новим варіантом. Ще більшу складність мають задачі з нерегулярними критеріями. В даному випадку складність полягає у відсутності у таких задач (за рідкісним виключенням) оптимальної підструктури. В задачі з мінімальним часом завершення робіт необхідно, щоб роботи закінчувалися якнайскоріше на кожному етапі, що робить можливим визначити у неї оптимальної підструктури. У випадку з нерегулярним критерієм довести, що оптимальне рішення задачі конструюється з оптимальних рішень допоміжних підзадач часто неможливо. В цьому випадку абсолютно неясно, яку з робіт ставити в розклад першою, яку - другою і т.д. Всі ці особливості подібних задач роблять неможливим застосування методу динамічного програмування і вимагають пошуку нових методів і підходів для розв'язування задач складання розкладів.

2.2.3. Метод найменших відхилень розв'язання задачі розподілу ресурсів.

Розглянемо один з різницевих методів розв'язання обернених задач розподілу однорідних ресурсів, який має назву методу найменших відхилень [74].

Обернену задачу можна сформулювати так. Необхідно визначити кількість засобів і обрати такий спосіб їх розподілення за обсягами запитів (ресурсів), які обслуговуються, щоб заданий рівень ефективності виконання засобами поставленої задачі досягався при їх мінімальних витратах.

Нехай N - кількість завдань (робіт), які необхідно виконати, $T_i, i = \overline{1,N}$, - задані тривалості виконання цих завдань, $Q_j,\ j = \overline{1,M}$ - обсяги ресурсів, наявні у засобів реалізації завдань (час роботи). Кожен з засобів забезпечує виконання деякої сукупності робіт в межах заданого часу використання.

План розподілу засобів для обслуговування робіт будемо характеризувати матрицею $H = \{h_{ij}\},\ i = \overline{1,N},\ j = \overline{1,M}$, де h_{ij}=1, якщо j-й засіб виділяється для виконання i-ї роботи, і, відповідно, h_{ij}=0, у протилежному випадку.

Таким чином, необхідно знайти такі значення M^* та H^*, які є розв'язком задачі

$$\min_{H \in \mathrm{H}} (M(H)), \tag{2.20}$$

$$\sum_{j=1}^{M} h_{ij} Q_j \geq T_i,\ i = \overline{1,N}, \tag{2.21}$$

де Q_j - обсяг ресурсів j-го засобу, $j=\overline{1,M}$, Н - множина можливих планів розподілу засобів для обслуговування робіт, що задається обмеженнями у вигляді

$$\sum_{j=1}^{M} h_{ij} \le 1, \quad i=\overline{1,N}, \tag{2.22}$$

тобто виконання кожного завдання може здійснюватися за пропорційною участю всіх засобів обслуговування.

Для розв’язання задачі за таких умов може бути запропонований алгоритм, що базується на принципі послідовного використання ресурсів наявного засобу, за котрим на кожному кроці процесу оптимізації підтримується такий розподіл ресурсів, який забезпечує найменше відхилення від оптимального розв’язку.

Розглянемо вектор ефективності розподілу засобів $Q=\{Q_j\}$, $j=\overline{1,M}$. Зрозуміло, що можна сформувати деяку матрицю призначень $H^*=\{h_{ij}^*\}$, $i=\overline{1,N}$, $j=\overline{1,M}$, яка відповідає вектору Q і визначає мінімальну кількість засобів, залучених для виконання завдань, без урахування обмеження (2.22). Така матриця описує нереальний план розподілу засобів, тому що деякі засоби в ньому одночасно виділяються на виконання різних завдань. Однак, цей план дає нижню границю значень критерію якості розв’язку поставленої задачі:

$$M_- = \inf M(H) = \sum_{i=1}^{N}\sum_{j=1}^{M} h_{ij}^* . \tag{2.23}$$

Отримане співвідношення перепишемо у вигляді

$$M_- = \sum_{j=1}^{M} m_j^*; \quad m_j^* = \sum_{i=1}^{N} h_{ij}^*, \tag{2.24}$$

де m_j^* - мінімальний обсяг ресурсів (час роботи), який необхідно витратити j-му засобу, $j=\overline{1,M}$, для виконання сукупності робіт за умов врахування обмеження $\sum_{j=1}^{M} h_{ij} Q_j \ge T_i$, $i=\overline{1,N}$, та без врахування обмеження (2.22).

Остаточно отримуємо, що значення h_{ij}^*, а відповідно, і m_j^*, $j=\overline{1,M}$, можуть бути визначені шляхом розв'язання M таких задач

$$m_j^{*} \underset{h_{ij}}{\rightarrow} \min \tag{2.25}$$

за умов $\sum_{j=1}^{M} h_{ij} Q_j \geq T_i$, $i = \overline{1,N}$, $j = \overline{1,M}$.

При призначенні одного з засобів на кожному кроці процесу розподілу будемо використовувати принцип найменшого відхилення від нижнього значення критерію M_{-} складу засобів, який отримано з урахуванням застосування засобу i^* для виконання завдання j^*.

Це означає, що при реальному призначенні потрібно обирати елемент (i^*, j^*), який забезпечує виконання умови

$$(i^*, j^*):\ \min_{i,j} \Delta_{ij},\ i = \overline{1,N},\ j = \overline{1,M}, \tag{2.26}$$

де $\Delta_{ij} = \sum_{k=1}^{M} (m_k^{ij} - m_k^{*})$, $\Delta_{ij} \geq 0$, $i = \overline{1,N}$, $j = \overline{1,M}$, m_k^{ij} - мінімальний обсяг ресурсів k-го засобу, який залучається при виконанні завдання i засобом j.

Склад засобів, якій відповідає призначенню засобу i^* для виконання завдання j^*, характеризується планом розподілу $H^{i^* j^*}$ і вектором $\{m_k^{i^* j^*}\}$, $k = \overline{1,M}$, що знаходяться шляхом розв'язання M таких задач

$$m_k^{i^* j^*} \underset{h_{ij}}{\rightarrow} \min\ ,\ k = \overline{1,M}, \tag{2.27}$$

за умов $\sum_{j=1}^{M} h_{ij} Q_j \geq T_i$, $i = \overline{1,N}$, $h_{i^* j^*} = 1$, $h_{i^* j} = 0$, $j = \overline{1,M}$, $j \neq j^*$.

У випадку, коли виконання сукупності задач забезпечується одним наявним засобом (виконавцем, людиною, процесором і т.і.), розподіл ресурсів полягає у знаходженні послідовності проведення виконання робіт.

При цьому, потрібно зауважити, що задача (2.20)-(2.22) передбачає повне забезпечення необхідних для виконання сукупності завдань ресурсів (часу роботи) у наявних засобів обробки завдань. Іншими словами, передбачається, що обсягу ресурсів достатньо для виконання усіх завдань. У випадку наявності обмеження, за яким неможливо повне виконання поставлених завдань в межах встановленого часового проміжку $\overline{T}$, алгоритм дозволяє обчислити коефіцієнт пропорційного зменшення тривалостей робіт $T_i, i = \overline{1,N}$, при якому задача розподілу засобів для виконання заданих завдань буде мати розв'язок.

2.2.4. Реалізація жадібного підходу до розв'язування задач розподілу ресурсів.

Один з варіантів реалізації наближених алгоритмів розв'язку задач часового розподілу та рекомбінації може бути створений на основі застосування жадібного підходу. Схеми жадібних алгоритмів ґрунтуються на введенні додаткових умов, які й визначають порядок вибору розв'язку [75], а основу цих алгоритмів складають так звані "жадібні" евристики.

За схемою *жадібного евристичного алгоритму розв'язання задачі про рюкзак* (1.15), (1.16), (1.18) припускається, що предмети впорядковані по відношенню вартості до об'єму у наступному вигляді:

$$\frac{c_1}{v_1} \geq \frac{c_2}{v_2} \geq \ldots \geq \frac{c_n}{v_n}. \tag{2.28}$$

Запропонуємо наступний алгоритм. Спочатку набір предметів порожній. Будемо послідовно перебирати предмети з номерами $1, 2, \ldots, n$. Наступний предмет розміщуємо у рюкзак за умови, що його додавання в набір не призводить до перевищення місткості рюкзака.

Формально

$$x_1 = \begin{cases} 1, \textit{ якщо } v_1 \leq V, \\ 0, \textit{ інакше.} \end{cases}$$

Залишок об'єму рюкзака:

$$y_1 = V - v_1 x_1.$$

Решта змінних визначаються аналогічно:

$$x_j = \begin{cases} 1, \text{ якщо } v_j \le y_{j-1}, \\ 0, \text{ інакше.} \end{cases} \qquad j = \overline{2,n},$$

$$y_j = y_{j-1} - v_j x_j,$$

Продемонструємо це на прикладі. Припустимо, що є рюкзак об'ємом $V = 10$, який необхідно заповнити предметами з об'ємами 6, 5 і 4 та вартостями відповідно 3, 4 і 5. Впорядкуємо предмети відповідно до (2.28):

$$5/4 \ge 4/5 \ge 3/6.$$

На основі наведеного алгоритму спочатку буде взято предмет вагою 4, а потім - вагою 5.

Зауважимо, що для цього прикладу на основі даної схеми евристичного алгоритму був отриманий оптимальний розв'язок.

Застосування жадібного принципу розв'язування задачі про розподіл часового ресурсу може розглядатися лише у випадку оптимізації бальної оцінки виконання сукупності робіт (2.7)-(2.8). У даному випадку співвідношення (2.28) алгоритму вибору порядку набувають вигляду

$$\frac{B_1}{t_1} \ge \frac{B_2}{t_2} \ge \dots \ge \frac{B_n}{t_n},$$

що є повною аналогією загального принципу, закладеному у жадібній схемі визначення порядку вибору.

В умовах загальної задачі про розподіл часового ресурсу (2.5)-(2.6) такий підхід не дозволяє визначитись з порядком виконання робіт: за цією схемою всі роботи є рівнозначними. Тому для розв'язання задачі у такій постановці застосовувався принцип впорядкування робіт у вигляді

$$\frac{t_{\pi_1}}{t_{\pi_2}} \ge \frac{t_{\pi_2}}{t_{\pi_3}} \ge \dots \ge \frac{t_{\pi_{n-1}}}{t_{\pi_n}}, \tag{2.29}$$

де $\pi_i, i = \overline{1,n}$, - елементи перестановки $\pi = (\pi_1, \dots, \pi_n)$ послідовності номерів виконання робіт.

Аналогічно сформулюємо схему *жадібного алгоритму для задачі рекомбінації*. Задачу пошуку перестановки π^*, на якій досягається оптимальне значення часу переналагодження, запишемо у вигляді задачі лінійного програмування з функцією цілі

$$s(\pi)=\sum_{i=1}^{n-1} s_{\pi_i\pi_{i+1}} y_{\pi_i\pi_{i+1}} \to \min \qquad (2.30)$$

та обмеженнями

$$\sum_{i=1}^{n-1} y_{\pi_i\pi_{i+1}} = n-1,\ y_{\pi_i\pi_{i+1}} \in \{0,1\},\ i=\overline{1,n-1}. \qquad (2.31)$$

Для знаходження наближеного розв’язку з використанням жадібної схеми розташуємо роботи за незростанням величини частки відношення тривалостей довільних двох робіт до необхідного в цьому випадку часу переналагодження:

$$\left. t_{\pi_i} \middle/ t_{\pi_{i+1}} \right/ s_{\pi_i\pi_{i+1}},\ i=\overline{1,n-1}. \qquad (2.32)$$

Впорядкування виконання робіт за даною схемою передбачає першочерговий вибір робіт з приблизно однаковою тривалістю та найменшим часом налагодження. Без обмеження загальності, можна стверджувати, що даний підхід дозволяє отримати допустимий розв’язок (перестановку), близький до оптимального.

2.2.5. Застосування економічних стратегій для розв’язання задач розподілу ресурсів.

Задача максимально ефективного використання часу $\overline{T}$ може розглядатися як одна з класичних економічних задач розподілу (резервування) ресурсів заданого об'єму за множиною категорій (робіт) [76]. Постановка такої задачі полягає в знаходженні плану витрачання наявного ресурсу (у нашому випадку таким ресурсом є час $\overline{T}$) на виконання N завдань, при якому сумарна оцінка (наприклад, сума балів за виконання) є максимальною.

Формальне використання даної постановки у разі розподілу часу як ресурсу, очевидно, не зовсім коректно. Початкова задача за наявності мети максимально ефективного використання часу $\overline{T}$ характеризується наявністю оцінок за рішення завдань в балах $B_i,\ i=\overline{1,N}$. Очевидно також, що в загальному випадку виділення часового ресурсу на рішення однієї задачі (навіть при максимальній оцінці за неї) не є оптимальним з погляду підсумкової оцінки. З іншого боку, рівномірний розподіл часу на виконання завдань допустимо лише у разі набору з N однакових завдань з однаковими витратами на їх виконання і, як наслідок, однією і тією ж кількістю балів як оцінка.

Зрозуміло, що для досягнення мети в умовах відсутності дефіциту часу, окрім рівномірного виділення інтервалів часу на рішення окремих завдань, можна запропонувати використовувати пропорційний розподіл ресурсу [78]. Для кожного завдання $i=\overline{1,N}$ часові інтервали їх рішення задаються величинами, що є тривіальним рішенням, в рамках якого подальше дослідження і моделювання процесу виконання завдань не має сенсу. Очевидно, що підсумкова оцінка за виконання робіт дорівнює $\sum_{i=1}^{N} B_i$.

Як вже підкреслювалось вище, більш прикладний аспект має випадок дефіциту часу, тобто $\sum_{i=1}^{N} T_i \ge \overline{T}$. Розподіл ресурсу часу $\overline{T}$ за пропорційним принципом припускає виділення на рішення кожного завдання проміжку λT_i, $i=\overline{1,N}$, де параметр λ вибирається так, щоб сума отриманих інтервалів дорівнювала $\overline{T}$. Неважко набути значення $\lambda = \overline{T} / \sum_{i=1}^{N} T_i$.

У такій постановці можна також використовувати методи обмеженого вирівнювання досягнень і обмеженого вирівнювання втрат [79]. Для першого з них час для вирішення i-го завдання складає $\min\{\lambda, T_i\}$, $i=\overline{1,N}$, де параметр λ є рішенням параметричного рівняння $\sum_{i=1}^{N} \min\{\lambda, T_i\} = \overline{T}$. При обмеженому вирівнюванні втрат час для вирішення кожного завдання визначається із співвідношень $\max\{T_i - \lambda, 0\}$, $i=\overline{1,N}$, де параметр λ шукається як рішення параметричного рівняння $\sum_{i=1}^{N} \max\{T_i - \lambda, 0\} = \sum_{i=1}^{N} T_i - \overline{T}$.

Використання розглянутих стратегій при розподілі ресурсу характерний для постановок економічних задач про банкрутство. В умовах дефіциту часу використання даної постановки цілком коректно, а підсумковими оцінками у разі пропорційного методу і методів обмеженого вирівнювання будуть величини $\sum_{i \in I} B_i$, де I - множина виконаних завдань. Тут передбачається, що рішення кожного завдання може бути проведене в межах часового інтервалу, заданого відповідним обмеженням на час виконання $T_i, i = \overline{1, N}$. Як наслідок, множина I містить лише номери завершених завдань.

2.3. Висновки по другому розділу

У другому розділі розглянуто задачі оптимального розподілу часового ресурсу: наведено класифікацію задач теорії розкладів залежно від різних характеристик, викладено математичні постановки оптимізаційних задач розподілу часового ресурсу, сформульовано загальної задачі складання розкладу в термінах лінійного цілочисельного програмування, досліджено задачу складання розкладу виконання заданої кількості робіт для однієї машини, наведено постановку і спосіб розв'язання задачі оптимальної рекомбінації виконання робіт.

Проведено огляд методів розв'язання задач оптимального розподілу часового ресурсу та рекомбінації, їх властивостей та умов застосування. Викладено особливості використання методу динамічного програмування для розв'язування задач теорії розкладів у випадках з регулярним та нерегулярним критеріями.

На основі методу найменших відхилень запропоновано спосіб розв'язання задачі розподілу часового ресурсу при виконанні заданої кількості робіт. Запропоновано схему реалізації жадібного алгоритму для розв'язування задач часового розподілу та рекомбінації. Реалізовано підходи до застосування економічних стратегій при розв'язанні задач розподілу часових ресурсів.

РОЗДІЛ 3
МОДЕЛІ ТА МЕТОДИ РОЗВ'ЯЗУВАННЯ НЕЧІТКИХ ЗАДАЧ ОПТИМАЛЬНОГО РОЗПОДІЛУ ЧАСОВОГО РЕСУРСУ

3.1. Нечітка задача про рюкзак як засіб розподілу нечітко визначеного часового ресурсу

Розглянемо, як і раніше, N завдань, що підлягають виконанню $Z_i, i=\overline{1,N}$, кожен з яких характеризується часом виконання $T_i, i=\overline{1,N}$. Вирішення кожного завдання оцінюється відповідною кількістю балів B_i, $i=\overline{1,N}$. Час на виконання усіх завдань обмежений нечіткою величиною $\tilde{\overline{T}}$, яка подається у вигляді правого нечіткого трикутного числа $(\overline{T}, \overline{T}, \overline{T}+\Delta)$, де Δ - максимально можливе продовження терміну виконання.

Для випадку визначення робіт, виконання яких забезпечує максимальну бальну оцінку і при цьому інтервали часу на переналагодження завдань не враховуються, можна сформулювати задачу у вигляді оптимізаційної задачі про рюкзак. Позначимо через $x_i \in \{0,1\}$, $i=\overline{1,N}$, - бульові змінні, одиничні значення яких відповідають виконанню завдання, а нульові – його невиконанню. Тоді задача оптимізації підсумкової оцінки при виконанні сукупності робіт, має вигляд

$$W=\sum_{i=1}^{N} B_i x_i \to \max \tag{3.1}$$

$$\sum_{i=1}^{N} T_i x_i \le \tilde{\overline{T}}, \tag{3.2}$$

яка є задачею бульового лінійного програмування з нечіткими ресурсними обмеженнями у правій частині (див. підрозділ 1.2.3). Найкраща підсумкова бальна оцінка визначатиметься значенням $W^o = W(x^0, \lambda^0)$ цільової функції (3.1) на оптимальному нечіткому розв'язку Белмана-Заде (x^0, λ^0) ЗЛП виду (1.50).

Прикладом таких постановок служать звичайні ситуації оцінювання знань студентами на підставі тестових завдань по різних дисциплінах на модульних контрольних роботах або підсумкових семестрових заліках і іспитах, в процесі яких необхідно виконати ряд завдань зі встановленим регламентом часу на кожне з них і на сукупність в цілому.

Якщо враховувати час переналагодження при перемиканні з однієї роботи на іншу, стає важливим порядок виконання робіт.

За таких умов маємо, що розв’язком задачі оптимального використання нечітко заданого часу $\tilde{\tilde{T}}$ буде перестановка $p=(p_1,...,p_m)$, $m \le N$, $p_i \in \overline{1,N}$, $i=\overline{1,m}$, яка визначає порядок виконання робіт і яка є розв’язком оптимізаційної задачі, аналогічної (3.1), (3.2):

$$W=\sum_{i=1}^{m} B_{p_i} y_{p_i} \to \max_{p} \tag{3.3}$$

$$\sum_{i=1}^{N}\sum_{\substack{j=1\\ j\ne i}}^{N}(T_i+\pi_{ij})x_{ij} \le \tilde{\tilde{T}}\,. \tag{3.4}$$

Тут $x_{ij} \in \{0,1\}$ - змінні, що характеризують порядок виконання робіт (при одиничному значенні робота i виконується безпосередньо перед роботою j, в решті випадків - 0), $i, j=\overline{1,N}, i\ne j$, $y_i=\begin{cases}1, якщо\, x_{ij}=1,\\ 0, інакше,\end{cases}$, $i=\overline{1,m}$.

3.2. Задача складання розкладу з нечітко заданими інтервалами налаштувань

Розглянемо m виконавців (машин), що володіють різною продуктивністю. Припустимо, що задано сукупність робіт, які згруповано в p типів (без обмеження можна вважати задано p робіт). Кожен виконавець $k \in \{1,...m\}$ може виконувати будь-яку роботу, але для виконання роботи i-го типу виконавці можуть витрачати різний час. Цей факт враховується технологічною матрицею

$$T = \begin{pmatrix} t_{11} & t_{12} & \dots & t_{1m} \\ t_{21} & t_{22} & \dots & t_{2m} \\ \dots & \dots & \dots & \dots \\ t_{p1} & t_{p2} & \dots & t_{pm} \end{pmatrix} \tag{3.5}$$

де t_{ik} – час виконання роботи типу i виконавцем k. Необхідно розподілити p робіт по m виконавцям (скласти план робіт) так, щоб загальний час виконання робіт, що обчислюється з початку роботи першого виконавця і закінчується за фактом звільнення всіх виконавців, був мінімальним. Відповідно до нотації, описаної в [79], сформульовану задачу позначимо у вигляді $Rm||Cmax$.

У класичному випадку, де $t_{ik} \in R$, задачу можна звести до задачі дискретного програмування наступного вигляду

$$\max_k \sum_{i=1}^{p} c_{ik} t_{ik} \to \min$$

$$\sum_{k=1}^{m} c_{ik} = n_i,\ i = \overline{1, p}, \tag{3.6}$$

$$c_{ik} \ge 0,\ i = \overline{1, p},\ k = \overline{1, m},$$

де c_{ik} – кількість призначених робіт типу i для виконавця k, n_i – загальна (сумарна) кількість робіт типу i. Сукупність обмежень в задачі (3.6) гарантує, що всі роботи будуть залучені до розкладу. Розв'язок задачі (3.6) формується у вигляді матриці $C = \{ c_{ik} \}_{i=\overline{1,p};k=\overline{1,m}}$.

■Задача складання розкладу в умовах неточних даних.

Припустимо, що початкова інформація для задачі $Rm||Cmax$ є наближеною і задається у вигляді трикутних нечітких чисел $\tilde{t}_{ik} = (\underline{t_{ik}}, t_{ik}, \overline{t_{ik}})$, які представляють собою один з підходів до формалізації неточних величин. Подання інформації про час виконання робіт у такому вигляді є найпоширенішим способом представлення невизначеної інформації.

Отже, початкові дані мають вигляд нечіткої технологічної матриці $\tilde{T} = \{\tilde{t}_{ik}\}_{pxm}$ – матриці, компоненти якої подаються нечіткими трикутними значеннями часу виконання робіт. Для розв'язання задачі планування розглянемо підхід, що дозволить отримати субоптимальний план.

Під *зведенням* задачі до дискретного варіанту задачі оптимізації (3.6) будемо розуміти, що з матриці $\widetilde{T}$ можна отримати матрицю T за допомогою обчислення значень елементів α-зрізів за формулою [80]

$$\xi_\alpha(t_{ik}) = \alpha \underline{t_{ik}} + (1-\alpha)\overline{t_{ik}}, \tag{3.7}$$

де $\alpha \in [0,1]$.

При різних α отримуємо такі варіанти планування:

- $\alpha \to 1$, - песимістичне планування (оптимізація за максимальним часом виконання робіт);
- $\alpha \to 0$, - оптимістичне планування;
- $\alpha \to 0{,}5$, - планування за середніми величинами;
- $\alpha \sim \alpha_{ik}$, - планування, в рамках якого формується окрема матриця $\mathrm{A} = \{\alpha_{ik}\}_{pxm}$, за якою кожному елементу t_{ik} ставиться у відповідність свій коефіцієнт α_{ik} для перетворення $\widetilde{T}$ у T.

Матриця $\mathrm{A} = \{\alpha_{ik}\}_{pxm}$ може бути сформована особою, що приймає рішення (експертом), тому, як наслідок, можна говорити про використання експертної інформації. Однак, знань експертів може не вистачити для точного визначення коефіцієнтів, із-за чого розв'язки задачі будуть неоптимальними. Автоматизувати налаштування матриці А можна за допомогою залучення оберненого зв'язку, хоча при цьому мають місце такі проблеми:

- проблема навчання – цілком зрозуміло, що отримати оптимальні значення коефіцієнтів α_{ik} неможливо, тому їх вибір перетворюється в окремий ітераційний процес, причому з урахуванням змін умов задачі $R_m||C_{max}$; крім того, необхідно забезпечити поетапну активну пробну перевірку на основі спеціально організованого пошуку методом спроб і помилок;
- проблема оцінки – важко визначити, наскільки поточне значення α_{ik} краще або гірше за інше.

Незалежно від вибору α невизначеність щодо часу виконання робіт «ліквідується» до етапу розв'язування задачі дискретного програмування (3.6).

Інший спосіб базується на тому, що розв'язок задачі (3.6) формується з використанням функції пристосування. Це вимагає проведення ранжування результатів розв'язання задачі (3.6), спираючись на правила обробки нечітких трикутних чисел [81].

Обчислимо функцію пристосованості для розв'язку C, що містить невизначеність (враховуючи $c_{ik} \geq 0$):

$$R(C) = \max_k \sum_{i=1}^{p} c_{ik} \tilde{t}_{ik} = \max_k \sum_{i=1}^{p} [c_{ik} \underline{t_{ik}}, c_{ik} \overline{t_{ik}}] = \max_k (\underline{[\sum_{i=1}^{p} c_{ik} t_{ik}}, \overline{\sum_{i=1}^{p} c_{ik} t_{ik}}] = [\underline{C}, \overline{C}]. \quad (3.8)$$

Для реалізації оператора порівняння значень функції порівняння застосуємо спосіб порівняння інтервально заданих чисел [82]

$$\xi(R(C_1)) > \xi(R(C_2)) \Leftrightarrow R(C_1) > R(C_2), \quad (3.9)$$

при α=0.5 [8-9]. Однак, конкретно для задачі (3.6) допустимим є варіант

$$R(C_1) > R(C_2) \Leftrightarrow (\overline{C_1} > \overline{C_2}) \vee ((\overline{C_1} = \overline{C_2}) \wedge (\underline{C_1} > \underline{C_2})). \quad (3.10)$$

Згідно з (3.10) при порівнянні планів вводиться пріоритет у послідовності перевірки: спочатку аналізується найгірший можливий випадок виконання робіт за планом, після чого, якщо наслідки однакові, переглядається найкращий випадок.

3.3. Задачі теорії розкладів з нечітким вимірюванням часу

Емоції в людських організмах виконують функцію локального критерію управління, підказуючи організму, що добре і що погано в даних конкретних умовах, тобто емоція грає роль «пеленга», який підказує організму, чи рухається він до мети або від мети. У першому випадку емоція позитивна («задоволення»), в другому - негативна («страждання»). При формалізації емоцій можна спиратися на введений Г.А.Голіциним принцип максимуму взаємної інформації між умовами середовища і реакціями системи [83]. Згідно цього принципу емоції розглядаються як засоби квазіоптимального управління поведінкою системи (суб'єкта), що направляють її на досягнення максимуму її цільової функції (максимуму взаємної інформації між умовами середовища і реакціями системи).

Збільшення деякої цільової функції L супроводжується позитивними емоціями, зменшення - негативними емоціями. Оскільки L залежить від деяких змінних $x = (x_1, \ldots, x_n)$ (чинників впливу на емоції), то величини емоцій e виникають внаслідок змін цих параметрів. Більш того, емоції, по суті, можна вважати результатом інтегральної оцінки ситуації не за всіма факторами, що описують її (таких параметрів може виявитися дуже багато), а лише за декількома найбільш важливими характеристиками. Відповідно, емоції можуть «запускати» поведінку, не обов'язково оптимальну в конкретній недостатньо вивченій ситуації, але таку, яка найбільшою мірою дозволить уникнути (може й з великими втратами), катастрофічних наслідків перевищення деякого важливого (наприклад, часового) ресурсу.

Існує гіпотеза [84], що моделі емоцій реалізується у людини на основі нейроподібних розподілених обчислювальних структур, завдання яких оперувати оцінками (як у сфері матеріальних благ, так і благ нематеріальних).

З іншого боку, метою побудови і розвитку теоретико-множинних підходів у сфері моделювання людської поведінки завжди було бажання адаптувати математичні моделі до реального життя, отримати можливість органічно суміщати потенціал обчислювальних методів зі специфікою людського мислення. Тому, в якості найпростіших підходів до формування моделей людської поведінки застосовуються підходи з теорії нечітких множин, в рамках чого інтуїтивно обираються зрозумілі об'єкти та поняття, з інтерпретацією яких необхідно мати справу.

Необхідно відмітити, що особливу увагу у сфері людських ресурсів займає час. Наприклад, можна сформулювати нечіткі множини на основі виразів, котрі нечітко визначають плин часу у вигляді лінгвістичних термів «швидке реагування», «звичайний часовий відлік» або «довге очікування». Відповідно, при розв'язанні задач, в яких виникає необхідність в реалізації нечітких вербальних термів для опису часового відліку, треба враховувати таку нерівномірність плину часу. Формалізація нечітких термів визначається за допомогою конкретних функцій належності деяких нечітких множин, які будуються на базі сукупності знань, отриманих зі сховища або на основі обробки експертної інформації.

3.3.1. Застосування структурованих нечітких чисел для опису вимірювання відліку часу.

Як вже було підкреслено в огляді нечітких величин та чисел, важливою характеристикою нечіткої множини є поняття носія. Носієм нечіткої множини $\tilde{A}$ є звичайна множина $\operatorname{supp}\tilde{A} = \{x \in X : \mu_{\tilde{A}}(x) > 0\}$, що містить ті і лише ті елементи універсального простору нечіткої множини, для котрих значення функції належності цієї нечіткої множини відрізняються від нуля. Для нечіткого числа носієм є інтервал. При цьому, для нечіткого трикутного числа $\tilde{A} = (a,b,c)$ носієм буде інтервал (a,c), для правого нечіткого трикутного – інтервал $(-\infty,c)$, для лівого нечіткого трикутного – інтервал $(a,+\infty)$.

Над нечіткими множинами визначені всі операції, характерні для традиційної теорії точних множин, але, природно, з своєю специфікою, що задається самою сутністю нечітких множин. Операції перетину, об’єднання, доповнення та інші для нечітких множин визначаються відомими співвідношеннями [80], але не можуть повністю забезпечити ряд корисних властивостей нечітких чисел, що узагальнюють і розширюють операції зі звичайними (чіткими) числами.

Означення 3.1. Нечітке трикутне число $\tilde{E}[u,v]$ з носієм, що задається числовим інтервалом $[u,v]$, називатимемо нечітким оригіналом.

Нечіткий оригінал $\tilde{E}(u,v]$ будемо називати правим, якщо носій відповідного нечіткого числа задається інтервалом $(u,v]$, а нечіткий оригінал $\tilde{E}[u,v)$ називатимемо лівим, якщо носій задається інтервалом $[u,v)$.

Виходячи з цього, зауважимо, що довільне праве нечітке трикутне число *(b, b, c)* можна розглядати як правий нечіткий оригінал з відповідними значеннями $u=-\infty, v=c$, а ліве нечітке трикутне число *(a, b, b)* - як лівий нечіткий оригінал з $u=a, v=+\infty$.

Очевидно, можна покласти $u=0, v=1$. Тоді, нечіткий оригінал $\tilde{E}[0,1]=\tilde{E}$ називатимемо нечітким одиничним оригіналом, і, відповідно, $\tilde{E}(0,1]$ - правим нечітким одиничним оригіналом, $\tilde{E}[0,1)$ - лівим нечітким одиничним оригіналом. Нечіткий оригінал $\tilde{E}[0,v]$ з носієм $[0,v]$ називатимемо початковим.

Назва введеного поняття пояснюється тим, що на основі заданого нечіткого оригіналу можна будувати нові нечіткі множини з деякими специфічними властивостями.

Означення 3.2. Нечітка числова множина $\tilde{R}[s,t]$ з носієм, що задається числовим інтервалом $[s,t]$, буде реплікацією нечіткого оригіналу $\tilde{E}[u,v]$, якщо справедлива умова $|v-u|=|t-s|$.

Необхідно зауважити, що функція належності, яка визначена на нечіткому оригіналі, не обов'язково має співпадати з функцією належності реплікації $\tilde{R}[s,t]$. У випадку $|t-s|=1$ множина $\tilde{R}[s,t]$ є реплікацією нечіткого одиничного оригіналу. Зрозуміло також, що за додаткової умови $[s,t]\cap[u,v]=\varnothing$ оригінал і реплікація не перетинаються, а у випадку $[s,t]\cap[u,v]\neq\varnothing$ - частково або повністю перекриваються.

Означення 3.3. Нечітку числову множину $\tilde{R}=\bigcup_{i=1}^{n}\tilde{R}_i[s_i,t_i]$, де $\tilde{R}_i[s_i,t_i]$, $i=\overline{1,n}$, є реплікаціями нечіткого оригіналу $\tilde{E}[u,v]$ з носіями, заданими числовими інтервалами $[s_i,t_i]$, $i=\overline{1,n}$, $\bigcap_{i=1}^{n}[s_i,t_i]=\varnothing$, будемо називати n-кратною реплікацією нечіткого оригіналу $\tilde{E}[u,v]$.

При цьому допускається частковий перетин оригіналу з не більш як двома реплікаціями або повне перекриття з однією з реплікацій. Кратність реплікації може бути нескінченною, відповідно до чого таку реплікацію називатимемо реплікацією нечіткого оригіналу нескінченної кратності.

Означення 3.4. Нечітку числову множину $\tilde{R}=\bigcup_{i=1}^{n}\tilde{R}_i[s_i,s_{i+1}]$, де $\tilde{R}_i[s_i,s_{i+1}]$, $i=\overline{1,n}$, є реплікаціями нечіткого оригіналу $\tilde{E}[u,v]$ з носіями, заданими числовими інтервалами $[s_i,s_{i+1}]$, $i=\overline{1,n}$, будемо називати n-кратною послідовною реплікацією нечіткого оригіналу $\tilde{E}[u,v]$.

У точках з'єднання реплікацій $s_2,s_3,...,s_n$ значення функції належності отримуємо у вигляді $\mu_{\tilde{R}}(s_{i+1})=\max\{\mu_{R_i}(s_{i+1}),\mu_{R_{i+1}}(s_{i+1})\}$, $i=\overline{1,n-1}$.

Далі розглянемо випадок реплікацій, для яких повністю зберігаються величини функцій належності елементів нечіткого оригіналу.

Означення 3.5. Нечітку реплікацію $\tilde{C}[s,t]$ заданого оригіналу $\tilde{E}[u,v]$ назвемо копією, якщо для всіх $x=u+\delta$, $y=s+\delta$, $0\le\delta\le\Delta$, $\Delta=v-u=t-s$, виконується умова $\mu_{\tilde{C}}(y)=\mu_{\tilde{E}}(x)$.

Означення 3.6. Нечітку реплікацію $\tilde{C}=\bigcup_{i=1}^{n}\tilde{C}_i[s_i,t_i]$, де $\tilde{C}_i[s_i,t_i]$, $i=\overline{1,n}$, є копіями нечіткого оригіналу $\tilde{E}[u,v]$ з носіями, заданими числовими інтервалами $[s_i,t_i]$, $i=\overline{1,n}$, $\bigcap_{i=1}^{n}[s_i,t_i]=\varnothing$, будемо називати n-кратною копією нечіткого оригіналу $\tilde{E}[u,v]$.

Означення 3.7. Нечітку реплікацію $\tilde{C}=\bigcup_{i=1}^{n}\tilde{C}_i[s_i,s_{i+1}]$, де $\tilde{C}_i[s_i,s_{i+1}]$, $i=\overline{1,n}$, є копіями нечіткого оригіналу $\tilde{E}[u,v]$ з носіями, заданими числовими інтервалами $[s_i,s_{i+1}]$, $i=\overline{1,n}$, будемо називати n-кратною послідовною копією нечіткого оригіналу $\tilde{E}[u,v]$.

Як і раніше, у випадку нескінченної кратності копії називатимемо копіями нечіткого оригіналу нескінченної кратності.

Використаємо нечіткі структуровані числові множини для опису невизначеності у вимірюванні відліку часу. Тривалість часових інтервалів, поданих у лінгвістичній формі, є відбиттям сприйняття суб'єктом відрізку часу, що, як відмічалось вище, піддається емоційному впливу специфічних умов, які зв'язані з режимом відслідковування часу. Відомо, що дефіцит часу, що виникає у ситуаціях з оперативним прийняттям рішень, швидкісного тестування, різних змагань з обмеженням часу, приводить до «прискорення» часу. Відповідно, при його надлишку і відсутності процесів, що «споживають» час (очікування подій, тривала бездіяльність і т.і.), плин часу сповільнюється. Іншими словами, вимірювання відліку часу в різних ситуаціях визначається суб'єктивною оцінкою, яку можна описати нечіткою величиною трикутного вигляду.

Припустимо, що вимірювання часу відбувається за допомогою інтервалів однієї тривалості (такими інтервалами можна вважати будь-яку одиницю часу, наприклад, 1 секунду, 1 день або 1 рік, в залежності від процесу). Для відслідковування нечіткого відліку часу будемо інтуїтивно оцінювати проміжок, який залишається до завершення кожного інтервалу часу. В цьому випадку, «швидкий» плин одиниці часу може бути заданий правим нечітким трикутним числом с носієм, довжина якого менше тривалості часового інтервалу, а «повільний» - такого ж вигляду нечітким числом з носієм, довжина якого більша за тривалість інтервалу. Очевидно, що, якщо час «тече» природним чином, то величина носія даного нечіткого числа за довжиною співпадає з величиною одиничного часового інтервалу.

Таким чином, маючи зразок відліку вимірювання інтервалу часу у вигляді лівого початкового оригінала $\widetilde{E}[0,v)$, можна визначити нечітку n-кратну послідовну копію на його основі, яка буде описувати динаміку зміни часу на заданому часовому проміжку.

Нехай $\widetilde{E}^{Q}[0,v_{Q})$ - лівий нечіткий оригінал у формі трикутного числа з лінійною спадною функцією належності, який визначає «швидкий» одиничний проміжок часу ($v_Q < 1$), а $\tilde{E}^D[0, v_D)$ - аналогічний нечіткий оригінал, що визначає «повільний» одиничний проміжок ($v_D > 1$). Визначимо $\tilde{C}^Q = \bigcup_{i_Q=1}^{n_Q} \tilde{C}^Q_{i_Q}[s_{i_Q}, s_{i_Q+1}]$ - нечітку n_Q-кратну послідовну копію оригінала $\tilde{E}^Q[0, v_Q)$, що описує «швидку» зміну часу на проміжку $[s_{i_Q}, s_{i_Q+1}]$, і $\tilde{C}^D = \bigcup_{i_D=1}^{n_D} \tilde{C}^D_{i_D}[s_{n_Q+i_D}, s_{n_Q+i_D+1}]$ - нечітку n_D-кратну послідовну копію оригінала $\tilde{E}^D[0, v_D)$, яка визначає «швидку» зміну часу на проміжку $[s_{n_Q+i_D}, s_{n_Q+i_D+1}]$. Тут $\tilde{C}^Q_{i_Q}[s_{i_Q}, s_{i_Q+1}]$, $i_Q = \overline{1, n_Q}$, $\tilde{C}^D_{i_D}[s_{n_Q+i_D}, s_{n_Q+i_D+1}]$, $i_D = \overline{1, n_D}$ - копії лівих нечітких оригіналів $\tilde{E}^Q[0, v_Q)$ і $\tilde{E}^D[0, v_D)$ відповідно, з носіями, заданими числовими інтервалами заданої довжини v_Q і v_D.

Тоді нечітка числова множина вигляду $\tilde{C}^Q \cup \tilde{C}^D$ буде описувати відлік часу на проміжку $[s_1, s_{n_Q+n_D+1}]$. Цей відлік складається з двох фаз: фази швидкої зміни часу на $[s_1, s_{n_Q+1}]$ і повільної - на $[s_{n_Q+1}, s_{n_Q+n_D+1}]$. Очевидно, що, змінюючи характер відліку часу на проміжках та об'єднуючи отримані нечіткі копії, можна побудувати нечітку числову множину, яка відбиває нечіткість у сприйнятті плину часу. Крім цього, отримана нечітка числова множина може бути доповнена копією одиничного оригінала $\tilde{E}$, який визначає природній відлік часу.

Необхідно відмітити, що «швидкий» або «повільний» відлік часу характерний не лише для людського сприйняття, але й для різних процесів у технічних системах. Швидкість функціонування технічного пристрою достатньо часто визначається частотою (кількістю в одиницю часу) тактових імпульсів, що подаються на вхід. Збільшення частоти в межах допустимого інтервалу дозволяє виконати одну й ту ж роботу швидше або повільніше. Низька частота тактових імпульсів фактично визначає «швидкий» плин часу, а висока частота – навпаки, його «повільний» плин. Таким чином, це ще раз підтверджує, що нечітке оцінювання швидкості зміни відліку часу можна отримати на основі базового нечіткого оригінала, який описує вербальний терм «залишок до завершення» одиниці часу.

3.3.2. Нечітка задача про рюкзак як засіб розподілу часового ресурсу з нечітко заданими термінами виконання.

Застосування нечіткого вимірювання відліку часу можна розглядати у різних математичних оптимізаційних задачах, що виникають при визначенні порядку виконання завдань з урахуванням або без урахування яких-небудь додаткових обмежень і цільових функцій, які зв'язані з процесом обробки завдань.

Припустимо, як і раніше, є N задач (завдань, що підлягають виконанню) Z_i, $i=\overline{1,N}$, кожна з яких характеризується нечітким часом розв'язання $\widetilde{T}_i=(T_i,T_i,T_i+\Delta_i)$, $i=\overline{1,N}$, який залежить від умов виконання завдання. Тут Δ_i, $i=\overline{1,N}$, - можливі затримки у проведенні робіт. Можливо також, що для переключення з одного завдання на інше необхідний часовий проміжок (час переналагодження) π_{ij}, $i=\overline{1,N}$, $j=\overline{1,N}$, $i\neq j$. Виконання завдань проводиться одним виконавцем. Час на виконання усіх завдань обмежено деякою величиною $\overline{T}$.

Формулювання задачі розподілу часового ресурсу з нечітко заданими термінами виконання зв'язані не лише з неможливістю оцінювання часу проведення робіт, а й намаганням врахувати вплив на швидкість виконання суб'єктивних факторів сприйняття плину часу.

За умов, що інтервали часу на переналагодження не враховуються або не є суттєвими, і при наявності дефіциту часу, тобто $\sum_{i=1}^{N} T_i \geq \overline{T}$, проблему найбільш ефективного використання часового ресурсу можна сформулювати як нечітку задачу бульового лінійного програмування

$$W = \sum_{i=1}^{N} B_i x_i \to \max , \qquad (3.11)$$

$$\sum_{i=1}^{N} \tilde{T}_i x_i \le \overline{T} , \qquad (3.12)$$

$$x_i \in \{0,1\},\ i = \overline{1,N} , \qquad (3.13)$$

з нечітко заданими термінами виконання окремих завдань $\tilde{T}_i$, $i = \overline{1,N}$.

Як вже було сказано у підрозділі 1.2.4, за умов подання нечітких термінів виконання завдань у вигляді нечітких величин трикутного вигляду обмеження даної задачі містять добуток λx_j, $j = \overline{1,n},$ і тому не є опуклими. Отже, для знаходження розв’язку задачі (3.11)-(3.13) необхідно застосовувати методи, призначені для пошуку рішень неопуклих оптимізаційних задач.

Якщо враховувати час на переналагодження при переключенні з однієї роботи на іншу, стає важливим порядок виконання робіт. Як вже було сказано вище, це приводить до необхідності застосування методів комбінаторної оптимізації. Розв’язком задачі оптимального використання часу $\overline{T}$ буде перестановка $p = (p_1,..., p_m)$, $m \le N$, $p_i \in \overline{1,N}$, $i = \overline{1,m},$ яка визначає порядок виконання робіт і яка є розв’язком оптимізаційної задачі, аналогічній (3.11)-(3.13):

$$W = \sum_{i=1}^{m} B_{p_i} y_{p_i} \to \max , \qquad (3.14)$$

$$\sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \ne i}}^{N} (\tilde{T}_i + \pi_{ij}) x_{ij} \le \overline{T} , \qquad (3.15)$$

де $x_{ij} \in \{0,1\}$ - як і раніше, змінні, що характеризують порядок виконання робіт (при одиничному значенні робота i виконується безпосередньо перед роботою j, у решті випадків - 0), $i, j = \overline{1,N}, i \ne j$, $y_i = \begin{cases} 1, якщо\, x_{ij} = 1, \\ 0, інакше, \end{cases}$, $i = \overline{1,m}.$

Припускаючи далі, що своєчасне виконання окремих завдань може надати позитивний вплив на виконання наступних завдань сукупності робіт, а невиконання або затримка завершення робіт – сповільнює розв'язання всієї сукупності завдань, виникають оптимізаційні задачі ефективного використання часу $\overline{T}$ виду (3.11)-(3.13) і (3.14)-(3.15) с урахуванням нечіткого відліку термінів виконання окремих завдань.

Для аналізу впливу динаміки відліку часу на отримані розв'язки було проведено чисельні експерименти. Формалізація нечіткого плину часу проводилася за спрощеною схемою. Виконання або невиконання завдання чергового моделювалося випадковим чином. При цьому, виконання завдання «сповільнювало» темп зменшення часу, а невиконання - навпаки, його «прискорювало». Зміна швидкості у процесі обчислень фіксувалось у вигляді відповідних модифікацій виділеного на виконання усіх завдань часу $\overline{T}$: величина збільшувалась пропорційно часу виконаного поточного завдання (ефект сповільнення темпу плину часу) і зменшувалась – при невиконанні поточного завдання (ефект прискорення). Результати чисельних розрахунків розподілів часового ресурсу при виконанні сукупності завдань у випадку нечітко заданих термінів виконання наведено у Додатку А.

3.3.3. Гібридна модель динаміки процесу обробки сукупності завдань.

Одним із стандартних підходів, що використовуються при моделюванні динаміки систем, є опис еволюції станів системи шляхом задання початкових значень координат системи та рівнянь, які визначають зміну координат з часом. Складна динамічна система, як правило, представляє собою сукупність підсистем, для кожної з яких можлива власна математична формалізація. Традиційно система називається гібридною, якщо її підсистеми описуються різними типами моделей. Окрім цього, вивчення процесів поведінки таких систем зумовлює розгляд і врахування ієрархічної структури взаємодії окремих складових (підсистем) системи [85].

Проведемо дослідження гібридної динамічної системи, яка описує процес обробки N задач, які потребують виконання за допомогою єдиного обчислювального пристрою (процесору).

Припустимо, що задачі, які мають виконуватися, характеризуються поняттям складності, яку можна оцінити за обсягом необхідного для виконання часу. Відповідно до цього обчислювальний пристрій виділяє необхідний часовий ресурс пропорційно складності кожної задачі, що виконується в даний момент. Обробка задачі у часі приводить до зменшення складності завдання зі швидкістю, обернено пропорційною загальній складності усіх задач.

Опис стану обчислювального пристрою може бути здійснений на основі підходу, що розглядається для моделювання процесів у нейронах. Роботу процесора опишемо за допомогою моделі функціонування нейроподібного елементу, що має n входів і один вихід. Стан елемента подається скалярною величиною завантаженості $u(t)$, яка називається потенціалом (величину $u(t)$ можна інтерпретувати як усереднене за часом значення завантаженості обчислювального пристрою [86]).

Позначимо через $x_i(t)$ – величину складності i-ї задачі у момент часу t, $i=1,2,...,N$; $w(t)=(w_1(t),...,w_N(t))$ – вектор бульовських компонент, одиничне значення яких $(w_i(t)=1)$ визначає обслуговування, а нульове $(w_i(t)=0)$ - очікування обслуговування i-ї задачі у момент часу t, $i=1,2,...,N$; $b(t)$ – сумарну величину складності задач у момент часу t. Розглянемо процес функціонування обчислювального пристрою на інтервалі $[t_0,T]$, де T - довільне додатне число.

Потенціал процесора зростає за часом пропорційно зваженій сумі $\frac{1}{N}\sum_{i=1}^{N} w_i(t)x_i(t)$ поточних рівнів складності задач $x_i(t)$, $i=1,2,...,N$. Поява кожної нової задачі та завершення виконання задач визначають дискретні моменти часу t_0^i і t_E^i, $i=1,2,...,N$, відповідно. Для всіх проміжків $T_i=t_E^i-t_0^i$, $i=1,2,...,N$, можна знайти найбільше спільне кратне Δt, таке, що $t_E^i=t_0^i+k_i\Delta t$, $k_i\in \mathrm{N}$, $i=1,2,...,N$. Без обмеження загальності, можна вважати, що $t_0=0$ і $\Delta t=1$.

Інтервал роботи кожної задачі $t\in[t_0^i,t_0^i+k_i\Delta t]$, $i=1,2,...,N$, в межах якого процесор здійснює виконання задачі, розглядається у вигляді суми проміжних часових інтервалів $[t_0^i + k_j^i \Delta t, t_0^i + k_{j+1}^i \Delta t]$, $j = 0,1,2,...,r_i$, $i = 1,2,...,N$, $k_j^i \in \mathrm{N} \cup \{0\}$, $k_0^i = 0$, $\sum_{j=0}^{r_i+1} k_j^i = k_i$. Це розбиття необхідне, тому що в межах кожного інтервалу виконання i-ї задачі відбуваються події, зв'язані з появою та завершенням інших задач. Зрозуміло, що якщо на інтервалі роботи i-ї задачі $[t_0^i, t_0^i + k_i \Delta t]$, $i = 1,2,..., N$, жодних інших подій не спостерігається, то $r_i = 0$ і відповідно маємо $k_0^i = 0$ та $k_1^i = k_i$.

Зміна сумарної величини складності задач $b(t)$, що виконуються у момент часу t, також пов'язана з моментами появи і завершення задач. Відповідно до запропонованих припущень щодо часової дискретизації визначаються моменти часу $t_k = k\Delta t = k$, $k \in \mathrm{N} \cup \{0\}$, в які відбувається зміна величини $b(t)$. Нескладно перевірити, що

$$b(t_{k+1}) = b(t_k) + \sum_{\substack{i=1 \\ w_i(t_{k+1}) > w_i(t_k)}}^{N} 2^{i-1} - \sum_{\substack{i=1 \\ w_i(t_{k+1}) < w_i(t_k)}}^{N} 2^{i-1}, \; b(t_0) = b(0) = 0, \tag{3.16}$$

і на кожному проміжку $[t_k, t_{k+1})$ величина $b(t) \equiv const$. Крім цього, послідовність $t_k = k$, $k \in \mathrm{N} \cup \{0\}$ повністю визначає моменти подій, що відбуваються в системі і описуються за допомогою розглянутих вище значень $t_0^i + k_j^i \Delta t$, $j = 0,1,2,...,r_i$, $i = 1,2,...,N$, $k_j^i \in \mathrm{N} \cup \{0\}$, $k_0^i = 0$, $\sum_{j=0}^{r_i+1} k_j^i = k_i$.

Таким чином, гібридна динамічна система, яка використовується для моделювання процесу обробки заданої кількості задач за допомогою обчислювального пристрою, складається з однієї лінійної функціональної стаціонарної підсистеми (3.16) та *N+1* лінійної диференціальної підсистеми, які записуються у вигляді

$$\dot{x}_i(t) = -\frac{1}{b(t)} x_i(t), \; x_i(t_0^i) = x_0^i, \; i = 1,2,...,N, \tag{3.17}$$

$$t \in [t_0^i + k_j^i \Delta t, t_0^i + k_{j+1}^i \Delta t],\ j = 0,1,2,...,r_i,\ i = 1,2,...,N,\ k_j^i \in \mathrm{N} \cup \{0\},\ \sum_{j=0}^{r_i+1} k_j^i = k_i,$$

$$\dot{u}(t) = -u(t) + \frac{1}{N}\sum_{i=1}^{N} w_i(t) x_i(t),\ u(t_0) = u(0) = 0,\ t \in [t_0, T], \tag{3.18}$$

$$b(t) = b(t-1) + \sum_{\substack{i=1 \\ w_i(t) > w_i(t-1)}}^{N} 2^{i-1} - \sum_{\substack{i=1 \\ w_i(t) < w_i(t-1)}}^{N} 2^{i-1},\ b(0) = 0,\ t = k,\ k \in \mathrm{N} \cup \{0\}, \tag{3.19}$$

$$w_i(t) = \begin{cases} 0 \\ 1 \end{cases},\ i = 1,2,...,N,\ t = k,\ k \in \mathrm{N} \cup \{0\}, \tag{3.20}$$

де x_0^i - початкова складність i-ї задачі, що поступає на виконання, $i = \overline{1, N}$.

Зважаючи на різну складність завдань, будемо впорядковувати наявні на обробці завдання за спаданням складності. Це не впливає на вигляд моделі, а лише призводить до іншої нумерації задач і, відповідно, до перестановки елементів вектора $w(t) = (w_1(t),..., w_N(t))$ у моменти появи нових задач.

Відомо, що для лінійних стаціонарних диференціальних рівнянь вигляду

$$\dot{x}(t) = Ax(t),\ x(t) \in R^n,\ t \ge 0, \tag{3.21}$$

розв'язок задачі Коші може бути поданий у вигляді добутку фундаментальної матриці рішень, нормованої при $t = t_0 = 0$, на вектор, що задає початковий стан

$$x(t) = e^{At} x_0. \tag{3.22}$$

Тут e^{At} - матрична функція, яка є матричним рядом, що називається матричним експоненціалом [87].

Розглянемо узагальнену гібридну динамічну систему, складену з однієї лінійної диференціальної стаціонарної підсистеми розмірності n та однієї лінійної функціональної підсистеми наступного вигляду

$$\dot{x}(t) = A(y([t]))x(t),\ x(t_0) = x(0) = x_0,\ t \ge 0, \tag{3.23}$$

$$y(k) = C(k)x(k) + y(k-1),\ k = 0,1,2,... . \tag{3.24}$$

У даній системі A-квадратна $(n \times n)$-матриця з постійними коефіцієнтами на кожному з інтервалів $t \in [k, k+1)$, $k = 0,1,2,...$, $x(t) \in R^n$, $t \ge 0$, $y(k) \in R^1$, $k = 0,1,2,...$, C- вектор з n елементів. Під розв'язком системи (3.23), (3.24) будемо розуміти вектор-функцію $(x_1(t),...,x_n(t),y(t))$, $t\ge 0$, що складається з векторної функції $x(t)$, яка є кусково неперервно-диференційовною та має розриви похідної у вузлових точках $t=k$, $k=1,2,...$, і кусково-постійної функції $y(t)$, яка має зліва розриви першого роду у вузлових точках $t=k$, $k=0,1,2,...$. Початкові умови для розв'язку системи (3.23), (3.24) мають вигляд

$$x(0)=x_0,\quad y(t)\equiv 0,\quad -1\le t\le 0. \tag{3.25}$$

Отримаємо представлення фундаментальної матриці рішень системи (3.23), (3.24) і, відповідно, розв'язок задачі Коші (3.23), (3.24), (3.25).

Вектор x_0 початкових станів системи будемо записувати у вигляді $x^0(0)$, а розв'язок $(x(t),y(t))=(x_1(t),...,x_n(t),y(t))$, визначений на проміжку $k\le t<k+1$, $k=0,1,2,...$, - у вигляді $(x^k(t),y(k))$.

Лема 3.1. Справедливе наступне співвідношення

$$\int_{i-1}^{i} e^{A(1-s)}ds=\sum_{k=0}^{\infty}\frac{A^k}{(k+1)!}\left[(2-i)^{k+1}-(1-i)^{k+1}\right]. \tag{3.26}$$

Причому, якщо матриця A невироджена, тобто існує A^{-1}, то

$$\int_{i-1}^{i} e^{A(1-s)}ds=A^{-1}\left[e^{A(i-2)}-e^{A(i-1)}\right]. \tag{3.27}$$

Доведення. Використовуючи вираз для матричного экспоненціалу e^{At} [87], отримуємо наступне співвідношення

$$\int_{i-1}^{i} e^{A(1-s)}ds=\int_{i-1}^{i}\left[I+A\frac{1-s}{1!}+A^2\frac{(1-s)^2}{2!}+...+A^k\frac{(1-s)^k}{k!}+...\right]ds=$$

$$=\left[-I\frac{1-s}{1!}-A\frac{(1-s)^2}{2!}-A^2\frac{(1-s)^3}{3!}-...-A^k\frac{(1-s)^{k+1}}{(k+1)!}+...\right]_{s=i-1}^{s=i}=$$

$$=\left[I\frac{2-i}{1!}+A\frac{(2-i)^2}{2!}+A^2\frac{(2-i)^3}{3!}+...+A^k\frac{(2-i)^{k+1}}{(k+1)!}+...\right]-$$

$$-\left[I\frac{1-i}{1!}+A\frac{(1-i)^2}{2!}+A^2\frac{(1-i)^3}{3!}+...+A^k\frac{(1-i)^{k+1}}{(k+1)!}+...\right].$$

Отже, маємо $\int\limits_{i-1}^{i} e^{A(1-s)}ds = \sum\limits_{k=0}^{\infty} \frac{A^k}{(k+1)!}\left[(2-i)^{k+1} - (1-i)^{k+1}\right]$, звідки випливає справедливість співвідношення (3.26).

Припустимо, що матриця A невироджена, тобто існує A^{-1}. Тоді отриману вище залежність можна спростити наступним чином

$$\int\limits_{i-1}^{i} e^{A(1-s)}ds = A^{-1}\left[I + A\frac{2-i}{1!} + A^2\frac{(2-i)^2}{2!} + \ldots + A^{k+1}\frac{(2-i)^{k+1}}{(k+1)!} + \ldots - I\right] -$$

$$- A^{-1}\left[I + A\frac{1-i}{1!} + A^2\frac{(1-i)^2}{2!} + \ldots + A^{k+1}\frac{(1-i)^{k+1}}{(k+1)!} + \ldots - I\right] = A^{-1}\left(e^{A(2-i)} - e^{A(1-i)}\right),$$

звідки випливає справедливість співвідношення (3.27).

Наслідок 3.1. Справедливе співвідношення

$$\int\limits_{k-1}^{t} e^{A(t-k+1-s)}ds = \sum\limits_{j=0}^{\infty} \frac{A^j}{(j+1)!}\left[t^{j+1} - (1-k)^{j+1}\right]. \qquad (3.28)$$

Причому, якщо матриця A невироджена, тобто існує A^{-1}, то

$$\int\limits_{k-1}^{t} e^{A(t-k+1-s)}ds = A^{-1}\left[e^{At} - e^{A(1-k)}\right]. \qquad (3.29)$$

Доведення. Повторюючи викладення, наведені в лемі, отримаємо

$$\int\limits_{k-1}^{t} e^{A(t-k+1-s)}ds = -\left(I\frac{t-k+1-s}{1!} + A\frac{(t-k+1-s)^2}{2!} + \ldots + A^j\frac{(t-k+1-s)^{j+1}}{(j+1)!} + \ldots\right)\Bigg|_{s=k-1}^{s=t} =$$

$$= \left[I\frac{t}{1!} + A\frac{t^2}{2!} + \ldots + A^j\frac{t^{j+1}}{(j+1)!} + \ldots\right] - \left[I\frac{1-k}{1!} + A\frac{(1-k)^2}{2!} + \ldots + A^j\frac{(1-k)^{j+1}}{(j+1)!} + \ldots\right] =$$

$$= \sum\limits_{j=0}^{\infty} \frac{A^j}{(j+1)!}\left[t^{j+1} - (1-k)^{j+1}\right].$$

Якщо матриця A невироджена, то

$$\int\limits_{k-1}^{t} e^{A(t-k+1-s)}ds = A^{-1}\left[I + A\frac{t}{1!} + A^2\frac{t^2}{2!} + \ldots + A^{j+1}\frac{t^{j+1}}{(j+1)!} + \ldots - I\right] -$$

$$- A^{-1}\left[I + A\frac{1-k}{1!} + A^2\frac{(1-k)^2}{2!} + \ldots + A^{j+1}\frac{(1-k)^{j+1}}{(j+1)!} + \ldots - I\right] = A^{-1}\left[e^{At} - e^{A(1-k)}\right].$$

Для гібридної системи (3.23), (3.24) справедливе твердження.

Теорема 3.1. На проміжку $k \le t < k+1$ фундаментальна матриця розв'язків гібридної системи (3.23), (3.24) має вигляд

$$\Phi(t) = Z_k(t) Z(k), \tag{3.30}$$

$$Z_k(t) = \begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix}, \;\; k \le t < k+1, \tag{3.31}$$

$$Z(k) = \begin{pmatrix} \prod\limits_{j=1}^{k} e^{A(y(k-j))} & 0 \\ \sum\limits_{j=1}^{k} C(k-j+1) \prod\limits_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix}, \; j = \overline{1,k}. \tag{3.32}$$

Доведення. Розглянемо перший проміжок $0 \le t < 1$ інтегрування системи (3.23), (3.24). Враховуючи, що $y(0) = 0$, підсистема диференціальних рівнянь системи (3.23), (3.24) буде мати вигляд $\dot{x}^0(t) = A(y(0))x^0(t) = A(0)x^0(t)$, $0 \le t < 1$.

Розв'язок $x^0(t)$ підсистеми (3.23) на проміжку $0 \le t < 1$ має вигляд

$$x^0(t) = e^{A(y(0))t} x^0(0) = e^{A(0)t} x^0(0),$$

де e^{At} - матричний експоненціал $e^{At} = I + A\frac{t}{1!} + A^2 \frac{t^2}{2!} + \ldots + A^k \frac{t^k}{k!} + \ldots$, а загальний розв'язок системи (3.23), (3.24) записується у вигляді

$$\begin{pmatrix} x^0(t) \\ y(0) \end{pmatrix} = \begin{pmatrix} e^{A(0)t} x^0(0) \\ 0 \end{pmatrix}. \tag{3.33}$$

Розглянемо другий проміжок $1 \le t < 2$ інтегрування системи (3.23), (3.24). У цьому випадку розв'язок функціональної підсистеми має вигляд $y(1) = C(1)x^1(1) + y(0)$.

Виходячи умови неперервності векторної функції $x(t)$ у вузловій точці $t = 1$, маємо $x^1(1) = x^0(1) = e^{A(0)} x^0(0)$. Тоді розв'язок $y(1)$ запишеться у вигляді

$$y(1) = C(1)x^0(1) + y(0) = C(1)e^{A(0)} x^0(0) + y(0). \tag{3.34}$$

Повернемося до першої підсистеми. На проміжку $1 \le t < 2$ вона приймає вигляд $\dot{x}^1(t) = A(y(1))x^1(t)$, розв'язок $x^1(t)$ якої записується у формі

$$x^1(t) = e^{A(y(1))(t-1)} x^1(1). \tag{3.35}$$

Враховуючи умови неперервності, отримуємо

$$x^1(t)=e^{A(y(1))(t-1)}x^1(1)=e^{A(y(1))(t-1)}x^0(1)=e^{A(y(1))(t-1)}e^{A(y(0))}x^0(0).$$

Продовжуючи цей процес далі, отримаємо, що на проміжку $k\le t<k+1$ розв'язок має вигляд

$$\begin{pmatrix} x^k(t) \\ y(k) \end{pmatrix}=\begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix}\begin{pmatrix} x^k(k) \\ y(k) \end{pmatrix}. \qquad (3.36)$$

Розглянемо послідовно вузлові точки $t=1,2,...,k-1,k$. Як випливає з умов неперервності

$$\begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix}\begin{pmatrix} x^0(1) \\ y(0) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix}\begin{pmatrix} e^{A(y(0))} & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}=\begin{pmatrix} e^{A(y(0))} & 0 \\ C(1)e^{A(y(0))} & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix},$$

$$\begin{pmatrix} x^2(2) \\ y(2) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(2) & 1 \end{bmatrix}\begin{pmatrix} x^1(2) \\ y(1) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(2) & 1 \end{bmatrix}\begin{bmatrix} e^{A(y(1))} & 0 \\ 0 & 1 \end{bmatrix}\begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix}=\begin{bmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{bmatrix}\times$$

$$\times\begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix}=\begin{bmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix}\begin{pmatrix} e^{A(y(0))} & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}=\begin{bmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{bmatrix}\times$$

$$\times\begin{pmatrix} e^{A(y(0))} & 0 \\ C(1)e^{A(y(0))} & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix},\ \dots, \qquad (3.37)$$

$$\begin{pmatrix} x^k(k) \\ y(k) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(k) & 1 \end{bmatrix}\begin{pmatrix} x^{k-1}(k) \\ y(k-1) \end{pmatrix}=\begin{bmatrix} 1 & 0 \\ C(k) & 1 \end{bmatrix}\begin{bmatrix} e^{A(y(k-1))} & 0 \\ 0 & 1 \end{bmatrix}\begin{pmatrix} x^{k-1}(k-1) \\ y(k-1) \end{pmatrix}=$$

$$=\begin{bmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ C(k-1) & 1 \end{bmatrix}\begin{pmatrix} e^{A(y(k-2))} & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x^{k-2}(k-2) \\ y(k-2) \end{pmatrix}=\begin{bmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{bmatrix}\times$$

$$\times\begin{pmatrix} e^{A(y(k-2))} & 0 \\ C(k-1)e^{A(y(k-2))} & 1 \end{pmatrix}\begin{pmatrix} x^{k-2}(k-2) \\ y(k-2) \end{pmatrix}=...=\begin{pmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{pmatrix}\begin{pmatrix} e^{A(y(k-2))} & 0 \\ C(k-1)e^{A(y(k-2))} & 1 \end{pmatrix}...\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}=$$

$$=\begin{pmatrix} \prod_{j=1}^{k} e^{A(y(k-j))} & 0 \\ \sum_{j=1}^{k} C(k-j+1)\prod_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}.$$

Враховуючи (3.36), маємо

$$\begin{pmatrix} x^k(t) \\ y(k) \end{pmatrix}=\begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix}\begin{pmatrix} \prod_{j=1}^{k} e^{A(y(k-j))} & 0 \\ \sum_{j=1}^{k} C(k-j+1)\prod_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix}\begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}, \qquad (3.38)$$

$k\le t<k+1$.

Таким чином, фундаментальна матриця розв’язків системи (3.23), (3.24) складається з двох частин, неперервної і дискретної. Як випливає (3.38), фундаментальна матриця рішень гібридної системи (3.23), (3.24) на проміжку $k \le t < k+1$ має вигляд $\Phi(t) = Z_k(t) Z(k)$, де $Z_k(t)$ і $Z(k)$ мають вигляд (3.31), (3.32) відповідно.

Наслідок 3.2. Для гібридної системи (3.23), (3.24) розв’язок задачі Коші з початковими умовами $x(0) = x_0$, $y(t) \equiv y_0$, $-1 \le t \le 0$ на проміжку $k \le t < k+1$ має вигляд

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \Phi(t) \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \tag{3.39}$$

де фундаментальна матриця $\Phi(t)$ задається у вигляді (3.30).

Повернемося до системи (3.17)-(3.20). Запишемо її у вигляді гібридної моделі (3.23), (3.24), динаміка станів якої подається вектором $(x_1(t),..., x_N(t), \ldots, u(t), y(t))$ що складається з векторної функції $x(t) = (x_1(t),..., x_N(t), x_{N+1}(t))$ ($N+1$ змінна визначає стан потенціалу $u(t)$ виконавця), $t \ge 0$. Квадратна матриця A з постійними коефіцієнтами на кожному з інтервалів $t \in [k, k+1)$, $k = 0,1,2,...$, розмірності $(N+1 \times N+1)$ має вигляд

$$A = \begin{pmatrix} -1/b(t) & 0 & \ldots & 0 & 0 \\ 0 & -1/b(t) & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \\ 0 & 0 & \ldots & -1/b(t) & 0 \\ w_1(t)/N & w_2(t)/N & \ldots & w_N(t)/N & -1 \end{pmatrix}, \tag{3.40}$$

$y(k) = b(k) \in R^1$, $k = 0,1,2,...$, C- вектор з N елементів, формальний результат застосування якого у виразі $C(k)x(k)$, $k = 0,1,2,...$, визначається величиною $\sum\limits_{\substack{i=1 \\ w_i(t) > w_i(t-1)}}^{N} 2^{i-1} - \sum\limits_{\substack{i=1 \\ w_i(t) < w_i(t-1)}}^{N} 2^{i-1}$.

Без обмеження загальності моделі будемо вважати початковим моментом функціонування системи $t = 0$ момент початку виконання першої задачі. Тоді розв’язок задачі Коші для системи (3.17)-(3.20) з початковими умовами

$x(0) = (x_1(0),..., x_N(0),0) = x_0$, $y(t) = y_0 \equiv 0$, $-1 \le t \le 0$ на проміжку $k \le t < k+1$ буде мати вигляд

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \Phi(t) \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \tag{3.41}$$

де фундаментальна матриця $\Phi(t)$ задається у вигляді (3.30).

3.4. Метод перетворення області допустимих розв'язків з урахуванням близькості обмежень задачі

Пошук оптимального або близького до оптимального розв'язку в задачах дослідження операцій на основі методів математичного програмування здійснюється з урахуванням системи обмежень, яка визначає область допустимих розв'язків (ОДР). В задачах лінійного програмування система обмежень визначається, як правило, системою лінійних нерівностей. За невеликої кількості обмежень область може бути легко побудована, а оптимальний розв'язок – знайдений шляхом перебору значень цільової функції в усіх симплексах (вершинах) відповідного багатогранника.

Зрозуміло, що для розв'язування задач математичного (в тому числі, лінійного) програмування існує багато ефективних методів, які дозволяють отримати оптимальний або близький до нього розв'язок, не проводячи аналізу складу системи обмежень. В той же час система обмежень часто містить сукупності незалежних співвідношень, які є близькими, що за умов великої кількості обмежень суттєво знижує ефективність роботи алгоритмів пошуку.

Близькі обмеження незначним чином впливають на вигляд області допустимих розв'язків, що дозволяє розглядати задачу оптимізації з урахуванням обмеженої кількості умов, які визначають деяке наближення початкової області ДР. Зрозуміло, що говорити про оптимальність отриманого при цьому розв'язку для вихідної задачі дуже складно.

Розглянемо стандартну задачу лінійного програмування, записану у векторно-матричному вигляді

$$\max c^T x \tag{3.42}$$

при обмеженнях

$$Ax \le b;\ \ x \ge 0, \tag{3.43}$$

де A - матриця виробничих обмежень розміру $m \times n$, b - вектор-стовбець вільних членів (розміру m), x - вектор змінних (розміру n), $c = (c_1, c_2, ..., c_n)$ - вектор коефіцієнтів цільової функції. Значення елементів вектору b визначають величини m видів ресурсів, що використовуються у технологічному процесі. Тому, без обмеження загальності, будемо вважати їх невід'ємними.

Введемо позначення: $A_i = (a_{i1}, a_{i2}, ..., a_{in})^T$ - вектор коефіцієнтів у лівій частині обмеження з номером i, $\overline{A}_i = (a_{i1}, a_{i2}, ..., a_{in}, a_{in+1})^T$ - розширений вектор, що містить окрім коефіцієнтів лівої частини i-тої нерівності значення вільного члена правої частини $a_{in+1} = b_i$, $i = \overline{1, m}$.

Для групування нерівностей будемо досліджувати взаємозв'язки між ними, використовуючи такі означення.

Означення 3.8. Два обмеження системи лінійних нерівностей (3.43) з номерами i та j, $1 \le i, j \le m$, назвемо γ-слабкозв'язаними, якщо для величини

$$v_{ij} = (A_i, A_j) \big/ (\|A_i\| \|A_j\|),\ 1 \le i, j \le m, \tag{3.44}$$

справедливе співвідношення

$$v_{ij} \le \gamma,\ 0 \le \gamma < 1. \tag{3.45}$$

Тут $(A_i, A_j) = \sum_{s=1}^{n} a_{is} a_{js}$ - скалярний добуток векторів A_i, A_j, $1 \le i, j \le m$.

Величина параметра γ дозволяє задати рівень взаємозв'язку двох нерівностей. Легко помітити, що значення v_{ij} співпадає зі значенням косинуса куга між векторами градієнтів A_i, A_j, $1 \le i, j \le m$, ($v_{ij} = \cos\varphi_{ij} = \cos\varphi(A_i, A_j)$), і при невеликих значеннях γ дві нерівності з номерами i та j, $1 \le i, j \le m$, суттєво відрізняються.

Означення 3.9. Два обмеження системи лінійних нерівностей (3.43) з номерами i та j, $1 \le i, j \le m$, назвемо γ-сильнозв'язаними, якщо для величини v_{ij} виду (3.44) справедливе співвідношення

$$v_{ij} \ge \gamma,\ 0 < \gamma \le 1. \tag{3.46}$$

Означення 3.10. Два обмеження системи лінійних нерівностей (3.43) з номерами i та j, $1 \le i, j \le m$, назвемо $\bar{\gamma}$-слабкозв'язаними, якщо для величини

$$\bar{v}_{ij} = (\bar{A}_i, \bar{A}_i) \big/ (\|\bar{A}_i\| \|\bar{A}_j\|),\ 1 \le i, j \le m, \tag{3.47}$$

справедливе співвідношення

$$\bar{v}_{ij} \le \bar{\gamma},\ 0 \le \bar{\gamma} < 1. \tag{3.48}$$

Означення 3.11. Два обмеження системи лінійних нерівностей (3.43) з номерами i та j, $1 \le i, j \le m$, назвемо $\bar{\gamma}$- сильнозв'язаними, якщо для величини $\bar{v}_{ij}$ виду (3.47) справедливе співвідношення

$$\bar{v}_{ij} \ge \bar{\gamma},\ 0 < \bar{\gamma} \le 1. \tag{3.49}$$

Зрозуміло, що поняття $\bar{\gamma}$-слабкозв'язаних і $\bar{\gamma}$-сильнозв'язаних обмежень, як і у випадку γ-слабкозв'язаних і γ-сильнозв'язаних обмежень, визначаються величиною кута між векторами ($\bar{v}_{ij} = \cos \bar{\varphi}_{ij} = \cos \bar{\varphi}(\bar{A}_i, \bar{A}_j)$), обчисленою для розширених векторів $\bar{A}_i, \bar{A}_j$, $1 \le i, j \le m$.

Розглянемо довільні обмеження з номерами i та j, $1 \le i, j \le m$. Ступінь їх $\bar{\gamma}$-зв'язності $\bar{v}_{ij}$ визначається величиною

$$\bar{v}_{ij} = \max \bar{\gamma} = \bar{\gamma}^0 = \sum_{s=1}^{n+1} a_{is} a_{js} \Big/ (\|\bar{A}_i\| \|\bar{A}_j\|),$$

для якої, враховуючи невід'ємність елементів вектору b, нескладно отримати таку оцінку:

$$\bar{\gamma}^0 = \left(\sum_{s=1}^{n} a_{is} a_{js} + a_{in+1} a_{jn+1} \right) \Big/ (\|\bar{A}_i\| \|\bar{A}_j\|) = \sum_{s=1}^{n} a_{is} a_{js} \Big/ \left((\sum_{s=1}^{n+1} a_{is}^2)^{1/2} (\sum_{s=1}^{n+1} a_{js}^2)^{1/2} \right) +$$

$$+ a_{in+1} a_{jn+1} \Big/ \left((\sum_{s=1}^{n+1} a_{is}^2)^{1/2} (\sum_{s=1}^{n+1} a_{js}^2)^{1/2} \right) \le \left(\sum_{s=1}^{n} a_{is} a_{js} \right) \Big/ \left((\sum_{s=1}^{n} a_{is}^2)^{1/2} (\sum_{s=1}^{n} a_{js}^2)^{1/2} \right) +$$

$$+a_{in+1}a_{jn+1}\Bigg/\left((\sum_{s=1}^{n+1}a_{is}^2)^{1/2}(\sum_{s=1}^{n+1}a_{js}^2)^{1/2}\right)\le \nu_{ij}+a_{in+1}a_{jn+1}\Bigg/\left((\sum_{s=1}^{n}a_{is}^2)^{1/2}(\sum_{s=1}^{n}a_{js}^2)^{1/2}\right)$$

Звідси маємо

$$\bar{\nu}_{ij}\le \nu_{ij}+a_{in+1}a_{jn+1}\big/(\|A_i\|\|A_j\|)\,. \tag{3.50}$$

Крім цього, справедливе

Твердження 3.1. Припустимо, що довільні обмеження з номерами i та j, $1\le i,j\le m$, системи нерівностей (3.43) є γ-сильнозв'язаними, тобто $\nu_{ij}\ge\gamma$. Тоді за умови $\bar{\nu}_{ij}(\beta^0)\ge\nu_{ij}$, де $\bar{\nu}_{ij}(\beta)=\left((A_i,A_j)+\beta\right)\Big/\left(\sqrt{\|A_i\|^2+\beta^2}\sqrt{\|A_j\|^2+1}\right)$, $a_{in+1}=\beta$, $a_{jn+1}=1$, $\beta^0=\|A_i\|^2\big/(A_i,A_j)$, існує значення $\bar{\gamma}\ge\gamma$, для якого ці обмеження будуть $\bar{\gamma}$-сильнозв'язаними, тобто $\bar{\nu}_{ij}\ge\bar{\gamma}$.

Доведення. Розглянемо довільні обмеження з номерами i та j, $1\le i,j\le m$, системи нерівностей (3.43). Не обмежуючи загальності, будемо вважати, що $a_{jn+1}=1$, що легко отримати, розділивши усі елементи вектора $\bar{A}_j$ на величину правої частини нерівності.

Позначимо величину $a_{in+1}=\beta$ і проведемо дослідження функції $\bar{\nu}_{ij}(\beta)=\left((A_i,A_j)+\beta\right)\Big/\left(\sqrt{\|A_i\|^2+\beta^2}\sqrt{\|A_j\|^2+1}\right)$. Нескладно перевірити, що вона має єдину точку екстремуму (максимуму) $\beta^0=\|A_i\|^2\big/(A_i,A_j)$.

За умовою твердження обмеження є γ-сильнозв'язаними. Розглянемо граничний випадок $\gamma=\nu_{ij}$. Тоді за умови $\bar{\nu}_{ij}(\beta^0)\ge\nu_{ij}$, обмеження з номерами i та j будуть $\bar{\gamma}$-сильнозв'язаними рівня $\bar{\gamma}=\gamma$. У випадку виконання нерівності $\bar{\nu}_{ij}(\beta^0)>\nu_{ij}$ існує значення $\bar{\gamma}>\gamma$, для якого ці обмеження будуть $\bar{\gamma}$-сильнозв'язаними. Твердження доведено.

В задачах лінійного програмування з великою кількістю обмежень досить часто має місце близькість або лінійна залежність обмежень, що суттєво знижує ефективність роботи алгоритмів пошуку оптимального розв'язку. Наведені вище означення дозволяють встановити зв'язаність обмежень і запропонувати спосіб їх модифікації, замінюючи пару нерівностей на одне нове. Розглянемо послідовність перетворень для найпростішого випадку ЗЛП (3.42), (3.43) при m лінійних обмеженнях:

- задаємо значення величини γ, яке визначає ступінь сильної зв'язаності пар обмежень;

- для всіх пар нерівностей, для яких значення $v_{ij} \geq \gamma$, $1 \leq i, j \leq m$, будуємо нове обмеження у вигляді $\sum_{k=1}^{n} p_k x_k \leq r$, де коефіцієнти $p_k, k = \overline{1,n}, r$ визначаються з рівняння гіперплощини, вектор нормалі до якої співпадає з вектором суми векторів $(a_{i1}, a_{i2},..., a_{in})^T$ і $(a_{j1}, a_{j2},..., a_{jn})^T$, та яка проходить через гіпермножину точок перетину гіперплощин $\sum_{k=1}^{n} a_{ik} x_k = b_i$ і $\sum_{k=1}^{n} a_{jk} x_k = b_j$, $1 \leq i, j \leq m$.

Для найпростішого випадку ЗЛП (3.42), (3.43) при $n = 2$ і m лінійних обмеженнях процедура перетворення області допустимих розв'язків передбачає побудову нового обмеження у вигляді $px_1 + qx_2 \leq r$, де коефіцієнти p, q, r визначаються з рівняння прямої, вектор нормалі якої співпадає з вектором суми векторів $(a_{i1}, a_{i2})^T$ і $(a_{j1}, a_{j2})^T$, та яка проходить через точку перетину прямих $a_{i1}x_1 + a_{i2}x_2 = b_i$ і $a_{j1}x_1 + a_{j2}x_2 = b_j$, $1 \leq i, j \leq m$.

Слід зауважити, що область допустимих рішень задачі лінійного програмування при цьому не зменшується, а у випадку виконання умов $\bar{v}_{ij} \geq v_{ij} \geq \gamma$, $1 \leq i, j \leq m$, є перетином областей, що визначаються усіма нерівностями, крім i-ї та j-ї, та області, яка визначається новою нерівністю $\sum_{k=1}^{n} p_k x_k \leq r$, з урахуванням вимог $x_k \geq 0, k = \overline{1,n}$.

На жаль, подібне перетворення області допустимих рішень не гарантує близькості заново отриманого розв'язку до оптимального розв'язку початкової задачі.

■ Метод перетворення ОДЗ нечіткої задачі лінійного програмування на основі оцінки близькості обмежень.

Наведений вище підхід для перетворення ОДЗ можна застосувати для пошуку розв'язків нечітких задач лінійного програмування з нечітко заданими ресурсами:

$$\sum_{j=1}^{n} c_j x_j \to \max , \tag{3.51}$$

з нечіткими обмеженнями

$$\sum_{j=1}^{n} a_{ij} x_j \le \tilde{b}_i , \quad i = \overline{1,m}, \ \ x_j \ge 0, \ j = \overline{1,n} \ , \tag{3.52}$$

де праві частини обмежень (3.52) подаються у вигляді правих нечітких трикутних чисел $\tilde{b}_i = (b_i, b_i, b_i + b_i^0)$, $b_i^0 \ge 0$, $i = \overline{1,m}$. Допустимі відхилення $b_i^0 \ge 0$, $i = \overline{1,m}$, визначають величини граничних змін ресурсів моделі.

З урахуванням понять нечітких трикутних чисел, задача лінійного програмування (3.51)-(3.52) може бути переписана як задача оптимізації цільової функції (3.51) з обмеженнями

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i + b_i^0 - \lambda b_i^0 , \ \ \lambda \in [0,1], \ \ i = \overline{1,m}, \ \ x_j \ge 0, \ j = \overline{1,n} . \tag{3.53}$$

У цьому випадку нечітка задача лінійного програмування (3.51), (3.53) переписується у формі оптимізаційної задачі: знайти значення параметра $\lambda \in [0,1]$, яке є розв'язком задачі лінійного програмування

$$\max_x \lambda$$

$$\lambda(U - L) - \sum_{j=1}^{n} c_j x_j + L \le 0,$$

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i + b_i^0 - \lambda b_i^0 , i = \overline{1,m}, x \ge 0, 0 \le \lambda \le 1. \tag{3.54}$$

Тут $L = \min(Z_l, Z_u)$, $U = \max(Z_l, Z_u)$, $U \ge L$, Z_l, Z_u - оптимальні значення функцій цілі задач (3.51), (3.53) при $\lambda = 1$ і $\lambda = 0$ відповідно, а оптимальне значення цільової функції моделі (3.51), (3.53) буде належати інтервалу $[L,U]$ при деякому значенні параметра $\lambda^* \in [0,1]$.

Запропонована вище послідовність для перетворення системи обмежень повинна враховувати подання величин ресурсів у вигляді правих трикутних нечітких чисел, $\tilde{b}_i = (b_i, b_i, b_i + b_i^0)$, $i = \overline{1,m}$.

Припустимо, що задача оптимізації має вигляд (3.42), (3.43) при m лінійних обмеженнях. Тоді схема перетворення області допустимих розв’язків набуває такого вигляду:

- задаємо значення величини γ, що визначає ступінь сильної зв’язаності пар обмежень;

- для всіх пар нерівностей, для яких значення $v_{ij} \ge \gamma$, $1 \le i, j \le m$, будуємо нове обмеження у вигляді $\sum_{k=1}^{n} p_k x_k \le s - \lambda v_{ij}$, $0 \le \lambda \le 1$, де коефіцієнти $p_k, k = \overline{1,n}, s$ визначаються з рівняння гіперплощини, вектор нормалі якої співпадає з вектором суми векторів $(a_{i1}, a_{i2}, ..., a_{in})^T$ і $(a_{j1}, a_{j2}, ..., a_{jn})^T$ та яка проходить через множину точок перетину гіперплощин $\sum_{k=1}^{n} a_{ik} x_k = b_i + b_i^0$ і $\sum_{k=1}^{n} a_{jk} x_k = b_j + b_j^0$, $1 \le i, j \le m$. Отримане в новому обмеженні значення $s - \lambda v_{ij}$ відповідає нечітко заданій величині ресурсу, яка подається правим нечітким трикутним числом $(s - v_{ij}, s - v_{ij}, s)$.

Приклад. Продемонструємо результат застосування даного алгоритму для нечіткої ЗЛП такого вигляду:

$$
\begin{gathered}
\max\ x_2 \\
2x_1 + 9x_2 \le 19 - \lambda_1, \\
3x_1 + 4x_2 \le 14 - 2\lambda_2, \\
-x_1 + x_2 \le 2 - \lambda_3, \\
x_1 \ge 0, x_2 \ge 0, \lambda_i \in [0,1], i = \overline{1,3}.
\end{gathered}
\tag{3.55}
$$

Використання для розв'язування задачі (3.55) описаного підходу у вигляді оптимізаційної задачі (3.54) дає розв'язок $x_1 = 0.45, x_2 = 1{,}96$, $\lambda_1 = \lambda_2 = \lambda_3 = 0{,}49.$

Перетворення перших двох нерівностей до вигляду $5x_1 + 13x_2 \le 30{,}9 - 0{,}911\lambda_4$, $\lambda_4 \in [0,1]$, дозволяє отримати оптимальний розв'язок нової ЗЛП $x_1 = 0.61, x_2 = 2{,}1$, $\lambda_3 = \lambda_4 = 0{,}4995.$ В даному випадку, розв'язок залишається достатньо близьким до розв'язку початкової задачі, зберігаючи при цьому рівень нечіткості розв'язку.

■Лінеаризація добутку змінних

До появи системи обмежень високої розмірності може привести лінеаризація добутку змінних в оптимізаційних моделях. Якщо математична модель містить квадратичний вираз вигляду $x_i x_j$, $i, j = \overline{1,n}$, де $x_i, x_j \in \{0,1\}$, то для нього можна запропонувати еквівалентне лінійне пере формулювання, що призводить до розгляду лінійної задачі оптимізації з багатьма обмеженнями.

Для цього введемо нову бульову змінну y_{ij}, таку, що $y_{ij} = x_i x_j$, тобто $y_{ij} = 1$тоді і лише тільки тоді, коли $x_i = 1$ і $x_j = 1$. Іншими словами, якщо $y_{ij} = 1$, то $x_i = 1$; якщо $y_{ij} = 1$, то $x_j = 1$ та якщо $x_i = 1$ і $x_j = 1$, то $y_{ij} = 1$.

Застосовуючи перше та друге правило, отримуємо три наступні нерівності:

$$1 - x_i \le 1 - y_{ij},\ 1 - x_j \le y_{ij},\ 1 - y_{ij} \le 1 - x_i + 1 - x_j, \qquad (3.56)$$

або, у більш спрощеному вигляді:

$$y_{ij} \le x_i,\ y_{ij} \le x_j,\ x_i + x_j - y_{ij} \le 1,\ x_i, x_j, y_{ij} \in \{0,1\},\ i, j = \overline{1,n}. \qquad (3.57)$$

Цікавим і корисним спостереженням з точки зору лінійного програмування є те, що якщо відмовитися від бульовості змінних y_{ij} і перейти до неперервних значень з відрізку [0,1], то нерівності (3.57) будуть виконуватися лише у випадку, коли $y_{ij} = 0$ або $y_{ij} = 1$.

3.5. Пошук розв'язків за наявності систем альтернативних обмежень

Припустимо, що розглядається традиційна задача лінійного програмування:

$$\max \sum_{j=1}^{n} c_j x_j \tag{3.58}$$

при обмеженнях у вигляді системи нерівностей розмірності *m₁*

$$\begin{aligned} & a_{11}^1 x_1 + a_{12}^1 x_2 + \ldots + a_{1n}^1 x_n \le b_1^1; \\ & a_{21}^1 x_1 + a_{22}^1 x_2 + \ldots + a_{2n}^1 x_n \le b_2^1; \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ & a_{m_1 1}^1 x_1 + a_{m_1 2}^1 x_2 + \ldots + a_{m_1 n}^1 x_n \le b_{m_1}^1; \end{aligned} \tag{3.59}$$

або системи нерівностей розмірності m_2

$$\begin{aligned} & a_{11}^2 x_1 + a_{12}^2 x_2 + \ldots + a_{1n}^2 x_n \le b_1^2; \\ & a_{21}^2 x_1 + a_{22}^2 x_2 + \ldots + a_{2n}^2 x_n \le b_2^2; \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ & a_{m_2 1}^2 x_1 + a_{m_2 2}^2 x_2 + \ldots + a_{m_2 n}^2 x_n \le b_{m_2}^2; \end{aligned} \tag{3.60}$$

$x = (x_1, x_2, ..., x_n) \ge 0$.

Ця задача може бути зведена до оптимізаційної задачі, у котрій використовується задана кількість обмежень. Дійсно, введемо $m_1 + m_2$ штучних змінних $y_i^1 \ge 0,\ y_i^1 \le 1, y_i^1 \in Z,\ \ i = \overline{1, m_1}$, $y_i^2 \ge 0,\ y_i^2 \le 1, y_i^2 \in Z,\ \ i = \overline{1, m_2}$ і задамо деяке число $M >> 0$. В результаті отримуємо модель дискретного лінійного програмування з функцією цілі (3.58) та набором обмежень у вигляді

$$\begin{aligned} & a_{11}^1 x_1 + a_{12}^1 x_2 + \ldots + a_{1n}^1 x_n \le b_1^1 + (1 - y_1^1)M; \\ & a_{21}^1 x_1 + a_{22}^1 x_2 + \ldots + a_{2n}^1 x_n \le b_2^1 + (1 - y_2^1)M; \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ & a_{m_1 1}^1 x_1 + a_{m_1 2}^1 x_2 + \ldots + a_{m_1 n}^1 x_n \le b_{m_1}^1 + (1 - y_{m_1}^1)M; \end{aligned} \tag{3.61}$$

$$\begin{aligned} & a_{11}^2 x_1 + a_{12}^2 x_2 + \ldots + a_{1n}^2 x_n \le b_1^2 + (1 - y_1^2)M; \\ & a_{21}^2 x_1 + a_{22}^2 x_2 + \ldots + a_{2n}^2 x_n \le b_2^2 + (1 - y_2^2)M; \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ & a_{m_2 1}^2 x_1 + a_{m_2 2}^2 x_2 + \ldots + a_{m_2 n}^2 x_n \le b_{m_2}^2 + (1 - y_{m_2}^2)M; \end{aligned} \tag{3.62}$$

$$\sum_{i=1}^{m_1} y_i^1 + \sum_{i=1}^{m_2} y_i^2 = m_1(1 - \sum_{i=1}^{m_2} y_i^2 \Big/ m_2) + m_2(1 - \sum_{i=1}^{m_1} y_i^1 \Big/ m_1)\,; \tag{3.63}$$

$y_i^1 \geq 0,\ y_i^1 \leq 1, y_i^1 \in Z,\ \ i = \overline{1, m_1}$, або $y_i^2 \geq 0,\ y_i^2 \leq 1, y_i^2 \in Z,\ \ i = \overline{1, m_2}$.

Нескладно перевірити, що у випадку, коли $\sum_{i=1}^{m_1} y_i^1 = m_1$ (тобто $y_i^1 = 1,\ i = \overline{1, m_1}$), в задачі діють обмеження системи (3.61). Рівність (3.63) при цьому має вигляд $\sum_{i=1}^{m_1} y_i^1 + \sum_{i=1}^{m_2} y_i^2 = m_1$. У протилежному випадку, коли $\sum_{i=1}^{m_2} y_i^2 = m_2$ (тобто $y_i^2 = 1,\ i = \overline{1, m_2}$), встановлюємо, що будуть задіяні обмеження системи (2.62), а рівність (2.63) буде мати вигляд $\sum_{i=1}^{m_1} y_i^1 + \sum_{i=1}^{m_2} y_i^2 = m_2$.

Застосування даного підходу для розв'язання нечіткої задачі лінійного програмування з цільовою функцією (3.58) та з нечіткими ресурсними обмеженнями у вигляді системи нерівностей розмірності m_1

$$\begin{array}{l} a_{11}^1 x_1 + a_{12}^1 x_2 + ... + a_{1n}^1 x_n \leq \tilde{b}_1^1; \\ a_{21}^1 x_1 + a_{22}^1 x_2 + ... + a_{2n}^1 x_n \leq \tilde{b}_2^1; \\ \text{..............................} \\ a_{m_1 1}^1 x_1 + a_{m_1 2}^1 x_2 + ... + a_{m_1 n}^1 x_n \leq \tilde{b}_{m_1}^1; \end{array} \tag{3.64}$$

або системи нерівностей розмірності m_2

$$\begin{array}{l} a_{11}^2 x_1 + a_{12}^2 x_2 + ... + a_{1n}^2 x_n \leq \tilde{b}_1^2; \\ a_{21}^2 x_1 + a_{22}^2 x_2 + ... + a_{2n}^2 x_n \leq \tilde{b}_2^2; \\ \text{..............................} \\ a_{m_2 1}^2 x_1 + a_{m_2 2}^2 x_2 + ... + a_{m_2 n}^2 x_n \leq \tilde{b}_{m_2}^2; \end{array} \tag{3.65}$$

$x = (x_1, x_2, ..., x_n) \geq 0$, є надто громіздким.

В якості альтернативи розглянемо таку схему. Аналогічно викладеному вище у підрозділі 1.2.2 матеріалу задамо конкретний рівень Z, який визначає границю оптимального розв'язку. Це дозволяє сформулювати дві задачі оптимізації виду (1.50)

$$\max_{x \geq 0} \lambda^1 \tag{3.66}$$

за умов

$$\sum_{j=1}^{n} a_{ij}^1 x_j \leq b_i^1 + \overline{b}_i^1 - \lambda^1 \overline{b}_i^1, \ \ i = \overline{1, m_1}, \ x \geq 0, \tag{3.67}$$

і

$$\max_{x \geq 0} \lambda^2 \tag{3.68}$$

за умов

$$\sum_{j=1}^{n} a_{ij}^2 x_j \leq b_i^2 + \overline{b}_i^2 - \lambda^2 \overline{b}_i^2, \ \ i = \overline{1, m_2}, \ x \geq 0, \tag{3.69}$$

де $\overline{b}_i^1$, $i = \overline{1, m_1}$, $\overline{b}_i^2$, $i = \overline{1, m_2}$, - величини допустимих порушень (відхилень) в ресурсних обмеженнях систем (3.64), (3.65), відповідно.

Якщо позначити оптимальний розв'язок задачі (3.66)-(3.67) у вигляді вектору (λ^1, x^{1O}), а оптимальний розв'язок задачі (3.68)-(3.69) - у вигляді вектору (λ^2, x^{2O}), то x^{1O}, x^{2O} будуть оптимальними розв'язками нечітких задач (3.58), (3.64) і (3.58), (3.65) відповідно.

Параметри λ^1, λ^2 визначають загальний гарантований рівень нечітко заданих ресурсів і у якості розв'язку задачі (3.58), (3.64), (3.65) можна вибрати вектор $x = x^{pO}$, де

$$p = \begin{cases} 1, & \textit{если } (\lambda^1 > \lambda^2) \vee ((\lambda^1 = \lambda^2) \Lambda (c^T x^{1O} \geq c^T x^{2O})), \\ 2, & \textit{если } (\lambda^1 < \lambda^2) \vee ((\lambda^1 = \lambda^2) \Lambda (c^T x^{1O} < c^T x^{2O})) \end{cases}. \tag{3.70}$$

Іншими словами, оптимальним розв'язком нечіткої лінійної задачі оптимізації може бути обраний розв'язок, що відповідає більш високому рівню нечітко заданих ресурсів.

Якщо рівень λ, який відповідає меншому гарантованому рівню ресурсів, є допустимим для особи, що приймає рішення (ОПР), то кінцевий вибір оптимального розв'язку задачі може бути проведений на основі порівняння значень цільових функцій $F_1 = c^T x^{1O}$ і $F_2 = c^T x^{2O}$.

У загальному випадку нечіткої лінійної задачі оптимізації з $r, (r > 2)$, взаємно виключними підсистемами обмежень пошук оптимальних розв'язків можна здійснювати шляхом вибору ефективного (компромісного) розв'язку на множині пар $\{(\lambda^s, c^T x^{sO})\}$, $s = \overline{1,r}$, де (λ^s, x^{sO}) - оптимальні розв'язки задач вигляду (3.66)-(3.67) або (3.68)-(3.69), $s = \overline{1,r}$.

3.6. Висновки по третьому розділу

Третій розділ присвячено розгляду моделей і методів, які застосовуються при розв'язанні нечітких задач оптимального розподілу часового ресурсу. Сформульовано нечітку задачу про рюкзак як засіб розподілу нечітко визначеного часового ресурсу. Описано задачу складання розкладу з нечітко заданими інтервалами налаштувань.

Особливу увагу приділено задачі теорії розкладів з нечітким вимірюванням часу. Визначено поняття структурованих нечітких чисел, які використовуються для опису вимірювання відліку часу. Розглянуто нечітку задачу про рюкзак з нечітко заданими термінами виконання.

На основі сильної та слабкої зв'язності обмежень лінійних задач оптимізації розроблено метод перетворення області допустимих розв'язків з урахуванням близькості обмежень. Проілюстровано вплив запропонованого методу на результати розв'язання нечітких ЗЛП. Розроблено схему розв'язання нечітких задач лінійного програмування за наявності систем альтернативних обмежень.

Побудовано гібридну модель динаміки процесу обробки сукупності завдань, отримано твердження про фундаментальну матрицю її розв'язків.

РОЗДІЛ 4
ПРИКЛАДИ ПРАКТИЧНОГО РОЗВ'ЯЗУВАННЯ ЗАДАЧ ОПТИМАЛЬНОГО РОЗПОДІЛУ ЧАСОВОГО РЕСУРСУ

Серед прикладних задач оптимального розподілу часового ресурсу важливе місце займають задачі, що виникають на різних етапах організації та проведення навчання у закладах освіти. Досить давно поставлена і привертає увагу багатьох дослідників задача автоматичного формування розкладу навчальних занять [88]. Серед нових задач – планування, проведення й опрацювання результатів тестування. Сучасне тестування являє собою комплекс стандартизованих методів вимірювання параметрів, через які визначають рівень підготовки людини і відповідність освітнім стандартам у конкретній галузі знань, в якому широко використовуються математичні методи та сучасні комп'ютерні технології опрацювання даних. Об'єктивний контроль знань, вмінь і навичок вдається здійснити при критеріально-орієнтованій інтерпретації тестування.

Із впровадженням персональних комп'ютерів у сферу освіти почалась розробка систем контролю знань. В них намагаються реалізувати принципи адаптивності за різними критеріями, але разом із тим, залишається ряд тенденцій, які вже декілька десятиліть супроводжують процеси створення та використання таких систем. Зокрема:

– перелік питань складається викладачем, виходячи із суб'єктивних переконань;
– питання найчастіше мають замкнений тестовий характер;
– для отримання оцінки необхідно відповісти на певну, наперед визначену кількість питань;
– трудомісткість реалізації адаптивних елементів призводить до відсутності практичних розробок з реалізованими теоретичними напрацюваннями.

Такий підхід призводить до створення недостатньо ефективних систем, використання яких тим не менше має дві переваги: менший час на контроль знань та зменшення суб`єктивізму оцінювання. Разом із тим, в автоматизованих системах навчання та контролю знань залишається ще ряд недоліків. Серед таких проблем необхідно виділити майже повну відсутність технологічних елементів, які забезпечили б дійсно ефективний процес навчання та контролю знань.

Тому питання комп’ютерного контролю знань, є об’єктом неабиякого інтересу для викладачів навчальних закладів і розробників засобів реалізації такого контролю.

4.1. Розклад занять академічних груп і викладачів в межах аудиторного фонду навчального закладу протягом типового тижня

Розклад занять у навчальному закладі будь-якого рівня повинен передбачати необхідну кількість випадків одночасного перебування кожного студентського формування (потоку, групи, підгрупи) і відповідного викладача в одній з можливих аудиторій (лабораторій). При цьому навчальна дисципліна, з якої здійснюється викладання в конкретній аудиторії (лабораторії) на конкретній парі, визначається складом присутніх учасників навчального процесу: академічних груп і викладача. Якщо викладач здійснює навчальний процес у тому самому студентському формуванні з декількох навчальних дисциплінах, то їх конкретизація для кожного аудиторного заняття може бути покладена на самого викладача.

У такому аспекті задача складання розкладу занять вписується в клас задач маршрутизації об'єктів, у якості яких виступають академічні групи та викладачі навчального підрозділу освітнього закладу. В рамках даного дослідження задача формування розкладу розглядалась як задача дискретної оптимізації в умовах обмеженості ресурсів аудиторного фонду, що властиво при організації навчання на навчально-підготовчих курсів та при проведенні підвищення кваліфікації.

Така задача виникає тоді, коли відома конкретна кількість викладачів та визначена кількість аудиторій. Існує багато різновидів даної задачі. Розглянемо один із можливих варіантів.

Нехай маємо M груп студентів, P викладачів та S аудиторій. Введемо двійкові змінні x_{ijkt}, що задовольняють наступним умовам:

x_{ijkt}=1, якщо j-й викладач читає лекцію для i-ої групи в k-й аудиторії в t-й день тижня; x_{ijkt}=0, - в іншому випадку; $i=\overline{1,M}$; $j=\overline{1,P}$; $k=\overline{1,S}$; t=1,2,… .

Вважаємо, що з шести робочих днів тижня лекційними є лише q днів. ($1 \le q \le 6$). Максимальна кількість лекційних годин в день, згідно з розкладом, дорівнює h. Кожна лекція може тривати або 1 годину, або 2 години (неподільною одиницею вимірювання лекційного часу є година). При використанні наявних аудиторій для проведення лекцій різними викладачами в різних групах виникають різні обмеження [89]. Так, наприклад, викладач не може в принципі проводити лекцію в аудиторії, котра не обладнана всім необхідним для подання матеріалів відповідного професійного спрямування.

Припустимо, що в i-й групі та j-му викладачеві відповідає вектор

$$O_{ij}=(O_{ij1},O_{ij2},...,O_{ijS}), \tag{4.1}$$

в якому якого k-ий елемент дорівнює 1, якщо k-а аудиторія може бути використана для проведення лекцій j-м викладачем в i-й групі, і дорівнює 0 в іншому випадку. Крім того, з кожним j-м викладачем зв'язаний вектор

$$d_j=(d_{j1},d_{j2},...,d_{j,qh}), \tag{4.2}$$

складові якого задовольняють наступній умові:

d_{jt}=1, якщо j-й викладач в t-й час має можливість для викладання лекції, та d_{jt}=0, в іншому випадку. При цьому, якщо кажуть про t-й момент часу, то мають на увазі відносне положення відповідного відрізку часу серед qh навчальних годин тижня. У тижні дійсно міститься qh одногодинних інтервалів, якщо під h мати на увазі кількість одногодинних інтервалів, що містяться в межах одного начального дня.

Розглянемо різні обмеження, враховуючи реальні умови функціонування навчального закладу.

Якщо в t-й час j-й викладач має можливість, він може провести лекцію всього лише одній групі студентів; або ж він може не проводити заняття. Тому

$$\sum_{i=1}^{M}\sum_{k=1}^{S} x_{ijkt} = d_{jt}, \; j = \overline{1,P};\; t = 1,2,\ldots,qh. \qquad (4.3)$$

В t-й інтервал часу i-а група студентів може слухати будь-яку лише одну лекцію, тобто

$$\sum_{j=1}^{P}\sum_{k=1}^{S} x_{ijkt} \le 1, \; i = \overline{1,M};\; t = 1,2,\ldots,qh. \qquad (4.4)$$

В заданий t-ий час k-та аудиторія може бути зайнята лише однією i-ю групою з j-м викладачем, тобто

$$\sum_{i=1}^{M}\sum_{j=1}^{P} x_{ijkt} \le 1, \; k = \overline{1,S};\; t = 1,2,\ldots,qh. \qquad (4.5)$$

Що стосується ймовірного використання k-ї аудиторії j-м викладачем для викладання протягом тижня лекцій для i-ї групи, то тут отримуємо наступні обмеження:

$$\sum_{t=1}^{qh} x_{ijkt} \le qhO_{ijk}, \; i = \overline{1,M};\; j = \overline{1,P};\; k = \overline{1,S}. \qquad (4.6)$$

Нарешті, побудуємо матрицю $C = \{C_{ij}\}$, $i = \overline{1,M}; j = \overline{1,P}$, таким чином, щоб матричний елемент C_{ij} дорівнював кількості лекційних годин j-го викладача в i-й групі. Тоді має виконуватись співвідношення

$$\sum_{k=1}^{S}\sum_{t=1}^{qh} x_{ijkt} = C_{ij}, \; i = \overline{1,M};\; j = \overline{1,P}. \qquad (4.7)$$

Перелік обмежень може змінюватися. Наприклад, в деяких навчальних закладах вимагається, щоб навчальні групи займалися з одним і тим самим викладачем (або однією і тією ж дисципліною) не більше 2 години підряд. Враховуючи цю обставину, можна записати ще одне обмеження.

Найчастіше мають місце наступні вимоги:

1. один і той самий викладач не може мати більше λ годин лекційного навантаження в день;
2. протягом одного дня сумарна тривалість лекцій для будь-якої з груп не повинна перевищувати μ годин;

3. лекції з деяких дисциплін не можуть проводитись у п'ятницю, оскільки, у процесі навчання протягом тижня накопичується втома, що приводить до зменшення рівня засвоєння матеріалу, і т.п.

Планування навчальних процесів, як правило, не підлягає строгій оптимізації. Доводиться задовольнятись можливістю пошуку прийнятного розв'язку (воно може бути і не єдиним), що задовольняє усім обмеженням. Проте, можна було б казати і про оптимізаційні критерії, використання яких дозволило б розширити якість окремих задач. Ось наприклад, можна прагнути:

1. максимально сконцентрувати зайнятість викладачів (відповідно, і студентів) у певні інтервали часу;
2. максимально рознести в часі лекції з найбільш складних дисциплін;
3. максимізувати кількість двохгодинних лекцій (що може бути зручним для викладача), або навпаки, максимізувати кількість лекцій тривалістю 1 година (за бажанням викладачів) і т.п.

Зрозуміло, що оптимізувати відразу по багатьом критеріям неможливо, якщо не брати до уваги ті рідкісні випадки, коли оптимальні рішення за різними критеріями співпадають [89].

Розглянемо чисельний максимально простий приклад, в якому використано чисельні данні, яким навмисно надано «розкиданий характер».

Нехай в нас є п'ять груп студентів E_1, E_2, E_3 ,E_4 та E_5 і троє викладачів P_1, P_2 та P_3. Дані стосовно кількості лекційних годин кожного викладача (з вказівкою на групу студентів, для яких ці лекції викладаються) наведені в табл. 2.1. Припускається, що кількість лекційних годин в день дорівнює 4; при цьому, кількість навчальних днів дорівнює 6. Таким чином, використовуючи введені вище позначення, отримуємо $M=S, P=3, S=3, h=4, q=6$.

Дані, що відносяться до режиму використання аудиторій, наведено в табл. 2.2, кожен рядок таблиці визначає вектор O_{ij}. Можливості викладацького складу, котрі визначаються часом, які кожен викладач може виділити для проведення лекції, наведено в табл. 2.3; тут кожен рядок задає вектор d_j.

Допустимий розв'язок наведений у табл. 2.4. Оскільки в даному прикладі точно задана кількість вільних аудиторій і кількість викладачів невелика, знаходження допустимого розв'язку не є складним. Якщо б кількість викладачів була більшою (без одночасного збільшення у відповідній пропорції числа аудиторій), ситуація з пошуком розв'язку задачі могла б кардинально змінитися [90].

Таблиця 2.1

	P_1	P_2	P_3	
E_1	2	0	2	4
E_2	2	5	0	7
E_3	0	3	8	11
E_4	1	6	0	7
E_5	6	0	0	6
	11	14	10	

Таблиця 2.2

O_{ijk}	S_1	S_2	S_3
E_1P_2	1	1	0
E_1P_3	0	0	0
E_1P_3	0	1	0
E_2P_1	1	1	0
E_2P_2	1	0	0
E_2P_3	0	0	0
E_3P_1	0	0	0
E_3P_2	1	0	1
E_3P_3	0	1	1
E_4P_2	1	1	0
E_4P_3	1	0	1
E_4P_3	0	0	0
E_5P_2	1	1	0
E_5P_3	0	0	0
E_5P_3	0	0	0

Таблиця 2.3

d_j	***Понеділок***				***Вівторок***				***Середа***				***Четвер***				***П'ятниця***				***Субота***			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
P_2	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
P_3	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Цікаво з'ясувати кількість змінних x_{ijkt} в прикладі, що розглядається, що не перетворюються тотожно в нуль. Перш за все відомо, що кількість незалежних змінних x_{ijkt} дорівнює 5*3*3*24 = 1080. Згідно табл. 2.2, значення 696 змінних апріорі є нульовим. Враховуючи у розрахунку дані з табл. 2.3 про можливості викладацького складу, можна побудувати табл. 2.4, з якої видно, що кількість змінних, що не перетворюються тотожно в нуль, дорівнює 244. Визначивши кількість змінних, що не дорівнюють тотожно нулю, остаточно отримуємо спрощений варіант пошуку (мається на увазі, що він здійснюється «вручну») оптимального розв'язку при плануванні навчального процесу.

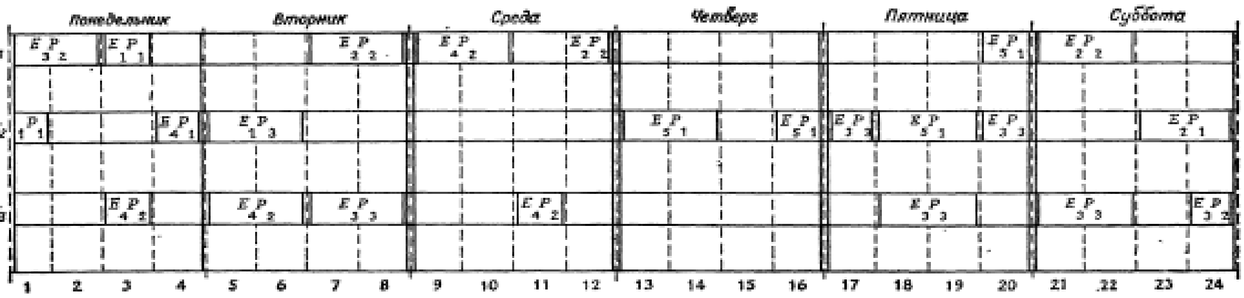
Проте, в реальних умовах задачі такого типу, очевидно, мають набагато більшу розмірність (більшу кількість викладачів, аудиторій і т.п). В таких випадках задачу слід розбити на підзадачі та намагатись застосувати алгоритми евристичного характеру, які в умовах, коли не вдається отримати рішення в аналітичному вигляді, можуть виявитись досить прийнятними.

Один з допустимих розв'язків наведено у табл. 2.5. Цей розв'язок можна було б розглядати з позиції певного критерію; тоді можна ставити питання щодо його покращення, як по відношенню до обраного критерію, так і по відношенню до інших критеріїв, якщо, зрозуміло, це становить практичний інтерес.

Таблиця 2.4. Число змінних, що не перетворюються тотожно в нуль

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
1,1	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	32
1,2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1,3	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	12
2,1	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	32
2,2	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	16
2,3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3,1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3,2	2	2	2	2	2	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	32
3,3	0	0	0	0	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	24
4,1	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	32
4,2	2	2	2	2	2	2	2	2	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	32
4,3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5,1	2	2	2	2	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	32
5,2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5,3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Таблиця 2.5.



4.2. Моделювання динаміки розподілу часового ресурсу при проведенні процесу тестування

Розглянемо один з прикладів дослідження та моделювання процесу формування розкладу з урахуванням динаміки величини ресурсів, адаптуючи задачу оптимізації (2.5)-(2.6) (або (2.7)-(2.8)) для знаходження раціонального розв'язку проблеми оцінювання результатів тестування.

При проведенні професійного оцінювання на основі тестів вважається, що кожен, хто бере участь у тестуванні, має вирішити n завдань. Завдання характеризуються складністю s, величини якої без обмеження загальності належать інтервалу [0,1]. Різні рівні складності завдань потребують відповідного часу на розв'язування $T_C(s)$, $s \in [0,1]$.

Кожна особа, що проходить тестування, має конкретний рівень підготовки α, що також задається значеннями з інтервалу [0,1]. Цей рівень отримується в результаті проведення попередніх тестів і, отже, впливає на оцінку часу виконання кожного завдання $T_L(\alpha)$, $\alpha \in [0,1]$.

При формуванні тестових завдань час виконання кожного з них розраховується, виходячи з середнього рівня підготовки особи ($\alpha = 0{,}5$), яка буде проходити тестування. Таким чином, апріорі час виконання особою всього тесту за умов її рівня підготовки вище середнього має бути кращим (меншим) за час, необхідний для виконання тесту особами з підготовкою, нижче за середню.

На виконання всіх завдань тесту виділяється максимальний час $\bar{T}$. Ставиться задача визначення послідовності розв'язування тестових завдань, для якої загальний час виконання тесту буде мінімальним.

Проведемо моделювання процесу виконання сукупності завдань у часі з урахуванням окремих передумов і обмежень на характер динаміки процесу. Оцінювання якості послідовності розв'язування завдань будемо проводити на основі обсягу часу, що залишається у особи до завершення часу тестування $\bar{T}$.

Зрозуміло, що цього можна досягти, якщо мінімізувати тривалість виконання завдання кожного завдання. Нехай функція поточної складності завдання, що розв'язується, вимірюється часом, який потрібен для завершення завдання, і є монотонно спадною функцією двох параметрів $T(t,\alpha)$, $\alpha \in [0,1]$. За відсутності підготовки час на розв'язання кожного завдання ніколи не завершується і, як наслідок, час до кінця тесту не зменшується. З іншого боку, за найвищого рівня підготовки особи кожне завдання вирішується найбільш швидко. Ці припущення дозволяють визначити схематичний характер поведінки функції $T(t,\alpha)$ у вигляді поверхні, наведеної на рис.4.1 (тут 0,…,3,…,6,… – моменти часу t, [0,1] - рівні підготовки, [0,10] - значення функції):

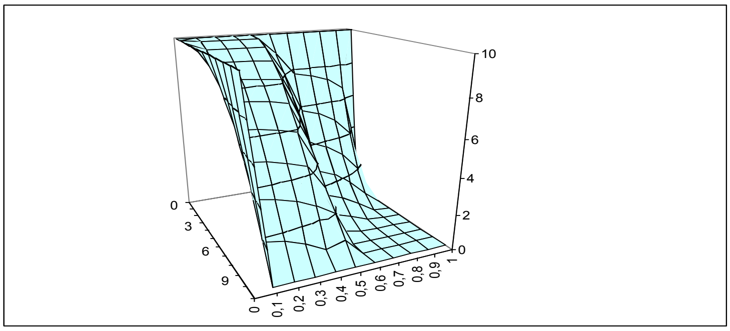


Рис. 1. Схематичний вигляд функції $T(t,\alpha)$.

За таких умов на площині (t,α) отримуємо криву, яка визначає залежність часу виконання кожного завдання $t = t(\alpha)$ від рівня підготовки $\alpha \in [0,1]$. Чим вище рівень підготовки, тим швидше завершується виконання завдання і, відповідно, більше залишається часу на виконання інших завдань тесту. Зрозуміло, що ця крива співпадає з графіком функції $T_L(\alpha)$, $\alpha \in [0,1]$.

Як вже було сказано вище, номінальний розрахунок часу виконання здійснюється для $\alpha = 0{,}5$. При цьому, максимальний час визначається величиною $T_L(0)$, а мінімальний – величиною $T_L(1)$.

Припускаючи монотонний характер зменшення величини необхідного часу для виконання завдання, формалізуємо цю поведінку за допомогою деякої неперервної функції з потрібними властивостями. Візьмемо функцію $t(\alpha)$, наприклад, такого вигляду

$$t(\alpha) = \left(T_L(0) - T_L(1)\right)\sqrt{1-\alpha^2} + T_L(1), \qquad (4.8)$$

де величину $T_L(0)$ можна покласти рівною $\overline{T}$ для будь-якого завдання тесту. Таке припущення дозволяє отримати наближені значення функції $T_L(\alpha) = t(\alpha)$, $\alpha \in [0,1]$, не проводячи відповідних вимірювань.

Аналогічно змоделюємо динаміку процесу зміни часу, що залишається до повного розв'язання задачі, за допомогою монотонно спадної функції вигляду

$$T(t,\alpha) = \begin{cases} T_L(0.5)\sqrt{1 - t^2 / T_L{}^2(\alpha)}, & \alpha \le 0.5; \\ T_L(0.5)\left(-T_L(\alpha)\right)^2 / T_L{}^2(\alpha), & \alpha > 0.5; \end{cases}, \ t \in [0, T_L(\alpha)], \qquad (4.9)$$

де $T(0,\alpha) = T_L(0.5)$, $T(T_L(\alpha),\alpha) = 0$, $\alpha \in [0,1]$.

Чисельне моделювання процесу тестування проведено з урахуванням різної кількості тестових завдань. Для заданого часу розв'язування тесту обчислювалась величина часу, що залишається наприкінці вирішення усіх тестових завдань. Отримано результати у випадках незростання і неспадання складності $S_i \in [0,1]$, $i = \overline{1,n}$, вхідних завдань (див. Додаток Б). Результати тестування підтверджують висновок, що у системі з n завдань і одного обчислювального пристрою (особи, що проходить тестування) максимальна тривалість виконання дорівнюють сумі тривалостей розв'язування n завдань і однакова для усіх $n!$ можливих впорядкувань.

Враховуючи, що тестування проходить людина, можна передбачити, що на швидкість розв'язування кожного наступного завдання буде впливати ефект наявності завершених на попередніх етапах завдань. Змоделювати цей процес можна на основі, наприклад, опису процесу відліку часу до завершення тесту $T(t)$ у вигляді

$$dT/dt = -1 + \sum_{i=1}^{n} B_i w_i(t),\ T(0) = \overline{T}, \tag{4.10}$$

де $w_i(t) \in \{0,1\}, i = \overline{1,n},$ - булеві змінні, $w_i(t) = 0$ при $t \in [0, t_i)$ і $w_i(t) = 1$ при $t \in [t_i, \overline{T}]$, t_i - момент початку виконання i - го завдання, $i = \overline{1,n}$.

За допомогою такого принципу відлік часу сповільнюється за наявності завершених завдань і залишається незмінним у випадку нерозв'язаних завдань для будь-якого порядку їх розв'язування. Сповільнення часу означає більш повільний плин часового обліку при виконанні завдань по відношенню до початкового обліку.

Попередні числові розрахунки було доповнено результатами, що враховують модель динаміки часового обліку (див. Додаток Б). У результаті проведених обчислень отримано підтвердження впливу зміни величини $T(t)$ на розв'язок оптимізаційної задачі, при чому для рівня підготовки $\alpha \in [0.5,1]$ час, що залишився наприкінці тестування, для неспадної за складністю послідовності завдань виявився меншим за залишковий час для випадку незростаючої послідовності.

Таким чином, формалізація процесу впорядкування сукупності робіт на прикладі проведення тестових випробувань дозволяє отримати гібридну модель з оптимізаційної задачі теорії розкладів та диференційного рівняння, що описує динаміку зміни одного з параметрів задачі оптимізації. Позитивний аналіз результатів чисельного моделювання дає підґрунтя для створення програмного забезпечення для визначення оптимального порядку виконання тестових задач.

4.3. Комп'ютерна система для проведення дистанційного тестування

Системи створення та проведення online-тестувань з кожним днем набирають все більшої популярності. Такі системи широко застосовуються як навчальними закладами, так і великими корпораціями, що проводять підготовку та перепідготовку своїх спеціалістів або підвищення кваліфікації для розв'язування певних задач.

Основні задачі, які має вирішувати сучасна система online-тестувань:

– визначення рівню підготовки учасників тестувань;

– підготовка, моделювання поведінки та тренування користувачів системи до виконання завдань (наприклад до складання іспитів або перевірки реагування користувача на певні події).

Основні можливості, якими має володіти система тестування:

– незалежність системи від програмного забезпечення, встановленого на комп'ютері користувача;

– підтримка реєстрації користувачів та розподілу прав доступу за ролями;

– формування навчальних груп за рівнем підготовки;

– широкий спектр типів запитань і завдань;

– зручний інтерфейс створення тестів та наповнення їх запитаннями;

– можливість зміни параметрів тесту;

– наявність підсистеми обробки статистичної інформації за результатами тесту;

– наявність навчального журналу та обліку відвідування студентів;

– забезпечення інформування користувачів про найближчі події;
– завантаження та збереження у системі навчальних матеріалів;
– забезпечення можливості спілкування між користувачами в системі.

Комп’ютерні системи для організації дистанційних тестувань за наявними можливостями поділяють на:
– портали тестування з готовими тестами (без можливості створення);
– конструктори online-тестів;
– системи перевірки знань, які є складовими систем дистанційного навчання.

Система проведення дистанційних тестувань та організації навчальних процесів в режимі “онлайн” – EdUni за своїми можливостями відноситься до розподілених інформаційних систем корпоративного типу. Основною відмінністю та перевагою EdUni перед існуючими аналогами є те, що система від самого початку орієнтована на спрощення процесу проведення навчання в студентських групах завдяки наявності усіх необхідних для цього функціональних можливостей (завантаження матеріалів, розширене тестування, облік відвідування та оцінок тощо) та без необхідності додаткових налаштувань і додаткового програмування системи її користувачами.

Для розробки системи дистанційних тестувань EdUni використовується мова програмування PHP (5.4) із застосуванням PHP-фреймворку Kohana 3. Опис системи наведено у додатку В.

4.4. Висновки по четвертому розділу

У четвертому розділі розглянуто приклади практичного розв’язування задач оптимального розподілу часового ресурсу. Запропоновано програмну реалізацію підходу для складання розкладу занять академічних груп і викладачів в межах аудиторного фонду навчального закладу протягом типового тижня. Проведено комп’ютерне моделювання динаміки розподілу часового ресурсу при проведенні процесу тестування.

Коротко описано систему для проведення дистанційних тестувань, її можливостей та вимог для використання.

ВИСНОВКИ

Дисертаційна робота присвячена розробці методів та алгоритмів розв'язування задач розподілу часового ресурсу при складанні розкладів виконання сукупності робіт з нечітко заданими ресурсними обмеженнями та термінами виконання, формалізації підходів для моделювання поведінки динамічних процесів з урахуванням зміни темпів плину часу (на прикладі процесу тестування).

Проведено огляд класів задач лінійного програмування як основних задач дослідження операцій. Детально викладено зміст задач про рюкзак, про складання розкладу та про вибір заявок. Наведено основні положення про нечіткі величини та способи їх формалізації. Викладено принцип формування нечіткого розв'язку Белмана-Заде нечітких задач лінійного програмування, розглянуто різні варіанти нечітких моделей ЗЛП (з нечіткими технологіч-ними коефіцієнтами та ресурсними обмеженнями). Сформульовано нечіткі оптимізаційні задачі та проаналізовано способи для знаходження нечітких розв'язків. Розглянуто властивості та проаналізовано методи розв'язування оптимізаційних задач планування, розміщення та розподілу ресурсів.

В роботі розглянуто та визначено методику розв'язування чітких і нечітких задач розподілу часового ресурсу як задач складання розкладу: наведено класифікацію задач теорії розкладів залежно від різних характеристик, викладено математичні постановки оптимізаційних задач розподілу часового ресурсу, сформульовано загальної задачі складання розкладу в термінах лінійного цілочисельного програмування, досліджено задачу складання розкладу виконання заданої кількості робіт для однієї машини, наведено постановку і спосіб розв'язання задачі оптимальної рекомбінації виконання робіт.

Викладено спосіб використання методу динамічного програмування для розв'язування задач теорії розкладів у випадках з регулярним та нерегулярним критеріями. Запропоновано метод найменших відхилень розв'язання задачі розподілу часового ресурсу, схему реалізації жадібного підходу до розв’язування задач розподілу ресурсів, підходи до застосування економічних стратегій для розв’язання задач розподілу ресурсів.

Наведено детальний розгляд моделей і методів, які застосовуються при розв’язанні нечітких задач оптимального розподілу часового ресурсу. Сформульовано нечітку задачу про рюкзак як засіб розподілу нечітко визначеного часового ресурсу. Описано задачу складання розкладу з нечітко заданими інтервалами налаштувань.

Особливу увагу приділено задачі теорії розкладів з нечітким вимірюванням часу. Визначено поняття структурованих нечітких чисел, які використовуються для опису вимірювання відліку часу. Розглянуто нечітку задачу про рюкзак з нечітко заданими термінами виконання.

На основі сильної та слабкої зв’язності обмежень лінійних задач оптимізації розроблено метод перетворення області допустимих розв’язків з урахуванням близькості обмежень. Проілюстровано вплив запропонованого методу на результати розв’язання нечітких ЗЛП. Розроблено схему розв’язання нечітких задач лінійного програмування за наявності систем альтернативних обмежень.

Побудовано гібридну модель динаміки процесу обробки сукупності завдань, отримано твердження про фундаментальну матрицю її розв’язків.

Отримані результати можуть бути використані при розв’язанні актуальних прикладних задач в умовах невизначеності. Розглянуто приклади практичного розв’язування задач оптимального розподілу часового ресурсу. Запропоновано програмну реалізацію підходу для складання розкладу занять академічних груп і викладачів в межах аудиторного фонду навчального закладу протягом типового тижня. Проведено комп’ютерне моделювання динаміки розподілу часового ресурсу при проведенні процесу тестування.

Коротко описано систему для проведення дистанційних тестувань, її можливостей та вимог для використання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Зайченко Ю.П. Дослідження операцій. – К.: «Слово», 2006. – 815 с.

2. Elamvazuthi I., Ganesan T., Vasant P., Webb J.F. Application of a fuzzy programming technique to production planning in the textile industry// International Journal of Computer Science and Information Security. 2009. V.6. №3. P.238-243.

3. Дюбуа Д., Прад А. Теория возможностей. Приложения к представлению знаний в информатике. М.: Радио и связь, 1988. – 288с.

4. Zimmermann, H.J. Fuzzy Set Theory and its application. Boston: Kluwer. 1992. – 399 p.

5. Zimmermman, H. J. Application of Fuzzy Set Theory To Mathematical Programming// Information Sciences. 1985. 36. P. 25-58.

6. Delgado, M., Verdegay, J.L., Vila, M. A. A General Model For Fuzzy Linear Programming// Fuzzy Sets and Systems. 1989. 29. P. 21-29.

7. Zadeh, L. A. Fuzzy sets// Information and Control. – 1965. – V.8. – P.338-353.

8. Orlovsky, S. A. On Formalization Of A General Fuzzy Mathematical Programming Problem//Fuzzy Sets and Systems. 1980. 3. P. 311-321.

9. Zadeh L.A. Fuzzy sets as a basis for a theory of possibility//Fuzzy Sets and Systems. 1978. 1. Pp.3–28.

10. Yager R.R. A characterization of the extension principle//Fuzzy Sets and Systems. 1986. 18. Pp.205–217.

11. Вятченин Д.А. Нечеткие методы автоматической классификации. Минск: Технопринт, 2004. - 219 с.

12. Kickert, W.J. Fuzzy Theories on Decision-Making: Frontiers in Systems Research. Leiden, The Netherlands: Martinus Nijhoff. 1978.

13. Zimmermann, H.J. Fuzzy Sets, Decision Making and Experts Systems. Boston: Kluwer. 1987.

14. Zadeh L.A. Fuzzy Logic Neural Networks and Soft Computing// Communications of the ACM, 1994, Vol. 37 No. 3. PP. 77—84.

15. Afraimovich L.G., Prilutskii M.Kh. Multiindex Resource Distributions for Hierarchical Systems// Automation and remote control. 2006. V. 67 No. 6, PP. 1007–1016.
16. Борисов А.Н., Крумберг О.А., Федоров И.П. Принятия решений на основе нечетких моделей – Рига: Зинатне, 1990. – 184с.
17. Sakawa M., Yano H. Interactive decision making for multi-objective linear fractional programming problems with fuzzy parameters // Cybernetics Systems. – 1985. – No.16. – P.377-394.
18. Zimmermann H.J. Fuzzy programming and linear programming with several objective functions // Fuzzy Sets and System. – 1978. – No.1. – P.45-55.
19. Tamiz M. Multi-objective programming and goal programming: theories and applications. Germany: Springer-Verlag. 1996.
20. Ross T. J. Fuzzy Logic with Engineering Applications. New York: McGraw-Hill. 1995.
21. Klir G. J., Yuan B. Fuzzy Sets and Fuzzy Logic: Theory and Applications. Upper Saddle River, NJ: Prentice Hall PTR. 1995.
22. Yager R.R., Ovchinikov S., Tong R.M., Nguyen H.T. Fuzzy Sets and Applications. Selected Papers by L.A. Zadeh. New York:John Wiley. 1987.
23. Bellman R.E., Zadeh L.A. Decision making in a fuzzy environment //Management Science. – 1970. – No.17. – P.141-164.
24. Воронин А.А., Мишин С.П. Оптимальные иерархические структуры. М.: ИПУ, 2003.
25. Elsasser R., Monien B. and Preis R. Diffusion Schemes for Load Balancing on Heterogeneous Networks// Theory Comput. Syst., 2002, vol. 35, no. 3, pp. 305–320.
26. Gairing M., Lucking T., Mavronicolas M. and Monien B. Computing Nash Equilibria for Scheduling on Restricted Parallel Links// Proc. 36th Annual ACM Sympos. Theory Comput., 2004, pp. 613–622.
27. Dunne P.E. Extremal Behavior in Multiagent Contract Negotiation// Jour. Artificial Intelligence Res., 2005, vol. 23, no. 1, pp. 41–78.

28. Prilutskii M.Kh. Distribution of a Homogeneous Resource in Tree-Structured Hierarchical Systems// Proc. III Int. Conf. Identification of Systems and Control Problems (SICPRO’2000), Москва: ИПУ. - 2000, pp. 2038–2049.
29. Прилуцкий М.Х. Потоковые алгоритмы распределения ресурсов в иерархических системах / М.Х.Прилуцкий, А.Г. Картомин // Исследовано в России. – 2003. – 39. – С. 444-452. – Режим доступу до журн. http://zhurnal.ape.relarn.ru/ articles/2003/039.pdf
30. Prilutskii M.Kh. Many-Criteria Distribution of a Homogeneous Resource in Hierarchical Systems// Автоматика и телемеханика, 1996, №2, pp. 24–29.
31. Graham, R.L. Optimization and approximation in deterministic sequencing and scheduling; a survey/ R. L. Graham, E. L. Lawler, J. K. Lenstra // Annals of Discrete Math. - 1979. - V. 5. - P. 287-326.
32. Fischer, H. Probabilistic learning combinations of local job scheduling rules / H.Fischer, G.L.Thompson // Industrial Scheduling. Englewood Cliffs, New Jersey : Prentice Hall, 1963. - P.225-251.
33. Giffler, B. Algorithms for solving production scheduling problems/ B. Giffler, G. L. Thompson // Operation Research. - 1960. - N 8. - P. 487-503.
34. Sevastjanov, S.V. On some geometric methods in scheduling theory: a survey / S.V. Sevastjanov // Discrete Applied Mathematics. - 1994. – N. 55. - P. 59-82.
35. Юдин Д.Б. Экстремальные модели в экономике/ Д.Б.Юдин, А.Д.Юдин. – М.: Экономика, 1979. – 288с.
36. Волконский В.А. Принципи оптимального планирования/ В.А.Волконский. – М.: Экономика, 1973. – 216с.
37. Жданова Е.Г. Теория расписаний. – Москва, 2000. – 136 с.
38. Floudas C.A., Kallrath J., Pitz H.J., Shaik M.A. Production scheduling of a large-scale industrial continuous plant: short-term and medium-term scheduling // Comp. Chem. Eng. – 2009. – V. 33. – P. 670-686.
39. Bianco L., Blazewicz J., Dell'Ohno P., Drozdowski M. Scheduling preemptive multiprocessor tasks on dedicated processors // Performance Evaluation. – 1994. – V. 20. – P. 361-371.

40. Pochet Y., Wolsey L.A. Production planning by mixed integer programming. – Berlin, Heidelberg, New York: Springer-Verl., 2006. – 477 p.

41. Березнев В.А. О полиномиальной сложности одной модификации симплекс-метода // Журнал вычислительной математики и математической физики. 2004. Т. 44. № 7. С. 1244–1260.

42. Булавский В.А., Звягина Р.А., Яковлева М.А. Численные методы линейного программирования. Специальные задачи. – М.: Наука. 1977.

43. Нестеров Ю.Е. Метод линейного программирования с трудоемкостью O(n3L) операций // Экономика и математические методы. 1988. Т. 24. № 16. С. 174–176.

44. Liu S.T., Kao Chiang. Solving transportation problems based on extension principle// European Journal of Operation Research, 2004. 153. Pp. 661-674.

45. Lai Y.J., Hwang C.L. Fuzzy Mathematical Programming. Lecture notes in Economics and Mathematical systems // Springer-Verlag, 1992.

46. Isermann H. The numeration of all efficient solutions for a linear multiobjective transportation problems// Naval Research Logistic Quarterly. – 1979. – No.26. – P. 123–139.

47. Ringuest L., Rinks D.B. Interactive solutions for the linear multiobjective transportation problem// European Journal of Operational Research. – 1987. – No.32. – P. 96–106.

48. Chanas S., Kuchta D. Fuzzy programming in multi-objective linear programming-parametric approach // Fuzzy Set and System. – 1989. – No.29. – P.303-313.

49. Tien Fuling. Applying interactive fuzzy multi-objective Linear programming to transportation planning decisions // Journal of information and optimization sciences. – 2006. – V.27. – №1. – P.107-126.

50. Dubois D. Linear programming with fuzzy data / D. Dubois // Analysis of Fuzzy Information / J. C. Bezdek (ed.). Boca Raton : CRC Press. – 1987. - Vol. 3: Applications in Engineering and Science. – P. 241-263.

51. Tanaka H., Asai K. Fuzzy linear programming problems with fuzzy numbers // Fuzzy Sets and Systems. – 1984. – No.13. – P.1-10.
52. Bablu Jana, Tapan Kumar Roy. Multi-Objective Fuzzy Linear Programming and Its Application in Transportation Model// Tamsui Oxford Journal of Mathematical Sciences. – 2005. – V.21. – No.2. – P.243-268.
53. Gasimov R.N., Yenilmez K. Soving fuzzy linear programming with linear membership functions // Turk. J.Math. – 2002. – No.26. – P.375-396.
54. Ivokhin E.V., Almodars Barraq Subhi Kaml. Single-Objective Linear Programming Problems With Fuzzy Coefficients and Resources// Computational and Applied Math. – 2013. – №1. – P. 117-125.
55. Схрейвер А. Теория линейного и целочисленного программирования: в 2 т. – М.: Мир. 1991.
56. Eisenbrand F. Fast integer programming in fixed dimension // Algorithms - ESA 2003. Lecture Notes in Computer Science. 2003. V. 2832. P. 196–207.
57. Lenstra H.W. Jr. Integer programming with a fixed number of variables // Mathematics of Operations Research. 1983. V. 8. N. 4. P. 538–548.
58. Прилуцкий М.Х., Афраймович Л.Г. Распределение ресурсов в иерархических системах транспортного типа. Учебно-методический материал по программе повышения квалификации «Новые подходы в исследованиях и разработках информационно-телекоммуникационных систем и технологий». Нижний Новгород, 2007. 80с.
59. Orlin J.B. A Faster strongly polynomial minimum cost flow algorithm // Operations research. 1993. V. 41. N 2. P. 338–350.
60. Burkard R.E., Dell'Amico M., Martello S. Assignment Problems. – Philadelphia: SIAM. 2009.
61. Pentico D.W. Assignment problems: A golden anniversary survey // European Journal of Operational Research. 2007. V. 176. P. 774–793.
62. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. – М.: Мир. 1982.

63. Лазарев А.А., Гафарова Е.Р. Теория расписаний: задачи и алгоритмы. – М.: МГУ им.М.В.Ломоносова, 2011. – 222 с.

64. Зайченко Ю.П. Исследование операций: Учебн. для вузов. – 6-е изд. К.: Изд.дом «Слово», 2003. – 688с.

65. Кофман А., Анри-Лабодер А. Методы и модели исследования операцій. – М.: Мир, 1977. – 432 с.

66. Кофман А. Введение в теорию нечетких множеств / А.Кофман. — М.: Радио и связь, 1982. - 432 с.

67. Заде Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л.А.Заде – М.: Мир, 1976. – 175с.

68. Nahmias S. Fuzzy variables / S.Nahmias // Fuzzy Sets and Systems. – 1978. – №. 1. – P.97-110.

69. Лю Б. Теория и практика неопределенного программирования / Б. Лю. – М.: БИНОМ. Лаборатория знаний, 2005. – 416 с.

70. Dehghan M., Hashemi B. Iterative solution of fuzzy linear systems// Appl. Math. Comput., 2006. – 175. - P.645-674.

71. Zimmermann H.J., Zysno P. Quantifying vagueness in decision models / H.J. Zimmermann, P.Zysno.// European Journal of Operational Reseach. 1985. – №22. – P.148–158.

72. Юдин Д.Б., Гольштейн Е.Г. Линейное программирование. – М.: Наука, 1969. – 424с.

73. Bellman R.E., Zadeh L.A. Decision making in a fuzzy environment //Management Science. – 1970. – No.17. – P.141-164.

74. Берзин Е.А. Оптимальное распределение ресурсов и элементы синтеза систем. – М.: «Советское радио», 1974. – 304 с.

75. Еремеев А.В. О сложности оптимальной рекомбинации для одной задачи составления расписаний с переналадками / А.В.Еремеев, Ю.В.Коваленко // Дискретный анализ и исслед. операций. – 2012. – Т. 19, № 3. – С. 13 - 26.

76. Вентцель Е. С. Исследование операций. – М.: «Советское радио», 1972. – 552 с.

77. Moulin H. Axiomatic cost and surplus sharing // Handbook of Social Choice and Welfare, ed.1, v.1 / Herve Moulin. – Amsterdam: Elsevier, 2002. – С. 289–357.

78. Maimonides M. Book of Judgements, translated by Rabbi Elihahu Touger / Moses Maimonides. – New York / Yerusalem: Moznaim Publishing Corporation, 2000.

79. Pinedo M. L. Scheduling: theory, algorithms, and systems. – Springer Science & Business Media, 2012.

80. Орловский С.А. Проблемы принятия решения при нечеткой исходной информации. – М.: Наука, 1981. – 206с.

81. Алефельд Г. Введение в интервальные вычисления./ Алефельд Г., Херцбергер Ю. – М.:Мир, 1987. – 360 с.

82. Добронец Б. С. Интервальная математика. Учеб. пособие. – Красноярск: СФУ, 2007. – 287 с.

83. Голицын Г.А., Петров В.М. Информация - поведение - творчество. - М: Наука, 1991.

84. Golitsyn G. A., Petrov V. M. Information and Creation. - Basel:Birkhauser Verlag,

85. Михалевич В.С. Вычислительные методы исследования и проектиро–вания сложных систем / В.С.Михалевич, В.Л.Волкович. – М.: Наука, 1982. – 286 с.

86. Jang J.S. Neuro–Fuzzy and soft computing/ J.S. Jang, C.T. Sun, E. Mizutani – N.Y.: Prentice Hall, 1997. – 176 p.

87. Гантмахер Ф.Р. Теория матриц /Ф.Р.Гантмахер. – М.: Наука, 1966.– 576с.

88. Івохін Є.В. Досвід впровадження інформаційних технологій в процеси автоматизації управління навчально–методичною роботою у ВНЗ / Є.В. Івохін // Праці 5 між.конф. «Сучасні технології в освіті». – Київ, 2010. – С.73 – 77.

89. Снитюк В.Е., Юрченко К.Н. Интелектуальное управление оцениванием знаний. – Черкаси: ТОВ «Маклаут», 2013. – 262 с.

90. Аванесов В.С. Основы научной организации педагогического контроля в высшей школе, Учебное пособие, М.: Исследовательский центр, 1989, - 167 С.

Додаток А. Результати розрахунків розподілів часового ресурсу у випадку нечітко заданих термінів виконання.

Для аналізу впливу динаміки відліку часу на хід виконання сукупності завдань при нечітко заданих термінах обробки проведено чисельні експерименти. В табл. А.1 наведено результати розв'язання задачі (3.14)-(3.15) для таких параметрів: $M = 10$, $\overline{T} = 100$, $T_i, i = \overline{0,9}$: 20.0, 20.0, 12.885, 12.741875, 9.53875, 9.5375, 3.33875, 2.635, 1.134375, 0.64125. Формалізація нечіткого плину часу проводилась за спрощеною схемою. Виконання або невиконання завдання визначалось випадково. Виконання чергового завдання «сповільнювало» темп зменшення часу, а невиконання - навпаки, «прискорювало». Зміна швидкості в процесі обчислень фіксувалось у вигляді відповідних модифікацій виділеного на виконання всіх завдань часу $\overline{T}$: величина збільшувалась пропорційно часу виконаного поточного завдання (ефект сповільнення темпу плину часу) і зменшувалась – при невиконанні поточного завдання (ефект прискорення).

Таблиця А.1. Результати розв'язання задачі оптимізації (3.14)-(3.15) з урахуванням зміни темпу плину часу.

Заданий час на виконання $\overline{T}$	Час на виконання без врахування корекції темпу	Час на рекомбінацію	Час на виконання з урахуванням корекції темпу	Модифікований час виконання $\overline{T}$	Послідовність розв'язання завдань (нерозв'язані завдання)
60	50.173122	11	53.173122	60.362801	4,0,9,2,8,6,7 (1,3,5)
70	46.825626	23	60.825623	69.501183	0,5,7,9,4,8,6 (1,2,3)
80	69.113754	12	77.113754	80.177444	9,1,7,2,4,3,8,5 (0,6)
90	82.915001	8	87.915001	90.501381	2,9,4,8,1,6,3,0, 7 (5)
100	79.567497	20	91.567497	99.338676	5,7,8,3,9,0,6,1, 4 (2)

На рисунку А.1 наведено графік поетапної (з урахуванням виконання / невиконання завдань) зміни часу виконання всіх завдань для заданого значення

$\bar{T}$ =70, а на рисунку А.2 - графік зміни часу виконання всіх завдань для заданого значення $\bar{T}$ =90.

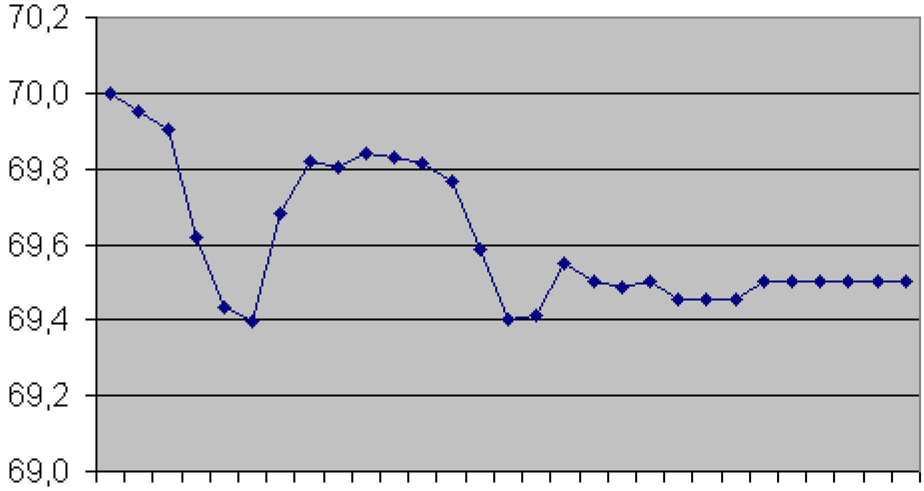


Рис.А.1. Графік поетапної зміни часу при виконанні всіх завдань ($\bar{T}$ =70).

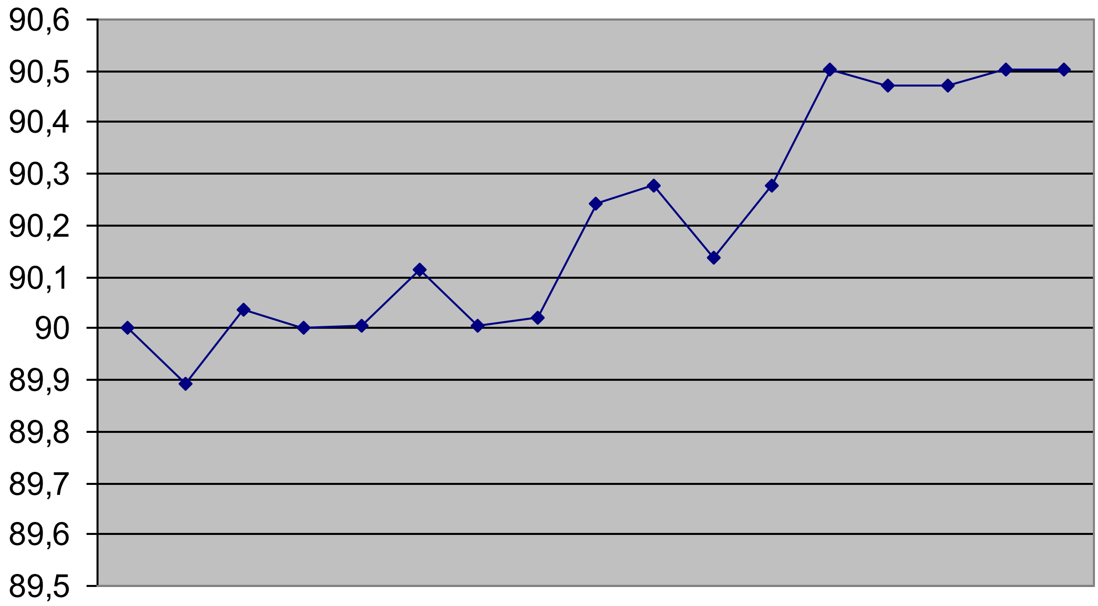
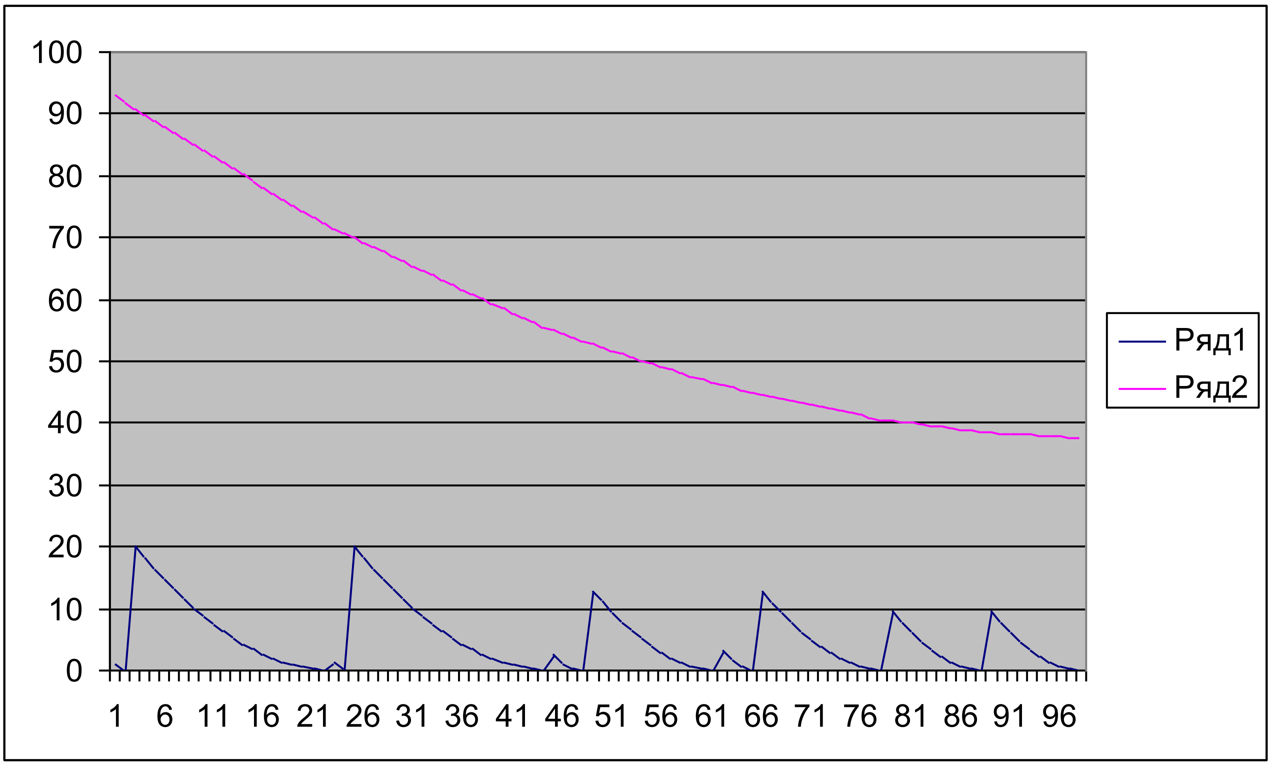


Рис.А.2. Графік поетапної зміни часу при виконанні всіх завдань ($\bar{T}$ =90).

Додаток Б. Результати моделювання динаміки розподілу часового ресурсу.

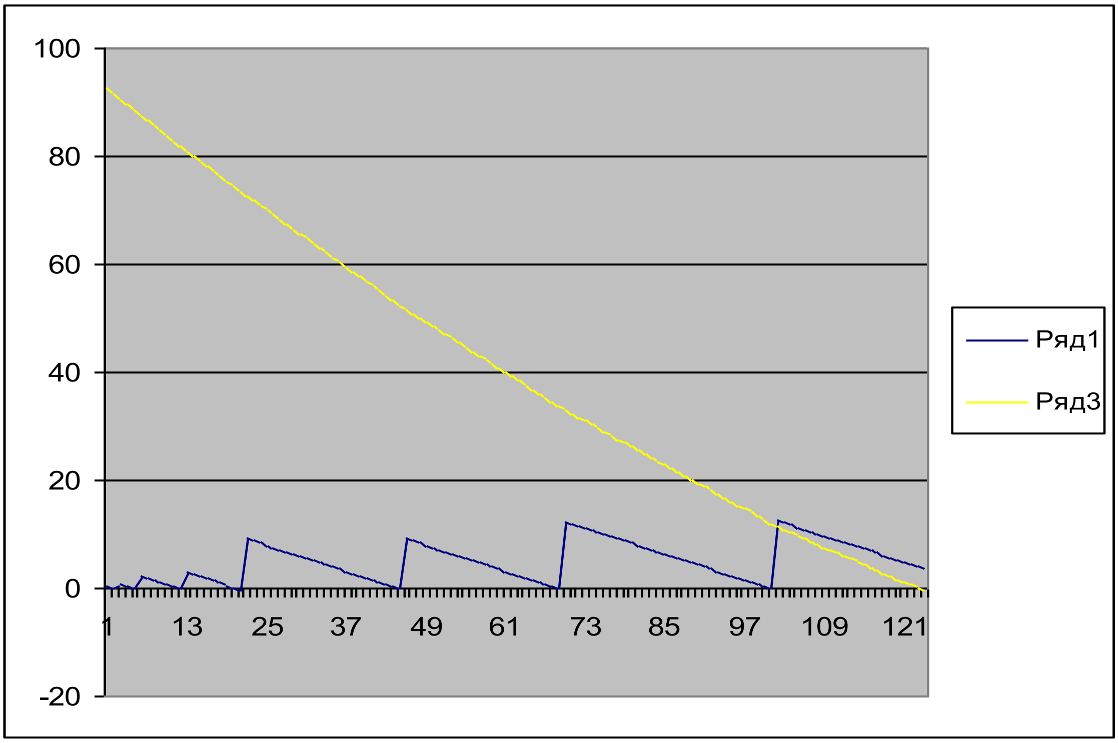
		Total	steps=		98
		Last	deltaT(98)=	-0,1	
Test	Executed	All	tasks executed	Sum of Balls = 50	
Tasks level direction	increase	alfa=0,8	Time leave=37,58		

Level of task	Ball's cost
1,064125	1
1,134375	1
2,635	2
3,33875	2
9,5375	5
9,53875	5
12,74188	7
12,885	7
20	10
20	10

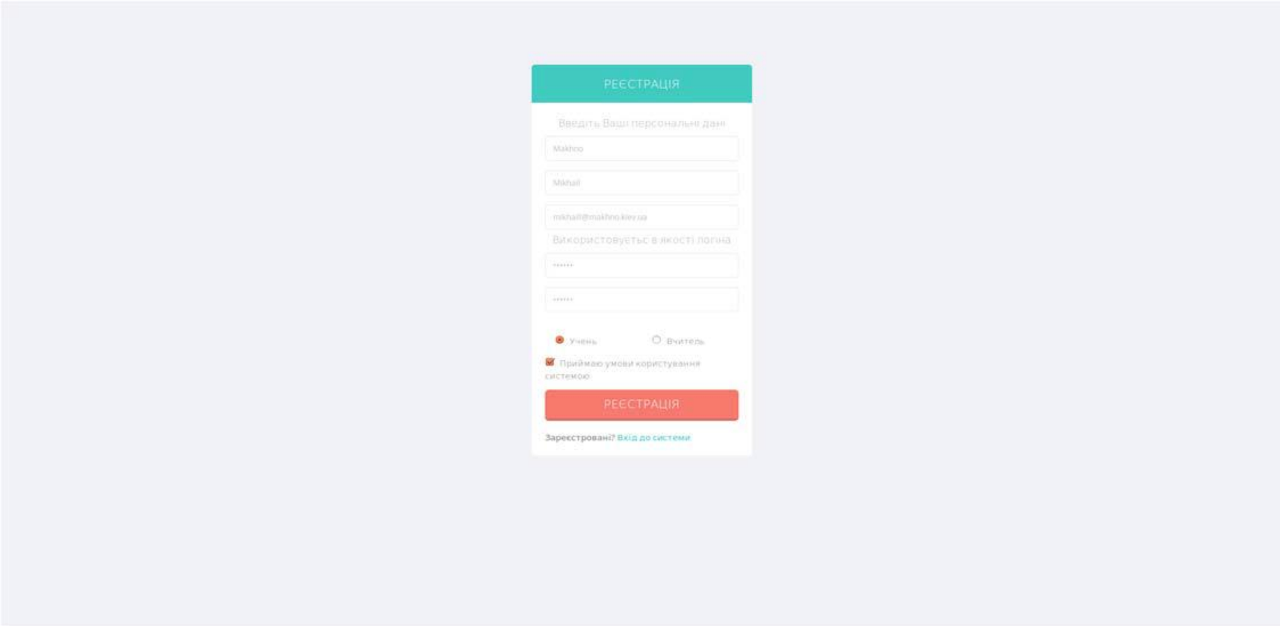


		Total	steps=	121
		Last	deltaT(121)=	-0,54
Test	Didn't Execute	7	tasks executed	Sum of Balls = 23
Tasks level direction	increase	alfa=0,4	Time leave=0,03	

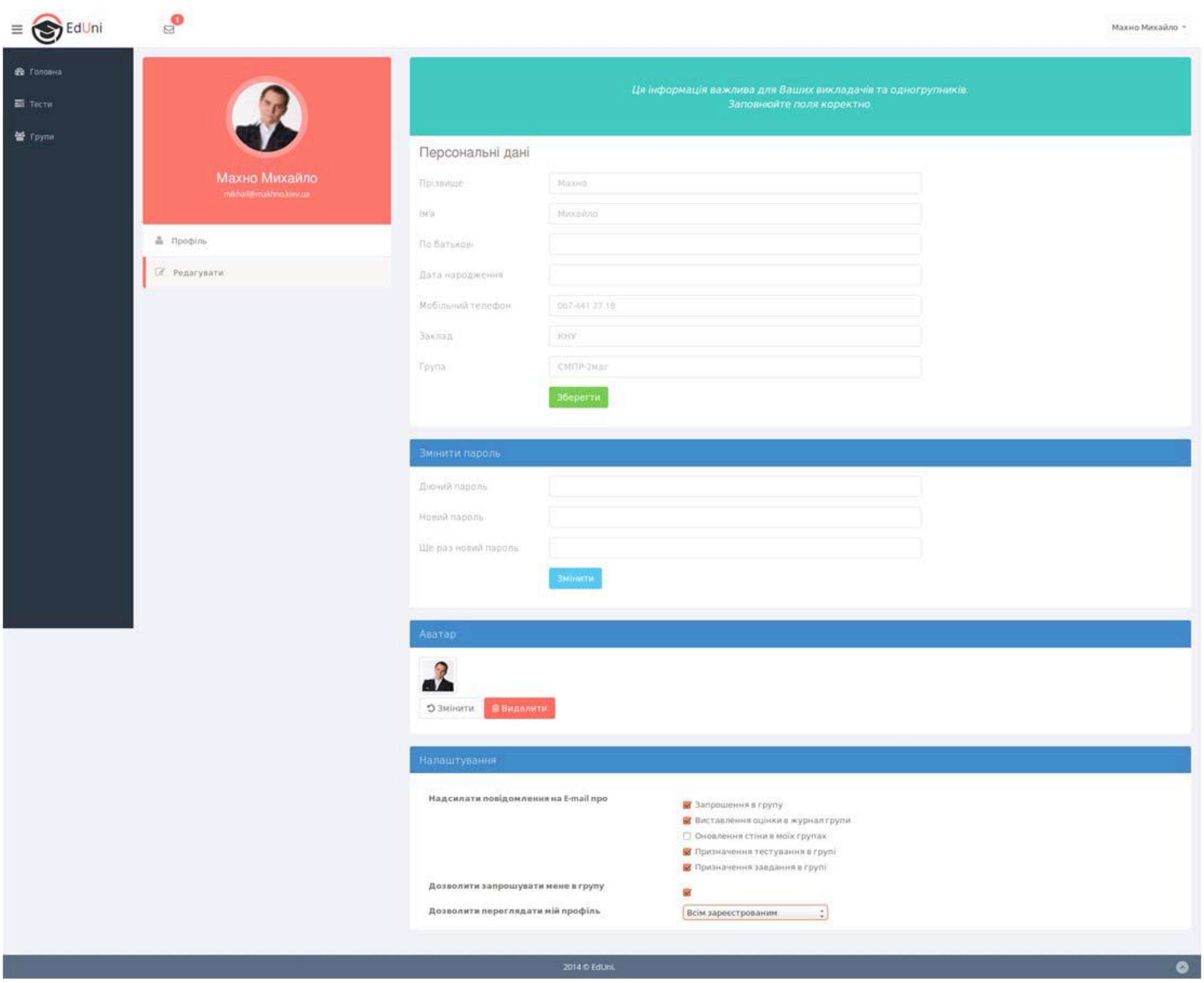
Level of task	Ball's cost
0.641250	1
1.134375	1
2.635000	2
3.338750	2
9.537500	5
9.538750	5
12.74187	7
12.88499	7
20.00000	10
20.00000	10



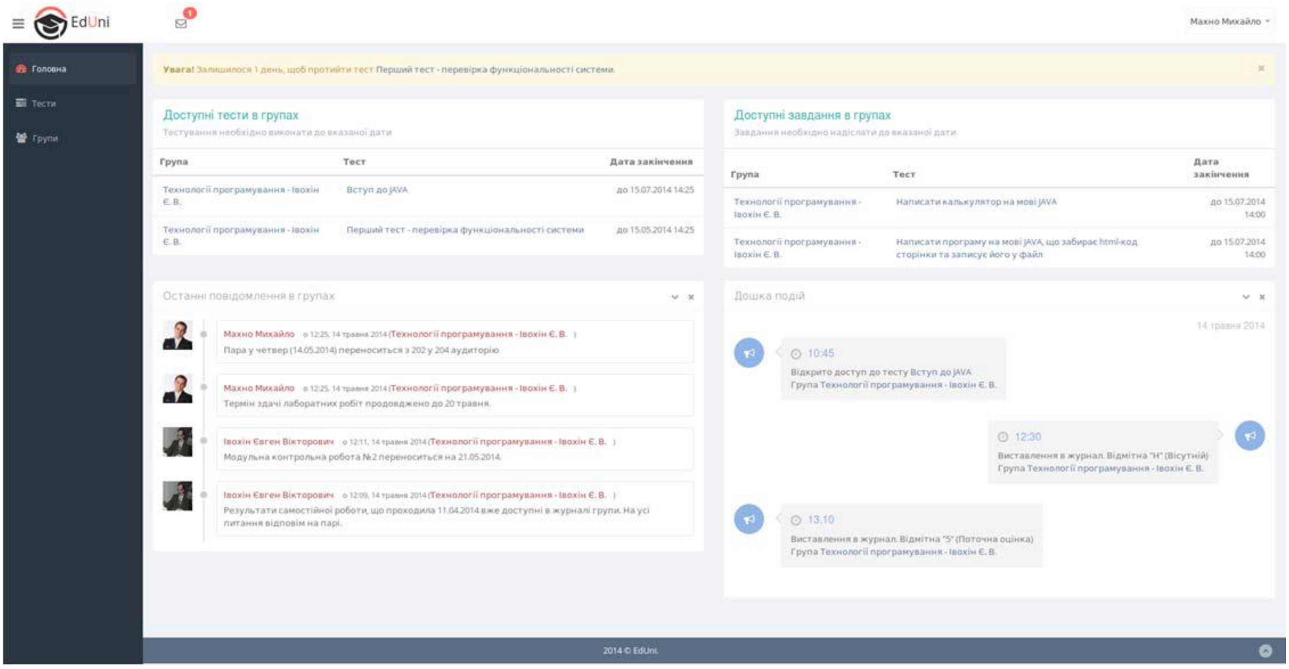
Додаток В. Опис системи EdUni для проведення дистанційних тестувань.



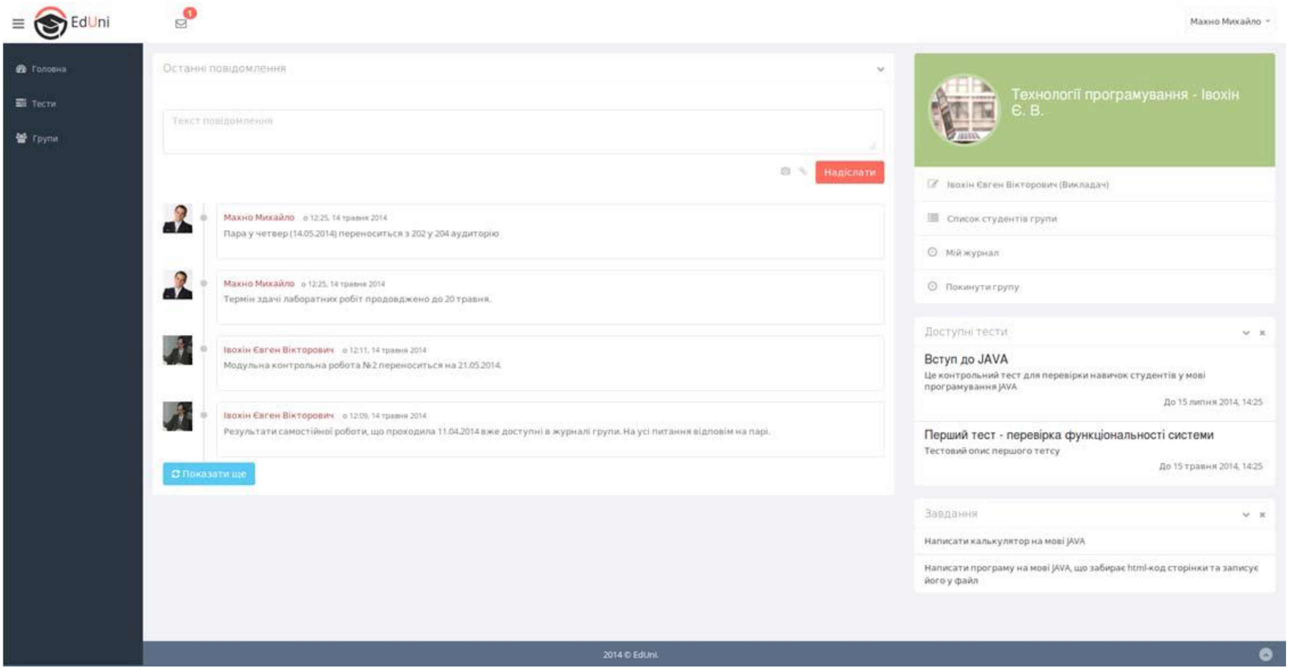
Мал. В.1. - Реєстрація в системі



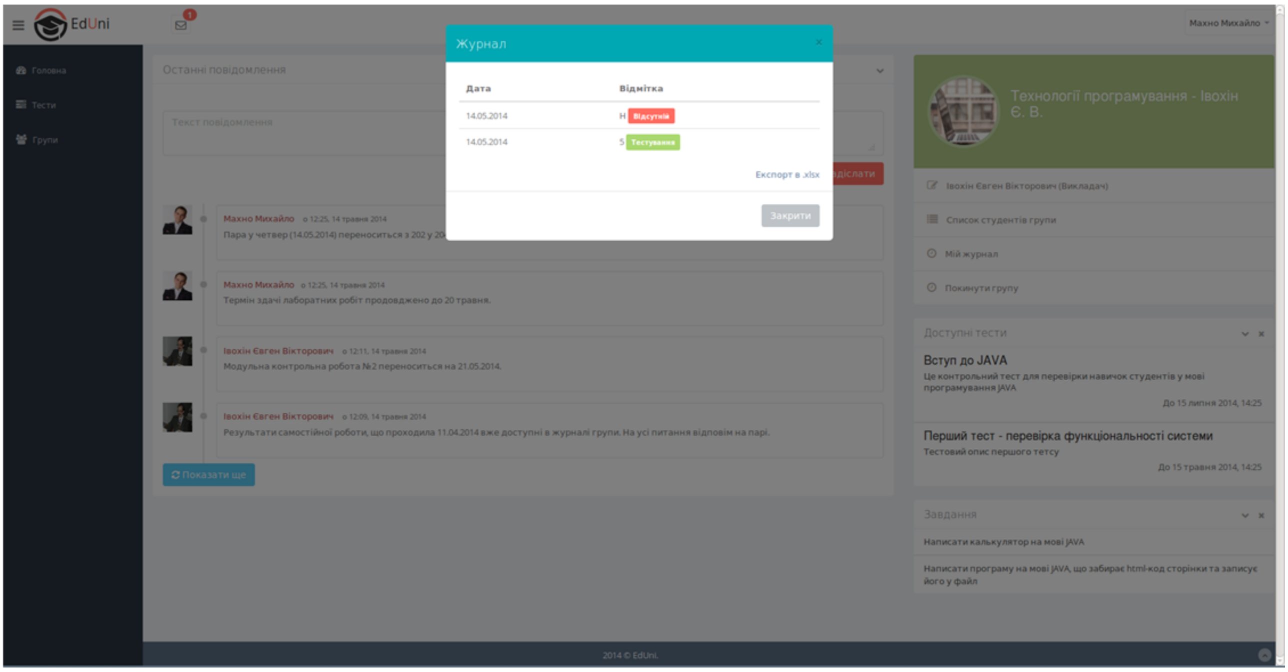
Мал. В.2. – Редагування профілю користувача



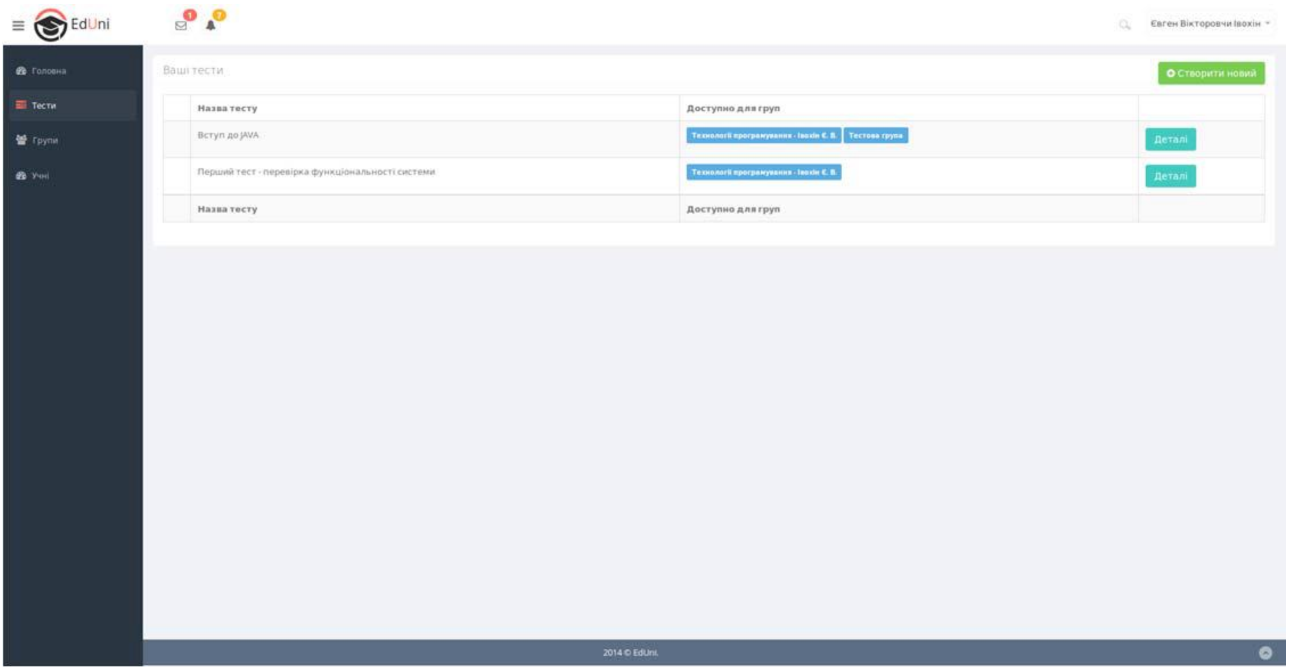
Мал. В.3. – Інформаційна панель студента



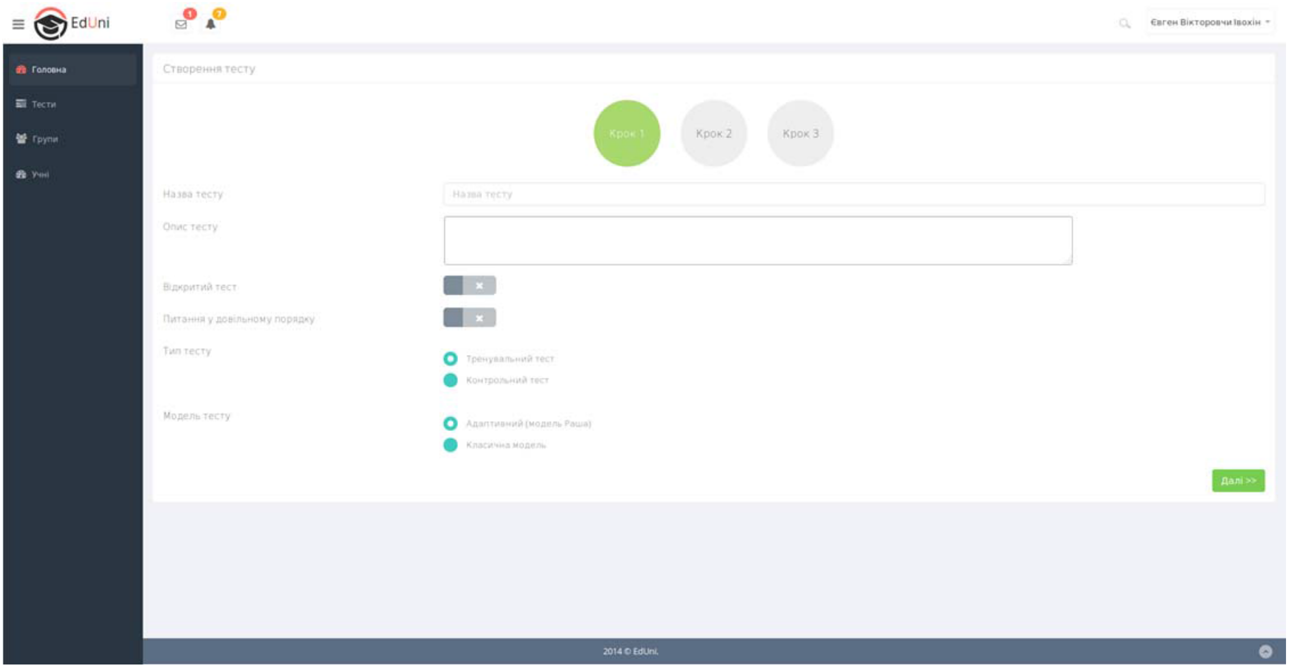
Мал. В.4. – Сторінка групи



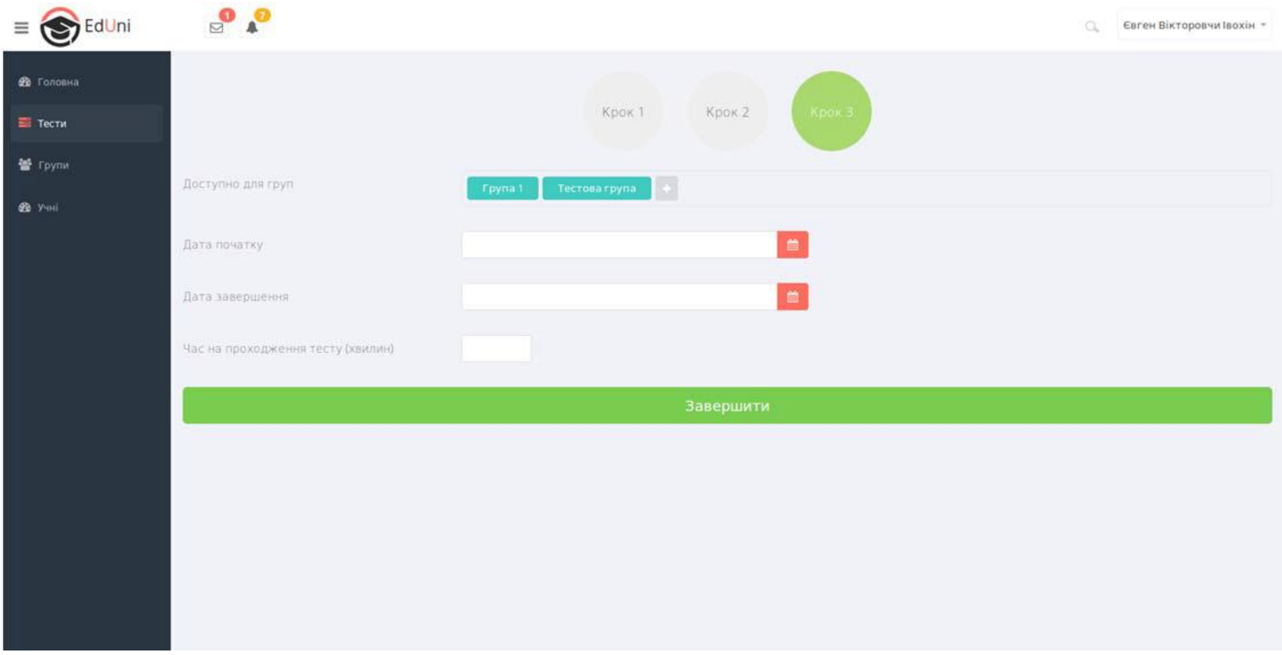
Мал. В.5. – Журнал



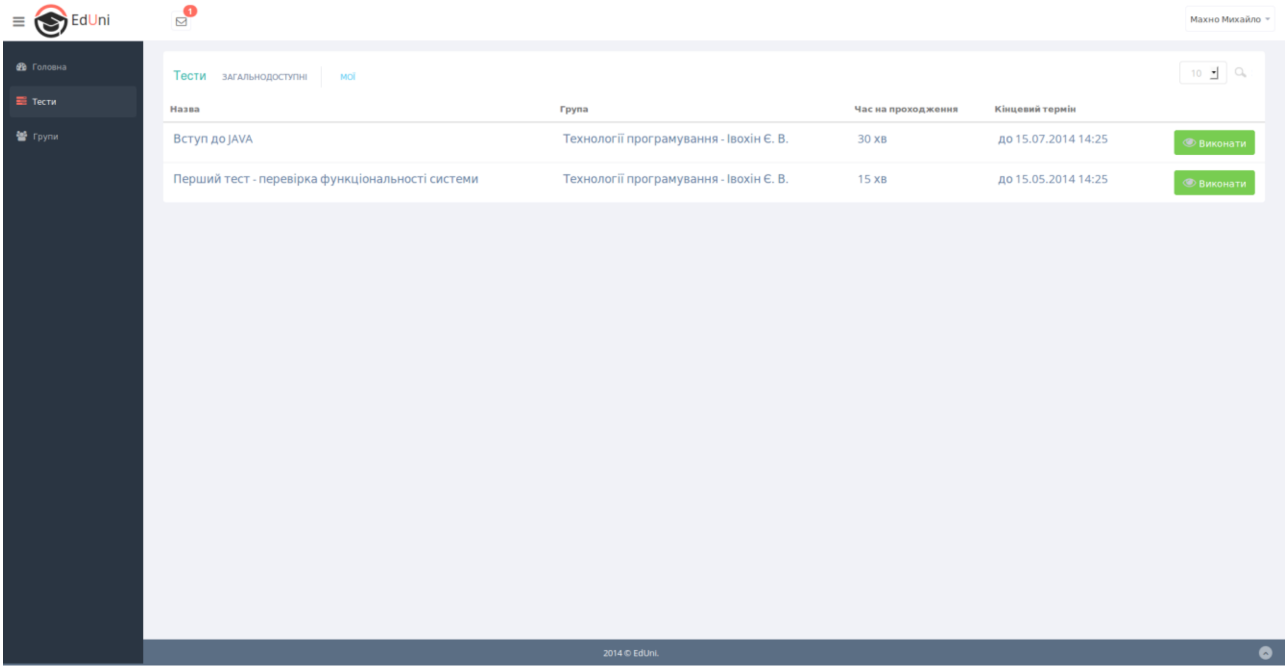
Мал. В.6. – Список створених тестів



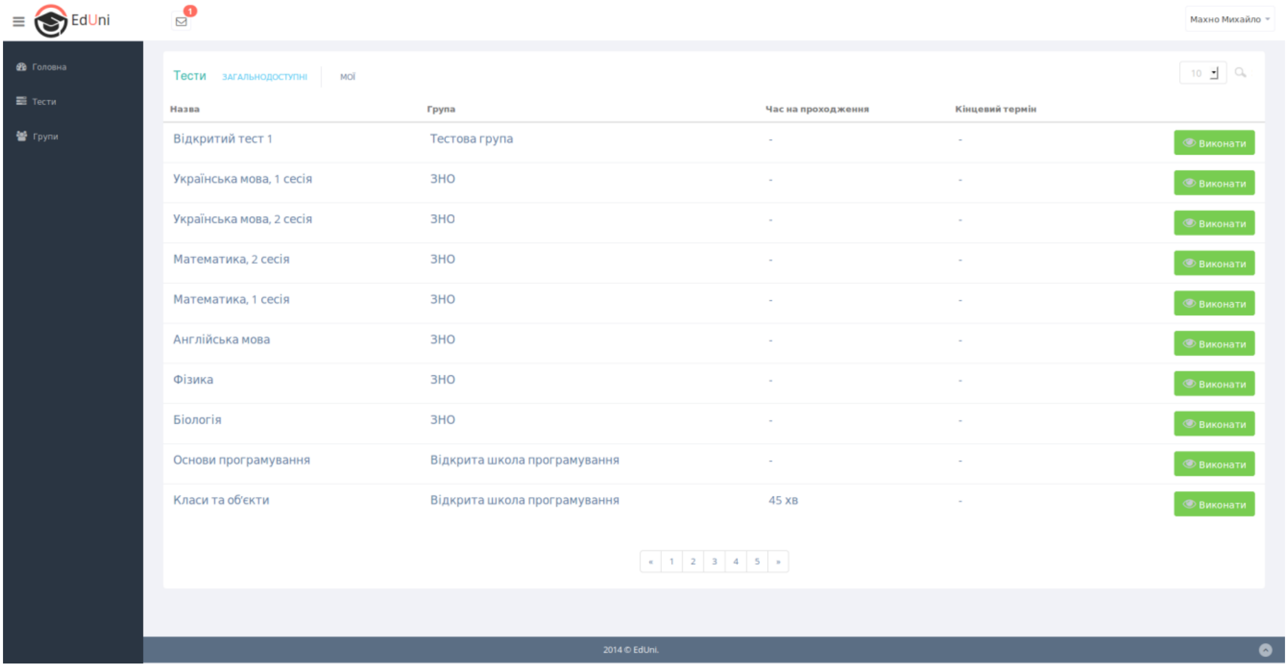
Мал. В.7. – Створення тесту



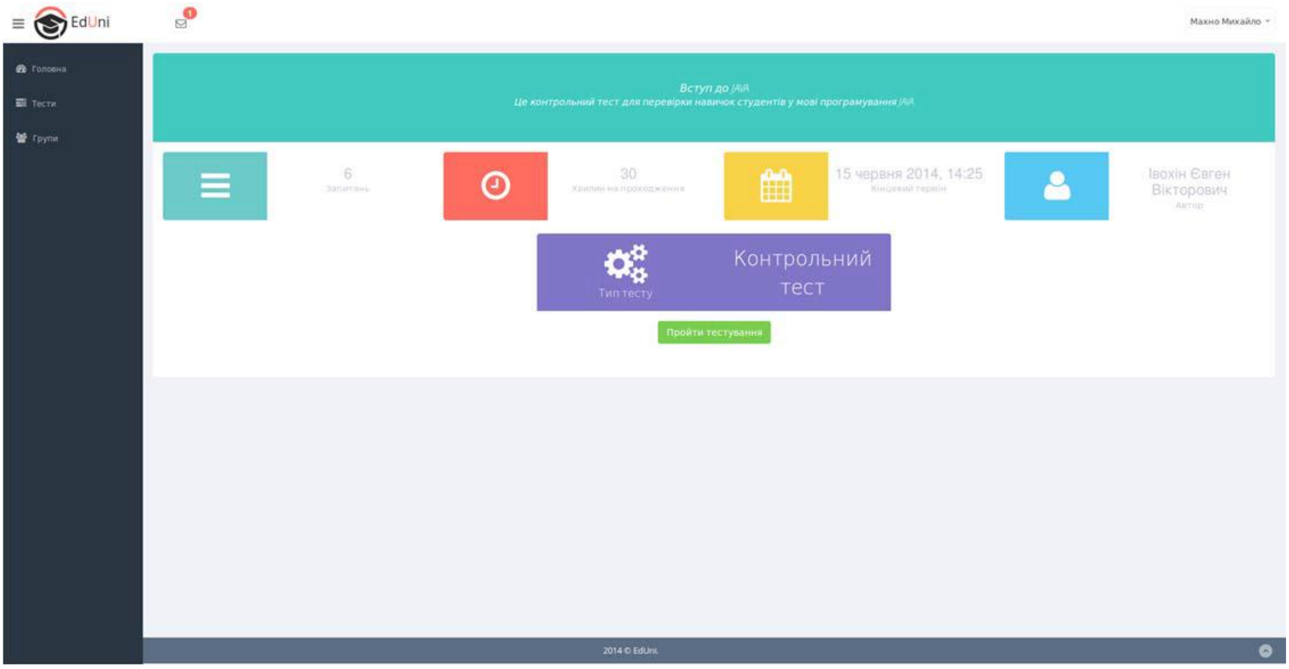
Мал. В.8. – Останній крок створення тесту



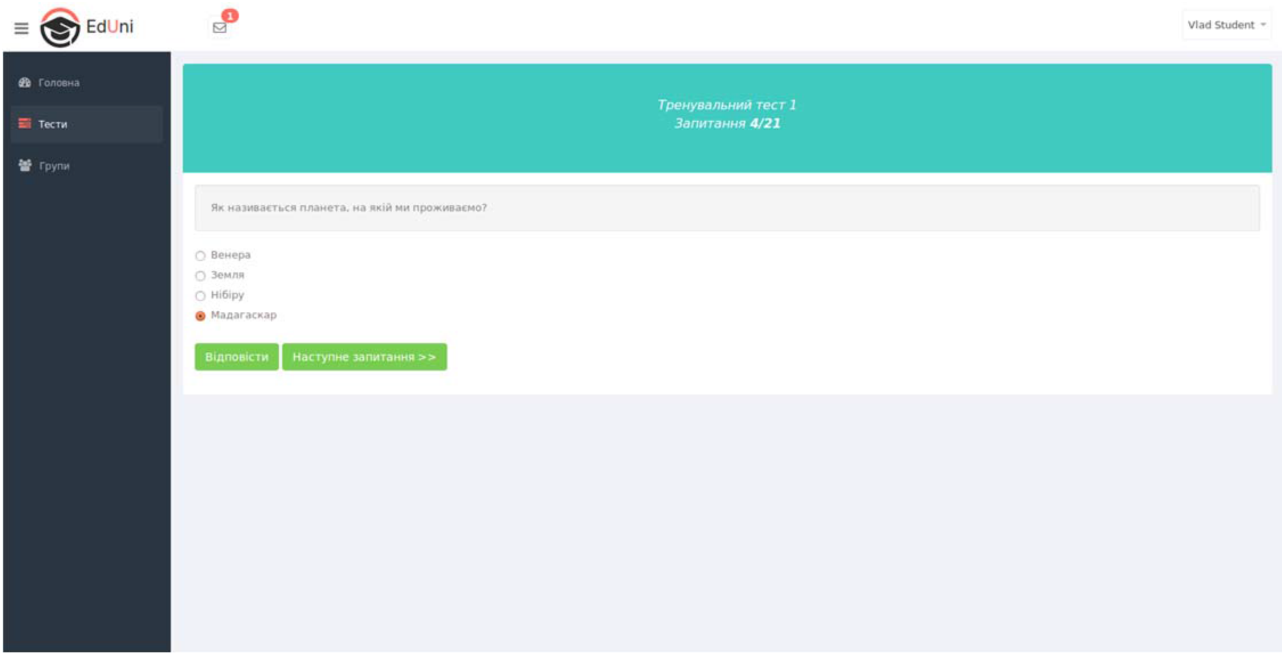
Мал. В.9. – Тести, доступні для проходження користувачеві



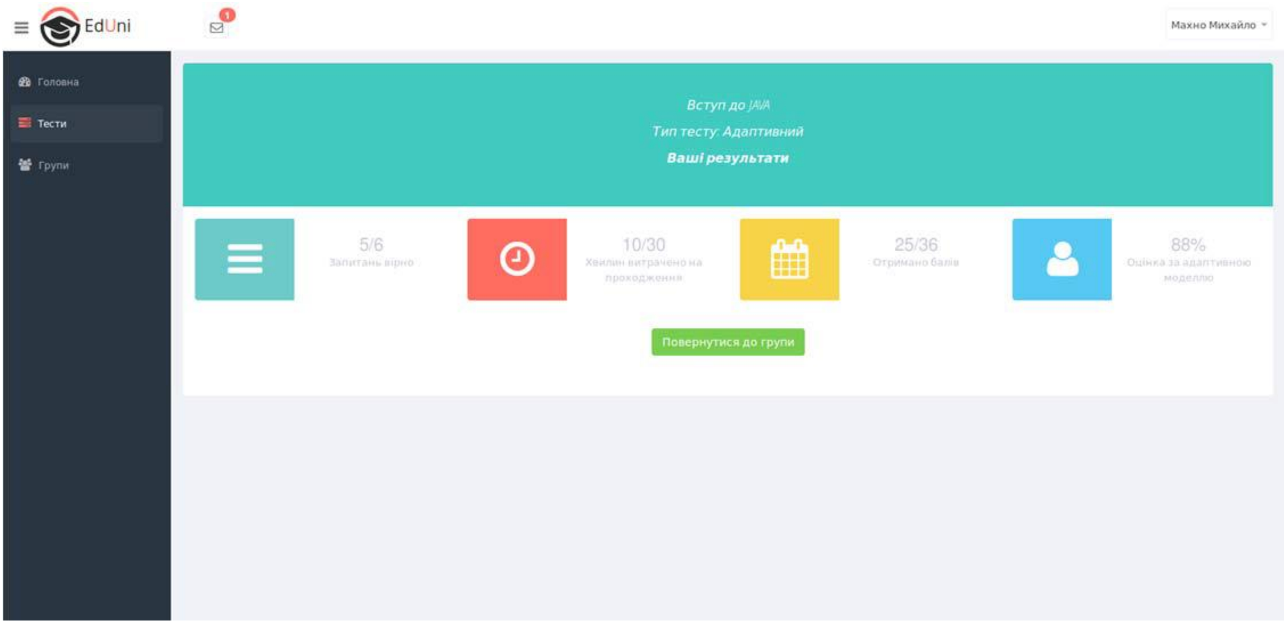
Мал. В.10. – Усі відкриті тести



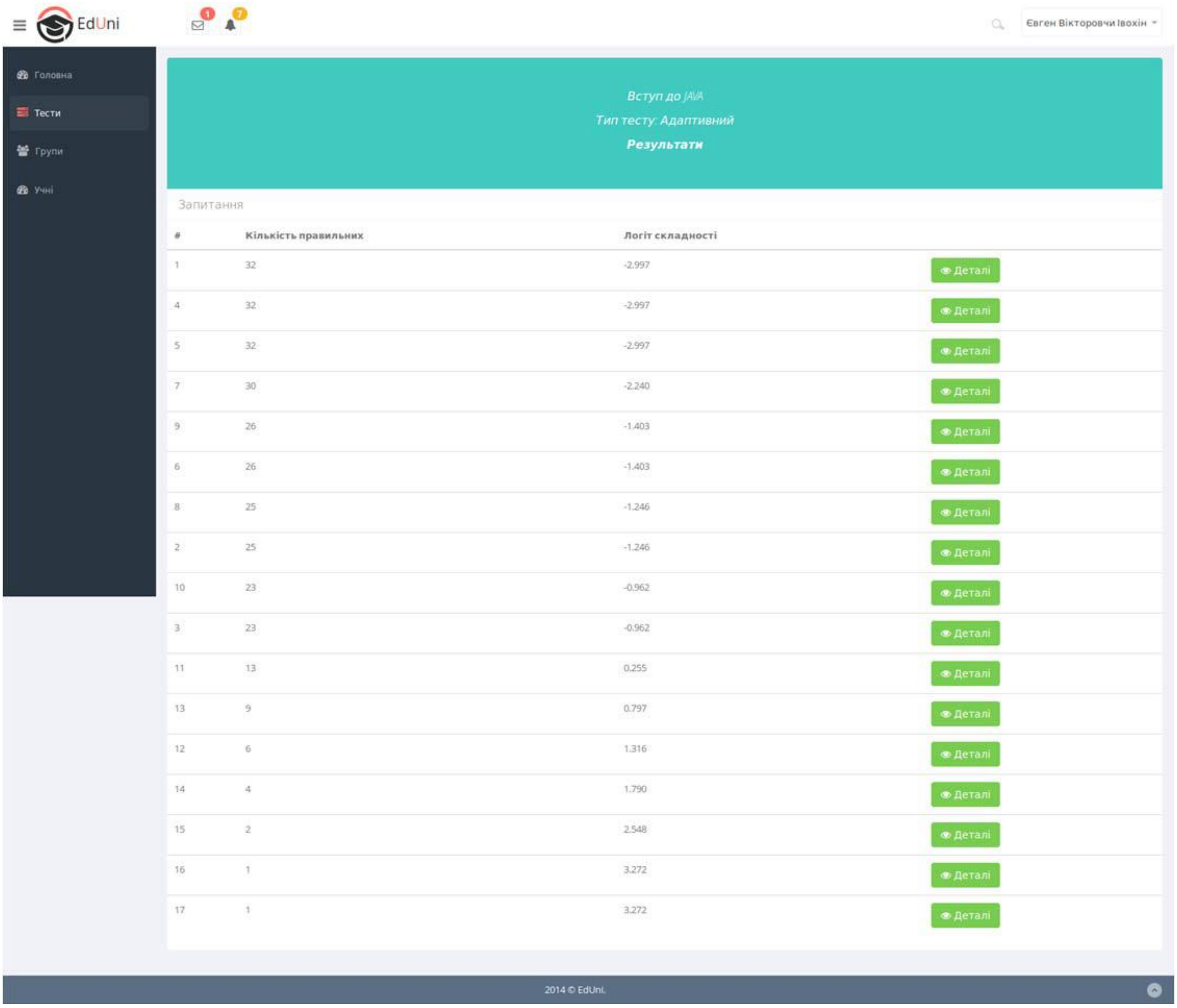
Мал. В.11. – Сторінка з інформацією про тест



Мал. В.12. – Проходження тесту



Мал. В.13. – Результати тестування для студента



Мал. В.14. – Результати тестування для викладача

Додаток Г. Наукові опубліковані праці

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Махно М.Ф. Про гібридну модель динаміки процесу обробки сукупності завдань / М.Ф.Махно // Вісник КНУ імені Тараса Шевченка. Серія: Кибернетика. – 2015. – №1(15). – С.22-27.
2. Івохін Є.В. Розробка засобів адаптивного тестування та автоматичного оцінювання знань / Є.В.Івохін, М.Ф.Махно // Вісник Київського університету ім. Тараса Шевченка. Серія: Фізико-математичні науки. – 2015. – №1. – С.130-133.
3. Івохін Є.В. Про один підхід до розв'язання нечіткої задачі рекомбінації / Є.В.Івохін, М.Ф.Махно // Вісник КНУ імені Тараса Шевченка. Серія: Фізико-математичні науки. – 2016. – №2. – С.94-97.
4. Івохін Є.В. Про наближений розв'язок чіткої та нечіткої задачі лінійного програмування з системою обмежень великої розмірності / Є.В.Івохін, В.О.Навродський, М.Ф.Махно //Вісник КНУ імені Тараса Шевченка. Серія: Фізико-математичні науки. – 2016. – №3. – С.69-72.
5. Ивохин Е.В. О подходе к построению структурированных нечетких множеств и их использовании для описания нечеткого отсчета времени/ Е.В.Ивохин, М.Ф.Махно // Проблемы управления и информатики. – 2017. – №5. – С.147-156.
6. Івохін Є.В. Про одну гібридну модель задачі теорії розкладів / Є.В.Івохін, М.Ф.Махно // Сборник трудов Міжнародной научной конференции имени Т. А.Таран "Интеллектуальний анализ информации" (IAI-2016), Київ, 18-20 травня 2016. – С.90-96.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

7. Івохін Є.В. Про підхід до формалізації процесу тестування на основі гібридної моделі задачі розкладів / Є.В.Івохін, М.Ф.Махно // Матеріали VII міжнар. конф. ["Сучасні проблеми математичного моделювання,

прогнозування та оптимізації"], (Камя’нець-Подільський національний університет імені Івана Огієнка, березень 2016). - Камя’нець-Подільський, 2016. - С. 85.

8. Makhno M. On one hybrid model of a schedule theory problem/ M.Makhno, D.Apanasenko//Abstracts XXVII International Conference [“Problems of descision making under uncertainties”], (May 23-27, 2016, Tbilisi-Batumi, Georgia). Tbilisi-Batumi, 2016. – P.108.
9. Івохін Є.В. Про один підхід до розв’язання нечіткої задачі теорії розкладів / Є.В.Івохін, М.Ф.Махно // Тези VIII міжнар. школи-сем. [«Теорія прийняття рішень»], (Ужгорож, 26.09-01.10.2016). - Ужгорож, 2016. – С.129.
10. Івохін Є.В. Про спосіб перетворення області допустимих розв’язків в чітких та нечітких задачах лінійного програмування/ Є.В.Івохін, М.Ф.Махно// Матер. IV міжнар. наук.-техн. конф. [“Обчислювальний інтелект” (ОІ-2017)], (Київ, 16-18 травня 2017р.). - Київ, 2017. – С.118-119.