

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**Розробка методу та програмних засобів фінансування досліджень
поширення COVID-19**

Виконав: студент II курсу групи 1ПІ-20м
спеціальності 121 – Інженерія програмного забезпечення

Ковальчук Дмитро
Олександрович

Керівник: к.т.н., доц. каф. ПЗ Хошаба О.М.

«_____»2021 р.

Опонент: д.т.н., проф. каф. Васілевський О.М.

«_____»2021 р.

Допущено до захисту

Завідувач кафедри ПЗ

д.т.н., проф. Романюк О. Н.

(прізвище та ініціали)

«_____»_2021 р.

ВНТУ – 2021

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення
Рівень вищої освіти II-й (магістерський)
Галузь знань 12 – Інформаційні технології
Спеціальність 121 – Інженерія програмного забезпечення
Освітньо-професійна програма – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ
Завідувач кафедри ПЗ
Романюк О. Н.
« 13 » вересня 2021 р.

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

Ковальчуку Дмитру Олександровичу

1. Тема роботи – розробка методу та програмних засобів фінансування досліджень поширення COVID-19.

Керівник роботи: Хошаба Олександр Мирославович, к.т.н., доцент кафедри ПЗ, затверджені наказом вищого навчального закладу від « 24 » вересня 2021 р. № 277.

2. Строк подання студентом роботи

1 грудня 2021 р.

3. Вихідні дані до роботи: системний аналіз процесу дерева рішень, формальні методи дерева рішень, основні етапи типового процесу прийняття рішення - не менше трьох етапів, моделі прийняття рішень - для умов нелінійних розподілених задач, моделі алгоритму - прямий та зворотній прогон для задач упорядкування, методи еволюційних та евристичних алгоритмів, основна теорема антагоністичних ігор - дві особи, математична модель дерева графу - на конкретному прикладі фінансування досліджень.

4. Зміст розрахунково-пояснювальної записки: аналіз сучасного стану методів на основі дерева рішень, системний аналіз процесу дерева рішень, формальні методи дерева рішень, основні етапи типового процесу прийняття рішення, моделі прийняття рішень в умовах нелінійних розподілених задачах, моделі алгоритму прямого і зворотного прогону для задач упорядкування, методи еволюційних та евристичних алгоритмів, основна теорема антагоністичних ігор двох осіб, розробка інтерфейсу програми; розробка загального алгоритму роботи програми.

5. Перелік графічного матеріалу: модель особи яка приймає рішення, постановка задачі використання теорії дерева рішень, етапи типового процесу прийняття рішення, розробка дерева рішень для задачі з реалізації проектів, алгоритм роботи програмного засобу, скріншоти роботи програмного засобу.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Хошаба О. М., к.т.н., доцент кафедри ПЗ	15.09.2021	01.12.2021
5	Ратушняк О.Г., к.т.н., доцент кафедри ЕПВМ	14.11.2021	30.11.2021

7. Дата видачі завдання 14 вересня 2021 р.

КАЛЕНДАРНИЙ ПЛАН

/п	Назва етапів магістерської кваліфікаційної Роботи	Строк виконання етапів роботи	Примітка
1	Аналіз сучасного стану методів та моделей на основі дерева рішень	15.09.2021 – 31.10.2021	Вик.
2	Розробка моделей прийняття рішень на основі теорії дерева рішень	31.10.2021 – 07.11.2021	Вик.
3	Розробка методів та моделей на основі теорії ігор	07.11.2021 – 10.11.2021	Вик.
4	Розробка блок-схеми алгоритму та програмного засобу	10.11.2021 – 14.11.2021	Вик.
5.	Економічна частина	14.11. 2021 – 30.11.2021	Вик.

Студент _____ Ковальчук Д. О.
(підпис) (прізвище та ініціали)

Керівник магістерської кваліфікаційної роботи _____ Хошаба О. М.
(підпис) (прізвище та ініціали)

Опонент магістерської кваліфікаційної роботи _____ Васілевський
О.М.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

УДК 004.338:578.834.1

Ковальчук Д. О. Розробка методу та програмних засобів фінансування досліджень поширення COVID-19. Магістерська кваліфікаційна робота зі спеціальності 121 – інженерія програмного забезпечення, освітня програма – інженерія програмного забезпечення. Вінниця: ВНТУ, 2021., 133 с.

У магістерській кваліфікаційній роботі розроблено програмний засіб з рішення задачі в галузі проведення досліджень з фінансування досліджень поширення COVID-19.

Під час виконання магістерської кваліфікаційної роботи проведено аналіз сучасного стану методів на основі дерева рішень, де визначені основні напрямки в дослідженні процесу, розглянуті моделі прийняття рішень на основі теорії дерева рішень, де визначались основні етапи типового процесу прийняття рішення. Розглянуті моделі прийняття рішень на основі теорії ігор, де визначалась формалізація моделі гри.

Дістав подальшого розвитку метод прийняття рішень в умовах нелінійних розподілених задачах, який, на відміну від існуючих, придатний для проведення досліджень в галузі фінансування досліджень, що на певних рівнях дерева прибутків визначає також й збитки.

Реалізація контрольного прикладу рішення задачі виконано на мові високого рівня Java, який реалізує складені вимоги, що розроблені у постановці задачі.

Ключові слова: програмний додаток, дерево рішень, теорія ігор, прийняття рішень, мова програмування Java.

ABSTRACT

Kovalchuk D.O. Development of a method and software for financing research on the spread of COVID-19. Master's degree in specialty 121 – software engineering, educational program – software engineering. Vinnytsia: VNTU, 2021. 133p.

In the master's qualification work, a software tool was developed to solve the problem in the field of research on the financing of research on the spread of COVID-19.

During the master's qualification work, an analysis of the current state of methods based on the decision tree, which identified the main directions in the study of the process, considered models of decision-making based on decision tree theory, which determined the main stages of a typical decision-making process. Models of decision-making based on game theory are considered, where the formalization of the game model was determined.

The method of decision-making in the conditions of nonlinear distributed problems, which, unlike the existing ones, is suitable for research in the field of research funding, which at certain levels of the profit tree also determines losses, has been further developed.

Implementation of the control example of the problem solution is performed in the high-level Java language, which implements the complex requirements developed in the problem statement.

Keywords: software application, decision tree, game theory, decision making, Java programming language.

ЗМІСТ

ВСТУП.....	8
1 АНАЛІЗ СУЧАСНОГО СТАНУ МЕТОДІВ НА ОСНОВІ ДЕРЕВА РІШЕНЬ.	11
1.1. Основні напрямки в дослідженні процесу дерева рішень.....	11
1.2. Загальні положення теорії дерева рішень	14
1.3. Відомі обмеження у використанні теорії дерева рішень.....	17
1.4. Класифікація задач, які використовують теорію дерева рішень.....	20
1.5. Системний аналіз процесу дерева рішень.....	24
1.6. Формалізація цілі у методах дерева рішень.....	26
1.7. Використання формальних методів дерева рішень.....	30
2 МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕОРІЇ ДЕРЕВА РІШЕНЬ....	32
2.1. Визначення основних етапів типового процесу прийняття рішення....	32
2.2. Моделі прийняття рішень в умовах нелінійних розподілених задачах.....	37
2.3. Моделі алгоритм прямого і зворотного прогону для задач упорядкування..	40
2.4. Використання методу еволюційних алгоритмів.....	43
2.5. Визначення показників ефективності евристичних алгоритмів.....	47
3 МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕОРІЇ ІГОР.....	49
3.1. Формалізація моделі гри.....	49
3.2. Використання основної теореми антагоністичних ігор двох осіб.....	51
3.3. Використання методу лінійного програмування.....	56
3.4. Використання моделі багатокритеріальних задач.....	59
3.5. Прийняття рішень на основі системного підходу.....	63
3.6. Математичний підхід до прийняття рішень на основі теорії ігор.....	68
4 РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАСОБУ З ФІНАНСУВАННЯ	
ДОСЛІДЖЕНЬ.....	72
4.1. Постановка задачі до розробки програмного засобу на основі методу	
дерева у прийнятті рішень.....	72
4.2. Основні етапи проектування програмного засобу на основі методу	
дерева рішень.....	75
4.3. Вибір математичних моделей дерева графу.....	79
4.4. Використання різних математичних моделей у вигляді графів.....	84

4.5 Розробка основної структури програмного засобу рішення задачі.....	86
4.6 Реалізація контрольного прикладу рішення задачі.....	88
5 ЕКОНОМІЧНА ЧАСТИНА.....	92
5.1 Оцінювання комерційного потенціалу розробки.....	92
5.2 Прогнозування витрат на виконання науково-дослідної роботи.....	95
5.3 Розрахунок економічної ефективності науково-технічної розробки....	100
5.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності.	101
5.5	
Висновки.....	104
ВИСНОВКИ.....	105
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	108
ДОДАТКИ.....	112
ДОДАТОК А. Технічне завдання.....	113
ДОДАТОК Б. Протокол перевірки.....	117
ДОДАТОК В. Лістинг коду.....	118
ДОДАТОК Г. Ілюстративна частина.....	129

ВСТУП

Обґрунтування вибору теми дослідження. Свідома діяльність людини нерозривно пов'язана з процесами прийняття рішень на основі дерева рішень. Так, люди завжди приймали рішення, ґрунтуючись на своєму досвіді, інтуїції та здоровому глузді. При цьому, як правило, точний шлях, що привів до вибору рішення, сам автор рішення писати не в змозі, хоча є підстави вважати, що він якимось чином враховував і зважував усі аспекти прийнятого рішення. Уміння приймати рішення, що дають найкращі результати у різних складних ситуаціях завжди розглядалося як мистецтво.

Саме тим часом публікуються дослідження математиків-економістів Курно, Квінсі, Вальраса. Трохи пізніше Парето, які започаткували теорію ігор; розробляються основи лінійного програмування Жорданом, Мінковським, Фаркашем; теорії прийняття рішень (ТПР) почалося з середини століття і призвело до утворення таких самостійних математичних дисциплін як теорія ігор, багатокритеріальний вибір і векторна оптимізація, математичне програмування. Причому остання дисципліна включає такі цілком самостійні розділи як лінійне програмування, опукле програмування, дискретне програмування, календарне планування, стохастичне програмування і т.д.

Але, в теперішній час, в зв'язку з погрозою розвитку пандемії COVID-19 досить актуальними стають питання боротьби на рівні організаційних заходів, а саме удосконалення методів прийняття рішень на основі теорії дерев, де важливими є фінансування досліджень та заохочення інвесторів за допомогою надання відомостей з результатів роботи математичних моделей. Саме цьому й присвячена робота.

Мета та завдання дослідження:

Метою роботи є підвищення ефективності проведення фінансування досліджень поширення COVID-19 за рахунок використання методу дерева рішень.

У відповідності до поставленої мети потрібно виконати такі **завдання**:

- виконати аналіз сучасного стану методів на основі дерева рішень;
- визначити моделі прийняття рішень на основі теорії дерева рішень та теорії ігор;
- виконати розробку та впровадження програмного засобу з фінансування досліджень COVID-19.

Об'єктом дослідження є процес прийняття рішень у дослідженнях поширення COVID-19.

Предметом дослідження є методи та засоби прийняття оптимальних рішень у дослідженнях поширення COVID-19.

Методи дослідження. У процесі дослідження використовувались: теорія дерева рішень, теорія ігор, системний аналіз процесу дерева рішень, моделі багатокритеріальних задач, моделі прийняття рішень в умовах нелінійних розподілених задачах, моделі алгоритму прямого і зворотного прогону для задач упорядкування, методи лінійного програмування, методи еволюційних та евристичних алгоритмів.

Наукова новизна одержаних результатів:

- подальшого розвитку дістав метод прийняття рішень в умовах нелінійних розподілених задачах, який, на відміну від існуючих, придатний для проведення досліджень в галузі фінансування досліджень, що на рівні прибутків визначає також й збитки, і орієнтований на обчислення показників за допомогою імовірнісних моделей, що забезпечує можливість прийняття рішень відносно інвестування визначеного проекту;

- удосконалено методи прямого і зворотного прогону для задач упорядкування даними економічного характеру, який, на відміну від існуючих, дозволяють приймати рішення відносно обрання заданого проекту фінансування досліджень, що надає можливість зменшити інвестиційні збитки та підвищити ймовірність отримання прибутків;

- здобуло подальший розвиток метод еволюційних алгоритмів, який, на відміну від існуючих, на основі даних, які визначають сукупність гілок дерева рішень як можливі шляхи вирішення задачі задає спосіб кодування параметрів як «хромосоми», що дозволяє створити вихідну популяцію в якості початкового

рішення та надає можливість розраховувати функцію пристосованості кожної особини, що позначає з великим ступенем ймовірності конкретну виграшну дію для прийняття рішення з фінансування проекту серед всіх можливих пропозицій.

Практична цінність отриманих результатів. Практичне значення отриманих результатів полягає в тому, що на основі отриманих в магістерській кваліфікаційній роботі теоретичних положень запропоновано алгоритми та розроблено програмні засоби фінансування досліджень поширення COVID-19 на основі методу дерева.

Особистий внесок здобувача. У магістерській кваліфікаційній роботі усі результати дослідження здобуті автором даної роботи самостійно. У роботі [41], опублікованій самостійно, автору належить формалізація проблеми фінансування досліджень поширення COVID-19, розкриття мети роботи, аналізу та пропонування рішення проблеми з фінансування досліджень поширення COVID-19, розгляд контрольного прикладу з реалізації проектів фінансування досліджень поширення COVID-19.

Апробація матеріалів магістерської кваліфікаційної роботи. Основні положення магістерської кваліфікаційної роботи доповідалися та обговорювалися на Міжнародній науково-практичній Інтернет конференції "Електронний інформаційні ресурси: створення, використання, доступ" (Суми/Віннця, 2021).

Публікації. Основні результати досліджень опубліковано в одноосібній науковій праці у матеріалах конференції.

РОЗДІЛ 1

АНАЛІЗ СУЧАСНОГО СТАНУ МЕТОДІВ НА ОСНОВІ ДЕРЕВА РІШЕНЬ

1.1. Основні напрямки в дослідженні процесу дерева рішень

Важливість проблем, пов'язаних з аналізом на основі дерева рішень, привертає увагу широкого кола науковців і практичних працівників, що представляють часом далеко віддалені один від одного області наукових знань, такі як математика, кібернетика, психологія, соціологія, логіка, нейрофізіологія, економіка, право, технічні науки.

Значні успіхи в галузі вирішення цих проблем та розвитку електронно-обчислювальної техніки зробили реальною практичну реалізацію в різних додатках багатьох математичних результатів (які до цього мали суто теоретичне значення) та стимулювали розробку нових математичних моделей та методів на основі дерева рішень [1,2].

З іншого боку, у зв'язку з різким зростанням темпів науково-технічного процесу, загострення конкуренції, зростання динамізму навколишнього середовища, перед керівниками, колективами, організаціями постійно виникають завдання прийняття складних технічних, економічних, соціальних, організаційно-управлінських рішень. Причому, типовою є така ситуація, коли наслідки прийнятих рішень стосуються великої кількості людей, пов'язані з великими матеріальними витратами, що зумовлює багаторазове підвищення ступеня відповідальності за наслідки прийнятого рішення. Як приклад таких завдань подаємо наступний перелік сучасних напрямків в галузі фінансування досліджень:

- формування виробничої програми підприємства;
- розподіл матеріальних ресурсів;
- вибір технологічного оснащення виробництва;
- визначення кращого проектного рішення;
- підбір виконавця для відповідальної роботи;
- оперативне управління технологічним процесом.

Ознайомившись з цим далеко не повним переліком, можна зробити висновок, що складні завдання прийняття рішень мають місце як під час проектування, так і при функціонуванні установ різних рівнів. Причому ці завдання характеризуються такими особливостями: велика розмірність, багатоаспектний характер, слабка структурованість. Це зумовлює необхідність застосування математичних методів теорії прийняття рішень при автоматизації процесу розв'язання таких завдань.

Ефективність його автоматизації багато в чому визначає ефективність функціонування відповідної установи.

Теорія прийняття рішень як самостійна наукова дисципліна бере свій початок з робіт Дж. Фон Неймана і О. Моргенштерна. Вона розвивалася не шляхом послідовного і поступового узагальнення експериментальних даних, доводячи їх до розробки найзагальніших висновків і положень, а зовсім навпаки - як практична реалізація системного підходу. Були спроби описати основні елементи процесу прийняття рішень на основі формально-логічних і математичних методів.

При цьому вихідною точкою теоретичних досліджень були найзагальніші уявлення про сутність процесу вирішення проблем та можливості використовуваних формалізованих методів їх опису та моделювання. Математичні методи використовувалися для того, щоб знайти найбільш ефективний шлях досягнення певної мети. Вони відповідають на питання як, а не що оптимізувати.

У сучасній теорії в сукупності методів прийняття рішень поєднуються не тільки чисто наукові, а й такі підходи, які характеризують мистецтво прийняття рішень, тобто організаційні, інтелектуальні та психологічні здібності і досвід людей [2,3].

Зазвичай виділяють два напрямки в дослідженні процесу прийняття рішень:

- 1) нормативне;
- 2) дескриптивне.

У першому випадку дослідники будують теорію на вивченні формалізованих методів, переважно математичних моделей. У другому випадку дослідники будують свої положення на основі вивчення психології процесу прийняття рішень, поведінки особи, що приймає рішення.

Методи нормативного підходу на основі дерева рішень дозволяють оцінювати проблемну ситуацію, враховувати ресурси та обмеження, аналізувати різні варіанти дій, вибирати з них оптимальні, використовуючи формальні, як правило, математичні методи. Знання дескриптивної (або, як ще її називають, психологічної) теорії прийняття рішень дозволяє краще зрозуміти, як люди роблять свій вибір, які об'єктивні і суб'єктивні чинники впливають на правильність прийнятого ними рішення, які психологічні механізми керують процесом прийняття рішень, яка роль соціального середовища, в якій діє керівник, які механізми групової поведінки.

Обидва підходи до прийняття рішень на основі дерева рішень - і нормативний, і дескриптивний є необхідними і доповнюють один одного. Існують такі області діяльності (зміни в організаційних структурах, маркетинг, вибір варіантів інвестицій, визначення виробничої або зовнішньополітичної стратегії та ін.), де можна прийняти рішення тільки на основі якісних характеристик процесів, що відбуваються або обійтися одними математичними методами.

Численні дослідження показують, що без додаткової аналітичної підтримки рішення приймаються спрощені, а іноді і просто суперечливі. Очевидно, що складні проблеми, що виникають в економіці, політиці, екології, управлінні, технічних системах (для яких характерний високий рівень невизначеності), в принципі не можна сформулювати у вигляді точно поставлених математичних завдань - для цього треба тим чи іншим чином зняти невизначеність. Тому дослідження проблем прийняття рішення в умовах невизначеності не може бути вичерпано тільки формальними методами. Уміння експерта, професіонала в цій галузі, збройного аналітичними методами, буває не тільки необхідним, але часом і вирішальним. Під аналітичними методами будемо розуміти повністю або частково формалізовані процедури розробки

рішення, починаючи з формальних моделей лінійного програмування і закінчуючи імітаційним моделюванням, діловими іграми та іншими методами, що включають суб'єктивні судження і інтуїцію.

Для застосування кількісних методів дослідження в будь-якій області завжди потрібно будь-яка математична модель. При побудові моделі реальне явище неминуче спрощується, схематизується, і ця модель описується за допомогою того чи іншого математичного апарату. Чим вдаліше підібрана математична модель, чим краще вона відображає характерні риси явища, тим корисніше впливають з її використання рекомендації. Люди, які не володіють професійними математичними методами, іноді думають, що будь-яка проблема може бути переведена на мову математики і, отже, вирішена її засобами. Нерідко можна зустріти і протилежну точку зору. Відомий американський математик, творець кібернетики Н. Вінер вважав принцип невизначеності настільки суттєвою особливістю соціальних систем, що, на його думку, математичний апарат, розроблений для опису фізичних і навіть біологічних процесів, взагалі непридатний для соціально-економічних об'єктів.

Більшість проблем пов'язано з прийняттям рішення в умовах невизначеності, що далеко не завжди дозволяє будувати формальну модель пошуку оптимального рішення. Модель найчастіше дозволяє лише звузити множину допустимих альтернатив в залежності від ступеня невизначеності.

В основі процесу прийняття рішення лежить процедура вибору, що припускає наявність безлічі варіантів рішення. Вибір єдиного рішення повністю покладається на людину, його досвід і судження є вирішальними.

1.2. Загальні положення теорії дерева рішень

У теорії дерева рішень процес прийняття рішення розглядається як процес перетворення інформації, що включає етапи змістовної постановки завдання, формування переліку можливих дій (альтернативних рішень) та їх наслідків, вибір альтернативного рішення, аналіз результатів прийнятого рішення. Часто такий процес носить ітеративний характер, наприклад, в результаті отриманого рішення може бути скоригована вихідна множина альтернативних рішень.

У найзагальнішому вигляді задача прийняття рішення (ЗПР) може бути сформульована в термінах умов – цілей. Умови включають множину станів об'єкта та множину операторів, які переводять об'єкт з одного стану до іншого. Іншими словами, завдання умов представляється формальним описом засобів, результатів та способу їх зв'язку. Ціль визначає бажаний стан об'єкта. Різноманітність різних практичних завдань прийняття рішень обумовлює різноманіття деталізацій цієї загальної постановки.

Введемо до розгляду множини Ω і U , елементи множини Ω називатимемо альтернативними рішеннями або просто альтернативами, елементи множини U – результати. Тоді завдання прийняття рішення полягає у виборі альтернативи $x \in \Omega$, що призводить до деякого результату $u \in U$. Ефективність рішення визначається ступенем відповідності отриманого результату поставленої цілі.

Формалізація цієї постановки передбачає виконання наступних етапів: формалізацію цілі, побудова формального опису деякої універсальної множини альтернативних рішень Ω_u , (множини всіх можливих у різних реалізаціях даної ЗПР альтернативних рішень), аналіз цієї множини та виділення множини допустимих (з урахуванням обмежень конкретної реалізації ЗПР) альтернатив $\Omega \in \Omega_u$, побудова формального опису безлічі результатів U та взаємозв'язку Ω і U і, нарешті, побудова самої формальної процедури вибору $x \in \Omega$.

У різних моделях ТПР під завданням прийняття рішення часто розуміється лише сам одноразовий акт вибору x із заданого Ω . Таке обмеження проблеми може бути виправдано тим, що в деяких практичних задачах множини U і Ω фіксовані, але, незважаючи на це, проблема прийняття рішення залишається складною. З іншого боку, в різних складних задачах прийняття рішень задача вибору $x \in \Omega$ (або послідовність таких завдань) завжди має місце і є центральною. У даній роботі досліджені математичні моделі і методи вибору «найкращого» альтернативного рішення x з множини допустимих рішень Ω .

Процес вироблення альтернатив має першорядне значення, оскільки успішне його завершення гарантує, що всі «хороші» альтернативи братимуть участь у виборі. Тому автоматизація процесів прийняття рішень має

охоплювати всі етапи процесу прийняття рішення, а відповідні програмно-алгоритмічні засоби повинні включати процедури підтримки цих етапів.

У процесі дерева рішення беруть участь:

1. особа, яка приймає рішення (ОПР);
2. експерти;
3. консультанти.

Особою, яка приймає рішення, називають людину, яка має ціль, що служить мотивом постановки завдання і пошуку її вирішення; людини, що є компетентним фахівцем у своїй галузі, відповідно до уявлень якого про проблемну ситуацію здійснюється її формалізація; людини наділена необхідними повноваженнями, відповідно до яких він приймає остаточне або проміжне рішення і несе повну або часткову відповідальність за його наслідки.

При формалізації постановки ЗПР використовується одержана від ОПР інформація про його переваги [4]. Під уподобаннями ОПР будемо розуміти сукупність зазвичай спочатку не структурованих його уявлень, пов'язаних з перевагами і недоліками різних альтернативних рішень (тобто має місце деяка вихідна невизначеність переваг). Інформація про переваги використовується на різних етапах формалізації ЗПР і її рішення.

Експертом називають фахівця, здатного надавати інформацію, необхідну для формалізації ЗПР, але не несе відповідальність за наслідки прийнятого рішення. Отримана від експертів інформація часто використовується для опису Ω , U і взаємозв'язку Ω і U .

Консультантом (дослідником, системним аналітиком) називають фахівця з теорії прийняття рішень, який здійснює формалізацію ЗПР, розробляє процедуру прийняття рішень, організовує роботу експертів і ОПР [5, 10].

Іноді під ОПР розуміється не єдина людина, а група людей, що виконують зазначені функції в процесі прийняття рішень. За типом їх участі в процесі формування дерева рішення можуть бути виділені наступні класи задач:

1. Індивідуальне прийняття рішень. На різних етапах прийняття рішення бере участь єдина ОПР, вона же несе повну відповідальність за прийняте рішення.

2. Колективне прийняття рішень. У цьому випадку має місце багатоступінний вибір. На різних етапах здійснюється вибір, що звужує допустиму множину альтернатив Ω різними особами. Остаточний вибір здійснюється єдиною особою, відповідно до її цілі.

3. Групове прийняття рішень. Тут ставиться завдання вироблення узгодженого рішення на основі індивідуальних виборів багатьох осіб, які переслідують власні цілі. У цьому випадку немає «головної ОНР», а потрібно знайти «найбільш справедливе» рішення, що «найкращим чином» задовольняє цілям (часто суперечливим) різних ОНР.

При формалізації мети в ЗНР всіх трьох перерахованих типів доводиться мати справу з вихідною невизначеністю індивідуальних переваг. У ЗНР типу 1 необхідно формалізувати переваги єдиної ОНР, в задачах типу 2 і 3 - декількох ОНР. У ЗНР типу 3 має місце вихідна невизначеність узгодження індивідуальних переваг.

У цьому сенсі задачі типу 2 і 3 можуть бути розділені до задач типу 1.

Виникнення проблеми пов'язано з певними умовами. Ці умови в ТНР називають ситуаціями. Сукупність проблем і ситуацій утворюють проблемну ситуацію. В процесі прийняття рішення формуються альтернативи або варіанти, а також оцінюється їх перевагу.

Для цього:

- індивідуальна ОНР визначає критерій вибору,
- групова ОНР виробляє принцип узгодження групового вибору.

1.3. Відомі обмеження у використанні теорії дерева рішень

Прийняте рішення у теорії дерева рішень вважається допустимим, якщо воно задовольняє вихідним обмеженням. Допустиме рішення вважається оптимальним, якщо воно забезпечує екстремум критерію вибору або задовольняє принципу узгодження при груповому відборі.

Рішення на основі теорії дерева рішень вважається тим ефективніше, чим більше ступінь досягнення цілей і чим менше витрат на їх реалізацію

наступним чином. Для індивідуальної ОНР загальна постановка задачі формування дерева рішення записується у вигляді:

$$\text{ОНР: } = \langle S_0, T, Q | S, A, B, Y, f, K, Y^* \rangle$$

Зліва від вертикальної риски знаходяться вихідні дані задачі дерева рішення, справа - невідомі:

S_0 – проблемна ситуація,

T – час для прийняття рішення,

Q – ресурси;

S – множина альтернативних ситуацій, що визначають ситуацію до повної групи взаємовиключних, незалежних або гіпотетичних ситуацій, причому сума ймовірностей їх виникнення дорівнює 1;

A – множина цілей,

B - множина обмежень при прийнятті рішень;

Y – множина альтернативних варіантів рішень, з яких має бути вибрано єдине оптимальне рішення Y^* .

Вербальне формулювання задачі прийняття рішень індивідуальною ОНР: в умовах проблемної ситуації S_0 , наявного часу T і ресурсів Q необхідно визначити ситуацію множиною альтернативних ситуацій S , сформулювати множину цілей A , обмежень B , альтернативних рішень Y , провести оцінку переваг функції f і знайти оптимальне рішення Y^* з множини Y , керуючись критерієм K .

Для групової ОНР загальна постановка задачі прийняття рішення має вигляд, представлений на рис. 1.1, де $F(f) = F(f_1, f_2, \dots, f_d)$ - функція групової переваги, що залежить від індивідуальних переваг осіб, що входять в ГОНР; L - принцип узгодження індивідуальних переваг (наприклад, принцип більшості).

Вербальна формулювання завдання прийняття рішень груповою ОНР: в умовах проблемної ситуації S_0 , наявного часу T і ресурсів Q необхідно визначити ситуацію множини альтернативних ситуацій S , сформулювати безліч цілей A , обмежень B , альтернативних рішень Y , зробити оцінку функції групової переваги $F(f)$ і знайти оптимальне рішення Y^* з множини Y , керуючись принципом узгодження L .

Постановка задач формування дерева рішень, що визначає обмеження для індивідуального і групової ОПР дозволяє сформулювати ряд тверджень, які характеризують більшість технічних, управлінських і організаційних рішень:

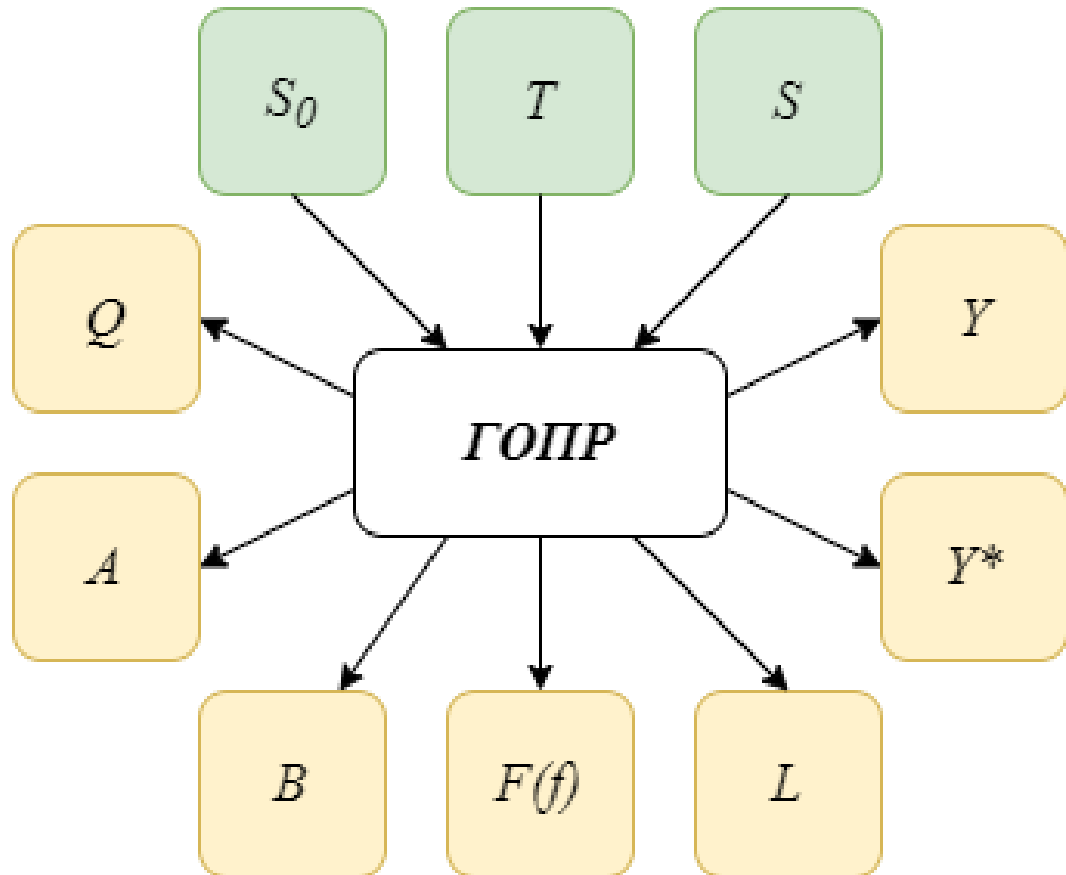


Рисунок 1.1 – Постановка задачі прийняття рішень для групової ОПР

1. Кількість невідомих елементів задач прийняття рішень може бути істотно більше, ніж відомих.

2. Експерти, що залучаються ОПР для прийняття рішень, виконують допоміжну роль, однак у реальності несуть відповідальність за свої рекомендації.

3. Оцінка рішення повинна проводитися на основі порівняння альтернативних варіантів рішень.

4. В умовах невизначеності може не існувати єдиного оптимального рішення.

5. Для ОПР з різними вподобаннями оптимальні рішення можуть виявитися різними.

Складність задач визначення обмеження у використанні прийняття рішень на основі дерева багато в чому визначається ступенем формалізації моделі проблемної ситуації.

1.4. Класифікація задач, які використовують теорію дерева рішень

В теорії дерева рішень існує введена Саймоном і Ньюеллом (1958) класифікація, що виділяє три групи: добре структуровані, неструктуровані та слабоструктуровані задачі [6,8].

1. Добре структуровані задачі, в яких найбільш істотні залежності між факторами середовища, альтернативами і їх наслідками визначені настільки добре, що допускають суворий кількісний опис. Тому такі задачі інакше називають формалізованими. Для їх вирішення можна побудувати математичну модель, досліджувати з її допомогою різні варіанти вибору і прийняти оптимальне рішення. До добре структурованих відносяться задачі математичного програмування, дослідження операцій, масового обслуговування.

2. Неструктуровані задачі, що мають тільки якісний опис. Інформація, необхідна для прийняття рішень, є неточною, неповною, не кількісною, а формальні моделі досліджуваної системи або занадто складні, або відсутні. Задачі цього класу називають не формалізованими. Для їх вирішення використовуються суб'єктивні судження, інтуїція, здогадки, припущення, що допомагають ОПР виявити свої переваги і визначити основні взаємозв'язки між елементами задач. Прикладами є проблеми вибору професії, місця роботи, кандидата на заміщення вакантної посади, перспективних проектів наукових досліджень і розробок, стратегії розвитку фірми і багато інших.

3. Слабоструктуровані, тобто змішані задачі вибору в умовах невизначеності при багатьох критеріях, які частково описуються деякою математичною моделлю, але за браком інформації про проблему і перевагах ОПР не мають однозначного алгоритмічного рішення, наприклад, задачі планування. У таких задачах недолік об'єктивної інформації принципово

неусувний на момент прийняття рішення та заповнюється суб'єктивними судженнями ОПР, що відображають його знання, інтуїцію і переваги.

На практиці реальні задачі прийняття рішень, як правило, є слабо структурованими, тобто такими, в постановці яких далеко не всі залежності виявлені чітко, є детермінованими, виражені кількісно або символічною, а процес прийняття рішень в умовах невизначеності не завжди заснований на раціональному підході і послідовному пошуку оптимального рішення.

Відображення множини рішень Y в множині критеріїв K може мати детермінований, імовірнісний або невизначений характер, відповідно до якого задачі і методи прийняття рішень можна розділити на детерміновані, в умовах ризику і задачі в умовах невизначеності. Якщо множина критеріїв K складається тільки з одного елемента, то говорять про задачі скалярної оптимізації, якщо множина K включає кілька критеріїв, то це задачі векторної оптимізації (багатокритеріальна задача прийняття рішень). У будь-якому випадку, при наявності ситуації вибору, багатокритеріальності і здійсненні вибору в умовах невизначеності або ризику задача прийняття рішень є не тривіальною.

Методи оцінювання та вибору рішень на основі класифікації задач поділяються на кількісні і якісні. Кількісні методи дуже різноманітні і добре опрацьовані, в тому числі в рамках таких наукових дисциплін, як методи оптимізації, дослідження операцій, системний аналіз:

До кількісних відносяться такі методи:

- класичні методи математичного програмування та обчислювальної математики (сфера їх застосування часто обмежена умовами диференціювання цільових функцій);

- методи перерахунку, засновані на переборному пошуку в дискретному просторі рішень, часто призводять до проблеми комбінаторного вибуху («прокляття розмірності»);

- методи, які використовують елемент випадковості (ймовірні, ігрові, еволюційні), які довели свою ефективність при вирішенні складних задач, але не гарантують отримання оптимального рішення.

Разом з кількісними методами в ТПР застосовуються методи, які дозволяють отримувати і аналізувати якісну інформацію, наприклад методи змістовного (вербального) аналізу ситуацій, багатокритеріального аналізу, експертного оцінювання і т.д. Застосування сучасних методів і технологій прийняття рішень – життєвою важливе завдання ОПР. У жорсткій конкурентній боротьбі, при інших рівних умовах, досягають успіху і стабільно розвиваються лише ті організації, які поставили собі на службу додаткові можливості, що надаються сучасними технологіями.

Використовуючи в якості класифікаційної ознаки тип залежності результатів від альтернатив, отримаємо наступні класи задач прийняття рішень:

1. Задачі прийняття рішень в умовах визначеності (детерміновані задачі) - коли кожна альтернатива приводить до єдиного результату. Тут є функціональна залежність результатів від альтернатив.

2. Задачі прийняття рішень в умовах стохастичності, коли кожна альтернатива може привести до одного з декількох випадків, кожен з яких має певну ймовірність появи. У цьому випадку має місце стохастична залежність результатів від альтернатив.

Оскільки в таких задачах вибір альтернативи однозначно визначає результат, то в подальших формальних побудовах рішень обмежуються розглядом множини Ω .

Можливий ефект від застосування методів ТПР на деякому автоматизується об'єкті залежить від двох основних чинників: ефекту від одноразового застосування математичних методів до деякої ЗПР і від регулярності рішення даної ЗПР на даному об'єкті.

Таким чином, по регулярності задач дерева рішення можуть бути умовно розділені на три класи [10]:

1. Унікальні задачі. Це, найчастіше, слабо структуровані задачі, що не мають добре опрацьованих аналогів. До таких задач, як правило, відносяться задачі розробки великих цільових програм. По регулярності це одноразово розв'язувані задачі на даному об'єкті.

2. Специфічні задачі. До даного класу будемо відносити більш часто зустрічаються, як правило, слабо структуровані задачі, що мають аналоги, але в кожній реалізації вимагають обліку проблемної ситуації. Даний клас задач дуже широкий, до нього ставляться такі задачі, як розробка виробничої програми підприємства, визначення основних проектних рішень по автоматизації підприємства, розподіл ресурсів і т.д. Частота рішення таких задач на одному об'єкті - кілька разів на рік (причому, проблемна ситуація кожен раз істотно видозмінюється).

3. Типові задачі. До цього класу належать як добре-так і слабо структуровані задачі, які мають добре опрацьовані аналоги і значний досвід їх вирішення на конкретному об'єкті в подібних ситуаціях. Це такі задачі, як прийняття рішення оператором, диспетчеризація, діагностування і т.д. Частота рішення - кілька разів на день-місяць на одному об'єкті (причому можливості змін проблемної ситуації незначні і добре вивчені).

Додаток теорії прийняття рішень для задач цих класів різниться. Якщо розроблена для деякої типової ЗПР модель адекватна і враховує можливість незначного зміни проблемної ситуації, то ефект автоматизації цієї задачі очевидний. Дійсно, додаток ТПР для типових задач привело до відчутного і швидкого успіху: розроблені автоматизовані і експертні системи в рамках яких ефективно вирішуються задачі вибору оптимального маршруту, розкрою матеріалу, діагностування і т.д.

Спроби використовувати ті ж підходи до вирішення специфічних задач очікуваного ефекту не принесли. Мінливість проблемної ситуації і слабка структурованість цих завдань унеможливають їх вирішення в рамках будь-якої жорсткої схеми із застосуванням одного будь-якого математичного методу, який навіть міг би бути ефективний для одноразового рішення подібної специфічної задачі. В даний час в ТПР для вирішення подібних задач використовується адаптивний підхід, який передбачає при вирішенні конкретної специфічної задачі коригування моделі, настройку математичних видів і уточнення самої схеми їх використання.

1.5. Системний аналіз процесу дерева рішень

При вирішенні унікальних задач на основі дерева рішень досить часто акценти зміщуються в бік системного аналізу. При цьому, ефективність рішення в основному визначається вдалою декомпозицією вихідної задачі на ряд задач, що мають аналоги в класах специфічних і типових задач.

За типом цілі задачі прийняття рішень на основі системного аналізу можуть бути розділені на задачі вибору і задачі оцінювання. Задачі оцінювання близькі за своєю суттю до задач класифікації. В цьому випадку задано розбиття множин альтернатив Ω на класи Ω_i ($\cup \Omega_i = \Omega$, $\cap \Omega_i = \emptyset$) і задана деяка альтернатива $x \in \Omega$. Потрібно визначити до якого класу Ω_i відноситься розглянута альтернатива x . До таких завдань, наприклад, відноситься задача діагностування.

У задачах вибору дерева рішень під час системного аналізу потрібно побудувати формальний опис деякої універсальної множини альтернативних рішень Ω_u , формалізувати ціль, виділити на основі побудованого опису множин допустимих альтернативних рішень $\Omega \subseteq \Omega_u$ і здійснити з Ω вибір кращого рішення (кількох кращих рішень).

Надалі будемо розглядати задачі дерева вибору на основі системного аналізу. Відзначимо, що при прийнятті практичного вирішення ці задачі можуть комбінуватися, наприклад, при визначенні способу лікування спочатку вирішується задачі оцінювання (постановка діагнозу), а потім здійснюється вибір кращого варіанта лікування з можливих.

З точки зору системного аналізу, за типом представлення результатів рішення задачі вибору можуть бути поділені на задачі вибору з Ω єдиного кращого рішення x , вибору заданого числа рішень $\mathcal{K} \subseteq \Omega$, вибору і впорядкування по перевазі заданого числа рішень (зокрема, впорядкування всієї множини Ω). Найбільш часто на практиці зустрічаються задачі вибору єдиного кращого рішення.

Оскільки розглядаються тільки детерміновані задачі індивідуального прийняття рішень, то для подальшої формалізації загальної постановки ЗПР

потрібно із застосуванням деяких мовних засобів побудувати формальний опис універсальної множини альтернатив Ω_u , виділити множину доступних альтернатив $\Omega \subseteq \Omega_u$, формалізувати ціль і здійснити вибір з Ω кращого альтернативного рішення.

В даний час теорія прийняття рішень на основі системного аналізу являє собою синтез таких історично самостійно розвинутих дисциплін як математичне програмування, теорія ігор, теорія автоматичного регулювання, математична економіка і т.д. (рис.1.2)



Рисунок 1.2 - Міждисциплінарні взаємозв'язки теорії дерева рішень на основі системного аналізу

Внесок теорії дерева рішення в освітню програму складається в професійному освоєнні методів прийняття ефективних технічних рішень, розумінні сучасних інженерних проблеми в цій галузі. Цьому відповідають цілі ТДР як наукового напрямку: допомогти людині в пошуку кращих варіантів рішень з урахуванням багатьох критеріїв, випадкових і невизначених факторів, загострити інтуїцію особи, яка приймає рішення.

Звідси і використання різних мов для опису ситуації і механізмів вибору в теорії прийняття рішень [9,11]. Тут під мовою розуміється усталена у

відповідній дисципліні сукупність понять і термінів. Причому має місце ситуація коли, з одного боку, в різних типах задач застосування різних мов має різну ефективність. Різні мови по-різному чутливі до окремих аспектів ЗПР.

1.6. Формалізація цілі у методах дерева рішень

У деяких практичних задачах прийняття рішення формалізація цілі може бути природно зведена до побудови деякої функції, заданої на множині альтернатив (результатів) і приймаючої дійсні значення, а рішення задачі - до екстремізації даної функції. Ця функція носить назву цільової функції. У цьому випадку використовується мова математичного програмування. Така формалізація, як правило, виникає коли при ухваленні рішення один будь-який аспект прийнятого рішення є для ОПР визначальним. Для широкого класу задач характерною рисою є їх багатоаспектний характер. Якщо інтенсивність властивості деякого аспекту може бути виражена кількісно, то така властивість називають критерієм якості, і формалізація цілі в цьому випадку може бути здійснена на мові критеріїв якості. У деяких випадках вдається побудувати скалярну функцію, визначену на множині значень критеріїв якості таку, що альтернатива, якої відповідає максимальне значення цієї функції має для ОПР найбільшу цінність (функцію корисності або функцію цінності), тоді має місце мова теорії корисності. Відзначимо, що в цьому випадку при вирішенні задач ми послідовно будемо мати справу з мовою критеріїв (на першому етапі формалізації цілі), з аксіоматичною мовою теорії корисності (при обґрунтуванні існування і побудові функції корисності) і з мовою математичного програмування (при екстремізації функції корисності).

Не завжди формалізація цілі може бути зведена до побудови деякої функції цінності. У таких випадках ефективно використовується мова бінарних відносин. Його використання в ТПР ґрунтується на понятті відносини переваги, формалізує індивідуальні переваги ОПР. Ідея застосування відносини переваги полягає в декомпозиції проблеми вибору альтернативного рішення з Ω на вирішення задач парного порівняння альтернатив, визначення принципу відбору кращої альтернативи за результатами парних порівнянь і реалізації

цього принципу. Однак, застосування мови бінарних відносин ефективно не для всіх ЗПР, а тільки для задач, які мають наступну властивість: перевагу між двома альтернативами $x, y \in \Omega$ повинні бути незалежними від співвідношень цих альтернатив з іншими альтернативами $\Omega \setminus \{x, y\}$.

У разі, коли це властивість не виконується (має місце ситуація контекстного вибору) може бути використана мова функцій вибору. В теорії прийняття рішень використані мови можуть бути розбиті на два класи - концептуальні мови вибору (відповідають на питання «що вибирати?») і мови механізмів вибору (відповідають на питання «як вибирати?»). Так, при використанні функції корисності, як концептуальної мови використовується аксіоматична мова теорії корисності (що включає елементи мови критеріїв), а в якості мови механізму вибору - мова математичного програмування. Такий поділ дещо умовний, наприклад, мова бінарних відносин в одних підходах використовується як концептуальна, в інших - як мова механізму вибору, а в-третє - як зв'язка між концептуальною мовою і мовою механізму вибору. Тим часом, виділення концептуальної мови і мови механізму вибору методологічно представляється доцільним, оскільки останнім часом численні дослідження в теорії прийняття рішень спрямовані на розробку єдиної мови, а їх результати дозволяють надіятись, що в доступному для огляду майбутньому можна очікувати побудови єдиної концептуальної мови прийняття рішень. Що ж стосується мови механізму вибору, то, ймовірно, в конкретній ЗПР все ж буде ефективніше використовувати той зі сформованої, або розробленої нової мови, яка найбільш повно враховує характерні особливості задачі.

Вибір є дією, що надає всій діяльності цілеспрямованість. Саме вибір реалізує підпорядкованість всієї діяльності певної мети або сукупності цілей. Рано чи пізно настає момент, коли подальші дії можуть бути різними, що приводять до різних результатів, а реалізувати можна тільки одну дію, причому повернутися до ситуації, що мала місце в той момент, вже, як правило, не можна.

Повна формалізація знаходження найкращого рішення можлива, але лише для добре вивчених, структурованих завдань; для вирішення слабо

структурованих задач повністю формальних алгоритмів не існує (якщо не брати до уваги тривіального і далеко не завжди прийняттого алгоритму перебору, тобто методу проб і помилок), але досвідчені і здатні фахівці часто роблять вибір, що виявляється кращим.

Технології вибору дерева рішень під час формування цілі можуть бути досить різноманітними. Для цієї мети повинні залучатися високопрофесійні фахівці, здатні відібрати варіанти рішень, гідні подальшої, більш глибокого опрацювання і порівняльної оцінки [12]. При відборі основних варіантів рішень необхідно враховувати:

- їх досить високу порівняльну оцінку;
- відсутність дублювання, щоб спектр варіантів рішень, відібраних для більш глибокого опрацювання, був досить повним і в той же час ненадлишковим;
- специфічні особливості ситуації, встановлені в процесі її діагностики.

Якщо відібрані основні варіанти рішень потребують адекватної порівняльної оцінки і в більш глибоке опрацювання під час формування цілі у методах дерева рішень, то повинна бути здійснена постановка задач. Тому, необхідно сформувати оціночну систему для постановки задачі, що включає:

- основні чинники (приватні критерії), що впливають на розвиток ситуації;
- оцінку їх порівняльної важливості;
- шкали для визначення значень факторів при порівняльній оцінці основних варіантів.

Експертиза варіантів дерев рішення для формування цілі повинна, з одного боку, дати оцінку можливості бути реалізованим рішенням і досягнення з їх допомогою поставлених цілей, а з іншого - дозволити проранжувати їх на основі застосування оціночної системи (наприклад, рівень очікуваного досягнення цілі, необхідні витрати ресурсів і часу, наслідки).

Сучасна тенденція практики вибору адекватної формування цілі у методах дерева рішень полягає в поєднанні здатності людини вирішувати неформалізовані задачі з можливостями формальних методів і комп'ютерного моделювання [13,18].

Ухвалення рішення слід представляти як дію над множиною альтернатив, в результаті якого виходить підмножина обраних.

Звуження множини альтернатив можливо, якщо є спосіб порівняння альтернатив між собою і визначення найбільш бажаних. Кожен такий спосіб називається критерієм переваги (відбору).

Проблема вибору адекватних формувань цілі у методах дерева рішень далеко не проста і допускає істотно різні як строгі математичні постановки задач, так і неформальні процедури. Це пов'язано з тим, що кожна компонента ситуації вибору може реалізовуватися в якісно різних варіантах, основними з яких є:

1) множина альтернатив може бути кінцевим, рахунковим або континуальним;

2) оцінка альтернативи може здійснюватися по одному або за кількома критеріями, які, в свою чергу, можуть мати як кількісний, так і якісний характер;

3) режим вибору може бути одноразовим (разовим) або повторюваним, що допускають навчання на досвіді;

4) наслідки вибору можуть бути точно відомі (вибір в умовах визначеності), мати імовірнісний характер, коли відомі ймовірності можливих результатів після зробленого вибору (вибір в умовах ризику), або мати неоднозначний результат, який не допускає введення ймовірностей (вибір в умовах невизначеності);

5) відповідальність за вибір може бути односторонньою (в окремому випадку індивідуальної) або багатосторонньою. Відповідно розрізняють індивідуальний і груповий вибір;

6) ступінь узгодженості цілей при багатосторонньому виборі може варіюватися від повного збігу інтересів сторін (кооперативний вибір) до їх протилежності (вибір в конфліктній ситуації). Можливі також проміжні випадки, наприклад компромісний вибір, коаліційний вибір, вибір в умовах наростаючого конфлікту і т.д.

Таким чином, різні поєднання перерахованих варіантів формування цілі у методах дерева рішень призводять до різноманітних завдань вибору, які вивчені не в однаковій мірі та потребують розгляду у подальшому

1.7. Використання формальних методів дерева рішень

Для отримання кращого рішення слід використовувати формальні методи дерев рішень. До теперішнього часу склалися три основних формальних напрямлення опису вибору формальних методів дерева рішень [14]:

- критеріальна мова опису вибору методів дерева рішень;
- опис вибору на мові бінарних відносин у методах дерева рішень;
- мова функцій вибору у методах дерева рішень.

Найпростішою, найбільш розвиненою і, може бути, тому частіше вживаною на практиці є критеріальна мова опису вибору методів дерева рішень. Ця назва пов'язана з основним припущенням про те, що кожному окремо взяту альтернативу можна оцінити конкретним числом (значенням критерію) і порівняння альтернатив зводиться до порівняння відповідних їм чисел.

Друга, на якому описується вибір - це мова бінарних відносин. Його велика, ніж у критеріальної мови, спільність заснована на врахуванні того факту, що в реальності дати оцінку окремо взятої альтернативи часто важко або неможливо. Однак якщо розглядати її не в окремо, а в парі з іншою альтернативою, то знаходяться підстави сказати, яка з них краща.

Деякі особливості вибору привели до побудови третьої, ще більш спільної мови його опису - на основі функцій вибору. По-перше, нерідко доводиться стикатися з ситуаціями, коли перевагу між двома альтернативами залежить від інших альтернатив. Наприклад, перевагу покупця при виборі між чайником і кавоваркою може залежати від наявності в продажу кавомолки.

По-друге, можливі такі ситуації вибору, коли поняття перевагу взагалі позбавлене сенсу. Наприклад, по відношенню до множини альтернатив досить звичайними є правила «найбільш відмінного, оригінального», втрачають сенс в разі двох альтернатив.

Для вироблення ефективного рішення необхідні:

- інформація;
- професійні знання по суті проблеми;
- знання формальних правил вироблення рішення.

Перші два моменти не викликають сумнівів. Третій вимагає пояснень. Людина, приймаючи щодня безліч рішень, робить це автоматично, не замислюючись про їх послідовності. Проте процес прийняття рішень при уявній простоті дуже непростий. У ньому досить багато тонкощів і підводних рифів, добре знайомих професійним менеджерам [15].

Є помилкою думку про те, що процеси прийняття рішення на основі формальних методів дерева рішень так само різноманітні, як різноманітна природа проблем, які вимагають їх прийняття. Тому вважають, що процес вирішення проблеми за допомогою формальних методів дерева рішень, наприклад підготовки кадрів, здійснюється інакше, ніж вирішення інших різноманітних економічних і політичних проблем, що будуються на інших методах рішення задач.

Таким чином, у кожній організації здійснюється розробка рішень на основі дерева рішень з використанням формальних методів дерева рішень. І в кожній організації практика їх розробки і прийняття має свої особливості, які визначаються характером і специфікою її діяльності, її організаційною структурою, діючою системою комунікацій, внутрішньою культурою. Однак є загальне для будь-якого процесу, що будується на основі дерева прийняття рішень, де б він не здійснювався і незалежно від характеру проблеми. Тому, це той єдиний стрижень, який формує технологію розробки формальних методів дерева рішень, що використовується в будь-якій організації яка займається фінансуванням перспективних досліджень.

РОЗДІЛ 2

МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕОРІЇ ДЕРЕВА РІШЕНЬ

2.1. Визначення основних етапів типового процесу прийняття рішення

Будь-який процес прийняття рішення на основі теорії прийняття рішень передбачає наявність наступних елементів:

1) Особа, що приймає рішення (ОПР), тобто того, кому належить його прийняти і вирішити проблему; це може бути індивідуум, великий чи малий колектив людей;

2) керованих змінних ситуації, якими може керувати ОПР. Такі змінні можуть бути як кількісними, так і якісними;

3) некерованих змінних ситуації, якими не може управляти ОПР, але які спільно з керованими змінними можуть впливати на результат вибору. Ці змінні також можуть бути кількісними або якісними. У сукупності з керованими змінними вони утворюють зовнішнє середовище (фон) ситуації (проблеми). Слід мати на увазі, що некеровані змінні зовсім не обов'язково мають властивість абсолютної некерованості - вони можуть регулюватися іншими особами (організаціями). В ієрархічній організації кожен рівень управляє тими змінними, які не можуть контролюватися більш низькими рівнями;

4) внутрішніх і зовнішніх обмежень на можливі значення керованих і некерованих змінних;

5) не менше двох можливих ліній поведінки і можливих результатів вибору; в іншому випадку проблеми вибору не виникає, так як його немає.

Дослідження процесу формування і прийняття рішень шляхом поділу його на елементарні операції показало, що він являє собою ряд послідовних етапів або дій, в результаті здійснення яких виробляється ефективне рішення [16, 20].

Весь процес вироблення рішень відбувається за допомогою управлінських операцій, які можна об'єднати в певні цільові комплекси. Комплекси і складові

їх операції виконуються в строгій послідовності і можуть бути представлені у вигляді схеми (рис. 2.1).

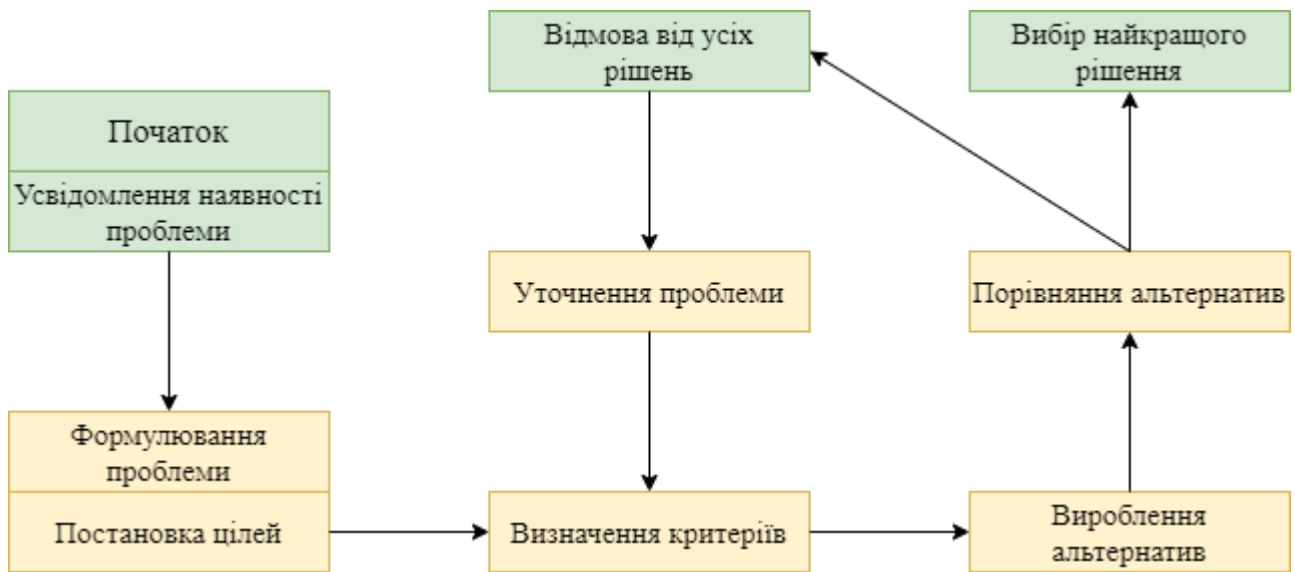


Рисунок 2.1 - Етапи типового процесу прийняття рішення

Етап 0. Аналіз ситуації, що виникла і виявлення проблеми. Будь-який процес прийняття рішення повинен починатися з аналізу необхідності прийняття даного рішення. Вихідний момент дій - збір і накопичення первинної інформації про об'єкт управління і про ситуації, що склалася. Всебічно аналізується причина виникнення даної ситуації, підрозділи та особи, які мають до неї відношення, що застосовуються в аналогічних випадках заходи. Аналогом ситуації може виявитися досвід інших організацій, працівників різних підрозділів своєї організації. Можуть бути використані повідомлення в спеціалізованих виданнях про діяльність інших організацій, патентні фонди, наукові розробки і т.д. Всі попередні і подальші рішення утворюють деякий ланцюг рішень, і важливо знайти місце вирішення даної проблеми в цьому ланцюзі.

Етап 1. Формулювання проблеми, постановка цілей. Після виявлення конкретної ситуації і визначення потреби в прийнятті управлінського рішення необхідно поставити діагноз проблеми, бажано повний і правильний. Про проблему можна дізнатись по тому, що не сталося те, що мало статися. Вирішуючи проблему, що з'явилася таким чином, згладжують відхилення від

норми. Як проблеми можуть виступати ситуації, в яких сталося те, що не повинно було статися. Як проблему можна розглядати і потенційну можливість.

Корисно пошукати аналогічні проблеми і переконатися, що розглянута проблема - це дійсно проблема, а не симптом чого-небудь більш серйозного.

Визначити проблему в повній мірі часто важко, оскільки всі частини організації взаємопов'язані, а у великій організації можуть бути сотні таких взаємозалежностей, тому правильно визначити проблему - значить наполовину вирішити її. Діагностування складної проблеми починається з встановлення симптомів ускладнень або усвідомлення наявних можливостей.

Для виявлення причин виникнення проблеми необхідно зібрати і проаналізувати внутрішню і зовнішню (щодо організації) інформацію [17]. Інформацію можна зібрати формальними методами (аналіз звітів, ринку і т.д.) і неформально, розмовляючи про ситуації, що склалася і роблячи особисті спостереження. При цьому, необхідно мати на увазі, що:

- 1) збільшення кількості інформації не обов'язково підвищує якість рішення;
- 2) не стосується справи є шум (перешкоди) для керівника;
- 3) існує ризик свідомого спотворення інформації в чийсь інтересах.

Рішення зазвичай пов'язане з необхідністю зробити вибір, і усвідомлення цього задає ОПР напрямок в пошуку альтернатив по досягненню кінцевих результатів і виключає альтернативи, що лежать за межами поставленої цілі.

Ціль рішення може бути однозначною, встановленою при виникненні проблеми. Вона може бути задана керуючою організацією. Найчастіше мета конкретного рішення зумовлюється будь-якої глобальної мети і допускає кілька варіантів дій.

При формуванні цілі рішення необхідно враховувати цілі керівної системи, об'єктивні можливості організації, ресурсні обмеження. Визначення цілей, їх декомпозиція і формування підцілей дозволяють уточнити спочатку сформульовану проблему. Правильність постановки цілей рішення підтверджується його зв'язком з ситуацією, яку треба розглянути, а також з попередніми рішеннями.

Етап. 2. Визначення критеріїв. Про рішення судять, перш за все, за отриманими результатами. З розгляду результатів слід почати процес вибору. Критерії визначають, які чинники слід враховувати, роблячи вибір. Набір критеріїв використовується як основа для порівняння варіантів рішення, однак вони мають різне значення. Наприклад, одні критерії є обов'язкові обмеження, а інші просто фіксують бажані характеристики рішення. Щоб прийняти ефективне рішення, слід розділити критерії на жорсткі обмеження й бажані характеристики, без яких можна було б і обійтися.

Критерії, віднесені до категорії бажаних, можливо проаранжувати. Розташувати бажані критерії в порядку пріоритетності, можна, коли настане момент вибору, здійснивши його на користь більш важливих критеріїв.

Етап 3. Вироблення альтернатив. Комплекс дій з підготовки варіантів рішення включає, перш за все, розробку моделі їх аналізу. При цьому з'ясовується, чи немає готових моделей, що застосовувалися раніше в подібних або аналогічних ситуаціях.

Модель - це відображення досліджуваного процесу в спрощеному вигляді. Залежно від характеру проблеми модель може бути простою (елементарною) або складною. Прості моделі в залежності від ступеня формалізації можуть бути частково або повністю програмованими.

На даному етапі необхідно забезпечити формування всіх можливих рішень, що дозволяє в подальшому вибрати оптимальний варіант. Для повноти охоплення можливих рішень розробку варіантів доцільно вести за принципом декомпозиції або використовувати будь-якої іншої метод, що дозволяє гарантувати повноту розглянутих варіантів. У разі стандартних рішень цей крок для керівника не представляє труднощі, так як ці рішення відрізняються обмеженим набором альтернатив.

Інформація, що характеризує альтернативи, знаходиться в безпосередній залежності від критеріїв рішення і являє собою набір значень критеріїв для даної альтернативи.

Етап 4. Порівняння альтернатив. Кваліфіковане прийняття рішень вимагає вибору найкращої альтернативи.

Іноді всі варіанти рішень виглядають прийнятними і жоден з них не здається краще за інших. Для того щоб зробити вибір, керівник потребує певних засобів для порівняння альтернатив. Аналіз альтернатив здійснюється на основі обраної моделі.

Найважливішою передумовою прийняття кращого рішення є перебір варіантів. У формалізованих моделях вибір оптимального варіанта досягається шляхом спрямованого перебору з застосуванням спеціальних методів: математичного програмування, статистичного моделювання, теорії масового обслуговування, теорії ігор і т.д.

У ряді простих випадків, коли можливе застосування моделі оптимізації рішення, цілком ймовірно єдине, найкраще рішення. Якщо при цьому досягнутий результат відповідає цілі, рішення може бути прийнято. Якщо ціль не досягається, необхідно від неї відмовитися, так як поліпшити результат неможливо. При цьому в більшості випадків можлива зміна спочатку певних умов або моделі рішення задачі для досягнення поставленої цілі. Тому якщо розглянуті варіанти не забезпечують досягнення цілі, необхідно перевизначити проблему або переглянути модель рішення. Процес наближення (ітерації) до наміченої цілі може повторюватися до тих пір, поки ціль не буде досягнута або ж випробування не доведуть неможливість її досягнення.

Етап 5. Вибір кращого рішення. У загальній кількості варіантів рішення виявиться якесь число таких, які за своїми формальними результатами будуть найбільш близькі до поставленої цілі. Коли ці варіанти відібрані, здійснюється комплекс дій по прийняттю конкретного рішення.

Обговорення відібраних варіантів має очолюватися керівником, що володіє правом прийняття і / або затвердження рішення. Число учасників обговорення і відводиться для цього час залежать від важливості рішення і термінів його дії. У процесі оперативного управління керівники часто одноосібно виробляють, оцінюють ступінь прийнятності та приймають рішення.

Після обговорення обраного варіанту відповідним колом компетентних представників може виникнути необхідність узгодження його з верхнім рівнем управління або з різними зовнішніми організаціями державного, відомчого або

громадського характеру. Погоджений варіант затверджується і вважається прийнятим.

2.2. Моделі прийняття рішень в умовах нелінійних розподілених задачах

Уявіть, що ви керівник групи розробників софту, якого не влаштовують показники роботи групи. Необхідно здійснити заходи по збільшенню обсягу випуску програмної продукції, підвищенню його якості, зниженню собівартості, забезпечення ритмічності і т.д. Якщо домагатися всіх цих цілей відразу, то задача може виявитися некоректною, оскільки досягнення цілей залежить від множини різних факторів (кількості виконавців, рівня їх кваліфікації тощо) [19]. Якщо вам відомі форми залежностей цілей від факторів, якими можна управляти, то виникає задача пошуку поєднання чинників, що забезпечує оптимум для поставленої мети.

Задача пошуку екстремуму (максимуму або мінімуму) деякої функції при наявності обмежень на значення її змінних називається спільним завданням математичного програмування.

За допомогою комп'ютера можна розбити відрізок $[A, B]$ на 100 частин, обчислити значення функції $F(x)$ в утворених інтервалах і знайти серед них максимальне. Напрошується висновок про те, що рішення цієї задачі в разі однієї змінної не представляє особливих труднощів.

Якщо задача максимізації $F(x, y)$ дві змінні з умовами $A < x < B$, $C < y < D$, то спроба скористатися апаратом класичної математики буде успішною лише в разі, коли є впевненість, що максимум досягається всередині прямокутника, що визначає множину допустимих точок (граничних точок тут безліч). Якщо використовувати чисельну рішення, то при тій же точності доведеться розбити інтервали по x і y на 100 частин і обчислювати значення функції в 100×100 точках. У разі n змінних аналогічний підхід до вирішення потребують обсяг обчислень близько 100^n (при $n > 10$ труднощі будуть навіть у суперкомп'ютера).

Так, для вирішення задач лінійного програмування (завдань з лінійної цільової функцією і лінійними обмеженнями) існує універсальний симплекс-метод, який має безліч модифікацій і дає рішення задачі невеликій розмірності в прийнятний час навіть вручну.

Для вирішення цілочислових лінійних задач існує кілька методів (метод Гоморі, метод гілок і меж), які, тим не менш, не гарантують отримання оптимального цілочисельного плану в прийнятні терміни.

Для вирішення задач опуклого програмування (мінімізації нелінійної опуклою функції або максимізації увігнутою функції на опуклій множині допустимих рішень) розроблені різні методи, що базуються на теоремі Куна - Таккера і понятті градієнта функції.

Детерміновані задачі скалярної оптимізації відносяться до класу добре структурованих задач прийняття рішень, про які є достатня і достовірна кількісна інформація. Це означає, що задача має бути добре формалізована (є адекватна математична модель реального об'єкта); існує єдина цільова функція (критерій оптимізації), що дозволяє судити про якість розглянутих альтернативних варіантів; є можливість кількісної оцінки значень цільової функції; задача має певні ступені свободи (ресурси оптимізації), тобто деякі параметри функціонування системи можна довільно змінювати в деяких межах з метою поліпшення значень цільової функції.

Детерміновані розподільчі задачі - один з найбільш поширених класів задач прийняття рішень - мають свої методи вирішення. Зокрема, в 1957 р опублікована монографія Р. Беллмана, що поклала початок одному з оригінальних методів дослідження багатокрокових процесів прийняття рішень - методом динамічного програмування. Великий внесок у розвиток методів оптимізації подібних задач внесла група радянських математиків на чолі з Л.С. Понтрягиним.

У самих різних областях діяльності доводиться мати справу з необхідністю поетапного прийняття рішень для досягнення певної кінцевої мети. Елементарними прикладами такого роду процесів можуть служити будь-які ігри з метою досягнення максимального можливого виграшу, управління

космічним кораблем шляхом поетапної коректури режимів з метою підтримки заданого віддалення від Землі, всі види господарської діяльності тощо

Якщо для однокрокових процесів прийняті рішення, як правило, відносно прості, то в багатокрокових процесах структура рішення незрівнянно складніше. Для ілюстрації математичних проблем, що виникають при дослідженні багатокрокових процесів прийняття рішень, розглянемо наступну ідеалізовану постановку нелінійної розподільної задачі:

Нехай необхідно виконати певний обсяг робіт V . В розпорядженні є N видів обладнання. Задана (аналітично або таблично) нелінійна залежність експлуатаційних витрат кожного виду обладнання від обсягу виконуваних робіт. Потрібно так розподілити обсяги робіт по N видам обладнання, щоб сумарні витрати були мінімальними.

Якщо залежність експлуатаційних витрат від обсягів робіт задана аналітично, то для вирішення задачі можна, наприклад, застосувати метод невизначених множників Лагранжа. Якщо ж залежність задана таблично, то застосовують, як правило, метод динамічного програмування. Він заснований на принципі Беллмана:

Оптимальні за Беллманом стратегії прийняття рішень мають наступну властивість: які б не були початкові стани і початкові рішення, всі наступні рішення повинні прийматися виходячи з оптимальної стратегії на даному етапі, з урахуванням стану, що впливає з першого рішення.

Позначимо залежність експлуатаційних витрат обладнання i -го виду від обсягу робіт x_i , що виконується обладнанням i -го виду, через $Y_i(x_i)$. У даній розподільній задачі в якості обмеження прийmemo $\sum_{i=1}^n x_i = V, x_i \geq 0$. Необхідно мінімізувати функцію $Y = \sum Y_i(x_i) \rightarrow \min$.

Позначимо через $f_N(x)$ величину мінімальних витрат при оптимальнє розподіле обсягу робіт x по N видам обладнання. Якщо $x = 0$, то $f_N(0) = 0$, де $N=1,2, \dots$ (при нульовому обсязі робіт обладнання не працює, тому витрати дорівнюють 0).

Нехай $N=1$. Тоді при будь-якому обсязі робіт x справедливо:

$$f_N(x) = Y_1(x)$$

Використовуючи принцип Беллмана, можна знайти рекурентне співвідношення між $f_N(x) \sim f_{N-1}(x)$, що дозволить по індукції обчислювати кожне наступне значення $f_i(x)$, $i = 2, \dots, N$, по відомому початкового значення $f_1(x)$.

Дійсно, якщо початковий обсяг робіт для N -го виду обладнання дорівнює x_N , то витрати складуть величину $Y_N(x_N)$, а мінімальні витрати при оптимальному розподілі залишкового обсягу робіт $(x - x_N)$ для $(N-1)$ -го виду обладнання, складуть $f_{N-1}(x - x_N)$. Оптимальним рішенням буде такий вибір x_N , при якому витрати складуть величину

$$f_N(x) = \min[Y_N(x_N) + f_{N-1}(x - x_N)], 0 \leq x_N \leq x \leq V \quad (1)$$

причому $f_1(x) = Y_1(x)$.

Це співвідношення дозволяє спростити обчислювальну складність задач, зводячи N -мірну багатокрокову задачу прийняття рішень до рішення N -одновимірних задач оптимізації згідно із зазначеним рекурентному співвідношенню (1). Таким чином, метод відображає покроковий процес прийняття рішень.

2.3. Моделі алгоритм прямого і зворотного прогону для задач упорядкування

Існують дві обчислювальні схеми моделі: алгоритм прямого і зворотного прогону. Відповідно до алгоритму прямої прогонки розрахунки ведуться в природному порядку $f_1 \rightarrow f_2 \rightarrow \dots \rightarrow f_N$, на відміну від зворотного прогону, де обчислення починаються з останнього етапу. Процедура прямої прогонки видається більш ясною і логічною, однак практика показує, що зворотній прогін в ряді випадків більш ефективний (задачі управління запасами, планування і т.д., де хронологія подій в зворотному порядку зручніша).

З наведених вище зауважень і рішень можна зробити деякі висновки щодо методу динамічного програмування, що базується на принципі оптимальності Беллмана [21].

1. Метод динамічного програмування дозволяє звести N -мірну задачу оптимізації до сукупності задач меншої розмірності (очевидно, що легше вирішити 10 одновимірних задач оптимізації, ніж одну 10-мірну).

2. Метод орієнтований на рішення не конкретної задачі, а цілого класу подібних задач.

3. Поява додаткових обмежень часом полегшує вирішення задач за рахунок зменшення обсягу перебору варіантів.

4. Головним недоліком методу є, кажучи словами Беллмана, «прокляття розмірності» - складність методу істотно зростає зі збільшенням розмірності задачі.

З задачами впорядкування при пошуку найбільш привабливого рішення в умовах різних обмежень ми стикаємося постійно. Це важкі комбінаторні задачі, пов'язані з побудовою розкладів, визначенням оптимальної послідовності обробки виробів, масивів інформації, вибором найкращих маршрутів руху і т.д. У задачах упорядкування від порядку виконання робіт залежить розподіл ресурсів по кроках.

Методологічну основу для вирішення задач упорядкування послідовності робіт надає теорія розкладів. Теорія розкладів - один з розділів дослідження операцій, в якому вивчаються математичні постановки та методи розв'язання задач календарного планування і оперативного управління, упорядкування в часі фіксованої системи ресурсів для виконання певної сукупності робіт. В теорії розкладів основна увага приділяється питанням оптимального розподілу та впорядкування кінцевої множини вимог, що обслуговуються детермінованими системами (планування виробництва, виконання потоку обчислювальних задач, складання розкладу навчальних занять і т.д.)[4, 21]. При цьому враховуються структура і тимчасові параметри технологічного процесу.

Тимчасова ув'язка всієї множини дій, пов'язаних з досягненням заданої цілі, вже сама по собі складна задача. Якщо ж необхідно знайти оптимальне

впорядкування або побудувати найкраще розклад, та ще в найкоротший термін, то складність задачі незмірно зростає. Розглянемо ідеалізовану постановку задачі впорядкування:

Є m виробів, кожне з яких спочатку має бути оброблено на першій машині, а потім на другій. Відомо час обробка j -го виробу ($j = 1, 2, \dots, m$) на першій машині $t_{1,j}$ і на другій машині $t_{2,j}$. Знайти порядок з мінімальним сумарним часом обробки всіх виробів на двох машинах.

Перелічимо основні обмеження в даній задачі:

- перша машина працює без простоїв;
- час переходу виробу від однієї машини до іншої незначне, і ним можна знехтувати;
- не можна починати обробку чергового виробу, не завершивши обробку попереднього.

Позначимо через t_{nj} час простою другої машини, якщо вона вже вільна, а обробка чергового виробу на першій машині ще не завершена. Тоді цільова функція, яку необхідно мінімізувати, має вигляд

$$T = \sum t_{nj} + \sum t_{2j} \rightarrow \min.$$

Загальна кількість варіантів вирішення цієї задачі дорівнює $m!$. Задачу можна вирішувати методом динамічного програмування. Однак для двох машин відомий ефективний алгоритм Джонсона [22]:

1. Пошук мінімального елемента серед t_{1j} і t_{2j} .
2. Перестановка виробів. Якщо мінімальний елемент відноситься до першої машини, то відповідно j -е виріб поставити на перше місце. якщо мінімальний елемент відноситься до другої машини, то на останній місце. Якщо ж кілька однакових мінімальних елементів відносяться до 1-ї (2-й) машині, то на перше (останнє) місце ставлять виріб, якому відповідав би більший елемент, що відноситься до 2-ї (1-й) машині.

3. Викреслюємо переставлені вироби і переходимо до п.1 до тих пір, поки не буде вичерпано список виробів. Внаслідок отримаємо оптимальну послідовність обробки виробів на двох машинах.

Алгоритм Джонсона можна застосувати для трьох машин лише при дотриманні одного з наступних двох нерівностей:

1) $\min \{t_{1j}\} \geq \max \{t_{2j}\}$ (якщо при заданому технологічному порядку обробки виробів дана умова виконується, то підсумовуються t_{1j} і t_{2j} , і задача зводиться до двох машин);

2) $\min \{t_{3j}\} \geq \max \{t_{2j}\}$ (якщо при заданому технологічному порядку обробки виробів дана умова виконується, то підсумовуються t_{2j} і t_{3j} , і задача зводиться до двох машин).

Промислові системи і програмні продукти, що реалізують рішення реальних, а не ідеалізованих задач упорядкування, складання розкладів і планування, використовують різноманітні методи евристичного програмування. Ці методи дозволяють знайти оптимізовані, а не оптимальні рішення. До сучасних методів пошуку оптимізованих рішень, які не гарантує знаходження глобального оптимуму, відносяться еволюційні алгоритми, алгоритми ройового інтелекту, моделювання відпалу та ін.

2.4. Використання методу еволюційних алгоритмів

У еволюційних алгоритмах застосовуються поняття, що традиційно відносяться до біології, такі, як популяція, кросингвер, мутація і природний відбір, для створення кращих рішень задачі. У евристиці цих алгоритмів використовується гіпотеза про аналогію між природним відбором і процесом вибору найкращого рішення з множини можливих. Наприклад, в генетичних алгоритмах носіями рішень є особини, в «хромосомах» яких закодовані ті чи інші суттєві параметри. Моделюючи відбір кращих особин як процес еволюції в популяції особин, можна отримати оптимізоване рішення задачі, задавши початкові умови еволюційного процесу, сформувавши популяцію і вказавши

цілі еволюційного процесу. Ключове питання тут - можливість налаштувати оцінки пристосованості особин (fitness function) або цільові функції.

Загальна схема роботи ЕА включає наступні кроки:

1. Задається спосіб кодування параметрів задачі в «хромосомах».
2. Створюється вихідна популяція (початкове рішення).
3. Розраховується функція пристосованості кожної особини.
4. Для найбільш пристосованих особин проводиться схрещування (кросинговер) і народження нових особин, що містять ознаки обох батьків. При цьому передача ознак здійснюється відповідно до одним з обраних способів «обміну ділянками хромосом». Найменш пристосовані особини «відмирають» і замінюються новонародженими. Випадково здійснюється мутація окремих особин.
5. Повторюйте кроки 2 - 4 ітераційно задану кількість разів.
6. З отриманої популяції вибираються кращі особини, «хромосоми» декодуються, а отримане рішення відповідає локального оптимуму.

Переваги еволюційних алгоритмів в простоті реалізації, щодо високої швидкості роботи, паралельному знаходити власні шляхи розв'язання, недоліки - в деякій складності вибору схеми кодування і опису обмежень в задачі, можливе виродження популяції.

У евристичних алгоритмах ройового інтелекту багатовимірний простір пошуку населяється роєм частинок (рішень) [24]. Крім координат кожна частка має швидкість переміщення і прискоренням. Визначаються координати частинки з найкращим поточним оптимумом, до якого на наступному кроці спрямовуються інші частинки. Елемент випадковості моделюється деяким числом «божевільних» частинок, закон руху яких відрізняється від закону руху інших.

Загальна схема роботи алгоритму рою включає наступні кроки:

1. Створюється вихідна «випадкова» популяція часток.
2. Розраховується цільова функція для кожної частки.
3. Краща частка з точки зору цільової функції оголошується «центром тяжіння». Вектори швидкостей всіх частинок, за винятком «божевільних»,

спрямовуються до цього центру. Чим далі частка знаходиться від центру, тим більшим прискоренням вона володіє.

4. Розраховуються нові координати частинок в просторі рішень.

5. Повторюйте кроки 2 - 4 ітераційно повторюються задану кількість разів.

6. Останній «центр ваги» відповідає знайденому локальному оптимуму.

Мурахи, бджоли, птахи вміють виконувати роботу спільними зусиллями. На відміну від людей, вони обходяться без керівного центру. Інтернет теж децентралізований, і, можливо, рішення його мережевих проблем варто шукати саме у громадських комах.

Ця схема була успішно застосована для завдання маршрутизації інформації в мережі, а також для розробки більш ефективних схем кешування даних в Інтернеті, для оптимізації цифрових фотографій. Алгоритм ройового інтелекту може давати результати, навіть кращі ніж при використанні еволюційних алгоритмів і нейронних мереж. Рій не знає, де саме знаходиться їжа, але на кожній ітерації рій знає, як далеко вона знаходиться. Ефективною стратегією буде слідування за особою, яка на даний момент знаходиться на їжі найближче.

Для більшості задач упорядкування об'єм популяції в 10 часток достатній, щоб отримати хороші результати. Типовий розмір популяції - $20 \div 40$ частинок. Для деяких спеціальних задач упорядкування розмір популяції може досягати від 100 до 200 часток. Число ітерацій t_{max} вибирається в такий спосіб. Якщо, наприклад, цілочисельна змінна x змінюється на інтервалі $[-10, 10]$, то $t_{max} = 20$. Коефіцієнти k_1 і k_2 зазвичай вибираються рівними 2, або варіюються на інтервалі $[0, 4]$.

Метод відпалу був запропонований на початку 80-х років для вирішення комбінаторних завдань оптимізації (Кіркпатрік, 1983). Як природного аналога був узятий процес кристалізації емалевої субстанції. В ході її охолодження ступінь вільного руху молекул поступово стає обмеженим, досягаючи точки рівноваги з мінімальним енергетичним рівнем у кристалічній структурі. Важливою особливістю даного процесу є швидкість охолодження. Позначимо через S множину станів системи, через T температуру системи при термічній рівновазі. Тоді $p_{T(a)}$ є ймовірністю того, що система при температурі T

знаходиться в стані a , незалежно від енергетичного рівня E а цього стану. згідно з розподілом Больцмана [20]

$$p_T(a) = \frac{1}{\sum_{b \in S} \exp\left(\frac{-E_b}{k \cdot T}\right)} \cdot \exp\left(\frac{-E_b}{k \cdot T}\right),$$

де k є константою Больцмана. Припустимо, що система знаходиться в момент часу t в стані a з енергетичним рівнем E_a . Шляхом невеликої випадкової зміни в момент часу $t + 1$ система переходить в стан b з енергетичним рівнем E_b . Якщо різниця $\Delta E = E_b - E_a \leq 0$, то актуальним стає стан b , якщо $\Delta E > 0$, то стан b запам'ятовується з ймовірністю $\exp\left(\frac{E_a - E_b}{k \cdot T}\right)$. При великих t дане правило забезпечує перехід системи в стан рівноваги шляхом генерації послідовності рішень задачі оптимізації. аналогія з фізичним процесом тут полягає в наступному:

- рішення оптимізаційної задачі відповідають станам системи в процесі охолодження фізичної субстанції;
- цільова функція F відповідає енергетичному рівню субстанції;
- процедура пошуку оптимального рішення аналогічна пошуку стану системи з мінімальним енергетичним рівнем;
- температура T є параметром для управління процедурою оптимізації.

Значний вплив на результативність методу відпалу надає «план охолодження», який встановлює динаміку зміни параметра T_t і кількість проведених ітерацій I_t . Один з варіантів плану полягає в установці високого початкового значення T_0 , для того щоб спочатку практично кожне рішення запам'ятовувалося з певною ймовірністю. Завдяки цьому на початковому етапі процесу оптимізації можна уникнути передчасного скорочення простору пошуку. Надалі зазвичай застосовується пропорційне зменшення параметра T : $T_{t+1} = \alpha T_t$, де α - деяка константа (як правило, $0,8 \leq \alpha \leq 0,99$). Таким чином, забезпечується асимптотичне наближення до значення $T_{min} = 0$.

Відзначимо, що даний план є далеко не єдиним. На практиці метод відпалу

демонструє толерантність до неоптимальних рішень: ймовірність «виживання» рішення тим вище, чим вище значення T і чим менше різниця між порівнюваними рішеннями (це допомагає уникнути попадання в локальний екстремум цільової функції). Навпаки, на завершальній фазі оптимізаційного процесу, чим менше значення T , тим менше простір пошуку, а при $T=0$ «виживають» тільки кращі рішення. Повільне «охолодження», на відміну від швидкого, призводить до кращих результатів, тобто вибирати необхідно між швидкістю збіжності та результативністю методу.

Процес, що моделюється за методом відпалу, є марковським, і при певних припущеннях щодо застосовуваного «плану охолодження» можна довести його збіжність до глобального оптимуму з ростом часу його пошуку. Тому при обмеженні на час оптимізації метод розглядається як евристичний, що допускає в залежності від застосовуваного «плану охолодження», що глобальний оптимум не буде знайдений.

Іншою важливою особливістю при реалізації методу відпалу є встановлення відносини сусідства і механізму, за допомогою якого здійснюється перехід від одного рішення до іншого. Визначення відносини сусідства залежить від розв'язуваної задачі, а вибір нової точки серед сусідніх в більшості випадків робиться випадково.

2.5. Визначення показників ефективності евристичних алгоритмів

Показниками ефективності евристичних алгоритмів є швидкість і стійкість пошуку. Швидкість пошуку визначається часом роботи алгоритму і часом досягнення рішенням заданої якості. Стійкість означає пошук до потрапляння в точки локальних екстремумів.

Особливості цільових функцій, такі як багатоекстремальність і ізольованість в найбільшій мірі впливають на швидкість роботи розглянутих евристичних алгоритмів розв'язання задачі впорядкування [26, 30].

Багатоекстремальне створює множину помилкових аттракторів - асимптотично стійких точок, які не є глобальним оптимумом в просторі пошуку, до яких прагне траєкторія пошуку (найчастіше, якщо траєкторія

пошуку підходить досить близько до аттрактору, то з часом вона вже не залишає його околиці і підходить до нього все ближче і ближче). Спостерігається ефект тяжіння до аттрактору, який перевіряється на складних функціях з відомим оптимумом. Ці функції зазвичай використовуються для порівняльного аналізу алгоритмів по часу і якості пошуку.

При кількості змінних x , що дорівнює 10, дана функція має 1010 - 1 локальних і одним глобальним екстремумів.

Іншою особливістю цільових функцій, що впливає на швидкість роботи розглянутих евристичних алгоритмів, є ізольованість («пошук голки в копиці сіна»).

Це означає, що функція не надає ніякої інформації, «підказує», в якій області шукати екстремум. Лише випадкове потрапляння особини в глобальний екстремум може вирішити задачу.

В цілому порівняння конкуруючих евристичних алгоритмів для вирішення задач скалярної оптимізації з точки зору їх результативності показує наступне. Нехай A_1 і A_2 - два конкуруючих евристичних алгоритму, F - множина цільових функцій (задач) оптимізації, H - гістограма їх значень F , m - кількість отриманих рішень, $p(H/F, m, A)$ - ймовірність отримання за допомогою A -алгоритму m різних значень F , що мають вигляд гістограми H .

Вольперт і Макріді сформулювали і довели наступну NFL-теорему (No-Free-Lunch): для будь-якої пари евристичних алгоритмів пошукової оптимізації (A_1 і A_2) справедливо наступне співвідношення:

$$\sum_F p(H|F, m, A_1) = \sum_F p(H|F, m, A_2).$$

Це означає, що всі евристичні алгоритми прийняття рішень для задач скалярної оптимізації в середньому однаково результативні при їх порівнянні з усім F . Якщо за алгоритмом A_1 виходять результати кращі, ніж за алгоритмом A_2 , то існують оптимізаційні задачі з іншого цільовою функцією, для яких кращі результати дає алгоритм A_2 . При цьому час виконання задач конкуруючими алгоритмами не враховується.

Таким чином, до евристичних прийомів прийняття рішень відносяться всі алгоритми з аксіоматично недоведеними особливостями завдання, тобто використовують емпірично побудовану модель прийняття рішень в задачі, замість точної моделі.

РОЗДІЛ 3

МОДЕЛІ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕОРІЇ ІГОР

3.1. Формалізація моделі гри

Добро і зло, світло і темрява перебувають у вічному протиборстві. У ТПР протиборство характеризується нанесенням конфліктуючими сторонами взаємного збитку і прагненням здобути перемогу над суперником. Конфліктна ситуація – це ситуація прихованого або відкритого протиборства двох або кількох учасників, кожен з яких має свої цілі і мотиви, засоби і способи вирішення значущої проблеми.

Формальною моделлю протиборства є гра.

Гра - модель конфліктної ситуації, що включає чіткі правила дій гравців, для досягнення виграшу в результаті прийняття деякої стратегії.

Теорія ігор (ТІГ) - це прикладна міждисциплінарна наука, що вивчає математичні моделі прийняття рішень в конфліктних ситуаціях [5, 26].

Відповідно до визначення теорія ігор займається моделями прийняття рішень, а не їх поведінковими, психологічними аспектами або питаннями реалізації рішень. Якщо інтереси учасників протилежні, то ці моделі називаються антагоністичними іграми; якщо інтереси не збігаються, але не протилежні, то мова йде про ігри з непротилежними інтересами. В іграх двох осіб йдеться про протиборство тільки двох гравців. Якщо в грі беруть участь n осіб, то вони можуть вступати між собою в постійні або тимчасові коаліції, а гра називається коаліційною.

Повна класифікація різновидів ігор поки не розроблена. Їх перелік також включає біматричні, диференціальні, матричні, статистичні гри з природою та інше.

Суть гри в тому, що кожен з учасників з безлічі альтернатив обирає таку стратегію дій, яка, можливо, забезпечить йому найбільший виграш або найменший програш. Причому гравцеві ясно, що результат залежить не тільки від нього, а й від дій противника. Це означає, що він приймає рішення в умовах невизначеності.

Метою теорії ігор є вироблення рекомендацій для розумної поведінки гравців в конфліктних ситуаціях, тобто вироблення оптимальних стратегій для кожного гравця. Одне із завдань теорії ігор полягає в тому, щоб з'ясувати, чи можливо (якщо так, то за яких умов) певну рівновагу (компроміс, сідлова точка), найбільшою мірою влаштовує гравців.

Основною особливістю теорії ігор є розширення поняття оптимальності, включаючи в нього компроміс, що влаштовує гравців. Це використовується, наприклад, в економіці при виборі оптимальних рішень для підвищення якості продукції або визначення запасів. «Конфлікти» тут полягають:

- 1) у прагненні випустити більше продукції, витративши на неї менше праці, і зробити продукцію краще;
- 2) в бажанні так заpastися ресурсами, щоб застрахуватися від випадковостей і не заморожувати кошти.

Подібні задачі вирішуються і іншими способами. Це не випадково. Наприклад, багато задач теорії ігор можуть бути зведені до задач лінійного програмування, і навпаки.

Іншою особливістю ігрових моделей є пошук стійких рішень, коли відхід від оптимальної стратегії не вигідний обом гравцям. Доведено, що при багаторазовому повторенні гри і різних в кожному розігравші стратегіях, сідлова точка і стійкі рішення існують. Однак гравцям треба вибирати стратегію за жеребом, інакше противник, виявивши закономірності в рішеннях, може вгадати хід і виграти.

Розглянемо спочатку гри з ненульовою сумою. Якщо для кінцевої безкоаліційної гри двох осіб ставити у відповідність стратегіям 1-го гравця рядки деякої таблиці, стратегіям 2-го гравця - її стовпці, а клітини таблиці заповнювати значеннями виграшу 1-го гравця, то отримана таблиця називається матрицею виграшу 1-го гравця. Якщо клітини тієї ж таблиці заповнити значеннями функції виграшу 2-го гравця, то вийде матриця виграшів 2-го гравця.

Ця пара матриць повністю описує біматричну гру. Якщо біматрична гра є антагоністичною, то вона повністю описується єдиною матрицею виграшів

одного з гравців і називається матричною грою з нульовою сумою. У ній виграш одного гравця означає програш іншого, а їх сума дорівнює нулю.

Біматрична гра не обов'язково є антагоністичною, тобто інтереси гравців не повністю протилежні (є можливість повідомляти один одному про свої наміри, координувати свої дії, а також застосовувати блеф, загрози і інші способи обміну інформацією). Виграш одного гравця не обов'язково означає програш іншого, а сума виграшів - не обов'язково дорівнює нулю. Це ігри з ненульовою сумою.

3.2. Використання основної теореми антагоністичних ігор двох осіб

Нехай дана кінцева антагоністична гра двох осіб з нульовою сумою. При матричному способі завдання вона задається трійкою множин

$$G = (X, Y, M),$$

де $X = \{x_1, x_2, \dots, x_m\}$ - множина стратегій 1-го гравця; $Y = \{y_1, y_2, \dots, y_n\}$ - множина стратегій 2-го гравця; $M = M(x, y)$ - платіжна функція (обмежена числова функція одного з гравців, визначена на декартовій множині стратегій $X \times Y$ і рівна математичному очікуванню виграшу / програшу гравця).

Оскільки антагоністична гра двох осіб з нульовою сумою повністю описується єдиною матрицею виграшів одного з гравців, то зазвичай функцію M задають у вигляді платіжної матриці виграшів 1-го гравця:

$$Q = \|q_{ij}\|_{m \times n}, \text{ де } q_{ij} = M(x_i, y_j).$$

У ситуації протистояння цілі гравців вважаємо протилежними: 1-й гравець прагне максимізувати свій виграш, а 2-й гравець - мінімізувати свій програш (або навпаки). Труднощі в цих задачах полягають в тому, що жоден з гравців не контролює повністю платіжну функцію $M(x, y)$. Сутність рекомендацій теорії ігор полягає в подоланні цієї проблеми.

Сказане справедливо лише для ігор двох осіб, так як якщо гравців більше двох, то виникає можливість утворення коаліцій з договірними розподілом виграшу.

В антагоністичних іграх двох осіб з нульовою сумою єдиним критерієм вибору оптимальної стратегії є наступний критерій:

Оптимальною стратегією гравця є така стратегія, використання якої при багаторазовому повторенні гри забезпечує гравцеві максимально можливий середній виграш (або мінімально можливий середній програш).

Для пошуку оптимальної стратегії в теорії ігор використовується «песимістичний» критерій, званий критерієм максиміна / мінімакса і заснований на виборі найкращого з найгірших можливостей. Теорія виходить з припущення про те, що обидва гравці однаково сильні і не прощають помилок. Передбачається також, що оптимальне рішення досягнуто, якщо жодному з гравців не вигідно змінювати свою стратегію (точка рівноваги або сідлова точка), тобто досягнення компромісу в реальному конфлікті сторін, що мають протилежні інтереси, є вигідним для кожного з супротивників.

У сідлової точки чудова властивість: вона одночасно є мінімальним елементом рядка і максимальним елементом в стовпці платіжної матриці.

Якщо порівняти максимін і мінімакс в будь-якій антагоністичній грі двох осіб з нульовою сумою, то справедлива теорема, з якої випливають практично корисні властивості оптимальних стратегій.

У антагоністичній грі двох осіб з нульовою сумою максимін завжди не більш мінімакса:

$$\alpha \leq \beta \text{ або } \max_x \min_y M(x, y) \leq \min_y \max_x M(x, y).$$

Результат гри, який має сідлову точку, є визначеним. Він не залежить від глибини аналізу гравців, а залежить тільки від умов гри, які вичерпуються задачею функції виграшу $M(x, y)$. Тому ігри, що мають сідлові точки, називають цілком певними, чи іграми з повною інформацією (кожен гравець знає всі попередні ходи, як свої, так і противника) [28].

На практиці більшість антагоністичних ігор двох осіб з нульовою сумою не мають сідлової точки, тобто немає рішення в чистих стратегіях.

Чиста стратегія - це окремий випадок змішаної стратегії. Дійсно, якщо в змішаній стратегії i -я чиста стратегія застосовується з імовірністю 1, то всі інші чисті стратегії не застосовуються. Для дотримання секретності кожен гравець застосовує свої стратегії незалежно від вибору іншого гравця (лотерея).

Можливість знаходження кожним гравцем своєї оптимальної стратегії базується на основній теоремі теорії ігор. Вона є доказом існування рішення для кожної кінцевої антагоністичної гри двох осіб з нульовою сумою. У 1927 р Е.Борель сформулював основну теорему теорії ігор, а в 1928 р Дж.фон Нейман довів цю теорему. Наведемо формулювання теореми (її доказ засноване на теоремі Брауера про нерухому точку і в силу цього є неконструктивним, тобто не дає способу знаходження рішення гри).

Для основної теореми антагоністичних ігор двох осіб, де використовується метод послідовних наближень належать наступні припущення. Будь-яка кінцева антагоністична гра двох осіб з нульовою сумою має ціну v^* і у кожного гравця є, щонайменше, одна оптимальна стратегія (ξ^*, η^*) .

Далі, в процесі збору та аналізу даних, побудови моделі у вигляді матриці виграшів проводяться вимірювання. Будь-яке вимірювання має певну похибку, помилки можуть накопичуватися. До того ж пошук оптимального рішення гри при великій кількості стратегій гравців є досить трудомісткий процес. Тому точність у визначенні значення гри і оптимальних стратегій гравців виправдана не завжди.

Похибка в оцінці гравцем свого виграшу не може привести до практично серйозних наслідків і невелике відхилення гравця від оптимальної стратегії не тягне за собою істотної зміни в його виграші. Досить знайти наближене рішення, яке дає середній виграш, близький до ціни гри, і наближені оптимальні стратегії гравців. Орієнтовне значення ціни гри може дати вже простий аналіз матриці виграшів і визначення нижньої і верхньої ціни гри. Якщо вони близькі, то пошуками точного рішення займатися не обов'язково, так як досить вибрати відповідні чисті максиміна і мінімаксної стратегії гравців. Якщо ж вони не близькі, можна отримати прийнятне для практики рішення за допомогою чисельних методів розв'язання ігор.

Одним з найпростіших послідовних наближених методів рішення матричної гри є метод Брауна-Робінсон (процес пошуку наближеного рішення було сформульовано в роботі Г. Брауна, а збіжність процесу була доведена Дж. Робінсон) [31]. Ідея цього методу полягає в наступному. Імітується багаторазове повторення гри і набирається статистика, що показує, які стратегії максимізують виграш, і таким чином приблизно визначається оптимальна стратегія.

Для аналізу антагоністичної гри з деякою платіжною матрицею будується ітераційний процес, кожен крок якого – розіграш гри. У першому розіграші у гравців ще немає ніякої інформації, тому вони вибирають свої стратегії довільно. Далі кожен гравець вибирає таку чисту стратегію, яка є найкращою з урахуванням всіх попередніх ходів противника, що розглядаються як деяка «змішана» стратегія, тобто послідовно, при кожному розіграші вибирається та стратегія, яка першому гравцеві дає максимальний середній виграш, а другого – мінімальний середній виграш. Після кожного розіграшу обчислюється середнє значення виграшу першого гравця, програшу другого гравця, і їх середнє арифметичне приймається за наближене значення ціни гри. Після завершення ітераційного процесу обчислюються частоти використання гравцями своїх стратегій, які є наближеними значеннями ймовірностей в оптимальних змішаних стратегіях гравців.

Наприклад, нехай дана наступна матриця виграшів 1-го гравця:

Таблиця 3.1 – Матриця виграшів 1-го гравця

	Y_1	Y_2	Y_3
X_1	2	1	0
X_2	2	0	3
X_3	-1	3	-3

У грі немає сідлової точки ($\alpha = 0$, $\beta = 2$). Результати обчислень за методом послідовних наближень зручно представити в табличному вигляді:

Таблиця 3.2 – Результати обчислень за методом послідовних наближень

N	i(N)	L ₁ (N)	L ₂ (N)	L ₃ (N)	v _j (N)	j(N)	U ₁ (N)	U ₁ (N)	U ₁ (N)	v̄(N)	v
1	x ₁	2	1	0	0	y ₃	0	3	-3	3	1,5
2	x ₂	4	1	3	0,5	y ₂	1	3	0	1,5	1
3	x ₂	6	1	6	0,3	y ₂	2	3	3	1	0,7
4	x ₂	8	1	9	0,25	y ₂	3	3	6	1,5	0,9
5	x ₃	7	4	6	0,8	y ₂	4	3	9	1,8	1,3
6	x ₃	6	7	3	0,5	y ₃	4	6	6	1	0,75
7	x ₂	8	7	6	0,9	y ₃	4	9	3	1,3	1,1
8	x ₂	10	7	9	0,9	y ₂	5	9	6	1,1	1
9	x ₂	12	7	12	0,8	y ₂	6	9	9	1	0,9
10	x ₂	14	7	15	0,7	y ₂	7	9	12	1,2	0,95
11	x ₃	13	10	12	0,9	y ₂	8	9	15	1,4	1,1
12	x ₃	12	13	9	0,75	y ₃	8	12	12	1	0,9

У таблиці 2 прийняті наступні позначення стовпців:

N – номер розіграшу (партії);

i(N) - номер чистої стратегії 1-го гравця, обраної в партії N;

L_j(N) - загальний виграш 1-го гравця після N партій, якщо 2-й гравець весь час застосовує стратегію у_j;

$v^-(N) = \min_j (L_j(N))/N$ - найменший середній виграш 1-го гравця за N ходів;

j(N) - номер чистої стратегії 2-го гравця в партії N;

U_i(N) - загальний виграш 1-го гравця після N партій, якщо він весь час використовує стратегію x_i;

$v^+(N) = \max_i (U_i(N))/N$ - найбільший середній виграш 1-го гравця за N ходів;

$v = (v^+ + v^-)/2$ – наближене значення ціни гри.

У таблиці наведено розіграші для 12 партій. В результаті ціна гри $v \approx 0,9$, наближене значення оптимальної стратегії 1-го гравця $\xi \approx (1/12,$

7/12, 4/12), 2-го гравця – $\eta \approx (0,8 / 12,4 / 12)$. Наведемо для порівняння точне рішення: $v^*=1, \xi^*=\eta^* = (0, 8/12, 4/12)$.

Даний метод легко автоматизувати (в електронних таблицях, в якому-небудь середовищі програмування), тому він іноді краще відомості матричної гри до задачі лінійного програмування і використання симплекс-методу. Для зупинки ітераційного процесу комп'ютерного моделювання гри необхідно вказати число ітерацій. Додатковою умовою зупинки є досягнення стану «рівноваги», при якому всі виграші $L_j(N)$ або $U_i(N)$ збігаються між собою.

Доведено, що процес сходиться при числі партій $\mathcal{N} \rightarrow \infty$, причому наближене значення v прагне до ціни гри v^* , а ξ і η можуть «коливатися» біля значення оптимальних стратегій ξ^* і η^* . Ітераційний процес веде гравців до мети повільно, збіжність тут не монотонна. При цьому швидкість збіжності зменшується з ростом розмірності матриці і зростанням числа стратегій гравців. Її порядок пропорційний $N-1 / (m + n-1)$. Однак поряд з таким недоліком можна виділити і гідності методу: складність і обсяг обчислень порівняно слабо зростають у міру збільшення числа стратегій гравців (m і n). До того ж трудомісткість методу знижується, якщо в ході розіграшів визначати не тільки номер черговий чистої стратегії, але і скільки разів поспіль вона повинна застосовуватися.

3.3. Використання методу лінійного програмування

В процесі збору та аналізу даних, побудови моделі у вигляді матриці виграшів проводяться вимірювання.

Розглянуті методи вирішення ігор (геометричний і послідовних наближень) не гарантують отримання оптимального рішення. Методом, що дозволяє знайти точне рішення гри, є метод відомості ігрової задачі до задачі лінійного програмування [33].

У розмові з Д. Данцигом (автор симплекс-методу в його матричної формі) Дж. Фон Нейман виклав взаємозв'язок теорії подвійності в лінійному

програмуванні з основною теоремою матричних ігор. Суть цього зв'язку полягає в наступному:

- будь-яка кінцева гра двох осіб може бути зведена до пари двоїстих задач лінійного програмування;
- задачі лінійного програмування може бути представлена як гра двох осіб з нульовою сумою.

Це дозволяє після зведення ігрової задачі до пари двоїстих задач лінійного програмування застосувати будь-який відомий метод (наприклад, симплекс-метод) і отримати оптимальне рішення матричної гри. Відзначимо, що зведенням задач лінійного програмування до матричної гри користуються рідше.

Проаналізуємо процес зведення матричної гри до пари двоїстих задач лінійного програмування. Нехай задана гра матрицею Q розміром $(m \times n)$:

Таблиця 3.3 – Матриця Q розміром $(m \times n)$

	Y_1	Y_2	...	Y_n
X_1	Q_{11}	Q_{12}	...	Q_{1n}
X_2	Q_{21}	Q_{22}	...	Q_{2n}
...
X_m	Q_{m1}	Q_{m2}	...	Q_{mn}

Згідно з теоремою (якщо один з гравців дотримується своєї оптимальної змішаної стратегії, то його виграш залишається незмінним і рівним ціні гри незалежно від того, яку стратегію застосовує інший гравець, якщо тільки той не виходить за межі своїх активних стратегій), оптимальна стратегія 1 го гравця ξ^* гарантує йому виграш не менше v^* , незалежно від стратегії, обраної 2-м гравцем:

$$\begin{aligned} q_{11}\xi_1 + q_{21}\xi_2 + \dots + q_{m1}\xi_m &\geq v, \\ q_{12}\xi_1 + q_{22}\xi_2 + \dots + q_{m2}\xi_m &\geq v, \\ &\dots \dots \dots \end{aligned}$$

$$q_{1n}\xi_1 + q_{2n}\xi_2 + \dots + q_{mn}\xi_m \geq v,$$

$$\text{де } \xi_1 + \xi_2 + \dots + \xi_m = 1; \quad \xi_i \geq 0, \quad i = 1, 2, \dots, m. \quad (2)$$

Будемо вважати, що $v > 0$. Якщо це не так, то на підставі теореми (оптимальні змішані стратегії (ζ^*, η^*) в матричній грі $\|q_{ij}\|_{m \times n}$ з ціною v будуть оптимальними і в матричній грі $\|aq_{ij}+b\|_{m \times n}$ з ціною $v^*=av+b$, де $a > 0$) елементи матриці q_{ij} завжди можна зробити позитивними, додавши деяку константу (це вплине на ціну гри, але не на оптимальні стратегії). Ціна гри v^* з позитивними елементами матриці не може бути негативною.

Перетворимо систему (2), розділивши її на позитивне число v . Введемо нове позначення: $p_i = \xi_i/v$, $i=1,2,\dots,m$. Отримаємо

$$\begin{aligned} q_{11}p_1 + q_{21}p_2 + \dots + q_{m1}p_m &\geq 1, \\ q_{12}p_1 + q_{22}p_2 + \dots + q_{m2}p_m &\geq 1, \\ &\dots \dots \dots \end{aligned}$$

$$q_{1n}p_1 + q_{2n}p_2 + \dots + q_{mn}p_m \geq 1,$$

$$\text{де } p_1 + p_2 + \dots + p_m = \frac{1}{v}; \quad p_i \geq 0, \quad i = 1, 2, \dots, m. \quad (3)$$

Оскільки 1-й гравець прагне максимізувати ціну гри v , то в (3) величина, зворотна $1/v$, буде мінімізуватися.

Звідси оптимальна стратегія 1-го гравця визначається з прямої задачі лінійного програмування: знайти мінімум функції $P_{min} = p_1 + p_2 + \dots + p_m = \frac{1}{v}$ при обмеженнях (3).

Аналогічно розмірковуючи, оптимальна стратегія 2-го гравця гарантує йому програти не більш ціни гри, незалежно від вибору стратегії x_i 1-го гравця, тобто справедлива наступна система нерівностей:

$$\begin{aligned} q_{11}\eta_1 + q_{12}\eta_2 + \dots + q_{1n}\eta_n &\leq v, \\ q_{21}\eta_1 + q_{22}\eta_2 + \dots + q_{2n}\eta_n &\leq v, \\ &\dots \dots \dots \end{aligned}$$

$$q_{m1}\eta_1 + q_{m2}\eta_2 + \dots + q_{mn}\eta_n \leq v,$$

$$\text{де } \eta_1 + \eta_2 + \dots + \eta_n = 1; \quad \eta_j \geq 0, \quad j = 1, 2, \dots, n. \quad (4)$$

Перетворимо систему (4), розділивши її на позитивне число v . Введемо нову змінну: $z_j = \eta_j / v$, $j = 1, 2, \dots, n$. отримаємо

$$\begin{aligned} q_{11}z_1 + q_{12}z_2 + \dots + q_{1n}z_n &\leq 1, \\ q_{21}z_1 + q_{22}z_2 + \dots + q_{2n}z_n &\leq 1, \\ &\dots \dots \dots \dots \dots \dots \\ q_{m1}z_1 + q_{m2}z_2 + \dots + q_{mn}z_n &\leq 1, \end{aligned}$$

де $z_1 + z_2 + \dots + z_n = \frac{1}{v}$, $z_j \geq 0$, $j = 1, 2, \dots, n$. (5)

Оскільки 2-й гравець прагне мінімізувати свій програш, то в (5) величина зворотна $1/v$ буде прагнути до максимуму. Звідси оптимальна стратегія 2-го гравця визначається з оберненої задачі лінійного програмування: знайти максимум функції $Z_{max} = z_1 + z_2 + \dots + z_n = \frac{1}{v}$ при зазначених обмеженнях.

Таким чином, рішення матричної гри зведено до вирішення пари двоїстих задач лінійного програмування. З теорії лінійного програмування відомо, що задача лінійного програмування не завжди має рішення. З іншого боку, згідно з основною теоремою теорії матричних ігор будь-яка кінцева гра двох осіб з нульовою сумою завжди має рішення [35]. Отже, необхідно довести, що в отриманій парі двоїстих задач завжди існує допустиме рішення, а цільова функція є обмеженою.

Припустимо, що в отриманій прямій задачі лінійного програмування немає допустимого рішення. Щоб це спростувати, потрібно вказати на існування хоча б однієї точки в просторі пошуку рішень, в якій всі умови-обмеження виконуються.

Знайдемо цю точку. Оскільки все $q_{ij} > 0$, то позначимо через $\mu = \min_i \min_j q_{ij}$. Тоді точка з координатами $p_1 = 1/\mu$, $p_2 = p_3 = \dots = p_m = 0$ є допустимим рішенням, так як вона задовольняє умовам (3), що легко перевірити шляхом підстановки цього рішення в умови-обмеження прямої задачі.

Цільова функція P для прямої задачі мінімізується, тому вона повинна бути обмеженою знизу. Справді, все $p_i \geq 0$, а коефіцієнти при змінних p_i в цільовій функції (3) є позитивними, отже, цільова функція обмежена знизу нулем.

Таким чином, в отриманій парі двоїстих задач завжди існує допустиме рішення, а цільова функція є обмеженою.

Після зведення ігрової задачі до пари двоїстих задач лінійного програмування і її рішення, наприклад, симплекс-методом, необхідно повернутися до вихідної ігрової задачі, виконавши необхідні заміни змінних, визначити ціну гри та оптимальні стратегії гравців.

3.4. Використання моделі багатокритеріальних задач

Критерій (від грец. κριτήριον - мірило) - це кількісна оцінка цілі, величина, на підставі якої порівнюються і вибираються кращі рішення. Ціль зазвичай вимірюється в номінальній шкалі, а її критерій – сильнішою шкалою, що дає можливість зіставляти альтернативні рішення.

На відміну від раніше розглянутих задач скалярної оптимізації в задачах багатокритеріального вибору завжди існує невизначеність, пов'язана з зіставленням оцінок за різними критеріями. Ця невизначеність є принциповою і не може бути виключена на основі використання моделей і об'єктивних розрахунків. ОПР є єдиним джерелом інформації, що дозволяє оцінити варіанти рішень і вибрати з них найкращий [36].

На перший погляд здається, що в способі вирішення проблеми відбулися порівняно невеликі зміни – для усунення невизначеності, яка виникає через наявність багатьох критеріїв, використовуються переваги ОПР. Однак це не так. Інформація ОПР, заснована на її досвіді або інтуїції, є суб'єктивною і залежить від особистості ОПР. Суб'єктивність не означає, що ЛПР «робить, що хоче». У ділових рішеннях людина повинна бути раціональним, а суб'єктивна модель – стійкою. У такій суб'єктивності немає нічого поганого. Ми постійно приймаємо рішення, значна частина з них не може бути названа науково обґрунтованою, якщо розуміти під словом «науково» точні аналітичні критерії. У типовій ситуації з недостатньо визначеними наслідками прийнятих рішень, з динамічною мінливою обстановкою тільки вміння людей будувати гіпотези і доповнювати ними відсутню інформацію може врятувати положення. Досвідчені керівники, які беруть непогані рішення, добре усвідомлюють,

скільки особистого і суб'єктивного вони вносять в ці рішення. По-іншому багато рішень просто не можна прийняти.

Отже, в багатокритеріальних задачах найчастіше «об'єктивне» неможливо, а якість «суб'єктивного» рішення сильно залежить не тільки від особистості ОПР, а й від методів і процедур розробки і обґрунтування рішень. Саме цими методами і процедурами займається теорія багатокритеріальних рішень.

В теорії багатокритеріальних рішень прийняті наступні вимоги до критеріїв.

1) Повнота. Сукупність критеріїв K_1, K_2, \dots, K_m забезпечує об'єктивність оцінки множини рішень Y_1, Y_2, \dots, Y_n , в тому числі відображення особистих інтересів ОПР.

2) Незалежність. Критерії одного рівня ортогональні (це перевіряється шляхом розрахунку коефіцієнта попарної кореляції).

3) Несуперечність. Критерії суперечливі, якщо вони близькі за змістом, але їх напрямки оптимізації протилежні (максимізувати площа кімнат і мінімізувати загальну площу).

4) Ненадлишкових. Критерій надлишковий, якщо він не забезпечує розрізнення рішень.

Багатокритеріальне оцінювання рішень виконується в умовах визначеності, якщо оцінюються всі варіанти на кожному рівні дерева рішень (пошук в ширину). Багатокритеріальне оцінювання рішень пов'язане з невизначеністю, якщо варіанти рішень на наступному кроці залежать від результатів вибору на попередньому кроці (пошук в глибину).

Поставимо у відповідність критерію K_j ($j = 1, \dots, n$) j -ю вісь простору E_n , а кожному i -му рішенню Y_i ($i = 1, \dots, m$) – точку з координатами $\langle K_1(Y_i), K_2(Y_i), \dots, K_n(Y_i) \rangle$. Тоді простором критеріїв називають простір E_n , координати точок якого розглядаються як оцінки альтернативних рішень за відповідними критеріями.

Прийняття рішень зручно інтерпретувати як задачу пошуку в просторі критеріїв. Якщо пошук є цілеспрямованим, а не випадковим, то використовується модель графа у вигляді І/АБО-дерева. Задається множина

початкових вершин графа, з яких пошук може починатися, і множина кінцевих (цільових) вершин, при досягненні яких пошук припиняється.

I/АБО-граф має такі властивості:

1) при русі до деякої вершини реалізується або кон'юнкція (I-вершина), або диз'юнкція (АБО-вершина);

2) при активізації вершини активними стають або усі, хто виходив з вершини дуги (I-вершина), або тільки одна (АБО-вершина).

Відомі різні методи пошуку по дереву рішень. Вони розрізняються в способі обходу шляхів на графі:

1) пошук в глибину і в ширину, які розрізняються порядком обходу вершин (рис. 4),

2) прямий (від кореня до висячої вершини) і зворотний пошук,

3) пошук без повернення і з поверненням,

4) безумовний або умовний пошук (черговий хід залежить від попереднього),

5) повний і скорочений перебір ходів в дереві,

6) пошук без прогновної оцінки (методом проб і помилок) або пошук з прогновною оцінкою (метод гілок і меж, симплекс-метод).

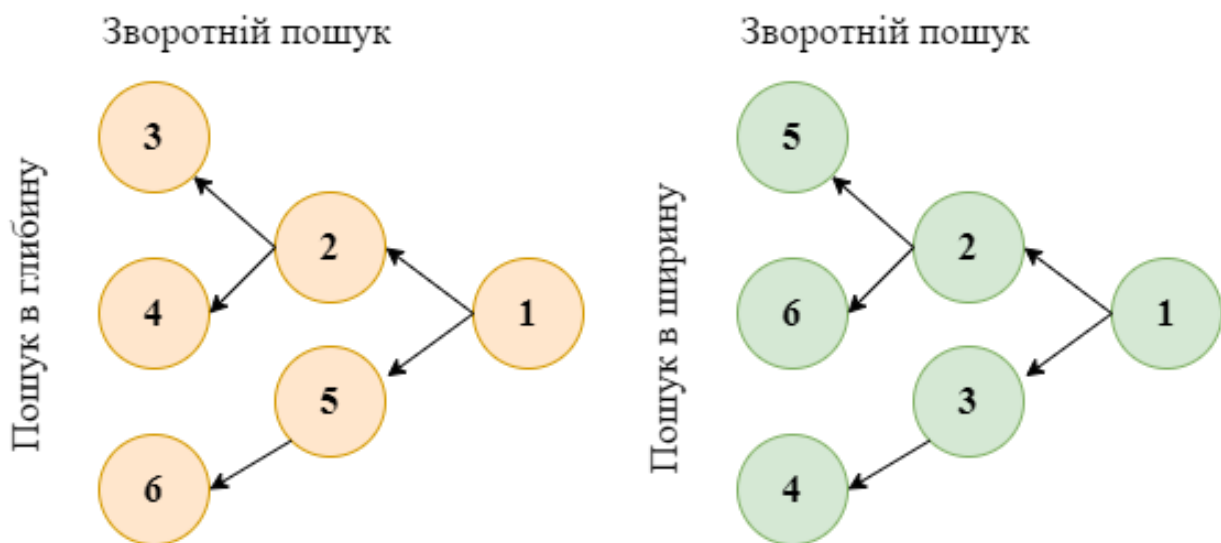


Рисунок 3.1 - Пошук по дереву рішень в глибину і ширину

Для NP-задач дерево пошуку може стати обчислювальною множиною, експоненціально зростаючим при лінійному збільшенні розмірності задачі (комбінаторний вибух).

Якщо в процесі пошуку рішень немає альтернатив, то немає і вибору. Тому для постановки задачі прийняття рішень необхідно мати хоча б дві альтернативи. Альтернативи можуть бути залежними і незалежними. Якщо альтернативи задані і треба знайти кращу з них, то це завдання вибору. Однак існують завдання, де частина альтернатив з'являється в процесі прийняття рішень.

Нехай $Y_i (i=1, \dots, m)$ - спільний простір альтернативних рішень. Щоб здійснити вибір єдиного оптимального рішення Y^* , використовується послідовне звуження множини альтернатив. Зокрема, спочатку вихідна множина Y звужується до множини допустимих рішень.

Областю допустимих рішень $Y_{доп}$ називається множина рішень, які відповідають ресурсним обмеженням завдання і цільовим критеріям, для яких вдалося дати кількісну оцінку.

Багатокритеріальність призводить до необхідності одночасного розгляду простору допустимих рішень $Y_{доп}$ і критеріїв K . Причому множина рішень відображається в множину критеріїв і навпаки:

$$f: Y \rightarrow K \text{ і } f^{-1}: K \rightarrow Y.$$

Відзначимо, що зворотне відображення не обов'язково однозначне, так як різні рішення можуть мати однакову критеріальну оцінку.

Областю досяжності в просторі критеріїв $K_{доc}$ називається образ множини допустимих рішень $Y_{доп}$, що задається відображенням $f: Y \rightarrow K$.

3.5. Прийняття рішень на основі системного підходу

Свідоме життя людини, особливо творча діяльність, представляє безперервну послідовність прийняття рішень по багатьох питань і проблем, що викликається потребою суспільства і його особисто.

Прийняття рішень в філософському розумінні представляється як діалектико-матеріалістичний процес пізнання, що йде по шляху виявлення і подолання протиріч. Це уявлення узгоджується з теорією пізнання істини у відомій тріаді: чуттєве сприйняття - абстрактне мислення - практика.

Уявімо алгоритми процесу прийняття рішення з різних позицій філософії, системного підходу та різноманітних практичних методів (табл. 3.4). З таблиці видно, що проглядається єдина діалектико-матеріалістична суть і принципова наскрізна схема пошуку - від постановки завдання через виявлення протиріч щодо їх вирішення (подолання) і, нарешті, осмислення результату.

Розглянуті методи прийняття рішень умовно розділимо на дві групи: загальні, що охоплюють необмежено широке коло проблем, і більш приватні, які стосуються синтезу нових технічних об'єктів, тобто безпосередньо до інженерної діяльності.

Фактори, що враховуються при ухваленні рішення:

- особа (особи), яка приймає рішення (ОПР), тобто той, кому потрібно вирішувати проблеми, може бути як окремим індивідумом, так і невеликою групою людей і навіть великим колективом;

- керовані змінні, тобто параметри і ситуації, якими може управляти ОПР;

- некеровані змінні, якими не може управляти ОПР; в сукупності ці змінні утворюють «навколишнє середовище», або фон проблеми;

- внутрішні або зовнішні обмеження на можливі значення керованих і некерованих змінних;

- можливі наслідки (рішення, результат) - має бути не менше двох нерівноцінних, так як в противному випадку не має значення, яке рішення прийняти.

Принципи пошуку рішень:

- аналіз поставленого завдання з точки зору її своєчасності і суспільної потреби в ній. Розкриття внутрішніх протиріч в процесах, що зумовили або обумовлюють постановку задачі;

- перевірка правомірності постановки завдання з точки зору загальних законів природи;

- перевірка здійсненності розв'язання задачі на сучасному рівні науки, техніки і виробництва;
- розробка методів розв'язання задачі, вибір головного експериментальних і аналіз отриманих результатів головного експерименту;
- знаходження взаємозв'язку рішень з поставленим завданням.

Таблиця 3.4. - Алгоритм прийняття рішення на основі системного підходу

Термін	Постановка задачі	Виявлення протиріч	Подолання протиріч	Осмислення результату
Філософія	Теорія пізнання	Чуттєве пізнання	Абстрактне мислення	Практика
Системний підхід	Дослідження необхідності. Постановка мети і розуміння задачі. Аналіз структури системи (Цілісність, елементи, зв'язку, взаємодії зі середовищем, функціональність). Вибір і обґрунтування критеріїв оцінки результату; аналіз з позицій надсистеми	Побудова і аналіз дерева протиріч. Аналіз структури системи за принципом ієрархічності: керуючі і керовані елементи і підсистеми. Аналіз впливу навколишнього середовища на систему	Пошук концепцій системи. Побудову і аналіз дерева функцій системи. Функціональні та конструктивні модулі системи. системотехніка	Оцінка рішення. Зворотні зв'язки. Вплив виходу на вхід.

Системний підхід - це напрям методології наукового пізнання і соціальної практики, в основі якого лежить розгляд об'єктів як систем. Системний підхід

орієнтує дослідників на розкриття цілісності об'єкта, на виявлення різноманітних зв'язків в ньому і зведення їх в єдину теоретичну картину [29, 38].

При системному підході система представляється двома складовими:

- зовнішнє оточення, що включає в себе вхід і вихід системи, зв'язок із зовнішнім середовищем і зворотний зв'язок;
- внутрішня структура, що забезпечує переробку входу системи і її вихід і досягнення цілей системи.

До системного підходу можна також віднести і комплексний підхід.

Системний підхід нерозривно пов'язаний з матеріалістичної діалектикою, є конкретизацією її основних параметрів.

Аналіз і синтез є фундаментальними поняттями в філософії і в системному підході. Якщо аналіз - процес уявного розчленування (Декомпозиції) або реального розбиття об'єкта на елементи з урахуванням наявних між ними зв'язків, то синтез - процес воз'єднання елементів в одне ціле.

Аналіз і синтез системи у взаємозв'язку виявляють, з яких частин складається цілісна система і як вони (частини) взаємодіють один з другим. Таким чином розкриваються принципи функціонально-структурної організації системи. Не слід повністю отожднювати діалектику і системний підхід. На відміну від діалектики системний підхід представляє собою спеціалізовану методологію, хоча і має загально-наукове значення.

Діалектичний матеріалізм - це цілісний світогляд, наукова теорія і методологія, а системний підхід - тільки одна з його граней. Системний підхід є конкретизацією діалектики на сучасному етапі розвитку.

Діалектика являє собою загальну теорію розвитку. Дія основних законів діалектики виявляється також у розвитку систем. Розглянемо основні закони діалектики в додатку до закономірності розвитку техніки.

Закон єдності і боротьби протилежностей. причину всякого процесу становить взаємодія двох протилежних начал, які провокують і перешкоджають розвитку. Суперечливу роль інформаційних процесів розкриває закон необхідної різноманітності Ешбі. Тільки постійно змінюючись під впливом

навколишнього середовища, тільки відбиваючи її зміни, система може залишитися собою, зберігаючи свою якість.

Прогресивний розвиток завжди пов'язане з обмеженням різноманіття. З безлічі можливих сценаріїв еволюції на практиці реалізується тільки один. Але цей процес супроводжується збільшенням складності, внутрішнього різноманіття прогресуючих систем. Отже, будь-який процес руху інформації пов'язаний зі знищенням, обмеженням одного виду різноманітності і одночасним збільшенням іншого його виду.

Закон переходу кількісних змін у якісні. У процесі еволюції будь-якої системи відбуваються накопичення кількісних змін і перехід їх в якісні зміни.

Закон заперечення заперечення. Цей закон діє в сфері висхідного розвитку, при якій відбувається збільшення параметрів системи.

У загальному випадку закон стверджує, що на вищому щаблі розвитку повторюються деякі риси, властиві вихідної ступені. Зміст, що досягається системою на послідовних ступенях розвитку, змінюється за рахунок чергових заперечень окремих його частин. Друге заперечення, що знімає перше, веде до часткового відтворення на якісно новому рівні інформації, збагаченої досвідом двох ступенів розвитку.

Система складається з вихідних одиниць - компонентів. В якості компонентів системи (в широкому сенсі) можуть розглядатися об'єктивні, властивості, зв'язку, відносини, стану, фази функціонування, стадії розвитку. Об'єкти, що представляють собою одиниці, з яких встановлено систему, можуть бути матеріальними і нематеріальними.

Властивості системи, специфічні для даного класу об'єктів, можуть стати компонентами системного аналізу. Властивості можуть бути як змінними, так і незмінними за даних умов існування системи: внутрішніх (власних) і зовнішніх. Власні властивості залежать тільки від зв'язків (взаємодій) всередині системи, це властивості системи «самої по собі». Зовнішні властивості існують лише тоді, коли є зв'язки, взаємодії з зовнішніми об'єктами (системами).

Зв'язки досліджуваного об'єкта також можуть бути компонентами при його системному аналізі. Зв'язки мають матеріально-енергетичний характер.

Логіка (грец. Logike) - наука про хід і способах доказів і спростувань. В основі формально-логічних методів прийняття рішень лежить використання логічних законів вивідного значення, отриманого логічно з попередніх знань без безпосереднього відношення до досвіду. Основна вимога логіки - обов'язковість послідовного несуперечливого, обґрунтованого мислення. Не можна вважати істинними знання, що містять логічні суперечності. Логіка допомагає інтенсифікувати будь-яку розумову діяльність.

Виняткову ефективність нешаблонного мислення ми можемо бачити в екстремальних і, здавалося б, безвихідних ситуаціях, а й тут не можна відкидати логіку. Різниця між шаблонним і нешаблонним мисленням полягає в тому, що при шаблонному мисленні логіка керує розумом, тоді як при нешаблонному вона його обслуговує. Шаблонне мислення називають «психологічної інерцією», під яким мається на увазі несвідоме схилення до якогось конкретного методу або способу мислення, які зазвичай характеризують виразом «йти протоптаною стежкою». психологічна інерція - це наслідок існуючих методів навчання, за якими вивчають не методи видобутку знань, а готові конкретні рецепти.

3.6. Математичний підхід до прийняття рішень на основі теорії ігор

Лише кілька десятиліть тому мистецтво прийняття рішень, базувалося на досвіді, інтуїції і здоровому глузді, а зараз стало перетворюватись в точну науку. Зараз проблеми прийняття рішень вивчаються спеціалістами в області системного аналізу, дослідження операцій і управління, використовуються багатовимірною теорією корисності як самостійна наукова дисципліна, методи багатокритеріальних задач прийняття рішень, методи оптимізації та прогнозування. Але і при математичних підходах основна частина алгоритму системного підходу до прийняття рішення залишається неформалізованою, виконується людиною при постановці задач і узагальненні результатів математичного аналізу.

Особа, яка приймає рішення (ОПР), має визначити критерії оцінки (цільову функцію) можливих рішень проблеми. Цими критеріями можуть бути,

наприклад, витрати, прибуток, підвищення продуктивності, запобігання аваріям.

Успішність вирішення проблеми залежить від того, наскільки успішно розроблені можливі альтернативи. Бажано не упустити жодної скільки-небудь значущої альтернативи, включаючи альтернативу нічого не робити. Для аналізу і порівняння альтернатив, вибору найкращої альтернативи часто застосовуються математичні методи, включаючи методи оптимізації і прогнозування.

Здійснення обраної альтернативи означає просто здійснення дій, позначених в альтернативі. Ухвалення рішення вимагає перевірки результатів його здійснення і, при необхідності, усунення помилки. На прийняття рішення завжди накладаються об'єктивні обмеження, викликані витратами, людськими можливостями, строків її виплати, технологією і наявністю інформації.

Причиною невдач технічних рішень є субоптимізація, коли приймаються оптимальні рішення для однієї складової частини системи, що призводять до неоптимального рішення всієї системи. вибір техніки аналізу визначається умовами, в яких приймаються рішення. Ці умови класифікуються за ступенем точності і впевненості. Існує три основні категорії умов: впевненість, ризик і невизначеність.

В умовах впевненості вибирається альтернатива, яка дає найбільше (найменше) значення основного критерію. Фактор ризику знаходиться між двома полярними випадками - впевненістю і невизначеністю.

Методи інтенсифікації розумового процесу

До методів інтенсифікації розумового процесу при прийнятті рішень відносяться мозкова атака, ділові ігри, конференції ідей, методи експертних оцінок та інші методи.

При мозковій атаці процес висунення ідей відбувається лавиноподібно. Висунута ідея породжує або творчу, або критичну реакцію, що стимулює появу нових ідей. Групове мислення виробляє на 70% більше цінних нових ідей, ніж сума індивідуальних мислень. Існує кілька модифікацій мозкової атаки: індивідуальна, масова, письмова, подвійна і зворотна, а також синектика.

В основу синектики (поєднання різнорідних елементів) покладена мозкова атака, яка відрізняється тим, що тут використовуються постійні групи у складі фахівців різних професій. Рекомендується, щоб члени синектичної групи (крім керівника) перед початком роботи не знали суті проблеми, що розглядається, що дозволяє їм відійти від звичного стереотипу мислення, успішно подолавши психологічну інерцію мислення. Бо розумова діяльність людини більш продуктивна в новій, незнайомій йому обстановці.

Конференція ідей - одна з різновидів колективної творчості. Від мозкової атаки вона відрізняється насамперед темпом роботи і проводиться у вигляді наради з висунення ідеї з допущенням бажаною критики в формі реплік, коментарів і т.д. Вважається що критика може навіть підвищити цінність висунутих ідей. Всі висунуті ідеї фіксуються в протоколі без вказівки авторів. У цьому закладається той суттєвий зміст, що результати конференції ідей є як би колективною працею [40].

Ділові гри є метод імітації прийняття управлінських та інших рішень в різних ситуаціях (виробничих і невиробничих) шляхом гри за заданими правилами групи людей або людини з ЕОМ. Програється безліч ситуацій як би довільних. Насправді ж, в силу специфічних дискусійних прийомів, плідність яких відзначали ще древні («істина народжується в суперечці »), виникає ряд альтернативних рішень.

Сутність експертних методів полягає у використанні досвіду роботи, ерудиції та інтуїції висококваліфікованих фахівців, здатних знаходити рішення в умовах важко формалізованих ситуацій і недостатньої інформації. Методи експертних оцінок дозволяють кваліфікувати (кількісно виразити) якісні характеристики досліджуваного об'єкта. При цьому реалізуються можливості системного підходу, оскільки інтегрально використовується інформація, якою володіє група експертів.

При системному підході прийняття рішень не можна виключити евристичні методи (здогади, осяяння і т.д.), над проблемою можна працювати роками, але ідея може виникнути миттєво, як результат осяяння.

У багатьох зрілих інженерів, вчених, педагогів на основі великого особистого досвіду виробляється «своя система» прийняття рішень. Але опанувавши системним підходом до прийняття рішень, можна швидше досягнути і вибрати для своєї діяльності оптимальні методи прийняття рішень.

Системний підхід до прийняття рішень полягає в наступному:

- прийняття рішень є не початковим, а завершальним етапом творчого циклу, який починається з виділення системи, що визначає проблемну ситуацію, потім триває у виявленні тих закономірностей, за якими розвивається і функціонує дана система, і тільки потім настає етап вибору методу прийняття рішення;

- можливість вибору методів прийняття рішень забезпечується використанням функціонально-структурного підходу;

- процес прийняття рішення не можна відокремити від «людського фактору», від психологічних і соціально-економічних чинників, від особливостей особистості, зокрема сміливості й уміння ввести (включити) в рішення деяку ступінь ризику.

Вироблення вірних рішень - це не тільки наука, а також інтуїція, досвід, чуття, все те, що називається словом «мистецтво». У їх єдності народжується вища мудрість.

Представляючи процес інженерного творчості як зв'язок трьох неподільних складових (системний підхід - закони розвитку – прийняття рішень), розкриємо зміст, вкладений в концепцію сучасного погляду на наукову, інженерну та навчальну діяльності:

- по-перше, світоглядна позиція, заснована на діалектичному матеріалізмі. Знаходить ця позиція своє відображення в системному (Функціонально-структурному) підході;

- по-друге, застосовуючи системний підхід до вирішення завдань технічних систем, необхідно базуватись на законах і закономірності їх розвитку;

- по-третє, результатом системного підходу до завдань розвитку техніки є прийняття рішення, яке виражає процес розгляду і подолання протиріч. Тут

дуже важливо володіти різноманітними методами активізації творчого мислення та використовувати накопичені в різних галузях техніки досвід, банки даних.

РОЗДІЛ 4

РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАСОБУ З ФІНАНСУВАННЯ ДОСЛІДЖЕНЬ

4.1 Постановка задачі до розробки програмного засобу на основі методу дерева у прийнятті рішень

Метод дерева рішень використовується в задачах класифікації та прогнозування, у випадках коли рішення доводиться приймати в умовах ризику, невизначеності та результат подій залежить від ймовірностей. Відомо, що на кожне рішення впливають певні чинники. Також, у кожного рішення є свої наслідки під час його прийому, яким притаманний імовірнісний характер.

У цих умовах процес прийняття рішень є послідовним і метод дерева рішень передбачає визначати, які дії слід зробити в кожній з його вершин.

Модель дерева рішень - це математична модель, яка задає процес прийняття рішень так, що будуть відображені кожне можливе рішення, попередні та наступні перед цим рішенням події або інші рішення та наслідки кожного кінцевого рішення.

Дерево рішень складається з наступних елементів: дуг, вузлів рішень, вузлів подій і кінцевих вузлів (результатів).

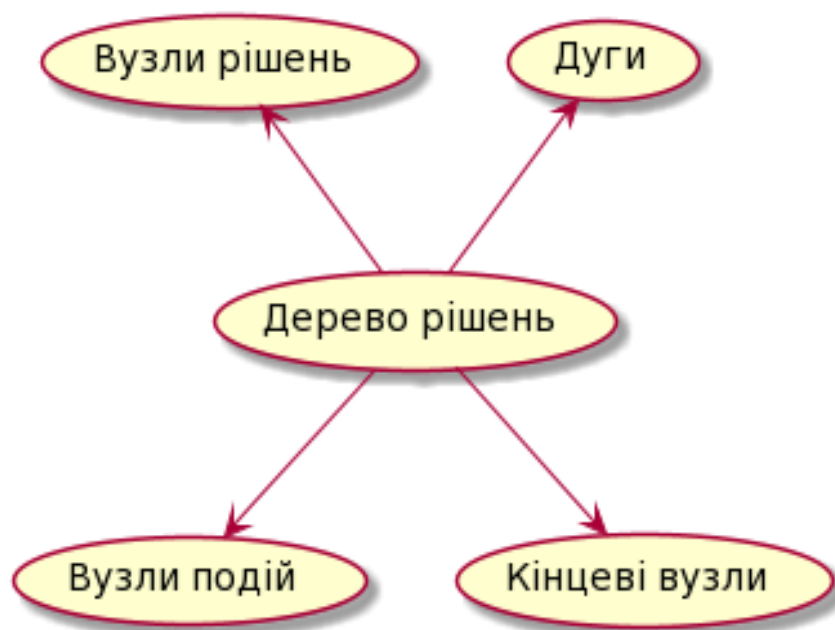


Рисунок 4.1 – Основні елементи дерева рішень

Перед розробкою постановки задачі необхідно розглянути приклади використання методу дерева рішень у фінансуванні досліджень. Розглянемо спочатку найпростіший приклад застосування методу дерева рішень і вирішимо наступну задачу.

Приклад аналізу методу дерева рішень 1.

Постановка задачі відповідно до предметної області фінансування досліджень може бути така: необхідно вирішити за допомогою методу дерева рішень, яку суму грошей необхідно витрати на реалізацію деякого проекту.

Рішення. Альтернативні події на першому етапі визначаються наступним чином: якщо проект буде зроблений своєчасно та поза строками. Далі відображаємо ці події в дереві рішень у вигляді вершин подій (кружечків).

Прийmemo, що, якщо проект буде зроблений своєчасно, то на його реалізацію доведеться витратити велику суму. Це один з результатів (кінцевих вузлів в дереві рішень) в даній задачі.

У разі, якщо проект не буде зроблений своєчасно, то можливі два альтернативних події: в ІТ компанії багато подібних проектів або мало.

Відображаємо ці події в дереві рішень у вигляді вершин подій (кружечків), до яких ведуть дуги з події "проект виконано не своєчасно" (рис. 4.2).

У разі, якщо новий проект виконується своєчасно, то потрібні додаткові кошти на його презентацію та рекламу. Також, якщо в компанії було виконано багато подібних проектів, то витрачається на подібний проект малу суму. У разі, якщо в будинку мало проектів, то витрачається на реалізацію подібного проекту середню суму. Це відбувається внаслідок того, що у програмних бібліотеках вже існують подібні функції, які реалізуються в проектах та на написання нового потрібно менше часу програміста.

Але навіть для цієї щойно розглянутої задачі альтернативні події можуть бути пов'язані з деякими можливостями. Тоді, модель задачі буде така.

Зокрема, можливо зі свого досвіду або ж від одного з можливих експертів відомо знати ймовірність того, що проект буде зроблений своєчасно.

Далі припустимо, ймовірність цієї події дорівнює 0,3. Тоді від одиниці віднімемо 0,3 і отримаємо 0,7 - ймовірність того, що проект не буде зроблений своєчасно. Події, які впливають з цієї події також можуть бути пов'язані з можливостями.

Розглянемо ще продовження даного прикладу. Припустимо, що не має інформації відносно того, скільки подібних проектів має ІТ компанія. Або взагалі, точна кількість подібних проектів в ІТ стане відома тільки через кілька днів. Також, у більшості випадків керівництво компанії оцінює ймовірності альтернативних подій.

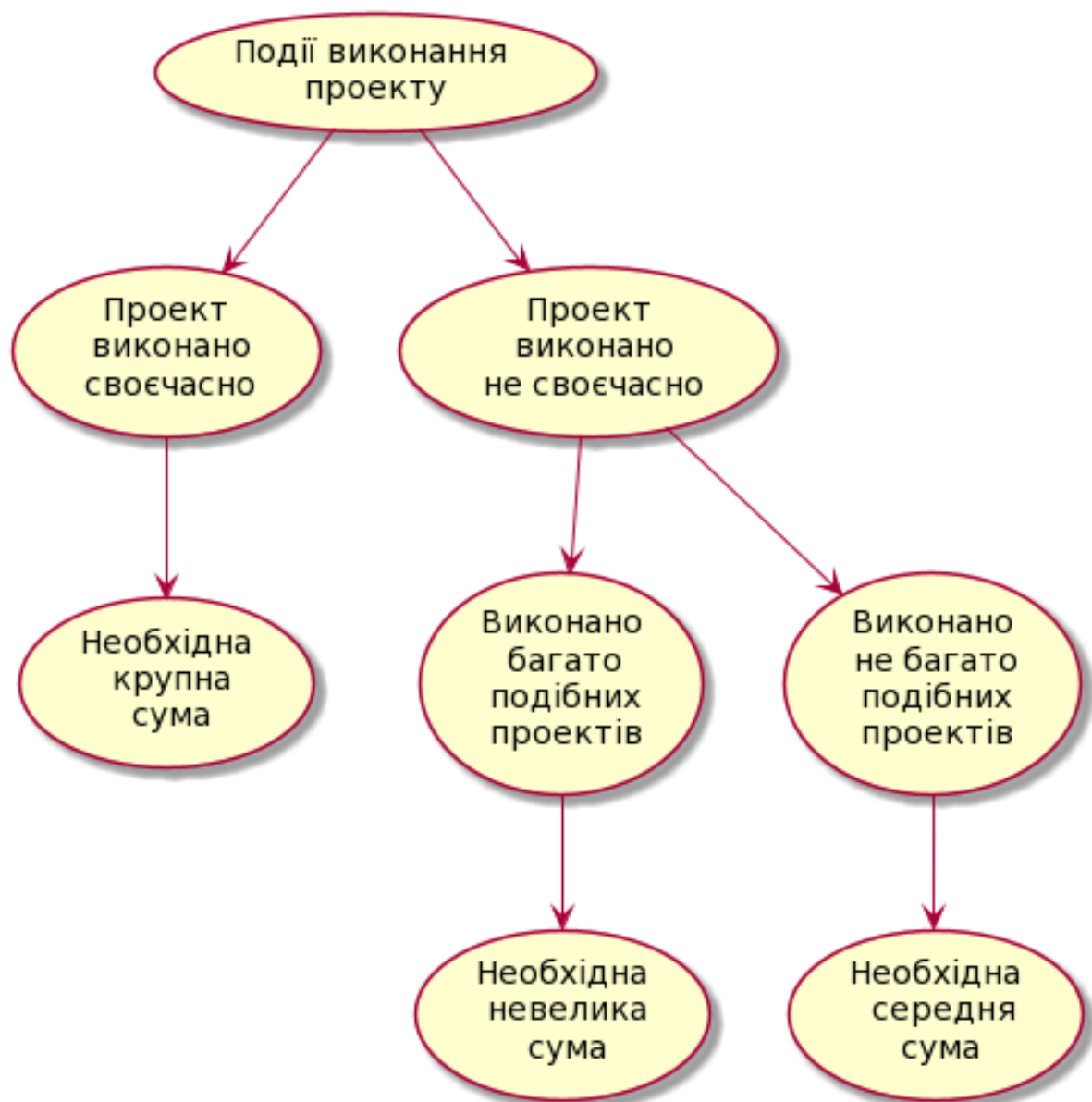


Рисунок 4.2 – Дерево рішень для задачі з реалізації проектів

Тому припустимо, що ймовірність того, що в ІТ компанії є багато подібних проектів дорівнює 0,4.

Тоді $1 - 0,4 = 0,6$ - ймовірність того, що в ІТ компанії мало подібних проектів. Отже, маючи оцінки ймовірностей альтернативних подій, вирішується та чи інша задача методом дерева рішень, за допомогою якого прогноуються результати розвитку ІТ компанії.

Таким чином, у постановку задачі щодо розробки програмного засобу на основі методу дерева у прийнятті рішень необхідно визначити загальний вигляд граф-дерева, який має такі властивості:

- є вершини альтернативних рішень, які зазвичай відображаються у вигляді прямокутників;
- є вершини подій, які зазвичай відображаються за допомогою кіл;
- є вершини кінцевих рішень (листя), які зазвичай відображаються за допомогою прямокутників, що мають інший колір, ніж вершини альтернативних рішень;
- вершина альтернативних рішень разом з дугами, що виходять з цієї вершини, називається виделкою рішень;
- вершина подій разом з дугами, що виходять з цієї вершини, називається стохастичною виделкою (стохастичною - має імовірнісний характер, так як в процесі прийняття рішення враховуються ймовірності кожного з подій);
- кожній дузі вилки рішень приписані ваги, які характеризують придбання або збитки при реалізації даного рішення;
- кожній дузі стохастичною вилки приписуються ваги, які характеризують ймовірність відповідного події, а також придбання та збитки при реалізації даної події.

4.2 Основні етапи проектування програмного засобу на основі методу дерева рішень

До основних етапів аналізу дерева рішень, що необхідні під час вирішення прикладних задач, знаходять у такий спосіб:

- на першому кроці проходять від початкової вершини до такої кінцевої вершині (листу), якій відповідає найбільша придбання, не враховуючи при цьому ймовірності подій, що передують рішенням;
- на другому кроці враховують ймовірності подій, що передують рішенням; створення вершин, що відображають кінцеві рішення; виконання розрахунків з урахуванням ймовірностей подій, що передують рішенням.

Розглянемо основні етапи проектування програмного засобу з фінансування досліджень на основі методу дерева рішень за допомогою наступного прикладу: необхідно вирішити методом дерева рішень завдання прогнозування придбань і збитків при альтернативних рішеннях.

Так, будівельна компанія, яка приймає участь у побудові житлового комплексу який необхідний для побудови лабораторій з дослідження COVID-19 збирається прийняти рішення про будівництво житлового комплексу (ЖК) в елітному районі.

Етап проектування 1. Спочатку потрібно прийняти рішення: чи проводити інформаційно-рекламну кампанію. Вона стоїть 500000 умовних одиниць (у.о.). Досвід показує, що лише в 25% випадків цей крок забезпечує успіх на ринку.

Якщо інформаційно-рекламна кампанія успішна, потрібно прийняти рішення: чи будувати великий чи малий ЖК. Будівництво малого ЖК обійдеться в 50000000, при цьому можна побудувати 300 квартир. Будівництво великого ЖК обійдеться в 200000000, при цьому можна побудувати 900 квартир.

При цьому проводяться дослідження прогнозу попиту. Вони показують, що існує ймовірність в 40% того, що відбудеться падіння попиту на елітне житло.

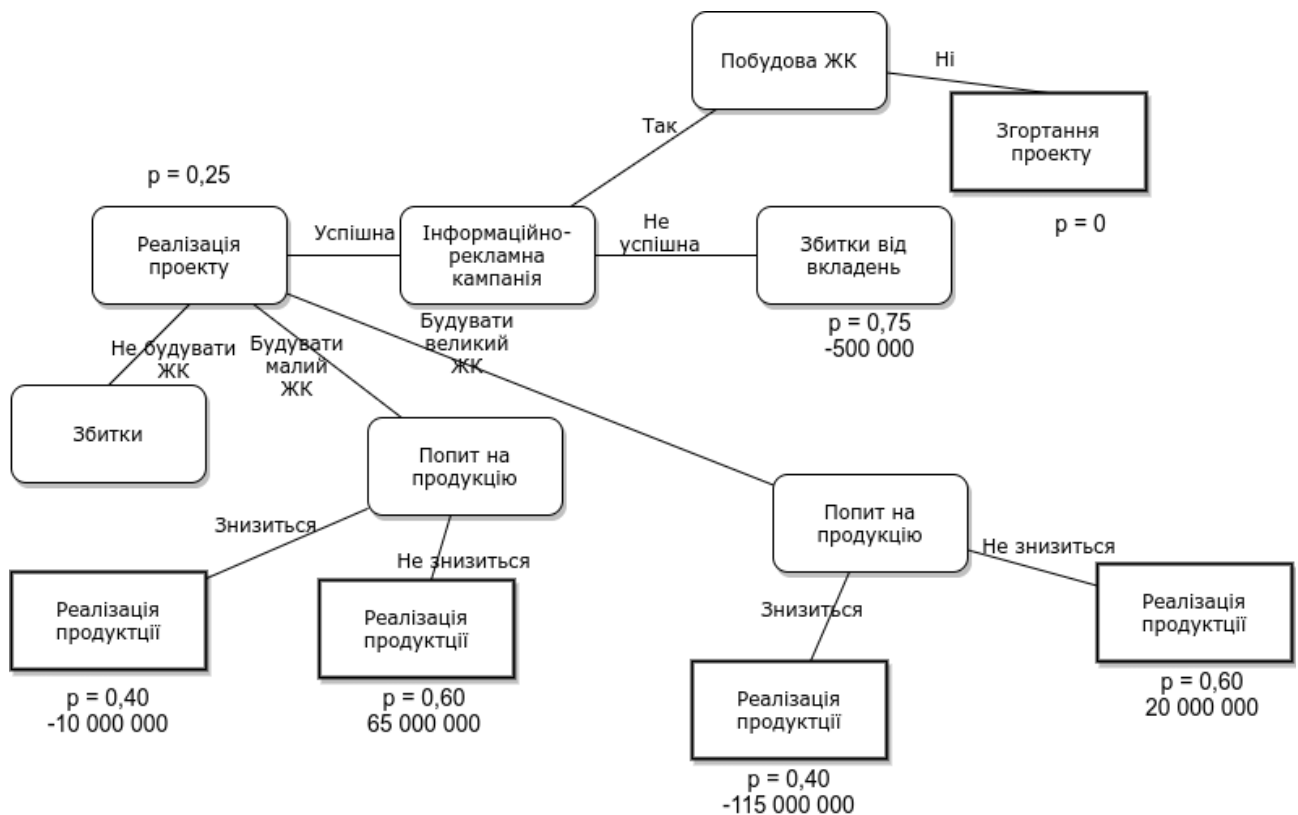
Етап проектування 2. Визначення кошторису житлового фонду. За попередніми розрахунками, середні ціни на квартири будуть визначатися в такий спосіб:

	Великий ЖК	Малий ЖК
Попит знизиться	100 000	150 000
Попит не знизиться	250 000	400 000

Розраховано, що витрати фірми перед і в період продажу всіх квартир в ЖК складуть 5000000, незалежно від величини ЖК.

Етап проектування 3. Визначення кошторису проведення рекламної кампанії. Потрібно прийняти рішення: чи проводити інформаційно-рекламну кампанію і починати будівництво ЖК.

Рішення. Будуємо дерево рішень:



Наслідки альтернативних рішень отримані при розрахунках, які наведені нижче.

Крок 1 (не враховуються ймовірності подій).

Для малого ЖК:

Попит знизився:

Доходи	Витрати	Прибуток
$400000 * 300 =$ 120000000	$50000000 + 5000000 =$ 55000000	$120000000 - 55000000 =$ 65000000

Попит не знизився:

Доходи	Витрати	Прибуток
$150000 * 300 = 45000000$	$50000000 + 5000000 =$ 55000000	$45000000 - 55000000 = -$ 10000000

Для великого ЖК:

Попит знизився:

Доходи	Витрати	Прибуток
$250000 * 900 =$ 225000000	$200000000 + 5000000 =$ 205000000	$225000000 - 205000000 =$ 20000000

Попит не знизився:

Доходи	Витрати	Прибуток
$100000 * 900 = 90000000$	$200000000 + 5000000 =$ 205000000	$90000000 - 205000000 = -$ 115000000

Крок 2 (враховуються ймовірності подій).

На цьому кроці прогнозовані придбання фірми розраховуються з урахуванням ймовірності подій, що передують альтернативам рішень.

Найменування	Витрати
--------------	---------

Малий ЖК	$0,60 * 65000000 + 0,40 * (-10000000) =$ 35000000
Великий ЖК	$0,60 * 20000000 + 0,40 * (-115000000)$ $= -34000000$

Етап проектування 3. Виконання розрахунків виникнення подій. На цьому етапі слід розрахувати придбання при вершині 3 дерева рішень (проводити інформаційно-рекламну кампанію). Для цього потрібно розрахувати придбання при вершинах 4 (неуспішна інформаційно-рекламна кампанія) і 5 (успішна інформаційно-рекламна кампанія).

Наступ події при вершині 5 (будівництво ЖК) означає максимальний прибуток 35000000, яку тільки можна отримати при виборі даного рішення. Наступ події при вершині 4 (неуспішна інформаційно-рекламна кампанія) означає збитки в 500000.

Далі можна розрахувати придбання при вершині 3 дерева рішень (проводити інформаційно-рекламну кампанію):

$$0,25 * 35000000 + 0,75 * (-500000) = 8750000 - 375000 = 8375000.$$

Етап проектування 4. Визначення результатів використання методу з прийняття рішень на основі дерева рішень. В результаті застосування методу дерева рішень вибирається наступна альтернатива: почати інформаційно-рекламну кампанію і готуватися до будівництва ЖК, віддаючи перевагу малому ЖК.

Слід зазначити, що на практиці при виборі рішення методом дерева рішень в разі інвестиційних проектів будуються песимістичний і оптимістичний прогнози. Вони відрізняються співвідношеннями ймовірностей сприятливого і несприятливого наслідків подій. Для кожного результату, таким чином, розраховуються оптимістичні і песимістичні значення.

4.3 Вибір математичних моделей дерева графу

Для розробки програмного засобу на основі прийняття рішень за допомогою дерева рішень необхідно визначити математичні моделі, які можуть бути використані (табл. 4.1).

Відомо, що блок-схема комп'ютерної програми (вершини - команди, ребра - переходи між командами), яка використовується для розробки і тестування самої програми доцільно для використання у програмному засобу. Далі, розглянемо інші приклади використання дерев графів.

Граф підпрограм (вершини - підпрограми, ребра - порядок виклику підпрограм), який використовується для проектування і аналізу комп'ютерних програм.

Граф структури даних (вершини - дані або найпростіші типи даних, ребра - відносини між даними), який використовується для проектування і оптимізації структур даних.

Граф залежності команд машинного коду (вершини - команди, ребра - залежності між командами), який використовується в оптимізаційній компіляції і конвейеризації роботи процесора.

Таблиця 4.1 - Порівняльна характеристика деяких математичних моделей у вигляді графа

Характеристика графа	Представлення вершин	Представлення ребер
Блок-схема комп'ютерної програми	Команди	Переходи між командами
Граф структури даних	Дані	Відносини між даними
Граф кінцевого автомата	Стан автомата	Переходи
Граф тверджень математичної теорії	Твердження	Відносини логічного слідування
Функціональний граф	Елементи множини	Відносини логічного слідування
Граф метрики	Будь-які фізичні або не	Геометрична,

	фізичні об'єкти	структурна, функціональна близькість цих об'єктів
Дерево рішень	Критичні стани	Рішення
Системний граф	Компоненти системи	Взаємодія компонент
Граф конфліктів	Стан будь-якої системи	Конфлікти між станами
Граф гри	Ігрові стани	Переходи між станами
Граф проекту	Роботи або стани проекту	Відносини між роботами
Граф доріг	Міста	Дороги

Граф процесів операційної системи (вершини - процеси операційної системи, ребра - акти генерування процесів), який використовується в моделюванні роботи операційної системи.

Граф кінцевого автомата (вершини - стану автомата, ребра - переходи), який використовується в дослідженні кінцевих автоматів і мов і в інженерії програмного забезпечення.

Граф тверджень математичної теорії (вершини - затвердження, ребра - відносини логічного слідування), який використовується для доказу математичних висновків і аналізу математичних теорій.

Функціональний граф (вершини - елементи множини, ребра - відносини логічного слідування), який має широке використання в математиці.

Граф величин-залежностей (вершини - чисельні величини і взаємозв'язку між ними, ребра - відносини залученості величини), який використовується в рішенні різних математичних задач.

Граф метрики (вершини - будь-які фізичні або не фізичні об'єкти або їх безлічі, ребра - геометрична, структурна, функціональна або еволюційна близькість цих об'єктів), який використовується для аналізу великих множин.

Дерево рішень (вершини - критичні стани, ребра - рішення), який використовується в прийнятті рішень в економіці, управлінні, діагностиці інженерних систем.

Системний граф (вершини - компоненти системи, ребра - взаємодія компонент), який використовується в проектуванні та аналізі систем.

Граф зворотних зв'язків (вершини - параметри будь-якого процесу, орієнтовані ребра з вагами "+" або "-" - залежність змін параметрів, відповідних вершин), який використовується в дослідженнях змін складових частин процесів або об'єктів.

Граф причинно-наслідкових зв'язків (вершини - стану будь-якої системи, орієнтовані ребра - причинно-наслідкові зв'язки), який використовується в дослідженнях великих систем і складних процесів.

Граф конфліктів (вершини - стану будь-якої системи, ребра - конфлікти між станами), який використовується в аналізі систем.

Граф гри (вершини - ігрові стани, ребра - дозволені правилами гри переходи між станами (ходи)), який використовується в розробці переможних стратегій в іграх.

Комп'ютерна мережа (вершини - комп'ютери або комунікаційні вузли, ребра - лінії зв'язку), яка використовується в проектуванні і аналізі комп'ютерних мереж.

Соціальний граф (вершини - люди або безлічі людей, ребра - відносини знайомства, економічні відносини або інші відносини), який використовується в аналізі суспільства і плануванні розвитку.

Організаційний граф (вершини - люди або безлічі людей, ребра - відносини, що характеризують організації, наприклад, приватні фірми, ієрархії), який використовується в створенні організацій та управлінні ними.

Граф проекту (вершини - роботи або стану проекту, ребра - відносини між роботами або роботи, що з'єднують стану), який використовується в керівництві проектами.

Генеалогічне древо (вершини - люди, ребра - відношення "батьки-діти"), який використовується в особистих дослідженнях.

Граф економічних агентів (вершини - економічні агенти - люди, фірми та ін., Ребра - економічні відносини), який використовується в економічних дослідженнях і плануванні.

Макроекономічний граф фінансового потоку (вершини - галузі економіки, ребра - фінансові потоки), що використовується в економічних дослідженнях і плануванні.

Граф доріг (вершини - міста, ребра - дороги), який використовується в розвитку транспортної мережі.

Граф вулиць (вершини - перехрестя, ребра - вулиці), який використовується в аналізі і плануванні потоку міського транспорту.

Граф електричного кола (вершини - електромагнітно-активні елементи, ребра - дроти і контакти), який використовується в побудові і аналізі електричних з чим.

Ланцюг живлення (вершини - породи тварин, ребра - відносини харчування), який використовується в аналізі біосистем.

Дерево еволюції (вершини - породи або популяції, ребра - відносини еволюційного походження), який використовується в біології.

Граф хімічних реакцій (вершини - вектори кількості хімічних речовин, ребра - хімічні реакції), що використовується в аналізі хімічних реакцій.

Граф попередників хімічної речовини (вершини - хімічні речовини, отримані в процесі виробництва, ребра - зміни речовин-попередників в процесі виробництва), що використовується в хімічній промисловості.

Граф реакційної здатності хімічних речовин (вершини - хімічні речовини, ребра - здатності до реакції), що використовується в аналізі складних хімічних реакцій.

Граф вибухонебезпечності (вершини - хімічні речовини, ребра - можливості вибуховий реакції між ними), який використовується для потреб безпеки праці.

Граф перешкод радіозв'язку (вершини - радіостанції, ребра - взаємне перекриття смуг радіохвиль), який використовується в плануванні радіозв'язку.

Політичний граф (вершини - держави, ребра - кордони), який використовується в геополітиці.

Граф кровоносних судин (вершини - вузли кровоносних судин, ребра - кровоносні судини), що використовується в медицині.

Граф нейронів (вершини - нейрони, ребра - місця зіткнення нейронів), що використовується в медицині.

Граф приготування кулінарного виробу (вершини - стану готовності кулінарного виробу, ребра - переходи між станами в процесі приготуванні), який використовується в кулінарії.

4.4 Використання різних математичних моделей у вигляді графів

Зазвичай математичні моделі створюються з метою кращого розуміння досліджуваного явища. Але досить складні системи неможливо аналізувати без допомоги спрощених моделей. Дослідження системи з перенесенням її властивостей на модель називається моделюванням. При цьому математичне моделювання дозволяє абстрагуватися від фізичних та інших специфічних властивостей об'єкта.

Таким чином, математична модель - це спрощене і самодостатнє в понятійному плані відображення реальності. У подальшому, для розгляду різних математичних моделей у вигляді графів будемо використовувати наступні визначення.

Визначення 1. Математична модель - завдання системи або явища будь-якого походження та будь-якої природи у вигляді математичних об'єктів і взаємозв'язків.

Визначення 2. Математична модель - наближений опис якого-небудь класу явищ зовнішнього світу, виражене за допомогою математичної символіки.

Будь-яка математична модель передбачає наявність об'єктів і взаємозв'язків між ними.

Далі, сформулюємо основні принципи створення моделей на основі моделі дерева рішень, які полягають у наступному.

За допомогою основних принципів створення моделей на основі теорії дерева рішень відбувається вибір певної стратегії, яка істотно впливає на процес вирішення завдання і на саме рішення.

До таких основних принципів створення моделей на основі теорії дерева рішень відносяться наступні:

- ступеня деталізації моделі можуть бути різними.
- більш докладна деталізація означає більшу складність моделі.
- кращі моделі - ті, які краще відображають реальність.
- бажано використовувати одночасно кілька моделей.
- існує незліченна безліч прикладів, коли в явищі, системі або процесі уgliedили відносини у вигляді графа, реалізували математичну модель на основі цих відносин і ця реалізація зробила аналіз завдання і рішення більш ефективними.

Наведемо основні типи математичних моделей у вигляді графів, які використовуються на основі основних принципів створення моделей на основі теорії дерева рішень:

- вершини і ребра - фізичні об'єкти, об'єкти-ребра фізично пов'язують вершини;
- вершини - фізичні або не фізичні об'єкти, ребра пов'язують вершини в залежності від структурних або функціональних властивостей вершин;
- вершини - фізичні або не фізичні об'єкти, можливо, в різних станах, що залежать від часу і розвитку, ребра показують часу, еволюційну або причинно-наслідковий зв'язок між вершинами.

При цьому, є дві умови створення ефективної математичної моделі системи у вигляді графа:

- визначені найважливіші підсистеми або стану модельованої системи, які відображаються у вигляді вершин графа;
- визначені найважливіші відносини між об'єктами-вершинами.

Найчастіше використовуються в галузі фінансування перспективних проектів досліджуються моделі, в яких основну роль відіграють вершини графа, а не його ребра.

Також існують системи, для яких неможливо побудувати моделі у вигляді графа. Зазвичай це системи, властивості яких неперервні, наприклад, атмосферні явища, потоки рідин та інші.

Таким чином, головна умова можливості побудови моделі у вигляді графа на основі принципів створення моделей є наступні:

- дискретний (кінцевий, переривчастий) характер системи, явлення або процесу;
- системи з властивостями безперервності доцільніше моделювати за допомогою рівнянь математичної фізики.

Але і системи з дискретним характером не завжди доречно моделювати за допомогою графів. Наприклад, якщо мета рішення завдання - оптимізація, то для цього більше підходить складання і рішення задачі математичного програмування, зокрема, лінійного програмування, яка потребує аналізу взаємозв'язків між об'єктами.

4.5 Розробка основної структури програмного засобу рішення задачі

Основна структура програмного засобу з проведення досліджень в галузі фінансування складається з таких основних компонентів які реалізують зазначені вимоги, що описані в підрозділі 4.1 та основні етапи, що показані в підрозділі 4.2.

До основних етапів з розробки програмного засобу відносяться визначення події рекламної кампанії, визначення кошторису житлового фонду, визначення кошторису рекламної кампанії, отримання результатів прибуткової або збиткової діяльності у гривневому еквіваленті. Загальний алгоритм роботи програмного засобу без налаштування певних розрахунків показано на рис. 4.3.

За наведеним алгоритмом (рис. 4.3) виконувалась розробка програмного коду на мові високого рівня Java, яка полягала в наступному. Спочатку виконуються введення реквізитів проекту, що розглядається. Потім йде визначення даних щодо проведення рекламної кампанії. Якщо результати проведення рекламної кампанії є не успішними, то визначається розмір збитків та алгоритм роботи програмного засобу завершується.

Далі для продовження алгоритму роботи визначається розмір житлового кошторису, де в свою чергу запитується дані про попит на реалізацію продукції. Завдяки цьому, виконуються розрахунки на збитки або прибутки відповідно до кожного з дерева рішень.

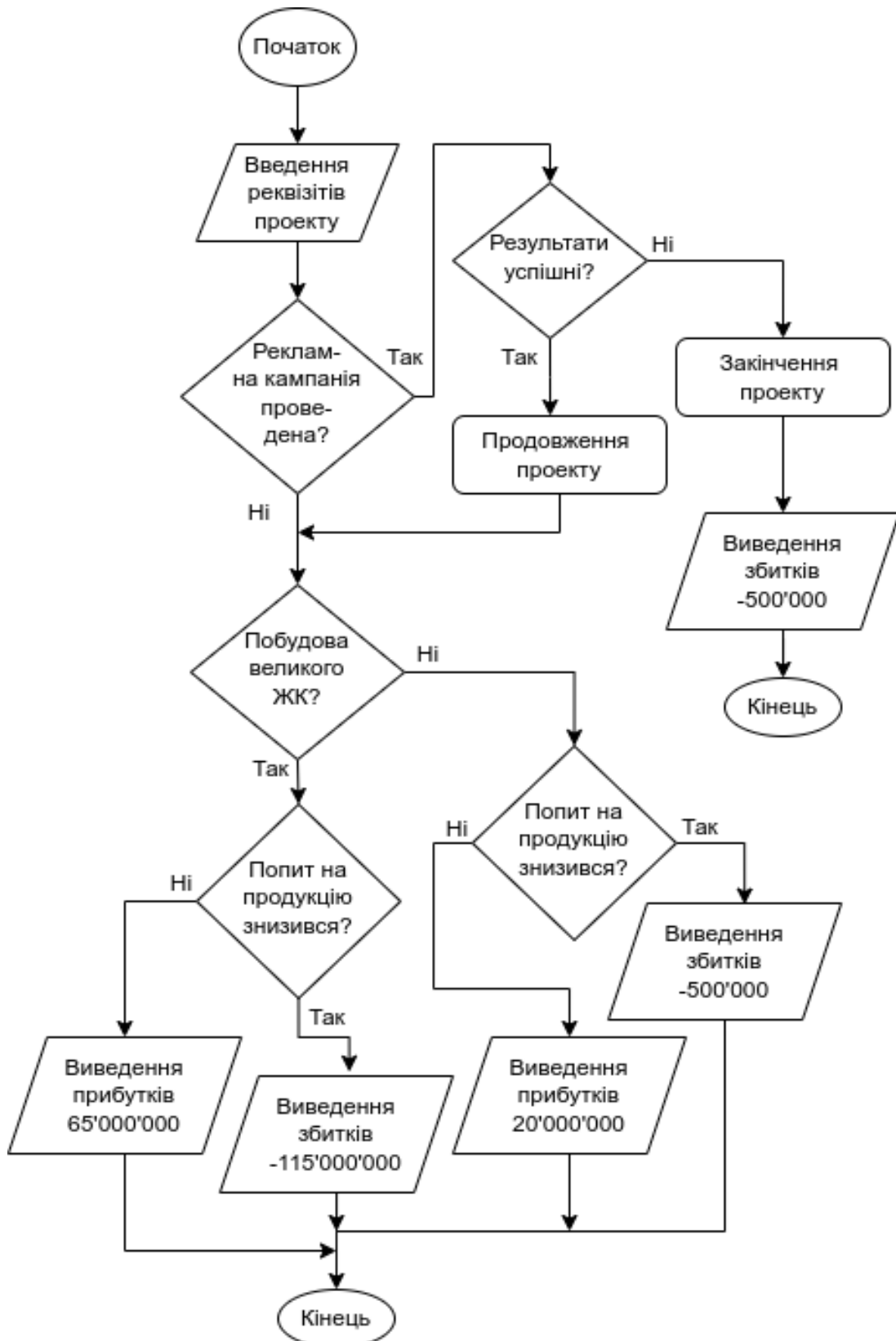


Рисунок 4.3 – Загальний алгоритм роботи програмного засобу

Розрахунки на основі яких виконується програмний засіб здійснюється за допомогою аналізу який отримала від аналітиків.

4.6 Реалізація контрольного прикладу рішення задачі

Контрольний приклад рішення задачі виконана на мові високого рівня Java. Розглянемо програмний код цього прикладу. На початку програмного коду описуємо стандартну графічну бібліотеку яка буде використовуватись:

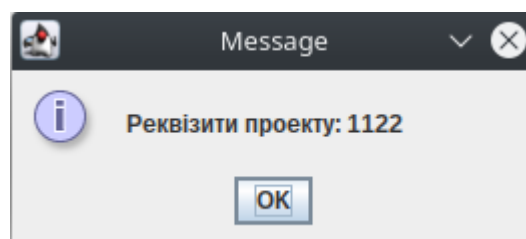
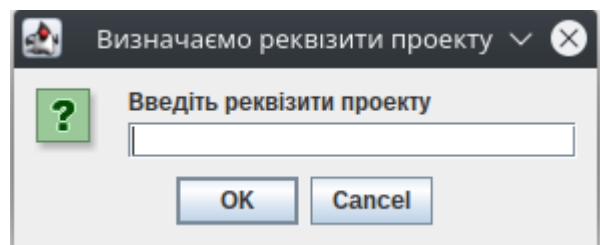
```
import static javax.swing.JOptionPane.*;
```

Потім описуємо назву класу, назву основного методу та змінні які будуть використовуватись подалі у програмного засобу:

```
class CovidFinance {  
    public static void main(String[] args) {  
        String recviziti, sreclama, sjk, spopit;  
        int ireclama, ijk, ipopit;
```

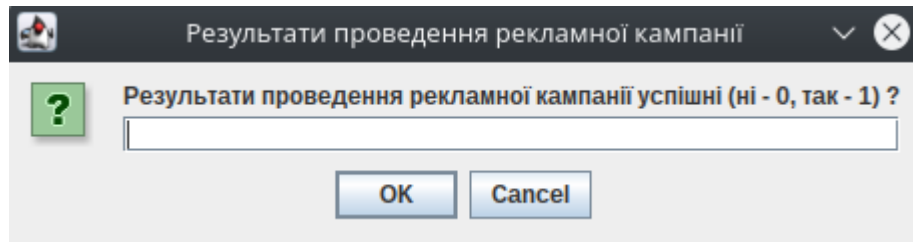
Далі виводимо графічне діалогове вікно для заповнення реквізитів проекту:

```
recviziti = showInputDialog(null, "Введіть реквізити проекту",  
    "Визначаємо реквізити проекту",  
    QUESTION_MESSAGE);  
showMessageDialog(null, "Реквізити проекту: " + recviziti);
```



Потім виводимо графічне діалогове вікно для визначення успішності результатів проведення рекламної кампанії:

```
sreclama = showInputDialog(null, "Результати проведення рекламної кампанії  
успішні (ні - 0, так - 1) ?",  
"Результати проведення рекламної кампанії",  
QUESTION_MESSAGE);
```

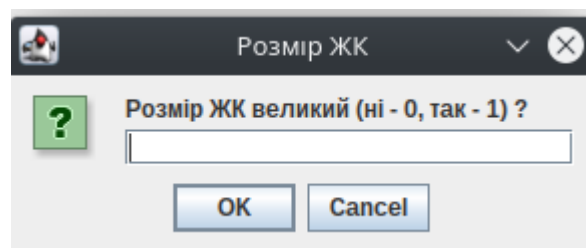


Потім визначаємо збитки за допомогою методу `lossesDefine` та параметру 1:

```
ireclama = Integer.parseInt(sreclama);  
if (ireclama == 0) {  
    // Рекламна кампанія відсутня, спостерігаються збитки  
    lossesDefine(1);  
    System.exit(0);  
}
```

Якщо збитки існують то завершимо виконання програми. Далі визначаємо розмір житлового кошторису:

```
sjk = showInputDialog(null, "Розмір ЖК великий (ні - 0, так - 1) ?",  
"Розмір ЖК",  
QUESTION_MESSAGE);  
ijk = Integer.parseInt(sjk);
```



Потім визначаємо попит на реалізовану продукцію:

```
sropit = showInputDialog(null, "Попит низився (ні - 1, так - 0) ?",
```

```

    "Попит на продукцію",
    QUESTION_MESSAGE);
ipopit = Integer.parseInt(spopit);

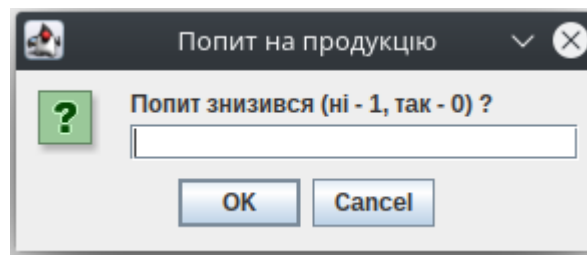
```

Якщо невеликий розмір житлового кошторису то викликаємо метод lossesDefine підрахунку збитків з параметром 2:

```

if (ijk == 0) {
    if (ipopit == 1)
        // Якщо попит знизився
        lossesDefine(2);
        // Якщо попит не змінився
    else
        profitDefine(2);
}

```



Якщо житловий кошторис великий, то викликаємо метод lossesDefine для підрахунку збитків з параметром 3:

```

else {
    if (ipopit == 1)
        // Якщо попит знизився
        lossesDefine(3);
        // Якщо попит не змінився
    else
        profitDefine(3);
}
}

```

Далі опишемо метод з визначення збитків:

```

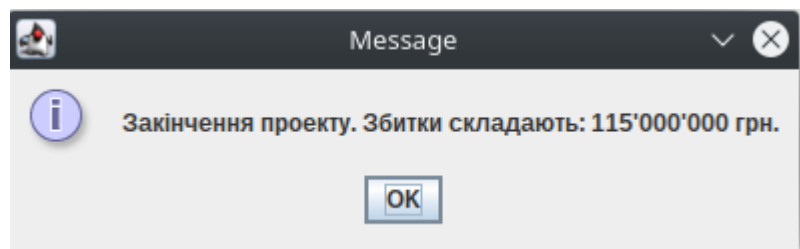
public static void lossesDefine(int param1) {

```

```

if (param1 == 1)
    showMessageDialog(null, "Закінчення проекту. Збитки складають: 500'000
грн.");
else
    if (param1 == 2)
        showMessageDialog(null, "Закінчення проекту. Збитки складають:
500'000 грн.");
    else
        if (param1 == 3)
            showMessageDialog(null, "Закінчення проекту. Збитки складають:
115'000'000 грн.");
}

```



Потім описуємо метод для визначення прибутків:

```

public static void profitDefine(int param1) {
    if (param1 == 2)
        showMessageDialog(null, "Закінчення проекту. Прибутки складають:
20'000'000 грн.");
    else
        if (param1 == 3)
            showMessageDialog(null, "Закінчення проекту. Прибутки складають:
65'000'000 грн.");
}
}

```

Таким чином виконується реалізація контрольної прикладу рішення задачі.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Оцінювання комерційного потенціалу розробки

Метою проведення комерційного та технологічного аудиту є оцінювання комерційного потенціалу впровадження методу та програмних засобів фінансування досліджень поширення COVID-19.

Для проведення технологічного аудиту було залучено 3-х незалежних експертів Вінницького національного технічного університету кафедри програмного забезпечення: Хошаба О. М., Котельніков А. П., Майданюк В. П. Для проведення технологічного аудиту було використано таблицю 5.1 [42] в якій за п'ятибальною шкалою використовуючи 12 критеріїв здійснено оцінку комерційного потенціалу.

Таблиця 5.1 – Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
ри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
	Експлу	Експлу	Експлуа	Експлуа	Експ

	атаційні витрати значно вищі, ніж в аналогів	атаційні витрати дещо вищі, ніж в аналогів	атаційні витрати на рівні експлуатаційних витрат аналогів	атаційні витрати трохи нижчі, ніж в аналогів	атаційні витрати значно нижчі, ніж в аналогів
--	--	--	---	--	---

Продовження табл. 5.1

Ринкові перспективи					
	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
0	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві

1	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
2	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Таблиця 5.2 – Рівні комерційного потенціалу розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0-10	Низький
11-20	Нижче середнього
21-30	Середній
31-40	Вище середнього
41-48	Високий

В таблиці 5.3 наведено результати оцінювання експертами комерційного потенціалу розробки.

Таблиця 5.3 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	Хошаба О.М.	Котельніков А.П.	Майданюк В.П.
	Бали, виставлені експертами:		
1	3	2	2
2	2	2	2
3	3	4	3
4	4	4	3
5	3	3	3
6	4	4	4
7	3	3	3
8	1	1	2
9	1	1	1
10	2	2	2
11	4	4	4
12	2	2	3
Сума балів	СБ ₁ =32	СБ ₂ =32	СБ ₃ =32
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = \frac{32 + 32 + 32}{3} = 32$		

Середньоарифметична сума балів, розрахована на основі висновків експертів склала 32 бали, що згідно таблиці 5.2 вважається, що рівень комерційного потенціалу проведених досліджень є вище середнього.

Програмний продукт, який розробляється в магістерській роботі буде цікавий інвесторам для розрахунку ризиків пов'язаних з будівництвом приміщень для хворих ковідом.

5.2 Прогнозування витрат на виконання науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи групуються за такими статтями: витрати на оплату праці, витрати на соціальні заходи, матеріали, паливо та енергія для науково-виробничих цілей, витрати на службові відрядження, програмне забезпечення для наукових робіт, інші витрати, накладні витрати.

1. Основна заробітна плата кожного із дослідників Z_0 , якщо вони працюють в наукових установах бюджетної сфери визначається за формулою:

$$Z_0 = \frac{M}{T_p} * t \text{ (грн)} \quad (5.1)$$

де M – місячний посадовий оклад конкретного розробника (інженера, дослідника, науковця тощо), грн.;

T_p – число робочих днів в місяці; приблизно $T_p \approx 21...23$ дні;

t – число робочих днів роботи дослідника.

Для розробки програмні засоби необхідно залучити програміста з посадовим окладом 10000 грн. Кількість робочих днів у місяці складає 40, а кількість робочих днів програміста складає 22. Зведемо сумарні розрахунки до таблиця 5.4.

Таблиця 5.4 – Заробітна плата дослідника в науковій установі бюджетної сфери

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату грн.
Керівник	14000	636,4	5	3182
Програмний інженер	10000	454,5	40	18182
UX та UI дизайнер	12000	545,5	20	10909
Всього				32273

2. Розрахунок додаткової заробітної плати робітників

Додаткова заробітна плата Z_d всіх розробників та робітників, які приймали участь в розробці нового технічного рішення розраховується як 10 - 12 % від основної заробітної плати робітників.

На даному підприємстві додаткова заробітна плата начисляється в розмірі 10% від основної заробітної плати.

$$Z_d = (Z_o + Z_p) * \frac{H_{\text{доп}}}{100\%} \quad (5.2)$$

$$Z_d = 0,11 * 32273 = 3550 \text{ (грн)}$$

3. Нарахування на заробітну плату $H_{3П}$ дослідників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою (4.3):

$$H_{3П} = (Z_o + Z_d) * \frac{\beta}{100} \text{ (грн)} \quad (5.3)$$

де Z_o – основна заробітна плата розробників, грн.;

Z_d – додаткова заробітна плата всіх розробників та робітників, грн.;

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування, % .

Дана діяльність відноситься до бюджетної сфери, тому ставка єдиного внеску на загальнообов'язкове державне соціальне страхування буде складати 22%, тоді:

$$H_{3П} = (32273 + 3550) * \frac{22}{100} = 7881 \text{ (грн)}$$

4. Витрати на комплектуючі вироби, які використовують при виготовленні одиниці продукції, розраховуються, згідно їх номенклатури, за формулою:

$$K = \sum_{i=1}^n H_i \cdot C_i \cdot K_i, \quad (5.5)$$

де H_i – кількість комплектуючих i -го виду, шт.;

C_i – покупна ціна комплектуючих i -го найменування, грн.;

K_i – коефіцієнт транспортних витрат (1,1...1,15).

Таблиця 5.5 – Комплектуючі, що використані на розробку

Найменува	Ціна за	Витраче	Вартість
-----------	---------	---------	----------

ння матеріалу	одиницю, грн.	но	витраченого матеріалу, грн.
Папір	140	1	140
Ручка	15	1	15
CD-диск	13	1	13
Флешка	145	1	145
Всього			313
З врахуванням коефіцієнта транспортування			344,3

5. Програмне забезпечення для наукової роботи включає витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення необхідного для проведення дослідження.

Балансову вартість програмного забезпечення розраховують за формулою:

$$B_{\text{прог}} = \sum_{i=1}^k C_{\text{инрг}} \cdot C_{\text{прог.і}} \cdot K_i, \quad (5.6)$$

де $C_{\text{инрг}}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{\text{прог.і}}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1, 10 \dots 1, 12$);

k – кількість найменувань програмних засобів.

Отримані результати необхідно звести до таблиці 5.6.

Таблиця 5.6 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
IntelliJ IDEA Ultimate	1	13 598	13598

AdobePhotoshop 2021	1	11 335	11335
Adobe illustrator 2021	1	12 866	12866
Всього			41578 ,9

6. Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час виконання даного етапу роботи

Дані відрахування розраховують по кожному виду обладнання, приміщенням тощо.

$$A = \frac{Ц \cdot T}{T_{кор} \cdot 12} \text{ [грн]}, \quad (5.7)$$

де Ц – балансова вартість даного виду обладнання (приміщень), грн.;

$T_{кор}$ – час користування;

T – термін використання обладнання (приміщень), цілі місяці.

Згідно пункту 137.3.3 Податкового кодекса амортизація нараховується на основні засоби вартістю понад 2500 грн. В нашому випадку для написання магістерської роботи використовувався персональний комп'ютер вартістю 24000 грн.

$$A = \frac{24000 \cdot 1}{2 \cdot 12} = 1000$$

7. До статті «Паливо та енергія для науково-виробничих цілей» відносяться витрати на всі види палива й енергії, що безпосередньо використовуються з технологічною метою на проведення досліджень.

$$B_e = \sum_{i=1}^n \frac{W_{yt} \cdot t_i \cdot C_e \cdot K_{впi}}{\eta_i} \quad (5.8)$$

де W_{yt} – встановлена потужність обладнання на певному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн;

$K_{впi}$ – коефіцієнт, що враховує використання потужності, $K_{впi} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

Для написання магістерської роботи використовується персональний комп'ютер для якого розрахуємо витрати на електроенергію.

$$B_e = \frac{0,3 \cdot 240 \cdot 4,1 \cdot 0,5}{0,8} = 184,5$$

Витрати на службові відрядження, витрати на роботи, які виконують сторонні підприємства, установи, організації та інші витрати в нашому дослідженні не враховуються оскільки їх не було.

Накладні (загальновиробничі) витрати $B_{нзв}$ охоплюють: витрати на управління організацією, оплата службових відряджень, витрати на утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Накладні (загальновиробничі) витрати $B_{нзв}$ можна прийняти як (100...150)% від суми основної заробітної плати розробників та робітників, які виконували дану МКНР, тобто:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (5.9)$$

де $H_{нзв}$ – норма нарахування за статтею «Інші витрати».

$$B_{нзв} = 32273 \cdot \frac{100}{100\%} = 32273 \text{ грн}$$

Сума всіх попередніх статей витрат дає витрати, які безпосередньо стосуються даного розділу МКНР

$$B = 322273 + 3550 + 7881 + 344,3 + 41578,9 + 1000 + 184,5 + 322273 = 119084,2$$

Прогнозування загальних втрат ZB на виконання та впровадження результатів виконаної МКНР здійснюється за формулою:

$$ZB = \frac{B}{\eta}, \quad (5.10)$$

де η – коефіцієнт, який характеризує стадію виконання даної НДР.

Оскільки, робота знаходиться на стадії науково-дослідних робіт, то коефіцієнт $\beta = 0,9$.

Звідси:

$$ZB = \frac{119084,2}{0,9} = 132315,73 \text{ грн.}$$

5.3 Розрахунок економічної ефективності науково-технічної розробки

У даному підрозділі кількісно спрогнозуємо, яку вигоду, зиск можна отримати у майбутньому від впровадження результатів виконаної наукової роботи. Розрахуємо збільшення чистого прибутку підприємства $\Delta\Pi_i$, для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, за формулою

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_o \cdot N + \Pi_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\nu}{100}\right) \quad (5.11)$$

де $\Delta\Pi_o$ – покращення основного оціночного показника від впровадження результатів розробки у даному році.

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки:

Π_o – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки:

l – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $l = 0,8333$.

ρ – коефіцієнт, який враховує рентабельність продукту. $\rho = 0,25$;

x – ставка податку на прибуток. У 2021 році – 18%.

Припустимо, що при впровадженні результатів наукової розробки інвестор зможе оцінити ризики щодо вкладання інвестицій в приміщення для ковідних цілей. Припустимо, що ціна від зросте на 1000 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року на 25 шт., протягом другого року – на 15 шт., протягом третього року на 10 шт. Реалізація

продукції до впровадження розробки складала 1 шт., а її ціна до складає 80000 грн. Розрахуємо прибуток, яке отримає підприємство протягом трьох років.

$$\begin{aligned}\Delta\Pi_1 &= [1000 \cdot 1 + (80000 + 1000) \cdot 25] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 346094,49 \text{ грн.}\end{aligned}$$

$$\begin{aligned}\Delta\Pi_2 &= [1000 \cdot 1 + (80000 + 1000) \cdot (25 + 15)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 554477,86 \text{ грн.}\end{aligned}$$

$$\begin{aligned}\Delta\Pi_3 &= [1000 \cdot 1 + (80000 + 1000) \cdot (25 + 15 + 10)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 692847,33 \text{ грн.}\end{aligned}$$

5.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Розрахуємо основні показники, які визначають доцільність фінансування наукової розробки певним інвестором, є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

Розрахуємо величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки.

$$PV = k_{\text{інв}} \cdot 3B, \quad (5.12)$$

$k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо ($k_{\text{інв}} = 2 \dots 5$).

$$PV = 2 \cdot 132315,73 = 264631,45$$

Розрахуємо абсолютну ефективність вкладених інвестицій $E_{\text{абс}}$ згідно наступної формули:

$$E_{\text{абс}} = (ПП - PV) \quad (5.13)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн.;

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (5.14)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн.;

T – період часу, протягом якою виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,2;

t – період часу (в роках).

$$ПП = \frac{346094,49}{(1+0,2)^1} + \frac{554477,86}{(1+0,2)^2} + \frac{692847,33}{(1+0,2)^3} = 1076284,36 \text{ грн.}$$

$$E_{abc} = (1076284,36 - 264631,45) = 811652,9 \text{ грн.}$$

Оскільки $E_{abc} > 0$ то вкладання коштів на виконання та впровадження результатів НДДКР може бути доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_e . Для цього користуються формулою:

$$E_e = \sqrt[T_{жс}]{\left(1 + \frac{E_{abc}}{PV}\right)} - 1, \quad (5.15)$$

$T_{жс}$ – життєвий цикл наукової розробки, роки.

$$E_e = \sqrt[3]{1 + \frac{811652,9}{264631,45}} - 1 = 0,93 = 93\%$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (5.16)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2021 році в Україні $d = (0,14 \dots 0,2)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05 \dots 0,1)$.

$$\tau_{\min} = 0,18 + 0,05 = 0,23$$

Так як $E_g > \tau_{\min}$ то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_g} \quad (5.17)$$

$$T_{ок} = \frac{1}{0,93} = 1,1 \text{ роки}$$

Так як $T_{ок} \leq 3 \dots 5$ -ти років, то фінансування даної наукової розробки в принципі є доцільним.

5.5 Висновки до економічного розділу

Було проведено оцінку комерційного потенціалу програмного засобу для оцінки ризиків щодо фінансування досліджень поширення COVID-19, який є на вище середньому рівні.

Прогнозування витрат на виконання науково-дослідної роботи по кожній з статей витрат складе 119084,2 грн. Загальна ж величина витрат на виконання та впровадження результатів даної НДР буде складати 132315,73 грн.

Вкладені інвестиції в даний проект окупляться через 1,1 роки при прогнозованому прибутку 1076284,36 грн. за три роки.

ВИСНОВКИ

У першому розділі магістерській кваліфікаційній роботі проведено аналіз сучасного стану методів на основі дерева рішень, де визначені основні напрямки в дослідженні процесу. Зроблено наголос на таких напрямках як нормативне та дескриптивне. У першому випадку дослідники будують теорію на вивченні формалізованих методів, переважно математичних моделей. У другому випадку дослідники будують свої положення на основі вивчення психології процесу прийняття рішень, поведінки особи, що приймає рішення.

Методи нормативного підходу на основі дерева рішень дозволяють оцінювати проблемну ситуацію, враховувати ресурси та обмеження, аналізувати різні варіанти дій, вибирати з них оптимальні, використовуючи формальні, як правило, математичні методи. Знання дескриптивної (або, як ще її називають, психологічної) теорії прийняття рішень дозволяє краще зрозуміти, як люди роблять свій вибір, які об'єктивні і суб'єктивні чинники впливають на правильність прийнятого ними рішення, які психологічні механізми керують процесом прийняття рішень, яка роль соціального середовища, в якій діє керівник, які механізми групової поведінки.

Обидва підходи до прийняття рішень на основі дерева рішень - і нормативний, і дескриптивний є необхідними і доповнюють один одного.

В цьому ж розділі показані загальні положення теорії дерева рішень, відомі обмеження у використанні теорії дерева рішень. Розглянута класифікація задач, які використовують теорію дерева рішень.

Системний аналіз процесу дерева рішень показав, що при вирішенні унікальних задач на основі дерева рішень досить часто акценти зміщуються в цей бік. При цьому, ефективність рішення в основному визначається вдалою декомпозицією вихідної задачі на ряд задач, що мають аналоги в класах специфічних і типових задач.

Формалізація цілі у методах дерева рішень показало, що у деяких практичних задачах прийняття рішення формалізація цілі може бути природно зведена до побудови деякої функції, заданої на множині альтернатив

(результатів) і приймаючої дійсні значення, а рішення задачі - до екстремізації даної функції.

В цьому ж розділі показано доцільне використання формальних методів дерева рішень.

В другому розділі розглянуті моделі прийняття рішень на основі теорії дерева рішень, де визначались основні етапи типового процесу прийняття рішення. Моделі прийняття рішень в умовах нелінійних розподілених задачах були визначені на основі задачі пошуку екстремуму (максимуму або мінімуму) деякої функції при наявності обмежень.

Для моделей алгоритму прямого і зворотного прогону в задачах упорядкування були розглянуті дві обчислювальні схеми: для алгоритму прямого і зворотного прогону. В цьому ж розділі були розглянуті використання методу еволюційних алгоритмів та визначення показників ефективності евристичних алгоритмів.

В третьому розділі були розглянуті моделі прийняття рішень на основі теорії ігор, де визначалась формалізація моделі гри, а використання основної теореми антагоністичних ігор двох осіб була показана як кінцева антагоністична гра двох осіб з нульовою сумою. Також було показано, що використання методу лінійного програмування доцільно проводити в процесі збору та аналізу даних, та під час побудови моделі у вигляді матриці виграшів проводяться вимірювання.

В цьому ж розділі розглядались питання використання моделі багатокритеріальних задач, прийняття рішень на основі системного підходу, математичний підхід до прийняття рішень на основі теорії ігор.

В четвертому розділі виконувалась розробка та впровадження програмного засобу з фінансування досліджень, де була визначена постановка задачі до розробки програмного засобу на основі методу дерева у прийнятті рішень, побудовано дерево рішень для задачі з реалізації конкретних проектів з визначенням певних ймовірностей. Подані основні етапи проектування програмного засобу на основі методу дерева рішень, що необхідні під час вирішення прикладних задач. Було виконано вибір математичних моделей

дерева графу, розглянуто використання різних математичних моделей у вигляді графів.

Розроблений програмний засіб з рішення задачі в галузі проведення досліджень з фінансування досліджень поширення COVID-19. Реалізація контрольного прикладу рішення задачі виконано на мові високого рівня Java, який реалізує складені вимоги, що розроблені у постановці задачі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Волошин О.Ф., Мащенко С.О. Теорія прийняття рішень : навч. посібн. – К., 2006.
2. Катренко А.В., Пасічник В.А., Пасько В.П. Теорія прийняття рішень. – К., 2009.
3. Arrow K. Social Choice and Individual Values. – N.-York, 1951. 52. Nash J.F. Equilibrium Points in n-Person Games // Proceedings of National Academia of Science (US). – 1951. – N 36.
4. Voloshyn A. Decision-Making Support Systems as Personal Intellectual Device of a Decision-Maker // Internation Journal "Information: Technologies&Knowledge". – 2007. – Vol. 1. – № 2. – P. 159–162.
5. Voloshyn O., Antosiak P. Procedures of Syquential Analyses and Sifting of Variants for the Linear Ordering Problem // International Book Series "Information Science & Computing". – 2009. – № 15. – P. 149-154.
6. Voloshyn O., Berezovskiyy K. Developing collective theaching computer software for the course "Decision theory" // International Journal "Information Technologies & Knowledge". – 2007. – Vol. 1. – P. 33-36.
7. Mas-Collel A., Whinston M.D., Green J.R. Microeconomic theory. N.Y.: Oxford Univ. Press, 1995. – 981 p.
8. Дмитриенко В. Д. Введение в теорию и методы принятия решений. Учеб. пособие/ В.Д. Дмитриенко, В.А. Кравец, С.Ю. Леонов. – Нац. техн. ун-т «Харьк. политехн. ин-т». – Х., 2008. – 141 с.
9. Катренко А.В. Теорія прийняття рішень. Підручник/ А.В. Катренко, В.В. Пасічник, В.П. Пасько. – К.: ВНУ, 2009. – 448с. –
10. Кушлик-Дивульська О.І. Теорія ймовірностей та математична статистика: [навч. посіб.] / О І. Кушлик-Дивульська, Н.В. Поліщук, Б П. Орел, П.І. Штабальук.– Вид. 2-ге, випр. і доп. – Київ, НТУУ «КПІ», 2012. – 220 с.

11. Jehiel P., Moldovanu B., Staccheti E. How (not) to Sell Nuclear Weapons // The American Economic Review. 1996.V. 86. № 4. P. 814–829.
12. Novikov D.A. Management of active systems: stability or efficiency //Systems science. 2001. Vol. 26. № 2. P.85-93.
13. Dixit A., Nalebuff B. Thinking Strategically: The Competitive Edge in Business, Politics and Everyday Life. – N.Y.: Norton, 1991. – 394 с.
14. Gibbons R. Game Theory for Applied Economists. – Princeton: Princeton University Press, 1992. – 268 p.
15. Шиян А.А. Економічна кібернетика: вступ до моделювання соціальних і економічних систем: Навчальний посібник. – Львів: «Магнолія 2006». – 2007. – 228 с.
16. Acemoglu, D., Robinson, J. A., Economic Origins of Dictatorship and Democracy. - Cambridge. - Cambridge University Press, Cambridge. - 2006. – 416 p.
17. Acemoglu, D. Introduction to Modern Economic Growth. – Princeton: Princeton University Press, 2009. – 1072 p.
18. Persson T., Tabellini G. Political Economics: Explaining Economic Policy. - Cambridge, MA: MIT Press, 2000. – 533 p.
19. Bolton P. Dewatripont M. Contract Theory. – Cambridge: MIT Press, 2005. – 724 p.
20. Grossman G.M., Helpman E. Special Interest Politics. — Cambridge, MA: MIT Press. – 2002. — 380 p.
21. Геець В.М. Інституційні перетворення і суспільний розвиток // Економіка і прогнозування. - 2005. - №2.- С. 9-36.
22. Катренко, А. В. Дослідження операцій [Текст]: підруч. / А. В. Катренко. – Л : «Магнолія – 2006», 2009. – 352 с.
23. Оптимізаційні методи та моделі : підручник / В.С. Григорків, М.В. Григорків. – Чернівці: Чернівецький нац. ун-т, 2016. – 400 с.
24. Файнзильберг Л.С. Обучаемая система поддержки принятия коллективного решения группы независимых экспертов // Управляющие системы и машины. – 2003. – № 4. – С. 62-67.

25. Жуковская О.А., Файнзильберг Л.С. Интервальное обобщение байесовской модели принятия коллективного решения в конфликтных ситуациях // Кибернетика и системный анализ. – 2005. – № 3. – С. 133–144.

26. Синеглазов В. М. Математичні методи оптимізації: навч. посібн./ В.М. Синеглазов, О. А. Зеленков, Ш. І. Аскеров. – Нац. Авіаційний ун-т. – К.: Освіта України, 2018. – Ч. 1. – 329 с.

27. Мащенко С.О. Стабільні за перевагами рівноваги в одноцільових некооперативних іграх // Вісн. Київ. ун-ту. Серія: фіз.-мат. науки. – 2009. – № 3. – С. 119-124.

28. Теория игр. Искусство мышления в бизнесе и жизни / Авинаш Диксит и Барри Нейлбафф; пер. англ. Н. Яцюк.- М.: Манн, Иванов и Фербер, 2015.– 464 с.

29. Бех, О. В. Математичне програмування [Текст]: навч. посіб. / О. В. Бех., Т. А. Городня, А. Ф. Щербак. – Л.: «Магнолія – 2006», 2009. – 200 с.

30. Послайко Н. І. Дослідження операцій. Задачі з умовами невизначеності та конфлікту : навч. посіб. / Н. І. Послайко. – Дніпро : Стандарт - Сервіс, 2019. – 53 с.

31. Корнієнко В.О., Денисюк С.Г., Шиян А.А. Моделювання процесів у політико-комунікативному просторі: Монографія. — Вінниця: УНІВЕРСУМ-Вінниця, 2009. — 185 с.

32. Шиян А.А. Теоретико-ігровий аналіз раціональної поведінки людини та прийняття рішень в управлінні соціально-економічними системами. – Вінниця: УНІВЕРСУМ-Вінниця, 2009. – 414 с.

33. Перспективні освітні технології: наук.-метод. посіб./ А.М. Алексюк, І.Д. Бех, Т.Ф. Демків, І.Г.Єрмаков, О.Завадський; за заг. ред. Г.С. Сазоненко. К.: Гопак, 2000. – 560 с. (Сучасна освіта України).

34. Богданова І. М. Технології в освіті: теоретико-методологічний аспект. Акад. пед. наук України. Одеса, 1999. - 146 с.

35. Бурлаєнко Т.І. Формування економічної компетентності майбутніх менеджерів освіти засобами ігрових форм навчання: автореф. дис. ... канд. пед. наук: 13.00.04. / НАПН України, ДВЗО «Ун-т менедж. освіти». К., 2013. - 19 с.

36. Варій М. Й. Загальна психологія: підручник для студ. вищих навч. закладів. 3-є вид., випр. і доп. К.: Центр учбової літератури, 2009. - 1007 с.
37. Ігнатенко С. В. Формування фахових компетенцій майбутніх інженерів педагогів засобами проблемно-ігрового навчання: автореф. дис. ... канд. пед. наук: 13.00.04 / НАПН України, Ін-т вищої освіти. К., 2011. - 20 с.
38. Ігри дорослих. Інтерактивні методи навчання / упоряд. Л. Галіцина. К.: Ред. загальнопед. газета, 2005. - 128 с.
39. Інтерактивні технології навчання: теорія, практика, досвід: метод. посібник / авт.-укладачі: О. Пометун, Л. Пироженко. К.: АПН, 2002. - 135 с.
40. Бурков В.Н., Заложнев А.Ю., Новиков Д.А. Теория графов в управлении организационными системами. М.: Синтег, 2001. – 246с.
41. Ковальчук Д. О. Використання сучасних засобів з теорії управлінських рішень боротьби з пандемією COVID-19 //Міжнародна науково-практична Інтернет-конференція "Електронні інформаційні ресурси: створення, використання, доступ". Збірник матеріалів - Суми/Вінниця: НІКО/ВНТУ, 2021. – С. 81-84.
42. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

ДОДАТКИ

Додаток А
Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ
д.т.н., проф. О. Н. Романюк
" ____ " _____ 2021 р.

Технічне завдання
на магістерську кваліфікаційну роботу «Розробка методу та
програмних засобів фінансування досліджень поширення COVID-19» за
спеціальністю
121 – Інженерія програмного забезпечення

Керівник магістерської кваліфікаційної роботи:

_____ к.т.н., доцент О.М. Хошаба
" 13 " _____ вересня _____ 2021 р.

Виконав:

_____ студент гр.1ПІ-20м Д.О. Ковальчук
" 13 " _____ вересня _____ 2021 р.

1. Найменування та галузь застосування

Магістерська кваліфікаційна робота: «Розробка методу та програмних засобів фінансування досліджень поширення COVID-19».

Галузь застосування - системи фінансування наукових досліджень.

2. Підстава для розробки.

Підставою для виконання магістерської кваліфікаційної роботи (МКР) є індивідуальне завдання на МКР та наказ № ректора по ВНТУ про закріплення тем МКР.

3. Мета та призначення розробки.

Метою роботи є підвищення ефективності проведення фінансування досліджень поширення COVID-19 за рахунок використання методу дерева рішень.

Призначення роботи – розробка методів і засобів проведення фінансування досліджень поширення COVID-19.

3 Вихідні дані для проведення НДР

Перелік основних літературних джерел, на основі яких буде виконуватись МКР.

1. Ковальчук Д. О. Використання сучасних засобів з теорії управлінських рішень боротьби з пандемією COVID-19 //Міжнародна науково-практична Інтернет-конференція "Електронні інформаційні ресурси: створення, використання, доступ". Збірник матеріалів - Суми/Вінниця: НІКО/ВНТУ, 2021. – С. 81-84.
2. Оптимізаційні методи та моделі : підручник / В.С. Григорків, М.В. Григорків. – Чернівці: Чернівецький нац. ун-т, 2016. – 400 с.
3. Шиян А.А. Теоретико-ігровий аналіз раціональної поведінки людини та прийняття рішень в управлінні соціально-економічними системами. – Вінниця: УНІВЕРСУМ-Вінниця, 2009. – 414 с.
4. Синєглазов В. М. Математичні методи оптимізації: навч. посібн./ В.М. Синєглазов, О. А. Зеленков, Ш. І. Аскеров. – Нац. Авіаційний ун-т. – К.: Освіта України, 2018. – Ч. 1. – 329 с.
5. Послайко Н. І. Дослідження операцій. Задачі з умовами невизначеності та конфлікту : навч. посіб. / Н. І. Послайко. – Дніпро : Стандарт - Сервіс, 2019. – 53 с.

4. Технічні вимоги

Системний аналіз процесу дерева рішень, формальні методи дерева рішень, основні етапи типового процесу прийняття рішення - не менше трьох етапів, моделі прийняття рішень - для умов нелінійних розподілених задач, моделі алгоритму - прямий та зворотній прогон для задач упорядкування, методи еволюційних та евристичних алгоритмів, основна теорема антагоністичних ігор - дві особи, математична модель дерева графу - на конкретному прикладі фінансування досліджень.

5. Конструктивні вимоги.

Конструкція пристрою повинна відповідати естетичним та ергономічним вимогам, повинна бути зручною в обслуговуванні та керуванні.

Графічна та текстова документація повинна відповідати діючим стандартам України.

6. Перелік технічної документації, що пред'являється по закінченню робіт:

- пояснювальна записка до МКР;
- технічне завдання;
- лістинги програми.

7. Вимоги до рівня уніфікації та стандартизації

При розробці програмних засобів слід дотримуватися уніфікації і ДСТУ.

8. Стадії та етапи розробки:

№/п	Назва етапів магістерської кваліфікаційної Роботи	Строк виконання етапів роботи
1	Аналіз сучасного стану методів та моделей на основі дерева рішень	31.10.2021
2	Розробка моделей прийняття рішень на основі теорії дерева рішень	07.11.2021
3	Розробка методів та моделей на основі теорії ігор	10.11.2021
4	Розробка блок-схеми алгоритму та програмного засобу	14.11.2021
5	Економічна частина	30.11.2021

9. Порядок контролю та прийняття.

Виконання етапів магістерської кваліфікаційної роботи контролюється керівником згідно з графіком виконання роботи. Прийняття магістерської кваліфікаційної роботи здійснюється ДЕК, затвердженою зав. кафедрою згідно з графіком

ДОДАТОК Б

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ)
РОБОТИ

Назва роботи: **Розробка методу та програмних засобів фінансування досліджень поширення COVID-19.**

Тип роботи: кваліфікаційна робота

Підрозділ : кафедра програмного забезпечення, ФІТКІ, 1ПІ – 20м

Науковий керівник: к.т.н. доц. Хошаба О. М.

Unicheck	
Оригінальність	94,0%
Схожість	6,0 %

Аналіз звіту подібності

■ **Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.**

Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.

Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомена з повним звітом подібності, який був згенерований Системою щодо роботи «Розробка методу та програмних засобів фінансування досліджень поширення COVID-19».

Автор _____

Ковальчук Дмитро Олександрович

Опис прийнятого рішення: **допустити до захисту**

Особа, відповідальна за перевірку
(підпис) (прізвище, ініціали)

Черноволик Г. О.

Експерт _____

(за потреби)

(підпис)

_____ (прізвище, ініціали, посада)

ДОДАТОК В

Лістинг коду

Лістинг програмного засобу

```
// Контрольний приклад рішення задачі
// згідно алгоритму на рис. 4.3
//
import static javax.swing.JOptionPane.*;

class CovidFinance {
    public static void main(String[] args) {
        String recviziti, sreclama, sjk, spopit;
        int ireclama, ijk, ipopit;

        recviziti = showInputDialog(null, "Введіть реквізити проекту",
            "Визначаємо реквізити проекту",
            QUESTION_MESSAGE);
        showMessageDialog(null, "Реквізити проекту: " + recviziti);

        sreclama = showInputDialog(null, "Результати проведення рекламної кампанії
        успішні (ні - 0, так - 1) ?",
            "Результати проведення рекламної кампанії",
            QUESTION_MESSAGE);

        ireclama = Integer.parseInt(sreclama);
        if (ireclama == 0) {
            // Рекламна кампанія відсутня, спостерігаються збитки
            lossesDefine(1);
            System.exit(0);
        }
    }
}
```

```
// Визначення розміру ЖК
sjk = showInputDialog(null, "Розмір ЖК великий (ні - 0, так - 1) ?",
    "Розмір ЖК",
    QUESTION_MESSAGE);
ijk = Integer.parseInt(sjk);

// Визначення попиту на продукцію
spopit = showInputDialog(null, "Попит знизився (ні - 1, так - 0) ?",
    "Попит на продукцію",
    QUESTION_MESSAGE);
ipopit = Integer.parseInt(spopit);

// Невеликий ЖК
if (ijk == 0) {
    if (ipopit == 1)
        // Якщо попит знизився
        lossesDefine(2);
        // Якщо попит не змінився
    else
        profitDefine(2);
}

// Великий ЖК
else {
    if (ipopit == 1)
        // Якщо попит знизився
        lossesDefine(3);
        // Якщо попит не змінився
    else
        profitDefine(3);
}
}
```

```

public static void lossesDefine(int param1) {
// Використання "заплаток"
    if (param1 == 1)
        showMessageDialog(null, "Закінчення проекту. Збитки складають: 500'000
грн.");
    else
        if (param1 == 2)
            showMessageDialog(null, "Закінчення проекту. Збитки складають: 500'000
грн.");
        else
            if (param1 == 3)
                showMessageDialog(null, "Закінчення проекту. Збитки складають:
115'000'000 грн.");
    }
public static void profitDefine(int param1) {
// Використання "заплаток"
    if (param1 == 2)
        showMessageDialog(null, "Закінчення проекту. Прибутки складають:
20'000'000 грн.");
    else
        if (param1 == 3)
            showMessageDialog(null, "Закінчення проекту. Прибутки складають:
65'000'000 грн.");
    }
}
}

```

Лістинг основного класу

```

package org.vntu.covidfinance.database.accessingdatamysql;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;

```

```
import javax.persistence.GenerationType;
import javax.persistence.Id;
@Entity
public class CovidFinance {
    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    private Integer id;
    private String name;
    private String catalog;
    private String source;
    private String producer;
    private String category;
    private String releaseForm;
    public Integer getId() {
        return id;
    }
    public void setId(Integer id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getCatalog() {
        return catalog;
    }
    public void setCatalog(String catalog) {
        this.catalog = catalog;
    }
}
```

```
public String getSource() {
    return source;
}

public void setSource(String source) {
    this.source = source;
}

public String getProducer() {
    return producer;
}

public void setProducer(String producer) {
    this.producer = producer;
}

public String getCategory() {
    return category;
}

public void setCategory(String category) {
    this.category = category;
}

public String getReleaseForm() {
    return releaseForm;
}

public void setReleaseForm(String releaseForm) {
    this.releaseForm = releaseForm;
}
}
```

Лістинг класу контролера програмного засобу

```
package org.vntu.covidfinance.database.accessingdatamysql;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
```

```

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller // This means that this class is a Controller
@RequestMapping(path="/covidfinance") // This means URL's start with /demo
(after Application path)
public class MainController {
    @Autowired // This means to get the bean called userRepository
        // Which is auto-generated by Spring, we will use it to handle the data
    private CovidFinanceRepository medicinesRepository;
    @PostMapping(path="/add") // Map ONLY POST Requests
    public @ResponseBody String addNewCovidFinance (@RequestParam String
name, @RequestParam String catalog, @RequestParam String source,
@RequestParam String producer,
        @RequestParam String category, @RequestParam String releaseForm,
@RequestParam String dosage,
        @RequestParam String activeSubstance, @RequestParam String appointment) {
    // @ResponseBody means the returned String is the response, not a view name
    // @RequestParam means it is a parameter from the GET or POST request

    CovidFinance covidfinance = new CovidFinance();
    covidfinance.setName(name);
    covidfinance.setCatalog(catalog);
    covidfinance.setSource(source);
    covidfinance.setProducer(producer);
    covidfinance.setCategory(category);
    covidfinance.setReleaseForm(releaseForm);
    covidfinanceRepository.save(medicines);

```

```

    return "Збережено у базі даних!";
}
@GetMapping(path="/all")
public @ResponseBody Iterable<CovidFinance> getAllCovidFinance() {
    // This returns a JSON or XML with the users
    return covidfinanceRepository.findAll();
}
}

```

Лістинг класу репозиторію програмного засобу

```

package org.vntu.covidfinance.database.accessingdatamysql;
import org.springframework.data.repository.CrudRepository;
import org.vntu.covidfinance.database.accessingdatamysql.CovidFinance;

// This will be AUTO IMPLEMENTED by Spring into a Bean called userRepository
// CRUD refers Create, Read, Update, Delete

public interface CovidFinanceRepository extends CrudRepository<CovidFinance,
Integer> {

}

```

Лістинг класу програмного засобу по роботі з базою даних

```

package org.vntu.pharmacy.database.accessingdatamysql;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
@SpringBootApplication
public class AccessingDataMysqlApplication {
    public static void main(String[] args) {
        SpringApplication.run(AccessingDataMysqlApplication.class, args);
    }
}

```

```
}
```

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>JdbcPool</groupId>
  <artifactId>JdbcPool</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>
  <dependencies>
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>5.1.41</version>
    </dependency>
    <dependency>
      <groupId>commons-dbcp</groupId>
      <artifactId>commons-dbcp</artifactId>
      <version>1.4</version>
    </dependency>
  </dependencies>
</project>
```

```
package com.jcg.jdbc.connection.pooling;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.sql.DataSource;
```



```

import org.apache.commons.dbcp.ConnectionFactory;
import org.apache.commons.dbcp.DriverManagerConnectionFactory;
import org.apache.commons.dbcp.PoolableConnectionFactory;
import org.apache.commons.dbcp.PoolingDataSource;
import org.apache.commons.pool.impl.GenericObjectPool;
public class ConnectionPool {
    // Назва JDBC Driver Name & Database URL
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String JDBC_DB_URL = "jdbc:mysql://localhost:3306/covidFinance";
    // Назва JDBC Database Credentials
    static final String JDBC_USER = "root";
    static final String JDBC_PASS = "admin@123";
    private static GenericObjectPool gPool = null;
    @SuppressWarnings("unused")
    public DataSource setUpPool() throws Exception {
        Class.forName(JDBC_DRIVER);

        // Створення об'єкта GenericObjectPool
        gPool = new GenericObjectPool();
        gPool.setMaxActive(5);

        // Створення ConnectionFactory Object Which Will Be Use by the Pool to
        Create the Connection Object!
        ConnectionFactory cf = new
        DriverManagerConnectionFactory(JDBC_DB_URL, JDBC_USER, JDBC_PASS);

        // Створення ConnectionFactory to Add Object Pooling Functionality!
        PoolableConnectionFactory pcf = new PoolableConnectionFactory(cf, gPool,
        null, null, false, true);
        return new PoolingDataSource(gPool);
    }
}

```

```

public GenericObjectPool getConnectionPool() {
    return gPool;
}
// Використання методу
private void printDbStatus() {
    System.out.println("Max.: " + getConnectionPool().getMaxActive() + "; Active:
" + getConnectionPool().getNumActive() + "; Idle: " +
getConnectionPool().getNumIdle());
}
public static void main(String[] args) {
    ResultSet rsObj = null;
    Connection connObj = null;
    PreparedStatement pstmtObj = null;
    ConnectionPool jdbcObj = new ConnectionPool();
    try {
        DataSource dataSource = jdbcObj.setUpPool();
        jdbcObj.printDbStatus();

        System.out.println("\n=====Making A New Connection Object For Db
Transaction=====\n");
        connObj = dataSource.getConnection();
        jdbcObj.printDbStatus();

        pstmtObj = connObj.prepareStatement("SELECT * FROM
technical_editors");
        rsObj = pstmtObj.executeQuery();
        while (rsObj.next()) {
            System.out.println("Username: " + rsObj.getString("tech_username"));
        }
        System.out.println("\n=====Releasing Connection Object To
Pool=====\n");
    }
}

```

```
} catch(Exception sqlException) {
    sqlException.printStackTrace();
} finally {
    try {
        // Закрытие ResultSet Object
        if(rsObj != null) {
            rsObj.close();
        }
        // Закрытие PreparedStatement Object
        if(pstmtObj != null) {
            pstmtObj.close();
        }
        // Closing Connection Object
        if(connObj != null) {
            connObj.close();
        }
    } catch(Exception sqlException) {
        sqlException.printStackTrace();
    }
}
jdbcObj.printDbStatus();
}
```

Додаток Г

ІЛЮСТРАТИВНА ЧАСТИНА

**РОЗРОБКА МЕТОДУ ТА ПРОГРАМНИХ ЗАСОБІВ ФІНАНСУВАННЯ
ДОСЛІДЖЕНЬ ПОШИРЕННЯ COVID-19**

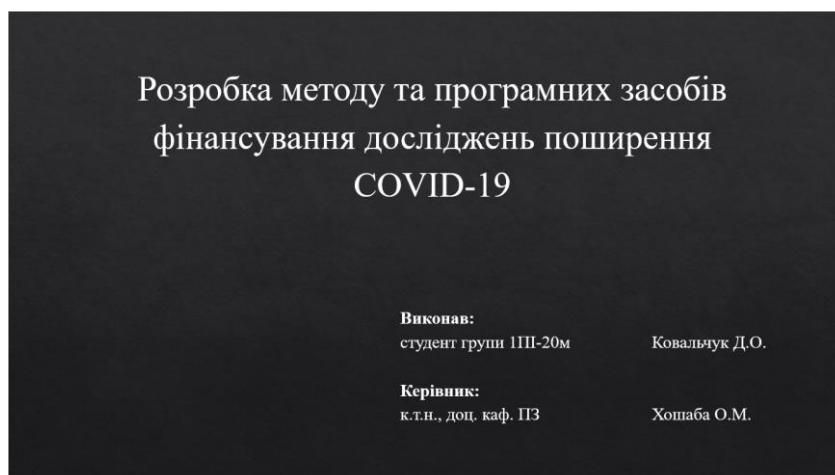


Рисунок Г.1 – Плакат 1

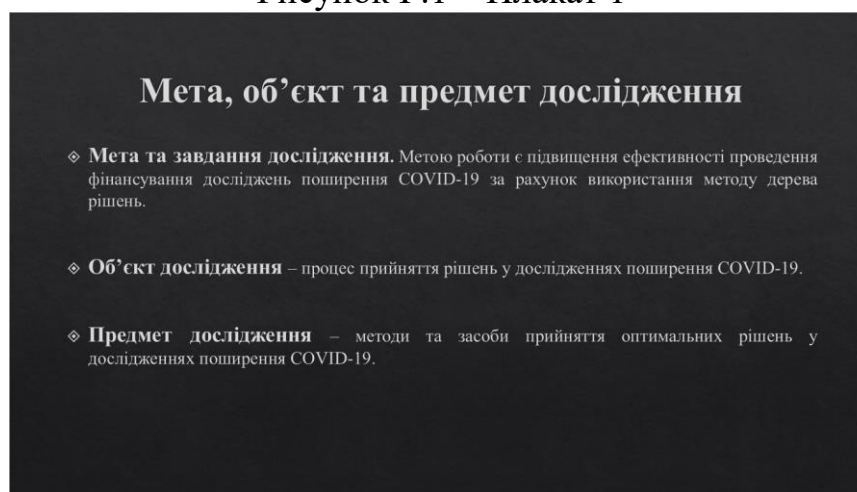


Рисунок Г.2 – Плакат 2

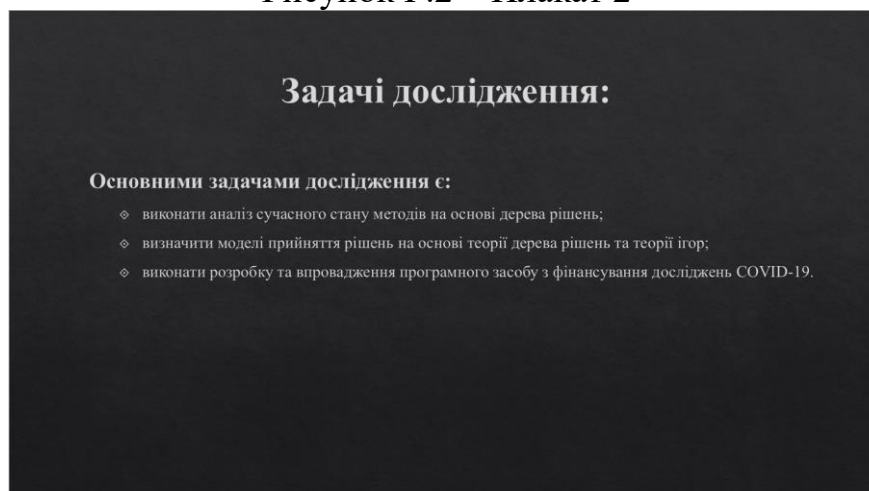


Рисунок Г.3 – Плакат 3

Актуальність розробки

Свідома діяльність людини нерозривно пов'язана з процесами прийняття рішень на основі дерева рішень. Так, люди завжди приймали рішення, ґрунтуючись на своєму досвіді, інтуїції та здоровому глузді. При цьому, як правило, точний шлях, що привів до вибору рішення, сам автор рішення писати не в змозі, хоча є підстави вважати, що він якимось чином враховував і зважував усі аспекти прийнятого рішення. Уміння приймати рішення, що дають найкращі результати у різних складних ситуаціях завжди розглядалося як мистецтво.

Саме тим часом публікуються дослідження математиків-економістів Курно, Квінсі, Вальраса. І трохи пізніше Парето, які започаткували теорію ігор; розробляються основи лінійного програмування Жорданом, Мінковським, Фаркашем; теорії прийняття рішень (ТПР) почалося з середини століття і призвело до утворення таких самостійних математичних дисциплін як теорія ігор, багатокритеріальний вибір і векторна оптимізація, математичне програмування. Причому остання дисципліна включає такі цілком самостійні розділи як лінійне програмування, опукле програмування, дискретне програмування, календарне планування, стохастичне програмування і т.д.

Але, в теперішній час, в зв'язку з погрозою розвитку пандемії COVID-19 досить актуальними стають питання боротьби на рівні організаційних заходів, а саме удосконалення методів прийняття рішень на основі теорії дерев, де важливими є фінансування досліджень та заохочення інвесторів за допомогою надання відомостей з результатів роботи математичних моделей. Саме цьому й присвячена робота.

Рисунок Г.4 – Плакат 4

Модель особи яка приймає рішення

$$\text{ОПР: } = \langle S_0, T, Q | S, A, B, Y, f, K, Y^* \rangle$$

S_0 – проблемна ситуація;

T – час для прийняття рішення;

Q – ресурси;

S – множина альтернативних ситуацій, що визначають ситуацію до повної групи взаємовиключних, незалежних або гіпотетичних ситуацій, причому сума ймовірностей їх виникнення дорівнює 1;

A – множина цілей,

B – множина обмежень при прийнятті рішень;

Y – множина альтернативних варіантів рішень, з яких має бути вибрано єдине оптимальне рішення Y^* .

Рисунок Г.5 – Плакат 5

Постановка задачі використання теорії дерева рішень

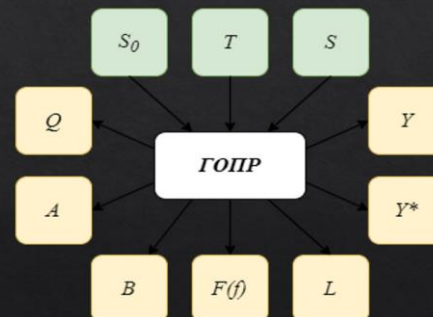


Рисунок Г.6 – Плакат 6



Рисунок Г.7 – Плакат 7



Рисунок Г.8 – Плакат 8

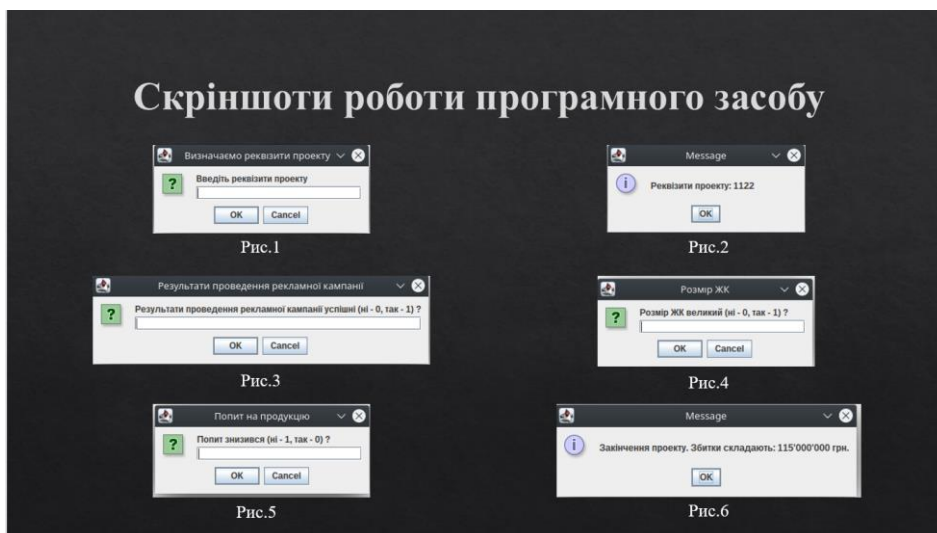


Рисунок Г.9 – Плакат 9

Економічний розділ

Було проведено оцінку комерційного потенціалу програмного засобу для оцінки ризиків щодо фінансування досліджень поширення COVID-19, який є на вище середньому рівні.

Прогнозування витрат на виконання науково-дослідної роботи по кожній з статей витрат складе 119084,2 грн. Загальна ж величина витрат на виконання та впровадження результатів даної НДР буде складати 132315,73 грн.

Вкладені інвестиції в даний проект окупляться через 1,1 роки при прогнозованому прибутку 1076284,36 грн. за три роки.

Рисунок Г.1 – Плакат 10

Висновки

Наукова новизна отриманих результатів.

- ◆ подальшого розвитку дістав метод прийняття рішень в умовах нелінійних розподілених задачах, який, на відміну від існуючих, придатний для проведення досліджень в галузі фінансування досліджень, що на певних рівнях дерева прибутків визначає також й збитки, і орієнтований на обчислення показників за допомогою імовірнісних моделей, що забезпечує можливість прийняття рішень відносно інвестування визначеного проекту;
- ◆ удосконалено методи прямого і зворотного прогону для задач упорядкування даними економічного характеру, який, на відміну від існуючих, дозволяють приймати рішення відносно обрання заданого проекту фінансування досліджень, що надає можливість зменшити інвестиційні збитки та підвищити ймовірність отримання прибутків;
- ◆ здобуло подальший розвиток метод еволюційних алгоритмів, який, на відміну від існуючих, на основі даних, які визначають сукупність гілок дерева рішень як можливі шляхи вирішення задачі задає спосіб кодування параметрів як «хромосоми», що дозволяє створити вихідну популяцію в якості початкового рішення та надає можливість розраховувати функцію пристосованості кожної особини, що позначає з великим ступенем ймовірності конкретну витрашну дію для прийняття рішення з фінансування проекту серед всіх можливих пропозицій.

Рисунок Г.1 – Плакат 11

Висновки

Практична цінність отриманих результатів. Практичне значення отриманих результатів полягає в тому, що на основі отриманих в магістерській кваліфікаційній роботі теоретичних положень запропоновано алгоритми та розроблено програмні засоби фінансування досліджень поширення COVID-19 на основі методу дерева.

Особистий внесок здобувача. У магістерській кваліфікаційній роботі усі результати дослідження здобуті автором даної роботи самостійно. У роботі, що опублікована самостійно, автору належить формалізація проблеми фінансування досліджень поширення COVID-19, розкриття мети роботи, аналізу та пропонування рішення проблеми з фінансування досліджень поширення COVID-19, розгляд контрольного прикладу з реалізації проектів фінансування досліджень поширення COVID-19.

Апробація матеріалів магістерської кваліфікаційної роботи. Основні положення магістерської кваліфікаційної роботи доповідалися та обговорювалися та обговорювалися на Міжнародній науково-практичній Інтернет конференції "Електронний інформаційні ресурси: створення, використання, доступ" (Суми/Вінниця, 2021).

Публікації. Основні результати досліджень опубліковано в одноосібній науковій праці у матеріалах конференції.

Рисунок Г.1 – Плакат 12