

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Програмної інженерії
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка
рівень вищої освіти – другий (магістерський)

Дослідження методів ідемпотентної математики для розробки програмних
середовищ управління проектами
(тема)

Виконала: студентка 2 курсу, групи ППЗМ-18-3
Бітюкова Є.І.
(прізвище, ініціали)

спеціальності 121 – Інженерія програмного забезпечення
(код і повна назва спеціальності)

Освітньо-наукової програми
(тип програми)

Інженерія програмного забезпечення
(повна назва освітньої програми)

Керівник доц., к.т.н Голян В.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф.

З.В.Дудар

2020 р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Комп'ютерних наук

Кафедра Програмної інженерії

Рівень вищої освіти – другий (магістерський)

Спеціальність 121 – Інженерія програмного забезпечення

(код і повна назва)

Тип програми освітньо-наукова програма

Освітня програма Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« ____ » _____ 20 ____ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові Бітюковій Єлизаветі Ігорівні

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів ідемпотентної математики для розробки програмних середовищ управління проектами
затверджена наказом університету від “ ____ ” _____ 20 ____ р. № _____

2. Термін подання студентом роботи до екзаменаційної комісії 22 травня 2020 р.

3. Вихідні дані до роботи алгоритми застосування методів ідемпотентної математики до вирішення задач планування проектів, пояснювальна записка. Використовувати ОС Windows, середовище планування проектів

4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз проблемної галузі і постановка задачі, огляд методів ідемпотентної математики, методи планування проектів, методи управління проектами, засоби відображення плану проекту

5 Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	доц., к.т.н. Голян В.В.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка *
1.	Аналіз предметної галузі	27 січня 2020 р.	виконано
2.	Огляд існуючих методів планування проектів	21 лютого 2020 р.	виконано
3.	Огляд методів та теорії ідемпотентної математики	10 березня 2020 р.	виконано
4.	Підготовка пояснювальної записки	25 березня 2020 р.	виконано
5.	Спецчастина	28 березня 2020 р.	виконано
6.	Підготовка презентації та доповіді	03 травня 2020 р.	виконано
7.	Попередній захист	08 травня 2020 р.	виконано
8.	Нормоконтроль, рецензування	12 травня 2020 р.	виконано
9.	Занесення диплома в електронний архів	16 травня 2020 р.	виконано
10.	Допуск до захисту у зав. кафедри	21 травня 2020 р.	виконано

Дата видачі завдання _____ 2020 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Голян В.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Атестаційна робота магістра містить: 76 с., 17 рис., 22 джерела.

ІДЕМПОТЕНТНА АЛГЕБРА, МЕТОДИ ПЛАНУВАННЯ ПРОЕКТУ, УПРАВЛІННЯ ПРОЕКТАМИ, ФОРМАЛІЗАЦІЯ ПРОЦЕСІВ ПРОЕКТУ, JAVA.

Метою роботи є застосування моделей та методів ідемпотентної алгебри для вирішення задач планування та управління проектами (мережевого планування).

Методами дослідження обрано аналіз існуючої інформації щодо методів ідемпотентної математики та синтез цієї інформації для моделювання процесів планування та управління проектами з метою їх вдосконалення.

У результаті роботи було розглянуто найпопулярніші критерії планування проектів, методи управління тривалістю проекту. Виявлено що ці методи легко можна описати з математичної точки зору, застосовуючи методи ідемпотентної математики. Розроблено настільний додаток на мові Java для демонстрації можливості практичного застосування.

IDEMPOTENT ALGEBRA, JAVA, PROJECT PLANNING METHODS, PROJECT MANAGEMENT, FORMALIZATION OF THE PROJECT PROCESS.

The object of the study is to apply models and methods of idempotent algebra to solve problems of planning and project management (network planning).

The research methods are used to analyze the existing information on the methods of idempotent mathematics and the synthesis of this information to model the processes of planning and project management in order to improve them.

As a result of work the most popular criteria of planning of projects, methods of management of duration of the project were considered. It was found that these methods can be easily described from a mathematical point of view, using the methods of idempotent mathematics. A desktop application in the Java language has been developed to demonstrate the possibility of practical application.

ЗМІСТ

Вступ.....	6
1 Аналіз стану проблеми	8
2 Опис проведених теоретичних досліджень.....	13
2.1 Огляд існуючих алгоритмів	13
2.2.1 Графік ганта.....	13
2.2.2 Метод критичного шляху.....	14
2.2.3 Метод метра потенціалу.....	16
2.2.4 Техніка оцінки та перегляду проекту.....	18
2.2.5 Техніка графічного оцінювання та огляду	22
2.3 Управління трудовими ресурсами.....	24
3 Аналіз результатів досліджень	27
3.1 Ідемпотентна математика.....	28
3.2 Задача мінімізації максимального часу виконання проекту.....	32
3.3 Задача пошуку мінімального загального часу	35
3.4 Задача пошуку мінімального відхилення від термінів.....	36
3.5 Опис проекту термінами ідемпотентної математики.....	38
4 Програмна реалізація	41
4.1 Архітектура програмної системи	41
4.2 Вимоги до програмного забезпечення.....	43
4.3 Технології та засоби розробки	44
4.4 Інтерфейс програмного забезпечення	45
4.5 Опис можливості використання отриманих результатів	53
Висновки	54
Перелік посилань.....	55
Додаток А Елементи програмного коду	57
Додаток Б Слайди презентації	61
Додаток В Апробація результатів роботи.....	69

ВСТУП

Управління проектами – один з найважливіших способів кваліфікаційної організації праці на будь-якому підприємстві. Для керівника проекту це означає раціональний розподіл операцій проекту з урахуванням різноманітних критеріїв оптимальності. Методи управління проектами стали формуватися в середині минулого сторіччя, за цей час вони склалися в специфічну галузь знань і практичну методологію, яка широко застосовується в різноманітних галузях людської діяльності. Існують міжнародні стандарти управління проектами [1], відповідно до яких будуються процеси управління найрізноманітнішими проектами від науково-дослідних до будівельних, а також управління всіма можливими змінами в компаніях. Проектна модель управління активно впроваджується організаціями, діяльність яких розповсюджена по різних галузях.

Актуальність теми обумовлена тим, що більшість завдань та процесів їх вирішення можна назвати проектами, які, очевидно, повинні бути сплановані та виконані в оптимальні терміни, отже вибір заздалегідь найефективнішого плану виконання проекту є дуже важливим та актуальним питанням на сьогоднішній час.

В даний час існує величезна кількість програмного забезпечення для організації управління проектами. Програми такого типу можна розділити за різними критеріями – за способом реалізації: від настільних додатків до інтернет додатків різних версій; по типу розглянутих проектів: від стохастичних до детермінованих. Програмне забезпечення не завжди реалізує великий набір методів. Зазвичай, це вузькоспеціалізована лише на одному з методів програма.

Метою роботи є застосування моделей і методів ідемпотентної алгебри в задачах управління проектами та розробка засобів планування часу операцій в проектах.

Задачі дослідження, які сприяють досягненню поставленої мети:

- вивчення методів ідемпотентної алгебри для вирішення завдань планування;
- на основі вивчених методів побудова алгоритмів планування часових характеристик проекту;
- реалізація алгоритмів у вигляді комплексу програмних засобів за допомогою обраних засобів розробки програмного забезпечення;

Об'єктом дослідження є процес планування та управління проектом.

Предметом дослідження є методи ідемпотентної математики, які можна застосувати для задачі планування проектів.

Методами дослідження обрано аналіз існуючої інформації щодо методів ідемпотентної математики та синтез цієї інформації для моделювання процесів планування та управління проектами з метою їх вдосконалення.

Елементом наукової новизни є застосування методів ідемпотентної математики в галузі управління проектами. В основі більшості програмних продуктів з організації управління проектами лежать алгоритми, створені ще в 50 - 60-х роках минулого століття. До цих алгоритмів відносяться СРМ (метод критичного шляху), GERT (метод графічної оцінки та аналізу), PERT (техніка оцінки та аналізу проектів) та багато інших.

Практичне значення полягає в тому, що результати роботи можуть бути використані при вирішенні задач планування проектів в різних сферах трудової діяльності від будівельних проектів до проектів з розробки програмного забезпечення.

За результатами роботи було опубліковано статтю на тему «Дослідження технологій та інструментів управління проектами» у збірнику «Вісник. Наука і практика» (оригінальна назва «Monografia rokonferencyjna»), випуск «Наука, дослідження, розвиток. Техніка і технологія. #27» (оригінальна назва «Science, research, development №27»).

1 АНАЛІЗ СТАНУ ПРОБЛЕМИ

В сучасному швидкоплинному світі організацій будь-яка діяльність є проектом – задачею з певними вхідними даними і необхідними результатами (цілями), що зумовлюють спосіб її вирішення. Проект включає в себе задум (проблему), засоби його реалізації (вирішення проблеми) і одержувані в процесі реалізації результати. Для досягнення цілей проекту потрібно пройти багато етапів від планування проекту до його реалізації та завершення, звісно, ці дії повинні бути скеровані та продумані – тут з'являється поняття управління проектами.

Управління проектами – це діяльність, спрямована на досягнення поставлених завдань, підготовку та реалізацію певних планів, використовуючи наявні ресурси: час, капітал, людей.

Існує багато рекомендацій та методологій щодо життєвого циклу проекту, побудови ієрархії відповідальностей та створення структури проекту але дуже важливо на початковому етапі обрати правильну стратегію для ефективного досягнення поставленої цілі.

Аналіз останніх досліджень і публікацій показує що тематика управління проектами є дуже популярною на сьогоднішній день, існує велика кількість робіт присвячених цій проблемі. Далі будуть надані цікаві факти – результати останніх досліджень у галузі управління проектами.

Управління проектами на рівні організації (Enterprise Project Management Office, ЕРМО) – це відділ, якій працює на стратегічному рівні, у співпраці з керівниками організації. Він розміщений на виконавчому рівні в ієрархії компанії та використовується в основному компаніями зі світовим охопленням [2]. Мета цього централізованого офісу – забезпечити керівництво, управління, стандартизовані процеси та найкращі практики, інструменти та методи управління портфелем проектів у масштабах усієї компанії.

Згідно з доповіддю Інституту управління проектами за 2019 рік, компанії, які мають створений стратегічний офіс управління проектами підприємств (ЕРМО),

повідомляють, що на 38% більше проектів відповідають їхнім початковим цілям та бізнес-намірам, а на 33% менше проектів провалюються. Від стартапів до великих організацій, необхідність узгодження проектів із ширшими цілями та баченням буде продовжувати набирати швидкість для підвищення ефективності проекту. Концепція ЕРМО, як правило, асоціюється з великими організаціями, але в 2019 році і далі, компанії будь-якого розміру і в усіх галузях промисловості приймають модель ЕРМО, щоб стати чемпіонами з управління проектами замість низьких показників.

Важливим фактором управління проектами є стандартизація процесів. Кожна організація має власну внутрішню політику, практику, процеси та впливи, які, в свою чергу, визначатимуть свої стандарти. Як правило, стандартизація може не існувати в традиційній структурі ЕРМО, тобто кожна з політик, практик, процесів, інструментів та культури може виглядати зовсім інакше, ніж це стосується решти компанії. Але стандартизація в цих областях повинна відповідати загальним стратегічним цілям вищого рівня. Це гарантує, що всі здійснені проекти служать безпосередньо для просування місії, бачення, цілей та загальних напрямків роботи компанії, визначених виконавчою командою. Це також зменшує витрачені ресурси та неоднозначність та збільшує рівень успішності з усіма ініціативами компанії.

Для організації та підтримки управління проектом потрібні спеціальні інструменти та прийоми – це механізми, за допомогою яких здійснюються процеси управління проектом в організації. Сюди входять, крім прийомів управління проектами (наприклад, структура розподілу роботи або управління заробленою вартістю) різні керівні принципи, в яких визначаються процеси організації, включаючи використання процедурних документів, контрольні списки, посібники для роботи та шаблони, а також використання програмних пакетів та різних баз даних.

Правильне використання інструментів та методів управління проектом повинно полегшити реалізацію принципів управління проектом. Наприклад, проект Інформаційна система управління (PMIS), визначена у дослідженні White and Fortune [3], як найбільш використовуваний інструмент і техніка – це

інструмент, який підтримує та полегшує реалізацію будь-якого проекту, особливо складного, за умови невизначеності та під ринковим, часовим та грошовим тиском чи іншими важкими для управління обмеженнями. Як стверджували Стюарт і Мохамед [4] «Без ефективного використання інформаційних технологій для полегшення процесу управління інформацією серед учасників проекту, малоймовірно, що значне покращення в процесі спілкування відбудеться шляхом продовження використання традиційного паперового процесу». Що стосується програмних засобів управління проектами, то ринок населений широким спектром різноманітних варіантів.

Кілька критеріїв можуть бути використані для керівництва організацією у виборі найбільш відповідних інструментів та прийомів у даному контексті, включаючи різні бази знань. Основа знання управління проектами – це сума знань у межах професії менеджера проекту. Повний комплекс знань про управління проектами включає перевірені традиційні практики, які широко застосовуються, як а також інноваційні практики, що виникають у професії.

Спроби систематизувати знання, необхідні для управління проектами, значною мірою базуються на припущенні, що існують ідентифіковані зразки та узагальнення, з яких можна встановити правила, контроль та вказівки щодо найкращої практики, які можна повторити, навіть якщо не за будь-яких обставин. База знань управління проектами була опублікована професійною асоціацією менеджерів проектів в кінці 1990-х. Відбулося виникнення безлічі стандартів знань, таких як:

- РМВоК® з Інституту управління проектами [5];
- АРМ ВОК від Асоціації з управління проектами [6];
- ІСВ3.0 від Міжнародної асоціації з управління проектами [7];
- Р2М від Асоціації управління проектами Японії [4].

Ці сукупності знань використовуються практиками як посібники з найкращої практики щодо того, що складається з дисципліни [3].

РМВоК®, АРМ ВОК та Р2М – це найвпливовіші публікації щодо бази знань професії [7]. Однак АРМ ВОК і Р2М, набагато ширші за концепцією та обсягом,

ніж PMI PMBoK® [5]. Найкорисніші практики управління проектами показано на рисунку 1.1.

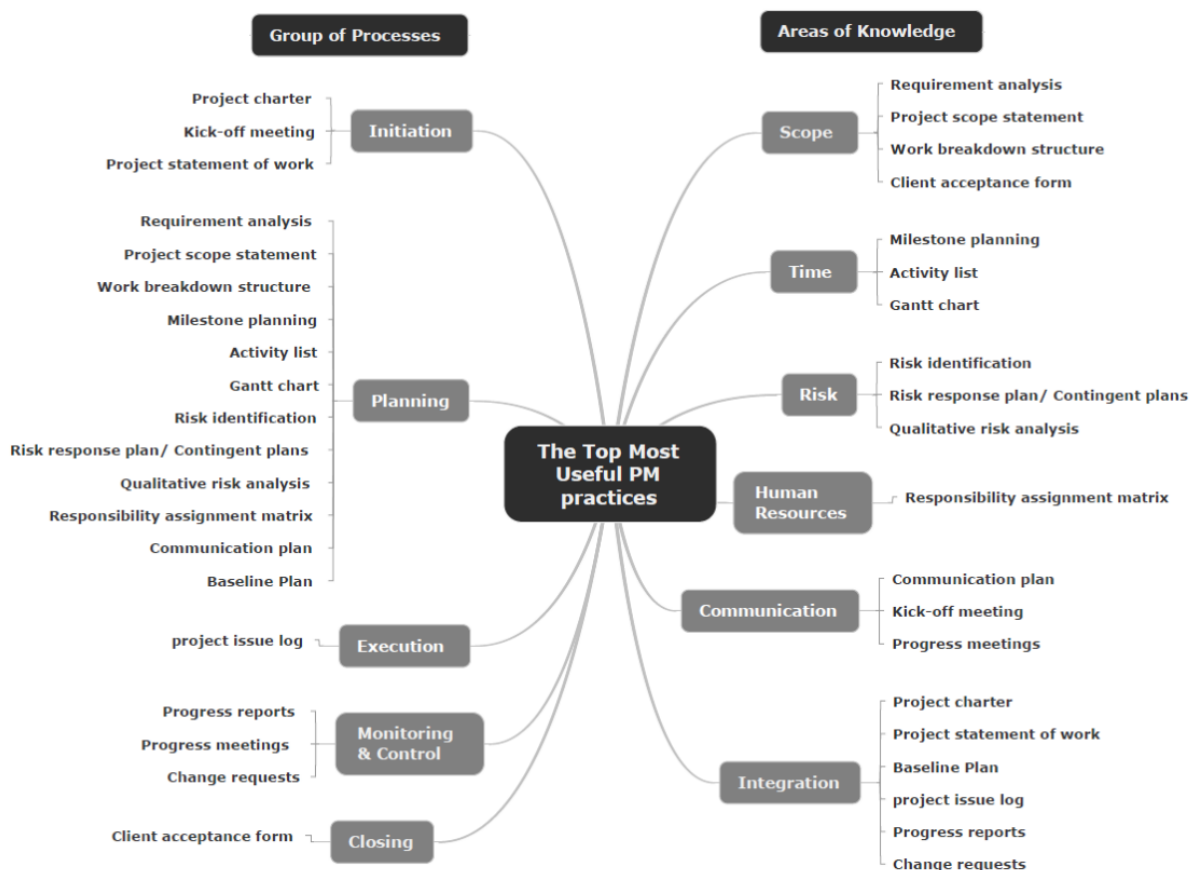


Рисунок 1.1 – Двадцять найкорисніших практик управління проектами за галузями знань та групами процесів

Було проведено конкретні емпіричні дослідження, які визначили найбільш використовувані інструменти. White and Fortune [3] провели опитування, яке було призначене визначити в якій мірі ті, хто бере участь в управлінні проектами, реально використовують методи та прийоми доступні та наскільки ефективними вважаються використовувані методи та прийоми. Автори перерахували 44 методи, методології, інструмента, технології та попросили респондентів вказати, що було використано в проекті, обраному для опитування. Варіанти, включені до списку відповідей, були взяті зі стандартних текстових книг про управління проектами (наприклад, Керцнер). З аналізу 236 учасників White and Fortune встановили, що

найпоширенішими інструментами були: програмне забезпечення «з полиці» (77% респондентів), Діаграми Ганта (64%) і аналіз витрат і вигоди (37%).

Перша двадцятка списку найкорисніших інструментів та прийомів складається з дуже відомих та широко використовуваних інструментів. Тут є кілька сюрпризів. На рисунку 1.1 видно, що визнані практики охоплюють загальну тривалість циклу управління проектом: від ініціації до завершення проекту, хоча особливе значення надається інструментам та методам планування.

Цікавим моментом є структура декомпозиції робіт, або її ще називають ієрархічною структурою робіт (англ. work breakdown structure, WBS) у проектному менеджменті та системотехніці вона є орієнтованою на dokonane виконання проекту, декомпозицією проекту на менші частини. Структура декомпозиції робіт є ключовою часткою робіт по проекту, яка організовує командну роботу по проекту у керовані частини. РМВоК визначає структуру декомпозиції робіт як «ієрархічну декомпозицію робіт, що має бути виконаною командою проекту та орієнтована на успішне завершення проекту».

Елементом структури декомпозиції робіт може бути продукт, дані, послуги або будь-яка комбінація вищезгаданого. Крім того, WBS надає необхідну структуру для детальної оцінки термінів та контролю, а також надає можливість для розробки графіків робіт і контролю їх виконання.

Задачею роботи є проаналізувати існуючі методи планування та управління проектами, дослідити методи ідемпотентної математики щодо застосування їх для вирішення підрахунку часу виконання проекту. Також розробити програмне забезпечення, що надає можливість планування проекту загально звісними методами та методами ідемпотентної математики.

Існуючі загально звісні методи планування проектів засновані на алгоритмах винайдених на початку або в середині минулого сторіччя, тому використання відносно нових методів ідемпотентної математики є обґрунтуванням цілей дослідження

2 ОПИС ПРОВЕДЕНИХ ТЕОРЕТИЧНИХ ДОСЛІДЖЕНЬ

2.1 Огляд існуючих алгоритмів

В якості основної технології візуалізації проекту вибрана мережева. Вона найбільш поширена при плануванні і контролі реалізації складного заходу (проекту). Базується на теорії графів. Структура проекту видається в формі графа типу "мережа". У мережі визначимо операції як вузли, а дуги це події (закінчення або завершення операції). Методами управління проектами називають алгоритми мережевого планування і контролю реалізації проектів, а також алгоритми оцінки трудомісткості і вартості роботи. Взагалі методи можна розділити на кілька груп:

- управління інтеграцією проекту;
- управління ризиками проекту;
- управління трудовими ресурсами;
- управління тривалістю проекту.

У даній роботі розглядається тільки останній пункт. Далі розглянуті деякі алгоритми управління проектами.

2.2.1 Графік Ганта

Графік Ганта – часова діаграма, вважається однією з найстаріших технологій планування. Створено на початку ХХ ст. Г.Л. Гант в США. Перевагою цієї діаграми є її наочність. Візуалізація робіт у вигляді відрізків, довжина яких пропорційна часу їх виконання полегшує сприйняття. Це дуже корисно, коли операції в проекті мають послідовно-паралельну структуру.

Залежно від характеру елементів, позначених на осі ординат, доводиться мати справу з двома видами часових діаграм:

- діаграми динаміки виконання робіт;
- діаграми використання виконавських потужностей.

Для нас інтерес представляє другий тип графіка, що зображений на рисунку 2.1. На осі ординат цього графіка відзначаються виконавці робіт (ресурси). Цю діаграму часто використовують для планування і контролю використання виконавських потужностей.

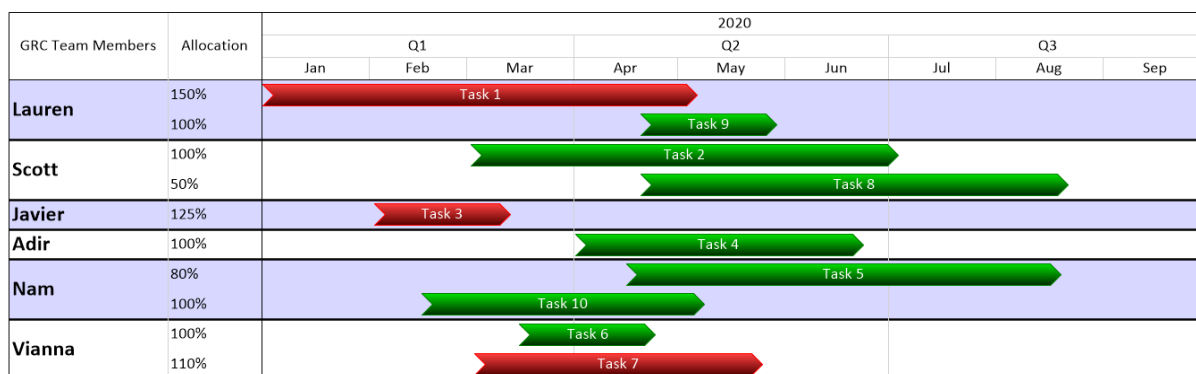


Рисунок 2.1 – Діаграма Ганта

Сьогодні діаграма Ганта найчастіше використовуються для відстеження графіків проектів. Для цього корисно мати змогу показати додаткову інформацію про різні завдання чи фази проекту, наприклад, як завдання відносяться один до одного, як далеко прогресувало кожне завдання, які ресурси використовуються для кожного завдання тощо.

2.2.2 Метод критичного шляху

CPM (англ. Critical Path Method) – технологія створена в 1957 році і виникла на базі графіків Ганта. Вона відноситься до групи детермінованих технологій мережевого планування. Результатом її використання є особливий мережевий граф (представляє операції проекту) з певними розрахунками. В мережевому графі не

повинно бути петель або зациклення. Результат цих розрахунків – план реалізації проекту. Терміни старту і завершення проекту визначаються на початку процесу планування з урахуванням зовнішніх умов. Подією називається настання деякого чітко визначеного стану. Операцією називається деяка задача, для вирішення якої потрібен час і яка повинна бути виконана між двома подіями. Можна виділити наступні основні етапи:

- представлення проекту у вигляді мережевого графіка;
- визначення часу, необхідного для виконання кожної операції;
- визначення дати (термінів) початку та закінчення проекту;
- розрахунок термінів закінчення і початку для кожної операції;
- розрахунок резервів часу;
- визначення критичної послідовності операцій, від якої буде залежати своєчасна реалізація проекту.

Роботи можуть бути реальними, а можуть бути фіктивними (з нульовою тривалістю). Критичний шлях – це послідовність операцій, найбільш важливих для дотримання встановленого терміну реалізації проекту. Критичний шлях визначається в процесі аналізу тривалості виконання операцій в мережі. Такий аналіз заснований на взаємозв'язках операцій та інформації про тривалість їх виконання.

Для кожної події в мережі можна розрахувати терміни її реалізації. Самий ранній термін настання події розраховується за формулою 2.1.

$$E_j = \max [E_i + t_{ij}] \quad (2.1)$$

де E_j – найраніший термін настання події j ,

t_{ij} – тривалість операції, що лежить між подіями i та j

Розрахунок ранніх можливих термінів настання подій в мережі називається прорахунком "вперед". Однак цього не досить, необхідно також знати найпізніші

допустимі строки настання події в мережі. Вони розраховуються аналогічно найранішим за формулою 2.2.

$$L_j = \min [L_i - t_{ij}] \quad (2.2)$$

де L_j – найпізніший термін настання події j ,

t_{ij} – тривалість операції, що лежить між подіями i та j

Такий прорахунок є прорахунком "назад". Розрахунок резерву часу виконується за формулою 2.3 та показує, наскільки можна затриматися при виконанні операції, що закінчується i -ою подією.

$$R_i = L_i - E_i \quad (2.3)$$

де R_i – резерв часу,

E_j – найраніший термін настання події j ,

L_j – найпізніший термін настання події j

Критичний шлях утворює послідовність операцій, яка об'єднує події з найменшими резервами часу.

Критичний шлях – найдовша послідовність операцій в мережі, і її тривалість – це термін завершення проекту в цілому.

Цей метод використовують такі відомі програми та корпорації як Microsoft Project, Oracle Primavera, Gantt Project та інші.

2.2.3 Метод Метра потенціалу

MPM (англ. Metra Potential Method) – так само як і метод критичного шляху (CPM), є мережевою технологією, яка розроблена в 1958 році у Франції. Проект

представлений в формі мережевого графіка, на якому відображені залежності між операціями. Тривалість операцій розраховується за певними правилами з урахуванням наявного досвіду. При використанні цього методу важливим є зв'язок кожної операції з її безпосередніми попередниками. Тому вводиться початкова операція "Start", тривалість якої дорівнює 0. Наприклад, якщо є завдання B і його виконання повинно початися одночасно з виконанням завданням A , то отримаємо твердження у вигляді формули 2.4.

$$ES(B) = LS(B) = 0 \quad (2.4)$$

де $ES(B)$ – найраніший термін початку роботи B відносно початку роботи A ,
 $LS(B)$ – найпізніший термін початку роботи B відносно початку роботи A

Ця технологія також як і попередня аналізує критичні операції. В мережевому графіку дозволені зворотні зв'язки. Для розрахунку ранніх термінів початку операцій використовується матриця відображення тимчасових взаємин між операціями. Для цього вводять допоміжні величини $ES(j)^+$ та $ES(j)^-$. Перший параметр враховує позитивні відносини між операціями в проекті. Його значення розраховується з складеної раніше матриці. Для кожної наступної роботи в процесі переміщення по стовпчиках зліва направо число $ES(j)^+$, що лежить на перетині цього стовпця з рядком, підсумовується зі значенням $ES(j)^+$ для цього рядка. Якщо в стовпці на перетині з різними рядками розташовується кілька чисельних значень, то вибирається максимальна сума. Для першої операції $ES(j)^+$ приймається рівним нулю.

Далі визначається параметр $ES(j)^-$, що враховує тільки негативні відносини між роботами. Процес обчислення в точності збігається з $ES(j)^+$, але для кожної операції враховуються тільки негативні значення. Найраніший термін закінчення робіт визначається за формулою 2.5.

$$ES(i)^k = ES(i)^- + t_i \quad (2.5)$$

де $ES(i)^k$ – найраніший термін закінчення робіт,
 $ES(j)^-$ – параметр, що враховує тільки негативні відносини між роботами,
 t_i – тривалість операції

А найпізніші терміни закінчення робіт визначаються за формулою 2.6.

$$LS(i)^k = LS(i)^- + t_i \quad (2.6)$$

де $LS(i)^k$ – найпізніший термін закінчення робіт,
 $ES(j)^-$ – параметр, що враховує тільки негативні відносини між роботами,
 t_i – тривалість операції

Резерви часу для конкретних робіт можна розрахувати за формулою 2.7.

$$R_i = LS(i)^k - ES(i)^k \quad (2.7)$$

де R_i – резерв часу для конкретної (i -ої) роботи,
 $LS(i)^k$ – найпізніший термін закінчення i -ої роботи,
 $ES(i)^k$ – найраніший термін закінчення i -ої роботи

Роботи, для яких резерви часу дорівнюють нулю, вважаються критичними. Затримки з їх виконанням викликають подовження терміну завершення проекту. Програмне забезпечення, що реалізує даний метод, виявити не вдалося.

2.2.4 Техніка оцінки та перегляду проекту

PERT (англ. Project Evaluation and Review Technique) – техніка оцінки та огляду проекту розроблена на замовлення ВМФ Сполучених Штатів Америки. Використовує метод критичного шляху (CPM) і дозволяє знаходити ймовірність

вчасної реалізації проекту із застосуванням стохастичних методів. Надає можливість статистичної оцінки часу виконання операцій та ймовірності своєчасної реалізації кожного етапу в проекті. У процесі планування виділяються події і операції.

Подія – наступ чітко визначеного стану в проекті. Подія позначає закінчення та/або початок однієї або декількох операцій.

Метод здійснює:

- складання мережевого графіка проекту;
- визначення тривалості кожної операції (найбільш ймовірної, оптимістичної і песимістичної);
- розрахунок математичного очікування і середньо-квадратичного відхилення тривалості виконання операцій;
- визначення критичного шляху.

Тривалість виконання операції – випадкова змінна, що має β - розподілення. Для її підрахунку необхідно заздалегідь знати найоптимістичнішу, найпесимістичнішу та найбільш ймовірну тривалість виконання операції n . Як правило, ці значення бере з голови керівник проекту.

Розрахунок очікуваної тривалості виконання кожної операції n , що входить до складу проекту, виконується за формулою 2.8.

$$t_{en} = \frac{a_n + 4m_n + b_n}{6} \quad (2.8)$$

де t_{en} – очікувана тривалість виконання кожної операції n ,

a_n – найоптимістичніша тривалість виконання операції n ,

b_n – найпесимістичніша тривалість виконання операції n ,

m_n – найбільш ймовірна тривалість виконання операції n

Параметр, що визначає ймовірне відхилення від очікуваного значення, який ще називають стандартним відхиленням, розраховується для кожної операції за формулою 2.9.

$$\delta_n = \frac{b_n - a_n}{6} \quad (2.9)$$

де δ_n – стандартне відхилення,

a_n – найоптимістичніша тривалість виконання операції n ,

b_n – найпесимістичніша тривалість виконання операції n

Стандартне відхилення дорівнює $\frac{1}{6}$ різниці між крайніми оцінками тривалості виконання роботи і вважається мірою невизначеності виконання цієї роботи за очікуваний час.

Критичним шляхом буде шлях з найбільш тривалою реалізацією. Стандартне відхилення очікуваної тривалості реалізації розраховується за формулою 2.10.

$$\delta T_e = \sqrt{\sum \delta_n^2} \quad (2.10)$$

де δT_e – стандартне відхилення очікуваної тривалості,

δ_n – стандартне відхилення тривалості події n

За відомою очікуваною тривалістю реалізації проекту і її стандартного відхилення можна розрахувати ймовірність завершення проекту до будь-якого довільного терміну за формулою 2.12.

$$z = \frac{T_H - T_e}{\delta T_e} \quad (2.11)$$

де z – ймовірність завершення проекту, має нормальний розподіл,

T_H – нормативна тривалість,

T_e – очікувана тривалість,

δT_e – стандартне відхилення очікуваної тривалості

Отримане z визначить ймовірність завершення всіх операцій в проекті протягом заданого часу.

Метод використовується в Microsoft Visio, Project, Oracle Primavera, Gantt Project та інших програмних продуктах.

Тривалість кожної роботи в проекті має обмеження від найкращого до найгіршого значення, а середній показник підлягає розрахунку.

За технікою оцінки та огляду проекту можна побудувати графік очікуваної тривалості операції, приклад якого зображено на рисунку 2.2, для наочного розуміння деталей та прогнозів розвитку проекту.

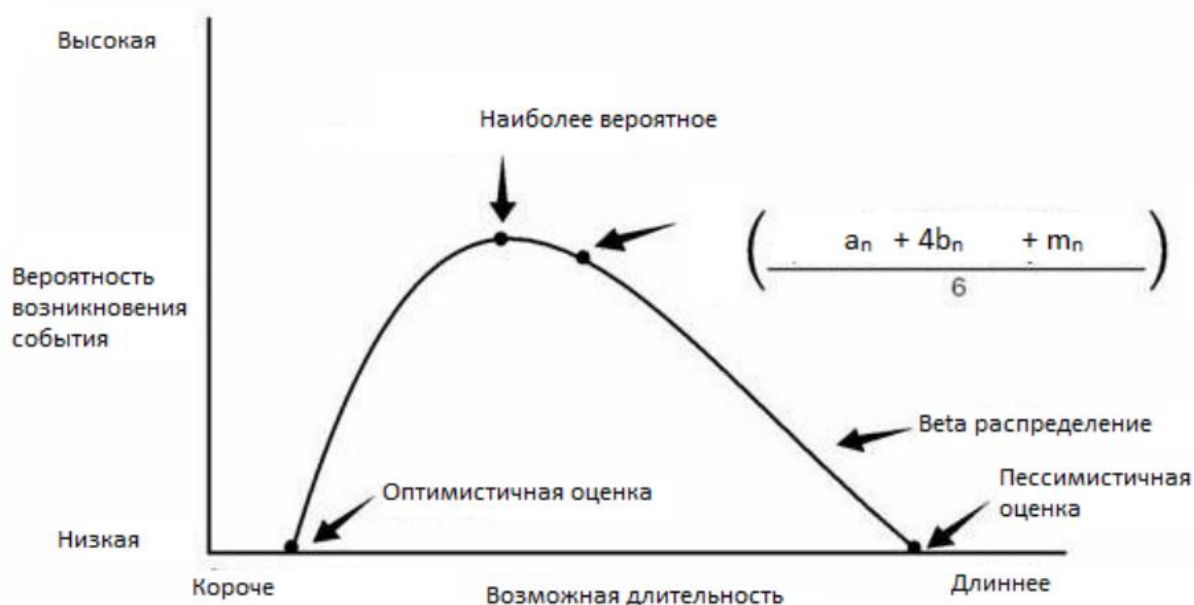


Рисунок 2.2 – Очікувана тривалість операції n

Ряд розподілених показників тривалості доступний до аналізу як сума середньозважених значень для операцій, включених до складу критичного шляху за формулою 2.8, що була наведена раніше. Отримання цього середньозваженого показника поряд з відхиленням дає керівнику проекту можливість розрахування ймовірності можливої тривалості операцій та проекту.

2.2.5 Техніка графічного оцінювання та огляду

На відміну від раніше описаних методів, Graphic Evaluation and Review Technique (GERT) дозволяє використовувати стохастичні мережі. Ці мережі складніше детермінованих, але з їх допомогою можна описувати різні варіанти залежностей між подіями в одній і тій самій мережі, а так само вибирати шляхи розвитку проекту, що відрізняються від певних заздальгідь. Технології, засновані на стохастичних мережах, вводять імовірнісні типи подій в формі логічних об'єднань операцій "або", що дозволяють розглядати альтернативні рішення.

Особливу увагу в проекті слід приділяти альтернативним операціям. Основними етапами альтернативних операцій є:

- опис стохастичної мережі;
- збір числових даних, що характеризують кожен дугу мережу;
- мінімізація побудованої стохастичної мережі;
- перетворення мережі до форми, яка дозволить розрахувати її тривалість та ймовірність реалізації [8].

Термін "подія" розуміється ширше ніж в детермінованих мережах. Події можуть бути не тільки детермінованими але і стохастичними. Такі операнди реалізовані як "І", "АБО" та "виключне АБО".

У першому випадку подія настає тільки тоді, коли всі попередні події здійснилися. У другому – хоча б одне з попередніх подій. У третьому – подія станеться якщо закінчиться одна і тільки одна з взаємовиключних попередніх операцій. Очевидно, що детерміновані входи і виходи подій мають форму "І".

Можна виділити початкові і кінцеві події в проекті: початкова подія може бути стохастичною, але кінцева завжди детерміновано. При використанні зворотних зв'язків або петель завжди вказується кількість повторень (лічильник).

Граф проекту визначається як впорядкована пара, записана формулою. 2.12.

$$G = \langle W, T \rangle \quad (2.12)$$

де G – граф проекту,

W – кінцева множина вершин w_i , яка визначає подію (стан),

T – кінцева множина переміщень t_{ij}

Переміщення визначені на кінцевій множині упорядкованих пар $\langle w_i, w_j \rangle$.
 U_{ij} – функціональні залежності між вершинами w_i і w_j , що зветься дугами. При $i = j$ має місце петля. S – безліч впорядкованих пар $u_{ij}, u_{ik}, u_{kl}, u_{li}$ називається контуром.

На рисунку 2.3 зображено приклад графа проекту, що являє собою стохастичну мережу, з позначками, описаними вище.

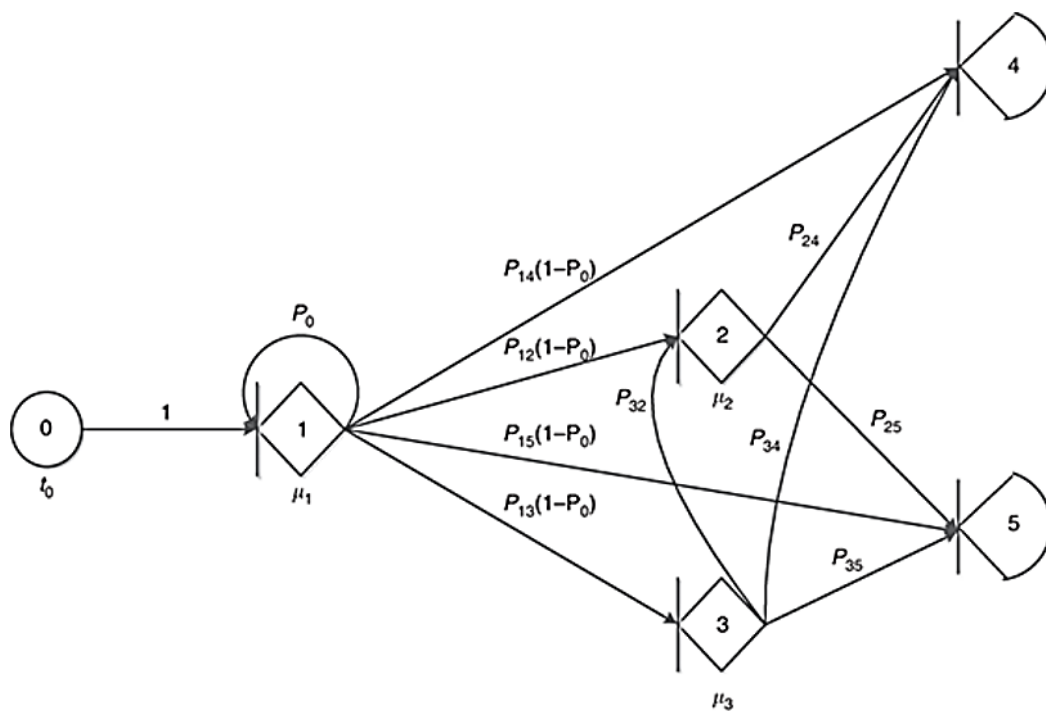


Рисунок 2.3 – Приклад стохастичної мережі

Мінімізація побудованої мережі полягає в спрощенні мережі із застосуванням різних методів, аж до отримання менш складної замісної мережі (або функції), яка в той же час описує оригінальну мережу. Кожна дуга представлена вектором показаним формулою 2.13.

$$w = [p_i, t_i] \quad (2.13)$$

де w – дуга,

p_i – ймовірність реалізації i -й дуги за умови, що подія, якій відповідає вихідний вузол, здійснилося,

t_i – тривалість виконання операції, відповідне i -й дузі

Основні методи скорочення мережі представлені в роботі «Управління проектами» [9].

На основі розрахованих ймовірностей і тривалості виконання робіт можна стверджувати, що реалізація проекту триватиме t одиниць часу з ймовірністю p .

2.3 Управління трудовими ресурсами

Крім управління часом метод критичного шляху (СРМ) можна використовувати для оптимізації використання обмежених ресурсів, у тому числі і трудових. Для цього складається графік трудових ресурсів. Основними етапами якого є:

- створення мережевого графіка проекту з даними про ресурси;
- складання графіка трудових ресурсів для найраніших термінів;
- складання графіка трудових ресурсів для найпізніших термінів;
- приведення потреб ресурсів відповідно до можливостей їх використання.

На рисунку 2.4 зображений графік використання двох ресурсів для семи операцій в проекті. Для кожної операції визначені найраніший час початку і найпізніший час закінчення (виділено темним фоном). Хрестиком позначені дні простою.

Графік трудових ресурсів з урахуванням найбільш ранніх і пізніх можливих термінів показують рівні потреби в ресурсах в процесі виконання проектів. При

оптимальному використанні ресурсів можна мінімізувати граничні значення, шляхом зміщення операцій або подій в межах резервного часу.

Слід зазначити, розглянуті методи вже відносяться до класичних і для вирішення реальних завдань мережевого планування застосовуються методи цілочисельного, лінійного та нелінійного програмування, які зазвичай дають тільки алгоритмічні рішення за допомогою ітеративних обчислювальних процедур. Такі рішення можуть потребувати великих витрат обчислювальних ресурсів. Методи ідемпотентної алгебри дають рішення в явному вигляді, в компактній векторній формі, що гарантує низьку обчислювальну складність рішень.

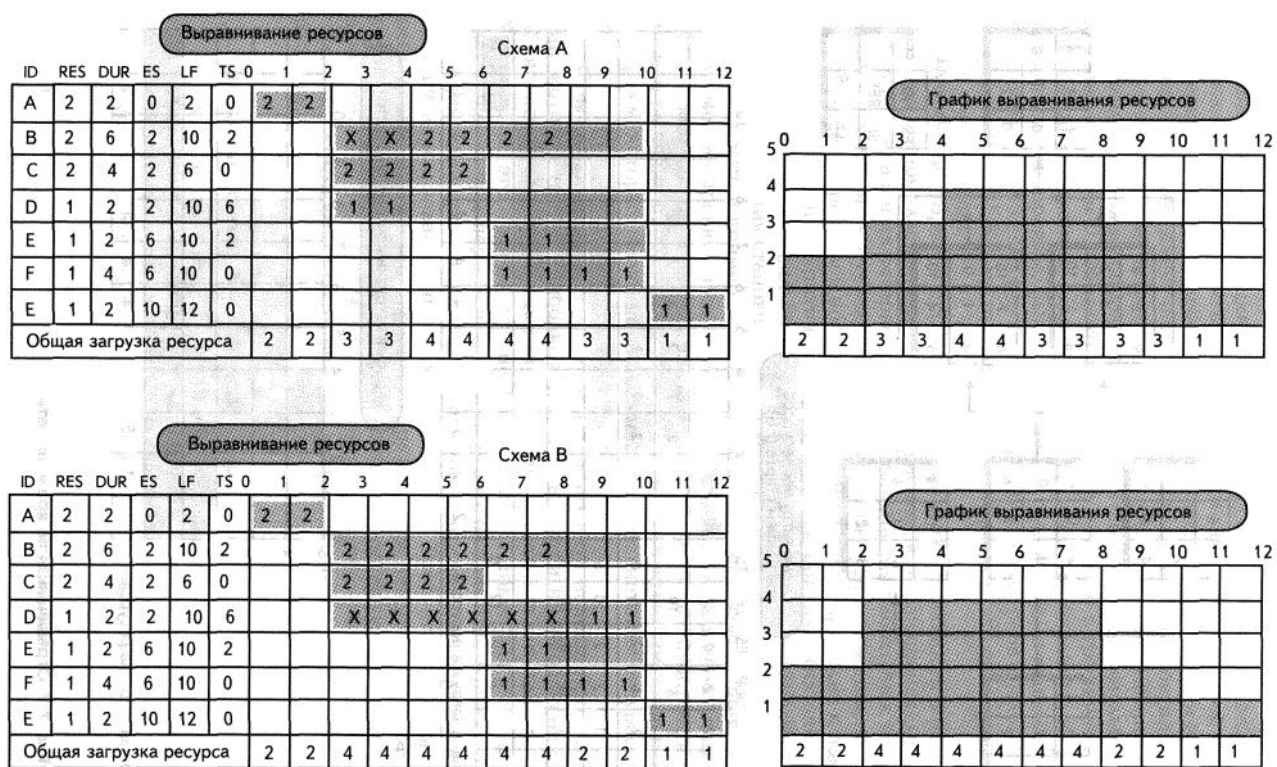


Рисунок 2.4 – Графіки використання ресурсів і варіант оптимізації завантаження

Обчислювальні процедури здатні шукати тільки приватні рішення або встановлювати, що рішень немає. У той час як для багатьох завдань методи ідемпотентної алгебри дозволяють знайти загальне (повне) рішення задачі, яке в явному вигляді описує всю множину рішень задачі.

Різні алгоритмічні рішення, як правило, кожного разу вимагають розробки нових програмних засобів. Вирішення різних завдань на основі методів ідемпотентної алгебри використовують один і той же математичний та обчислювальний апарат (матричні і векторні операції, рішення рівнянь і нерівностей, знаходження власних чисел і векторів та ін.), що робить можливим застосування одних і тих самих уніфікованих програмних засобів для вирішення різних задач.

Розпаралелювання ітеративних обчислювальних схем зазвичай ускладнено. Отримання рішення на основі ідемпотентної алгебри вимагає виконання простих операцій (наприклад, множення матриць), для яких існують різні ефективні схеми паралельних обчислень.

Завдання з області планування проектів зазвичай подаються у вигляді задач оптимізації. Такі проблеми, наприклад, виникають на виробництві при спробах створити план проекту, який мінімізує максимальне відхилення між часом завершення всіх завдань в проекті, при різних умовах, які накладаються на порядок виконання завдань.

Розглянемо проект, що складається з n операцій, які виконуються при обмеженнях типу старт-фініш і старт-старт. Обмеження старт-фініш встановлює кордон для мінімального часу затримки між початком одного завдання і закінченням іншого. Припустимо, що кожна задача негайно закінчується, як тільки зазначені обмеження виявляються виконаними. Обмеження старт-старт визначає мінімальну затримку між початком двох операцій. Одним із завдань є знаходження такого плану проекту, який забезпечує один загальний час завершення для всіх операцій, не порушуючи умов та порядок їх виконання.

3 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ

Аналіз результатів досліджень наочно показує детальний розгляд конкретної задачі, в якості якої було обрано завдання мінімального загального часу виконання проекту, так як ця задача зазвичай представляє найбільший інтерес в області управління проектами. Додаткова інформація про завдання міститься у джерелах [10-11]. Розглянемо проект, який складається з n робіт, які перебувають під тимчасовими обмеженнями типу старт-фініш, ранній старт і пізній старт. Обмеження старт-фініш встановлює межу для мінімального часу затримки між початком одного завдання і закінченням іншого. Припустимо, що кожна задача негайно закінчується, як тільки зазначені обмеження виявляються виконаними. Обмеження старт-старт визначає мінімальну затримку між початком двох операцій. Одним із завдань є знаходження такого плану проекту, який забезпечує один загальний час завершення для всіх операцій, не порушуючи умов на порядок їх виконання.

Для кожної роботи (задачі в проекті) $i = 1, \dots, n$, справедлива формула 3.1.

$$\begin{aligned} x_i + c_{ij} &\leq y_i, \\ j &= 1, \dots, n \end{aligned} \tag{3.1}$$

де x_i – час початку i -ої роботи проекту,

y_i – час завершення i -ої роботи проекту,

c_{ij} – мінімальний час затримки між початком робіт $j = 1, \dots, n$ і завершенням роботи i

Якщо c_{ij} не задано для деякого j – будемо вважати рівним рівняння $c_{ij} = -\infty$. Час завершення роботи i має задовольняти відноsinам обмежень типу старт-фініш [9].

У формулі 3.1 як мінімум одна нерівність повинна виконуватись як рівність. Тепер поєднаємо всі ці відносини у формулу 3.2.

$$y_i = \max_{1 \leq j \leq n} (x_j + c_{ij}), \quad i = 1, \dots, n \quad (3.2)$$

де x_i – час початку i -ої роботи проекту,

y_i – час завершення i -ої роботи проекту,

c_{ij} – мінімальний час затримки між початком робіт $j[1..n]$ і завершенням роботи i

Загальний час завершення є час між найранішим початком і найпізнішим завершенням робіт в проекті. Тобто критерій оптимальності, що показує на скільки план забезпечує один загальний час може бути записано формулою 3.3.

$$\max_{1 \leq j \leq n} y_j - \min_{1 \leq i \leq n} x_i = \max_{1 \leq j \leq n} y_j + \max_{1 \leq i \leq n} (-x_i) \quad (3.3)$$

де x_i – час початку i -ої роботи проекту,

y_i – час завершення i -ої роботи проекту

Якщо підставити певне, заздалегідь визначене, значення y_i до формули 3.2, тоді завдання створення графіка з мінімальним загальним часом виконання проекту може бути сформульоване в стандартній формі: дано c_{ij} для $i, j = 1, \dots, n$ знайти x_1, \dots, x_n , що вирішують завдання мінімізації $\max_{1 \leq i \leq n} \max_{1 \leq j \leq n} (x_j + c_{ij}) + \max_{1 \leq i \leq n} (-x_i)$.

3.1 Ідемпотентна математика

Розглянемо основні визначення і результати ідемпотентної математики для подальшого їх застосування.

Нехай \mathbb{X} – числова множина з двома операціями, що є асоціативними та комутативними: \oplus (додавання) і \otimes (множення); а також адитивним та мультиплікативним нейтральними елементами: $\mathbf{0}$ (нуль) і $\mathbf{1}$ (одиниця) [12].
 Додавання ідемпотентне тоді та тільки тоді, коли вірне твердження 3.4

$$x \oplus x = x, \quad \forall x \in \mathbb{X} \quad (3.4)$$

Операція множення дистрибутивна та зворотна щодо складання, а також для кожного елемента існує зворотний: $x^{-1} \oplus x = x$.

Зведення і цілу ступінь визначається як зазвичай. Для будь-якого $\forall x \in \mathbb{X}$ і цілого $p > 0$, можна записати $x^0 = \mathbf{1}$, $\mathbf{0}^p = \mathbf{0}$, $x^p = x^{p-1} \otimes x$, $x^{p-1} = (x^{-1})^p$.

Далі для стислості будемо вказувати знак множення \otimes , як і в звичайній математиці.

Систему $(\mathbf{0}, \mathbb{X}, \oplus, \otimes, \mathbf{1})$ називають ідемпотентним полуполе [13]. Полуполе лінійно впорядковане за допомогою часткового порядку \leq , такого, що $x \leq y$ тоді і тільки тоді, коли вірне твердження 3.5.

$$x \oplus y = y, \quad \forall x, y \in \mathbb{X} \quad (3.5)$$

Передбачається, що цей частковий порядок можна розширити до повного порядку на всьому \mathbb{X} . Далі символи відносин і формулювання завдань оптимізації розглядаються в контексті цього порядку.

Більш того, полуполе вважається алгебраїчно повним коли для будь-якого $a \neq \mathbf{0}$ та цілого m існує рішення рівняння $x^m = a$.

Для простоти знак множення \otimes будемо опускати, тобто не будемо писати.

Нехай $\mathbb{X}^{m \times n}$ – є множиною матриць з m рядками та n стовпцями над \mathbb{X} .
 Додавання, множення і скалярне множення визначаються стандартними правилами зі скалярними операціями \oplus та \otimes , що взяті замість звичайних операцій додавання

та множення. Матриця з усіма елементами, що дорівнюють $\mathbb{0}$, називається нульовою матрицею.

Для будь-яких матриць відповідного розміру $A = (a_{ij}), B = (b_{ij}), C = (c_{ij})$ та скаляра x , матричне додавання і множення, а також множення на скаляр визначається за формулами 3.6

$$\begin{aligned}\{A \oplus B\}_{ij} &= a_{ij} \oplus b_{ij}, \\ \{AB\}_{ij} &= \bigoplus_k a_{ik} b_{kj}, \\ \{xA\}_{ij} &= xa_{ij}\end{aligned}\tag{3.6}$$

Розглянемо множину $\mathbb{X}^{n \times n}$ матриць з n рядками та n стовпцями.

Матриця з $\mathbb{1}$ на головній діагоналі та $\mathbb{0}$ на інших позиціях називається одиничною та позначається I .

Для будь-якої матриці A , слід матриці розраховується за формулою 3.7.

$$trA = \bigoplus_{i=1}^n a_{ii}\tag{3.7}$$

Як у звичайній математиці, матриця, яка складається з одного рядка (стовпця) називається рядковим (стовпцевим) вектором. Позначимо множину стовпцевих векторів порядку n як \mathbb{X}^n .

Вектор, у якого всі компоненти дорівнюють $\mathbb{0}$, називається нульовим. Вектор вважається регулярним, якщо у нього немає нульових елементів.

Нехай x – регулярний вектор-стовпець та A – матриця. Не важко помітити, що вектор xA регулярний тільки тоді, коли матриця A регулярна по рядках. Так само вектор-рядок $x^T A$ регулярний тільки тоді, коли матриця A регулярна по стовпчиках.

Як зазвичай вектор y лінійно залежний від векторів x_1, \dots, x_n , якщо знайдуться такі скаляри $c_1, \dots, c_n \in \mathbb{X}$, що $y = c_1 x_1 \oplus \dots \oplus c_n x_n$. Вектор y колінеарний вектору x , якщо $y = cx$ для якого-небудь скаляра c .

Для будь-якого ненульового вектору $x = (x_i) \in \mathbb{X}^n$, його мультиплікативно-зв'язаний вектор – це рядковий вектор $\bar{x} = (\bar{x}_i)$, де $\bar{x}_i = x_i^{-1}$ якщо $x_i \neq \mathbb{0}$ або в іншому випадку $x_i = \mathbb{0}$.

Введемо також поняття власного значення матриці в термінах ідемпотентної математики. Якщо існує ненульовий вектор x такий, що виконується рівність $Ax = \lambda x$, то λ – власне значення матриці A , а вектор x – це її власний вектор. Максимальне власне число (в сенсі порядку на \mathbb{X}) зветься спектральним радіусом матриці A , який обчислюється за формулою 3.8.

$$\lambda = \bigoplus_{i=1}^m \text{tr}^{\frac{1}{m}}(A^m) \quad (3.8)$$

Спектральний радіус будь-якої матриці $A \in \mathbb{X}^{n \times m}$ має екстремальну властивість 3.9.

$$\min \bar{x}Ax = \lambda \quad (3.9)$$

Введемо також функцію 3.10, яка обчислює число (власне значення) матриці.

$$\text{Tr}(A) = \bigoplus_{m=1}^n \text{tr} A^m \quad (3.10)$$

За умови що $\text{Tr}(A) \leq \mathbb{1}$, можна визначити матрицю 3.11.

$$A^* = I \oplus A \oplus \dots \oplus A^{n-1} \quad (3.11)$$

при будь-якому $k \geq 0$ отримаємо $A^k \leq A^*$.

3.2 Задача мінімізації максимального часу виконання проекту

Тепер розглянемо задачу планування проекту, коли потрібно мінімізувати максимальний загальний час (час циклу) усіх завдань в проекті – іноді таку задачу називають flowtime-оптимізацією. Детально ця тема розглядається в роботі [14]. У такому завданні крім обмежень старт-старт і старт-фініш присутні обмеження ранній-старт і пізній-фініш. Для кожної операції, обмеження ранній старт і пізній фініш задають відповідно найраніший (за можливістю) час початку операції, та найпізніший (за можливістю) час завершення. Час початку кожної операції повинен задовольняти нерівності 3.12.

$$x_i + b_{ij} \leq x_j, \quad j = 1, \dots, n \quad (3.12)$$

де x – операція, що повинна бути виконана,

b_{ij} – мінімально можлива затримка між початком завдань j та i .

Якщо затримка не зазначена безпосередньо, то вона приймається за $-\infty$. Всі ці відносини взяті разом ведуть до нерівності 3.13.

$$\max(x_1 + b_{i1}, \dots, x_n + b_{in}) \leq x_i \quad (3.13)$$

де x – операція, що повинна бути виконана,

b_{ij} – мінімально можлива затримка між початком завдань j та i .

Так само, якщо врахувати обмеження на максимально ранній час старту операції, то операція не може починати раніше визначеного g_i , тобто справедлива нерівність 3.14.

$$\max(\max(x_1 + b_{i1}, \dots, x_n + b_{in}), g_i) \leq x_i, \quad j = 1, \dots, n \quad (3.14)$$

де x – операція, що повинна бути виконана,

b_{ij} – мінімально можлива затримка між початком завдань j та i ,

g_i – нижня межа часу початку операції

Більш того, для будь-якої операції обмеження старт-фініш вимагають, щоб час завершення задовольняв нерівностям 3.15.

$$x_i + a_{ij} \leq y_j, \quad j = 1, \dots, n \quad (3.14)$$

де y_i – час завершення операції,

a_{ij} – мінімально можлива затримка між початком завдань $j = 1, \dots, n$ і завершенням завдання i

Поєднаємо нерівності 3.13 та 3.14, а також введемо верхню межу часу завершення операції i , та отримаємо твердження 3.15.

$$\begin{aligned} \max(x_1 + a_{i1}, \dots, x_n + a_{in}) &= y_i, \\ y_i &\geq h_i, \quad i = 1, \dots, n \end{aligned} \quad (3.15)$$

де x_i – операція, що повинна бути виконана,

y_i – час завершення операції,

a_{ij} – мінімально можлива затримка між початком завдань $j = 1, \dots, n$ і завершенням завдання i ,

h_i – верхня межа часу завершення операції i

Далі можемо сформулювати задачу оптимізації з обмеженнями 3.16.

$$\begin{aligned} \max(y_1 - x_1, \dots, y_n - x_n) &\rightarrow \min, \\ \max(\max(x_1 + b_{i1}, \dots, x_n + b_{in}), g_i) &\leq x_i, \\ \max(x_1 + a_{i1}, \dots, x_n + a_{in}) &= y_i, \\ h_i &\geq y_i, \quad i = 1, \dots, n \end{aligned} \quad (3.15)$$

Тепер представимо задачу, що розглядається в термінах ідемпотентної математики та сформулюємо твердження 3.16.

$$\begin{aligned}
 & \bigoplus_{1 \leq j \leq n} x_j^{-1} y_j \rightarrow \min, \\
 & (b_{i1} x_1 \oplus g_i) \oplus \dots \oplus (b_{in} x_n \oplus g_i) \leq x_i, \\
 & x_1 a_{i1} \oplus \dots \oplus x_n a_{in} = y_i, \\
 & h_i \geq y_i, \quad i = 1, \dots, n
 \end{aligned} \tag{3.16}$$

Введемо матриці та вектори $A = (a_{ij})$, $B = (b_{ij})$, $g = (g_i)$, $h = (h_i)$, $x = (x_i)$, $y = (y_i)$ та перепишемо скалярний вигляд задачі 3.16 у матрично-векторну форму 3.17.

$$\begin{aligned}
 & \bar{x}y \rightarrow \min, \\
 & Bx \oplus g \leq x, \\
 & Ax = y, \\
 & h \geq y
 \end{aligned} \tag{3.17}$$

У роботі [15] представлено теорему, яка дозволить знайти загальне рішення задачі 3.17.

Нехай x та y – загальне рішення задачі (3.17) поставимо умови, що матриця A має ненульовий спектральний радіус $\Delta = \text{Tr}(B) \oplus \bar{h}AB^*g$ та умови 3.18.

$$\begin{aligned}
 \theta &= \bigoplus_{k=1}^n \bigoplus_{0 \leq i_0 + \dots + i_k \leq n-k}, \\
 & \text{tr}^{\frac{1}{k}} \left(B^{i_0} (AB^{i_0} \dots AB^{i_k}) (I \oplus g\bar{h}A) \right),
 \end{aligned} \tag{3.18}$$

Якщо $\Delta \leq 1$, тоді θ – мінімум в (3.18), який досягається при 3.19.

$$\begin{aligned}x &= S^*u, \\y &= AS^*u\end{aligned}\tag{3.19}$$

де $S = \theta^{-1}A \oplus B$,

u – будь-який регулярний вектор, що задовольняє нерівності $g \leq u \leq (\bar{h}AS^*)^-$

Якщо $\Delta > 1$, то немає регулярного рішення задачі.

3.3 Задача пошуку мінімального загального часу

Мінімальний загальний час виконання проекту іноді називають *makespan* [16], він обчислюється як різниця між часом початку найранішої операції проекту та часом завершення найпізнішої операції. Якщо розглядати множину робіт на множину ресурсів для виконання (за умови, що кожен ресурс може виконувати тільки одну операцію в даний момент часу), то задача являє собою задачу динамічного програмування. При звільненні з умов пов'язаних з ресурсами можна отримати опис завдання в термінах ідемпотентної алгебри, як демонструється у формулі 3.20.

$$\max_{1 \leq i \leq n} y_i - \min_{1 \leq i \leq n} x_i = \max_{1 \leq i \leq n} y_i + \max_{1 \leq i \leq n} (-x_i)\tag{3.20}$$

Далі підставимо замість y_i значення з формули 3.2 та отримаємо 3.21.

$$\max_{1 \leq i \leq n} \max_{1 \leq j \leq n} (x_i + c_i) + \max_{1 \leq i \leq n} (-x_i) \rightarrow \min\tag{3.21}$$

Якщо переписати 3.21 у термінах ідемпотентного напівполя $\mathbb{R}_{max,+}$, тоді отримаємо вигляд формули 3.22.

$$\bigoplus_{i=1}^n \bigoplus_{j=1}^n c_{ij} x_j \bigoplus_{k=1}^n x_k^{-1} \rightarrow \min \quad (3.22)$$

Тепер, введемо матрицю $C = (c_{ij})$ та вектор $x = (x_i)$. У цій нотації, завдання в векторній формі приймає вигляд 3.23.

$$\text{minimize } \mathbf{1}^T C x \bar{x} \mathbf{1} \quad (3.22)$$

Ця проблема детально розглядається в роботі [17]. Як наслідок двох теорем розглянутих в цій роботі, маємо подальший результат.

Нехай C – регулярна по рядках матриця, та $\Delta = \mathbf{1}^T C \mathbf{1} = \|C\|$. Тоді, мінімум в задачі (3.22) дорівнює Δ і досягається при 3.23.

$$\begin{aligned} \alpha \mathbf{1} \leq x \leq \alpha \|C\| (\mathbf{1}^T C)^{-1}, \\ \alpha > 0 \end{aligned} \quad (3.23)$$

Або 3.24, що еквівалентно 3.23.

$$\begin{aligned} x = (I \oplus \|C\|^{-1} \mathbf{1} \mathbf{1}^T C) u, \\ u > 0 \end{aligned} \quad (3.24)$$

3.4 Задача пошуку мінімального відхилення від термінів

Нехай для всіх робіт задані директивні терміни завершення їх виконання. Знайдемо терміни початку робіт, при яких час завершення збігається з директивними термінами. Якщо такі терміни початку робіт визначити неможливо,

то слід знайти наближені рішення задачі, які є оптимальними з точки зору мінімального порушення директивних термінів.

Позначимо через $b = (b_1, \dots, b_n)^T$ вектор директивних термінів завершення робіт. Тоді задача планування, що розглядається зводиться до дослідження в ідемпотентному напівполі $\mathbb{R}_{max,+}$ рівняння 3.25.

$$Ax = b \quad (3.25)$$

Найточніше наближення до вирішення рівняння знаходиться як розв'язок задачі 3.26.

$$(Ax)^- b \oplus Ax(b)^- \rightarrow \min \quad (3.26)$$

Позначивши через $\Delta = \Delta(A, b) = (A(b^- A)^-)^- b$, розглянемо випадок, коли є вірною рівність $\Delta = \mathbb{1} = 0$. Якщо рівність виконується – рішення рівняння 3.25 існує [18]. Рішення має вигляд 3.27.

$$x = (b^- A)^- \quad (3.27)$$

Рішення 3.27 задає час початку робіт, який забезпечує виконання директивних термінів завершення робіт. Якщо $\Delta > 0$, тоді рішенням буде 3.28.

$$\begin{aligned} x_0 &= \sqrt{\Delta}(b^- A)^-, \\ x_1 &= (b^- A)^-, \\ x_2 &= \Delta(b^- A)^- \end{aligned} \quad (3.28)$$

Апроксимовані терміни обчислюються за формулами 3.29.

$$y_0 = Ax_0, \quad (3.29)$$

$$y_1 = Ax_1 \leq b,$$

$$y_2 = Ax_2 \geq b$$

Відхилення в цьому випадку від директивних термінів записується у вигляді 3.30.

$$\begin{aligned} \rho(y_0, b) &= \sqrt{\Delta}, \\ \rho(y_1, b) &= \rho(y_2, b) = \Delta \end{aligned} \quad (3.30)$$

Тоді для визначення нових термінів завершення робіт можна взяти, наприклад, будь-який вектор b такий, що $b \in [y_1; y_2]$. При цьому величина відхилення нових термінів від початкових не буде перевищувати Δ .

Мінімум відхилення, що дорівнює $\sqrt{\Delta}$, досягається при $b = y_0$.

3.5 Опис проекту термінами ідемпотентної математики

Дослідження, проведені в третьому розділі, показали, що матриць, які характеризують обмеження старт-старт і старт-фініш недостатньо для повного математичного опису проекту – не явним чином вказуються ієрархічні взаємозв'язки між операціями (закінчення яких саме операцій тягне за собою початок інших). Для виправлення даної ситуації було введено матрицю 3.31.

$$B \in \mathcal{X}^{n \times n} \quad (3.31)$$

Матриця 3.31 приймає значення або $\mathbf{0}$ або $\mathbf{1}$, для кожної операції вказується після завершення яких операцій починається її виконання. Дана матриця буде

ідемпотентним аналогом матриці суміжності графа, який описує порядок виконання операцій. Напишемо обмеження фініш-старт у вигляді 3.32.

$$By \leq x \quad (3.32)$$

Після підстановки в нерівність 3.32 рівняння 3.31, що описує обмеження старт-фініш, отримаємо нерівність 3.33.

$$BAx \leq x \quad (3.33)$$

Отримане обмеження є вже відомим нам обмеженням на старт-старт, а це означає, що матриця $C = BA$.

В ході роботи вдалося, в деякому сенсі, узагальнити критерії оптимальності – для кожного критерію узагальнення дозволяє вводити всі характеристики старт-старт, старт-фініш, найраніші терміни початку та найпізніші терміни завершення. На даний момент можна стверджувати лише про дві властивості проекту: здійсненності або нездійсненності обмежень, а також, якщо обмеження здійсненні, то дотримання обмежень на найраніший старт операцій в проекті.

За умов, що матриця A має ненульовий спектральний радіус та власне число (виражене формулою 3.34) задовольняє обмеження $\Delta \leq 1$ то обмеження здійсненні і навпаки.

$$\Delta = Tr(B) \oplus h^{-} AB^*g \quad (3.34)$$

Функція $Tr(B)$ визначає матриці та дозволяє відповісти на питання чи має рівняння тривіальне рішення або у нього існують інші. Матриця B^* дозволяє обчислити відстані (в даному випадку час) від кожної вершини (операції в проекті) до всіх інших [19].

Помноживши матрицю B^* на матрицю старт-фініш A , у якій на діагоналі стоять тривалості операцій отримаємо матрицю AB^* , яка визначає час між усіма операціями в проекті з урахуванням тривалості самих операцій. Помноження цієї матриці зліва та справа на найпізніші та на найраніші терміни дозволяє визначити чи конфліктують обмеження між собою.

Дане припущення дозволяє вводити обмеження на найраніший старт та на найпізніший фініш до критеріїв оптимальності, які раніше не враховували дані тимчасові характеристики. До них відносяться: мінімізація (максимізація) розкиду часу завершення операцій, мінімізація загального часу виконання проекту. Якщо обмеження описані вище не виконуються – неможливо провести критерії оптимізації при даних умовах, тобто потрібно або змінити умови або не брати до уваги додаткові обмеження на найраніший час початку операцій та на найпізніший час закінчення операцій.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Архітектура програмної системи

Програмний продукт розроблено як настільний додаток, орієнтований на операційну систему Windows, але розроблене на мові Java програмне забезпечення дозволяє використовувати цей продукт також в операційних системах Linux.

UML діаграма дій відображає послідовність дій, що можуть виконуватися користувачем у розробленій системі, тобто моделюється послідовність бізнес-процесів. Діаграму дій показано на рисунку 4.1.

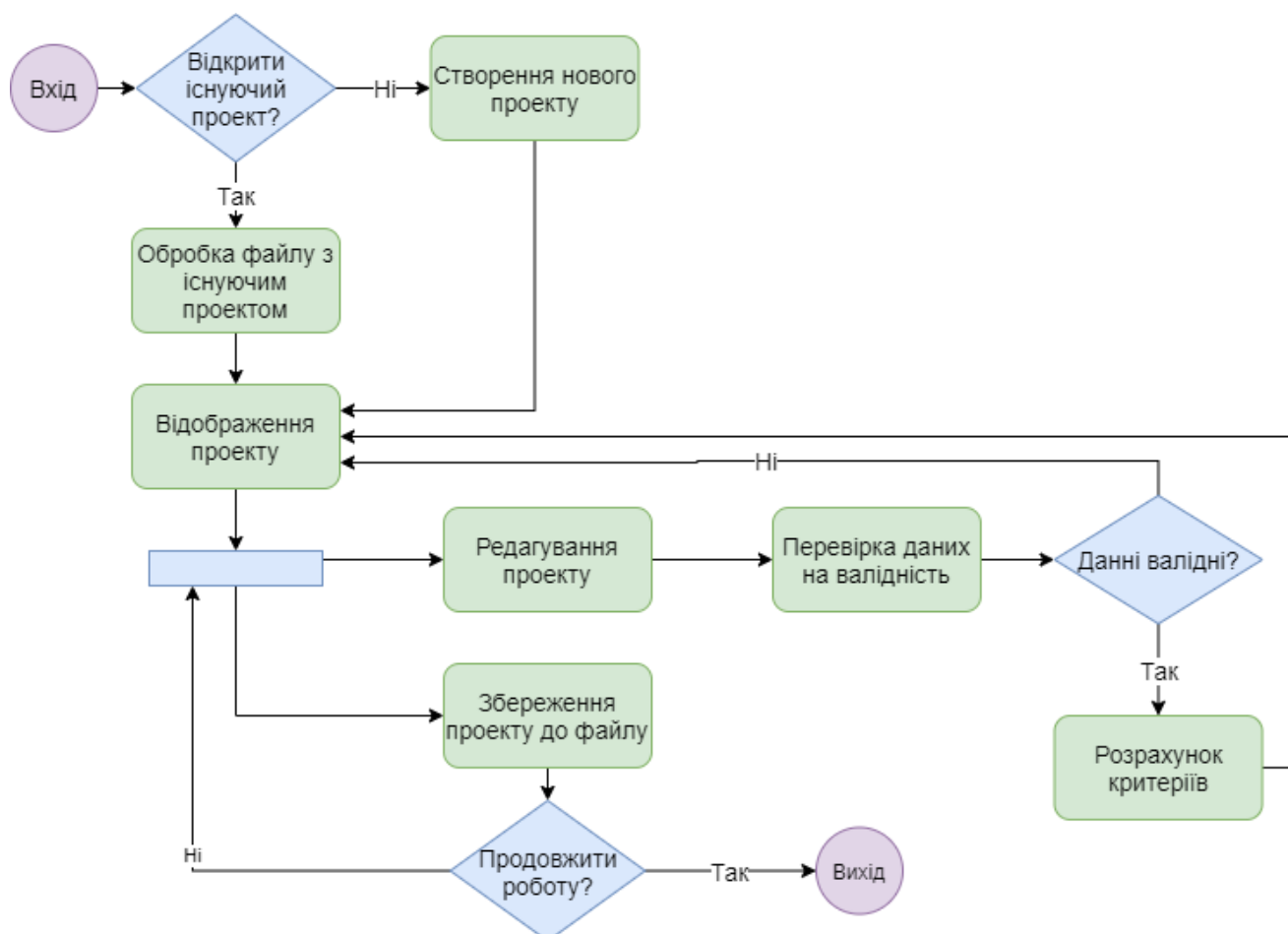


Рисунок 4.1 – Діаграма діяльності

На діаграмі діяльності показано основні сценарії взаємодії користувача з системою.

Структура розробленої системи може бути представлена у вигляді декомпозиції системи на підсистеми, що відповідають за конкретні завдання. Структурні діаграми допомагають застосувати структурний підхід до проектування програмного забезпечення.

Розроблена програмна система складається з наступних підсистем:

- система, що обробляє данні проекту;
- система, що обробляє дій користувача;
- система, що відповідає за відображення інтерфейсу користувача.

З кожною підсистемою можна точно так само провести декомпозицію. на малюнках

Діаграма пакетів розробленого програмного продукту представлена на рисунку 4.2.

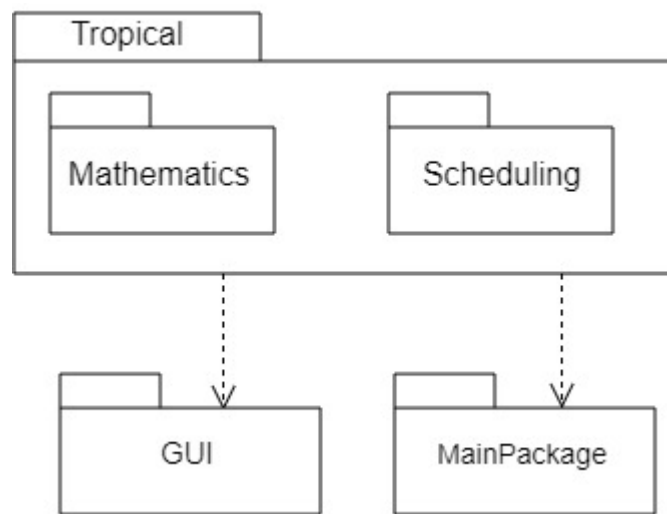


Рисунок 4.2 – Діаграма пакетів

Залежності між пакетами представлені у вигляді пунктирних стрілок.

Пакет Tropical містить у собі два інші пакети Mathematics та Scheduling. Пакет Mathematics відповідає за прості математичні обчислення, а пакет Scheduling відповідає за пошук критеріїв оптимізації проекту.

Пакет GUI забезпечує коректне відображення інтерфейсу програмної системи, а також забезпечує процеси введення даних користувачем та відображення результатів роботи.

Пакет MainPackage є проміжним між двома іншими та містить класи, що реалізують взаємозв'язок між інтерфейсом програми та її математичною складовою.

На рисунку 4.3 представлена діаграма основних класів, що входять до пакета Tropical.

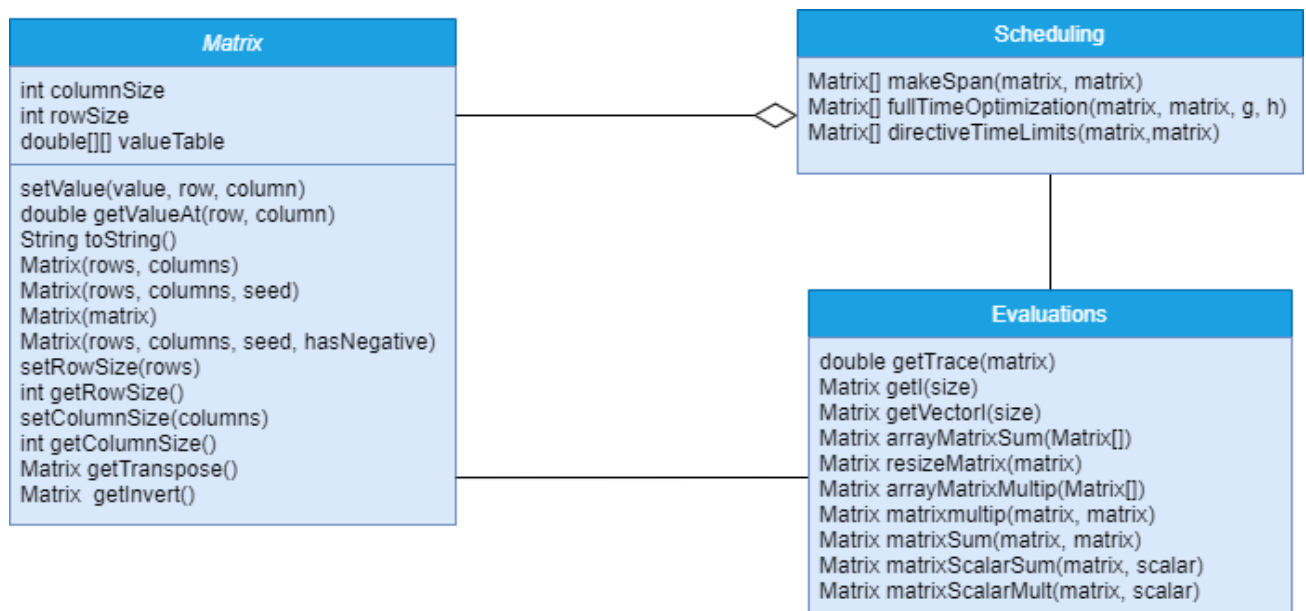


Рисунок 4.3 – Діаграма основних класів, що виконують обчислення

Для реалізації обчислень було створено набір класів, що відповідають за різні математичні операції стосовно матриць.

4.2 Вимоги до програмного забезпечення

До програмного забезпечення ставляться наступні вимоги, яким воно повинно відповідати:

- програмне забезпечення має забезпечувати надійну роботу без збоїв, за виключенням позаштатних ситуацій (неможливість функціонування системного програмного забезпечення не повинна відбуватися частіше ніж один раз на рік, якщо програмний продукт використовується в режимі цілодобової експлуатації);
- програмне забезпечення повинно забезпечувати вирішення комплексів функціональних завдань в інтерактивному режимі та в режимі реального часу;
- програмне забезпечення повинно забезпечувати ведення діалогу з користувачем, а також обмін інформацією між користувачем та машиною;

4.3 Технології та засоби розробки

В якості мови програмування було обрано об'єктно-орієнтовану мову Java [20]. Java-технології мають багато особливостей, що відрізняють їх від інших технологій розробки програмного забезпечення. До головних переваг мови Java можна віднести:

- переносимість;
- безпеку;
- надійність;
- наявність збирача сміття;
- стандартні бібліотеки;
- самодокументований код;
- різноманіття типів додатків.

Програми, що написані на мові програмування Java, можуть бути виконані на будь-якій платформі (після одноразової трансляції в байт-код) для якої встановлена віртуальна Java-машина.

Функціонування та робота програми повністю визначається (а також обмежується) віртуальною Java-машиною. Відсутні механізми для безпосередньої роботи з фізичною пам'яттю та іншим апаратним забезпеченням комп'ютера. У мові

Java відсутні механізми, які потенційно призводять до помилок. Такими механізмами можуть бути: неявне перетворення типів з втратою точності, арифметика покажчиків, та інше.

Присутній чіткий контроль типів, обов'язковий контроль виняткових ситуацій. Більшість логічних помилки, зроблених при написанні коду, виявляються на етапі компіляції. Ефективність виконання Java-програм знижує наявність додаткових перевірок.

Програмувати з використанням динамічно розподіленої пам'яті простіше і надійніше, тому що звільнення пам'яті при роботі програми здійснюється автоматично за допомогою збирача сміття. Багато задач, що зустрічаються при розробці програмного забезпечення, вже вирішені в рамках стандартних бібліотек [21].

Використання об'єктно-орієнтованого підходу дозволяє легко використовувати готові об'єкти в своїх програмах. Є механізм автоматичного генерування документації на основі коментарів, розміщених в тексті програм.

На мові Java можливо реалізувати абсолютно різні за способом функціонування та галуззю використання програми.

4.4 Інтерфейс програмного забезпечення

Діаграма Ганта – широко розповсюджений спосіб ілюстрації плану або графіка робіт по будь-якому проекту. Вона є головним способом візуалізації результатів в розробленому програмному забезпеченні.

Відображення та створення діаграми Ганта було реалізовано за допомогою бібліотеки `swiftGanttChart`.

Приклад діаграми Ганта створеної в розробленому програмному забезпеченні представлено на рисунку 4.4.

У вікні разом з діаграмою Ганта відображено її табличне представлення, де наочно можна побачити деталі часу виконання роботи.

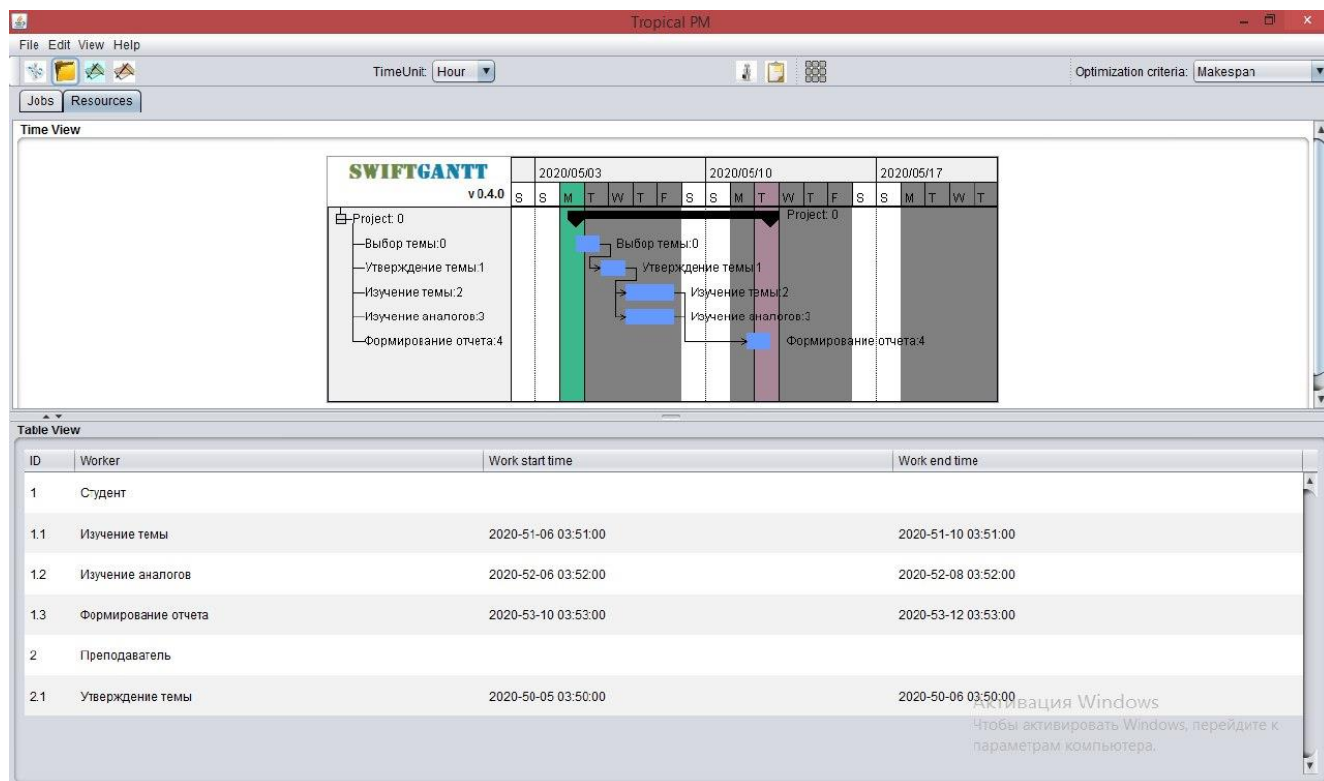


Рисунок 4.4 – Интерфейс программы: диаграмма Ганта

Мережевий графік представлений на рисунку 4.5, було створено за допомогою бібліотеки JGraphX. Цей графік дозволяє користувачеві інтерактивно працювати з проектом: редагувати існуючі операції, перевизначати послідовність операцій простим перетягуванням стрілок.

Перевірка на петлі та цикли проводиться при створенні операцій за допомогою алгоритму проходу графа в глибину та за допомогою перевірки на строгу трикутність матриці інцидентності мережі.

Якщо у проекті кілька стартових операцій або кілька операцій закінчення, то вводяться фіктивні операції початку (закінчення) проекту, які не відображаються користувачеві та поєднують попередні або наступні за всіма операції проекту.

Все взаємодія користувача з програмним забезпеченням здійснюється через інтерфейс користувача, приклади наведені на рисунках 4.4 та 4.5.

Для забезпечення простоти, дружності та зрозумілості, інтерфейс розподілений на два блоки: меню та панель інструментів. Зліва знаходяться дві вкладки, які відповідають за різні варіанти відображення операцій та ресурсів в проекті.

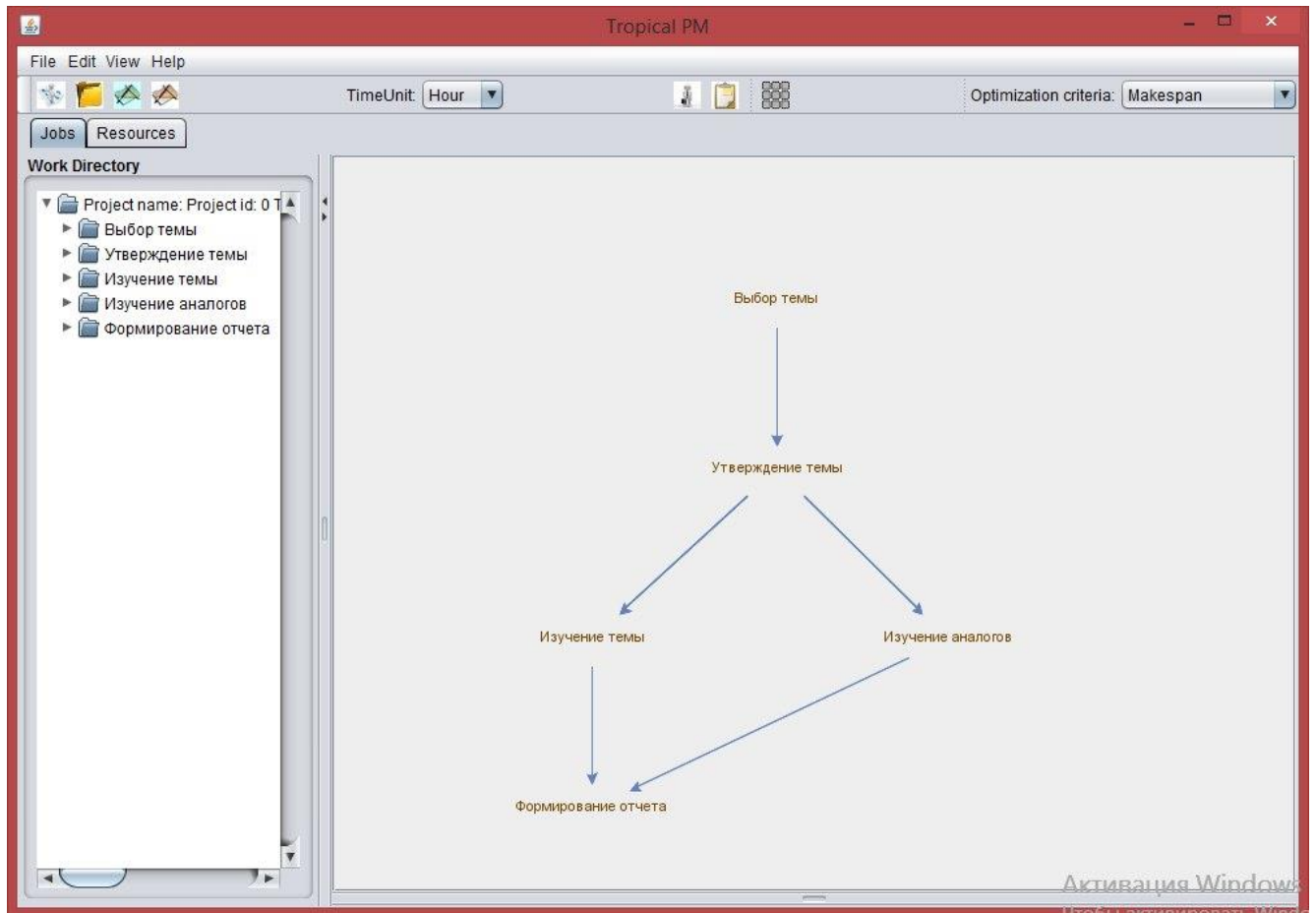


Рисунок 4.5 – Інтерфейс програми: мережевий графік робіт

Зліва посередині є вкладка Jobs, де знаходиться граф взаємодії операцій, а праворуч від неї, відображено дерево проекту. Вкладка Resources відображує діаграму Ганта, яка дозволяє переглянути розподіл операцій в часі, а також таблицю ресурсів та виконуваних робіт.

При першому запуску програми створюється порожній проект з порожньою множиною ресурсів та операцій. Якщо користувач хоче завантажити вже існуючий проект йому лише потрібно вибрати відповідну кнопку на панелі інструментів або в меню File вибрати опцію Open Project або натиснути комбінацію клавіш (Ctrl + O).

Після виконання однієї з попередньо описаних дій відкриється вікно вибору файлу, яке продемонстроване на рисунку 4.6.

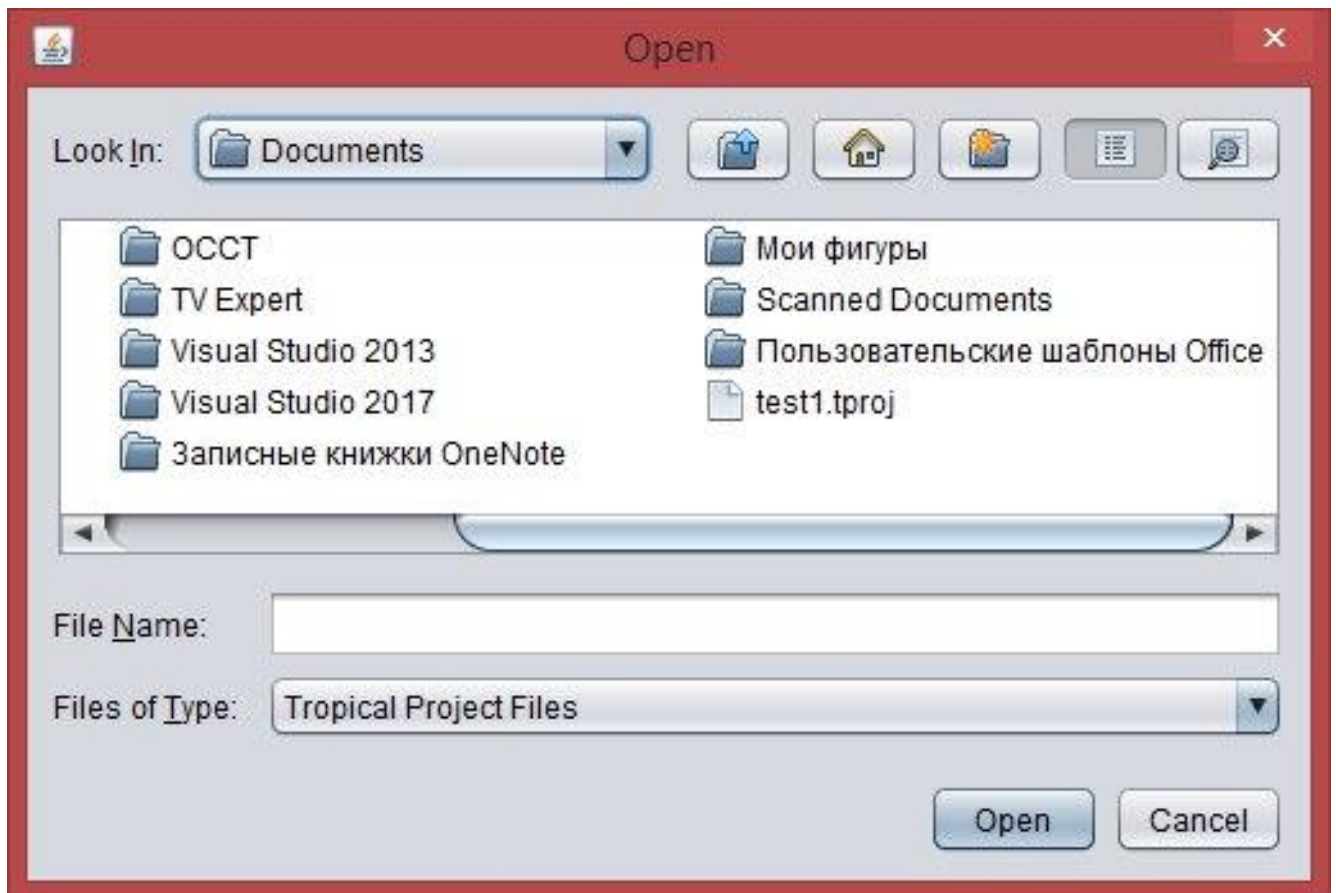


Рисунок 4.6 – Інтерфейс програми: вікно вибору файлу

Користувачеві слід вибрати файл з розширення `.tproj`. Після цього проект відобразиться у всіх вищеописаних вкладках. У вікні дерева проекту будуть представлені всі операції та пов'язані з ними ресурси.

У центрі буде розташований детермінований граф взаємодії операцій. У вкладці `Resources` відобразяться діаграма Ганта з таблицею ресурсів.

Для створення нової роботи користувачу потрібно обрати відповідну кнопку на панелі інструментів, або в меню `Edit` обрати пункт `Create job`, або натиснути комбінацію кнопок `Ctrl + J`.

Після виконання однієї з цих дій з'явиться вікно додавання нової роботи, яке показано на рисунку 4.7. В цьому вікні користувач може задати наступні

властивості для нової операції: ім'я, час початку та час закінчення, обмеження на ці часові характеристики, а також опціонально попередні та подальші роботи та якщо потрібно, зайнятий ресурс, що буде виконувати операцію. Слід зазначити, що зв'язки у проекті детерміновані, тому не допускається створення петель та зворотних зв'язків між роботами.

Рисунок 4.7 – Інтерфейс програми: вікно створення роботи

Перевірка на ациклічність проводиться за допомогою пошуку за графом в глибину. Є можливість визначити кілька попередніх та наступних операцій, це можна зробити у вікні редагування операції.

Також програма має обмеження на часові параметри – операція не може початися пізніше, ніж вона завершується, а обмеження на найраніший початок та найпізніше закінчення так само повинні обмежувати часові характеристики, не порушуючи природну обмеженість.

Тривалість роботи не може бути більше ніж її крайні обмеження. Якщо користувачеві необхідно відредагувати вже існуючу роботу, тоді потрібно обрати роботу подвійним кліком лівої кнопки миші в дереві проекту, або на графі взаємодії робіт або в таблиці ресурсів, що знаходиться на вкладці Resources. Після цього відкриється вікно редагування операції, що показано на рисунку 4.8.

Рисунок 4.8 – Інтерфейс програми: вікно редагування роботи

Інтерфейс створення та редагування ресурсів виглядає аналогічно інтерфейсу створення та редагування робіт. Для створення ресурсу можна або вибрати на панелі інструментів відповідну кнопку, або натиснути комбінацію кнопок Ctrl + J, або вибрати в меню Edit пункт New Resource. Кожній роботі може бути призначений тільки один ресурс, тому при створенні нового ресурсу в списку робіт будуть відсутні вже зайняті роботи.

Якщо потрібно призначити ресурсу відразу кілька робіт, то це легко можна зробити у вікні редагування ресурсу. Щоб відредагувати вже існуючий ресурс слід зробити подвійний клік лівою кнопкою миші по ресурсу або в дереві проекту або в таблиці ресурсів у вкладці Resources.

Вікно створення ресурсу продемонстровано на рисунку 4.9.



Рисунок 4.9 – Інтерфейс програми: вікно створення ресурсу

Вікно редагування ресурсу продемонстровано на рисунку 4.10.

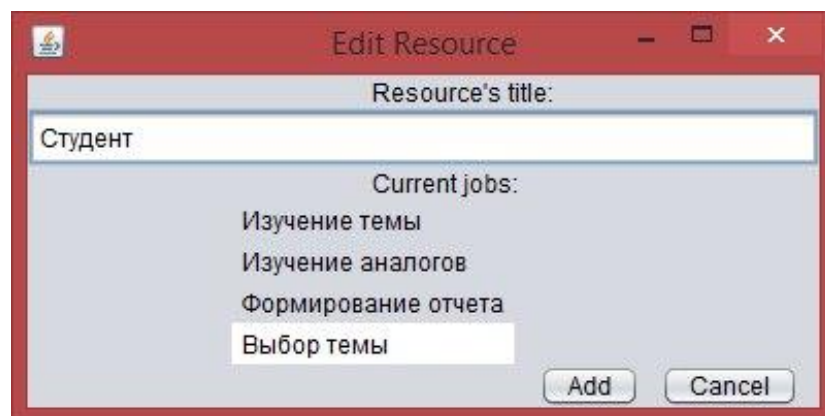


Рисунок 4.10 – Інтерфейс програми: вікно редагування ресурсу

Панель інструментів має різні корисні для зручної роботи елементи управління. За допомогою неї можна задавати точність графіка Ганта, критерій оптимізації і умови його обчислення (відразу або після певного моменту), можна

створювати нові проекти, операції та ресурси, а також зберігати або завантажувати проекти.

При наведенні курсора на будь-який елемент управління – спливає підказка, яка містить інформацію про функції кнопки. Праворуч знаходиться список доступних критеріїв розрахунку. Якщо стоїть галочка "Instant solution", тоді розрахунок обраного критерію проводиться при кожній зміні в проекті.

У центрі панелі інструментів знаходиться кнопка, натиснувши яку користувач має змогу побачити матриці обмежень, які використовуються в розрахунку критеріїв оптимізації. Це матриці обмежень старт-старт, старт-фініш та матриця інцидентності, які показані на рисунку 4.11.

The figure shows three screenshots of the 'Matrices of constraints' dialog box, each displaying a different type of constraint matrix. The dialog box has a title bar 'Matrices of constraints.' and a dropdown menu for 'Временные ограничения на:' (Time constraints on:).

Start - finish matrix:

	Выбор темы:0	Утверждение темы:1	Изучение темы:2	Изучение аналогов:3	Формирование отчета:4
Выбор темы:0	1.0	-Infinity	-Infinity	-Infinity	-Infinity
Утверждение темы:1	-Infinity	1.0	-Infinity	-Infinity	-Infinity
Изучение темы:2	-Infinity	-Infinity	4.0	-Infinity	-Infinity
Изучение аналогов:3	-Infinity	-Infinity	-Infinity	2.0	-Infinity
Формирование отчета:4	-Infinity	-Infinity	-Infinity	-Infinity	2.0

Start - start matrix:

	Выбор темы:0	Утверждение темы:1	Изучение темы:2	Изучение аналогов:3	Формирование отчета:4
Выбор темы:0	-Infinity	-Infinity	-Infinity	-Infinity	-Infinity
Утверждение темы:1	1.0	-Infinity	-Infinity	-Infinity	-Infinity
Изучение темы:2	-Infinity	1.0	-Infinity	-Infinity	-Infinity
Изучение аналогов:3	-Infinity	1.0	-Infinity	-Infinity	-Infinity
Формирование отчета:4	-Infinity	-Infinity	4.0	2.0	-Infinity

Incidence matrix:

	Выбор темы:0	Утверждение темы:1	Изучение темы:2	Изучение аналогов:3	Формирование отчета:4
Выбор темы:0	-Infinity	-Infinity	-Infinity	-Infinity	-Infinity
Утверждение темы:1	0.0	-Infinity	-Infinity	-Infinity	-Infinity
Изучение темы:2	-Infinity	0.0	-Infinity	-Infinity	-Infinity
Изучение аналогов:3	-Infinity	0.0	-Infinity	-Infinity	-Infinity
Формирование отчета:4	-Infinity	-Infinity	0.0	0.0	-Infinity

Рисунок 4.11 – Інтерфейс програми: вікно редагування ресурсу

З метою введення нових обмежень або зміни вже існуючих можна редагувати будь-яку матрицю.

Якщо нове значення є неприпустимим, про це видається попередження.

Ліворуч на панелі інструментів знаходиться список за допомогою якого можна контролювати часову шкалу діаграми Ганта. Можна вибрати в якості одиничного відрізка часу рік, місяць, тиждень або навіть день.

4.5 Опис можливості використання отриманих результатів

Результати даної роботи можуть використовуватись в різноманітних галузях для планування графіків робіт та проектів. Розрахунки, що дозволяє виконати дана робота допоможуть з більшою точністю виявити мінімальний загальний час виконання проекту, мінімально можливе відхилення від директивних строків для кожної операції, це дозволить все одно виконати проект вчасно. На рисунку 4.12 показано приклад результатів роботи програми, яка розраховувала мінімальний загальний час завершення проекту.

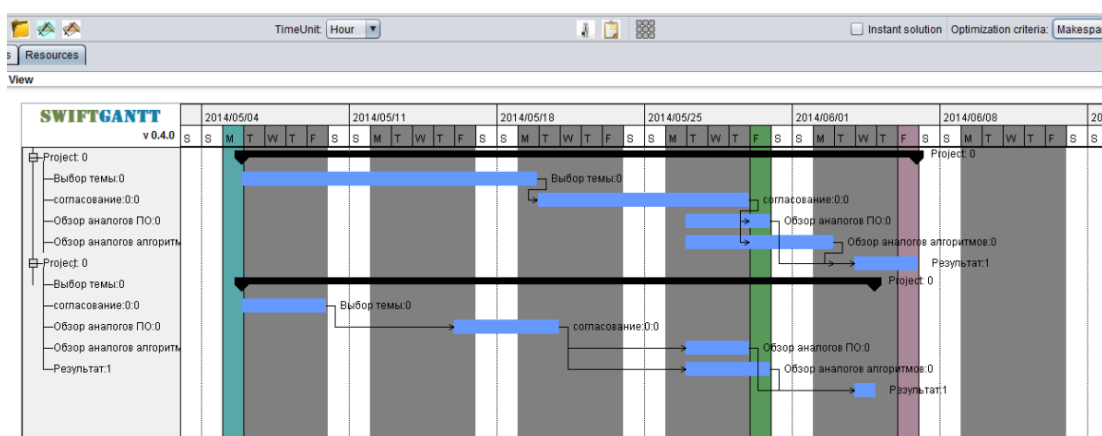


Рисунок 4.12 – Приклад результатів виконання програми

Застосування програми показує, що проект так якжна його операція можуть бути виконані раніше та терміни виконання, розраховані за допомогою методів Ідемпотентної математики наочно демонструються за допомогою диграми Ганта.

ВИСНОВКИ

В ході виконання науково-дослідницької роботи було розглянуто найпопулярніші критерії оптимізації проектів, було виявлено що їх легко можна описати з математичної точки зору. Більшість з них постають у вигляді різних завдань оптимізації.

Багато класичних задач оптимізації (задачі оптимізації на графах, задачі про призначення, динамічного програмування) представляються в ідемпотентній алгебрі у вигляді рішення лінійних рівнянь, знаходження власних чисел і векторів лінійного оператора та іншими обчисленнями даного характеру. За останні десятиліття ідемпотентна алгебра перетворилася в один з розділів математики, що найбільш інтенсивно розвивається, роль якого постійно росте як теоретичної дисципліни та ефективного інструменту вирішення практичних завдань в економіці, техніці, управлінні та інших областях.

Структура даного дослідження відповідає поставленим завданням. Перша частина присвячена огляду існуючих на даних момент і найбільш поширених алгоритмів з управління проектами, а також розгляданню реалізуючих їх різних видів програмного забезпечення. У другій частині розглядається ідемпотентна алгебра, а також уявлення завдань оптимізації проектів в термінах ідемпотентної математики.

Також у роботі вивчено існуючі методи вирішення завдань мережевого планування. Вивчено основи ідемпотентної алгебри для вирішення завдань мережевого планування. Вивчено моделі та завдання на основі ідемпотентної алгебри.

Виявлено методи дисципліни управління проектами, які поки неможливо уявити в термінах ідемпотентної алгебри – оптимальний розподіл операцій серед ресурсів і управління витратами проекту.

ПЕРЕЛІК ПОСИЛАНЬ

1. Керівництво до зводу знань з управління проектами. Четверте видання (Керівництво РМВОК). Американський національний стандарт 2004. Інститут управління проектами. Філадельфія, США, 2010. 380 с.
2. Томас Дж., Менгель Т. Підготовка керівників проектів з вирішення складності – Підвищення кваліфікації управління проектами. Міжнародний журнал управління проектами, т. 26, № 3, 2012, 315 с.
3. Квак І.Х., Анбарі Ф.Т. Аналіз доступності та впливу тенденцій управління проектами: перспективи суміжних дисциплін. Журнал управління проектами, т. 40, № 2, 2013. 103 с.
4. Томас Дж., Маллалі М. Дослідження цінності управління проектами. Ньютаун Сквер, США: Інститут управління проектами, 2015. 55 с.
5. Ши В. Переосмислені впровадження управління проектами: підхід до додавання вартості до карти шляху. Міжнародний журнал управління проектами, т. 29, № 3, 2011. 302 с.
6. Томас Дж., Маллалі М. Розуміння цінності управління проектами: перші кроки до міжнародного дослідження у пошуках цінності. Журнал управління проектами, т. 38, № 3, 2014, 89 с.
7. Уайт Д., Фортун Дж. Сучасна практика в управлінні проектами – емпіричне дослідження. Міжнародний журнал управління проектами, т. 20, № 1, 2012. 95 с.
8. Vira Golyan, Nataliia Golyan. Effective methods of intelligent analysis in business processes// 2019 IEEE . International Conferenc. April. 2019.
9. Троїцький М., Груча Б. Управління проектами. Київ, 2011. 156 с.
10. Ткіндат В., Білуат Дж. С. Багатокритерійне планування: теорія, моделі та Алгоритми. Спрингер, Берлін, 2016. 25 с.

11. Ванхоуке М. Управління проектами з динамічним плануванням. Спрінгер, Берлін, 2013. 32 с.
12. Маслов В. П., Колокольцов В. Н. Ідемпотентний аналіз та його застосування в оптимальному управлінні. Фізматлит, Москва 2014. 144 с.
13. Баселлі Ф., Кохен Дж., Куадрат Дж. П. Синхронізація та лінійність: алгебра для дискретних систем подій. Чичестер, Вілі, 2009. 514 с.
14. Бансал Н. Алгоритми планування потоку часу. Університет Карнегі Меллона, Пітсбург, 2013. 86 с.
15. Кривулін Н. Стримані проблеми тропічної оптимізації: повне рішення та приклад застосування URL: <http://arxiv.org/abs/1305.1454> (дата звертання 10.04.2020).
16. Демеулеместер Є. Л., Херроелен В. С. Планування проекту: Довідник з досліджень. Міжнародна серія в галузі наукових досліджень та управління наукою, 2012. 710 с.
17. Кривулін Н. Проблема тропічної оптимізації із застосуванням до проекту Скед'юлінг з мінімальним пробігом URL: <http://arxiv.org/abs/1403.0268> (дата звертання 07.04.2020).
18. Кривулін Н. Методи ідентичної алгебри проблем для моделювання та аналізу складних систем. Санкт-Петербурзька університетська преса, Санкт-Петербург, 2010. 42 с.
19. Карре Б.А. Алгебра для проблем з мережевою маршрутизацією. Журнал прикладної математики, т. 7, № 3, 2011. 294 с.
20. Портянкін І. Свінг. Ефектні користувальницькі інтерфейси. Лорі, 2007. 608 с.
21. Эккель Б. Філософія Java. 4 видання, видавництво Пітер, 2010. 640 с.
22. Golian N., Golian V., Kalynychenko O., Chalyi S., Bodyanskiy Y. Implementation of search mechanism for implicit dependence in process mining // 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS). – Berlin, 12-14 Sept. 2013. – P. 138-142.