

Хмельницький національний університет

Факультет інформаційних технологій

Кафедра комп'ютерних наук


КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Тема: Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

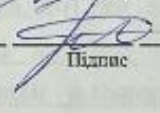
Галузь знань: 12 – Інформаційні технології
Шифр і назва галузі знань

Спеціальність: 122 – Комп'ютерні науки
Шифр і назва спеціальності

Освітня програма: Комп'ютерні науки
Назва освітньої програми

Виконав: студент 2 курсу, група КНМ-22-1  В.В. Охота
Курс, група виконавця Підпис Ініціали, прізвище

Керівник: к.ф.-м.н., доцент кафедри КН  В.Ц. Міхалевський
Науковий ступінь, посада Підпис Ініціали, прізвище

Нормоконтроль: к.т.н., доцент кафедри КН  Р.О. Багрий
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:

Зав. кафедри КН, д.т.н., професор  О.В. Бармак
Науковий ступінь, посада Підпис Ініціали, прізвище

08 грудня 2023 р.

Хмельницький, 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет: інформаційних технологій

Кафедра: комп'ютерних наук

Освітній ступінь: магістр

Галузь знань: 12 – Інформаційні технології

Спеціальність: 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук


Відпис

д.т.н., професор Бармак О. В.

« 01 » вересня 2023 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА**

1. Тема кваліфікаційної роботи магістра: «Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом»
2. Завдання видано студенту: Охоті Вадиму Володимировичу
Прізвище, ім'я, по батькові
3. Керівник роботи: к.ф.-м.н., доцент Михалевський Віталій Цезарійович
Посада, Прізвище, ім'я, по батькові
4. Затверджено наказом університету від «15» серпня 2023 р. № 30.
5. Зміст пояснювальної записки (перелік задач) та вихідні дані: Мета кваліфікаційної роботи магістра – розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Метод має забезпечити знаходження послідовності переправи з найбільшою корисністю та оперативно реагувати на поточну ситуацію. Потрібно вирішити проблеми ефективної організації річкової переправи, використовуючи ситуативну матрицю поведінки об'єктів. Передбачаються динамічні вхідні дані про наявність техніки, їхні властивості та характеристики, часові та вантажозалежні вимоги до завдань переправи, початкові та кінцеві позиції об'єктів переправи. Також необхідно порівняти ефективність роботи розробленого методу з існуючими рішеннями завдань річкової переправи.

Реферат

Кваліфікаційна робота магістра вирішує науково-технічну проблему оптимальної річкової переправи транспортних засобів на основі мурашиного алгоритму із застосуванням у інформаційній системі річкової переправи попередньо встановлених даних про наявність транспортних засобів, їхні характеристики та властивості, а саме їх кількість, місця розташування до і після переправи, множини доступних засобів переправи та динамічного переліку завдань.

Актуальність теми. В сучасних умовах динамічної зміни завдань транспортної логістики, особливо у період війни, необхідно мати технічні та інформаційні рішення для організації переправ через річки. Правильний вибір типу переправи з урахуванням усіх вимог дозволить забезпечити швидке, безпечне та економічно ефективно переміщення людей і вантажів. Сучасні технології автоматизації сприяють підвищенню якості та надійності функціонування переправ.

Особлива увага на даний час звертається на спеціальні переправи. Спеціальні переправи використовуються обмеженим колом осіб або спеціальними транспортними засобами. Це можуть бути військові мости, тимчасові переправи під час будівництва, переправи на промислових об'єктах, лісові переправи. Переправи через річки відіграють важливу роль в організації транспортних комунікацій. Вдосконалення технологій дозволяє будувати все більш складні інженерні споруди. Таким чином, при організації річкових переправ варто комплексно підходити до вирішення завдання з урахуванням усіх вимог, що дозволить оптимально реалізувати необхідне транспортне сполучення.

Одним із наступних кроків у транспортній логістиці може стати саме використання супровідного програмного забезпечення для організації переправ. Використання такого роду забезпечення несе за собою проблему ефективного використання ресурсів та продуктивної роботи системи.

Саме тому, ідея організації річкової переправи транспортних засобів зараз потребує серйозного опрацювання та її успішне застосування залежить від переліку факторів, одним із яких (можливо, найважливішим) є використання

програмного забезпечення для оптимальної роботи системи. Тому розробка чи вдосконалення існуючих методів з метою оптимізації річкової переправи, як у нашому випадку, є актуальною.

Мета і задачі роботи. Метою кваліфікаційної роботи магістра є розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.

Для досягнення поставленої мети визначено наступні задачі:

- провести аналіз предметної області та відомих підходів до організації річкової переправи;
- вдосконалити інформаційну модель річкової переправи транспортних засобів;
- розробити метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом;
- підготувати набір даних для застосування у мурашиному алгоритмі;
- застосувати мурашиний алгоритм для оптимізованого здійснення річкової переправи транспортних засобів;
- провести функціональне та прикладне дослідження ефективності розроблених засобів, у тому числі для тестового аналізу річкової переправи транспортних засобів.

Об’єкт дослідження – процес річкової переправи транспортних засобів.

Предмет дослідження – моделі, методи та засоби автоматизації процесу визначення оптимальної річкової переправи транспортних засобів.

Методи дослідження, застосовані для вирішення поставлених завдань: для організації річкової переправи транспортних засобів використовуються основні положення методів аналізу даних і теорії множин, а для реалізації інформаційної системи – методології проектування інформаційних систем і об’єктно-орієнтований підхід.

Наукова новизна одержаних результатів. В результаті проведеної роботи були отримані наступні результати:

- вдосконалено інформаційну модель річкової переправи транспортних засобів, яка відрізняється тим, що містить формальне подання всіх необхідних

сутностей для автоматизації операцій для оптимізації процесів річкової переправи транспортних засобів;

– розроблено метод здійснення річкової переправи транспортних засобів на основі мурашиного алгоритму, що дозволяє за множиною наявних параметрів транспортних засобів, початкових та кінцевих розташувань техніки, мережі, апаратно-транспортних засобів автоматично визначати множину можливих варіантів переправи, визначити оптимальний, що підвищує ефективність здійснення річкової переправи на 2-3% в залежності від вхідних даних.

Апробація результатів кваліфікаційної роботи магістра та публікації.

Основні наукові й практичні результати кваліфікаційної роботи магістра доповідались на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023» (17-18 листопада 2023 року) у доповіді на тему «Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом»; за темою роботи автором виконано наукову публікацію: Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». - Хмельницький, 2023. - С. 239-241.

Структура та обсяг роботи. Кваліфікаційна робота магістра складається із завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновку, переліку посилань із 41 найменувань та 9 додатків. Загальний обсяг кваліфікаційної роботи магістра становить 115 сторінок, з них 87 сторінок основного тексту та 28 сторінка додатків. У роботі наведено 41 рисунок та 23 таблиці.

Ключові слова: річкова переправа, технічний засіб, мурашиний алгоритм, транспортна логістика, трафік, інформаційна система, інформаційна модель, інформаційна технологія.

Зміст

Вступ.....	4
Розділ 1	7
Характеристика предметної області і постановка задачі.....	7
1.1 Аналіз предметної області	7
1.2 Аналіз рішень та інформаційного забезпечення предметної області.....	15
1.3 Аналіз існуючих публікацій у області побудови річкової переправи.....	20
1.4 Постановка задачі.....	21
Висновки до розділу 1	22
Розділ 2	23
Розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом	23
2.1 Концепція методу річкової переправи.....	23
2.2 Оптимізація методу річкової переправи.....	25
2.3 Метод мурашиного алгоритму для річкової переправи.....	28
2.4 Інформаційна модель методу річкової переправи транспортних засобів...	34
Висновки до розділу 2	44
Розділ 3	45
Програмна реалізація методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.....	45
3.1 Розробка структури даних для взаємодії із моделлю.....	45
3.2 Архітектура програмної реалізації методу.....	49
3.3 Розробка прикладних компонентів системи	51
3.4 Аргументація вибору засобів розробки методу.....	59
Висновки до розділу 3	62
Розділ 4	63
Дослідження методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом	63
4.1 Прикладне тестування методу	63
4.2 Функціональне дослідження та визначення ефективності методу здійснення річкової переправи транспортних засобів	67
Висновки до розділу 4	80
Загальні висновки.....	81
Перелік посилань.....	83
Додатки	

Перелік скорочень

Скорочення, термін, позначення	Пояснення
ПЗ	Програмне забезпечення
БД	База даних
СКБД	Система керування базами даних
СКВ	Середньоквадратичне відхилення
ПП	Програмний продукт
ІС	Інформаційна система
ГІС	Геоінформаційна система
ЕОМ	Електронна обчислювальна машина
ІТ	Інформаційні технології
КН	Комп'ютерні науки
КРМ	Кваліфікаційна робота магістра
МА	Мурашиний алгоритм
ТМА	Точний мурашиний алгоритм
AS	Ant System (Мурашина система)

Вступ

Актуальність теми. В сучасних умовах динамічної зміни завдань транспортної логістики, особливо у період війни, необхідно мати технічні та інформаційні рішення для організації переправ через річки. Правильний вибір типу переправи з урахуванням усіх вимог дозволить забезпечити швидке, безпечне та економічно ефективно переміщення людей і вантажів. Сучасні технології автоматизації сприяють підвищенню якості та надійності функціонування переправ.

Особлива увага на даний час звертається на спеціальні переправи. Спеціальні переправи використовуються обмеженим колом осіб або спеціальними транспортними засобами. Це можуть бути військові мости, тимчасові переправи під час будівництва, переправи на промислових об'єктах, лісові переправи. Переправи через річки відіграють важливу роль в організації транспортних комунікацій. Вдосконалення технологій дозволяє будувати все більш складні інженерні споруди. Таким чином, при організації річкових переправ варто комплексно підходити до вирішення завдання з урахуванням усіх вимог, що дозволить оптимально реалізувати необхідне транспортне сполучення.

Одним із наступних кроків у транспортній логістиці може стати саме використання супровідного програмного забезпечення для організації переправ. Використання такого роду забезпечення несе за собою проблему ефективного використання ресурсів та продуктивної роботи системи.

Саме тому, ідея організації річкової переправи транспортних засобів зараз потребує серйозного опрацювання та її успішне застосування залежить від переліку факторів, одним із яких (можливо, найважливішим) є використання програмного забезпечення для оптимальної роботи системи. Тому розробка чи вдосконалення існуючих методів з метою оптимізації річкової переправи, як у нашому випадку, є актуальною.

Мета і задачі роботи. Метою кваліфікаційної роботи магістра є розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.

Для досягнення поставленої мети визначено наступні задачі:

- провести аналіз предметної області та відомих підходів до організації річкової переправи;
- вдосконалити інформаційну модель річкової переправи транспортних засобів;
- розробити метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом;
- підготувати набір даних для застосування у мурашиному алгоритмі;
- застосувати мурашиний алгоритм для оптимізованого здійснення річкової переправи транспортних засобів;
- провести функціональне та прикладне дослідження ефективності розроблених засобів, у тому числі для тестового аналізу річкової переправи транспортних засобів.

Об’єкт дослідження – процес річкової переправи транспортних засобів.

Предмет дослідження – моделі, методи та засоби автоматизації процесу визначення оптимальної річкової переправи транспортних засобів.

Методи дослідження, застосовані для вирішення поставлених завдань: для організації річкової переправи транспортних засобів використовуються основні положення методів аналізу даних і теорії множин, а для реалізації інформаційної системи – методології проектування інформаційних систем і об’єктно-орієнтований підхід.

Наукова новизна одержаних результатів. В результаті проведеної роботи були отримані наступні результати:

- вдосконалено інформаційну модель річкової переправи транспортних засобів, яка відрізняється тим, що містить формальне подання всіх необхідних сутностей для автоматизації операцій для оптимізації процесів річкової переправи транспортних засобів;

– розроблено метод здійснення річкової переправи транспортних засобів на основі мурашиного алгоритму, що дозволяє за множиною наявних параметрів транспортних засобів, початкових та кінцевих розташувань техніки, мережі, апаратно-транспортних засобів автоматично визначати множину можливих варіантів переправи, визначити оптимальний, що підвищує ефективність здійснення річкової переправи на 2-3% в залежності від вхідних даних.

Апробація результатів кваліфікаційної роботи магістра та публікації.

Основні наукові й практичні результати кваліфікаційної роботи магістра доповідались на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023» (17-18 листопада 2023 року) у доповіді на тему «Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом»; за темою роботи автором виконано наукову публікацію: Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». - Хмельницький, 2023. - С. 239-241.

Структура та обсяг роботи. Кваліфікаційна робота магістра складається із завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновку, переліку посилань із 41 найменувань та 9 додатків. Загальний обсяг кваліфікаційної роботи магістра становить 115 сторінок, з них 87 сторінок основного тексту та 28 сторінка додатків. У роботі наведено 41 рисунок та 23 таблиці.

Розділ 1

Характеристика предметної області і постановка задачі

1.1 Аналіз предметної області

Річкова переправа [1] - це інженерна споруда чи засіб транспорту для переміщення людей, транспортних засобів та вантажів через річку. Існує кілька основних типів переправ, зокрема мости, поромні переправи, понтонні мости, канатні дороги та гідроцикли.

Міст [2] - найпоширеніший та надійний спосіб перетину річки. Зазвичай виконують з металевих чи залізобетонних конструкцій, що спираються на опори. Залежно від прольоту мости поділяють на балочні, аркові, висячі, рамні тощо.

Переваги мостів:

- велика пропускна здатність;
- можливість руху незалежно від погодних умов.

Недоліки:

- висока вартість будівництва та обслуговування.

Поромна переправа [3] - тимчасовий чи постійний засіб переправи за допомогою спеціального плавучого засобу - порома. Пороми приводяться в рух за допомогою двигунів або за течією річки за допомогою канатів.

Переваги:

- відносна дешевизна;
- можливість швидкого введення в експлуатацію.

Недоліки:

- обмежена пропускна здатність;
- залежність від погодних умов.

Понтонний міст [4] - тимчасова споруда, що складається з понтонів, з'єднаних між собою. Встановлюють, як правило, під час військових дій, для швидкого наведення переправи.

Переваги:

- швидкість зведення;

- мобільність;
- невисока вартість.

Недоліки:

- обмежене навантаження;
- нетривалий термін служби.

Канатна дорога [5] - підвісна дорога, в якій рухомий склад підвішений до каната. Може бути одно- чи двоканатною, з рухомим чи нерухомим канатом. Використовується для переправи через ущелини, яри, річки.

Переваги:

- Можливість долати значні перешкоди.

Недоліки:

- обмежена пропускна здатність;
- залежність від погодних умов;
- висока вартість.

Гідроцикл [6] - невелике судно на повітряній подушці, що рухається над поверхнею води. Використовується для переправи через неширокі річки та озера.

Переваги:

- мобільність;
- невелика осадка;
- можливість експлуатації на мілководді.

Недоліки:

- обмежена вантажопідйомність та кількість пасажирів.

При виборі типу переправи слід враховувати:

- ширину та глибину річки;
- характер течії та дно;
- кліматичні та погодні умови;
- обсяг та інтенсивність руху;
- наявність місцевих матеріалів та робочої сили;
- терміновість введення в експлуатацію;
- необхідний строк служби;

- можливість тимчасового чи постійного використання.

Для організації переправи також потрібно врахувати:

- підготовку під'їзних шляхів та майданчиків;
- укріплення берегів та дна в місцях спорудження;
- забезпечення відповідної пропускної спроможності;
- організацію контрольно-пропускного пункту;
- заходи безпеки та охорони переправи.

Для автоматизації роботи переправ активно застосовуються сучасні ІТ-технології, зокрема:

- автоматизовані системи управління рухом;
- електронні системи оплати проїзду;
- GPS-моніторинг транспорту;
- метеостанції та датчики контролю стану конструкцій;
- системи відеоспостереження.

Це дозволяє підвищити швидкість обслуговування, безпеку та надійність роботи переправи.

Отже, існує широкий спектр транспортних рішень для організації переправ через річки. Правильний вибір типу переправи з урахуванням усіх вимог дозволить забезпечити швидке, безпечне та економічно ефективне переміщення людей і вантажів. Сучасні технології автоматизації сприяють підвищенню якості та надійності функціонування переправ.

Залежно від призначення розрізняють переправи загального користування та спеціальні.

Переправи загального користування [7] призначені для перевезення пасажирів та вантажів усіх бажаних. До них належать автодорожні та залізничні мости, поромні переправи, канатні дороги.

Спеціальні переправи [8] використовуються обмеженим колом осіб або спеціальними транспортними засобами. Це можуть бути військові мости, тимчасові переправи під час будівництва, переправи на промислових об'єктах, лісові переправи.

За характером використання переправи поділяють на:

- постійні - функціонують цілорічно незалежно від пори року;
- сезонні - використовуються протягом певного періоду, наприклад, під час весняної повені;
- тимчасові - влаштовуються на короткий термін, наприклад, під час військових дій.

За конструкцією розрізняють:

- нерухомі переправи - мости, естакади, дамби;
- рухомі переправи - пороми, понтони;
- пересувні переправи - на плавучих опорах, за допомогою літальних апаратів;
- наведені переправи - влаштовуються шляхом з'єднання протилежних берегів через річку;
- самохідні переправи - гідроцикли, аерошути.

Основні вимоги до переправ:

- відповідність пропускній спроможності та інтенсивності руху;
- забезпечення безпеки руху та експлуатації;
- надійність та довговічність конструкцій;
- можливість швидкого монтажу та демонтажу споруд;
- стійкість до впливу зовнішніх факторів;
- економічна доцільність та окупність витрат;
- екологічна безпека та раціональне використання ресурсів.

Таким чином, при організації річкових переправ варто комплексно підходити до вирішення завдання з урахуванням усіх вимог, що дозволить оптимально реалізувати необхідне транспортне сполучення.

Переправи через річки відіграють важливу роль в організації транспортних комунікацій. Вдосконалення технологій дозволяє будувати все більш складні інженерні споруди.

Одним з найбільш технологічно складних видів мостів є висячі мости [9]. Їх відмінною особливістю є підвішування проїжджої частини до головних несучих канатів. Це дозволяє перекривати великі прольоти по 1000 м і більше.

До переваг висячих мостів можна віднести:

- можливість долати річки значної ширини;
- відносно невелика вага конструкції;
- висока міцність і довговічність;
- гарний зовнішній вигляд;
- відносно простий монтаж.

До недоліків слід віднести:

- складність розрахунків і проектування;
- великі коливання під дією вітру і навантаження;
- потреба у якісних будівельних матеріалах;
- регулярне обслуговування та моніторинг стану;
- висока вартість будівництва.

Прикладами висячих мостів є:

- міст Золоті Ворота в Сан-Франциско;
- міст Акаши-Кайкьо в Японії - найдовший висячий міст у світі;
- міст через протоку Босфор в Стамбулі;
- міст через річку Дунай в місті Русе, Болгарія.

Отже, висячі мости є складною, але ефективною інженерною конструкцією, що дозволяє з'єднати території, розділені широкими та глибокими перешкодами. Подальший розвиток мостобудування сприятиме реалізації все більш амбітних проектів переправ.

Поряд з традиційними мостами та поромами, перспективним напрямком є застосування амфібійних транспортних засобів для перетину водних перешкод.

Амфібії [10] – це транспортні машини, здатні пересуватися суходолом і водою. Вони оснащуються колесами або гусеничним рушієм для руху по суші та водометами, гребними гвинтами або гідрореактивним двигуном для руху по воді.

Переваги амфібій:

- здатність долати водні перешкоди безпосередньо, без використання поромів та мостів;

- висока прохідність по пересіченій місцевості;
- можливість висадки десанту на необладнаному березі;
- використання в рятувальних та природоохоронних операціях.

Недоліки:

- складність і вартість конструкції порівняно зі звичайними колісними або гусеничними машинами;

- менша швидкість та маневреність на суші порівняно зі звичайним транспортом;

- потреба в регулярному технічному обслуговуванні складних вузлів і агрегатів;

- обмеження щодо стану водного середовища (наявність хвиль, течії, мілководдя).

Приклади застосування:

- плаваючі гусеничні транспортери;
- колісні бронеавтомобілі-амфібії;
- амфібійні катери на повітряній подушці;
- амфібійні рятувальні та пожежні автомобілі;

Незважаючи на певні складнощі, амфібійні транспортні засоби мають хороші перспективи для використання при перетині річок та інших водних перешкод. Це дозволяє значно розширити можливості пересування по пересіченій місцевості.

Ще одним інноваційним напрямком є використання підводних тунелів для перетину річок та морських потоків.

Підводні тунелі [11] прокладаються під дном водойми і дозволяють з'єднати береги, уникаючи будівництва мостів або організації поромних переправ.

Переваги тунелів:

- відсутність обмежень щодо водного та повітряного транспорту при перетині водойми;

- захищеність від негативного впливу хвиль, течій, криги;
- відносна простота експлуатації і ремонту конструкцій;
- висока пропускна здатність;
- естетичність і відсутність псування ландшафту.

Недоліки:

- висока вартість і тривалі строки будівництва;
- складність інженерно-геологічних розвідок;
- потреба у складних технологіях проходки тунелів під водою;
- обмеження за глибиною закладання тунелів.

Приклади:

- євротунель під Ла-Маншем;
- тунелі під річкою Ельба в Гамбурзі;
- проект тунелю під Босфором у Стамбулі;

Отже, незважаючи на складнощі реалізації, підводні тунелі є ефективним технічним рішенням для організації переправ, особливо при значних глибинах та інтенсивному судноплавстві.

При проектуванні та будівництві переправ через річки важливо враховувати водний режим річок.

Водний режим [12] характеризує зміну в часі витрати води та рівнів води в річці. На нього впливають кліматичні фактори, фази водного живлення річки, особливості русла та басейну.

Основні характеристики водного режиму (див. таб. 1.1).

Таблиця 1.1 – Характеристики водного режиму

Параметр	Опис
Меженний рівень	Найнижчий рівень води в річці
Максимальний рівень	Найвищий рівень під час повені або паводку
Амплітуда коливання рівнів	Різниця між максимальним і меженним рівнями

Витрата води	Об'єм води, що протікає через поперечний переріз річки за одиницю часу
Мінімальна, середнє, максимальне навантаження	Меженна, середня та максимальна витрати води
Твердий стік	Частинки ґрунту, що переносяться течією

Вплив водного режиму:

- визначає глибину закладання фундаментів опор мостів та інших споруд;
- враховується при проектуванні прогонових споруд мостів;
- впливає на вибір конструкцій руслової частини мостів;
- визначає необхідність регулювання русла та берегів;
- враховується при розміщенні опор поромних переправ та причалів.

Детальний аналіз водного режиму річки є вкрай важливим для вибору оптимального рішення при будівництві переправи та забезпечення її надійності й довговічності.

Важливим аспектом при спорудженні переправ є забезпечення їх надійності та довговічності.

Надійність переправи [13] – це здатність виконувати задані функції протягом необхідного часу і зберігати значення встановлених експлуатаційних показників.

Таблиця 1.2 – Фактори надійності річкової переправи

Основні фактори, що впливають на надійність річкової переправи	Шляхи підвищення надійності річкової переправи
Якість проектування та будівництва.	Використання сучасних технологій при проектуванні. Моделювання всіх можливих ситуацій та впливів різних чинників для

	розрахунку номінальних можливостей. Використання сучасних будівельних матеріалів.
Характеристики будівельних матеріалів і конструкцій.	Резервування та дублювання несучих елементів. Влаштування додаткових опор та прольотів. Посилення фундаментів і укріплення русла та берегів.
Інтенсивність та склад руху по переправі.	Системи активного моніторингу та контролю пропускної здатності в залежності від природніх, транспортних факторів. Обмеження ризикованих режимів експлуатації.
Якість технічного обслуговування і моніторингу стану.	Своєчасне проведення ремонтів та реконструкцій. Системи активного моніторингу технічного стану.

Таким чином, комплексний підхід та дотримання факторів та шляхів надійності (див. таб. 1.2) до забезпечення надійності є запорукою безпечної та тривалої експлуатації переправ через річки.

1.2 Аналіз рішень та інформаційного забезпечення предметної області

Програмне забезпечення в даній предметній області зазвичай вузько направлене та не є універсальним. ПЗ здебільшого розроблене суто під завдання однієї компанії, яка надає послуги у заданій сфері. Найбільш поширене програмне забезпечення використовується для моделювання та проектування мостів через ріку. Але в наявних програмних продуктах часто немає аналізу та моделей використання алгоритмів для конкретних випадків.

Розглянемо програмне забезпечення, яке є доступним на ринку.

River Crossing Tool, а саме LARS Bridge. RCT [14] від компанії Bentley Systems - це інструмент, який спеціалізується на проектуванні та оптимізації переходів через річки в інфраструктурних проектах. Програмний продукт надає

рішення для планування та розташування мостів, тунелів, або інших інженерних структур для перетину водних об'єктів. Основні можливості River Crossing Tool наведені в таблиці нижче.

Таблиця 1.3 – Можливості LARS Bridge

Можливість	Опис
Геометричне проектування	Дозволяє визначити оптимальні геометричні параметри переходу, враховуючи особливості території та гідротехнічні умови.
Гідравлічний аналіз	Забезпечує оцінку впливу різних факторів на гідравлічні характеристики, такі як течія річки, рівень води та інші аспекти.
Структурний аналіз	Включає в себе можливості для проектування та аналізу структур, забезпечуючи їхню міцність та безпеку.
Оптимізація маршруту	Допомагає знаходити оптимальні маршрути для переходу через річку, з урахуванням екологічних та інженерних обмежень.
Інтеграція з іншими інженерними інструментами	Можливість обміну даними з іншими інженерними програмами для комплексного проектування та взаємодії.

LARS Bridge - це потужний інструмент для оптимізації та ефективного проектування інфраструктурних об'єктів, зокрема мостів, які перетинають водні тіла. Програма відзначається високою ефективністю та можливістю деталізації моделей мостів, надаючи інженерам та дизайнерам засоби для створення детальних та оптимізованих проектів.

Однією з ключових функцій LARS Bridge є його можливість моделювання 3D-структур мостів через річки. Програма включає в себе широкий спектр елементів будови мостів, що дозволяє користувачам максимально деталізувати їх

проекти. Принцип роботи полягає в створенні цифрових моделей, аналогічних тим, як це відбувається у відомому програмному продукті для проектування цифрових моделей Autocad.

Однією з переваг LARS Bridge є його висока функціональність, яку можна порівняти з Autocad. Програма дозволяє інженерам та архітекторам максимально деталізувати свої проекти, забезпечуючи велику кількість елементів будови для точного моделювання. Це робить LARS Bridge важливим інструментом для тих, хто працює у галузі проектування інфраструктури та бажає мати повний контроль над дизайном своїх об'єктів.

Окрім цього, програма включає в себе оптимізаційні можливості, що дозволяють автоматично підбирати оптимальні параметри конструкції моста. Це сприяє підвищенню ефективності проекту та зменшенню витрат часу на розробку. Завдяки цим функціям, LARS Bridge дозволяє інженерам швидше та ефективніше створювати проекти мостів, забезпечуючи при цьому високу якість та безпеку конструкцій.

Важливим аспектом роботи з програмою є можливість порівнювати різні варіанти проекту та обирати оптимальний з точки зору витрат та функціональності. Це забезпечує інженерам гнучкість у виборі рішення та дозволяє реалізовувати проекти з урахуванням конкретних вимог та обмежень.

Узагальнюючи, LARS Bridge є потужним інструментом для інженерів та дизайнерів, які займаються проектуванням мостів. Його висока функціональність, здатність до деталізації та оптимізації роблять його важливим інструментом для ефективної та точної роботи в галузі інфраструктурного проектування.

Розглянемо інтерфейс програми LARS Bridge і виділимо основні інструменти (див. Додаток Г). Спеціальні інструменти - це інструменти, призначені для виконання конкретних завдань. Вони відрізняються від загальних інструментів тим, що мають більш складну конструкцію і призначені для виконання певних операцій.

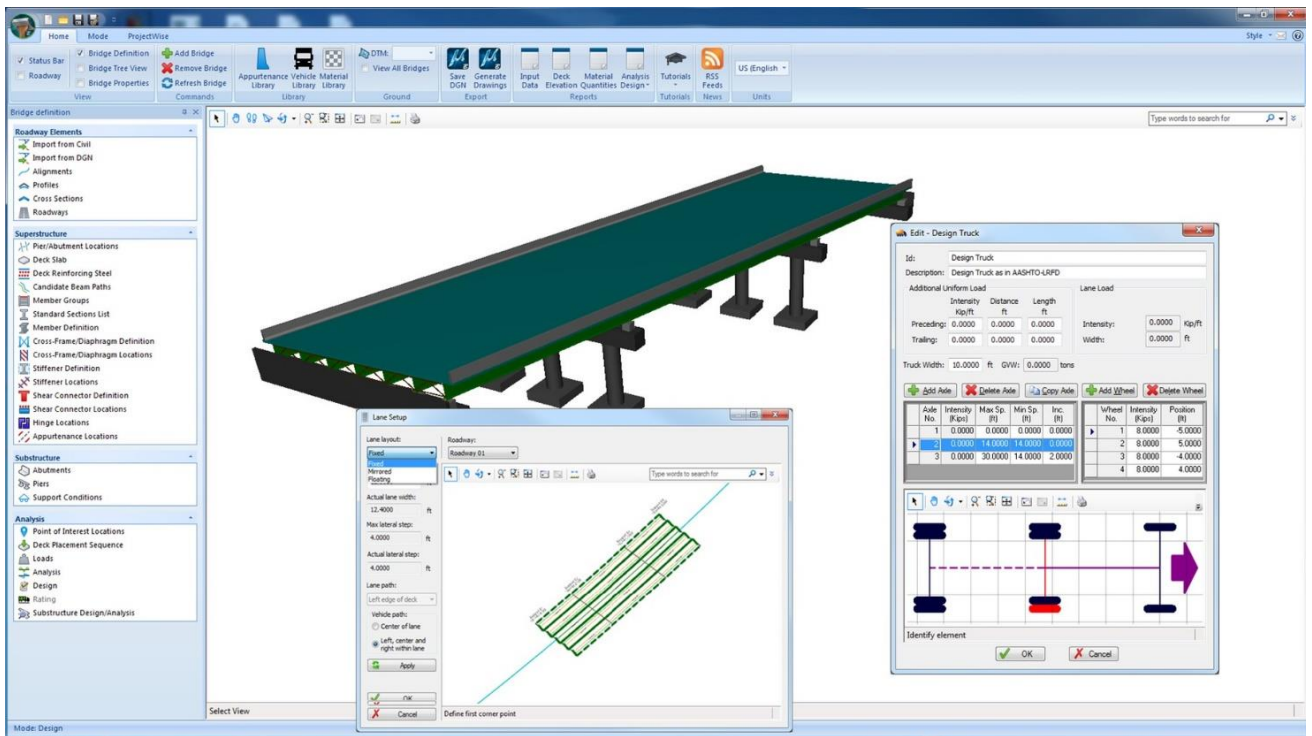


Рисунок 1.1 – Інтерфейс ПЗ LARS Bridge [39]

Маємо, що для створення 3D-моделі конвеєра використовуються спеціальні інструменти, які дозволяють швидко і легко створити модель із заданими параметрами. Спеціальні інструменти ПЗ (див. таб. 1.4).

Таблиця 1.4 – Спеціальні інструменти LARS Bridge

Спеціальні інструменти	Необхідні параметри
Інструмент створення конвеєра	<ul style="list-style-type: none"> – тип конвеєра (стрічковий, шнековий, канатний); – довжина конвеєра; – ширина конвеєра; – висота конвеєра; – профіль конвеєра; – матеріал конвеєра.
Інструмент редагування конвеєра	<ul style="list-style-type: none"> – зміна розміру конвеєра; – зміна форми конвеєра;

	<ul style="list-style-type: none"> – зміна положення конвеєра; – додавання або видалення елементів конвеєра.
--	--

Функції програмного забезпечення - це можливості, які надає програмне забезпечення для виконання певних завдань. Вони відрізняються від інструментів тим, що не мають конкретної фізичної форми, а реалізуються в програмному коді. Наприклад, функція "Відображення" дозволяє переглядати модель з різних ракурсів. Функція "Масштабування" дозволяє змінювати масштаб моделі. У наведеному інтерфейсі можна виділити такі функції (див. таб. 1.5).

Таблиця 1.5 – Функції LARS Bridge

Функція	Опис
Відображення	Дозволяє переглядати модель з різних ракурсів.
Масштабування	Дозволяє змінювати масштаб моделі.
Обертання	Дозволяє повертати модель навколо осей.
Переміщення	Дозволяє переміщувати модель в просторі.
Допомога	Надає інформацію про використання інструментів і функцій інтерфейсу. Допомога доступна в різних формах, наприклад, у вигляді текстових документів, відеоуроків та інтерактивних інструкцій.
Налаштування	Дозволяє налаштувати інтерфейс під свої потреби. Наприклад, можна змінити розміри вікон, розташування елементів інтерфейсу або налаштувати параметри вигляду.

Даний програмний продукт чудово підходить для створення моделі моста, побудови всіх складових ланок та аналізу надійності конструкції. Недоліки полягають в тому, що програмне забезпечення дозволяє розробляти лише мости, інші типи переправ не охоплені, що не підходить для всіх можливих випадків за різними факторами. Розробка алгоритмів для аналізу та знаходження кращого

маршруту річкової переправи актуальна особливо зараз, в період війни, при постійній зміні завдань та умов річкової переправи.

1.3 Аналіз існуючих публікацій у області побудови річкової переправи

Річкові переправи відіграють важливу роль в транспортній інфраструктурі, забезпечуючи зв'язок між берегами річок. В останні роки спостерігається підвищений інтерес до вивчення різних аспектів проектування та функціонування річкових переправ. Це пов'язано як з необхідністю будівництва нових переправ, так і з оцінкою технічного стану та продовженням терміну експлуатації існуючих. Одним з ключових питань є вибір оптимального варіанту переправи та її розташування з урахуванням конкретних умов річки та потреб транспортного сполучення.

В публікації [15] запропоновано підхід на основі алгоритму мурашиних колоній для визначення оптимального маршруту поромної переправи. Враховуються швидкість течії річки, її глибина, інтенсивність руху на маршруті переправи. Метод дозволяє мінімізувати час та витрати на здійснення переправи за рахунок знаходження найбільш ефективного варіанту. Переваги підходу полягають у простоті реалізації та можливості застосування для переправ різної складності. Недоліком є тривалий час розрахунку оптимального розв'язку для складних випадків.

У дослідженні [16] розглядають питання проектування тимчасового понтонного мосту, який може бути швидко розгорнутий для організації переправи через річку в екстрених ситуаціях. Застосовуються методи комп'ютерного моделювання для оцінки міцності та надійності конструкції мосту залежно від його конфігурації, розташування опор, ширини проїзної частини тощо. Розглядаються різні варіанти навантажень з урахуванням гідрологічних факторів. Отримані результати дозволяють обрати оптимальний проект мосту для конкретних умов. Перевагою є можливість моделювання різних сценаріїв, проте потрібні потужності для обчислень обмежують масштабність завдань.

Публікація [17] присвячена використанню безпілотних літальних апаратів для моніторингу стану мостових конструкцій річкових переправ. За допомогою дронів з камерами високої роздільної здатності та іншими датчиками можна отримати докладні дані про наявність пошкоджень в гідротранспортних спорудах. Порівняно з традиційними методами обстеження, технології на основі дронів дозволяють істотно підвищити оперативність контролю стану конструкцій, особливо важкодоступних ділянок. Це дає змогу своєчасно виявляти несправності та планувати ремонтні роботи. Обмеження пов'язані хіба що з погодними умовами.

Отже, існує цілий спектр підходів до забезпечення річкових переправ – від питань вибору типу конструкції та оптимізації її параметрів, до визначення раціонального маршруту руху поромів і моніторингу технічного стану за допомогою новітніх технологій. Загальною тенденцією є прагнення до комплексного розгляду даної задачі з застосуванням сучасних інформаційних технологій.

1.4 Постановка задачі

Метою кваліфікаційної роботи магістра є розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом[16]. Розрахунок та логіка алгоритму повинна знаходити оптимальне місце для розташування переправи через річку з урахуванням рельєфу місцевості, інфраструктури та інтенсивності транспортних потоків, також програмний продукт повинен бути універсальним та підходити для використання незалежно від умов місцевості, клімату та інших факторів. Його застосування дозволить мінімізувати витрати на будівництво та експлуатацію переправи.

Для досягнення поставленої мети визначено наступні задачі:

– провести аналіз предметної області та відомих підходів до організації річкової переправи;

- розробити метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом;
- підготувати набір даних для застосування у мурашиному алгоритмі;
- використати мурашиний алгоритм для оптимізованого здійснення річкової переправи транспортних засобів;
- провести функціональне та прикладне дослідження ефективності розробленого методу та визначити його загальну точність.

Успішне виконання завдання передбачає розробку методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом, досліджуючи місця розташування та варіанти річкової переправи. Це дозволить приймати обґрунтовані проектні рішення з урахуванням усіх важливих факторів.

Висновки до розділу 1

У даному розділі проведено аналіз проблеми проектування та знаходження оптимального маршруту для реалізації річкової переправи, а саме: розглянуто основні аспекти, проблеми, фактори при розробці даного методу та проектуванні відповідної системи.

При організації річкових переправ варто комплексно підходити до вирішення завдання з урахуванням усіх вимог, що дозволить оптимально реалізувати необхідне транспортне сполучення.

Переправи через річки відіграють важливу роль в організації транспортних комунікацій. Вдосконалення технологій дозволяє будувати все більш складні інженерні споруди.

Описано актуальність завдання по розробці методу ефективного знаходження локації для реалізації річкової переправи різного типу конструкцій, залежно від факторів впливу та використання.

Сформульовано постановку завдання, деталізовано задачі та акцентовано увагу на розробку методу здійснення переправи на основі мурашиного алгоритму.

Розділ 2

Розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

2.1 Концепція методу річкової переправи

Метод реалізації річкової переправи включає в себе розробку ефективного підходу [18] для подолання перешкоди у вигляді річки. При вивченні системи річкової переправи важливо враховувати логістичний аспект. Для уявлення концепції можна розглядати простий сценарій, де присутні три ключові сутності: початкова точка, кінцева точка та перешкода у вигляді водойми між ними (див. рис. 2.1).

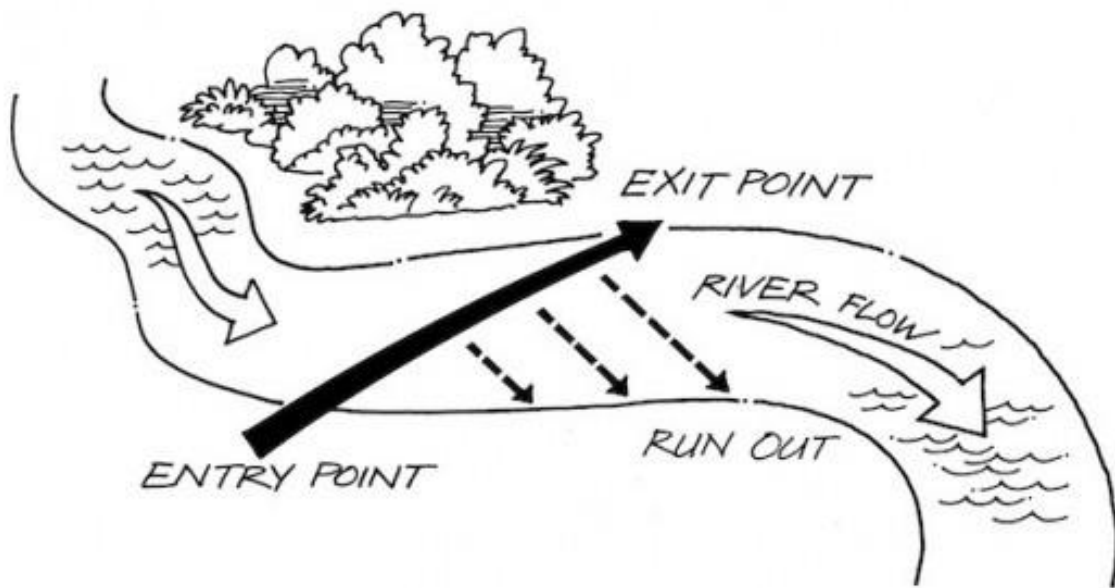


Рисунок 2.1 – Представлення шляху побудови переправи [18]

У базовій концепції була розроблена система, яка складається з трьох об'єктів, що визначають маршрут для перетину водойми. Однак цього недостатньо для визначення річкової переправи. Для уточнення концепції додаємо ще один

об'єкт: тип переправи. Цей тип представляє собою метод реалізації перетину річкової перешкоди, включаючи різноманітні варіанти та засоби, такі як мости, катери, пороми, помпові переправи та інші (див. рис. 2.2).

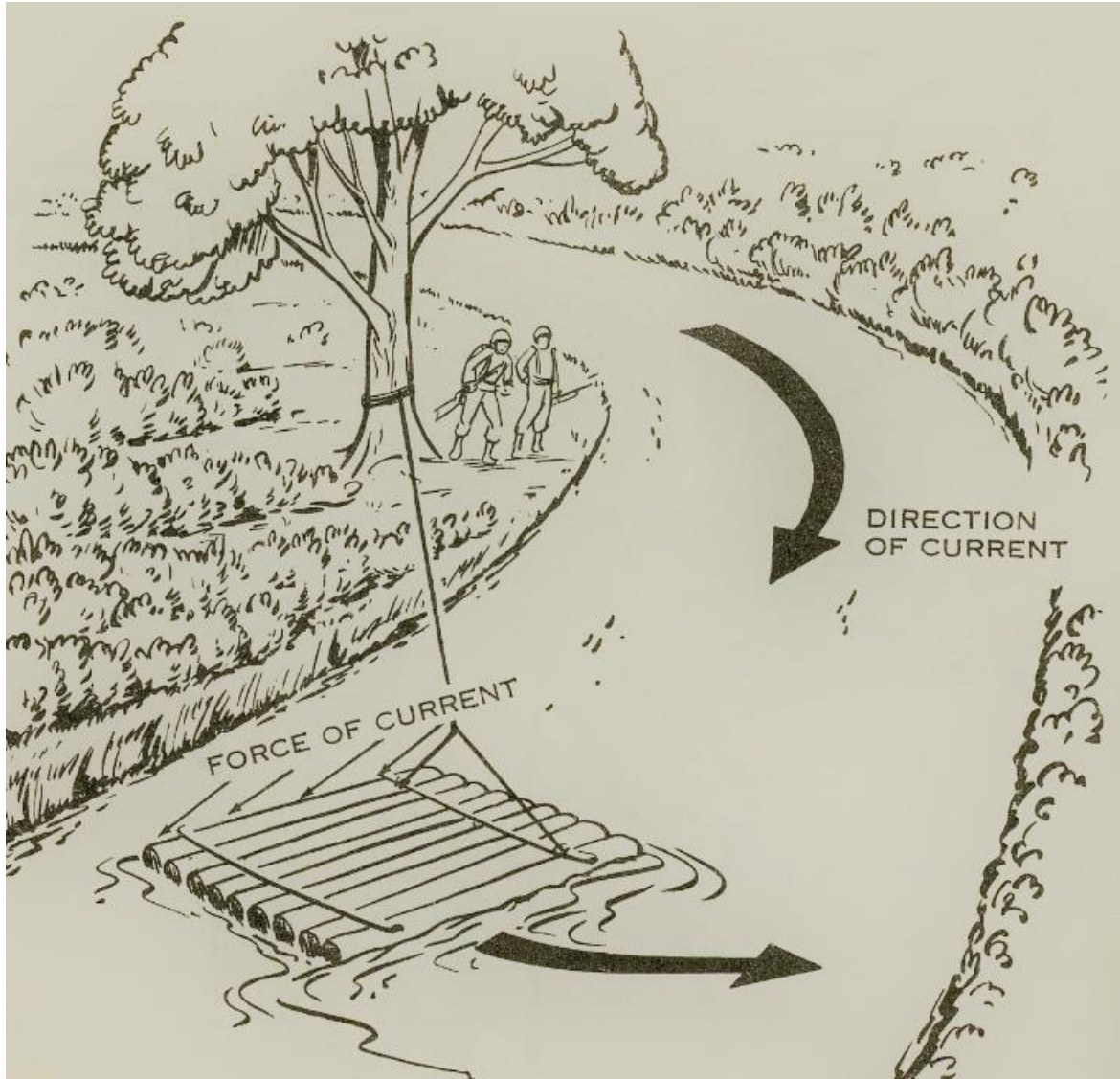


Рисунок 2.2 – Спосіб здійснення переправи за допомогою плоту [19]

У розширеній системі ми включили всі основні об'єкти (див. таб. 2.1), що дозволяють детально моделювати процес переправи через річку. Ця комплексна модель надає повний набір об'єктів, необхідних для методу переправи через річку.

Таблиця 2.1 – Об'єкти методу річкової переправи

Об'єкт	Опис
Початкова точка	Позначає місце, з якого починається шлях для перетину річки.
Кінцева точка	Визначає місце, куди спрямований маршрут через водойму.
Перешкода (водойма)	Представляє собою річку або іншу водну перешкоду, яку потрібно подолати.
Тип переправи:	Визначає метод або засіб, яким буде здійснено перетин річки. Включає різні варіанти, такі як мости, катера, пороми, помпові переправи та інші.

Отже, процес моделювання річкової переправи повинен враховувати усі об'єкти, що названі вище, та багато інших характеристик і властивостей.

2.2 Оптимізація методу річкової переправи

Розглянутий раніше метод дієвий, але не оптимізований. Проблема оптимізації полягає в тому, що не враховуються інші фактори впливу на метод реалізації переправи. Можливими перешкодами можуть бути такі фактори як тип ґрунту берегів, глибина річки, перешкоди на самій водоймі та інші чинники. Для оптимізації потрібно знайти підходящу ділянку для реалізації переправи.

По-перше, треба дослідити тип ґрунту на обох берегах річки в місці планованої переправи. Краще обрати ділянку з твердим ґрунтом - пісок, галька, глина, щоб забезпечити стійкість споруд. Болотистих ділянок слід уникати.

По-друге, варто оцінити глибину річки на цій ділянці. Занадто велика глибина ускладнить будівництво опор мосту або налаштування поромної переправи. Ідеальною є невелика глибина - 1-2 метри.

По-третє, треба врахувати наявність перешкод на воді - каменів, затонулих об'єктів, водоростей. Вони можуть заважати руху поромів або пошкодити конструкції мосту. Краще обрати чисту ділянку річки без перешкод.

По-четверте, важливо оцінити швидкість течії. Сильна течія ускладнює будівництво і експлуатацію переправи. Слід обрати ділянку з невеликою швидкістю течії.

По-п'яте, бажано вибрати прямолінійну ділянку річки, а не з крутими поворотами. Це полегшить розміщення конструкцій переправи.

По-шосте, треба врахувати ландшафт обох берегів. Краще обрати рівні ділянки, а не обривисті схили. Це полегшить будівництво під'їзних шляхів до переправи.

Отже, оптимальний варіант - це ділянка з твердими берегами, невеликою глибиною річки, без перешкод, з повільною течією, прямолінійним руслом і рівним ландшафтом. Саме така ділянка дозволить реалізувати надійну та ефективну переправу з мінімальними витратами. Провівши попереднє обстеження, можна знайти оптимальний варіант для розгортання переправи (див. рис. 2.3).

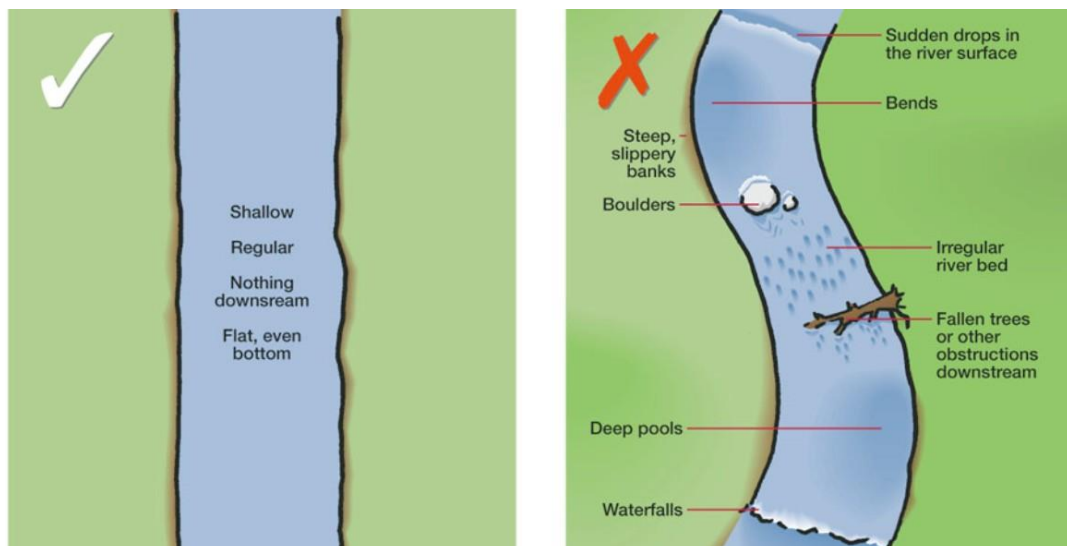


Рисунок 2.3 – Варіанти придатної та не придатної місцевості для переправи [20]

З описаної раніше інформації, яка потрібна для оптимізації методу і для максимальної ефективності підходу, будемо базуватися на багатокритеріальній оцінці варіантів переправи за наступними параметрами (див. Додаток Г), серед яких є: інтенсивність транспортного потоку, стан інфраструктури, кліматичні умови, екологічні обмеження, пропускна здатність та інше.

На основі цих даних для можливого варіанту переправи буде розраховано інтегральний критерій ефективності. Варіант з найвищим значенням критерію буде обраний як оптимальний. Такий підхід дозволить приймати виважені, обґрунтовані, оптимальні рішення щодо організації переправ, враховуючи всі важливі аспекти конкретної місцевості.

Таблиця 2.2 – Об’єкти методу річкової переправи з оптимізацією

Об’єкт	Опис
Початкова точка	Позначає місце, з якого починається шлях для перетину річки
Кінцева точка	Визначає місце, куди спрямований маршрут через водойму
Перешкода (водойма)	Представляє собою річку або іншу водну перешкоду, яку потрібно подолати
Тип переправи	Визначає метод або засіб, яким буде здійснено перетин річки. Включає різні варіанти, такі як мости, катера, пороми, помпові переправи та інші
Ефективність	Багатокритеріальна оцінка варіанту переправи за параметрами

Таким чином, для оптимізації методу до об’єктів методу річкової переправи додається ще один критерій: ефективність. Маємо нову таблицю об’єктів річкової переправи (див. таб. 2.2).

2.3 Метод мурашиного алгоритму для річкової переправи

Враховуючи попередньо оптимізований метод для здійснення переправи, можна зазначити, що цей метод не завжди повертатиме найкраще рішення. Цей метод базується на пошуку найкращої ділянки для створення переправи, проте не враховує можливість того, що початковий найкращий маршрут не обов'язково залишатиметься оптимальним протягом наступних циклів перетину річки. Іншими словами, існує ймовірність, що побудова переправи за маршрутом з меншою привабливістю на початковому етапі, може виявитися менш затратною протягом наступних циклів переправи, що в підсумку дасть кращий та ефективніший результат.

Ця проблема пов'язана з тим, що попередній метод не враховує динамічні зміни умов переправи в часі. Зокрема, під час повені рівень води може зростати, що вимагатиме зміни оптимального маршруту. Крім того, руслом річки можуть переміщуватися піщані наноси, що також впливатиме на вибір найкращої траєкторії в різні моменти часу. Отже, статичне визначення оптимального шляху на початку процесу не забезпечить глобальної оптимізації протягом усього часу функціонування переправи.

Для вирішення цієї проблеми потрібно модифікувати алгоритм таким чином, щоб він періодично перераховував оптимальний маршрут з урахуванням поточних умов. Це дозволить динамічно коригувати траєкторію переправи відповідно до змін рівня води, наносів та інших факторів. Крім того, доцільно закласти в алгоритм можливість прогнозування майбутніх умов на основі аналізу попередніх даних. Це дозволить заздалегідь підготуватися до очікуваних змін і ще більше оптимізувати процес переправи.

Гарним рішенням для модифікації методу річкової переправи виступає мурашиний алгоритм. Розглянемо спосіб переміщення та логіку мурах. Для ефективною взаємодії між комахами потрібна певна система, яка сприяє співпраці при вирішенні їхніх потреб. Непрямий спосіб взаємодії між комахами передбачає,

що представник виду модифікує своє навколишнє середовище з метою зміни поведінки інших особин. Спільне життя комах є яскравим прикладом того, як взаємодія між окремими індивідами може призвести до створення складних систем, які здатні вирішувати завдання, що були б неможливі для окремих індивідів. Один із прикладів непрямого зв'язку полягає в утворенні слідів феромону, що характерний для деяких видів мурах. Під час пошуку їжі мураха позначає свій шлях, залишаючи сліди феромону. Це призводить до того, що інші особини слідуєть за цим слідом і також займаються пошуком їжі. Принцип зміни навколишнього середовища для впливу на поведінку називається стигмергія. Цей метод є основним у системі організації мурашиних колоній, що дозволяє їм самоорганізовуватися.

Термін "самоорганізація" використовується для опису складної поведінки, яка виникає в результаті взаємодії відносно простих агентів. Завдяки самоорганізації мурахи можуть вирішувати складні завдання, з якими вони зіштовхуються щодня. Вигода самоорганізації особливо відчутна завдяки її розподіленому та ефективному характеру. Мурашині колонії можуть здійснювати розсудливу поведінку, навіть коли значна частина мурах не може окремо брати участь.

Для кращого розуміння механізму та здатності мурашиних колоній до знаходження оптимальних шляхів від гнізда до джерела їжі проводились експерименти. У одному з експериментів колонії мурах виду *Linepithema humile* з Аргентини були представлені два шляхи однакової довжини. Після певного часу спостерігали, що мурахи сходилися на один з цих шляхів, припиняючи використання альтернативних маршрутів.

Додатковий експеримент проводився на мості, де мурахи вибирали між коротким і вдвічі довшим шляхами. Це допомогло визначити, чи виберуть представники цього виду мурах найкоротший шлях серед багатьох альтернатив.

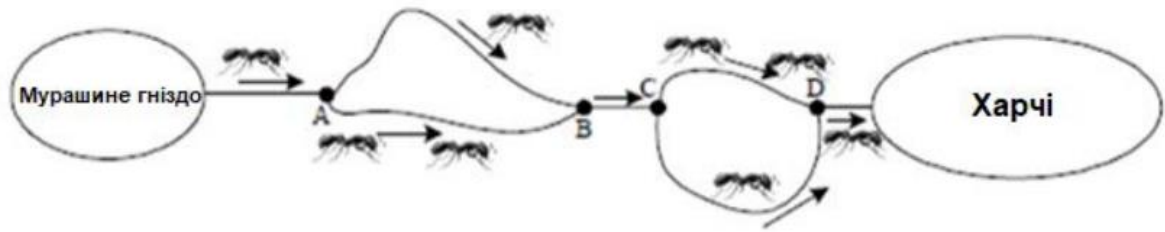


Рисунок 2.4 – Експеримент подвійного моста

Мурахи з Аргентини фактично не мають здатності до безпосереднього визначення найкоротших шляхів, оскільки вони є сліпими. Проте, результати експерименту показали, що колонія мурашок може виявити найкоротший шлях, який з'єднує гніздо з джерелом їжі.

У початковому стані всі мурахи розташовані в гнізді. Коли множина мурах вирушає в пошуках їжі, кожна з них залишає слід феромону під час проходження шляху і досягає першої точки, позначеної як А. Оскільки мурахи не мають інформації про те, який шлях є оптимальним, і жодна мураха раніше не проходила цим шляхом, то кожна з них має рівну ймовірність вибрати праворуч або ліворуч.

Після певного часу близько половини мурах обирають короткий шлях, а інші обирають довший шлях для досягнення точки В. Мурахи, які обрали короткий шлях, досягають точки В швидше і повинні вирішити, куди рухатися далі. Таким чином, виникає ситуація, коли немає жодної інформації для орієнтації. Одна частина мурах, яка досягла точки В, повертається до гнізда, в той час як інші продовжують рухатися до джерела їжі.

Однак мурахи, які обрали довший шлях, досягають точки В пізніше. Коли вони досягають точки В, вони знаходять сліди феромону, залишені мурахами, які обрали короткий шлях. Ці сліди феромону є більш інтенсивними, оскільки мурахи, які обрали короткий шлях, залишали їх частіше. Таким чином, мурахи, які обрали довший шлях, більш схильні вибрати короткий шлях при поверненні до гнізда.

Цей процес повторюється, і з часом все більше і більше мурах починає використовувати короткий шлях. В кінцевому підсумку, більшість мурах

використовує коротший шлях, і він стає основним маршрутом між гніздом і джерелом їжі.

Мурахи, які опинилися на більш довгій ділянці між пунктами А і В, досягають пункту В і також розділяються. Проте, оскільки кількість феромону на шляху назад до гнізда приблизно вдвічі більше, ніж на шляху до області з їжею, більшість мурах повертається до гнізда, прибуваючи туди одночасно з іншими мурахами, які обрали довший шлях.

Важливо відзначити, що оскільки більшість мурах вже вибрало коротший шлях між пунктами А і В порівняно з довшим, наступні мурахи, що залишають гніздо, також надають перевагу короткому шляху. Це пов'язано з тим, що мурахи слідуєть за найсильнішими слідами феромону, а коротший шлях має більш сильні сліди феромону, оскільки його використовували більше мурах.

Поведінка мурах на наступній ділянці між пунктами С і D фактично ідентична поведінці, яка була розглянута на першому мосту між пунктами А і В. В кінцевому підсумку, більшість мурах досягне кінцевого пункту і збере певну кількість їжі, щоб принести до гнізда.

По досягненню пункту D, мурахи надають перевагу коротшому шляху за тим самим принципом, як і мурахи, які почали переміщення раніше. Цей процес повторюється до тих пір, поки всі мурахи не почнуть використовувати коротший шлях.

Оскільки мурахи безперервно розподіляють феромони під час свого руху, короткий шлях постійно посилюється за допомогою все більшої кількості мурах. Це призводить до того, що величина феромону, який позначає шлях, стає настільки великою, що заглушує інші можливі варіанти маршрутів. В кінцевому підсумку, всі мурахи використовують найкоротший шлях.

Цей процес називається стигмергією. Він є прикладом самоорганізації, при якій складна поведінка виникає в результаті взаємодії відносно простих агентів.

Важливо враховувати, що феромон, який використовують мурахи, поступово випаровується з часом. Через це ділянки шляху, які не були обрані для проходження протягом певного часу, залишаються довгими і не містять

практично жодних слідів феромону. В результаті цього збільшується ймовірність того, що мурахи виберуть коротший шлях. Детальний механізм дії мурашиного алгоритму представлений на рисунку 2.5.

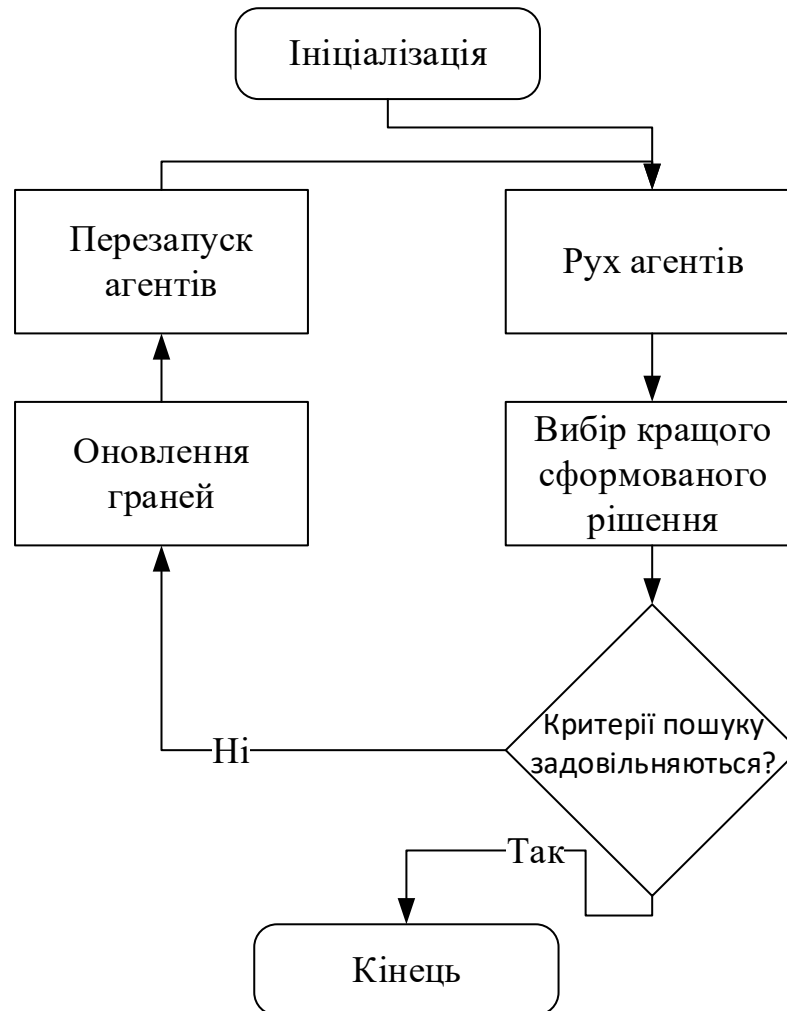


Рисунок 2.5 – Блок-схема роботи мурашиного алгоритму

Метод мурашиних систем (Ant System - AS)[21] є одним із перших методів, який використовує принцип мурашиних колоній для розв'язання задачі комівояжера (Traveling Salesman Problem, TSP) [22]. Ця задача полягає у пошуку найкоротшого шляху, який об'єднує всі пункти.

У методі мурашиних систем агенти [23], яких можна вважати за мурах, формують свої рішення, переміщуючись від одного вузла до іншого на графі рішень. Метод працює протягом певної кількості ітерацій (t_{max}). На кожній ітерації агенти формують свої рішення, виконуючи n кроків. На кожному кроці

вони використовують правило вибору наступного вузла, яке залежить від феромону, залишеного іншими мурашками на цьому вузлі.

Спочатку було запропоновано три різних методи мурашиних систем[24], які різнилися способом оновлення ребер графа. Ці методи включали щільнісний (ant-density), кількісний (ant-quantity) і циклічний (ant-cycle) методи мурашиних систем. У щільнісному і кількісному методах агенти залишали феромони в процесі формування рішення, в той час як у циклічному методі агенти залишали феромони після завершення переміщення, після вибору рішення.

Експерименти показали, що циклічний метод мав найкращу ефективність в порівнянні з іншими двома методами, тому в подальшому під назвою "метод мурашиних колоній" розуміють саме цей циклічний метод мурашиних систем (див. рис. 2.6).

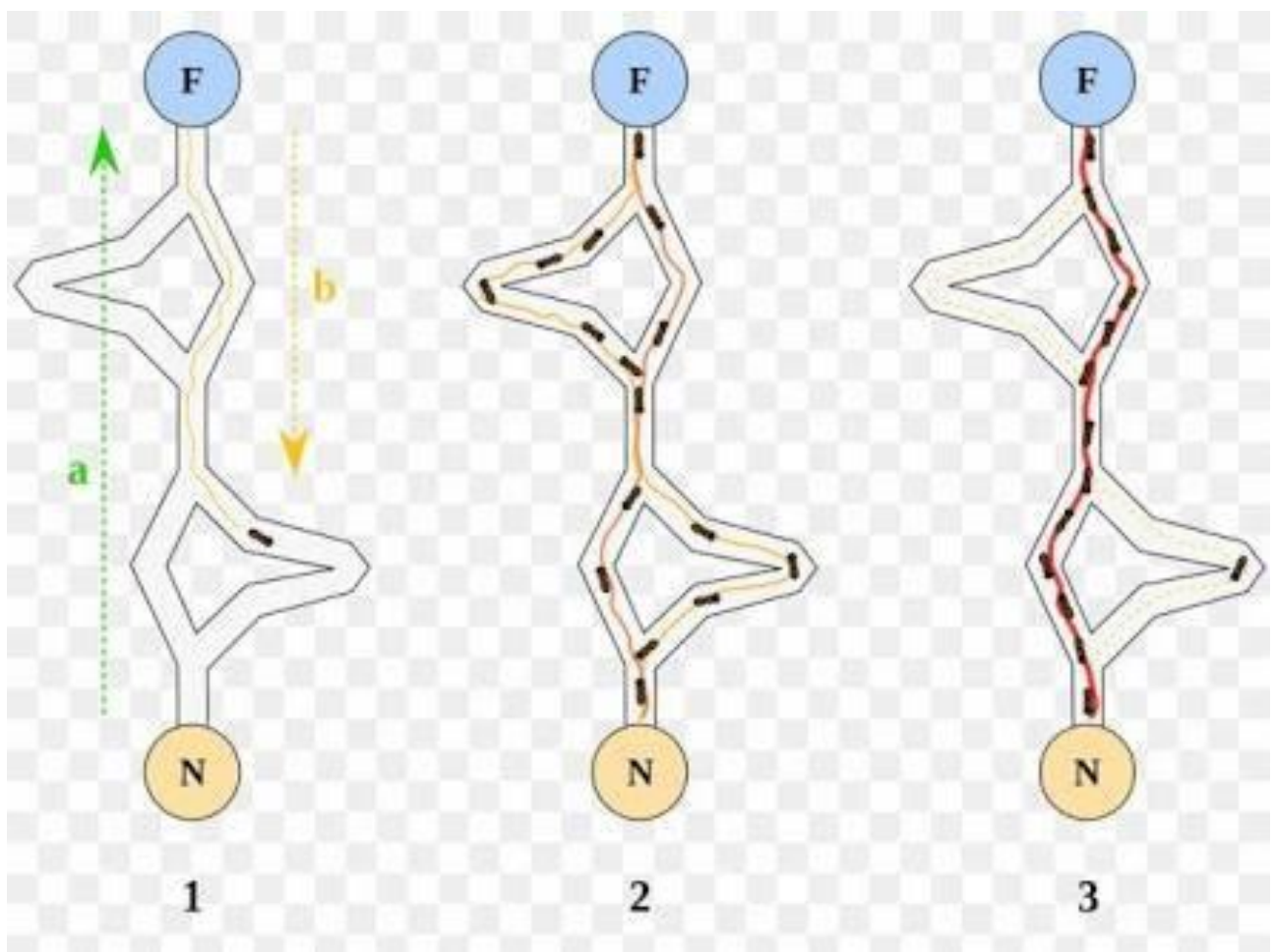


Рисунок 2.6 – Пошук кращого шляху мурашиним алгоритмом [25]

З цієї інформації бачимо, що мурашиний алгоритм (застосування феромонів) добре підходять для нашого методу. Даний алгоритм можна використовувати як первинний аналіз для створення річкової переправи, так і для детального постійного аналізу в реальному часі.

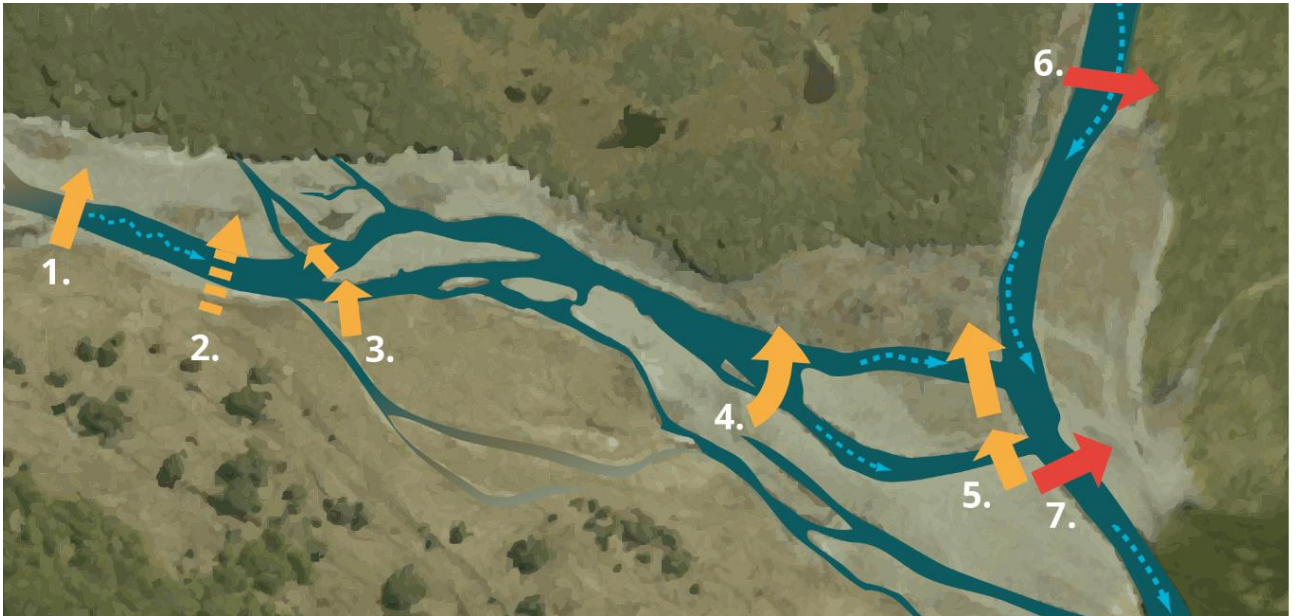


Рисунок 2.7 – Можливі маршрути для створення річкової переправи [26]

Тобто, тепер у алгоритм можна задати всі шляхи можливої річкової переправи (див. рис. 2.7), які виглядають привабливими та не привабливими, а метод з цих всіх варіантів розрахує та визначить найкращий варіант.

2.4 Інформаційна модель методу річкової переправи транспортних засобів

Для розробки інформаційної моделі методу річкової переправи транспортних засобів з використанням мурашиного алгоритму слід детально описати процеси взаємодії між сутностями.

Ідея мурашиного алгоритму базується на імітації колективної поведінки мурах [27] під час пошуку найкоротшого шляху до джерела їжі. Мурахи залишають феромонні сліди, які інші комахи можуть відчувати і йти по них. Чим

більше мурах пройшло конкретним шляхом, тим сильніший феромонний слід утворюється на ньому. Таким чином, найкоротший та найефективніший маршрут, яким скористалася більшість мурах, стає найпривабливішим завдяки найбільшій концентрації феромонів.

Цей принцип можна застосувати для оптимізації переправи транспортних засобів через річку. Розглянемо ситуацію, коли є декілька можливих маршрутів переправи з одного берега на інший. Після першої переправи якимось з обраних шляхів, на ньому залишається "феромонний слід" [28] у вигляді інформації про час, витрачені ресурси та успішність даної переправи.

Коли наступний технічний засіб повинен обрати маршрут, він аналізує наявні "феромонні сліди" і з більшою ймовірністю обирає той, де концентрація феромонів вища, тобто де переправа була найуспішнішою. Після його переправи цей слід підсилюється, стаючи ще привабливішим.

З часом феромонні сліди поступово розсіюються, дозволяючи "забувати" менш ефективні маршрути. Отже, чим більше разів якимось шляхом успішно здійснено переправу, тим сильніше він виділяється серед інших варіантів. Але при зміні умов (течія, перешкоди тощо) сліди розсіюються і починається новий цикл пошуку оптимального рішення (див. рис. 2.8).

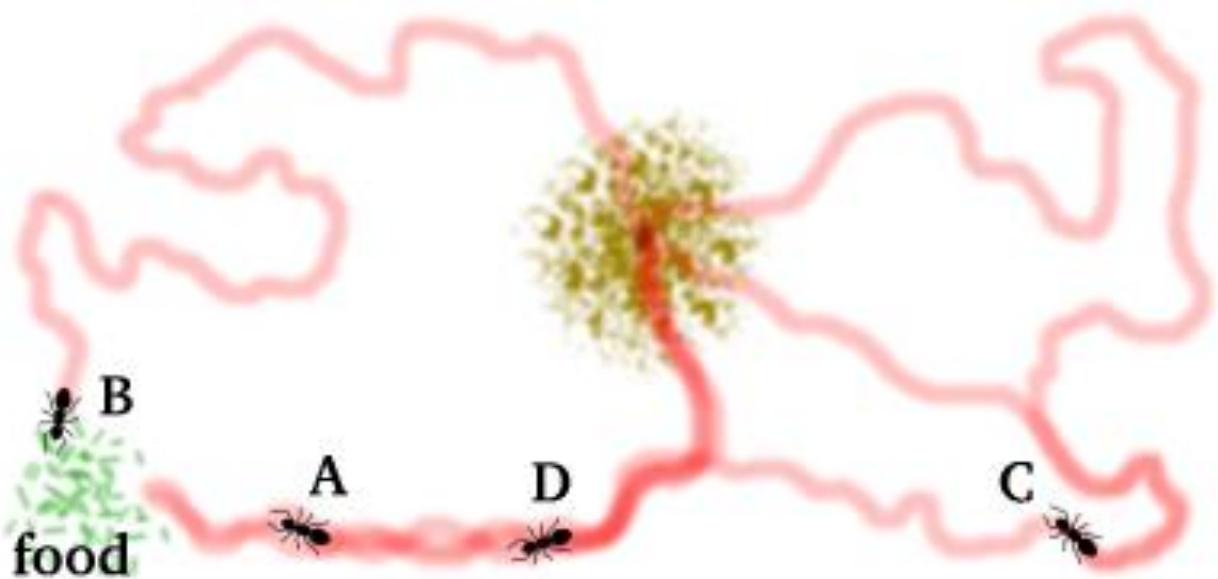


Рисунок 2.8 – Можливі маршрути для створення річкової переправи [29]

Розглянемо детальніше основні процеси такої інформаційної моделі:

- Ініціалізація маршрутів - На початку задаються усі можливі маршрути переправи між двома точками з їх характеристиками (див. Додаток Д).
- Перша переправа - Перший технічний засіб вибирає маршрут випадковим чином (або за заданими критеріями). Відбувається переправа, фіксуються фактичні показники.
- Формування феромонного сліду - На основі даних першої переправи формується феромонний слід на цьому маршруті з інтенсивністю, пропорційною ефективності переправи.
- Вибір наступного маршруту - Другий технічний засіб аналізує наявні маршрути і з більшою ймовірністю вибирає той, де інтенсивність феромонного сліду вища. Але можливий і випадковий вибір (з меншою ймовірністю) для виявлення потенційно кращих альтернатив.
- Підсилення сліду - Після переправи другого технічного засобу феромонний слід на цьому маршруті підсилюється, оскільки ефективність підтверджена. Його привабливість зростає.
- Розсіювання слідів - Через деякий час усі феромонні сліди поступово розсіюються. Швидкість розсіювання залежить від стійкості маршруту.
- Повторення процесу - Процес аналізу маршрутів, вибору, переправи, підсилення/розсіювання феромонних слідів повторюється для кожного наступного технічного засобу.
- Оновлення маршрутів - При значній зміні умов середовища (течія, руйнування доріг тощо) усі феромонні сліди обнуляються, і процес починається спочатку для пошуку нового оптимального рішення.

Таблиця 2.3 – Показники річкової переправи

Показник	Опис
Час	Час за скільки транспортний засіб за допомогою річкової переправи здійснив подолання річкової перешкоди.

	Вимірюється в секундах, хвилинах, годинах, днях, тощо. Час переправи залежить від довжини переправи, швидкості транспортного засобу, швидкості течії річки, наявності перешкод, тощо.
Ресурси	Кількість ресурсів, що були витрачені під час перетину водойми. До ресурсів можуть відноситись: пальне, мастильні матеріали, запчастини, робоча сила тощо. Витрати ресурсів впливають на вартість переправи.
Успішність	Якість здійснення переправи. Успішність переправи визначається такими факторами як: безпека переправи, неперервність переправи, пропускна здатність переправи тощо.

Отже, основними параметрами інформаційної моделі є:

- множина маршрутів переправи;
- параметри маршрутів (довжина, складність, ресурси) ;
- інтенсивність феромонних слідів;
- ймовірності вибору маршруту;
- швидкість розсіювання слідів;
- критерії ефективності переправи.

Розглянемо послідовність виконання однієї ітерації методу для створення оптимального проекту річкової переправи (див. Додаток В). У кожній ітерації початкові варіанти рішення генеруються на основі попереднього досвіду або випадковим чином. Потім вони моделюються, щоб оцінити їхню ефективність. Варіанти з найвищою ефективністю ранжуються і змінюються в певному напрямку. На основі результатів моделювання та ранжування варіантів вибирається оптимальне рішення. Це рішення перевіряється на відповідність вимогам і обмеженням. Якщо рішення не відповідає вимогам, то ітерація повторюється. Ітераційна розробка дозволяє поступово удосконалювати рішення, враховуючи результати попередніх ітерацій. Цей підхід є ефективним для

вирішення складних проблем, для яких неможливо точно визначити оптимальне рішення на початковому етапі.

Запропонована модель[30] побудови річкової переправи описує необхідні процеси, розрахунки, зв'язки та функції. Для реалізації цього проекту потрібно побудувати низку моделей та креслень на рівні конструкції, інженерних систем та ландшафту. Важливо розробити об'ємну модель мосту та відповідні креслення. Інтерфейс користувача повинен виконувати такі функції:

- відображення поточного стану ефективності;
- візуалізація розташування конструкцій;
- моделювання повені та навантажень;
- порівняльна статистика різних варіантів конструкції.

Проектування оптимальної конструкції річкової переправи є комплексним багатофакторним завданням, що вимагає системного підходу та сучасних програмних засобів. Система складається з 3 модулів (див. рисунок 2.9): інтерфейсу, застосунку, даних.

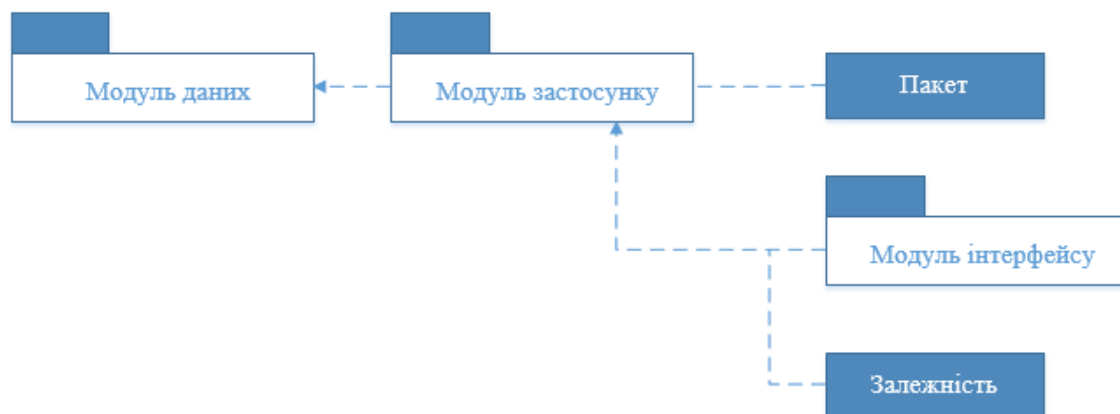


Рисунок 2.9 – Модулі інформаційної систем для взаємодії з методом

Ключовим компонентом моделі стає багаторівнева геоінформаційна модель району переправи. Вона буде включати цифрову модель рельєфу дна та

берегів, детальну гідрографічну інформацію про глибини, швидкості течії, коливання рівнів води, характеристики льодового режиму. Окремий шар міститиме дані про геологію та інженерно-геологічні властивості ґрунтів у місці переправи на різних глибинах. Ще один шар описуватиме наявну інфраструктуру, споруди, комунікації, які можуть вплинути на проект.

Наступним кроком є вибір типу переправи в інтерактивному режимі: міст, поромна переправа, тимчасовий наплавний міст, канатна дорога, понтонні мости тощо. Конструктор дозволить закласти основні параметри обраного типу.

Далі виконується детальне проектування конструкції. Для мосту це тип, кількість і розташування опор, конфігурація та розміри прогонів, ширина проїзної частини. Для поромної переправи - кількість та основні характеристики поромів. Для канатної дороги - опори, тип канатів, вагонів, пропускна здатність. Система забезпечить зручне варіювання параметрами.

Потрібно провести розрахунок навантажень та перевірок міцності за допомогою методу кінцевих елементів. Враховуватимуться постійні, змінні, кліматичні, сейсмічні та аварійні навантаження згідно норм. Окремий конструктор дозволить налаштувати вихідні дані та параметри розрахунку.

Проектування фундаментів здійснюватиметься на підставі геологічних даних та розрахунків напружень у ґрунті й осідань. Буде проаналізовано взаємовплив споруди та гідрологічного режиму річки.

Додаткові модулі дадуть змогу спроектувати системи інженерного забезпечення переправи: водопостачання, каналізацію, опалення, електропостачання, освітлення, сигналізацію, захист від корозії тощо. Також формуються звіти про обсяги робіт, специфікації матеріалів і устаткування, кошториси та графіки будівництва, виходячи з прийнятих проектних рішень.

Система дозволить порівнювати альтернативні варіанти переправ та вибирати оптимальний за сукупністю техніко-економічних критеріїв з урахуванням вимог експлуатаційної надійності та безпеки. Передбачено зручний графічний інтерфейс користувача для візуалізації проектних рішень та 3D-моделювання об'єкта в ландшафті на різних етапах проектування.

Гнучкість системи дозволить швидко оцінювати різні варіанти і знаходити оптимальне рішення, що відповідає усім технічним і економічним вимогам. Це суттєво прискорить та здешевить процес проектування переправ порівняно з традиційним підходом. Застосування такої інформаційної системи для взаємодії із методом надасть змогу реалізувати будівництво сучасних і надійних переправ, що слугуватимуть десятиліттями. UML діаграми потоків даних (information flow diagram) для проектування річкової переправи подано в додатку Д.

Для проектування оптимального рішення щодо річкової переправи потрібно розробити гнучку інформаційну систему, здатну моделювати різні варіанти: міст, поромну переправу, наплавний міст, понтонний міст, канатну дорогу тощо. Ключові об'єкти в системі:

- Переправа - визначає загальну концепцію (міст, пором, канатна дорога). Описує складові частини обраного типу переправи. Для мосту - опори, прогони, а для канатної дороги - опори, канати, вагони.

- Гідрологічна модель - характеристика русла річки.

- Транспортні засоби - у разі поромної переправи це опис поромів, для канатної - вагонів.

- Інфраструктура - допоміжні споруди (причали, дороги, лінії електропередачі).

- Моделі навантажень та геологічна модель, що враховують специфіку кожного типу переправи.

- Кошторис і календарний план також розраховуються окремо для конкретної системи.

Система програмного забезпечення для задачі річкової переправи транспортних засобів повинна включати в себе наступні компоненти:

- Управління даними. Цей компонент буде відповідати за зберігання і обробку даних про річку, транспортні засоби та інші фактори.

- Планування маршрутів. Компонент буде відповідати за планування маршрутів переправи транспортних засобів.

- Відстеження переправ. Даний компонент буде відповідати за відстеження переправ транспортних засобів і забезпечення їхньої безпеки.

- Компонент для взаємодії з користувачами. Він буде відповідати за взаємодію з користувачами, такими як оператори переправ і клієнти.

Компонент для управління даними буде відповідати за зберігання і обробку даних про річку, транспортні засоби та інші фактори. Ці дані можуть включати в себе:

- дані про річку, такі як ширина, глибина, швидкість течії та інші;
- дані про транспортні засоби, такі як габарити, швидкість і вантажопідйомність;
- дані про погодні умови, такі як рівень води, характер вітру та інші;
- дані про переправи, такі як час відправлення, час прибуття та інші.

Компонент для планування маршрутів буде відповідати за планування маршрутів переправи транспортних засобів. При плануванні маршрутів необхідно враховувати наступні фактори:

- обмеження транспортних засобів, такі як габарити, швидкість і вантажопідйомність;
- погодні умови, такі як рівень води, вітер і інші;
- динаміка навколишнього середовища, наприклад, зміна рівня води.

Компонент для відстеження переправ буде відповідати за відстеження переправ транспортних засобів і забезпечення їхньої безпеки. При відстеженні переправ необхідно враховувати наступні фактори:

- положення транспортних засобів;
- швидкість транспортних засобів;
- стан навколишнього середовища.

Компонент для взаємодії з користувачами буде відповідати за взаємодію з користувачами, такими як оператори переправ і клієнти. При взаємодії з користувачами необхідно забезпечити наступне:

- зручність використання;
- безпека;

– надійність.

Компоненти системи взаємодіють між собою і окремо з сервером. Використано сервер БД, та Апі для мап Google Maps. На рисунку 2.10 зображено діаграму потоків даних.

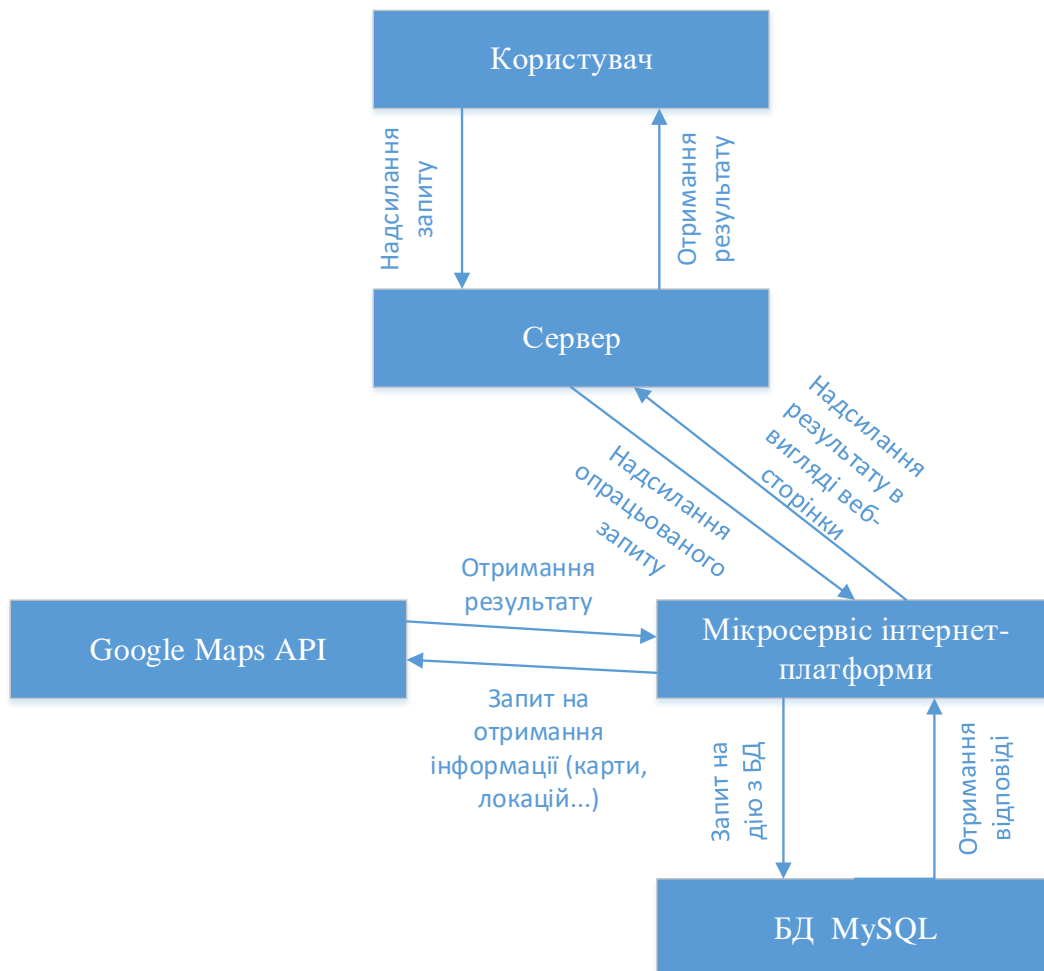


Рисунок 2.10 – Діаграма потоків даних

Діаграми потоків даних додатку показує переміщення даних між різними компонентами програмного забезпечення, для взаємодією із моделлю.

Спочатку формується набір можливих варіантів переправ - це випадкові комбінації типу конструкції, матеріалів, кількості прогонів тощо. Кожен варіант оцінюється за показниками міцності, вартості, пропускної здатності. Далі, найкращі варіанти відбираються і поєднуються для створення нових, ще ефективніших, проектів. Також застосовуються випадкові зміни параметрів -

мутації. Циклічне повторення моделювання, комбінування та мутації дозволяє послідовно поліпшувати характеристики переправи. Процес триває до досягнення оптимального проекту, який задовольняє усім вимогам за міцністю, вартістю, пропускною здатністю та іншими параметрами. Так поступово, крок за кроком, інформаційна модель наближається до ідеального варіанту переправи для конкретних умов застосування. Мурашиний алгоритм дозволяє знайти оптимальне рішення методом ітерацій. Графічно зображено модель на рис. 2.11.

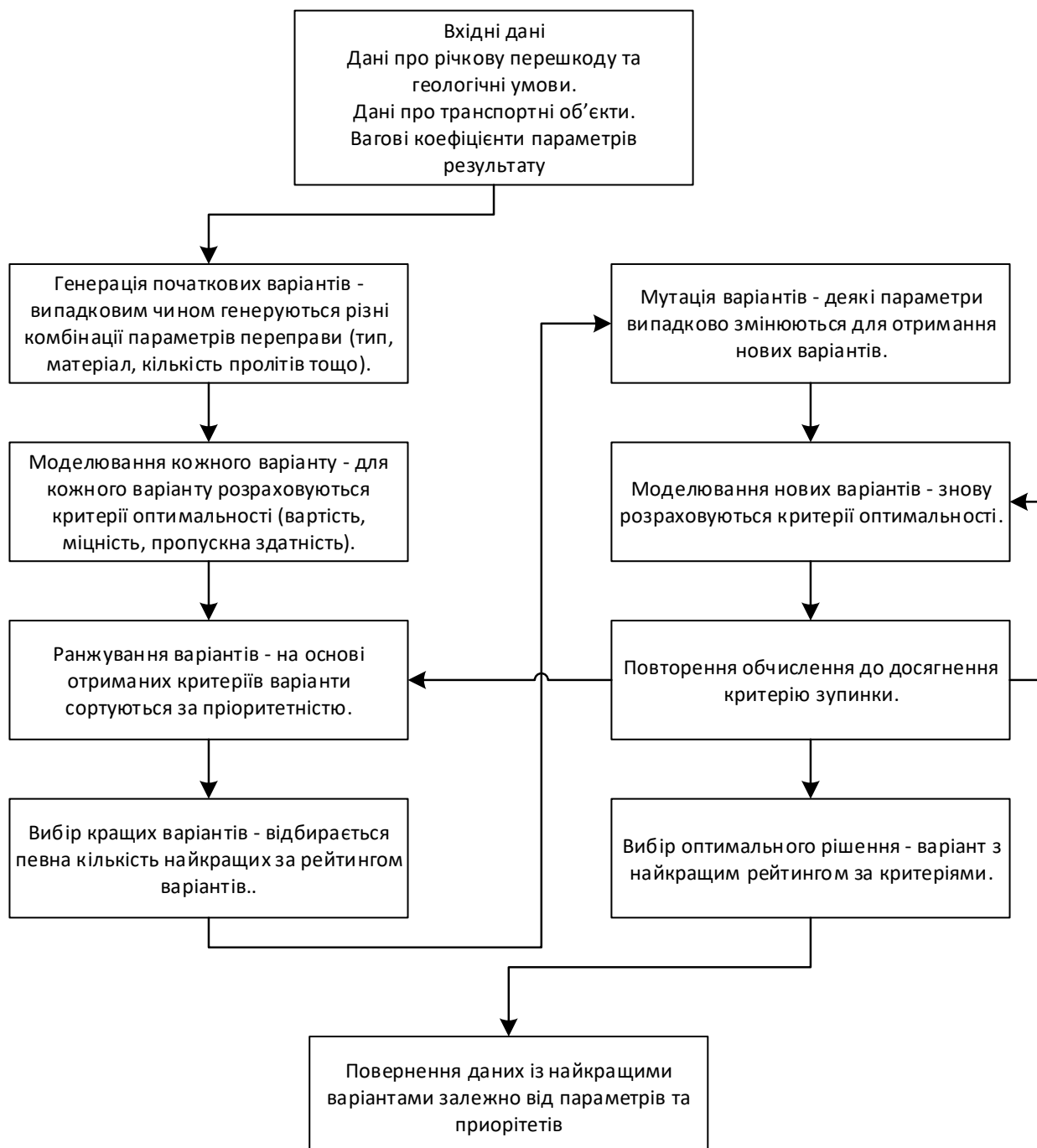


Рисунок 2.11 – Схема методу здійснення річкової переправи

Даний метод дозволяє оптимізувати вибір маршруту переправи на основі "колективного досвіду" і знайти найбільш ефективний шлях при конкретних умовах середовища. Ключовими критеріями оптимальності є мінімум витрат ресурсів, часу та максимум безпеки переправи.

Висновки до розділу 2

У даному розділі розроблено метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Розглянуто та детально описано концепцію річкової переправи, визначено основне завдання даного методу та об'єкти, які використовуються в методі.

Створено схему методу для реалізації переправи. Описано фактори, які впливають на ефективність реалізації переправи.

Опрацьовано метод на основі мурашиного алгоритму, який базується на використанні логіки мурах та ітерації для знаходження найкращого рішення для реалізації переправи.

Створено інформаційну модель річкової переправи для транспортних засобів, описано зв'язки між елементами методу в процесі опрацювання даних, подано фактори та показники, які використовуються для визначення найефективнішого способу здійснення переправи: формується набір можливих варіантів переправ; кожен варіант оцінюється за показниками міцності, вартості, пропускної здатності; найкращі варіанти відбираються і поєднуються для створення ефективніших проектів.

Циклічне повторення моделювання, комбінування та мутації дозволяє послідовно поліпшувати характеристики переправи. Мурашиний алгоритм дозволяє знайти оптимальне рішення методом ітерацій.

Розділ 3

Програмна реалізація методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

3.1 Розробка структури даних для взаємодії із моделлю

Розглянута інформаційна система планування та створення річкових переправ потребує збереження, зчитування та оновлення даних, тому виникає необхідність у базі даних. Оскільки модель рішення спроектовано як об'єктно-орієнтовану, доцільно використати реляційну базу даних. Всі сутності та їх властивості, описані раніше, легко конвертуються до даталогічного представлення (див. Додаток Б).

Нижче наведено детальний опис логічної моделі бази даних.

Таблиця Rivers призначена для зберігання даних про річки та має наступні атрибути (див. табл. 3.1).

Таблиця 3.1 – Атрибути таблиці «Rivers»

№ п/п	Назва	Тип даних	Опис
1	id_river	BIGINT	Первинний ключ
2	name	VARCHAR	Назва річки
3	length	DECIMAL	Довжина річки
4	width	DECIMAL	Ширина
5	depth	DECIMAL	Глибина
6	region	VARCHAR	Регіон розташування

Таблиця Crossings призначена для зберігання даних про заплановані переправи та має наступні атрибути (див. табл. 3.2).

Таблиця 3.2 – Атрибути таблиці «Crossings»

№ п/п	Назва	Тип даних	Опис
-------	-------	-----------	------

1	id_crossing	BIGINT	Первинний ключ
2	type	VARCHAR	Тип переправи (міст, поромна переправа тощо)
3	river_id	BIGINT	Ідентифікатор річки (зовнішній ключ з Rivers)
4	location	VARCHAR	Орієнтовне місце розташування
5	status	ENUM	Статус (заплановано, в процесі, побудовано)

Таблиця Crossing_Details призначена для зберігання детальної інформації про наявні та заплановані переправи та має наступні атрибути (див. табл. 3.3).

Таблиця 3.3 – Атрибути таблиці «Crossing_Details»

№ п/п	Назва	Тип даних	Опис
1	id_detail	BIGINT	Первинний ключ
2	crossing_id	BIGINT	Ідентифікатор переправи (зовнішній ключ з Crossings)
3	name	VARCHAR	Назва переправи
4	length	DECIMAL	Довжина переправи
5	build_date	DATETIME	Дата спорудження
6	trips_per_day	INT	Кількість рейсів на добу (для порому)

Таблиця Contractors призначена для зберігання даних про підрядників та має наступні атрибути (див. табл. 3.4).

Таблиця 3.4 – Атрибути таблиці «Contractors»

№ п/п	Назва	Тип даних	Опис
1	id_contractor	BIGINT	Первинний ключ

2	name	VARCHAR	Назва підрядника
3	address	VARCHAR	Адреса
4	phone	VARCHAR	Телефон

Таблиця Projects призначена для зберігання даних про проекти переправ та має наступні атрибути (див. табл. 3.5).

Таблиця 3.5 – Атрибути таблиці «Projects»

№ п/п	Назва	Тип даних	Опис
1	id_project	BIGINT	Первинний ключ
2	name	VARCHAR	Назва проекту
3	type	ENUM	Тип переправи
4	river_id	BIGINT	Ідентифікатор річки
5	contractor_id	BIGINT	Ідентифікатор підрядника
6	budget	DECIMAL	Кошторис
7	terms	VARCHAR	Терміни виконання
8	status	ENUM	Статус
9	crossing_id	BIGINT	Ідентифікатор переправи, що будується за проектом

Таблиця Permits призначена для зберігання даних про дозволи та узгодження щодо будівництва переправ та має наступні атрибути (див. табл. 3.6).

Таблиця 3.6 – Атрибути таблиці «Permits»

№ п/п	Назва	Тип даних	Опис
1	id_permit	BIGINT	Первинний ключ
2	project_id	BIGINT	Ідентифікатор проекту
3	permit_number	INT	Номер дозволу
4	issue_date	DATETIME	дата видачі
5	valid_thru	DATETIME	Термін дії

Така структура бази даних дозволяє зберігати всю необхідну інформацію про річки, переправи, підрядників, проекти та дозволи.

Зв'язки між таблицями дозволяють отримувати пов'язані дані, наприклад:

- інформацію про конкретну переправу разом з даними про річку, на якій вона розташована;
- дані про проект будівництва переправи разом з інформацією про річку, підрядника та саму переправу;
- перелік дозволів для конкретного проекту тощо.

Зовнішні ключі забезпечують цілісність даних та коректні зв'язки між таблицями. Для реалізації пошуку та фільтрації даних доцільно створити індекси на колонках (див. табл. 3.7).

Таблиця 3.7 – Індокси таблиць

Таблиця	Назва індексу	Опис
Rivers	Name	Пришвидшення пошуку за назвою.
Contractors	Name	
Projects	Name	
Crossings	River_id	Для пошуку пов'язаних записів.
Projects	River_id	
Crossings	Status	Для фільтрації записів за статусом.
Projects	Status	

Така структура бази даних є гнучкою і може бути легко масштабована та розширена за потреби. Наприклад, можна додати таблиці для зберігання історії змін даних, користувачів системи, журналу подій тощо.

Загалом, запропонована логічна модель забезпечує надійне зберігання всіх даних, необхідних для методу планування та створення річкових переправ, а також ефективний доступ та маніпулювання цими даними. Використання сучасної СКБД дозволить реалізувати всі необхідні функції такої системи.

На зображенні представлена архітектура системи переправи через річку. Система складається з двох шарів:

- Шар інтерфейсу відповідає за взаємодію з користувачем. Він включає в себе веб-застосунок, який дозволяє користувачеві вводити дані про річку та її береги, або скористуватися інтерфейсом для представлення інформації.

- Шар бізнес-логіки відповідає за виконання процесу переправи. Він включає в себе класи `RiverCrossing`, `RiverCrossingDTO`, і `BaseService`.

Шар інтерфейсу починає процес переправи, викликаючи метод `start()` класу `BaseController`. Цей метод створює новий об'єкт `RiverCrossing` за допомогою методу `create()` класу `RiverCrossingDTO`. Потім `BaseController` встановлює значення властивостей об'єкта `RiverCrossing` за допомогою методів: `setRiverCapacity()`, `setRiverLength()`, `setRiverDepth()`, `setFerryCapacity()` і `setPheromones()`. Нарешті, `BaseController` повертає об'єкт `RiverCrossing`.

Шар бізнес-логіки починає процес переправи, викликаючи метод `performCrossing()` класу `BaseService`. Цей метод обчислює фактори привабливості кожного берега річки за допомогою методів: `calculateShoreTypeFactor()`, `calculateDepthFactor()`, `calculateDistanceFactor()`, `calculateCapacityFactor()` і `calculateWidthFactor()`. Потім `BaseService` обирає призначення для переправи за допомогою методу `selectDestination()`. Нарешті, `BaseService` виконує переправу за допомогою методів `calculateShoreTypeFactor()`, `calculateDepthFactor()`, `calculateDistanceFactor()`, `calculateCapacityFactor()` і `calculateWidthFactor()`.

Архітектура системи є трирівневою. Шість класів, представлених на зображенні, розподілені між двома шарами. Класи у шарі інтерфейсу відповідають за взаємодію з користувачем. Два класи у шарі бізнес-логіки відповідають за виконання процесу переправи.

Архітектура системи є гнучкою та масштабованою. Класи шару інтерфейсу можуть бути легко змінені для підтримки нового інтерфейсу користувача. Два класи у шарі бізнес-логіки можуть бути легко змінені для підтримки нових алгоритмів переправи.

Переваги цієї архітектури:

– Розділення відповідальності: Класи у шарі інтерфейсу відповідають за взаємодію з користувачем, а два класи у шарі бізнес-логіки відповідають за виконання процесу переправи. Це дозволяє легко змінювати або масштабувати систему, не впливаючи на інші частини системи.

– Гнучкість: Класи у шарі інтерфейсу можуть бути легко змінені для підтримки нового інтерфейсу користувача. Два класи у шарі бізнес-логіки можуть бути легко змінені для підтримки нових алгоритмів переправи.

– Масштабованість: Система може бути легко масштабована шляхом додавання нових вузлів до шару інтерфейсу або шару бізнес-логіки.

Після розробки даної об'єктно-орієнтованої моделі було створено відповідне програмне забезпечення мовою Java. Воно дозволяє користувачеві задавати початкові параметри (характеристики річки, кількість мурах, кількість ітерацій алгоритму тощо), запускати моделювання та аналізувати побудований оптимальний маршрут переправи.

Таким чином, розроблена інформаційна система дає можливість ефективно досліджувати застосування мурашиного алгоритму для вирішення практичної задачі побудови оптимальної річкової переправи з урахуванням реальних умов. Це дозволяє вивчити переваги та недоліки даного підходу, проаналізувати його ефективність для конкретних умов застосування. Результати такого дослідження можуть бути корисними для подальшого вдосконалення мурашиних алгоритмів та їх практичного застосування для вирішення складних оптимізаційних задач.

3.3 Розробка прикладних компонентів системи

Отримана інформаційна система для взаємодії з методом для створення оптимальної річкової переправи за допомогою мурашиного алгоритму ґрунтується на взаємодії користувача з інтерактивною мапою місцевості, користувачеві також надається можливість задати початкові параметри: ширину та глибину річки, координати можливих точок переправи, наявні перешкоди

(каміння, острови тощо). На основі цих даних система моделює рух віртуальних мурах і шукає оптимальний шлях переправи.

Інтерактивна мапа відображає поточний стан моделювання: розташування мурах, їх пересування, залишені феромони, поточні шляхи. Користувач може змінювати параметри моделі, додавати/видаляти перешкоди, спостерігаючи за пошуком оптимального рішення. У процесі роботи системи на основі заданих вхідних даних (координати точок на карті, перешкоди тощо) відбувається імітація руху мурашиних агентів та пошук ними оптимального маршруту. Після завершення роботи алгоритму система будує оптимальну траєкторію переправи.

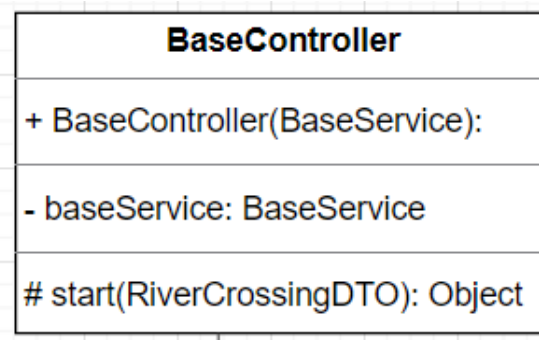


Рисунок 3.2 – Клас контролер BaseController

Клас BaseController (рис. 3.2) є Spring Boot контролером, який надає єдиний кінцевий пункт під шляхом /base/start для запуску переправи через річку. Цей контролер обробляє об'єкт RiverCrossingDTO як вхід і повертає об'єкт RiverCrossing як вихід.

Він позначений анотацією @RestController, вказуючи, що він є REST-контролером. Також має анотацію @RequestMapping("/base"), яка прив'язує контролер до шляху /base. У конструкторі цього контролера передається залежність від сервісу BaseService. В контролері існує лише один метод - start(), який анотований @PostMapping("/start") і відповідає за запуск переправи через річку. Метод start() отримує об'єкт RiverCrossingDTO як вхід, що містить необхідну інформацію для проведення переправи через річку, таку як кількість осіб і тварин для перевезення, початкові та кінцеві точки маршруту та доступні типи човнів (засобів переправи).

Після отримання вхідних даних метод викликає метод `BaseService.start()` для запуску процесу переправи через річку. `BaseService.start()` відповідає за виконання всієї логіки переправи, включаючи визначення порядку перевезення людей і тварин, а також вибір відповідних човнів (засобів переправи).

На виході метод повертає об'єкт `RiverCrossing`, який створюється в результаті виконання методу `BaseService.start()`. Об'єкт `RiverCrossing` містить інформацію про статус переправи через річку, таку як кількість перевезених осіб і тварин, кількість осіб і тварин, які все ще очікують на перевезення, та оціночний час завершення переправи.

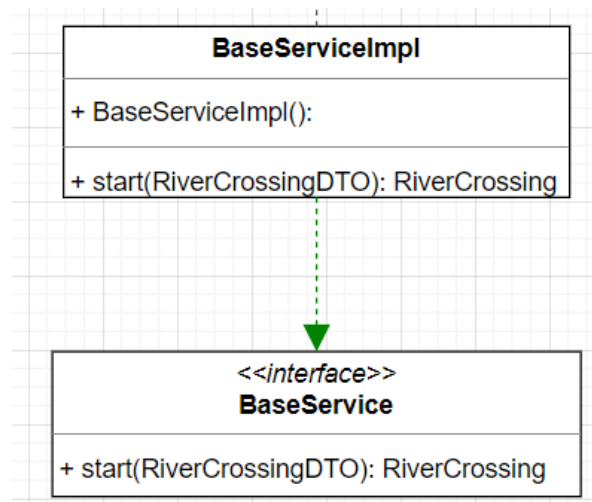


Рисунок 3.3 – Клас-сервіс та інтерфейс `BaseService`

Клас `BaseServiceImpl` є Spring Boot-сервісом, який імплементує інтерфейс `BaseService` (рис. 3.3). Інтерфейс `BaseService` визначає лише один метод - `start()`, призначений для запуску переправи через річку.

Важливі характеристики класу `BaseServiceImpl`:

- позначений анотацією `@Service`, що вказує на його призначення як Spring Boot-сервісу;
- реалізує інтерфейс `BaseService`;
- має єдиний метод `start()`, який перевизначає метод `start()` інтерфейсу `BaseService`.

Розглянемо роботу методу `start()` по кроках:

- Метод отримує об'єкт RiverCrossingDTO в якості вхідних даних. Цей об'єкт містить всю необхідну інформацію для проведення переправи через річку, таку як кількість осіб і тварин, яких потрібно перевезти, початкові та кінцеві точки маршруту та доступні типи човнів, мостів.
- Метод створює новий об'єкт RiverCrossing на основі об'єкта RiverCrossingDTO.
- Далі метод викликає метод RiverCrossing.performCrossing() для здійснення переправи через річку.
- Після цього метод виводить матрицю феромонів. Матриця феромонів містить значення, які відображають ймовірність вибору певного шляху.
- На завершення, метод повертає об'єкт RiverCrossing.

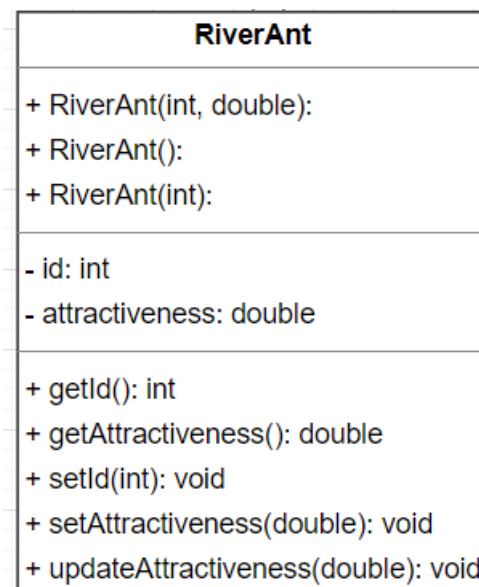


Рисунок 3.4 – Клас RiverAnt

Клас RiverAnt - це реалістична модель мурахи, яка зазвичай зустрічається на березі річки. Ця модель мурахи має важливі характеристики, що дозволяють відтворити її поведінку та вплив на інших мурах.

Ідентифікатор (id): Кожна мураха має унікальний ідентифікатор, який визначає її ідентичність серед інших мурах. Це допомагає відстежувати окремих особин та взаємодіяти з ними.

Привабливість (attractiveness): Привабливість мурахи визначає, наскільки інші мурахи схильні слідувати за нею. Ця характеристика відображає рівень впливу мурахи на своє оточення. Мурахи з вищим рівнем привабливості можуть бути більш успішними в формуванні груп та виконанні колективних завдань.

У класі `RiverAnt` доступні наступні конструктори для створення мурах:

- `RiverAnt()`: Цей безпараметричний конструктор створює нову мурашу з ідентифікатором, рівним нулю, та початковим рівнем привабливості 1,0. Це може бути використано для створення стандартної мурахи без спеціальних характеристик.

- `RiverAnt(int id)`: Конструктор приймає ідентифікатор як параметр і створює мурашу з заданим ідентифікатором та початковим рівнем привабливості 1,0. Він дозволяє створювати мурах з унікальними ідентифікаторами для подальшої ідентифікації.

У класі `RiverAnt` реалізовані наступні методи:

- `updateAttractiveness(double delta)`: Цей метод дозволяє оновити рівень привабливості мурахи на величину, задану параметром `delta`. Таким чином, мураха може змінювати свій вплив на інших мурах шляхом збільшення або зменшення свого рівня привабливості.

Клас `RiverAnt` створений для детального моделювання поведінки мурах на березі річки, зокрема, їхньої ідентичності та впливу на інших. Він надає можливість вивчати та аналізувати різноманітні аспекти поведінки цих комах, що є важливим для досліджень в біології та екології.

Клас `RiverCrossing` (Додаток E) є центральним класом моделі, що інкапсулює всі дані та алгоритми, необхідні для моделювання процесу переправи. Він містить такі основні поля:

- `ants` - список об'єктів класу `RiverAnt`. Кожен об'єкт представляє одну мурашу і має унікальний ідентифікатор (поле `id`).

- `pheromones` - двовимірний масив, що зберігає поточний рівень концентрації феромонів на кожному з можливих шляхів між парами берегів. Використовується мурахами для вибору оптимального шляху.

- `shoreTypes` - масив типів рядків, що зберігає тип берега (наприклад, "піщаний") для кожного з берегів. Впливає на привабливість шляху.

- `riverLengths` - масив довжин річки біля кожного берега. Використовується для розрахунку загальної відстані між берегами.

- `riverDepths` - масив глибин річки біля кожного берега. Впливає на привабливість шляху.

- `riverWidths` - аналогічно, масив ширин річки біля берегів.

- `ferryCapacity` - вантажопідйомність порома на кожному березі. Впливає на вибір шляху.

Конструктор класу `RiverCrossing` ініціалізує всі ці поля на основі даних, переданих в об'єкті класу `RiverCrossingDTO`. Зокрема, він:

- створює список мурах відповідно до кількості, заданої в `RiverCrossingDTO`;

- викликає метод `initPheromones` для ініціалізації матриці феромонів нулями;

- присвоює значення параметрів річки з відповідних полів `RiverCrossingDTO`.

Основний метод `performCrossing()` реалізує ітераційний процес моделювання переправи за алгоритмом:

- Виконує задану кількість ітерацій.

- На кожній ітерації: перебирає всіх мурах по черзі, для кожної мурахи: викликає метод `selectDestination()`, щоб обрати берег призначення, Оновлює рівень феромонів на шляху до обраного берега за допомогою методу `updatePheromones`.

- Переміщує мурашу на обраний берег.

- Після переміщення всіх мурах викликає метод `evaporatePheromones` для випаровування феромонів.

Таким чином, на кожній ітерації мурахи переправляються з одного берега на інший, залишаючи феромони і поступово концентруючи їх на оптимальних шляхах.

Метод `selectDestination()` обирає берег призначення для конкретної мурахи на поточній ітерації. Він базується на обчисленні ймовірностей переходу до кожного з можливих берегів. Ці ймовірності залежать від:

- кількості феромонів на шляху між поточним та цільовим берегом;
- Факторів привабливості цього шляху: відстань, типи берегів, глибина, ширина річки, вантажопідйомність порома.

Для обчислення цих факторів використовуються відповідні методи класу `RiverCrossing`. На основі отриманих ймовірностей обирається цільовий берег з використанням генератора випадкових чисел.

Методи обчислення факторів привабливості шляху:

- `calculateDistance()` - просто повертає відстань між берегами на основі їх індексів та масиву `riverLengths`.
- `calculateShoreTypeFactor()` - на основі комбінації типів берегів відправлення та призначення повертає числовий фактор привабливості. Використовує евристичний підхід з заданими коефіцієнтами для різних пар типів.
- `calculateDepthFactor()` та `calculateWidthFactor()` - обчислюють фактор на основі глибини/ширини річки біля берега відправлення та вантажопідйомності порома на березі призначення. Використовується логарифмічна формула для плавної зміни.
- `calculateCapacityFactor()` - аналогічно використовує вантажопідйомності поромів на обох берегах.

Щоб уникнути різких небажаних змін у вазі ребер графу, які використовуються для вибору оптимального маршруту, всі фактори, що впливають на ці ваги, обмежуються певним діапазоном значень. Зокрема, встановлено границі 0.6 та 1.2 для значень факторів. Нижня межа 0.6 гарантує, що фактори не можуть занадто сильно впасти, що призвело б до невинного знехтування деяких потенційних маршрутів. А верхня границя 1.2 не допускає занадто великого зростання факторів, яке надало б надмірної переваги певним шляхам. Таке обмеження створює умови для плавної зміни ваг ребер графу в залежності від параметрів задачі, таких як глибина, ширина річки та

вантажопідйомність поромів. При цьому унеможливаються різкі стрибки ваги в будь-якому напрямку, що забезпечує стабільний та поступовий пошук оптимального розв'язку.

Метод `updatePheromones` оновлює рівень феромонів на шляху після переходу мурахи. Кількість доданих феромонів залежить від довжини шляху - чим довший шлях, тим менше феромонів залишає мураха. Це стимулює вибір коротших шляхів.

Метод `evaporatePheromones` виконує випаровування феромонів на шляхах - зменшує їх концентрацію на фіксований відсоток. Це дозволяє уникнути великого накопичення феромонів і перезапуску пошуку оптимального шляху.

Клас `RiverAnt` досить простий, містить лише поле `id`, конструктор та геттери/сеттери. Необхідний для представлення мурах з унікальними ідентифікаторами.

`RiverCrossingDTO` - клас-ДТО (Data Transfer Object), використовується для передачі даних між компонентами програми. Містить поля для всіх даних моделі.

В цілому клас `RiverCrossing` демонструє використання багатьох ОО принципів та патернів:

- інкапсуляція даних та алгоритмів моделі всередині класу;
- поліморфізм при обчисленні факторів привабливості шляху;
- метод `selectDestination()` - приклад шаблону Стратегія, дозволяє легко змінити алгоритм вибору шляху;
- `RiverCrossingDTO` - приклад шаблону DTO;
- `RiverAnt` можна розглядати як сутність доменної моделі;
- класи розбиті на логічні одиниці з чіткими зв'язками, що дозволяє легко модифікувати та розширювати модель.

Загалом, реалізовано гнучку та ефективну модель переправи мурах з використанням ОО підходу, що дозволяє досліджувати різні стратегії та умови. Код зручний для подальшого розвитку та експериментів з алгоритмом.

Також для реалізації використано бібліотеки, зокрема для роботи з мапою та візуального представлення даних. Також використовуються бібліотеки для

моделювання логіки взаємодії об'єктів, тобто черг, автомобілів, автомобільних пробок та інших ситуацій, які можуть вплинути на результат переправи.

Реалізовані класи дозволяють повноцінно змоделювати всі етапи роботи мурашиного алгоритму та побудувати оптимальну траєкторію переправи з урахуванням усіх вхідних обмежень та умов.

3.4 Аргументація вибору засобів розробки методу

Java - це популярна мова програмування, яка використовується для розробки веб-додатків, десктопних додатків і інших типів програмного забезпечення. Вона має ряд переваг, які роблять її хорошим вибором для розробки системи ПЗ для задачі річкової переправи транспортних засобів:

- Модульність. Java є модульною мовою, що означає, що її можна розділити на окремі модулі. Це робить її більш гнучкою і легко підтримуваною.
- Портативність. Java-код можна виконувати на будь-якій платформі, яка має віртуальну машину Java (JVM). Це робить її ідеальною для розробки програмного забезпечення, яке потрібно запускати на різних платформах.
- Безпека. Java має вбудовані функції безпеки, які допомагають захистити програмне забезпечення від атак.
- Стабільність. Java є стабільною мовою, яка має довгу історію використання.

Ось кілька конкретних переваг використання Java для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів:

- Java має широкий спектр бібліотек і фреймворків, які можна використовувати для реалізації різних функцій системи. Наприклад, можна використовувати бібліотеки Java для роботи з БД, обробки даних і веб-розробки.
- Java-код можна легко протестувати і налагодити. Це важливо для забезпечення безпеки і надійності системи.
- Java-код можна легко підтримувати. Це важливо для забезпечення того, щоб система могла працювати в довгостроковій перспективі.

Звичайно, у Java є і деякі недоліки, які слід враховувати при виборі мови програмування для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів. Наприклад, Java може бути менш ефективною, ніж деякі інші мови програмування. Крім того, Java може бути складніше вивчити, ніж деякі інші мови програмування. В цілому, Java - це хороший вибір для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів. Вона має ряд переваг, які роблять її більш гнучкою, переносимою, безпечною і стабільною.

Spring - це фреймворк для Java, який спрощує розробку веб-додатків, десктопних додатків і інших типів програмного забезпечення. Він має ряд переваг, які можуть покращити розробку системи програмного забезпечення для задачі річкової переправи транспортних засобів:

- Автоматизація. Spring автоматизує багато поширених завдань розробки, таких як ін'єкція залежності, управління конфігурацією і взаємодія з базами даних. Це може заощадити час і зусилля розробників.

- Гнучкість. Spring є дуже гнучким фреймворком, який можна адаптувати до різних потреб. Це дозволяє розробникам створювати системи, які відповідають їхнім конкретним вимогам.

- Безпека. Spring має вбудовані функції безпеки, які допомагають захистити програмне забезпечення від атак.

- Стабільність. Spring є стабільним фреймворком, який має довгу історію використання.

Ось кілька конкретних переваг використання Spring для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів:

- Spring може використовуватися для реалізації різних функцій системи: управління даними, планування маршрутів і відстеження переправ;

- Spring може використовуватися для створення масштабованої і надійної системи;

- Spring може використовуватися для створення системи, яка відповідає вимогам безпеки.

Звичайно, у Spring є і деякі недоліки, які слід враховувати при виборі фреймворку для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів. Наприклад, Spring може бути більш складним, ніж деякі інші фреймворки. Крім того, Spring може бути більш дорогим, ніж деякі інші фреймворки. В цілому, Spring - це хороший вибір для розробки системи ПЗ для задачі річкової переправи транспортних засобів. Він має ряд переваг, які можуть покращити розробку і полегшити підтримку системи.

Ось кілька прикладів того, як Spring можна використовувати для розробки системи ПЗ для задачі річкової переправи транспортних засобів:

- Для управління даними можна використовувати Spring Data. Spring Data - це модуль фреймворка Spring, який спрощує роботу з базами даних.
- Для планування маршрутів можна використовувати Spring Boot. Spring Boot - це модуль фреймворка Spring, який спрощує створення веб-додатків.
- Для відстеження переправ можна використовувати Spring Cloud. Spring Cloud - це модуль фреймворка Spring, який дозволяє створювати масштабовані і надійне програмне забезпечення.

Вибір конкретного модуля Spring для реалізації певної функції системи буде залежати від конкретних вимог до системи програмування.

Для розробки системи програмного забезпечення для задачі річкової переправи транспортних засобів використовують такі патерни програмування:

- Для управління даними можна використовувати патерн Data Access Object (DAO). DAO - це патерн, який дозволяє відокремити код доступу до даних від коду бізнес-логіки. Це може полегшити тестування і підтримку системи.
- Для планування маршрутів можна використовувати патерн Strategy. Strategy - це патерн, який дозволяє використовувати різні алгоритми для вирішення однієї і тієї ж проблеми. Це може зробити систему більш гнучкою і масштабованою.
- Для відстеження переправ можна використовувати патерн Observer. Observer - це патерн, який дозволяє одному об'єкту повідомляти інші об'єкти про зміни в своєму стані. Це може допомогти зробити систему більш скоординованою.

Також використано сторонній сервіс для роботи з онлайн-мапами. API карт Google Maps - це популярний інструмент для створення інтерактивних мап від компанії Google. Він надає розробникам широкі можливості для налаштування та інтеграції картографічного функціоналу в свої додатки. Додавання карт Google дозволяє покращити досвід користувача, надаючи йому не лише адреси чи координати, а й візуальний контекст у вигляді інтерактивної мапи.

Вибрана мова програмування та бібліотеки добре підходять для реалізації розробленого методу та інформаційної системи. Вони дозволяють реалізувати усі необхідні функції та вимоги до системи з урахуванням особливостей предметної області. Застосовані архітектурні патерни та підходи до побудови таких систем дозволяють створити надійне, масштабоване та безпечне рішення. Використання сучасних технологій розробки дає можливість швидко реалізувати необхідну функціональність та в подальшому легко розширювати систему новими можливостями. Загалом обрані технології та підходи цілком задовольняють вимоги до розробки та експлуатації методу.

Висновки до розділу 3

У третьому розділі описано реалізацію веб-додатку методу здійснення річкової переправи транспортних засобів за мурашиними алгоритмом. Було розроблено інформаційну технологію та модель для вирішення задачі оптимізації логістики при плануванні та проведенні річкової переправи. Для цього було побудовано моделі предметної області, що дозволили структурувати дані та процеси. На основі моделей розроблено архітектуру майбутньої системи.

Обрано об'єктно-орієнтований підхід до програмної реалізації, який дозволить гнучко розширювати функціонал. Для зберігання даних спроектовано базу даних з необхідними сутностями та зв'язками.

Для розробки системи вибрано мову Java, фреймворк Spring та API від Google для роботи з мапами.

Розділ 4

Дослідження методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

4.1 Прикладне тестування методу

Для дослідження коректності роботи методу створення річкової переправи за допомогою мурашиного алгоритму було розроблено набір тестів. Ці тести дозволяють перевірити працездатність алгоритму в різних умовах - коли річка має один чи два описаних береги, коли відстань між берегами дуже мала чи дуже велика, для різної швидкості течії тощо.

Опис кейсу 1: Введення некоректних значень вхідних параметрів, відсутні значення для опису другого берегу, або другий берег задано, але не вказано або не подано усіх параметрів берегу. Приклад помилки (див. рис. 4.1).

Таблиця 4.1 – Тест-кейс 1

Тест-кейс №: 1	Пріоритет: 1	Створено: 14.11.2023, Охота В. В.
Назва: Перевірка коректності вхідної інформації		
Кроки		Очікуваний результат
1. Запустити веб-застосунок 2. Передати у контролер невірні значення іншого берега 3. Порівняти очікуваний результат із фактичним		Отримати повідомлення помилку про некоректність даних
Результат проходження тест-кейсу: успішний		

```

Pretty  JSON  [Icons]
1
2      "ErrMsg": "Помилка введення даних!
3      Некоректне значення другого берегу.
        Помилка параметру типу берегу."

```

Рисунок 4.1 – Приклад помилки вхідних даних

Опис кейсу 2: Перевірка логіки роботи методу зі мурашиним алгоритмом. Задаємо вірну інформацію для початку обрахунку логіки та перевіряємо мапу феромонів на коректність. Приклад мапи феромонів (див. рис. 4.2).

Таблиця 4.2 – Тест-кейс 2

Тест-кейс №: 2	Пріоритет: 1	Створено: 14.11.2023, Охота В. В.
Назва: Перевірка коректності вхідної інформації		
Кроки		Очікуваний результат
1. Запустити веб-застосунок 2. Передати у метод реалізації мурашиного алгоритму вірні значення двох берегів та річки 3. Порівняти очікуваний результат зі фактичним		Отримати мапу феромонів руху мурах
Результат проходження тест-кейсу: успішний		

Pheromones [0][0]: Infinity
 Pheromones [0][1]: 0,02
 Pheromones [0][2]: 0,01
 Pheromones [0][3]: 0,00
 Pheromones [0][4]: 0,00
 Pheromones [1][0]: 0,02
 Pheromones [1][1]: 0,00
 Pheromones [1][2]: 0,00
 Pheromones [1][3]: 0,00
 Pheromones [1][4]: 0,00
 Pheromones [2][0]: 0,01
 Pheromones [2][1]: 0,00
 Pheromones [2][2]: 0,00
 Pheromones [2][3]: 0,00
 Pheromones [2][4]: 0,00
 Pheromones [3][0]: 0,00
 Pheromones [3][1]: 0,00
 Pheromones [3][2]: 0,00
 Pheromones [3][3]: 0,00
 Pheromones [3][4]: 0,00
 Pheromones [4][0]: 0,00
 Pheromones [4][1]: 0,00
 Pheromones [4][2]: 0,00
 Pheromones [4][3]: 0,00
 Pheromones [4][4]: 0,00

Рисунок 4.2 – Мапа інтенсивності феромонів, сформована в результаті роботи програмної реалізації мурашиного алгоритму оптимізації маршруту

Опис кейсу 3: Перевірка взаємодії з зовнішнім інтерфейсом для роботи з онлайн картою Google-Maps, перевірка ініціалізації мапи, а саме взаємодії з ендпойнтами Console Google-Maps-API. Працюючий інтерфейс взаємодії з мапою та зовнішніми сетами інформації для опрацювання методом(див. рис. 4.3).

Таблиця 4.2 – Тест-кейс 3

Тест-кейс №: 3	Пріоритет: 1	Створено: 14.11.2023, Охота В. В.
Назва: Перевірка взаємодії з google maps api		
Кроки		Очікуваний результат

<ol style="list-style-type: none"> 1. Запустити клієнт для взаємодії з програмним застосунком 2. Викликати метод для роботи з мапою 3. Порівняти очікуваний результат зі фактичним 	<p>Відкриття працюючого інтерфейсу роботи з мапою</p>
<p>Результат проходження тест-кейсу: успішний</p>	

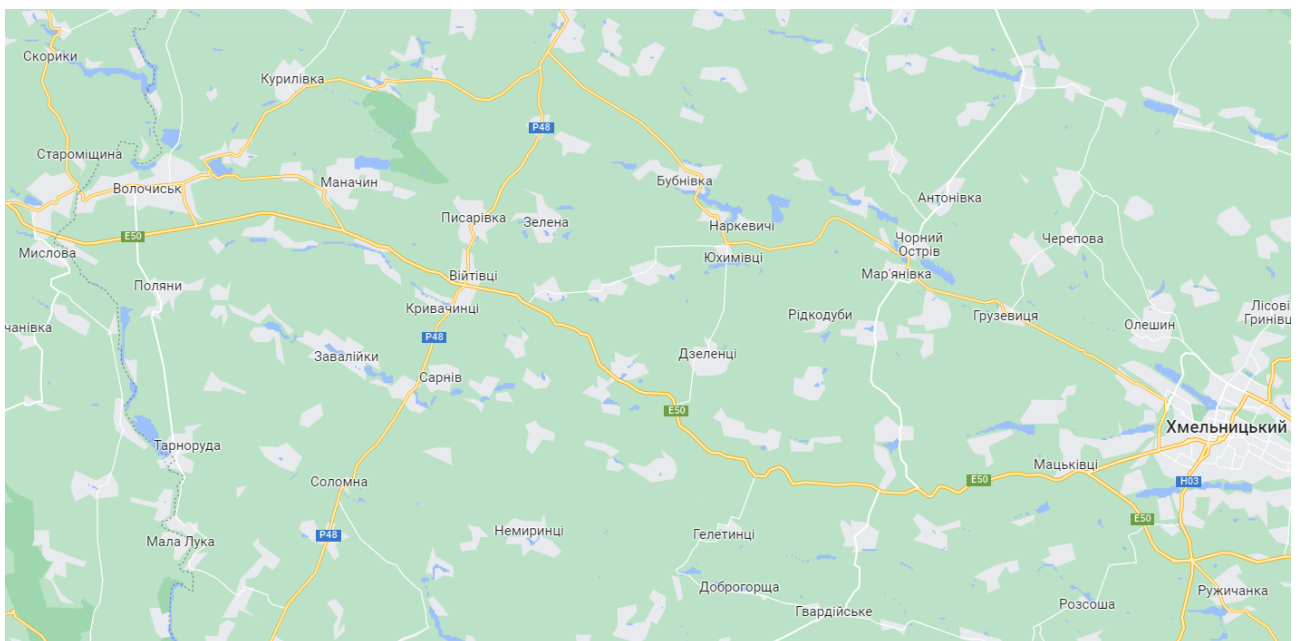


Рисунок 4.3 – Працюючий інтерфейс для взаємодії з мапою Google та вибірками інформації для роботи методу

Отже, як результат, створено множину тест-кейсів та здійснено прикладне тестування системи із використання методу реалізації річкової переправи на основі мурашиного алгоритму, тим самим доведено функціональну працездатність та коректність роботи бізнес-процесів методу та прикладних компонентів системи.

Перевірено взаємодії з зовнішнім інтерфейсом для роботи з онлайн картою Google-Maps, перевірено ініціалізацію мапи.

4.2 Функціональне дослідження та визначення ефективності методу здійснення річкової переправи транспортних засобів

Для дослідження ефективності застосування мурашиного алгоритму для створення річкової переправи для транспортних засобів було обрано ряд інструментів. Створено функціонал для запуску шаблонів користувачем, які представляють імітацію створення річкової переправи між двома берегами. Цим самим з'являється можливість провести дослідження у рівних умовах, з використанням мурашиного алгоритму так і без нього. Створений шаблон буде імітувати штучно викликане перевантаження транспортної системи, у даному випадку річкової переправи. Також у користувача буде можливість примусово зупинити та відновити процес імітації для спостереження різниці у вирішенні навантаження та регулювання його. Для побудови графіків створено інформаційне табло для відображення привабливості для створення переправи, час створення переправи, час для здійснення перетину річки по готовій переправі.

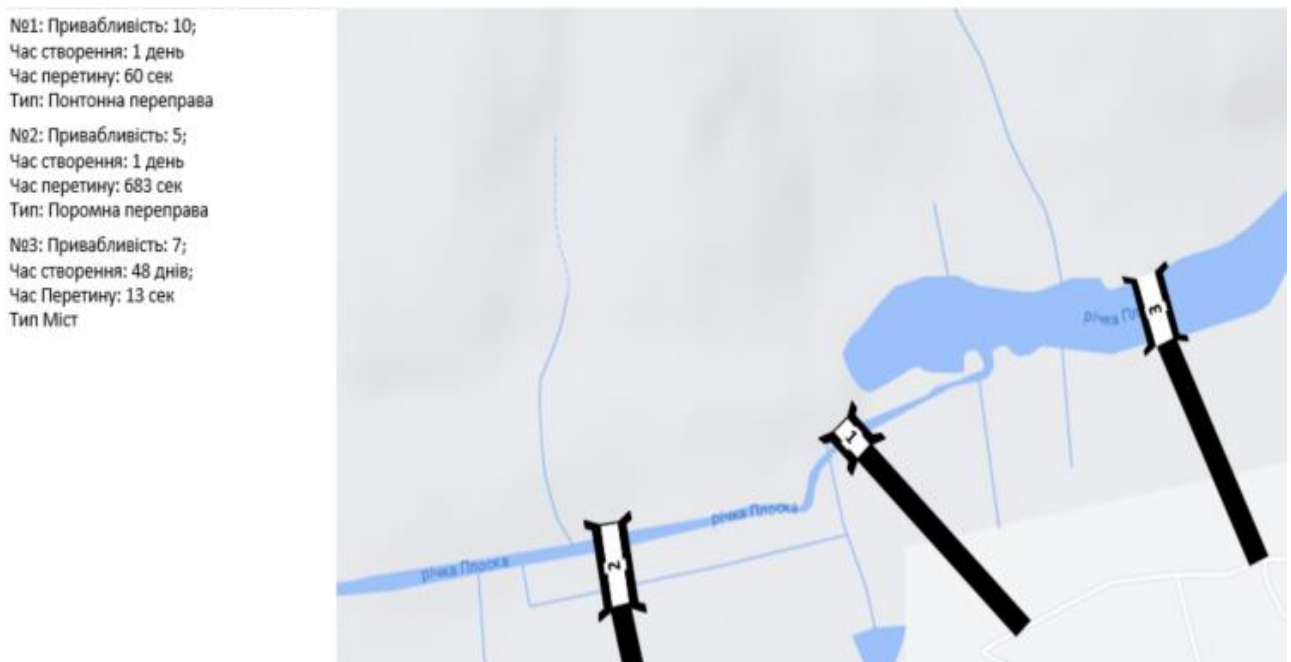


Рисунок 4.4 – Приклад отримання даних про переправи

На рисунку 4.4 зображено результат роботи методу та загалом програми, було наведено 3 можливих рішення створення річкової переправи за такими параметрами як: вага транспорту, пропускна здатність, відстань до найближчої дороги та параметрів водойми, час експлуатації переправи. Розраховано найкращий варіант №1 та наведено дані про результати роботи алгоритму.

Щоб краще оцінити ефективність роботи методу, необхідно впровадити стандартний шаблон чи процедуру, оскільки сам процес запуску цієї системи не надає достатньої інформації для оцінки її продуктивності. Такий шаблон допоможе у систематизації процесу та забезпечить можливість аналізу та порівняння різних запусків системи для оцінки її роботи. Порівнюємо жадібний алгоритм з мурашиним алгоритмом. Для порівняння виберемо ідентичні локації, запусимо алгоритми та порівняємо результати обох на ідентичних вхідних даних. Вхідні дані подано у таблицях 4.3 – 4.6. Для дослідження використано умову переправлення 10 легкових автомобілів вагою від 2,5 тон до 3,5 тон та набір різних параметрів, які впливають на роботу алгоритму: параметри транспорту, річкові перешкоди тощо.

Таблиця 4.3 – Вхідні дані про транспортні засоби

№	Ширина, м	Висота, м	Колісна база, м	Коля, м	Кліренс, м	Повна вага, т	Вага на вісь, т	Центр ваги, м	Радіус повороту, м
1	1,8	1,5	2,7	1,6	0,18	2,5	1,25	0,9	5,5
2	1,85	1,6	2,8	1,65	0,2	2,7	1,35	1	6
3	1,9	1,65	2,85	1,7	0,22	3	1,5	1,05	6,5
4	1,95	1,7	3	1,75	0,25	3,2	1,6	1,1	7
5	1,9	1,65	2,9	1,7	0,23	2,8	1,4	1,04	6,7
6	1,92	1,68	2,95	1,73	0,24	3,1	1,55	1,07	6,8
7	1,94	1,7	3	1,75	0,26	3,3	1,65	1,1	7
8	1,96	1,75	3,05	1,78	0,28	3,4	1,7	1,15	7,5
9	1,98	1,8	3,1	1,8	0,3	3,5	1,75	1,2	8
10	1,92	1,68	2,95	1,73	0,24	2,6	1,3	1,07	6,8

Таблиця 4.4 – Вхідні дані про річкову перешкоду

Параметр	Значення
Довжина річки	480 метрів
Ширина річки	60 метрів
Глибина річки	4 метрів
Швидкість течії	1 метри за секунду
Напрямок течії	північно-східний
Дно річки	кам'янисте
Береги річки	пологі, порослі очеретом
Погода	помірна

Таблиця 4.5 – Вхідні дані про берегові точки реалізації переправи

Ділянка	Тип берега	Довжина, метрів	Глибина, метрів	Ширина, метрів
1	піщаний	500	9	85
2	глиняний	200	4	18
3	піщаний	150	3,5	14,5
4	глиняний	180	4,2	16
5	піщаний	120	3,2	15,5

Таблиця 4.6 – Вхідні дані про переправу

Параметр	Значення
Тип	міст
Вантажопідіймність	45 тонн
Пропускна спроможність	40 транспортних засобів за годину
Пріоритетність	Час здійснення переправлення транспортного засобу

Розпочнемо з жадібного алгоритму. Жадібний алгоритм, приймаючи рішення на кожному кроці шляху, вибрав найкоротший шлях без урахування можливості економії ресурсів на майбутніх етапах. Це означає, що алгоритм обрав оптимальний варіант лише для поточного етапу, не враховуючи можливих переваг або збережень ресурсів, які могли б виникнути на подальших переходах через ту саму переправу.

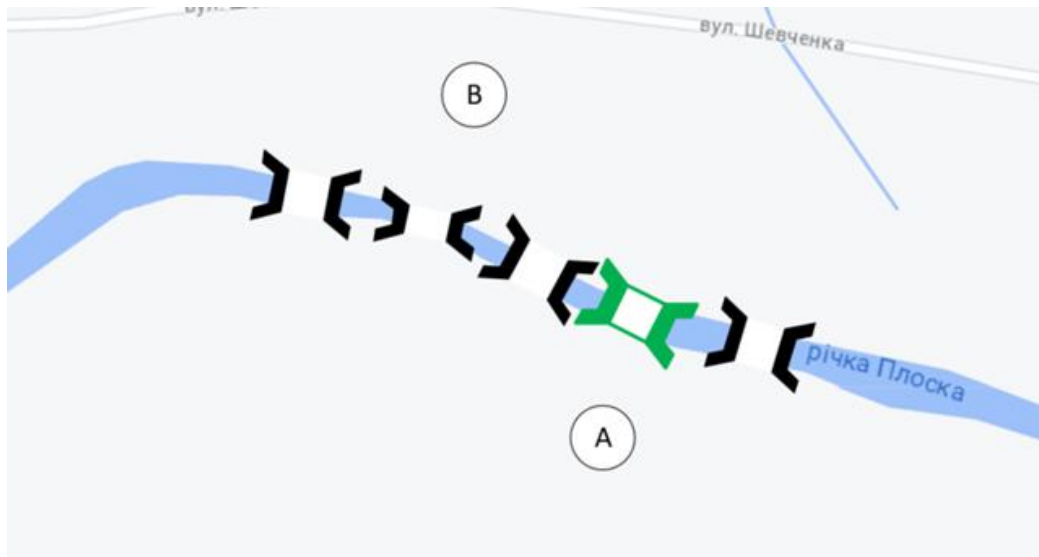


Рисунок 4.5 – Вибір маршруту жадібним алгоритмом

Графічне представлення результатів методу жадібного алгоритму (див. рис. 4.5). Результати системи наступні: час створення переправи 48 днів, витрата часу для переправи по готовому мосту складає в сумі 245 сек, час перетину по кожному транспортному засобу подано в таб. 4.7.

Таблиця 4.7 – Таблиця часу перетину за жадібним алгоритмом

№ транспортного засобу	Час перетину в секундах
1	25
2	27
3	29
4	28
5	26

6	24
7	22
8	20
9	23
10	21

Далі використаємо програмне забезпечення SAGA GIS для моделювання території та плагіни для знаходження оптимального маршруту для створення мосту, та плагін для моделювання використання моста, графічне представлення вибору маршруту (див. рис. 4.6).



Рисунок 4.6 – Вибір маршруту ПЗ SAGA GIS

Дане програмне забезпечення не має можливості для аналізу часу створення переправи, плагін розрахований лише для переправ у вигляді мостів, модулятор переправи повертає час перетину мосту об'єктами, сумарний час переправи всіх об'єктів складає 270 сек, час по кожному об'єкті представлено в таблиці (див. таб. 4.8).

№ транспортного засобу	Час перетину в секундах
1	21
2	25

3	31
4	37
5	28
6	25
7	22
8	19
9	24
10	18

Таблиця 4.8 – Час перетину за допомогою ПЗ SAGA GIS

Виконаємо запуск шаблону розробленого методу для створення річкової переправи на основі мурашиного алгоритму. Алгоритм опираючись на всі вхідні дані вибрав маршрут, який є не найпростіший, але при наступних ітераціях перетину транспортними засобами має кращі значення. Вибір маршруту для річкової переправи зображено на рисунку 4.7.



Рисунок 4.7 – Вибір маршруту за допомогою ПЗ SAGA GIS

Результати системи наступні: час створення переправи 49 днів, витрата часу для перетину по готовому мостові складає в сумі 240 сек, час перетину по одному транспортному засобі подано в табл. 4.9.

Таблиця 4.9 – Час перетину транспортним засобом річкової переправи за допомогою розробленого методу

№ транспортного засобу	Час перетину в секундах
1	26
2	28
3	24
4	29
5	27
6	24
7	22
8	21
9	20
10	19

Отже, за даними із таблиць результатів виконання програмних продуктів можна порівняти час, який знадобиться транспортним засобам для перетину водойми, рисунок 4.8.

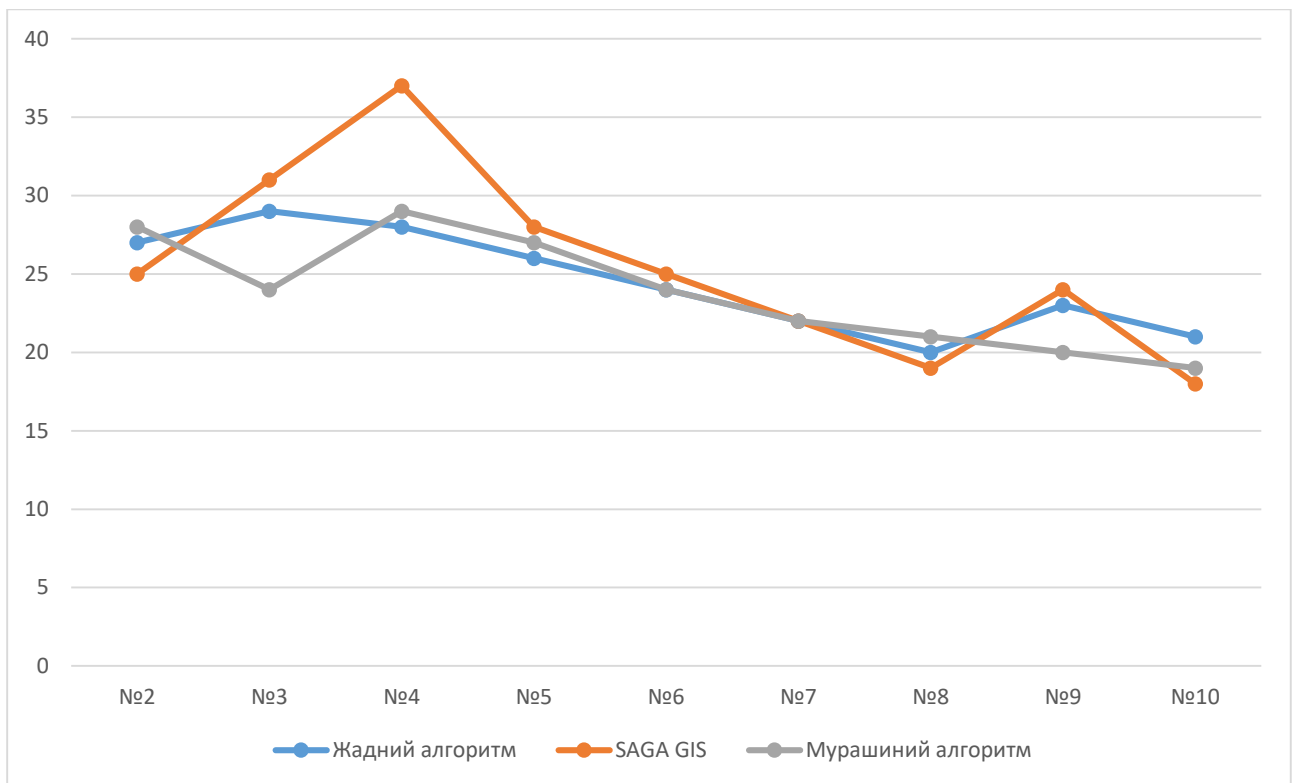


Рисунок 4.8 – Графік часу перетину транспортними засобами річкової переправи за різними алгоритмами

Даний графік відображає результати ефективності розробленого методу порівняно з іншими. Мурашиний алгоритм виявився особливо ефективним, постійно покращуючи свій результат з кожною ітерацією, що відрізняє його від інших алгоритмів. Середні часові показники склали: мурашиний алгоритм - 24 секунди, жадібний алгоритм – 24,5 секунди, програма SAGA GIS - 27 секунди. Ефективність розробленого методу в порівнянні з жадібним алгоритмом складає – 2,4%, з програмою SAGA GIS – 4,1%. В середньому ефективність методу становить: 3,25%.

Крім уже розглянутих критеріїв, методів з даними алгоритми можна також порівняти за розкидом значень. Розкид - це різниця між максимальним і мінімальним значеннями в наборі даних. Отже, новий набір даних буде виглядати так таблиця.

Таблиця 4.10 – Розкид значень для різних алгоритмів

№ транспортного засобу	Жадібний алгоритм	SAGA GIS	Мурашиний алгоритм
1	18	16	19
2	19	17	20
3	21	20	22
4	22	21	23
5	23	23	24
6	24	24	25
7	25	26	26
8	26	27	27
9	27	29	28
10	28	30	29

Аналізуючи отримані в ході експерименту дані, можна помітити, що розроблений метод ефективно обчислює оптимальні рішення, демонструючи широкий діапазон можливих значень між мінімальними та максимальними показниками, графічне відображення результатів представлено на рисунку 4.9.

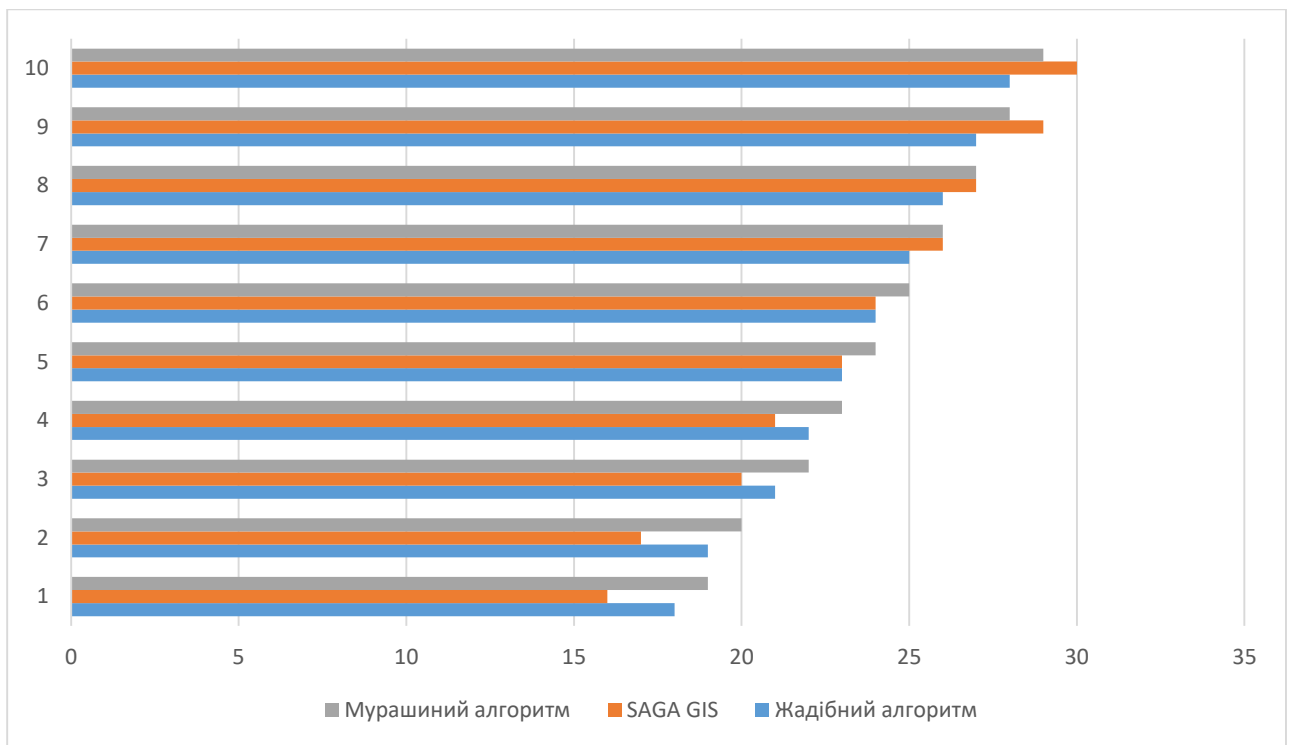


Рисунок 4.9 – Діаграми розкиду значень за різними алгоритмами

Ще один критерій для порівняння методів - це середньоквадратичне відхилення значень від середнього (див. формулу 1). Середньоквадратичне відхилення (СКВ) показує, наскільки в середньому значення даних відхиляються від середнього значення сукупності даних.

$$\sigma_{\bar{x}} \approx S_{\bar{x}} = \frac{s}{\sqrt{n}} \quad (4.1)$$

Чим менше СКВ, тим меншим є розкид даних і тим ближче вони групуються біля середнього. Тому СКВ можна використовувати для порівняння алгоритмів - чим менше воно в алгоритму, тим стабільніші і точніші результати він дає. Менше СКВ означає, що алгоритм має менший розкид результатів навколо середнього і, отже, є більш надійним. Дані порівняння алгоритмів представлено в таблиці 4.11.

Таблиця 4.11 – Середньоквадратичне відхилення методів

№ транспортного засобу	Жадібний алгоритм	SAGA GIS	Мурашиний алгоритм
1	22	20	23
2	23	21	24
3	20	23	25
4	27	24	26
5	26	27	27
6	29	29	29
7	29	30	30
8	28	32	31
9	30	33	32
10	31	26	28

Отже, аналізуючи наведені показники для трьох алгоритмів, можна зробити такі висновки:

- За середнім значенням мурашиний алгоритм (27.5) та SAGA GIS (27.5) показали однаковий результат, який дещо перевищує середнє значення для жадібного алгоритму (26.5).

- За показником середньоквадратичного відхилення мурашиний алгоритм показав кращий результат (3.41640), ніж SAGA GIS (4.04145) та жадібний (3.02765). Це означає, що значення для мурашиного алгоритму є більш згрупованими навколо середнього порівняно з іншими алгоритмами.

- Середньоквадратичне відхилення мурашиного алгоритму на 2.6% менше, ніж у SAGA GIS. Це свідчить про те, що за стабільністю значень мурашиний алгоритм перевершив SAGA GIS.

Отже, за результатами аналізу статистичних показників можна зробити висновок, що мурашиний алгоритм демонструє кращі результати порівняно з

алгоритмами SAGA GIS та жадібним алгоритмом, графічне представлення результатів наведено на рисунку 4.10.

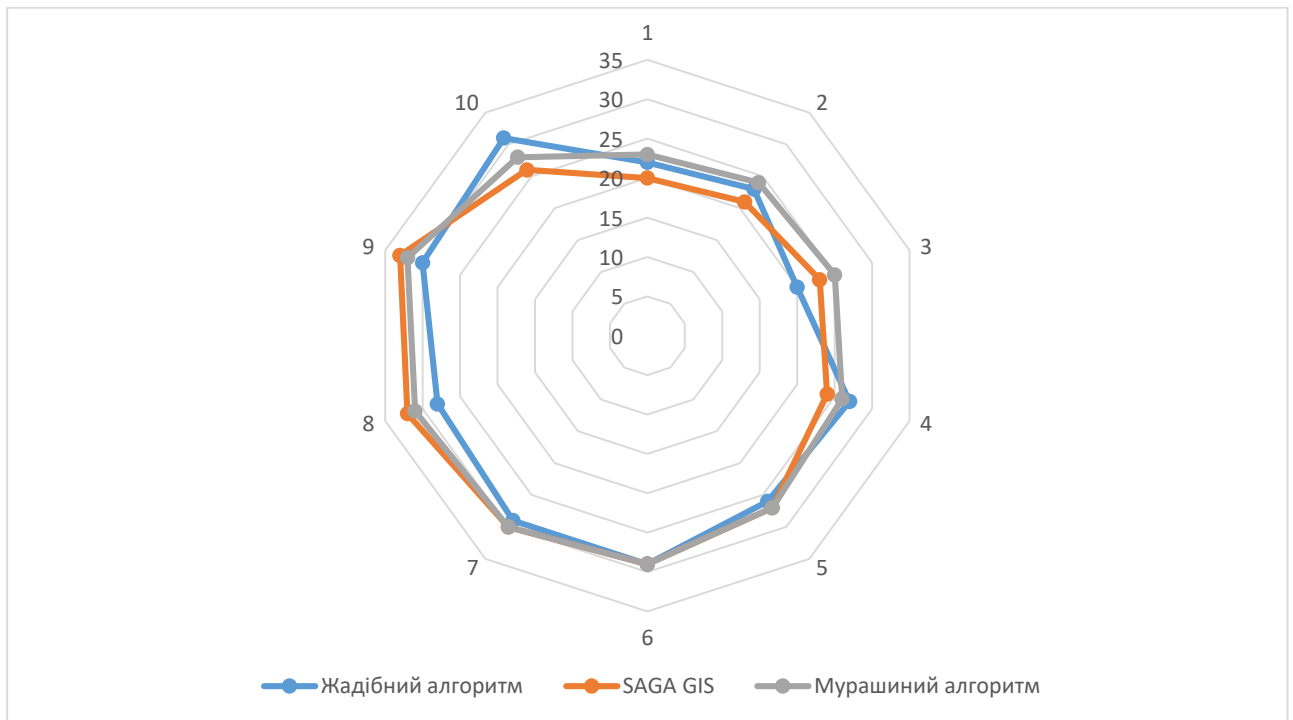


Рисунок 4.10 – Графік середньоквадратичного відхилення

Для наступного дослідження визначено завдання: побудувати тимчасову переправу для техніки комбінованого типу: 5 танків, 10 бронетранспортерів, 18 вантажівок, 25 легкових автомобілів. Річка Дніпро в Херсонській області — широка та глибока, глибина до 7-10 м, ширина в цьому районі сягає понад 800 м, швидкість течії 1,5-2 м/с, дно піщане з глибокими ямами. Береги переважно високі, скелясті. Головні параметри для кращого потрібного результату: час реалізації завдання, пропускна здатність.

Таблиця 4.12 – Результати реалізації завдання алгоритмами

Параметр	Мурашиний алгоритм	Жадібний алгоритм	SAGA GIS
Тип	Поромна переправа	Переправа повітряними поромами	Комбінований варіант з поромів і човнів

Максимальне навантаження	60 тон	80 тон	50 тон
Час реалізації	60 годин	120 годин	96 годин
Надійність	Достатня	Висока	Середня
Пропускна спроможність	45 транспортних засобів/год	40 транспортних засобів/год	38 транспортних засобів/год

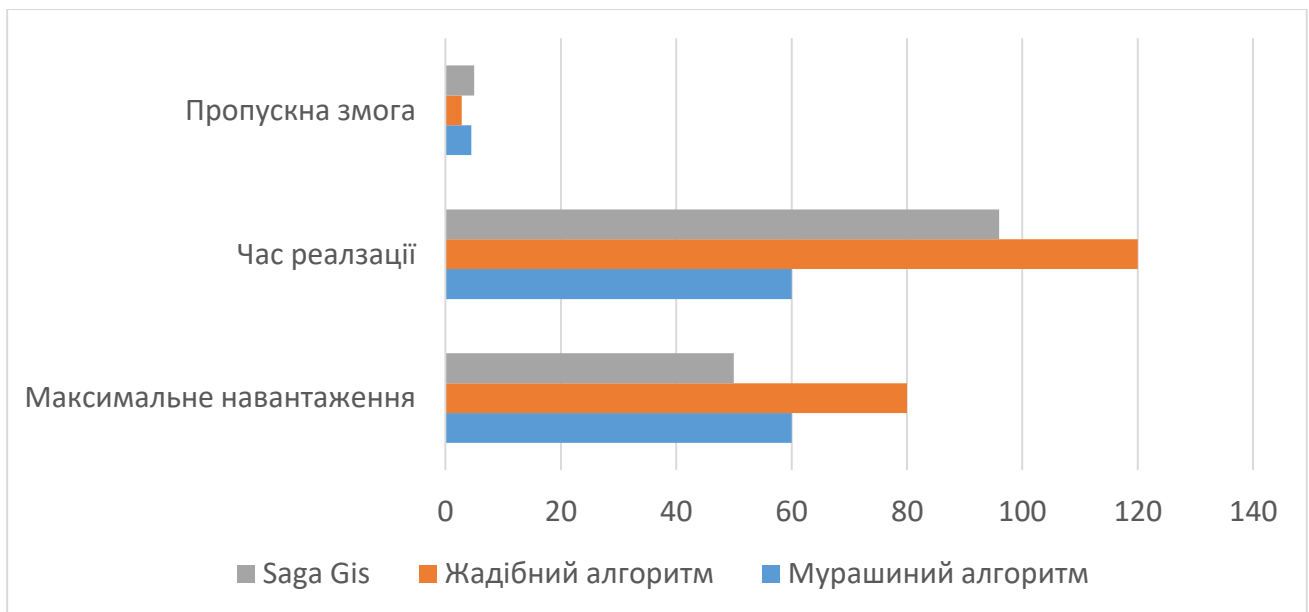


Рисунок 4.11 – Діаграма реалізації завдання за різними алгоритмами

Отже, виходячи з пріоритетних критеріїв, саме мурашиний алгоритм показує найкращі результати для даної задачі організації переправи (див рис. 4.11). Окрім того, варто зазначити, що розроблена система має низку переваг порівняно з подібними. Зокрема, вона дозволяє здійснювати розрахунок часу створення переправи, визначати оптимальний тип переправи (міст, пором, канатна дорога тощо) залежно від запланованого терміну експлуатації мосту, максимального сумарного навантаження транспорту, інтенсивності руху, відстані до найближчої інфраструктури (комунікацій, доріг). Інформаційна система розроблена з метою забезпечення можливості моніторингу технічного стану готових переправ та розрахунку їх ефективності й обмежень в режимі реального часу. Такий підхід дозволяє приймати виважені рішення щодо проектування, будівництва та експлуатації переправ з урахуванням усіх важливих факторів.

Висновки до розділу 4

В даному розділі проведено дослідження методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.

Було розроблено набір тестових сценаріїв для верифікації роботи системи. Перевірку пройшли коректність інтерфейсу, функціональність основних модулів, а також відповідність результатів роботи мурашиного алгоритму теоретичним очікуванням.

З метою порівняльного аналізу ефективності запропонованого підходу було реалізовано модуль для побудови маршрутів за жадібним алгоритмом та за допомогою програми SAGA GIS. Проведено серію експериментів на тестових даних, за результатами яких побудовано графіки залежності тривалості маршруту та інших показників від обраних параметрів.

Отримані результати демонструють переваги використання мурашиного алгоритму для оптимізації маршруту річкової переправи з урахуванням особливостей місцевості. Це дозволяє зменшити витрати ресурсів та часу на реалізацію проекту.

Загальні висновки

В результаті виконання кваліфікаційної роботи розроблено метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.

В процесі розробки методу були вирішені наступні задачі:

- проведено аналіз проблеми проектування оптимального маршруту річкової переправи. Розглянуто основні аспекти, проблеми та фактори впливу, такі як: місцевість, рельєф берегів, ширина та глибина річки, швидкість течії, тип ґрунту, наявність інфраструктури, погодні умови, сезонні коливання рівня води, пропускна спроможність, вантажопотік, вид транспорту, терміни будівництва, вартість тощо. Проаналізовано взаємний вплив цих факторів;

- досліджено існуючі методи вирішення подібних задач, виявлено їх сильні та слабкі сторони. Проаналізовано недоліки та обґрунтовано необхідність розробки нового ефективного методу на основі мурашиного алгоритму, який дозволить комплексно врахувати більшість важливих факторів;

- розроблено метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Запропонований метод побудови оптимального маршруту переправи враховує усі впливові фактори. Описано процедури ініціалізації, конструювання маршрутів, оцінки їх якості, оновлення феромонів тощо. Розроблено схему взаємодії основних компонентів: вхідні дані, обмеження, критерії оптимальності, алгоритмічна реалізація, результати моделювання;

- розроблено веб-додаток, що втілює запропонований метод оптимізації маршруту переправи на мові Java з використанням сучасних фреймворків та бібліотек. Проведено тестування на адекватність результатів;

- проведено валідацію розробленого методу на тестових наборах даних. Для перевірки адекватності результатів роботи алгоритму здійснено його порівняння з альтернативними методами: жадібним алгоритмом, інструментами геоінформаційної системи. Побудовано графіки залежності тривалості та вартості маршруту, переправ від різних факторів;

– експериментально підтверджено переваги мурашиного алгоритму для даної задачі. Він дозволяє знаходити близький до оптимального за сукупністю критеріїв маршрут з урахуванням геометричної складності місцевості та інших обмежень. Ефективність проведення річкової переправи із застосуванням розробленого методу збільшується на 2-3% в залежності від різних факторів.

Розроблений метод і програмне забезпечення можуть застосовуватися для оптимізації реальних проектів будівництва річкових переправ, дозволяючи знаходити ефективне рішення за прийнятний час та зменшувати витрати ресурсів.

Перспективним є розширення бібліотеки алгоритмів оптимізації маршрутів для порівняння ефективності мурашиного алгоритму з іншими методами на різних тестових сценаріях. Застосування мурашиних алгоритмів для оптимізації інших транспортних задач, таких як: прокладання оптимальних маршрутів наземного транспорту, планування маршрутів міського громадського транспорту, оптимізація вантажних перевезень, пошук найкоротших шляхів у складних логістичних мережах, розподіл транспортних потоків по альтернативних маршрутах

Отже, у кваліфікаційній роботі магістра запропоновано і розроблено ефективний метод оптимізації маршруту річкової переправи транспортних засобів з використанням мурашиного алгоритму.

Перелік посилань

1. Ford (crossing) [Електронний ресурс]. – Режим доступу: [https://en.wikipedia.org/wiki/Ford_\(crossing\)](https://en.wikipedia.org/wiki/Ford_(crossing))
2. Bridge [Електронний ресурс]. – Режим доступу: <https://en.wikipedia.org/wiki/Bridge>
3. How to Ford a Dangerous River [Електронний ресурс]. – Режим доступу: <https://theadventurerr.com/adventures/how-to-ford-a-dangerous-river-3-ways-to-safely-cross-a-river-or-stream/>
4. Water taxi [Електронний ресурс]. – Режим доступу: https://en.wikipedia.org/wiki/Water_taxi
5. Tunnel [Електронний ресурс]. – Режим доступу: <https://en.wikipedia.org/wiki/Tunnel>
6. Гідроцикл [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/%D0%93%D1%96%D0%B4%D1%80%D0%BE%D1%86%D0%B8%D0%BA%D0%BB>
7. Будівництво мостів : етапи, типи мостів [Електронний ресурс]. – Режим доступу: <https://virakran.com/budivnytstvo-mostiv-etapy-typu-mostiv/>
8. Переправа [Електронний ресурс]. – <https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%BF%D1%80%D0%B0%D0%B2%D0%B0>
9. Overhead line crossing [Електронний ресурс]. – Режим доступу: https://en.wikipedia.org/wiki/Overhead_line_crossing
10. Амфібіфія [Електронний ресурс]. – <https://uk.wikipedia.org/wiki/%D0%90%D0%BC%D1%84%D1%96%D0%B1%D1%96%D1%8F>
11. Підводний тунель [Електронний ресурс]. – https://uk.wikipedia.org/wiki/%D0%9F%D1%96%D0%B4%D0%B2%D0%BE%D0%B4%D0%BD%D0%B8%D0%B9_%D1%82%D1%83%D0%BD%D0%B5%D0%BB%D1%8C

12. Водний режим [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/%D0%92%D0%BE%D0%B4%D0%BD%D0%B8%D0%B9_%D1%80%D0%B5%D0%B6%D0%B8%D0%BC

13. Yang X.-S., He X.-S. Introduction to Optimization. SpringerBriefs in Optimization. Cham, 2019. С. 1–20. URL: https://doi.org/10.1007/978-3-030-16936-7_1 (дата звернення: 08.11.2023).

14. OpenBridge Designer CONNECT Edition. [Електронний ресурс]. – Режим доступу: <https://www.reisoftwareth.com/products-detail/OpenBridge/OpenBridge-Designer/>

15. Jili Tao, Ridong Zhang, Yong Zhu (2021). DNA Computing Based Genetic Algorithm Zhu Y., Zhang R., Tao J. DNA Computing Based Genetic Algorithm: Applications in Industrial Process Modeling and Control. Springer Singapore Pte. Limited, 2021. 274 с. URL: https://books.google.com.ua/books?id=uT2CzwEACAAJ&newbks=0&hl=uk&redir_esc=y (дата звернення: 11.11.2023).

16. Application of the ant colony optimization in the resolution of the bridge inspection routing problem. [Електронний ресурс]. – Режим доступу: <https://www.sciencedirect.com/science/article/abs/pii/S1568494618300401>

17. Ant-Inspired Metaheuristic Algorithms for Combinatorial Optimization Problems in Water Resources Management. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/370376156_Ant-Inspired_Metaheuristic_Algorithms_for_Combinatorial_Optimization_Problems_in_Water_Resources_Management

18. Introduction to Ant Colony Optimization [Електронний ресурс]. – Режим доступу: <https://www.geeksforgeeks.org/introduction-to-ant-colony-optimization/>

19. River Safety [Електронний ресурс]. – Режим доступу: <https://www.mountainsafety.org.nz/learn/skills/river-safety/>

20. How to Ford a River or Stream [Електронний ресурс]. – Режим доступу: <https://www.artofmanliness.com/skills/outdoor-survival/ford-river-stream/>

21. Okwu M. O., Tartibu L. K. Grasshopper Optimisation Algorithm (GOA). Metaheuristic Optimization: Nature-Inspired Algorithms Swarm and Computational Intelligence, Theory and Applications. Cham, 2020. С. 95–104. URL: https://doi.org/10.1007/978-3-030-61111-8_10 (дата звернення: 02.11.2023).
22. The natural genius of ants is helping us build better algorithms [Електронний ресурс]. – Режим доступу: <https://www.wired.co.uk/article/ant-algorithms-nature>
23. Soofastaei A. Application of Ant Colony Optimization. IntechOpen, 2022. С. 65–74. URL: https://books.google.com.ua/books?id=d5VwEAAAQBAJ&newbks=0&hl=uk&redir_esc=y (дата звернення: 02.11.2023).
24. Fidanova S. Ant Colony Optimization and Applications. Cham: Springer International Publishing, 2021. С. 35–56. URL: <https://doi.org/10.1007/978-3-030-67380-2> (дата звернення: 04.11.2023).
25. Ant colony optimization algorithms Mathematical optimization, ants, insects, ant png [Електронний ресурс]. – Режим доступу: <https://www.pngegg.com/en/png-naxug>
26. Rivers and river crossing [Електронний ресурс]. – Режим доступу: <https://wilderlife.nz/in-the-mountains/river-crossing/>
27. Ant Colony Optimization based algorithm to determine the optimum route for overhead power transmission lines [Електронний ресурс]. – Режим доступу: <https://www.sciencedirect.com/science/article/pii/S2090447923002332>
28. What Is the Genetic Algorithm [Електронний ресурс]. – Режим доступу: <https://www.mathworks.com/help/gads/what-is-the-genetic-algorithm.html>
29. What is Swarm Intelligence [Електронний ресурс]. – Режим доступу: <https://legalvision.com.au/what-is-swarm-intelligence/>
30. Evolutionary and Swarm-Based Feature Selection for Imbalanced Data Classification / F. Namous та ін. Algorithms for Intelligent Systems. Singapore, 2019. С. 231–250. URL: https://doi.org/10.1007/978-981-32-9990-0_11 (дата звернення: 05.11.2023).

31. On Test Case Prioritization Using Ant Colony Optimization Algorithm [Електронний ресурс]. – Режим доступу: <https://ieeexplore.ieee.org/document/8855691>
32. How Can Bio-Mapping the Foraging Excursions of Ants Translate Into a Prototype for Human Living on Mars [Електронний ресурс]. – Режим доступу: <https://simaud.org/2020/proceedings/60.pdf>
33. Path planning of lunar robot based on dynamic adaptive ant colony algorithm and obstacle avoidance [Електронний ресурс]. – Режим доступу: <https://journals.sagepub.com/doi/full/10.1177/1729881419898979>
34. Nature-Inspired Algorithms Applications / ред.: S. Balamurugan та ін. Wiley, 2021. С. 63–69. URL: <https://doi.org/10.1002/9781119681984> (дата звернення: 04.11.2023).
35. Rajendra S. Optimistic Perspective on Bio-Inspired Algorithms. Notion Press, 2020. С. 21–26. URL: https://books.google.com.ua/books?id=uT2CzwEACAAJ&newbks=0&hl=uk&redir_esc=y (дата звернення: 08.11.2023).
36. Mathias H. D., Wu A. S., Ruetten L. Heterogeneous Response Intensity Ranges and Response Probability Improve Goal Achievement in Multi-agent Systems. Lecture Notes in Computer Science. Cham, 2020. С. 148–160. URL: https://doi.org/10.1007/978-3-030-60376-2_12 (дата звернення: 06.12.2023).
37. Yang X.-S. Nature-Inspired Optimization Algorithms. Elsevier Science & Technology Books, 2020. 332 URL: https://books.google.com.ua/books?id=La_YDwAAQBAJ&newbks=0&hl=uk&redir_esc=y (дата звернення: 06.11.2023).
38. Davim J. P., Kumar K., Zindani D. Optimizing Engineering Problems Through Heuristic Techniques. Taylor & Francis Group, 2019. 138 p URL: https://books.google.com.ua/books?id=pF3BDwAAQBAJ&newbks=0&hl=uk&redir_esc=y (дата звернення: 06.11.2023).
39. Attribute Selection Method Based on Artificial Bee Colony Algorithm and Neighborhood Discrimination Matrix Optimization / Y. Ji та ін. Communications in

Computer and Information Science. Singapore, 2022. С. 71–87. URL: https://doi.org/10.1007/978-981-19-1256-6_6 (дата звернення: 08.11.2023).

40. Determination of the Shortest Route Towards the Tourist Destination Area Using the Ant Algorithm [Електронний ресурс]. – Режим доступу: <https://iopscience.iop.org/article/10.1088/1742-6596/1339/1/012038/pdf>

41. Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». - Хмельницький, 2023. - С. 239-241.

ДОДАТКИ

Додаток А

Фрагмент програмного коду реалізації методу для створення річкової переправи

Клас «RiverCrossing», реалізація мурашиного алгоритму в розробленій інформаційній системі:

```
@Getter
```

```
@Setter
```

```
@NoArgsConstructor
```

```
@AllArgsConstructor
```

```
public class RiverCrossing {  
    private List<RiverAnt> ants;  
    private double[][] pheromones;  
    private String[] shoreTypes;  
    private double[] riverLengths;  
    private double[] riverDepths;  
    private double[] riverWidths;  
    private double[] ferryCapacity;  
    public RiverCrossing(RiverCrossingDTO riverCrossingDTO) {  
        ants = new ArrayList<>();  
        for (int i = 0; i < riverCrossingDTO.getNumAnts(); i++) {  
            ants.add(new RiverAnt(i));  
        }  
        pheromones = new double[riverCrossingDTO.getNumAnts()][riverCrossingDTO.getNumAnts()];  
        for (int i = 0; i < riverCrossingDTO.getNumAnts(); i++) {  
            for (int j = 0; j < riverCrossingDTO.getNumAnts(); j++) {  
                pheromones[i][j] = 0.0;  
            }  
        }  
    }  
}
```

```

this.shoreTypes = riverCrossingDTO.getShoreTypes();

this.riverLengths = riverCrossingDTO.getRiverLengths();

this.riverDepths = riverCrossingDTO.getRiverDepths();

this.riverWidths = riverCrossingDTO.getRiverWidths();

this.ferryCapacity = riverCrossingDTO.getFerryCapacity();
}

public void performCrossing() {

    int numIterations = 1000;

    double evaporationRate = 0.1;

    for (int iteration = 0; iteration < numIterations; iteration++) {

        for (RiverAnt ant : ants) {

            int start = ant.getId();

            int destination = selectDestination(start);

            double deltaPheromone = 1.0 / calculateDistance(start, destination);

            pheromones[start][destination] += deltaPheromone;

            pheromones[destination][start] += deltaPheromone;

        }

        for (int i = 0; i < ants.size(); i++) {

            for (int j = 0; j < ants.size(); j++) {

                pheromones[i][j] *= (1 - evaporationRate);

            }

        }

    }

}

private int selectDestination(int start) {

    Random rand = new Random();

    double[] probabilities = new double[ants.size()];

    double totalProbability = 0.0;

```

```

for (int i = 0; i < ants.size(); i++) {
    if (i == start) {
        probabilities[i] = 0.0;
    } else {
        double attractiveness = pheromones[start][i] / calculateDistance(start, i);
        attractiveness *= calculateShoreTypeFactor(start, i);
        attractiveness *= calculateDepthFactor(start, i);
        attractiveness *= calculateWidthFactor(start, i);
        attractiveness *= calculateCapacityFactor(start, i);
        probabilities[i] = attractiveness;
        totalProbability += probabilities[i];
    }
}

double randomValue = rand.nextDouble() * totalProbability;
double cumulativeProbability = 0.0;
for (int i = 0; i < ants.size(); i++) {
    cumulativeProbability += probabilities[i];
    if (cumulativeProbability >= randomValue) {
        return i;
    }
}

return rand.nextInt(ants.size());
}

private double calculateDistance(int start, int destination) {
    return Math.abs(start - destination) * (riverLengths[start] + riverLengths[destination]);
}

private double calculateShoreTypeFactor(int start, int destination) {

```

```
String startShoreType = shoreTypes[start];

String destinationShoreType = shoreTypes[destination];

double sandyShoreFactor = 1.2;

double rockyShoreFactor = 0.8;

double gravelShoreFactor = 1.0;

double earthShoreFactor = 0.6;

double clayShoreFactor = 0.8;

switch (startShoreType) {

    case "пісчаний":

        switch (destinationShoreType) {

            case "пісчаний":

                return sandyShoreFactor;

            case "гравійний":

                return sandyShoreFactor * 1.1;

            case "земляний":

                return sandyShoreFactor * 0.8;

            case "глиняний":

                return sandyShoreFactor * 0.9;

        }

    case "піщано-кам'янистий":

        switch (destinationShoreType) {

            case "пісчаний", "гравійний":

                return rockyShoreFactor;

            case "земляний":

                return rockyShoreFactor * 0.8;

            case "глиняний":

                return rockyShoreFactor * 0.9;

        }

    case "гравійний":
```

```
switch (destinationShoreType) {  
    case "пісчаний", "гравійний":  
        return gravelShoreFactor;  
    case "земляний":  
        return gravelShoreFactor * 0.6;  
    case "глиняний":  
        return gravelShoreFactor * 0.7;  
}  
case "земляний":  
    switch (destinationShoreType) {  
        case "пісчаний":  
            return earthShoreFactor * 1.2;  
        case "гравійний", "земляний":  
            return earthShoreFactor;  
        case "глиняний":  
            return earthShoreFactor * 0.8;  
    }  
case "глиняний":  
    switch (destinationShoreType) {  
        case "пісчаний":  
            return clayShoreFactor * 1.2;  
        case "гравійний":  
            return clayShoreFactor * 0.7;  
        case "земляний":  
            return clayShoreFactor * 0.8;  
        case "глиняний":  
            return clayShoreFactor;  
    }  
}
```

```
    return 1.0;
}

private double calculateDepthFactor(int start, int destination) {
    double startDepth = riverDepths[start];
    double factor = getFactor(startDepth, ferryCapacity[destination]);
    return Math.min(Math.max(factor, 0.6), 1.2);
}

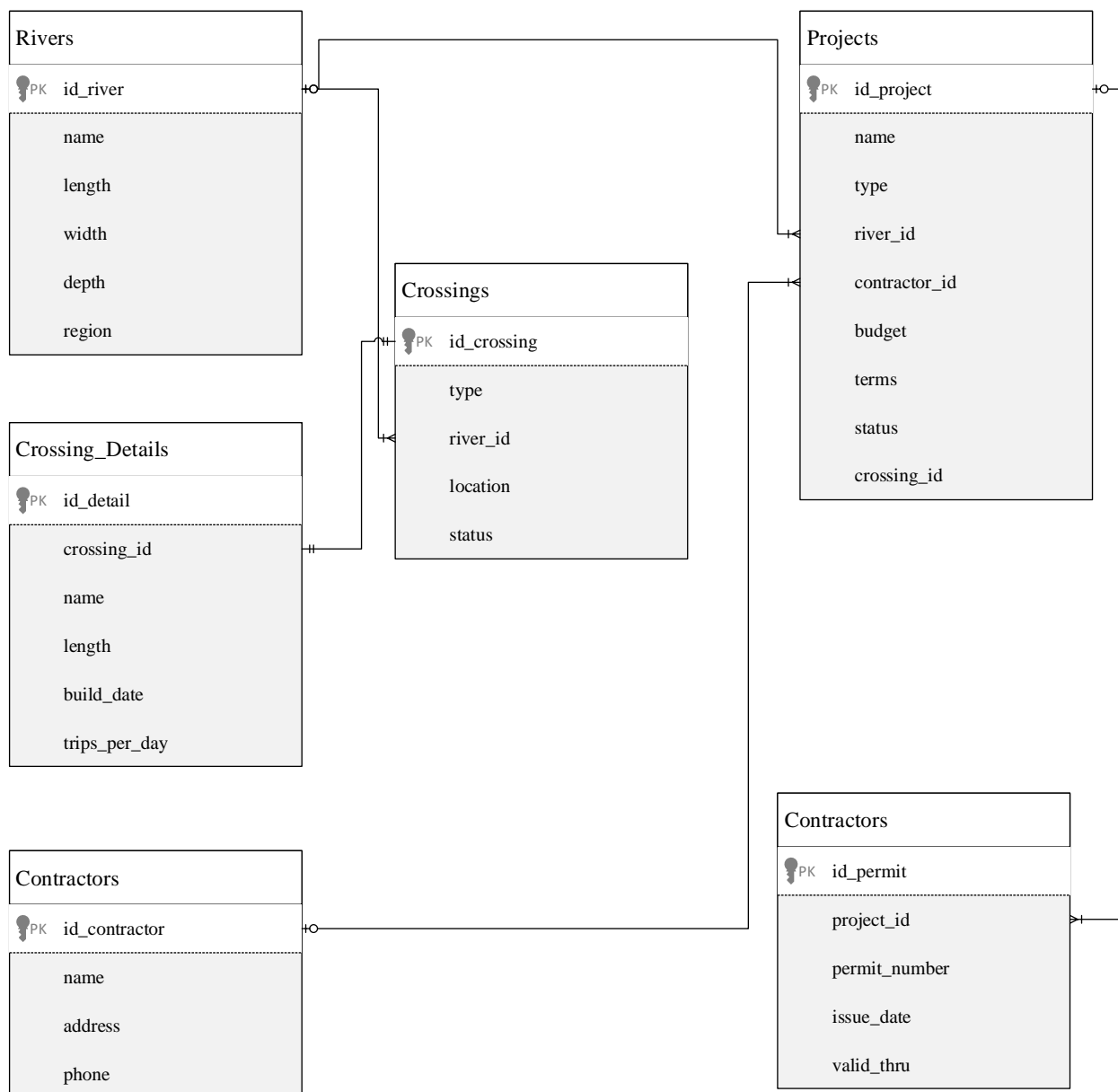
private double calculateWidthFactor(int start, int destination) {
    double startWidth = riverWidths[start];
    double factor = getFactor(startWidth, ferryCapacity[destination]);
    return Math.min(Math.max(factor, 0.6), 1.2);
}

private double calculateCapacityFactor(int start, int destination) {
    double startCapacity = ferryCapacity[start];
    double factor = getFactor(startCapacity, ferryCapacity[destination]);
    return Math.min(Math.max(factor, 0.6), 1.2);
}

private double getFactor(double startCapacity, double ferryCapacity) {
    double highCapacityFactor = 1.2;
    double lowCapacityFactor = 0.8;
    double startLogCapacity = Math.log10(startCapacity);
    double destinationLogCapacity = Math.log10(ferryCapacity);
    return highCapacityFactor + (destinationLogCapacity - startLogCapacity) * (lowCapacityFactor -
highCapacityFactor);
}
```

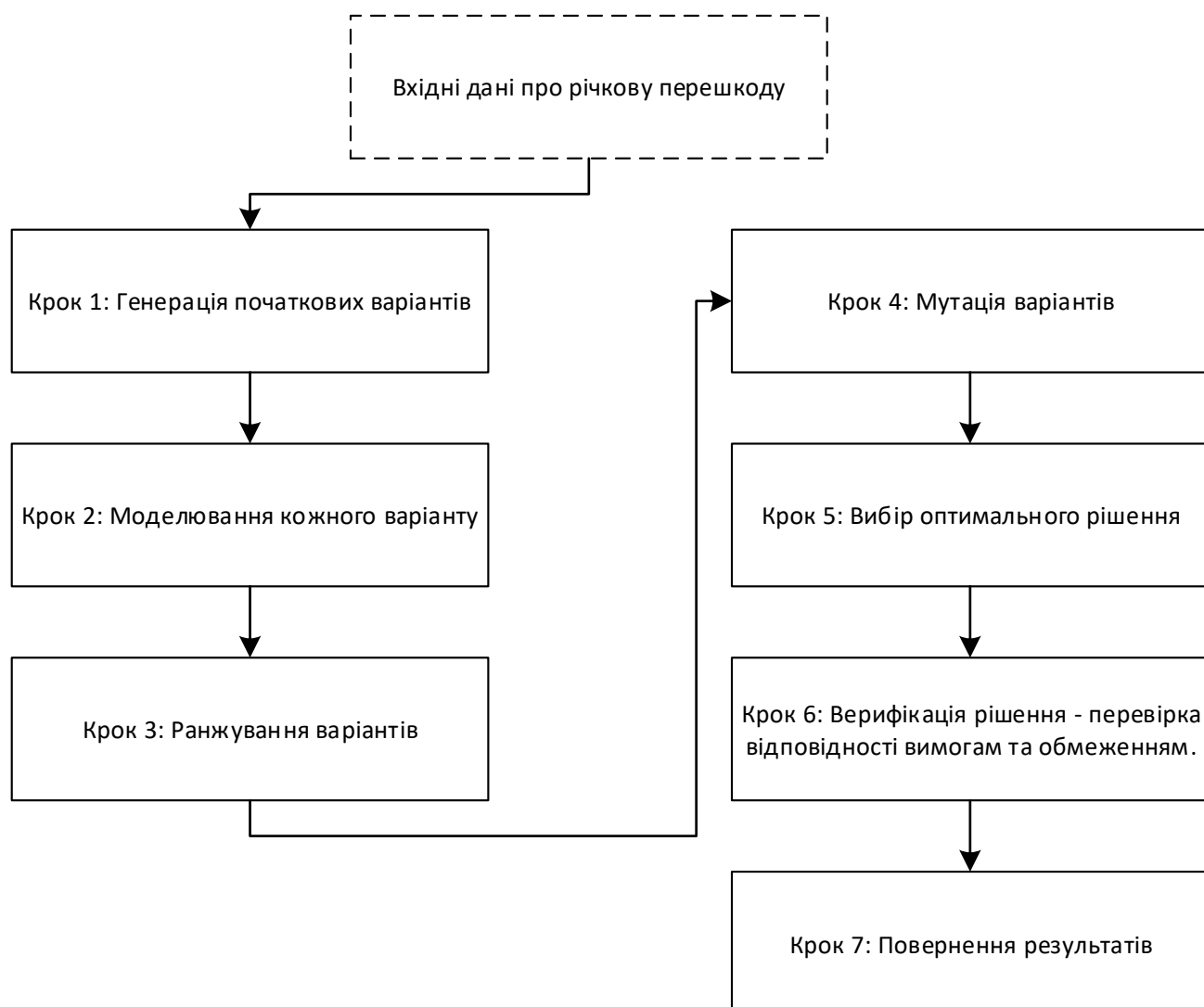
Додаток Б

Інфологічна модель бази даних



Додаток В

Схема послідовності виконання процесів однієї ітерації



Додаток Г

Таблиця інструменти ПЗ LARS Bridge

Інструмент	Опис
Вибір	Дозволяє вибрати об'єкти в моделі. Для цього можна використовувати різні методи, наприклад, вибір за допомогою миші, клавіатури або комбінації клавіш.
Робота з поверхнями	Дозволяє створювати, редагувати та видаляти поверхні. Для створення поверхні можна використовувати різні методи, наприклад, вводити координати точок, використовувати параметризовані інструменти або створювати поверхню за допомогою кривих. Для редагування поверхні можна використовувати різні інструменти, наприклад, змінювати розмір, форму та положення поверхні. Для видалення поверхні можна використовувати інструмент "Видалити".
Робота з лініями	Дозволяє створювати, редагувати та видаляти лінії. Для створення лінії можна використовувати різні методи, наприклад, вводити координати точок, використовувати параметризовані інструменти або створювати лінію за допомогою кривих. Для редагування лінії можна використовувати різні інструменти, наприклад, змінювати довжину, форму та положення лінії. Для видалення лінії можна використовувати інструмент "Видалити".
Робота з точками	Дозволяє створювати, редагувати та видаляти точки. Для створення точки можна використовувати різні методи, наприклад,

	<p>вводити координати точки або використовувати параметризовані інструменти. Для редагування точки можна використовувати різні інструменти, наприклад, змінювати координати точки. Для видалення точки можна використовувати інструмент "Видалити".</p>
--	---

Додаток Г

Таблиця параметри оцінки маршруту переправи для методу

Параметр	Опис
Ширина русла річки в місці переправи	Чим ширше русло, тим складніша та дорожча переправа.
Глибина річки	Впливає на тип фундаментів, довжину мосту або глибину закладення тунелю.
Швидкість течії	Визначає навантаження на конструкцію та складність будівництва опор.
Характер дна річки	М'яке дно ускладнює спорудження мостів.
Інтенсивність транспортного потоку	Впливає на пропускну спроможність переправи.
Відстань до інфраструктури	Чим далі, тим вище вартість під'їзних шляхів.
Кліматичні умови	Враховуються при виборі матеріалів та конструкцій.
Екологічні обмеження	Природоохоронні території можуть обмежувати тип переправи.
Пропускна здатність	Максимальна вага транспорту, яка може одночасно пересуватися. Максимальна ширина транспорту. Максимальна кількість транспорту, яка одночасно може пересуватися переправою.
Перешкоди	Перешкоди які потрібно усунути для реалізації переправи.

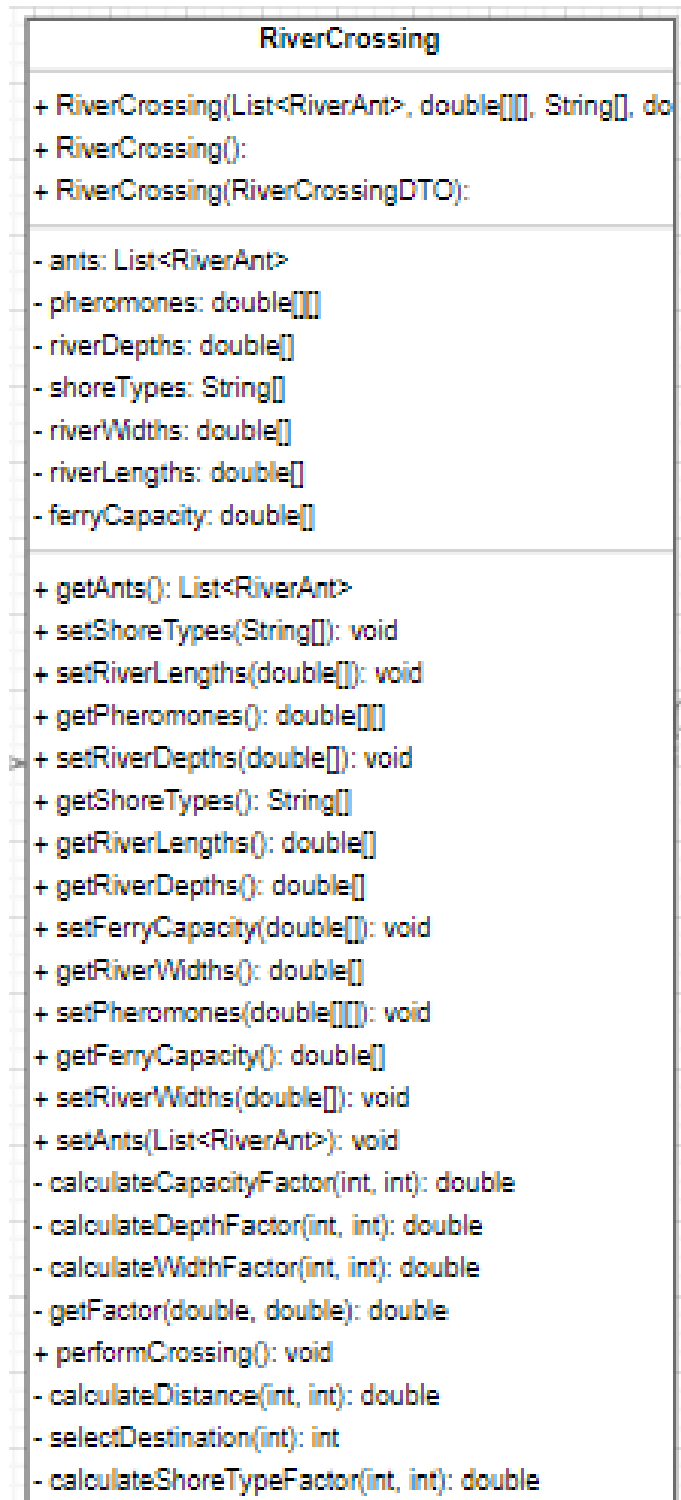
Додаток Д

Таблиця характеристика ініціалізації маршруту методом мурашиного алгоритму

Характеристика	Опис
Довжина	Відстань від початку переправи з першого берега до другого берега. Вимірюється в метрах або кілометрах. Довжина переправи є одним з найважливіших факторів, що впливають на її вартість та складність.
Складність	Перешкоди, течія, та інші фактори які ускладнюють реалізацію переправи. До перешкод можуть відноситись: глибина річки, швидкість течії, наявність порогів, островів, а також берегова лінія. Складність переправи впливає на вибір типу переправи та її вартість.
Затрата ресурсів	Кількість матеріалів, робочої сили яка потрібна щоб створити річкову переправу. До матеріалів можуть відноситись: деревина, метал, бетон, а також будівельні інвентар та інструменти. Робоча сила необхідна для виконання будівельних робіт. Затрати ресурсів впливають на вартість переправи.
Час орієнтовної переправи	Кількість часу яку потрібно потратити для реалізації переправи. Час переправи залежить від довжини переправи, складності, а також наявних ресурсів та робочої сили.

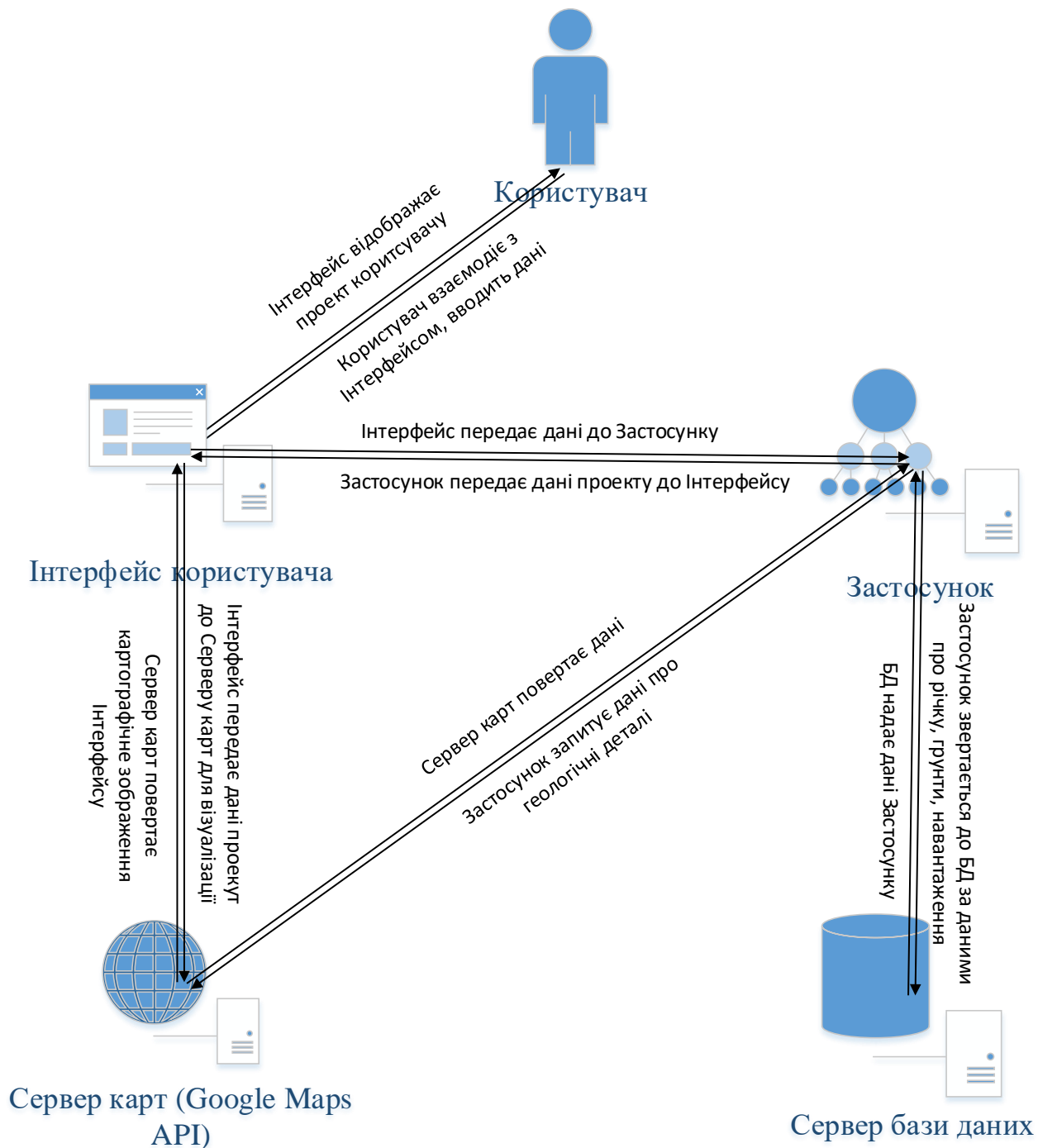
Додаток Е

Діаграма класу River Crossing



Додаток Є

UML діаграми потоків даних (information flow diagram) інформаційної системи проектування річкової переправи



Додаток Ж

Світлини наукових публікацій, виконаних при роботі над кваліфікаційною роботою магістра

Перелік наукових публікацій:

1. Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». - Хмельницький, 2023. - С. 239-241.

Міністерство освіти і науки України
Хмельницький національний університет



ЗБІРНИК НАУКОВИХ ПРАЦЬ
за матеріалами XV Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2023»

17-18 листопада 2023

Хмельницький 2023

Овчарук О.М., Мазурець О.В. Прогнозування значень параметрів за їх часовими рядами рекурентною темпоральною нейронною мережею	227
Олійник П.О. Удосконалений метод роботи з метриками покриття коду для забезпечення ефективного оцінювання результатів тестування програмного забезпечення	233
Онїкієнко С.С., Глухов В.Ю., Манзюк Е.А. Детектування об'єктів на зображеннях з невеликою роздільною здатністю	236
Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом	239
Павлова О.О., Рудик І.В. Пост-обробка сигналів тривоги систем відеоспостереження за допомогою нейромережі YOLOv8	242
Побережна А.Ю. Кіберфізична система моніторингу стану рослин в режимі реального часу	245
Присяжнюк О.О. Дослідження та проектування комп'ютерних систем штучного інтелекту	250
Рижий Я.О., Мельник М.М., Чешиун О.В., Орленко В.С. Класифікація атрибутів особи і формування цифрового підпису на їх основі	252
Ровінчук Д.Ю. Метод та програмні засоби виявлення метаморфних комп'ютерних вірусів.....	257
Родін О.О., Яшина О.М. Метод спектральних характеристик звукового сигналу для визначення рівня психологічного стану людини за допомогою глибинного навчання	260
Савчук А.В. Розробка бібліотеки для побудови та обчислень електричних кіл.....	264
Сверба А.А. Удосконалення методу роботи з метрикою середнього часу між відмовами для забезпечення ефективного оцінювання результатів тестування програмного забезпечення	266
Свистун С.О., Мельниченко О.В., Скрипник Т.К. Проектування робочої місії безпілотних літальних апаратів в тривимірному просторі.....	269

УДК 004.4

Охота В.В., Міхалевський В.Ц., Скрипник Т.К.

*Хмельницький національний університет***МЕТОД ЗДІЙСНЕННЯ РІЧКОВОЇ ПЕРЕПРАВИ ТРАНСПОРТНИХ ЗАСОБІВ
ЗА МУРАШИНИМ АЛГОРИТМОМ**

Розглянуто метод здійснення річкової переправи транспортних засобів за основи мурашиного алгоритму. Суть методу полягає в оптимізації процесу перетину великих водних перешкод. Шляхом використання цього алгоритму, ресурси та транспортні засоби розподіляються оптимально для максимально ефективного подолання річкових бар'єрів. Описано також інтерфейс, що надає можливість користувачам відслідковувати поточний стан процесу річкової переправи.

The method of carrying out the river crossing of vehicles based on the ant algorithm is considered. The essence of the method is to optimize the process of crossing large water obstacles. By using this algorithm, resources and vehicles are allocated optimally to overcome river barriers as efficiently as possible. An interface that enables users to monitor the current state of the river crossing process is also described.

Метод здійснення річкової переправи транспортних засобів на основі мурашиного алгоритму передбачає оптимізацію перетину великих водних перешкод.

Мурашині алгоритми базуються на імітації поведінки мурах при пошуку їжі та побудові маршрутів до місць її знаходження. Це включає в себе використання феромонів - хімічних речовин, які мурахи залишають за собою. Феромони служать якірними точками для інших мурах і вказують, який маршрут є більш оптимальним.

Вивчення поведінки мурашиних колоній важливе для комп'ютерних наук, оскільки це надає уявлення про роз'єднану організацію, що дуже корисно для вирішення складних задач оптимізації. Мурашині алгоритми стали джерелом натхнення для розробки оптимізаційних методів у сферах, де необхідно знаходити найкращі рішення в умовах обмежень та невизначеності.

Використовуючи такий алгоритм, ресурси та транспортні засоби розподіляються оптимально для максимально ефективного подолання річкових бар'єрів. Також описується інтерфейс, що дозволяє користувачам відстежувати поточний стан процесу річкової переправи.

Виконання дій в кілька етапів здійснюється за допомогою ітерацій одночасно кількома мурахами. Під час однієї ітерації кожна мураха проходить маршрут через всі вершини незалежно від інших. Для мурахи, яка перебуває у вершині *i*, перехід до вершини *j* залежить від трьох компонент: пам'яті мурахи,

видимості та віртуального сліду феромону. Пам'ять мурахи - це список вершин, які мураха вже відвідала і не повинна повторно відвідувати. Наявність цього списку гарантує, що мураха не відвідає ту саму вершину двічі. З рухом мурахи цей список зростає і обнуляється на початку кожної ітерації. $J_{i,k}$ - це список вершин, які мураха k , яка перебуває у вершині i , ще не відвідала. Ясно, що $J_{i,k}$ є доповненням до пам'яті мурахи.

Видимість – це величина обернена до відстані: $\eta_{ij} = 1/D_{ij}$, де D_{ij} — відстань між вершинами i та j . Видимість базується тільки на локальній інформації і являє собою «евристичну бажаність» вибору вершини j , під час перебування у вершині i . Чим ближче вершини i та j , тим більше бажання відвідати їх.

Віртуальний слід феромону на шляху між вершинами i та j є прагматичним бажанням мурахи відвідати місто j з міста i , підтвержене досвідом. На відміну від видимості, ця величина змінюється після кожної ітерації алгоритму. Кількість віртуального феромону на шляху з міста i в місто j на ітерації t позначається як $\tau_{ij}(t)$.

Основні сутності предметної області для задачі, що розглядається, наведені в Таблиці 1.

Таблиця 1 – Головні сутності алгоритму

Сутність	Опис
Річка	Природний бар'єр, через який потрібно переправити.
Береги	Два береги річки, між якими знаходяться транспортні засоби.
Транспорт	Засоби, які потрібно перевести через річку.
Мурахи	Віртуальні мурахи, які використовуються в алгоритмі для пошуку шляху.
Переправа	Процес переправи транспортних засобів через річку, включаючи вибір шляху та використання мурашиного алгоритму.

Запропонований метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом розробляє оптимальний підхід до подолання водних перешкод в умовах невизначеності. Цей метод спрямований на ефективне використання доступних ресурсів та забезпечення надійної переправи.

Система програмного забезпечення для задачі річкової переправи транспортних засобів включає наступні компоненти:

1) Компонент для управління даними: Цей компонент відповідає за зберігання та обробку даних, пов'язаних з річкою, транспортними засобами та іншими факторами. Ці дані можуть включати в себе інформацію про річку (ширина, глибина, швидкість течії), транспортні засоби (габарити, швидкість, вантажопідйомність), погодні умови (рівень води, вітер) та дані про переправи (час відправлення, час прибуття).

2) Компонент для планування маршрутів: Цей компонент відповідає за планування маршрутів переправи транспортних засобів. При плануванні маршрутів необхідно враховувати обмеження транспортних засобів (габарити, швидкість,

вантажопідйомність), погодні умови (рівень води, вітер) та динаміку навколишнього середовища, таку як зміна рівня води.

3) Компонент для відстеження переправ: Цей компонент відповідає за відстеження переправ транспортних засобів і забезпечення їх безпеки. При відстеженні переправ необхідно враховувати положення транспортних засобів, їх швидкість та стан навколишнього середовища.

4) Компонент для взаємодії з користувачами: Цей компонент відповідає за взаємодію з користувачами, включаючи операторів переправ і клієнтів. При взаємодії з користувачами важливо забезпечити зручність використання, безпеку та надійність.

Ці компоненти можуть бути реалізовані на різних мовах програмування та технологіях. Для реалізації системи програмного забезпечення для задачі річкової переправи транспортних засобів можна використовувати такі технології: **Java** (мова програмування, яка дозволяє розробляти різноманітні типи програмного забезпечення, включаючи веб-додатки та десктопні додатки); **Spring** (фреймворк для Java, який спрощує розробку різних типів додатків); **Spring Data** (модуль фреймворка Spring для роботи з даними, що допомагає спростити доступ до інформації); **Lombok** (бібліотека для спрощення кодування на Java, що полегшує розробку); **MySQL** (система керування базами даних для зберігання даних про річку, транспортні засоби та переправи); **Spring Web** (модуль фреймворка Spring для розробки веб-додатків); **Docker** (технологія, що дозволяє створювати контейнери для програмного забезпечення та спрощує розгортання додатків).

Майбутні дослідження будуть спрямовані на тестування та оптимізацію методу, а також на його інтеграцію з реальними системами для задачі річкової переправи.

Перелік посилань

1. Rukundo, O., Cao, H. Advances on image interpolation based on ant colony algorithm URL: <https://springerplus.springeropen.com/articles/10.1186/s40064-016-2040-9>.
2. Оптимізація колонії мурах. URL: <https://www.sciencedirect.com/topics/engineering/ant-colony-optimization>
3. Вступ до оптимізації колонії Ant (ACO) URL: <https://towardsdatascience.com/the-inspiration-of-an-ant-colony-optimization-f377568ea03f>
4. Алгоритми мурах для дискретної оптимізації URL: https://people.idisia.ch/~luca/ij_23-alife99.pdf

Додаток 3

Презентаційний матеріал



КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

ТЕМА

МЕТОД ЗДІЙСНЕННЯ РІЧКОВОЇ ПЕРЕПРАВИ
ТРАНСПОРТНИХ ЗАСОБІВ ЗА МУРАШИНИМ
АЛГОРИТМОМ

Виконав:
студент 2 курсу, група КНм-22-1
Охота Вадим Володимирович

Керівник:
к.ф.-м.н., доцент кафедри КН
Міхалевський Віталій Цезарійович

МЕТА ТА ЗАДАЧІ, ОБ'ЄКТ, ПРЕДМЕТ

Метою кваліфікаційної роботи магістра є розробка методу здійснення річкової переправи транспортних засобів за мурашиним алгоритмом.

Для досягнення поставленої мети визначено наступні **задачі**:

- провести аналіз предметної області та відомих підходів до організації річкової переправи;
- розробити метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом;
- підготувати набір даних для застосування у мурашиному алгоритмі;
- застосувати мурашиний алгоритм для оптимізованого здійснення річкової переправи транспортних засобів;
- провести функціональне та прикладне дослідження ефективності розроблених засобів, у тому числі для тестового аналізу річкової переправи транспортних засобів.

Об'єкт дослідження – процес річкової переправи транспортних засобів.

Предмет дослідження – моделі, методи та засоби автоматизації процесу визначення оптимальної річкової переправи транспортних засобів.

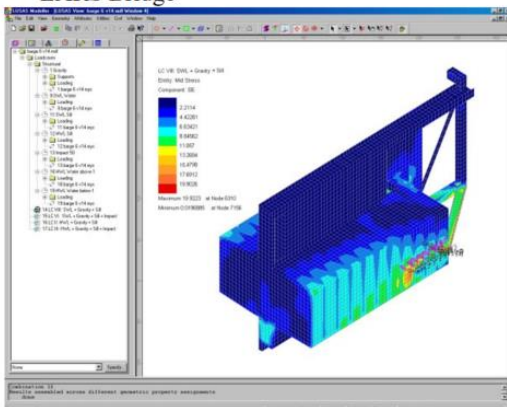
АКТУАЛЬНІСТЬ РОБОТИ

В сучасних умовах війни важливо мати технічні та інформаційні рішення для організації переправ через річки. Правильний вибір типу переправи дозволить забезпечити швидке, безпечне та ефективне переміщення людей і вантажів. Особлива увага приділяється спеціальним переправам для обмеженого кола осіб або спеціальних транспортних засобів. Вдосконалення технологій дозволяє будувати складні інженерні споруди. При організації переправ варто комплексно підходити до вирішення завдання. Розробка методів для оптимізації роботи системи переправ є актуальним.

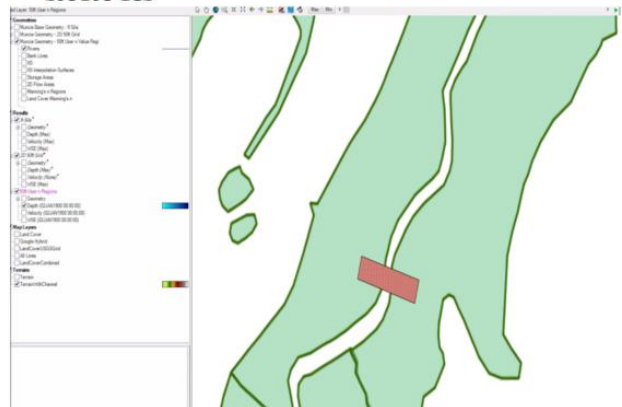
АНАЛІЗ НАЯВНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Наявне програмне забезпечення:

- LARS Bridge

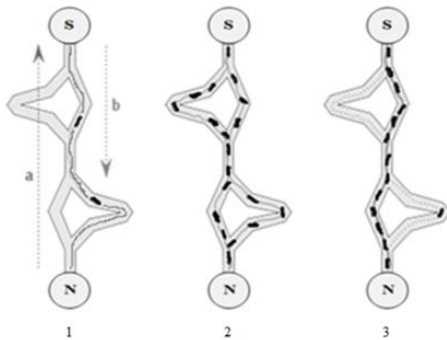


- SAGA GIS



ОПИС АЛГОРИТМУ

Пошук найкоротшого рішення виконується за рахунок феромонів та ітерацій процесу.

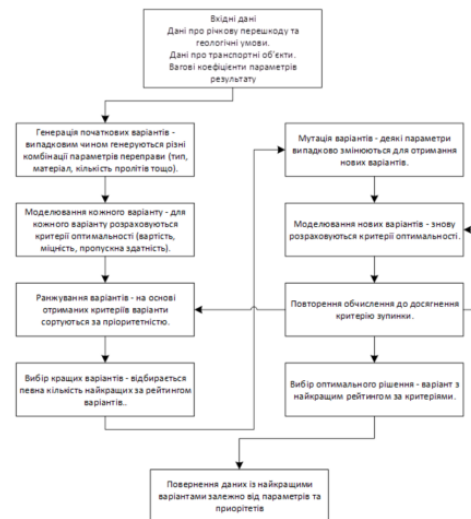


Алгоритм можливо використовувати як для первинного аналізу для створення річкової переправи, так і для детального постійного аналізу в реальному часі.



ІНФОРМАЦІЙНА МОДЕЛЬ МЕТОДУ НА ОСНОВІ МУРАШИНОГО АЛГОРИТМУ

- Модель генерує різноманітні варіанти переправ з різними конструкціями та матеріалами.
- Кожен варіант оцінюється
- Модель пристосовується до конкретних умов
- Ітераційне приближення до шуканих результатів



ЗАСОБИ РОЗРОБКИ МЕТОДУ

Для розробки методу та системи вибрано сучасні технології, що дозволять швидко реалізувати функціонал і забезпечити необхідну продуктивність та масштабованість. Зокрема, це мова Java, фреймворк Spring та API від Google для роботи з мапами.



РЕЗУЛЬТАТИ РОЗРОБКИ

Розроблений метод дозволяє здійснювати:

- Розрахунок часу створення переправи
- Визначати оптимальний тип переправи (міст, пором, канатна дорога тощо).
- Можливість використання вже створених переправ у реальному часі.

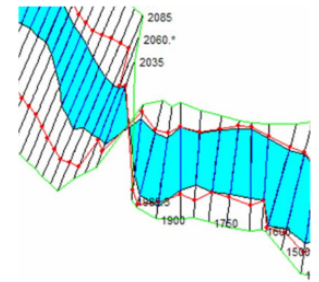
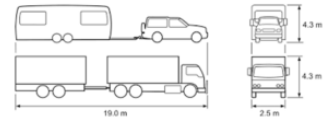
№1: Привабливість: 10;
Час створення: 1 день
Час перетину: 60 сек
Тип: Понтонна переправа
№2: Привабливість: 5;
Час створення: 1 день
Час перетину: 683 сек
Тип: Поромна переправа
№3: Привабливість: 7;
Час створення: 48 днів;
Час Перетину: 13 сек
Тип Міст



ВХІДНІ ПАРАМЕТРИ МЕТОДУ

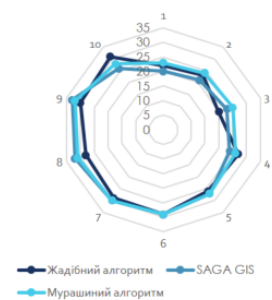
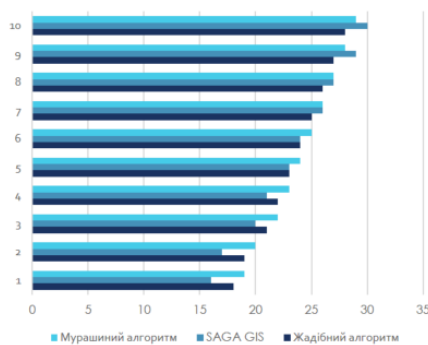
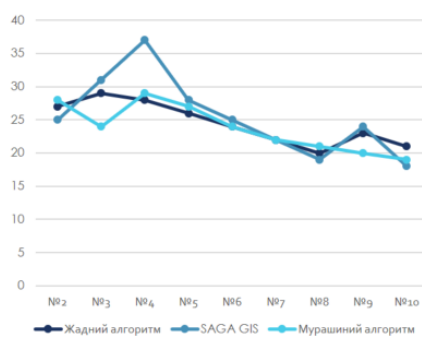
Вхідні параметри методу включають:

- Геометричні дані транспортних засобів (довжина, ширина, висота, колісна база, колія, кліренс)
- Вага транспортних засобів (повна вага, вага на ось)
- Центр ваги транспортних засобів (висота розташування)
- Можливість плавання (форсування) транспортних засобів (наявність/відсутність герметизації, плавучості)
- Маневреність транспортних засобів (радіус повороту)
- Габарити з вантажем (при транспортуванні вантажу)
- Кількість транспортних засобів (в тому числі з причепами)
- Параметри річки: ширина, глибина, швидкість і напрямок течії, характер дна, характеристики берегів
- Характеристики мостів та інших перешкод
- Відстань до найближчої інфраструктури (доріг, посадкових майданчиків)
- Наявність місць для стоянки та розвороту).
- Параметри річкової переправи: тип (міст, поромна, канатна переправа тощо), габарити, вантажопідйомність, пропускна спроможність
- Гідрологічні умови (рівень води, паводкова ситуація)
- Метеорологічні умови (дощ, сніг, температура)



ФУНКЦІОНАЛЬНЕ ДОСЛІДЖЕННЯ МЕТОДУ

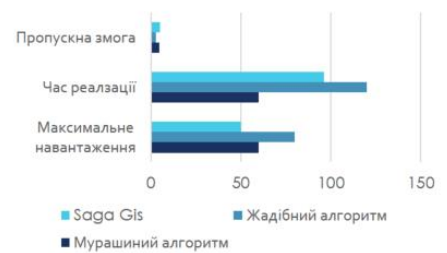
Для дослідження використано умову переправлення 10 легкових автомобілів вагою 2.5 - 3.5 тон. та набір різних параметрів, які впливають на роботу алгоритму.



ФУНКЦІОНАЛЬНЕ ДОСЛІДЖЕННЯ МЕТОДУ

Для наступного дослідження визначено завдання: побудувати тимчасову переправу для техніки комбінованого типу: 5 танків, 10 бронетранспортерів, 18 вантажівок, 25 легкових автомобілів. Річка Дніпро в Херсонській області — широка та глибока, глибина до 7-10 м, ширина в цьому районі сягає понад 800 м, швидкість течії 1,5-2 м/с, дно піщане з глибокими ямами. Береги переважно високі, скелясті. Головні параметри для кращого потрібного результату: час реалізації завдання, пропускна змога.

Параметр	Мурашиний алгоритм	Жадібний алгоритм	SAGA GIS
Тип	Поромна переправа	Переправа повітряними поромами	Комбінований варіант з поромів і човнів
Максимальне навантаження	60 тон	80 тон	50 тон
Час реалізації	60 годин	120 годин	96 годин
Надійність	Достатня	Висока	Середня
Пропускна спроможність	45 транспортних засобів/год	40 транспортних засобів/год	38 транспортних засобів/год



РЕЗУЛЬТАТИ ВПРОВАДЖЕННЯ

Основні наукові й практичні результати кваліфікаційної роботи магістра доповідались на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023» (18-19 листопада 2023 року) у доповіді на тему «Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом»; за темою роботи автором виконано наукову публікацію: Охота В.В., Міхалевський В.Ц., Скрипник Т.К. Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». - Хмельницький, 2023. - С. 239-241.

Ім'я користувача:
Кафедра КН

ID перевірки:
1015983896

Дата перевірки:
08.12.2023 12:57:28 EET

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
08.12.2023 13:13:48 EET

ID користувача:
100005671

Назва документа: КНм-22-1 Охота

Кількість сторінок: 87 Кількість слів: 14980 Кількість символів: 115994 Розмір файлу: 2.01 MB ID файлу: 1015664555

9.16% Схожість

Найбільша схожість: 2.37% з Інтернет-джерелом (<http://elar.khmnu.edu.ua/jspui/bitstream/123456789/12812/1/%d0%a1>).

8.65% Джерела з Інтернету 822 Сторінка 89

5.23% Джерела з Бібліотеки 143 Сторінка 96

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 1

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилки в документах: 7%**

ID: 122176 Назва: КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА Тема: Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом Додано в БД: 2023-12-08 Автора: В.В. Охота Керівники: В.Ц. Міхалевський Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	89762	1407	2560 (3%)	36 (3%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНИХ НАУК
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА ДО ЗАХИСТУ ЗА
РЕЗУЛЬТАТАМИ АНАЛІЗУ ЗВІТУ ПОДІБНОСТІ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

Автор: Охота В.В., група КНм-22-1

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: комп'ютерні науки

Науковий керівник: к.ф-м.н., доц. кафедри КН Міхалевський В.Ц.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) за програмою Anti-Plagiarism виявлені 1% запозичень, що підтверджує авторство дослідження.

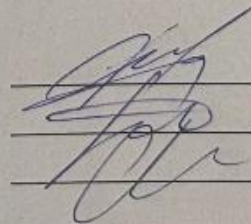
2) За програмою UNICHECK виявлені 9,16% запозичень є фрагментарними – містять поширені конструкції, загальновідомі терміни, скорочення та визначення.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 1% і 9,16% відповідно, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КН



В. Ц. Міхалевський

Р. О. Багрій

О. В. Бармак



ВІДГУК ОПОНЕНТА

на кваліфікаційну роботу магістра

гр. КНМ-22-1 Охоти Вадима Володимировича за темою: Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

1. Актуальність обраної теми

В сучасних умовах динамічної зміни завдань транспортної логістики, особливо у період війни, необхідно мати технічні та інформаційні рішення для організації переправ через річки. Правильний вибір типу переправи з урахуванням усіх вимог дозволить забезпечити швидке, безпечне та економічно ефективно переміщення людей і вантажів. Сучасні технології автоматизації сприяють підвищенню якості та надійності функціонування переправ.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Обрана тема розробки методу організації річкової переправи технічних засобів на основі мурашиного алгоритму та створення відповідного програмного забезпечення повною мірою відповідає предметній області спеціальності 122 "Комп'ютерні науки", оскільки охоплює питання моделювання складних систем, проектування та реалізації алгоритмів і програм, проведення обчислювальних експериментів. Розроблений в роботі метод та програмні засоби дозволяють автоматизувати та оптимізувати процес організації річкових переправ.

3. Повнота розкриття мети та завдань дослідження

В роботі автор повністю розкриває мету дослідження та поставленні в межах теми завдання. Результати дослідження ясно визначають основну мету. Розроблено метод для автоматизації вибору оптимального маршруту річкової переправи технічних засобів. Даний метод дозволяє отримувати різні варіанти рішень з урахуванням наборів параметрів та формувати оптимальний маршрут.

4. Наявність наукової новизни

В кваліфікаційній роботі представлена наукова новизна, що відповідає спеціальності 122 «Комп'ютерні науки» в межах обраної області дослідження. Продемонстровано й обґрунтовано результати, які мають наукове та інноваційне значення. Результати дослідження оприлюднені на науково-практичній конференції «АПКН-2023» та у Збірнику наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023».

5. Зміст кожного розділу роботи

Робота складається з 4 розділів. У першому розділі проаналізовано предметну область, розглянуто відомі підходи до вирішення цієї задачі. У другому розділі сформовано інформаційну модель організації переправи, розроблено метод здійснення переправи технічних засобів на основі мурашиного алгоритму. Третій розділ присвячено реалізації програмного забезпечення для організації переправи за розробленим методом. У четвертому розділі наведено результати дослідження ефективності запропонованого методу організації річкових переправ та проведено порівняльний аналіз з існуючими підходами.

6. Ступінь розкриття теми роботи

Тема кваліфікаційної роботи повною мірою розкрита та обгрунтована. Виконаний аналіз актуальності та попередніх досліджень у вибраній сфері, сформульовані та виконані поставлені завдання, а також проведений аналіз результатів практичного використання запропонованих методів та засобів.

7. Якість оформлення кваліфікаційної роботи

Оформлення роботи відповідає необхідним нормам та вимогам, які ставляться до оформлення кваліфікаційних робіт.

8. Недоліки кваліфікаційної роботи

Явних недоліків в роботі не виявлено. Можна було б провести *узагальнення роботи* шляхом розширення діапазону технічних засобів для річкової переправи та *можливості масштабування і перенесення роботи методу на інші види переправ.*

9. Загальний висновок (допускається чи не допускається до захисту), якої оцінки заслуговує кваліфікаційна робота

Враховуючи достатній рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка добре.

Опонент _____ д.фіз.-мат.н., проф. Бедратюк Л.П.



ВІДГУК НАУКОВОГО КЕРІВНИКА

на кваліфікаційну роботу магістра

гр. КНМ-22-1 *Охоти Вадима Володимировича* за темою: Метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом

1. Актуальність теми

Актуальність теми обґрунтована в достатній мірі: на сьогоднішній день існує ряд рішень, що допомагають вирішувати задачу здійснення річкової переправи, проте вони мають ряд недоліків. Враховуючи те, що апаратно-технічне забезпечення постійно вдосконалюється і, крім того, розвивається парк програмного забезпечення, то розробка нових методів і засобів річкової переправи на поточному етапі є актуальною.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

В кваліфікаційній роботі магістра було розроблено метод здійснення річкової переправи транспортних засобів за мурашиним алгоритмом. Детально досліджені процеси проведення переправи, методи отримання рішень щодо вибору оптимального маршруту та проведено аналіз ряду програмних продуктів для здійснення переправи.

Тема кваліфікаційної роботи магістра відповідає предметній області спеціальності 122 Комп'ютерні науки та вимогам до кваліфікаційної роботи магістра: 1) Виконавцем сплановано і реалізовано метод здійснення річкової переправи та програмного забезпечення, проведено тестування та порівняльний аналіз. 2) Виконавець обрав інформаційне середовище розробки та дослідження, що дозволило знайти правильне і ефективне рішення, а запропонований метод на основі мурашиного алгоритму дозволяє обрати найоптимальніший варіант здійснення річкової переправи транспортних засобів.

Проаналізовано, оцінено та порівняно різні технології процесів для встановлення пріоритетів у відповідності з критеріями продуктивності та якості, що визначені завданням. Проведено функціональне та прикладне дослідження ефективності розроблених засобів, у тому числі для тестового аналізу випадків здійснення річкової переправи.

3. Професійні та особистісні якості магістранта

Магістрант володіє в достатній мірі професійними якостями дослідника: 1) Має здатність збирати, формалізувати, систематизувати і аналізувати потреби та вимоги до методу, що розробляється. 2) Має здатність формалізувати предметну область проекту у вигляді відповідної інформаційної моделі.

Серед особистісних якостей магістранта слід виділити відповідальність, цілеспрямованість, здатність навчатися, нестандартність мислення.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Студент більшу частину роботи виконав самостійно. Особисто магістрантом досліджено предметну область, проведено порівняльний аналіз переваг та недоліків існуючих методів для здійснення переправи; підібрано алгоритм для опрацювання даних; розроблено нову інформаційну систему автоматизованого визначення оптимального варіанту здійснення річкової переправи, проведено функціональне та прикладне дослідження ефективності розробленого методу, у тому числі для тестового аналізу випадків здійснення річкової переправи.

5. Наукова новизна та оригінальність запропонованих підходів

У виконаній роботі наукова новизна присутня в достатній мірі. Інноваційний підхід проявлено в розробці методу здійснення річкової переправи транспортних заходів за мурашиним алгоритмом, що дозволяє за множиною наявних параметрів автоматизовано визначати множини маршрутів для здійснення переправи.

Результати дослідження доповідались на 1-й конференції та оприлюднені в 1-х тезах.

6. Ступінь оволодіння методами дослідження

Магістрант в достатній мірі оволоділа методами дослідження, які були використані у роботі: порівняння, аналізу, класифікації, узагальнення.

7. Повнота та якість розкриття теми роботи

Тема роботи розкрита достатньо в рамках поставлених завдань: спроектовано структуру і розроблено метод та прикладне програмне забезпечення здійснення річкової переправи транспортних засобів за множиною маршрутів здійснення переправи та вибору між ними оптимального. Інформаційна система включає в себе базу даних, базу знань і п'ять функціональних модулів, що мають різне призначення: модуль роботи користувача з експертними даними, модуль роботи користувача з каталоговими даними, модуль взаємодії користувача з оперативними даними, модуль формування множини можливих маршрутів та модуль оптимізації за критеріями часу, відстані та ваги замовлення і виведення користувачу результатів роботи.

8. Логічність, послідовність, аргументованість, літературна грамотність викладу матеріалу

Магістрант матеріал виклав логічно, послідовно, аргументовано. Наводилися наявні розробки, ставилося задача та послідовно розв'язувалася. Для аргументації отриманих рішень проводилося теоретичне обґрунтування та порівняльний аналіз експериментів.

Літературна та граматична якість матеріалу на достатньому рівні.

9. Можливість практичного застосування кваліфікаційної роботи, окремих її частин

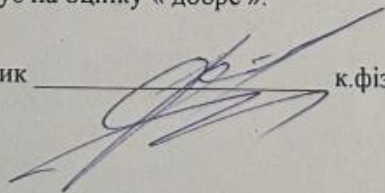
Результати кваліфікаційної роботи магістра можуть застосовуватися на практиці після налаштувань під конкретно запропоновану модель здійснення річкової переправи транспортних засобів. Практична цінність роботи полягає в тому, що при застосуванні методу оптимальної річкової переправи виконується не тільки відкидання параметрично неприйнятних маршрутів, а й запити користувачу на встановлення параметрів переправи, що дозволяє в процесі роботи методу розширити функціональну картину та взяти до розгляду відпочатку безпідставні маршрути.

10. Висновок про можливість допуску кваліфікаційної роботи до захисту, на яку оцінку заслуговує робота

Рекомендую допустити кваліфікаційну роботу магістра до захисту.

Робота заслуговує на оцінку « добре ».

Науковий керівник



к.фіз.-мат.н., доц. Віталій Міхалевський