



## ВИКОРИСТАННЯ МУРАШИНОГО АЛГОРИТМУ ДЛЯ РОЗВ'ЯЗАННЯ НЕЧІТКОЇ ЗАДАЧІ КОМІВОЯЖЕРА

**Вступ.** Сформульовано та наведено методику пошуку оптимальної тривалості маршруту для розв'язання задачі комівояжера у випадку визначення часу переміщення між містами у вигляді нечітких трапецієподібних чисел. Метою роботи є розроблення алгоритму на основі оптимізації колонії мурах і використання цього методу для розв'язування задач комівояжера з достатньо великою кількістю міст транспортної мережі.

**Методи.** Використано метод на основі алгоритму оптимізації мурашиної колонії.

**Результати.** Для досягнення поставленої мети запропоновано схему реалізації оптимізаційного алгоритму, що за умови невеликої кількості ітерацій дозволяє отримувати наближені до оптимальних розв'язків результати пошуку шляхів у нечіткій задачі комівояжера. Запропонований підхід може бути використаний для пошуку раціонального шляху в ситуаціях із неточно заданою тривалістю переміщень між містами. Показано, що вибір основних параметрів алгоритму оптимізації колонії мурах суттєво не впливає на якість отриманого наближеного розв'язку. Приклади використання алгоритму підтверджують конструктивність підходу для розв'язання задачі комівояжера у випадку нечітко заданої тривалості переміщень.

**Висновки.** Запропоновано схему реалізації алгоритму оптимізації мурашиної колонії для пошуку найкращого шляху в задачі комівояжера зі змінною тривалістю переміщень між містами, розроблено комп'ютерну програму, яка дозволяє розв'язувати різні логістичні задачі, в основу яких покладено задачу комівояжера з нечітко визначеними параметрами руху у транспортній мережі.

**Ключові слова:** нечітка задача комівояжера, оптимізаційний метод мурашиної колонії, трапецієподібні нечіткі числа, дефазифікація, оцінювання ефективності.

### Вступ

Один із перспективних напрямів науково-практичних досліджень соціальних та інформаційних процесів базується на використанні математичних методів, у яких закладено принципи реалізації природних механізмів для прийняття рішень. Ройовий інтелект – це відносно новий технологічний підхід, який формалізується на основі аналізу соціальної поведінки тварин і комах. Зокрема і спостереження за мурахами дозволили розробити ряд методів і прийомів, серед яких найбільш вивченим і найуспішнішим є метод оптимізації загального вигляду, відомий як оптимізація колонії мурах (Ant System Optimization, ASO). Імітація самоорганізації мурашиної колонії становить основу мурашиних алгоритмів оптимізації – нового перспективного методу природних обчислень. Іншими природними прототипами для оптимізаційних методів також можуть бути: поведінка бабок (алгоритм рою бабок, BFO), бджіл (алгоритм бджолиного рою, BA), термітів (алгоритм термітів), риб (алгоритм рибної зграї, FSO) та вовків (алгоритм вовчої зграї, WSA).

Під реалізацією ройового інтелекту розуміють спосіб розв'язування різноманітних оптимізаційних проблем за допомогою групи агентів, які взаємодіють між собою згідно з простими правилами, за якими функціонує складна поведінка всієї системи. Стосовно його використання для методики оптимізації найголовнішою перевагою є можливість знаходити глобальні оптимуми у задачах із великою кількістю параметрів та обмежень, а також гнучкість, масштабованість, можливість розподілених обчислень і захист від відмов. Системи на базі ройового інтелекту дозволяють оперативно знаходити ефективні розв'язки за умов динамічних змін параметрів, відмов окремих агентів, а також ці системи не потребують задання умов централізованого керування.

З іншого боку, серед недоліків ройового інтелекту варто зазначити складність налаштування параметрів, що збільшує ризик знаходження локальних оптимумів оптимальних розв'язків, високі вимоги до обчислювальних ресурсів і необхідність експериментального оцінювання.

Розглянемо застосування ройового інтелекту для розв'язування задач оптимізації на прикладі використання ASO (Dorigo, Maniezzo, & Colomi, 1996). Мурашині алгоритми широко використовуються вченими із середини 90-х рр. Першу версію алгоритму запропонував Марко Доріго 1992 р. (Dorigo, Maniezzo, & Colomi, 1991). Нині вже отримано непогані результати мурашиної оптимізації для розв'язання таких складних комбінаторних завдань: задачі оптимізації маршрутів вантажівок, завдання розмальовки графа, квадратична задача про призначення, оптимізація мережних графіків, задачі календарного планування тощо (Bell, 2004; Zhao, Luo, & Zhang, 2010; Raspinelli, 2002). Особливо ефективними є мурашині алгоритми за online оптимізації процесів у розподілених нестационарних системах, наприклад, для розв'язання проблем розподілу трафіка у телекомунікаційних мережах (Schoonderwoerd, 1996).

Колонія мурах може розглядатися як багатоагентна система, в якій кожен агент (мураха) функціонує автономно за визначеними правилами. Поведінка кожного агента обумовлена простими випадковими правилами. Цей принцип збігається з поведінкою мурах у реальному світі, де вони працюють разом для будівництва гнізд, пошуку їжі та захисту колонії. У роботах (Bonavear, 1999; Bullnheimer, 1999) показано, що на базі примітивної поведінки окремих агентів, поведінка сукупної системи дозволяє отримати найкращі результати для різного класу задач.

Ідея алгоритму сформована на основі поведінки мурашиної колонії, яка знаходить шлях до їжі, близький до оптимального. Основу поведінки мурах складає самоорганізація – сукупність динамічних механізмів, за допомогою яких система досягає глобальної мети в результаті взаємодії елементів на низькому рівні. Принциповою особливістю такої низькорівневої взаємодії є використання елементами системи лише локальної інформації, без будь-яких правил



централізованого керування та звернення до даних, що описують глобальні параметри зовнішнього середовища у вигляді обмежень. Самоорганізація агентів є наслідком взаємодії таких чотирьох компонентів:

- випадковість;
- додатний зворотний зв'язок;
- від'ємний зворотний зв'язок;
- багатократність взаємодій.

Багатократність взаємодії реалізується у формі послідовного ітераційного пошуку маршруту одночасного декількох мурахами. Кожна мураха починає власний рух випадковим чином, коли покидає мурашник у пошуках їжі. Вважають, що кожен агент не рухається певним відомим шляхом або заздалегідь відомим напрямком. Ця експлоративна поведінка дозволяє мурахам досліджувати широку ділянку навколо мурашника. Ключовим аспектом поведінки мурах є здатність залишати на власному шляху хімічні сліди – феромони. Ці феромони слугують сигналами для інших мурах і вказують на те, що шлях вже був досліджений і використовується. Наприклад, у класичній задачі пошуку маршруту комівояжера на мережі, заданій у вигляді графа, додатний обернений зв'язок реалізується таким стохастичним правилом: "імовірність включення ребра графа в маршрут мурахи пропорційна кількості феромона на ній". Коли мураха знаходить їжу, вона повертається до мурашника і залишає феромоновий слід, який допомагає іншим мурахам знаходити шлях до їжі. Кількість феромона, який відкладає мураха на етапі маршруту, є обернено пропорційною до довжини відповідної ділянки. Чим коротший шлях у процесі пошуку знайшла мураха, тим більше феромона буде відкладено на відповідних етапах маршруту (ребрах графа).

Зауважимо, що використання лише додатного оберненого зв'язку веде до швидкої (передчасної) збіжності алгоритму, тобто до випадку, коли усі мурахи рухаються тим самим субоптимальним маршрутом. Для запобігання перенасиченню шляхів феромони із часом випаровуються, що реалізує від'ємний обернений зв'язок. Це дозволяє мурахам адаптуватися до змін у навколишньому середовищі (напр., поява нових джерел їжі або завод на шляхах). Випаровування феромонів забезпечує динамічне оновлення інформації та дозволяє уникнути вибору неоптимальних шляхів (уникнення локальних оптимумів). Час випаровування феромона не повинен бути надто великим для запобігання загрози збіжності маршрутів руху всіх мурах до одного субоптимального розв'язку. З іншого боку, час випаровування не має бути й малим, щоб не призвести до некооперативної поведінки мурах через втрату пам'яті колонії.

Завдяки цим механізмам міжагентної взаємодії система саморганізується і дозволяє окремим мурахам використовувати оптимальні шляхи до джерел їжі з можливістю ефективного розв'язання задач розподілу та пошуку ресурсів без централізованого керування.

З математичного погляду модель ASO описується через такі базові компоненти, що пов'язані з поведінкою як окремих мурах, так і системи в цілому:

- механізми формування шляхів;
- розміщення феромонів для позначення пройдених шляхів;
- випаровування феромонів;
- правила вибору шляху мурахою.

Однією із задач, для якої можна запропонувати спосіб розв'язання на основі мурашиного алгоритму, є логістична задача комівояжера (Dantzig, 1954). **Метою** вказаного дослідження є адаптація алгоритму ASO для розв'язання задачі комівояжера (ЗК) з нечітко заданими параметрами переміщень на етапах транспортної мережі, досягнення якої пов'язано з проведенням **завдань** формалізації нечіткої задачі на основі використання нечітких трапецієподібних чисел, реалізації компонентів самоорганізаційної поведінки мурах для оптимізації маршруту комівояжера та проведення чисельних експериментів для визначення ефективності розробленого методу.

**Постановка задачі.** Стандартна постановка задачі комівояжера полягає у виборі найкоротшого за довжиною або часом замкненого шляху на мережі з  $n$  міст, що проходить через кожне місто рівно один раз. Кількість можливих варіантів дорівнює  $(n-1)!$ , а за умов симетричності етапів маршруту кількість унікальних маршрутів становить  $(n-1)!/2$ .

Знаходження оптимального шляху комівояжера, навіть у випадку відносно невеликої транспортної мережі у 60–70 міст, потребує надзвичайно великої кількості обчислювальних ресурсів для розрахунку, що призводить до необхідності використання наближених алгоритмів, таких як ASO.

У реальних задачах логістики поняття тривалості або вартості подорожі між окремими пунктами транспортної мережі не можуть бути фіксованими, вони визначається наближено, часто із впливом суб'єктивного фактора щодо оцінок часових термінів або вартості переміщення за ділянками маршруту. Це призводить до необхідності врахування невизначеності, її формалізації на основі різних методик. Одним із підходів, що використовуються у цьому випадку, є залучення нечітких чисел і реалізація засобів маніпуляції з ними.

Розглянемо постановку задачі комівояжера з нечітко заданою тривалістю переміщень на транспортній мережі. У такому випадку необхідно знайти циклічну перестановку номерів міст, які має відвідати комівояжер, відповідно до якої затрати часу будуть мінімальні з урахуванням обмеження щодо відвідування кожного з пунктів не більше одного разу. Математичне формулювання нечіткої задачі комівояжера можна записати так: потрібно мінімізувати відповідно до вказаного вище способу порівняння нечітких чисел цільову функцію

$$\sum_{i=1}^n \sum_{j=1}^n \tilde{t}_{ij} x_{ij}, \quad (1)$$

де часові витрати на переміщення між пунктами задаються у вигляді матриці  $\tilde{T} = \{\tilde{t}_{ij}\}$ ,  $i, j = \overline{1, n}$ , з елементами у вигляді нечітких чисел (Kumar, 2011), а можливі шляхи переміщень між містами визначаються матрицею  $X$ , за умови виконання обмежень:

$$\sum_{i=1}^n x_{ij} = 1 \text{ для всіх } j = 1, 2, \dots, n, \quad \sum_{j=1}^n x_{ij} = 1 \text{ для всіх } i = 1, 2, \dots, n, \quad (2)$$



$$i - j + nx_{ij} \leq n - 1, \quad 1 \leq i \neq j \leq n,$$

$$x_{ij} = 0 \text{ або } 1 \text{ для всіх } i, j = 1, 2, \dots, n.$$

Для програмної реалізації у матриці  $\tilde{T}$  діагональні елементи  $\tilde{t}_{ii}$  необхідно задати великими додатними числами, щоб отримувати у розв'язку величини  $x_{ii} = 0$  для всіх  $i = 1, 2, \dots, n$ .

Нечіткі величини тривалості переміщення  $\tilde{t}_{ij}$  між довільними містами  $i, j = \overline{1, n}$ , задаватимемо у вигляді трапецієподібних нечітких чисел.

**Означення.** Нечітким трапецієподібним числом  $\tilde{A}$  (Кутар, 2011) називають упорядковану четвірку дійсних чисел  $(a_1, a_2, a_3, a_4)$ ,  $a_1 \leq a_2 \leq a_3 \leq a_4$ , для яких визначено функцію належності  $\mu_{\tilde{A}}(x)$  вигляду

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x - a_1}{a_2 - a_1}, & \text{якщо } a_1 \leq x \leq a_2; \\ 1, & \text{якщо } a_2 \leq x \leq a_3; \\ \frac{a_4 - x}{a_4 - a_3}, & \text{якщо } a_3 \leq x \leq a_4. \end{cases} \quad (3)$$

Якщо до подання трапецієподібного нечіткого числа застосувати підхід на основі гауссівського розподілу з відповідними характеристиками, то в узагальненому випадку трапецієподібне нечітке число можна представити дещо в іншому вигляді як

$$\tilde{A} = (a_1, a_2, a_3, a_4) = ([a_2, a_3], \alpha, \beta) = (m, w, \alpha, \beta), \quad (4)$$

де використовується середня точка  $m = \frac{a_2 + a_3}{2}$  та півширина плато  $w = \frac{a_3 - a_2}{2}$ , а коефіцієнти  $\alpha = a_2 - a_1$  та  $\beta = a_4 - a_3$  визначають лівий і правий розподіл нечіткого числа  $\tilde{A} = (a_1, a_2, a_3, a_4)$  відповідно.

Для оперування з нечіткими числами потрібно визначити операції, з огляду на наведений вище опис. За середню точку беруть звичайне середньоарифметичне значення границь плато, лівий і правий розподіли розглядають відповідно до правила ґратки, за яким для довільних дійсних чисел  $a, b$  покладемо  $a \cup b = \max\{a, b\}$  та  $a \cap b = \min\{a, b\}$ .

Тоді для довільних трапецієподібних нечітких чисел  $\tilde{A} = (m(\tilde{A}), w(\tilde{A}), \alpha_1, \beta_1)$  та  $\tilde{B} = (m(\tilde{B}), w(\tilde{B}), \alpha_2, \beta_2)$  можна визначити операції додавання, віднімання, множення та ділення, які у загальному випадку позначимо символом  $\circ$ :

$$\begin{aligned} \tilde{A} \circ \tilde{B} &= (m(\tilde{A}) \circ m(\tilde{B}), w(\tilde{A}) \cup w(\tilde{B}), \alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2) = \\ &= (m(\tilde{A}) \circ m(\tilde{B}), \max(w(\tilde{A}), w(\tilde{B})), \max(\alpha_1, \alpha_2), \max(\beta_1, \beta_2)). \end{aligned} \quad (5)$$

Остаточно маємо

$$\begin{aligned} \tilde{A} + \tilde{B} &= (m(\tilde{A}) + m(\tilde{B}), w(\tilde{A}) \cup w(\tilde{B}), \alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2) = \\ &= (m(\tilde{A}) + m(\tilde{B}), \max(w(\tilde{A}), w(\tilde{B})), \max(\alpha_1, \alpha_2), \max(\beta_1, \beta_2)). \\ \tilde{A} - \tilde{B} &= (m(\tilde{A}) - m(\tilde{B}), w(\tilde{A}) \cup w(\tilde{B}), \alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2) = \\ &= (m(\tilde{A}) - m(\tilde{B}), \max(w(\tilde{A}), w(\tilde{B})), \max(\alpha_1, \alpha_2), \max(\beta_1, \beta_2)). \\ \tilde{A} \times \tilde{B} &= (m(\tilde{A}) \times m(\tilde{B}), w(\tilde{A}) \cup w(\tilde{B}), \alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2) = \\ &= (m(\tilde{A}) \times m(\tilde{B}), \max(w(\tilde{A}), w(\tilde{B})), \max(\alpha_1, \alpha_2), \max(\beta_1, \beta_2)). \\ \tilde{A} \div \tilde{B} &= (m(\tilde{A}) \div m(\tilde{B}), w(\tilde{A}) \cup w(\tilde{B}), \alpha_1 \cup \alpha_2, \beta_1 \cup \beta_2) = \\ &= (m(\tilde{A}) \div m(\tilde{B}), \max(w(\tilde{A}), w(\tilde{B})), \max(\alpha_1, \alpha_2), \max(\beta_1, \beta_2)). \end{aligned}$$



Для проведення операцій порівняння та ранжування нечітких чисел використаємо спосіб на основі медіанного середнього значення. Іншими словами, якщо для кожного  $\tilde{A} = (a_1, a_2, a_3, a_4) \in F(R)$  визначено функцію ранжування  $\mathfrak{R}: F(R) \rightarrow R$  із медіанним середнім значенням у вигляді  $\mathfrak{R}(\tilde{A}) = \left[ \left( \frac{a_2 + a_3}{2} \right) + \left( \frac{\beta - \alpha}{4} \right) \right]$ , тоді для довільних двох трапецієподібних нечітких чисел  $\tilde{A} = (a_1, a_2, a_3, a_4)$  та  $\tilde{B} = (b_1, b_2, b_3, b_4)$  маємо такі можливі варіанти порівняння:

- $\tilde{A} \succ \tilde{B}$  тоді і лише тоді, якщо  $\mathfrak{R}(\tilde{A}) > \mathfrak{R}(\tilde{B})$ ;
- $\tilde{A} \prec \tilde{B}$  тоді і лише тоді, якщо  $\mathfrak{R}(\tilde{A}) < \mathfrak{R}(\tilde{B})$ ;
- $\tilde{A} \approx \tilde{B}$  тоді і лише тоді, якщо  $\mathfrak{R}(\tilde{A}) = \mathfrak{R}(\tilde{B})$ .

Процеси оброблення нечітких чисел передбачають етап дефазифікації – перетворення нечіткого результату до чіткого (числового) значення. Це важливий крок у методиці застосування нечіткого підходу, особливо в задачах нечіткого керування та нечіткої бізнес-логіки, де потрібно перетворити нечіткі розв'язки на конкретні події або числові значення. Існують різні методи дефазифікації, серед яких найпоширенішими є: метод центра тяжіння (Center of Gravity, CoG) або центроїду, метод середнього максимуму та метод максимуму. Для порівняння результатів дослідження використовуватимемо метод центра тяжіння. У цьому методі точка дефазифікації обчислюється як центр тяжіння нечіткої множини. Для випадку неперервного способу подання нечітких чисел формула розрахунку центра тяжіння має вигляд

$$CoG = \frac{\int_{a_1}^{a_4} x \cdot \mu(x) dx}{\int_{a_1}^{a_4} \mu(x) dx}, \quad (6)$$

а для дискретної нечіткої множини:

$$CoG = \frac{\sum_{i=1}^n \mu(x_i) \cdot x_i}{\sum_{i=1}^n \mu(x_i)}, \quad (7)$$

де  $x_i$  – точки, що визначають результат, а  $\mu(x_i)$  – ступінь належності кожної точки нечіткій множині,  $i = \overline{1, n}$ .

### Методи

Формування шляхів у моделі оптимізації колонії мурах описують за допомогою графа. Нехай  $G = (V, E)$  являє собою граф, де  $V$  – це множина з  $m$  вершин, а  $E$  – множина ребер. Кожному ребру, що з'єднує вершини  $i$  та  $j$  графа  $G$ ,  $(i, j) \in E$ , ставлять у відповідність два параметри: час переміщення  $T_{ij}$  по ребру  $(i, j)$  (зазвичай пропорційний довжині шляху  $D_{ij}$ ) та інтенсивність феромона  $\tau_{ij}$  на ребрі  $(i, j)$ ,  $i, j = m$ .

Збереження рівня феромонів є ключовим процесом для міжагентної взаємодії мурах між собою. Інтенсивність феромонів на ребрі  $\tau_{ij}$  оновлюється на основі досвіду мурах, що пройшли за цим ребром. Для оновлення рівня феромона часто використовують формулу

$$\tau_{ij}(s+1) = (1-\rho) \cdot \tau_{ij}(s) + \Delta\tau_{ij}(s),$$

де  $\rho$  – коефіцієнт випаровування феромона,  $0 < \rho < 1$ , а  $\Delta\tau_{ij}(s)$  – кількість феромона, що залишається мураками на ребрі  $(i, j)$  на ітерації  $s$ ,  $s = 0, 1, 2, \dots$ .

Випаровування феромонів зменшує інтенсивність феромона на всіх ребрах, що дозволяє системі забувати попередні (можливо неоптимальні) шляхи й адаптуватись до змін. Це запобігає передчасній збіжності до локальних оптимумів.

Вибір кожного етапу шляху руху мурахи базується на випадкових правилах. Коли на ітерації  $s$  мураха  $k$  перебуває у вершині  $i$ , вона обирає наступну вершину  $j$  з певною ймовірністю  $P_{ij}^k(s)$ , яка залежить від інтенсивності феромона і видимості  $\eta_{ij}$ , яка є обернено пропорційною довжині шляху  $\eta_{ij} = \frac{1}{D_{ij}}$ :

$$\begin{cases} P_{ij}^k(s) = \frac{(\tau_{ij}(s))^\alpha (\eta_{ij})^\beta}{\sum_{l \in J_i^k} (\tau_{il}(s))^\alpha (\eta_{il})^\beta}, \text{ якщо } j \in J_i^k; \\ P_{ij}^k(s) = 0, \text{ якщо } j \notin J_i^k, \end{cases} \quad (8)$$



де  $\alpha$  та  $\beta$  – регульовані параметри, що контролюють відносний вплив феромона та видимості, відповідно, а сума у знаменнику розраховується за всіма доступними ребрами (множина доступних вершин  $J_i^k$  з вершини  $i$ ). Якщо  $\alpha = 0$ , то найвірогіднішим буде перехід у найближчі міста. У класичній теорії оптимізації це відповідає так званому, жадібному алгоритму. Якщо  $\beta = 0$ , тоді працює лише феромонне підсилення, що призводить до швидкого завершення роботи алгоритму через збіжність маршрутів усіх мурах до одного субоптимального розв'язку.

Зазначимо, що сума всіх імовірностей переходу з вершини  $i$  за всіма можливими варіантами з множини  $J_i^k$  на ітерації  $s$  дорівнює 1:

$$\sum_{j \in J_i^k} P_{ij}^k(s) = 1. \quad (9)$$

Основна мета мурашиного алгоритму – мінімізація довжини шляху  $L$ , який є сумою довжин ребер на шляху мурахи:

$$L = \sum_{(i,j) \in P} D_{ij}, \quad (10)$$

де  $P$  позначає множину ребер, з яких складається шлях мурахи.

Розглянемо схему реалізації чотирьох основних компонентів самоорганізаційної поведінки мурах під час оптимізації маршруту комівояжера. Послідовно реалізуючи ітераційні кроки, що відтворюють процедуру знаходження маршруту кожною мурахою, отримуємо схему функціонування мурашиного алгоритму для розв'язання нечіткої задачі комівояжера. Перехід мурахи з міста  $i$  в місто  $j$  на ітерації  $s$  алгоритму залежить від трьох складових: табу-списку, видимості та віртуального сліду феромона. Табу-список  $A_i^k$  – це перелік міст, які вже відвідані мурахою  $k$  до вершини  $i$  і заходять в які ще раз заборонено. Цей перелік збільшується з проходженням за маршрутом та очищується на початку кожної ітерації алгоритму. Позначимо через  $J_i^k$  перелік міст, які ще потрібно відвідати мураці  $k$ , що перебуває у місті  $i$ . Зрозуміло, що об'єднання переліків  $A_i^k$  та  $J_i^k$  дає множину всіх міст, указаних у задачі комівояжера.

На першій ітерації  $s = 0$  кількість феромонів на кожному шляху  $\tau_{ij}(0) = 0$ .

Генерується набір мурах в кожній вершині графа.

В процесі ітерації кожна мураха діє окремо. В першому місті  $i$  табу-перелік  $A_i^k$  для мурахи  $k$  складається з міста, в якому перебуває мураха  $-A_i^k = \{i\}$ . Далі кожна мураха обирає з певною ймовірністю наступне місто для переміщення, з урахуванням заданої часової тривалості пересування до найближчих міст за формулою (1). Для цього використовуватимемо генератор випадкових чисел.

Після обрання наступного міста  $j$ , додаємо це місто до табу-переліку  $A_j^k = \{i, j\}$ . На наступному кроці обираємо наступне місто для переходу і так далі до останнього міста в маршруті. Якщо маршрут не можна замкнути, то мураха вважається недієвою до наступної ітерації.

Після завершення ітерації розраховуємо випаровування феромона для кожного можливого ребра. Далі для кожної мурахи з успішно завершеним маршрутом після виконання ітерації  $s$  розраховуємо тривалість маршруту і для кожного ребра, використаного в маршруті, додаємо феромон у кількості, яка є обернено пропорційною тривалості маршруту для кожної мурахи:

$$\Delta\tau_{ij}(s) = \frac{Q}{L_k},$$

де  $L_k$  – тривалість успішного маршруту кожної мурахи  $k$ , а ребро  $(i, j) \in L_k$  належить до маршруту мурахи  $k$ ;  $Q$  – регульований параметр, значення якого обирають одного порядку з довжиною оптимального маршруту (Bullnheimer, 1999).

Отже, за умови коректно обраних значень параметрів  $\alpha$  та  $\beta$  з кожною ітерацією поступово отримуємо покращені результати.

#### Результати обчислень

Наведемо результати проведених чисельних експериментів. На попередньому етапі алгоритм ASO був адаптований до розв'язання задачі комівояжера з нечіткою тривалістю переміщень між містами. Для цього виконано фазифікацію задачі, наведеної в роботі [Ivohin, 2023] за допомогою трапецієподібних нечітких чисел (рис. 1).

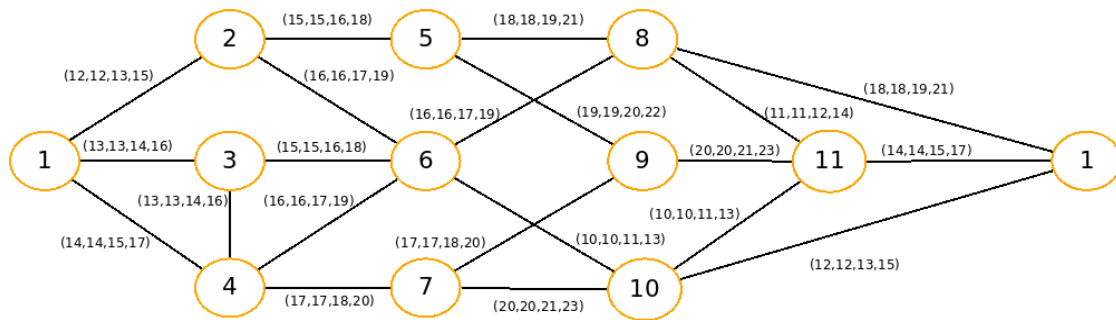


Рис. 1. Схема-приклад задачі комівояжера з нечіткою тривалістю

Ранжування маршрутів із нечіткою тривалістю виконували за допомогою методу обчислення COG у вигляді (6) для неперервного способу подання нечітких чисел.

Для даної задачі проведено чисельні розрахунки маршруту комівояжера методом повного перебирання та за допомогою алгоритму ASO. Застосування алгоритму ASO для задачі комівояжера у вказаній конфігурації характеризується високою швидкістю збіжності та забезпечує отримання найкращого результату за досить консервативних параметрів ASO (табл. 1).

Таблиця 1

Параметри алгоритму ASO

$\alpha$	$\beta$	$Q$	$V_{var}$	Ітерація
0,1	-1	20	0,05	4
0,1	-2	20	0,05	3
0,1	-4	20	0,05	2
0,15	-1	20	0,05	5
0,2	-1	20	0,05	5
0,25	-1	20	0,05	7
0,3	-1	20	0,05	8

У стовпці "Ітерація" вказано першу ітерацію, на якій отримано найкращий розв'язок із найбільшою ймовірністю.

Оптимальна тривалість маршруту в задачі комівояжера на основі повного перебирання визначається нечітким трапецієподібним числом (156, 156, 167, 189). Будемо вважати цей розв'язок точним. Під час застосування мурашиного алгоритму для розв'язання задачі отримано аналогічний розв'язок, причому результат досягається, зазвичай, вже на 3-й або 4-й ітерації, а його дефазифікована на основі методу CoG величина складає 167,92 одиниць.

Подальші експерименти з ASO проводили для оцінювання якості отриманого результату з урахуванням різної кількості міст транспортної мережі. У чисельних розрахунках генетичний алгоритм використовували для розв'язання 3К у разі випадкового розміщення  $n = 16, 17, 18$  міст на двовимірній площині  $200 \times 200$ , для якого величини переміщення між містами мережі визначались за середньою тривалістю нечіткого часу, яка вважалась пропорційною відстані між відповідними містами. Проведено три варіанти чисельних експериментів: кількість ітерацій алгоритму ASO дорівнює чотирикратній кількості міст (табл. 2), подвійній кількості міст (табл. 3), кількості міст (табл. 4). Кількість мурах дорівнює кількості міст у кожному випадку.

Для кожного експерименту проведено оцінювання впливу зміни параметрів  $\beta$  та  $\alpha$  на остаточний результат. Порівняння найкращого результату, отриманого за допомогою алгоритму ASO за задану кількість ітерацій, з найкращим результатом, отриманим методом повного перебирання, наведено в табл. 2, 3 та 4. Значення якості порівняння результатів подано у відсотках, які визначаються величиною отриманого збільшення довжини шляху щодо результату, який отримано методом повного перебирання.

Таблиця 2

Параметри наближеного оптимального розв'язку (кількість ітерацій дорівнює чотирикратній кількості міст)

$V_{var}$	$\alpha$	$\beta$	$Q$	Максимальне відносне відхилення від точного розв'язку для різної кількості міст, %		
				16	17	18
0,05	0,15	-1	20	3,72	4,31	4,0
0,05	0,15	-1,5	20	3,56	3,81	4,28
0,05	0,15	-2	20	3,67	3,73	4,19



0,05	0,15	-2,5	20	3,75	3,93	3,97
0,05	0,15	-3	20	3,68	4,18	4,27
0,05	0,15	-3,5	20	3,97	4,21	4,30
0,05	0,15	-4	20	3,88	4,11	4,33
0,05	0,1	-3	20	3,77	4,04	4,64
0,05	0,15	-3	20	3,68	3,78	3,87
0,05	0,2	-3	20	3,66	3,71	3,98
0,05	0,25	-3	20	4,02	4,10	4,19
0,05	0,3	-3	20	4,08	4,11	4,32

Таблиця 3

**Параметри наближеного оптимального розв'язку  
(кількість ітерацій дорівнює подвійній кількості міст)**

$V_{var}$	$\alpha$	$\beta$	$Q$	Максимальне відносне відхилення від точного розв'язку для різної кількості міст, %		
				16	17	18
0,05	0,15	-1	20	4,04	4,50	4,74
0,05	0,15	-1,5	20	3,94	4,48	4,78
0,05	0,15	-2	20	3,98	4,23	4,81
0,05	0,15	-2,5	20	3,99	4,17	4,26
0,05	0,15	-3	20	3,97	3,99	4,56
0,05	0,15	-3,5	20	4,10	4,26	4,38
0,05	0,15	-4	20	3,86	4,36	4,55
0,05	0,1	-3	20	3,88	4,11	4,58
0,05	0,15	-3	20	3,87	3,91	4,56
0,05	0,2	-3	20	4,01	4,40	4,47
0,05	0,25	-3	20	4,12	4,28	4,83
0,05	0,3	-3	20	4,34	4,44	4,72

Таблиця 4

**Параметри наближеного оптимального розв'язку  
(кількість ітерацій дорівнює кількості міст)**

$V_{var}$	$\alpha$	$\beta$	$Q$	Максимальне відносне відхилення від точного розв'язку для різної кількості міст, %		
				16	17	18
0,05	0,15	-1	20	4,75	4,92	4,96
0,05	0,15	-1,5	20	4,61	5,06	5,20
0,05	0,15	-2	20	4,42	4,45	4,69
0,05	0,15	-2,5	20	4,35	4,69	5,11
0,05	0,15	-3	20	4,54	4,57	4,75
0,05	0,15	-3,5	20	4,38	4,39	4,71
0,05	0,15	-4	20	4,11	5,12	5,15
0,05	0,1	-3	20	4,37	4,50	5,13
0,05	0,15	-3	20	4,34	4,47	4,75
0,05	0,2	-3	20	4,31	4,77	4,83
0,05	0,25	-3	20	4,45	4,70	4,83
0,05	0,3	-3	20	4,47	4,96	5,05

Як видно з наведених результатів за умови здійснення кількості ітерацій, що дорівнює кількості міст, алгоритм дозволяє отримати результати, що лежать у межах 5 % від найкращого. Збільшення кількості ітерацій веде до



покращення результатів. Варіація параметра  $\beta$  в межах від  $-4$  до  $-1$  та параметра  $\alpha$  в межах від  $0,1$  до  $0,3$  якісно впливають на результати, які у цьому випадку розміщені в межах  $0,5\%$  від найкращого результату розв'язування задачі комівояжера, який отримано методом повного перебирання.

#### Висновки

В роботі розглянуто та проаналізовано застосування алгоритму оптимізації колонії мурах для розв'язання задачі комівояжера з нечітко заданою тривалістю переміщень на транспортній мережі. Запропоновано схему реалізації алгоритму, що за умови невеликої кількості ітерацій дозволяє отримувати наближені до оптимальних розв'язків результати. Запропонований підхід може бути використаний для пошуку найкращого шляху в ситуаціях зі змінною тривалістю переміщень між містами. Показано, що вибір основних параметрів алгоритму оптимізації колонії мурах суттєво не впливає на якість отриманого наближеного розв'язку. Наведено приклади використання алгоритму, що підтверджують конструктивність підходу для знаходження розв'язків задачі комівояжера у випадку задання нечіткої тривалості переміщень.

**Внесок авторів.** Євген Івохін – розроблення методики та методології дослідження, написання висновків; Костянтин Юштин – огляд літературних джерел, адаптація алгоритму, проведення емпіричних досліджень та опис результатів.

#### Список використаних джерел

- Bell, J., & McMullen, P. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, 18(1), 41–48.
- Bonaventure, E., & Dorigo, M. (1999). *Swarm Intelligence: from Natural to Artificial Systems*. Oxford University Press.
- Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). A New Rank-Based Version of the Ant System: A Computational Study. *Central European Journal for Operations Research and Economics*, 1(7), 25–38.
- Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). Solution of a Large-scale Traveling Salesman Problem. *Operations Research*, 2.
- Dorigo, M., Maniezzo, V., & Colomi, A. (1991). *Positive feedback as a search strategy* Dipartimento di Elettronica, Politecnico di Milano, Tech. Rep. 91-016.
- Dorigo, M., Maniezzo, V., & Colomi, A. (1996). *The Ant System: Optimization by a colony of cooperating agents*. IEEE Transactions on Systems, Man, and Cybernetics. Part B. *Cybernetics*, 26(1), 29–41.
- Ivohin, E. V., Gavrylenko, V. V., & Ivohina, K. E. (2023). On the recursive algorithm for solving the traveling salesman problem on the basis of the data flow optimization method. *Radioelectronics, Computer Science, Control*, 3, 141–147.
- Kumar, A., & Gupta, A. (2011). Methods for solving fuzzy assignment problems and fuzzy travelling salesman problems with different membership functions. *Fuzzy Information and Engineering*, 3(1), 3–21.
- Raspinielli, J. M., Lopes, H. S., & Freitas, A. A. (2002). Data Mining with an Ant Colony Optimization Algorithm. *IEEE Trans. On Evolutionary Computation. Special issue on Ant Colony Algorithms*, 6(4), 321–332.
- Schoonderwoerd, R., Holland, O., Bruten, J., & Rothkranz, L. (1996). *Ant-based load balancing in telecommunication networks*. Adaptive Behavior, 5(2).
- Zhao, D., Luo, L., & Zhang, K. (2010). An improved ant colony optimization for the communication network routing problem. *Mathematical and Computer Modelling*, 52(11–12), 1976–1981.

#### References

- Bell, J., & McMullen, P. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, 18(1), 41–48.
- Bonaventure, E., & Dorigo, M. (1999). *Swarm Intelligence: from Natural to Artificial Systems*. Oxford University Press.
- Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). A New Rank-Based Version of the Ant System: A Computational Study. *Central European Journal for Operations Research and Economics*, 1(7), 25–38.
- Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). Solution of a Large-scale Traveling Salesman Problem. *Operations Research*, 2.
- Dorigo, M., Maniezzo, V., & Colomi, A. (1991). *Positive feedback as a search strategy* Dipartimento di Elettronica, Politecnico di Milano, Tech. Rep. 91-016.
- Dorigo, M., Maniezzo, V., & Colomi, A. (1996). *The Ant System: Optimization by a colony of cooperating agents*. IEEE Transactions on Systems, Man, and Cybernetics. Part B. *Cybernetics*, 26(1), 29–41.
- Ivohin, E. V., Gavrylenko, V. V., & Ivohina, K. E. (2023). On the recursive algorithm for solving the traveling salesman problem on the basis of the data flow optimization method. *Radioelectronics, Computer Science, Control*, 3, 141–147.
- Kumar, A., & Gupta, A. (2011). Methods for solving fuzzy assignment problems and fuzzy travelling salesman problems with different membership functions. *Fuzzy Information and Engineering*, 3(1), 3–21.
- Raspinielli, J. M., Lopes, H. S., & Freitas, A. A. (2002). Data Mining with an Ant Colony Optimization Algorithm. *IEEE Trans. On Evolutionary Computation. Special issue on Ant Colony Algorithms*, 6(4), 321–332.
- Schoonderwoerd, R., Holland, O., Bruten, J., & Rothkranz, L. (1996). *Ant-based load balancing in telecommunication networks*. Adaptive Behavior, 5(2).
- Zhao, D., Luo, L., & Zhang, K. (2010). An improved ant colony optimization for the communication network routing problem. *Mathematical and Computer Modelling*, 52(11–12), 1976–1981.

Отримано редакцією журналу / Received: 21.08.24  
Прорецензовано / Revised: 19.09.24  
Схвалено до друку / Accepted: 25.10.24

Eugene IVOHIN, DSc (Phys. & Math.), Prof.  
ORCID ID: 0000-0002-5826-7408  
e-mail: ivohin@knu.ua  
Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Kostyantyn YUSHTIN, Doctoral Student  
ORCID ID: 0009-0001-9881-2343  
e-mail: gkons@mail.univ.kiev.ua  
Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

## USE OF ANT COLONY OPTIMIZATION ALGORITHM FOR SOLVING FUZZY PROBLEM OF TRAVELING SALESMAN

**Background.** The method of finding the optimal route length for the traveling salesman problem in the case of determining the time of movement between cities in the form of fuzzy trapezoidal numbers is formulated and given. The purpose of the work is to develop an algorithm based on the optimization of an ant colony and use this method to solve problems of a traveling salesman with a sufficiently large number of cities in the transport network.

**Methods.** The method based on the ant colony optimization algorithm was used.

**Results.** In order to achieve the goal, a scheme for the implementation of the optimization algorithm is proposed, which, under the condition of a small number of iterations, allows obtaining results close to optimal solutions in the vague problem of the traveling salesman. The proposed approach can be used to find a rational path in situations with an imprecisely specified duration of movements between cities. It is shown that the selection of the main parameters of the ant colony optimization algorithm does not significantly affect the quality of the obtained approximate solution. Examples of the use of the algorithm confirm the constructiveness of the approach to solving the traveling salesman problem in the case of a vaguely specified duration of movements.



**Conclusions.** *A scheme for the implementation of the ant colony optimization algorithm for finding the best path in the problem of a traveling salesman with variable duration of movements between cities is proposed, a computer program has been developed that allows solving various logistics problems, which are based on the problem of a traveling salesman with vaguely defined movement parameters in the transport network.*

**Keywords:** *fuzzy traveling salesman problem, ant colony optimization method, trapezoidal fuzzy numbers, defuzzification, performance evaluation.*

Автори заявляють про відсутність конфлікту інтересів. Спонсори не брали участі в розробленні дослідження; у зборі, аналізі чи інтерпретації даних; у написанні рукопису; в рішенні про публікацію результатів.

All authors have read and agreed to the published version of the manuscript. The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.