

УДК 519.876.5

DOI: <https://doi.org/10.17721/3041-2323.2024.185-202>

Євген ІВОХІН¹, д-р фіз.-мат. наук, проф.
ORCID ID: 0000-0002-5826-7408
e-mail: ivohin@univ.kiev.ua

Костянтин ЮШТІН¹, канд. фіз.-мат. наук, доц.
ORCID ID: 0009-0001-9881-2343
e-mail: gkons@univ.kiev.ua

Валерій ГАВРИЛЕНКО², д-р фіз.-мат. наук, проф.
ORCID ID: 0000-0001-9682-4204
e-mail: vvgavrilenko1953@gmail.com

Максим БОГУСЛАВСЬКИЙ², асп.
e-mail: maxbogus09@gmail.com
ORCID ID: 0009-0000-1264-133X

¹Київський національний університет
імені Тараса Шевченка, Київ, Україна
²Національний транспортний університет,
Київ, Україна

ВИКОРИСТАННЯ МУРАШИНОГО АЛГОРИТМУ ДЛЯ РОЗВ'ЯЗАННЯ НЕЧІТКОЇ ЗАДАЧІ КОМІВОЯЖЕРА

Задача комівояжера (TSP) – це класична комбінаторна задача оптимізації, яка передбачає пошук найкоротшого або найшвидшого маршруту серед набору міст. Щоб формалізувати невизначеність і неточність у вхідних даних, часто викликає суб'єктивними оцінками інтервалів часу подорожі, у цій статті використано нечіткі числа. Форма цих нечітких чисел базується на підході, подібному до гаусівського. Розглянуто особливості застосування алгоритму оптимізації мурашиної колонії (ASO) і запропоновано підхід до його оптимального використання. Проаналізовано вплив параметрів алгоритму на якість апроксимованого найкращого рішення. Задачу проілюстровано числовими прикладами з участю достатньо великої кількості міст у транспортній мережі.

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

© Івохін Євген, Юштін Костянтин, Гавриленко Валерій,
Богуславський Максим, 2024

Вступ

Один з актуальних напрямів науково-практичних досліджень соціальних та інформаційних процесів базується на використанні математичних методів, у яких закладено принципи реалізації природних механізмів для прийняття рішень. Груповий (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 є також 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 за простими правилами, за якими функціонує складна поведінка всієї системи. Щодо його використання для методики оптимізації, то 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 і не потребують задання умов централізованого керування.

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

Розглянемо застосування ройового інтелекту для розв'язування задач нечіткої оптимізації на прикладі використання ASO

(Dorigo, & Caro, 1999). Мурашині алгоритми широко використовуються вченими із середини 90-х рр. Першу версію алгоритму запропонував Марко Доріго у 1992 р. (Dorigo, 1992). Нині вже отримано непогані результати мурашиної оптимізації у розв'язанні таких складних комбінаторних завдань як задачі оптимізації маршрутів вантажівок, завдання розмальовки графа, квадратичної задачі про призначення, а також оптимізація мережеских графіків, задачі календарного планування тощо (Stützle, & Hoos, 2000; Dorigo, 2004; Bouamama, Blum, & Fages, 2019). Особливо ефективними є мурашині алгоритми в online оптимізації процесів у розподілених нестационарних системах, наприклад, для розв'язання проблем розподілу трафіка у телекомунікаційних мережах (Dorigo, & Caro, 1998).

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 (Kennedy, & Eberhart, 2001; Karaboga, 2009) 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 глобальних параметрів зовнішнього середовища, заданих у вигляді обмежень.

Багатократність взаємодії реалізується у формі послідовного ітераційного пошуку маршруту одночасно кількома мурахами. Кожна мураха починає власний рух випадковим способом, коли покидає мурашник у пошуках їжі. Вважають, що кожен агент не рухається певним відомим шляхом або заздалегідь відомим

напрямок. 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 є сигналами для інших мурах і вказують на те, що шлях уже був досліджений та використований. 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 величиною до довжини відповідної ділянки. Чим коротший шлях у процесі пошуку знайшла мураха, тим більше феромона буде відкладено на відповідних етапах маршруту (ребрах графа).

Потрібно зауважити, що використання лише додатного оберненого зв'язку призводить до швидкої (передчасної) збіжності алгоритму, тобто до випадку, коли всі мурахи рухаються тим самим субоптимальним маршрутом. Для запобігання перенасиченню шляхів феромони із часом випаровуються, що реалізує від'ємний обернений зв'язок, що дозволяє мурахам адаптуватись до змін у навколишньому середовищі (напр., поява нових джерел їжі або завад на шляхах). Випаровування феромонів забезпечує динамічне оновлення інформації та уникнення вибору неоптимальних шляхів (уникнення локальних оптимумів).

З математичного погляду модель ASO описують через ті 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 цього дослідження є адаптація алгоритму ASO для розв'язання задачі комівояжера з нечітко заданими параметрами переміщень на етапах транспортної мережі на основі використання нечітких трапецієподібних чисел, реалізації компонентів самоорганізаційної поведінки мурах для оптимізації маршруту комівояжера та проведення числових експериментів для визначення ефективності розробленого методу.

Об'єктом дослідження цієї роботи є процес опису та розв'язування нечіткої задачі комівояжера з урахуванням невизначеності вхідних параметрів, отриманих на основі суб'єктивних оцінок користувача.

Предметом дослідження є інтелектуальні методи й алгоритми для розв'язання нечіткої задачі оптимізації на основі використання методики групової поведінки мурашиної колонії.

Новизна результатів. У дослідженні розглянуто та проаналізовано застосування алгоритму оптимізації колонії мурах для розв'язання задачі комівояжера з нечітко заданою тривалістю переміщень на транспортно-логістичній мережі. Запропоновано та реалізовано алгоритм, застосування якого за умови невеликої кількості ітерацій дозволяє отримати наближені до оптимального маршруту результати.

Постановка задачі. Стандартна постановка задачі комівояжера полягає у виборі найкоротшого за довжиною або часом замкнутого шляху на мережі з n міст, що проходить через усі міста рівно один раз. Кількість можливих варіантів дорівнює $(n - 1)!$, а за умов симетричності етапів маршруту кількість унікальних маршрутів становить $\frac{(n-1)!}{2}$.

Завдання полягає у знаходженні оптимального шляху комівояжера, визначення якого потребує надзвичайно великої кількості обчислювальних ресурсів, що приводить 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. Один із підходів, що використовується у цьому випадку, полягає у залученні нечітких чисел і реалізації засобів маніпуляції з ними.

Постановка нечіткої задачі комівояжера. Розглянемо постановку задачі комівояжера з нечітко заданою тривалістю переміщень на етапах транспортно-логістичної мережі. У цьому 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 з урахуванням деякого 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, а можливі шляхи переміщень між містами визначаються матрицею X , за умови виконання обмежень:

$$\begin{aligned} \sum_{i=1}^n x_{ij} &= 1 \text{ для всіх } j = 1, 2, \dots, n, \\ \sum_{j=1}^n x_{ij} &= 1 \text{ для всіх } i = 1, 2, \dots, n, \\ i - j + nx_{ij} &\leq n - 1, 1 \leq i \neq j \leq n, \\ x_{ij} &= 0 \text{ або } 1 \text{ для всіх } i, j = 1, 2, \dots, n. \end{aligned} \quad (2)$$

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 $i, j = \overline{1, n}$, будемо задавати у вигляді трапецієподібних нечітких чисел.

Означення. Нечітким трапецієподібним числом \tilde{A} (Dubois, & Prade, 1987) 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 гаусівського розподілу з відповідними характеристиками, то в узагальненому випадку трапецієподібне нечітке число можна представити дещо в іншому вигляді

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

де використовується середня точка $m = \frac{a_2+a_3}{2}$ та напівширина 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 поданий вище опис. За середню точку беруть звичайне середнє арифметичне значення границь плато, лівий і правий розподіли розглядають відповідно до правила решітки, 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 :

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

Остаточнo маємо

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

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

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

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

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:

- $\tilde{A} > \tilde{B}$ тоді і лише тоді, якщо $\mathfrak{R}(\tilde{A}) > \mathfrak{R}(\tilde{B})$;
- $\tilde{A} < \tilde{B}$ тоді і лише тоді, якщо $\mathfrak{R}(\tilde{A}) < \mathfrak{R}(\tilde{B})$;
- $\tilde{A} \approx \tilde{B}$ тоді й 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 управління 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 центроїда, 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, $i = \overline{1, n}$.

Алгоритм пошуку маршруту комівояжера в нечіткій задачі.

Для формування шляхів у моделі оптимізації ASO розглянемо граф $G = (V, E)$, в якому V – це множина з m вершин, а E – множина ребер. Кожному ребру $(i, j) \in E$ присвоюють два ключові параметри: час переміщення T_{ij} по ребру (i, j) (зазвичай пропорційний довжині шляху D_{ij}) та інтенсивність феромона τ_{ij} на ребрі (i, j) , $i, j = m$.

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

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

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

Випарування феромонів зменшує інтенсивність феромона на всіх ребрах, що дозволяє забувати попередні (можливо неоптимальні) шляхи, адаптуватись до змін і запобігати 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 імовірністю $P_{ij}^k(s)$, яка залежить від інтенсивності феромона і видимості η_{ij} , яка є обернено пропорційною довжині шляху $\eta_{ij} = 1/D_{ij}$:

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

де α та β – це 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 класичному випадку 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 із множини J_i^k на ітерації s дорівнює 1:

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

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 й заходити 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 мурасі k , що перебуває у місті i . Зрозуміло, що об'єднання переліків A_i^k та J_i^k дає множину всіх міст, вказаних у задачі комівояжера.

Опис алгоритму. На першій ітерації $s = 0$ кількість феромонів на кожному шляху дорівнює $\tau_{ij}(0) = 0$. Генерується набір мурах у кожній вершині графа.

У процесі ітерації кожна мураха діє окремо. В першому місті i табу-список A_i^k для мурахи k складається з міста, в якому перебуває мураха $A_i^k = \{i\}$. Далі кожна мураха обирає з певною імовірністю наступне місто для переміщення, з огляду на задану часову тривалість пересування до найближчих міст за формулою (1). Для цього використовують генератор випадкових чисел.

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

Після завершення ітерації розраховують випаровування феромона для кожного можливого ребра. Далі для кожної мурахи з успішно завершеним маршрутом після виконання ітерації s розраховується тривалість маршруту і для кожного ребра, використаного в маршруті, додається 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 (Yang et al., 2024).

Отже, за умови коректно обраних значень параметрів α та β з кожною ітерацією поступово отримуємо покращені результати.

Результати обчислень. Наведемо результати проведених числових експериментів. На попередньому етапі алгоритм АСО був адаптований до розв'язання задачі комівояжера з нечіткою тривалістю переміщень між містами. Для цього проведено фазифікацію задачі, наведеної в роботі (Ivohin, Gavrylenko, & Ivohina, 2023) за допомогою трапецієподібних нечітких чисел (рис. 1).

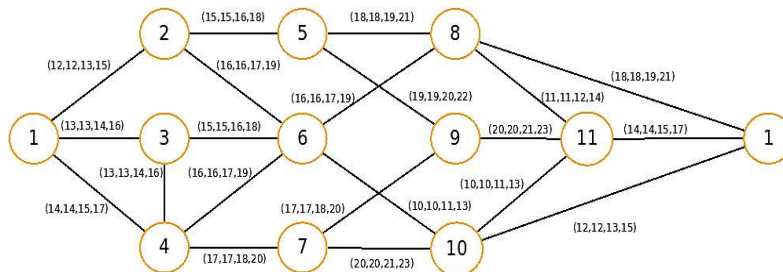


Рис. 1. Схема-приклад задачі комівояжера з нечіткою тривалістю

Ранжування маршрутів із нечіткою тривалістю виконувалось за допомогою методу COG для неперервного способу подання нечітких чисел (6).

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

Таблиця 1

Параметри ASO для розв'язання задачі комівояжера

α	β	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 колонці "Ітерація" вказано першу ітерацію, на якій отримано найкращий розв'язок із найбільшою імовірністю.

Оптимальна тривалість маршруту в задачі комівояжера на основі повного перебору визначається нечітким трапецієподібним числом (156, 156, 167, 189). Застосовуючи мурашиний алгоритм у

задачі отримано аналогічний розв'язок, при чому 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 проводились для оцінювання якості отриманого результату з урахуванням різної кількості міст транспортної мережі. У процесі числових розрахунків генетичний алгоритм використовувався для розв'язання задачі комівояжера у випадковому розміщенні $n = 16, 17, 18$ міст на двовимірній площині 200×200 , для якого величини переміщення між містами мережі визначались за середньою тривалістю нечіткого часу, яка вважалась пропорційною відстані між відповідними містами.

Проведено такі експерименти, коли кількість ітерацій дорівнює: чотирикратній кількості міст (табл. 2), подвійній кількості міст (табл. 3), кількості міст (табл. 4). Кількість мурах дорівнює кількості міст у кожному випадку.

Таблиця 2

Кількість ітерацій дорівнює чотирикратній кількості міст

V_{var}	α	β	Q	Кількість міст, якість		
				16	17	18
0.05	0.15	-1	20	103.72 %	104.31 %	104.0 %
0.05	0.15	-1.5	20	103.56 %	103.81 %	104.28 %
0.05	0.15	-2	20	103.67 %	103.73 %	104.19 %
0.05	0.15	-2.5	20	103.75 %	103.93 %	103.97 %
0.05	0.15	-3	20	103.68 %	104.18 %	104.27 %
0.05	0.15	-3.5	20	103.97 %	104.21 %	104.30 %
0.05	0.15	-4	20	103.88 %	104.11 %	104.33%
0.05	0.1	-3	20	103.77 %	104.04 %	104.64 %
0.05	0.15	-3	20	103.68 %	103.78 %	103.87 %
0.05	0.2	-3	20	103.66 %	103.71 %	103.98 %
0.05	0.25	-3	20	104.02 %	104.10 %	104.19 %
0.05	0.3	-3	20	104.08 %	104.11 %	104.32 %

Таблиця 3

Кількість ітерацій дорівнює подвійній кількості міст

V_{var}	α	β	Q	Кількість міст, якість		
				16	17	18
0.05	0.15	-1	20	104.04 %	104.50 %	104.74 %
0.05	0.15	-1.5	20	103.94 %	104.48 %	104.78 %
0.05	0.15	-2	20	103.98 %	104.23 %	104.81 %
0.05	0.15	-2.5	20	103.99 %	104.17 %	104.26 %
0.05	0.15	-3	20	103.97 %	103.99 %	104.56 %
0.05	0.15	-3.5	20	104.10 %	104.26 %	104.38 %
0.05	0.15	-4	20	103.86 %	104.36 %	104.55 %
0.05	0.1	-3	20	103.88 %	104.11 %	104.58 %
0.05	0.15	-3	20	103.87 %	103.91 %	104.56 %
0.05	0.2	-3	20	104.01 %	104.40 %	104.47 %
0.05	0.25	-3	20	104.12 %	104.28 %	104.83 %
0.05	0.3	-3	20	104.34 %	104.44 %	104.72 %

Таблиця 4

Кількість ітерацій дорівнює кількості міст

V_{var}	α	β	Q	Кількість міст, якість		
				16	17	18
0.05	0.15	-1	20	104.75 %	104.92 %	104.96 %
0.05	0.15	-1.5	20	104.61 %	105.06 %	105.20 %
0.05	0.15	-2	20	104.42 %	104.45 %	104.69 %
0.05	0.15	-2.5	20	104.35 %	104.69 %	105.11 %
0.05	0.15	-3	20	104.54 %	104.57 %	104.75 %
0.05	0.15	-3.5	20	104.38 %	104.39 %	104.71 %
0.05	0.15	-4	20	104.11 %	105.12 %	105.15 %
0.05	0.15	-2.5	20	104.35 %	104.69 %	105.11 %
0.05	0.15	-3	20	104.54 %	104.57 %	104.75 %
0.05	0.15	-3.5	20	104.38 %	104.39 %	104.71 %
0.05	0.15	-4	20	104.11 %	105.12 %	105.15 %
0.05	0.1	-3	20	104.37 %	104.50 %	105.13 %
0.05	0.15	-3	20	104.34 %	104.47 %	104.75 %

Закінчення табл. 4

V_{var}	α	β	Q	Кількість міст, якість		
				16	17	18
0.05	0.2	-3	20	104.31 %	104.77 %	104.83 %
0.05	0.25	-3	20	104.45 %	104.70 %	104.83 %
0.05	0.3	-3	20	104.47 %	104.96 %	105.05 %

Для кожного експерименту проведено оцінювання впливу зміни параметрів β та α на остаточний результат. Порівняння найкращого результату, отриманого за допомогою алгоритму ASO за задану кількість ітерацій, з найкращим результатом, отриманим методом повного перебору, 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 перебору.

Як можна бачити з наведених результатів, за умови здійснення кількості ітерацій, що дорівнює кількості міст, алгоритм дозволяє отримати результати, що в межах 5 % від найкращого. Збільшення кількості ітерацій веде до покращення результатів. Варіація параметра β в межах від -4 до -1 та параметра α в межах від 0.1 до 0.3 якісно впливають на результати, які в цьому випадку розміщені в межах 0.5 % від найкращого результату розв'язування задачі комівояжера, який отримано методом повного перебору.

Дискусія і висновки

У роботі розглянуто та проаналізовано застосування алгоритму оптимізація 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. Приклади використання алгоритму підтверджують конструктивність підходу для розв'язання проблеми комівояжера у випадку задання нечіткої тривалості переміщень.

Список використаних джерел

- Bouamama, S., Blum, C., & Fages, J. (2019). An algorithm based on ant colony optimization for the minimum connected dominating set problem. *Applied Soft Computing*, 80, 672–686. <https://doi.org/10.1016/j.asoc.2019.04.032>
- Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). Solution of a large-scale traveling salesman problem. *Operations Research*, 2, 393–410. <https://doi.org/10.1287/opre.2.4.393>
- Dorigo, M. (1992). *Optimization, learning and natural algorithms* [Doctoral dissertation]. Politecnico di Milano.
- Dorigo, M. (2004). *Ant colony optimization*. MIT Press.
- Dorigo, M., & Di Caro, G. A. (1999). Ant algorithms for discrete optimization. *Artificial Life*, 5(2), 137–172. <https://doi.org/10.1162/106454699568728>
- Dorigo, M., & Di Caro, G. A. (1998). AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9, 317–365. <https://doi.org/10.1613/jair.525>
- Dubois, D., & Prade, H. (1987). The mean value of a fuzzy number. *Fuzzy Sets and Systems*, 24(3), 279–300. [https://doi.org/10.1016/0165-0114\(87\)90189-9](https://doi.org/10.1016/0165-0114(87)90189-9)
- 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. *Radio Electronics, Computer Science, Control*, 3, 141–147. <https://doi.org/10.15588/1607-3274-2023-3-16>
- Karaboga, D. (2009). A new design method based on artificial bee colony algorithm for digital IIR filters. *Journal of the Franklin Institute*, 346(4), 328–348. <https://doi.org/10.1016/j.jfranklin.2008.11.003>
- Kennedy, J., & Eberhart, R. C. (2001). *Swarm intelligence*. Morgan Kaufmann.
- Stützle, T., & Hoos, H. (2000). MAX–MIN ant system. *Future Generation Computer Systems*, 16(8), 889–914. [https://doi.org/10.1016/S0167-739X\(00\)00043-1](https://doi.org/10.1016/S0167-739X(00)00043-1)
- Yang, L., Wang, X., He, Z., Wang, S., & Lin, J. (2024). Review of traveling salesman problem solution methods. In *Bio-Inspired Computing: Theories and Applications* (pp. 3–16). Communications in Computer and Information Science. https://doi.org/10.1007/978-3-031-50887-5_1

References

- Bouamama, S., Blum, C., & Fages, J. (2019). An algorithm based on ant colony optimization for the minimum connected dominating set problem. *Applied Soft Computing*, 80, 672–686. <https://doi.org/10.1016/j.asoc.2019.04.032>
- Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). Solution of a large-scale traveling salesman problem. *Operations Research*, 2, 393–410. <https://doi.org/10.1287/opre.2.4.393>
- Dorigo, M. (1992). *Optimization, learning and natural algorithms* [Doctoral dissertation]. Politecnico di Milano.
- Dorigo, M. (2004). *Ant colony optimization*. MIT Press.
- Dorigo, M., & Di Caro, G. A. (1999). Ant algorithms for discrete optimization. *Artificial Life*, 5(2), 137–172. <https://doi.org/10.1162/106454699568728>
- Dorigo, M., & Di Caro, G. A. (1998). AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9, 317–365. <https://doi.org/10.1613/jair.525>

Dubois, D., & Prade, H. (1987). The mean value of a fuzzy number. *Fuzzy Sets and Systems*, 24(3), 279–300. [https://doi.org/10.1016/0165-0114\(87\)90189-9](https://doi.org/10.1016/0165-0114(87)90189-9)

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. *Radio Electronics, Computer Science, Control*, 3, 141–147. <https://doi.org/10.15588/1607-3274-2023-3-16>

Karaboga, D. (2009). A new design method based on artificial bee colony algorithm for digital IIR filters. *Journal of the Franklin Institute*, 346(4), 328–348. <https://doi.org/10.1016/j.jfranklin.2008.11.003>

Kennedy, J., & Eberhart, R. C. (2001). *Swarm intelligence*. Morgan Kaufmann.

Stützle, T., & Hoos, H. (2000). MAX–MIN ant system. *Future Generation Computer Systems*, 16(8), 889–914. [https://doi.org/10.1016/S0167-739X\(00\)00043-1](https://doi.org/10.1016/S0167-739X(00)00043-1)

Yang, L., Wang, X., He, Z., Wang, S., & Lin, J. (2024). Review of traveling salesman problem solution methods. In *Bio-Inspired Computing: Theories and Applications* (pp. 3–16). Communications in Computer and Information Science. https://doi.org/10.1007/978-3-031-50887-5_1

Отримано редакцією журналу / Received: 17.09.24

Прорецензовано / Revised: 27.09.24

Схвалено до друку / Accepted: 01.10.24

Eugene IVOHIN¹, DSc (Phys. & Math.), Prof.
ORCID ID: 0000-0002-5826-7408
e-mail: ivohin@univ.kiev.ua

Kostantin YUSHTIN¹, PhD (Phys. & Math.), Assoc. Prof.
ORCID ID: 0009-0001-9881-2343
e-mail: gkons@univ.kiev.ua

Valeriy HAVRILENKO², DSc (Phys. & Math.), Prof.
ORCID ID: 0000-0001-9682-4204
e-mail: vvgavrilenko1953@gmail.com

Maxym BOHUSLAVSKYI², PhD Student
ORCID ID: 0009-0000-1264-133X
e-mail: maxbogus09@gmail.com

¹Taras Shevchenko National University of Kyiv, Ukraine

²National transport university, Kyiv, Ukraine

USING THE ANT COLONY ALGORITHM TO SOLVE THE FUZZY TRAVELLING SALESMAN PROBLEM

The traveling salesman problem (TSP) is a classical combinatorial optimization problem that involves finding the shortest or fastest route among a set of cities. To formalize the uncertainty and imprecision in input data, often caused by subjective evaluations of the travel time intervals, this paper employs fuzzy numbers. The form of these fuzzy numbers is based on a Gaussian-like approach. This work examines the specifics of applying the

ant colony optimization (ACO) algorithm and proposes an approach for its optimal use. The impact of the algorithm's parameters on the quality of the approximated best solution is analyzed. The problem is illustrated with numerical examples involving a sufficiently large number of cities in the transportation 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.

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.