

МОДИФІКАЦІЯ МЕТОДУ МУРАШИНОЇ КОЛОНІЇ ДЛЯ РОЗВ’ЯЗАННЯ ЗАДАЧІ КОМІВОЯЖЕРА КОЛЕКТИВОМ АВТОНОМНИХ АГЕНТІВ

© Голембо В.А., Муляревич О.В., 2012

Розглянуто можливість використання колективу автономних агентів для розв’язання задач комбінаторної оптимізації на прикладі розв’язання задачі комівояжера.

Ключові слова: автономні агенти, задача комівояжера, метод мурашиної колонії, вдосконалення.

This paper is devoted to the solving one of the Combinatorial optimization task – the Travelling Salesman Problem (TSP) by using collective autonomous agents.

Key words: autonomous agents, TSP, ant colony method, improvements.

Вступ

У роботі розглядаються нові підходи до розв’язання задач комбінаторної оптимізації колективом автономних агентів на базі методів “ройового інтелекту” на прикладі задачі комівояжера (ЗК) [1]. ЗК полягає у тому, що комівояжеру потрібно відвідати кожне з n пунктів, вершин графа. Для кожної пари пунктів i та j згідно з маршрутом слідування встановлена певна вартість (наприклад відстань, час, ціна квитка) $L(i,j)$, де L – матриця вартостей, розмірністю $n \times n$. Якщо для матриці вартостей виконується умова: $L(i,j) = L(j,i)$, ЗК є симетричною, інакше – асиметричною. Потрібно знайти шлях мінімальної вартості, який починається з деякого початкового міста, забезпечує однократне відвідування усіх пунктів, що залишилися, та повернення до пункту відправлення, тобто знайти цикл Гамільтона у графі [7]. Сьогодні ЗК на 85900 пунктів є найбільшою симетричною задачею комівояжера, що була розв’язана. Для знаходження оптимального маршруту цієї ЗК знадобилось майже три роки обчислень на великій мережі комп’ютерів [11].

Огляд існуючих методів розв’язання задачі комівояжера

Методи розв’язання ЗК поділяють на: 1) точні – алгоритми знаходження оптимального маршруту найменшої можливої вартості за заданих умов ЗК; 2) неточні (евристичні) – алгоритми, що забезпечують знаходження результуючого маршруту наближеного до оптимального, тобто квазі-оптимального.

Під час досліджень було проведено аналіз деяких методів розв’язання ЗК: алгоритм повного перебору, жадібний алгоритм, дерев’яний алгоритм, алгоритм Дейкстри, метод гілок і меж [10], надалі називатимемо їх класичними. З проведеного аналізу було виявлено такі недоліки класичних методів розв’язання ЗК:

1) розв’язання ЗК лише з статичною вхідною мережею пунктів. Тобто за умови динамічних змін, що відбуваються з значеннями матриці вартостей: зникнення чи поява нових пунктів або недоступність певних ребер на момент обчислення, ще не отриманий результат стає мало-ефективним, а при великій динаміці змін – безкорисним;

2) слабка або неефективна здатність до розпаралелення виконання алгоритму, через залежність по даних;

3) неспроможність більшості алгоритмів знайти припустимо точні розв’язки ЗК за допустимий на практиці час для кількості вхідних пунктів $N > 100$.

Усі ці особливості розглянутих алгоритмів роблять їх використання на практиці для розв'язання ЗК з великою кількістю пунктів неефективним. З метою збільшення кількості пунктів обходу ведеться пошук нових та вдосконалення існуючих методів розв'язання ЗК у двох перспективних напрямках: 1) методи локальної оптимізації [2], що включають методи кластеризації та декомпозиції ЗК; 2) методи розв'язання ЗК за допомогою колективу агентів (еволюційні та алгоритми соціальних комах).

Перший напрямок дозволяє позбутись двох останніх недоліків класичних методів розв'язання ЗК, проте методи локальної оптимізації також є малоефективними для розв'язання ЗК за умови динамічних змін початкових умов ЗК. Такий самий недолік мають методи розв'язання ЗК колективом автономних агентів на базі еволюційних алгоритмів. Позбутись цього недоліку дозволяють менш досліджені методи, що були запозичені з біології – алгоритми соціальних комах, які відносяться до методів ройового інтелекту (swarm intelligence) [3]. Більшість цих методів має лише теоретичний опис алгоритмів поведінки соціальних комах та варіанти їх практичного застосування. Для задач комбінаторної оптимізації найбільший інтерес становить розвиток методу мурашиної колонії (АСО – ant colony optimization), запропонований М. Доріго [4,5]. Він полягає у використанні сукупності автономних мобільних [9] мурах-агентів, які вирішують проблеми пошуку шляхів за допомогою хімічної регуляції – системи міток, без існуючої комунікації між мурахами-агентами, тобто без прямого обміну повідомленнями між мурахами. Кожна мураха-агент залишає за собою на землі доріжку особливих речовин – феромонів, що моделюються за допомогою відповідних числових міток на ребрах графа. Інша мураха-агент, відчувши слід на землі, рухається по ньому. Чим більше по одному і тому самому шляху пройшло мурах – тим чіткіше слід, а чим чіткіше слід – тим більше “бажання” піти в ту ж сторону виникає у інших мурах-агентів.

Оскільки мурахи, що знайшли найкоротший шлях до “об’єкта їжі”, витрачають менше часу на шлях туди і назад, їхній слід швидко стає найпомітнішим. Він привертає увагу все більшої кількості інших мурах, які формують шлях, по якому рухаються мурахи-агенти. Інші шляхи, які менш використовуються, з часом зникають, через випаровування феромону.

Метод мурашиної колонії належить до конструктивних методів розв'язання ЗК, тобто методів, в яких квазі-оптимальний шлях знаходиться поступово, з можливістю отримати на кожному етапі частину остаточного розв'язку.

Розроблення узагальненої граф-схеми розв'язку ЗК методом мурашиної колонії

З метою чіткішого розуміння принципу розв'язання ЗК методом мурашиної колонії було побудовано узагальнену граф-схему розв'язку ЗК цим методом (рис. 1). Метод мурашиної колонії вимагає використання сукупності початкових (вхідних) параметрів, таких як: 1) кількість мурах-агентів (k); 2) параметри процесів “нанесення феромону” та “випаровування феромону” (α , β); 3) кількість ітерацій циклу “запуску мурах”, яка необхідна для розв'язання ЗК.

Розглянемо основні процедури узагальненого розв'язку ЗК базовим методом мурашиної колонії, представлені в граф-схемі (рис. 1):

Процедура D (decision). Якщо мураха ще не закінчила шлях, тобто не відвідала усі вузли графа мережі, то для визначення наступного ребра шляху використовується імовірнісне рівняння:

$$P = \frac{\tau(r,u)^\alpha \times \eta(r,u)^\beta}{\sum_k \tau(r,u)^\alpha \times \eta(r,u)^\beta}, \quad (1)$$

де $\tau(r, u)$ – значення числової мітки (інтенсивність феромону) на ребрі між вузлами r та u , $\eta(r, u)$ – функція, яка відображає вимір зворотної відстані для грані, α – вага ферменту, β – коефіцієнт евристики.

Для відображення кількості феромону, який був залишений на кожному ребрі шляху для k -ї мурахи-агента, використовується рівняння:

$$\Delta \tau_{ij}^k(t) = \frac{Q}{L^k(t)}, \quad (2)$$

де Q – константа; $L^k(t)$ – вартість ребра (ціна, відстань, час) пройденого k -ю мурахою.

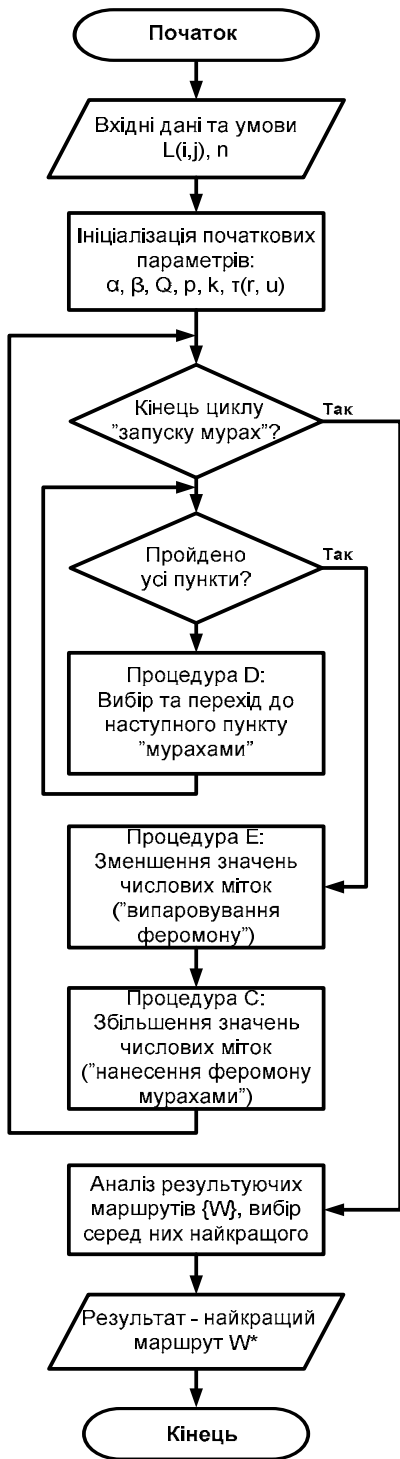


Рис. 1. Граф-схема розв'язку ЗК базовим методом "мурашиної колонії"

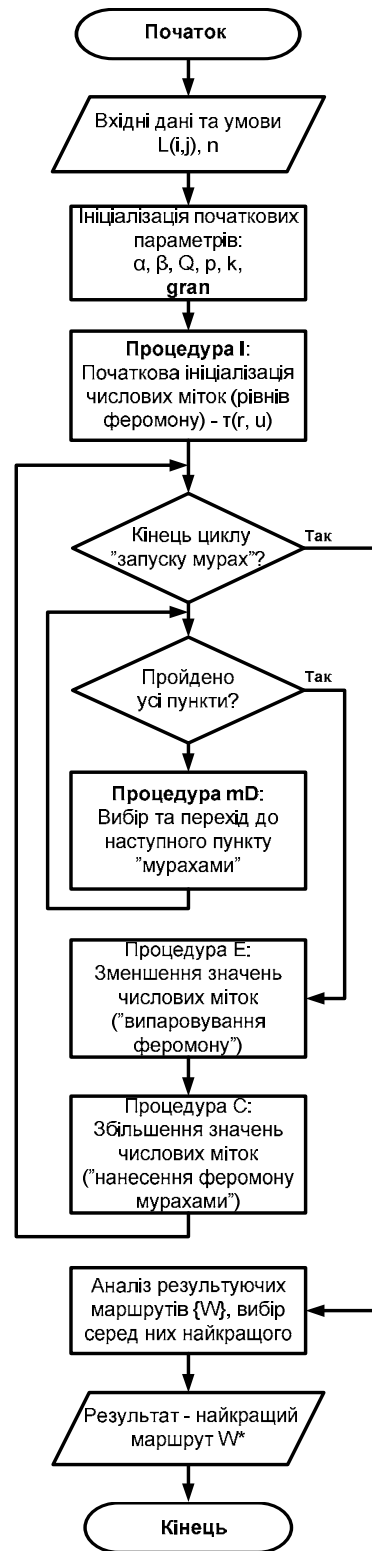


Рис. 2. Граф-схема розв'язку ЗК модифікованим методом "мурашиної колонії"

Процедура E (evaporation). Щоб поступово видалити ребра, які входять в гірші шляхи графа, до усіх ребер згідно з алгоритмом застосовується процедура зменшення значень числових міток на ребрах графа ("випаровування феромону"), що описується рівнянням:

$$\tau_{ij}(t+1) = \tau_{ij}(t) \times (1 - p), \quad (3)$$

де p – константа, що визначає інтенсивність зменшення (випаровування).

Процедура С (covering). Процедура збільшення значень числових міток на ребрах графа (“нанесення феромону”) згідно з алгоритмом відбувається в кінці ітерації циклу “запуску мурах”, після процедури E, за таким рівнянням:

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \sum_k \Delta\tau_{ij}(t). \quad (4)$$

Після завершення циклу “запуску мурах” відбувається аналіз результуючих k маршрутів {W} та вибір серед них найкращого маршруту W*, тобто, що має найменшу сумарну вартість.

Нові пропозиції до розв'язку ЗК у межах методу мурашиної колонії

Для базового методу мурашиної колонії необхідна певна кількість ітерацій циклу запуску мурах, однак кожна ітерація потребує певного часу виконання.

Для зменшення часу обчислення результуючого маршруту ЗК пропонується новий спосіб початкової ініціалізації числових міток на ребрах графа (рівнів феромону) $\tau_0(x, u)$ (Процедура I, рис. 2) та відповідно новий спосіб вибору та переходу до наступного пункту мурахами-агентами (Процедура mD, рис. 2), а також введений додатковий параметр gran.

Процедура I (initialization). У класичному методі “мурашиної колонії” числові мітки (рівні феромону) на ребрах графа ініціалізуються ненульовими однаковими значеннями, це призводить до рівноімовірнісного обирання будь-якого ребра графа мурахами-агентами на першій ітерації циклу “запуску мурах”.

Пропонується нерівномірна початкова ініціалізація числових міток (рівнів феромону), що відбувається згідно з доступністю ребра графа та його вартості. Запропонований підхід дозволив отримати такі переваги:

1. Фактичний “пробіг одразу по всіх ребрах” мурахами-агентами, згідно з чим аналізуються усі ребра, жодне з яких не залишиться без уваги, крім того одразу виявляються найменш придатні ребра для прокладання маршруту;

2. Від самого старту алгоритму агенти орієнтуються у виборі шляху не на просту випадковість, а на вже сформований початковий досвід – ініціалізовану “пам'ять мурашника”, що дає змогу значно пришвидшити алгоритм знаходження оптимального шляху;

3. Зменшення кількості “запусків мурах”, якої необхідно для визначення оптимального шляху, та збільшення ймовірності знаходження оптимальнішого розв'язку за рахунок іншого параметра – кількості мурах-агентів;

4. Отримання прийняттого результату лише за одну ітерацію циклу “запуску мурах”.

Процедура mD (modified decision). Новий підхід до початкової ініціалізації числових міток на ребрах графа призводить до збільшення важливості критерію вартості ребра при виборі мурахою-агентом наступного пункту свого шляху.

Пропонується модифікувати систему визначення мурахою-агентом наступного пункту методом введення певної граничної ймовірності – gran, як додаткового початкового параметра, що буде визначати область для випадкового обирання мурахою-агентом ребер графа, ймовірність вибору яких більша за граничну.

Запропонований спосіб вибору ребра та переходу до наступного пункту “мурахами” зменшив ймовірність зведення методу “мурашиної колонії” до методу “найближчого сусіда” [7], який дає результати, за точністю не прийнятні на практиці. Розроблений підхід забезпечує гнучкість методу “мурашиної колонії”, постійний розгляд можливих оптимальніших шляхів від вже існуючого знайденого квазі-оптимального маршруту.

Моделювання колективної поведінки агентів для розв'язку ЗК

Для перевірки ефективності розроблених підходів до розв'язку ЗК, було розроблено систему моделювання колективної поведінки агентів. Для зменшення часу виконання обчислень результуючого шляху було вирішено розробляти систему моделювання з використанням технології паралельних та розподілених обчислень MPI (message passing interface) [8]. Програмна система моделювання складається з головного синхронізуючого процесу (master), який виконує функцію інтерфейсу та вмістилища даних, та підлеглих процесів (slaves), які виконують функцію мурах-агентів згідно з методом мурашиної колонії.

Розроблена система моделювання розрахована на роботу в комп'ютерній системі, що складається з сукупності обчислювальних вузлів, але дослідження для зручності та можливості порівняння отриманих результатів з результатами інших методів проводились на одному персональному одноядерному комп'ютері (CPU: Celeron 2,93 ГГц, 2 ГБ ОЗУ, ОС – Windows XP SP2). На графіку (рис. 3, а) наведено залежність часу виконання обчислень від кількості пунктів ЗК при кількості мурах-агентів – 2. Під час проведених досліджень було отримано результуючі маршрути для кількості пунктів обходу 100, 500 та 1000. Найбільш вражаючий результат, який було досягнуто – розв'язок ЗК на 10000 пунктів. На графіку (рис. 3, б) наведено залежність часу виконання обчислень від кількості мурах-агентів (за кількості пунктів $n = 100$), кількість мурах-агентів розглядалась від 1 до 100.

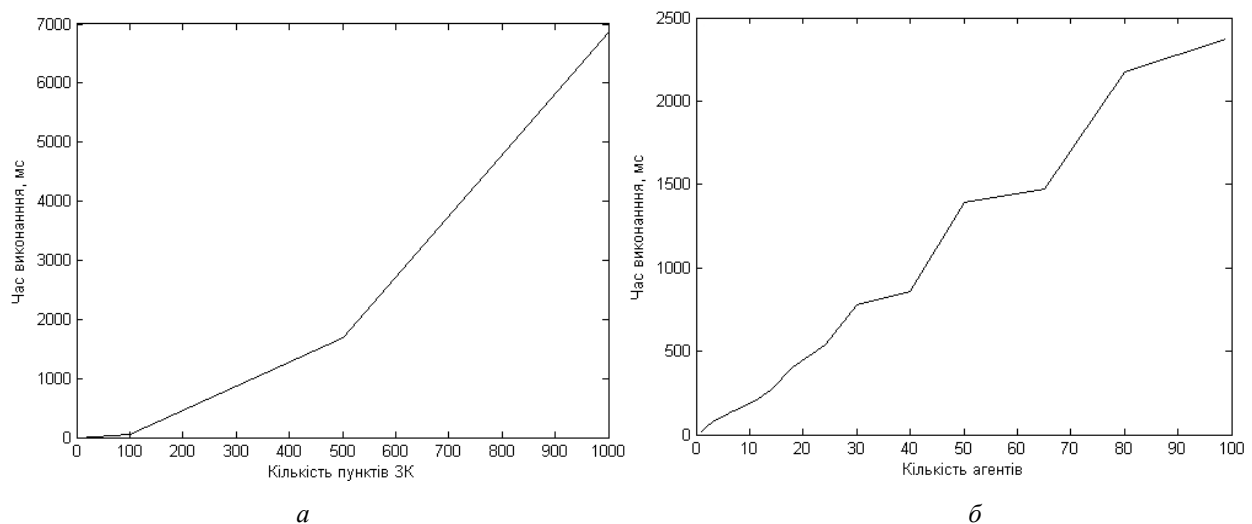


Рис. 3. Графіки залежностей часу виконання обчислень:
а – від кількості пунктів ЗК; б – від кількості агентів ЗК

На рис.4 зображено початкове випадкове розміщення пунктів на площині (рис. 4, а) та розв'язок ЗК (результуючий маршрут W^*) на 1000 пунктів відвідування (рис. 4, б). Кружечками позначені пункти ЗК, які відвідує комівояжер. На рисунках колом обведено область знаходження початкового, з якого комівояжер починає свій рух, та кінцевого пункту маршруту; ці пункти збігаються.

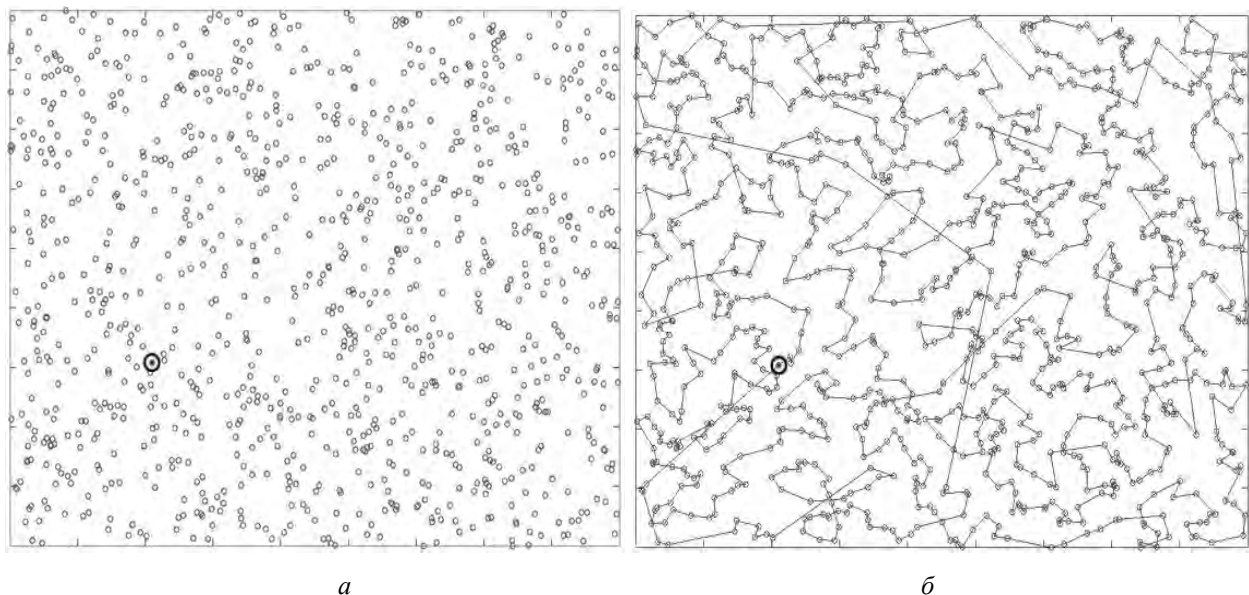


Рис. 4. Задача комівояжера на 1000 пунктів обходу:
а – початкове розміщення вершин $\{V\}$ (пунктів), $n=1000$;
б – розв'язок ЗК – маршрут W^* для $n=1000$

На рис. 5 і 6 аналогічними позначеннями зображено початкове випадкове розміщення пунктів на площині та розв'язок ЗК на 10000 пунктів відвідування. На обчислення результуючого маршруту для ЗК на 10000 пунктів знадобилося менше двох годин.

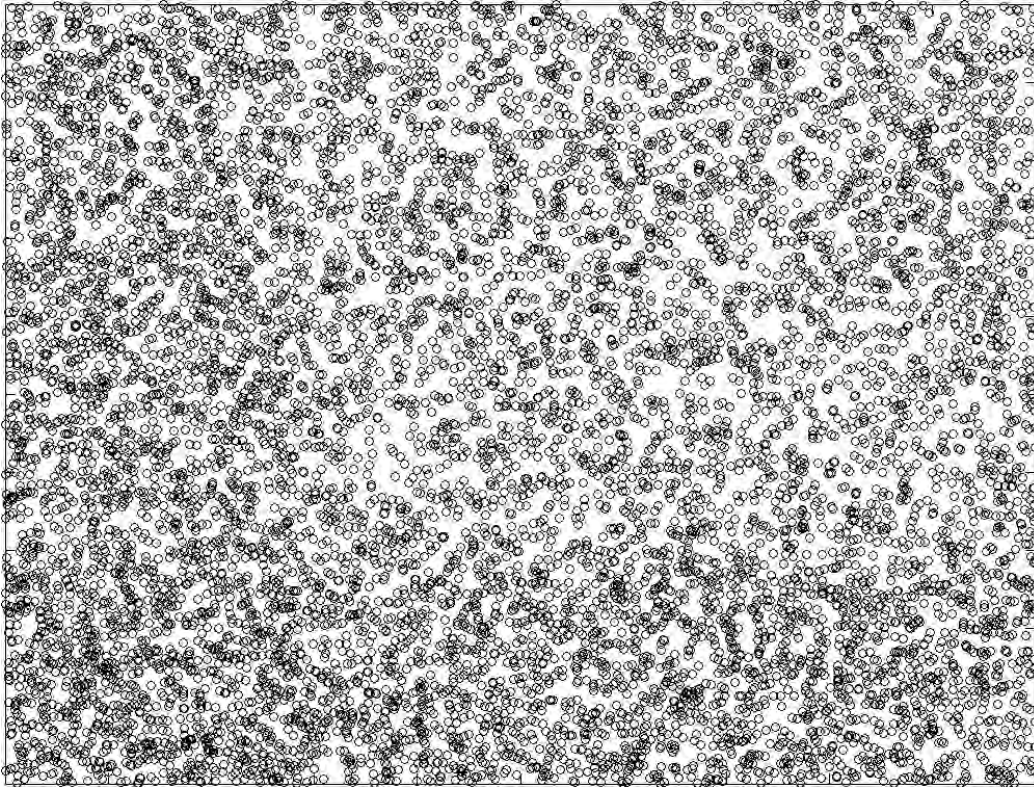


Рис. 5. Початкове розміщення вершин $\{V\}$ (пунктів), $n=10000$

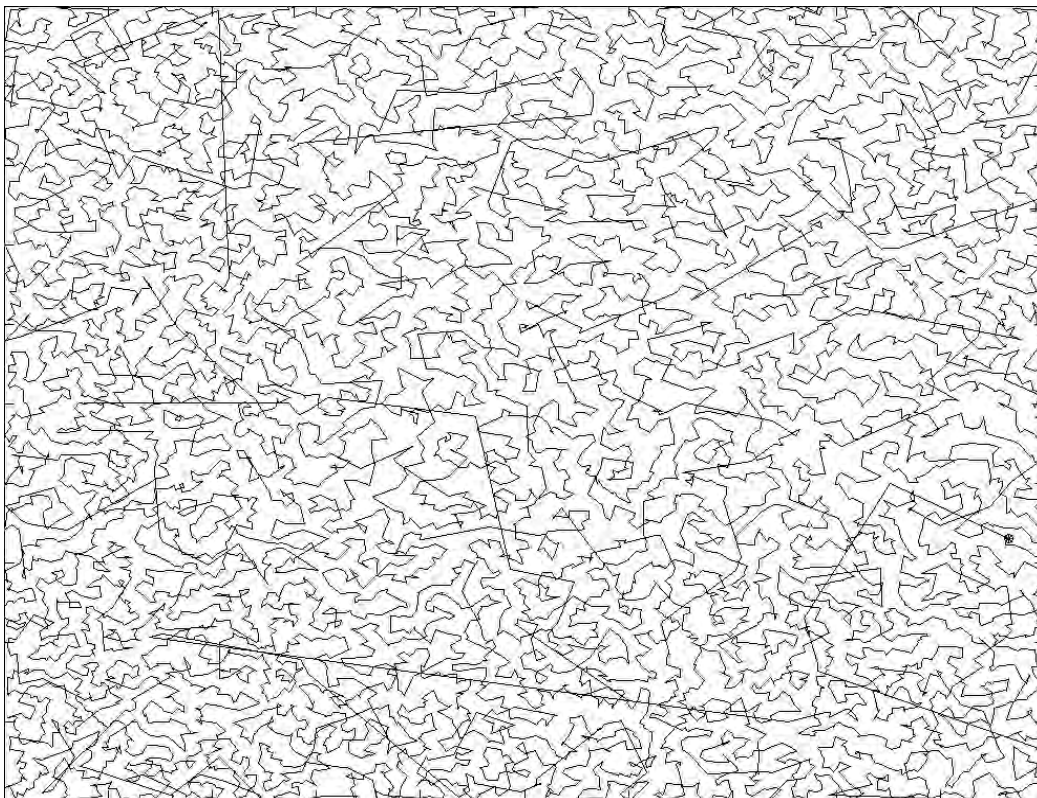


Рис. 6. Розв'язок ЗК – маршрут W^ для $n=10000$*

Отримані результати на 1000 та 10000 пунктів є квазі-оптимальними, тобто існує висока ймовірність знаходження аналогічних маршрутів, наближених до оптимального. Для знаходження оптимального маршруту необхідно переглянути усі можливі, для ЗК на 1000 пунктів їх кількість буде 1000!, а для ЗК на 10000 пунктів обходу – 10000!.

Слід зазначити, що за 100 запусків системи моделювання для розв'язання ЗК на 100 пунктів було отримано 76 різних маршрутів, причому ймовірність знаходження вже отриманого маршруту знижується з розмірністю ЗК. Найбільшу ймовірність знаходження при повторному запуску системи моделювання мають маршрути з низькою вартістю результуючого маршруту. Вартість більшості отриманих системою моделювання маршрутів вища від вартості обчисленого за точним методом оптимального маршруту в межах 10 – 20 %.

Розроблена система моделювання має високий потенціал практичного застосування в умовах динамічних змін значень матриці вартостей завдяки таким перевагам: 1) здатність від початку обчислень отримувати поетапно кінцевий результуючий маршрут; 2) висока швидкодія методу з врахуванням розроблених модифікацій; 3) низька розбіжність з результатами точних методів.

Висновки

Результати моделювання показали переваги запропонованих підходів до розв'язку ЗК на основі методу “мурашиної колонії”, зокрема, можливість отримувати результат лише за одну ітерацію циклу “запуску мурах”. За рахунок цього підвищуються адаптивні можливості запропонованого розв'язку в умовах динамічних змін значень матриці вартостей L . Запропоновані підходи вже дозволили розв'язати ЗК з кількістю пунктів відвідування більше 10000 за допустимий час обчислення (на відміну від більшості існуючих аналогів).

У подальшому планується розробити розв'язок ЗК для кількості пунктів більше ніж 100000 та розглянути ще одну задачу комбінаторної оптимізації – розв'язання ЗК декількома комівояжерами.

Розроблену систему моделювання доцільно застосовувати для розв'язання ЗК з динамічною мережею пунктів. Це особливо важливо для автоматизації складних процесів управління в таких сферах господарства, як транспортна, інформаційна, телекомунікаційна.

1. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. – 2-е изд. – М.: Вильямс, 2005.
2. Базилевич Р., Кутельмах Р. Дослідження ефективності існуючих алгоритмів для розв'язання задачі комівояжера. // Вісник Нац. ун-ту “Львівська політехніка”. – 2009. – № 650.
3. Bonabeau E., Dorigo M., Theraulaz G. *Swarm Intelligence*. – Oxford University Press, 1999.
4. Dorigo M., *Optimization, Learning, and Natural Algorithms*. – “Doctorate in Systems and Information Electronic Engineering”, Politecnico di Milano, 1992.
5. Stützle T., López-Ibáñez M., Pellegrini P., Maur M., Oca M., Birattari M., Michael Maur, Dorigo M., *Parameter Adaptation in Ant Colony Optimization* // Technical Report, IRIDIA, Université Libre de Bruxelles, 2010.
6. Вейтман В. Разговор о маршрутизации не окончен // Компьютерное обозрение, №27 (546), 18 июля 2006.
7. Касьянов В.Н., Евстигнеев В.А. Графы в программировании: обработка, визуализация и применение. – СПб.: БХВ-Петербург, 2003.
8. Оленев Н.Н. Основы параллельного программирования в системе MPI. – М.: ВЦ РАН, 2005.
9. Мельник А.О., Голембо В.А., Бочкарьов О.Ю. Нові принципи побудови вимірально-обчислювальних мереж на основі інтелектуальних агентів // Вісник Нац. Ун-ту “Львівська політехніка”. – 2003. – №492.
10. Юдин Д.Б., Гольштейн Е.Г. *Линейное программирование. Теория, методы и приложения*. – 3 изд. – М., 1969.
11. David L. Applegate, Robert E. Bixby, Vasek Chvátal & William J. Cook *The Traveling Salesman Problem: A Computational Study*. – Princeton University Press 2006.