

УДК 656.053; 519.863

# ВПЛИВ ПАРАМЕТРІВ МУРАШИНОГО АЛГОРИТМУ НА РОЗВ'ЯЗОК ЗАДАЧІ КОМІВОЯЖЕРА

**І. А. Могила**  
Асистент\*

E-mail: ihor.mohyla@gmail.com

**І. І. Лобач\***

**О. А. Якимець\***

\*Кафедра транспортних технологій  
Національний університет  
«Львівська політехніка»

вул. Бандери, 12, м. Львів, Україна, 79013

*Наведено коротку характеристику задачі комівояжера та методів її розв'язування. Докладно розкрито принципи роботи та основні залежності мурашиного алгоритму, який використовується для розв'язування цієї задачі. За результатами досліджень для мереж різних розмірів встановлено, що на ефективність роботи мурашиного алгоритму впливають керуючі параметри, та знайдено такі значення параметрів, за яких найшвидше досягається оптимальний розв'язок*

*Ключові слова: маршрутизація дрібногуртових перевезень, задача комівояжера, мурашиний алгоритм, керуючі параметри алгоритму*

*Приведена краткая характеристика задачи коммивояжера и методов ее решения. Детально раскрыты принципы работы и основные зависимости муравьиного алгоритма, который используется для решения этой задачи. По результатам исследований для сетей разных размеров установлено, что на эффективность работы муравьиного алгоритма влияют управляющие параметры, и найдены такие значения параметров, при которых наиболее быстро достигается оптимальное решение*

*Ключевые слова: маршрутизація мелкопартионных перевозок, задача коммивояжера, муравьиный алгоритм, управляющие параметры алгоритма*

## 1. Вступ

Перевезення більшості вантажів торгівлі, пошти, деяких промислових виробів з баз постачання до споживачів тощо належить до класу дрібногуртових перевезень, коли автомобіль протягом одного рейсу виконує доставку вантажів відразу декільком отримувачам (або збирає вантаж від декількох відправників на адресу одного отримувача, або водночас розвозить і збирає вантаж). При цьому, якщо вантаж з розподільчого центру доставляється до  $n$  споживачів, а їх сумарний попит не перевищує вантажності автомобіля, то вантаж всім отримувачам може бути доставлений за один рейс. У разі, якщо не накладається обмеження на тривалість роботи на маршруті, то для розв'язування задачі потрібно побудувати кільцевий маршрут з найменшою довжиною, який включає усі пункти доставки вантажу. Отже, така задача маршрутизації зводиться до задачі комівояжера [1, 2]. При цьому відомо, що використання комп'ютерних методів для оптимізації маршрутів доставки вантажів часто виражається в заощадженні 5–20 % його загальної вартості [3]. Тому актуальним є розроблення та вдосконалення методів розв'язування задачі комівояжера, які забезпечуватимуть швидкий пошук оптимального розв'язку.

## 2. Аналіз літературних джерел та формулювання проблеми

### 2.1. Формулювання задачі комівояжера та аналіз методів її розв'язування

У задачі комівояжера, маючи матрицю найкоротших відстаней між  $n$  пунктами, потрібно побудувати

найкоротший кільцевий маршрут об'їзду всіх пунктів  $\tilde{R} = \{j_1, j_2, \dots, j_n, j_1\}$  так, щоб його довжина  $L(\tilde{R})$  була мінімальною:

$$L(\tilde{R}) = \min_{\{R\}} [L(R)], \tag{1}$$

де  $\{R\}$  – множина можливих кільцевих маршрутів, які включають всі пункти.

Довжина  $L(R)$  маршруту  $R = \{j_1, j_2, \dots, j_n, j_1\}$  визначається за формулою

$$L(R) = \sum_{k=1}^{n-1} d(j_k, j_{k+1}) + d(j_n, j_1), \tag{2}$$

де  $d(j_k, j_{k+1})$  – елемент матриці найкоротших відстаней між пунктами.

Для цієї задачі не мають значення обсяги завезення вантажів отримувачам, оскільки вважається, що кільцевий маршрут забезпечує можливість задоволення всього попиту. Крім того, для побудови маршруту немає значення, який із пунктів  $1, 2, \dots, n$  є відправником. Для виконання такого маршруту доцільно вибрати автомобіль мінімальної вантажомісткості, здатний вмістити вантажі відразу всіх отримувачів [1, 2].

На сьогодні задача маршрутизації перевезень, в т. ч. задача комівояжера, є однією з розповсюджених задач оптимізації. За своєю складністю вона відноситься до класу NP-складних задач, тобто складність обчислення експоненційно залежить від обсягу початкових даних [1]. Для розв'язування цієї задачі використовують точні, евристичні та метаевристичні методи [3].

Точні методи розв'язування задачі комівояжера зазвичай орієнтуються на її загальне формулювання,

в якому пропонується симетрична чи несиметрична матриця відстаней між пунктами. Вони забезпечують пошук оптимального розв'язку, проте через надмірне зростання тривалості обчислень за збільшення кількості пунктів їх неможливо застосовувати для задач з більш ніж 25–30 вершинами [2, 3].

Евристичні або класичні методи передбачають відносно обмежений пошук розв'язків, і зазвичай знаходять близький до оптимального розв'язок значно швидше, порівняно з точними методами, проте їх ефективність знижується при наближенні до кінця обчислень [3]. Ефективністю алгоритму тут вважається здатність знаходити якомога коротші маршрути.

Більшість метаевристичних методів базуються на спостереженнях за явищами живої і неживої природи. До них належать генетичний алгоритм, клітинкові автомати, алгоритм імітації відпалу, алгоритм на основі мурашиних колоній, пошук з вилученнями та інші [3–5].

Значна кількість робіт, присвячених метаевристичним методам розв'язування задачі комівояжера, створили ситуацію невизначеності, коли немає змоги однозначно визначити найкращий алгоритм для практичного впровадження. Оскільки фізична основа мурашиного алгоритму – поведінка мурах під час пошуку їжі – є найближчою до формулювання цієї задачі, тому для її розв'язування нами обрано цей алгоритм.

## 2. 2. Характеристика та етапи мурашиного алгоритму

Ідея мурашиного алгоритму виникла при спостереженні за процесом пошуку їжі в колонії мурах. У своїх пошуках мурахи залишають слід (феромон) на пройденому шляху, який дає інформацію іншим мурахам, приваблюючи їх запахом. Феромон – досить стійка речовина, яка може сприйматися мурахами кілька діб, причому чим вища концентрація феромона на шляху, тим більше мурах буде рухатися по ньому. З часом феромон випаровується, що дозволяє мурахам адаптувати свою поведінку до змін довкілля. Розподіл феромона по простору пересування мурах є свого роду динамічно змінюваною глобальною пам'яттю мурашника. Будь-яка мураха у фіксований момент часу може сприймати і змінювати лише один локальний осередок цієї глобальної пам'яті [5–7]. Узагальнену блок-схему мурашиного алгоритму наведено на рис. 1 [8, 9].

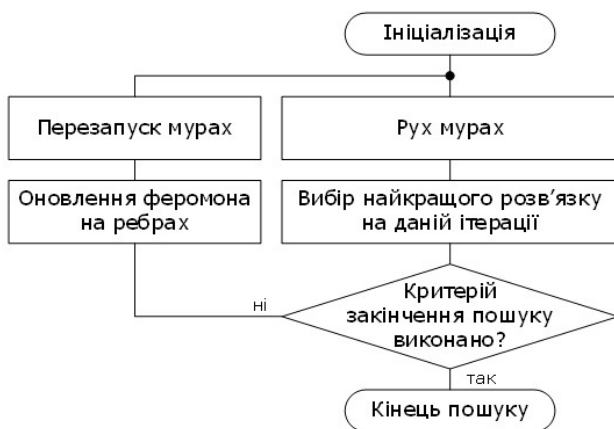


Рис. 1. Узагальнена блок-схема мурашиного алгоритму

Під час ініціалізації алгоритму задаються такі параметри:  $\alpha$  – коефіцієнт, що визначає відносну значущість шляху (кількість феромона на шляху),  $\beta$  – параметр, що показує значимість відстані,  $\rho$  – коефіцієнт кількості феромона, який мураха залишає на шляху,  $Q$  – константа, що відноситься до кількості феромона, який було залишено на шляху та кількість ітерацій алгоритму. Крім цього, створюється популяція мурах, які поміщаються в один або декілька пунктів мережі.

У разі, якщо мураха ще не закінчила рух, тобто не відвідала всі вузли мережі, для визначення наступного ребра шляху розраховується ймовірність переходу з  $r$ -ого вузла в  $u$ -ий за формулою:

$$P_{ru} = \frac{\tau_{ru}(t)^\alpha \cdot \eta_{ru}(t)^\beta}{\sum_{k \in J} \tau_{rk}(t)^\alpha \cdot \eta_{rk}(t)^\beta}, \quad (3)$$

де  $P_{ru}$  – ймовірність того, що мураха переміститься з  $r$ -го вузла в  $u$ -ий вузол;  $\tau_{ru}(t)$  – рівень феромона на ребрі між вузлами  $r$  і  $u$  в момент часу  $t$ ;  $\eta_{ru}(t)$  – «видимість» ребра  $ru$ ;  $J$  – множина вузлів, не відвіданих мурахою.

Мураха може переміщуватись лише до тих вузлів, які ще не відвідані. Тому ймовірність переходу розраховується лише для ребер, які ведуть до таких вузлів. Цей крок повторюється доти, поки кожна мураха не завершить шлях. Цикли (замкнуті маршрути) заборонено, оскільки алгоритм передбачає ведення списку відвіданих вузлів.

Після завершення переміщень мурах визначається довжина шляху. Вона дорівнює сумі всіх ребер, по яких пройшла мураха. Кількість феромона, залишеного на кожному ребрі шляху  $i$ -ою мурахою, визначається за формулою:

$$\Delta\tau^i(t) = \frac{Q}{L^i(t)}, \quad (4)$$

де  $L^i(t)$  – довжина шляху  $i$ -ої мурахи.

Отриманий результат використовується для збільшення кількості феромона вздовж кожного ребра, пройденого  $i$ -ою мурахою за формулою:

$$\tau_{ru}(t) = \tau_{ru}(t-1) + \rho \cdot \sum_{i=1}^{N_{ru}} \Delta\tau^i(t), \quad (5)$$

де  $N_{ru}$  – кількість мурах, які відвідали ребро  $ru$ .

На початку руху мурахи в кожного ребра є шанс бути обраним. Щоб поступово видалити ребра, які входять у гірші шляхи в мережі, до всіх ребер застосовується процедура випаровування феромона:

$$\tau_{ru}(t) = \tau_{ru}(t) \cdot (1 - \rho). \quad (6)$$

Після закінчення руху мурах та оновлення значень феромона на ребрах виконується наступна ітерація алгоритму. Після закінчення всіх ітерацій визначається найкоротший маршрут, який і є розв'язком. На початкових ітераціях розв'язок не є точним і навіть може бути одним із гірших, проте на наступних алгоритм

може сформулювати оптимальний або близький до нього результат.

До недоліків мурашиного алгоритму належать [8]:

- складність теоретичного аналізу, оскільки підсумкове рішення формується в результаті послідовності випадкових рішень;
- невизначеність часу збіжності при тому, що збіжність гарантується;
- висока ітеративність методу;
- сильна залежність результатів роботи алгоритму від початкових параметрів пошуку, які підбираються експериментально.

Для задач комівояжера з великою кількістю пунктів описаний мурашиний алгоритм неефективний у зв'язку з [4]:

- можливість втрати найкращого знайденого розв'язку через імовірнісне правило вибору наступного пункту шляху;
- поганою збіжністю поблизу оптимуму через приблизно однаковий вклад як кращих, так і гірших розв'язків в оновлення феромона;
- зберіганням в пам'яті колонії неперспективних варіантів, що значно розширює область пошуку для задач великої розмірності.

Зважаючи на це, для пришвидшення збіжності в околі оптимуму використовуються різні способи покращення мурашиних алгоритмів. Одним з таких є введення елітних мурах, які на кожній ітерації алгоритму проходять шлях, найкоротший із знайдених на даний момент. Підсилений слід феромона вздовж найкращого маршруту буде спрямовувати інших мурах до пошуку розв'язків, що містять декілька ребер найкращого на той момент маршруту [4, 6, 8, 9].

Ще одним способом прискорення пошуку оптимального маршруту за допомогою мурашиних алгоритмів є використання для деяких пунктів шаблону – пункту, який обов'язково буде відвіданий після поточного без обчислення ймовірності переходу за формулою (3). При цьому суттєво скорочується простір пошуку оптимального розв'язку та, відповідно, кількість обчислень, що підвищує ефективність роботи мурашиного алгоритму [10].

### 2.3. Формулювання проблеми

У роботі [11] наведено результати пошуку оптимального маршруту у системі постачання продукції для мережі з одним відправником та 14-ма споживачами з використанням мурашиного алгоритму. При цьому отримано маршрути за критеріями загальної відстані та загальної тривалості руху. Мурашиний алгоритм використовується і для складніших задач. Зокрема, у роботі [12] розроблено мурашиний алгоритм для задачі комівояжера, коли споживачі мають часові вікна доставки вантажу, а у роботі [13] – для узагальненої задачі комівояжера, коли споживачів розділено на групи (кластери).

Однак у цих роботах не проаналізовано впливу параметрів мурашиного алгоритму на його ефективність. Проте, як зазначено у [3], підбір керуючих параметрів алгоритму потрібно виконувати не тільки для різних типів задач, але часто навіть для кожного нового набору початкових даних.

У роботі [14] Марко Доріго встановив комбінації значень керуючих параметрів  $\alpha$  і  $\beta$ , за яких мурашиний алгоритм дає змогу знаходити добрі розв'язки за ко-

роткий період часу. Схожі результати отримано в [15] для задачі комівояжера на мережі з 50-ма пунктами.

У роботах [16, 17] вказується, що використання одних і тих же значень параметрів  $\alpha$  і  $\beta$  недоцільне у різних фазах алгоритму. Під час перших ітерацій на вибір маршруту більший вплив має відстань, оскільки в мурах практично відсутні знання про кращі маршрути. Однак в подальшому вплив кількості феромона повинен бути збільшеним, оскільки саме він відображає інформацію про кращі шляхи. Зважаючи на це, у роботі [16] проведено дослідження роботи мурашиного алгоритму для різних мереж так, що значення параметрів протягом виконання ітерацій змінювалось від  $\alpha=1$ ,  $\beta=5$  до  $\alpha=5$ ,  $\beta=1$ , та встановлено, що це дає змогу прискорити пошук оптимального маршруту. У роботі [17] під час роботи алгоритму змінювалось лише значення параметра  $\beta$  (від 5 до 2), причому критерієм зміни є ентропія випадкової величини

$$E = -\sum_i p_i \log p_i, \quad (7)$$

де  $p_i$  – імовірність входження кожного маршруту у матрицю значень феромонів.

Однак варто відзначити, що у цих роботах не розглянуто впливу значення параметра  $\rho$ , який відповідає як за кількість феромона, який мураха залишає на шляху (позитивний зворотній зв'язок), так і за випаровування феромона (негативний зворотній зв'язок). При цьому випаровування феромона є засобом запобігання збіжності алгоритму до локально-оптимального розв'язку (якщо б феромон не випаровувався, то перший знайдений шлях був би найбільш привабливим). Дослідження впливу значення параметра  $\rho$  на ефективність роботи алгоритму наведено у роботі [18], однак тут розглядалась мережа з 10-ма пунктами, а реальні системи постачання (чи збору) вантажу часто містять більшу кількість споживачів.

Проведений аналіз вказує на потребу дослідження з підбору керуючих параметрів мурашиних алгоритмів на мережах з різною кількістю пунктів для подальшого використання отриманих результатів у складніших задачах маршрутизації перевезень.

---

### 3. Мета та задачі дослідження

---

Метою роботи є підвищення ефективності роботи мурашиного алгоритму для задачі комівояжера за рахунок використання відповідних значень керуючих параметрів алгоритму.

Задачами дослідження є реалізація мурашиного алгоритму у середовищі MATLAB, проведення дослідження його роботи у три етапи для мереж з різною кількістю пунктів та встановлення таких значень керуючих параметрів, за яких алгоритм швидше знаходить оптимальний маршрут.

---

### 4. Методика виконання дослідження мурашиного алгоритму

---

Для проведення дослідження було створено матриці відстаней для мереж з кількістю пунктів 15, 20, 25

та 30, що відповідає реальним маршрутам розвезення вантажів одним автомобілем. Розглядати мережі з меншою кількістю пунктів недоцільно, оскільки точні методи швидко знаходять для них оптимальні розв'язки. Розглядати мережі з більшою кількістю пунктів також недоцільно, оскільки в реальних системах збору чи доставки вантажу одним автомобілем складно виконати об'їзд такої кількості пунктів протягом робочої зміни з одним завантаженням.

Мурашиний алгоритм реалізовано у середовищі MATLAB. На початку кожної ітерації у всіх пунктах мережі розміщується по одній мураші. Для вибору подальшого напрямку руху на маршруті складається список з індексів невідвіданих пунктів та за формулою (3) визначається наступний пункт маршруту. Коли усі пункти відвідано, мураха повертається у свій початковий пункт. Обчислюються довжини маршрутів, отриманих всіма мурашками та вибирається найкоротший. Він порівнюється з поточним значенням довжини найкоротшого на момент ітерації  $i$  маршруту і якщо він є меншим, то знайдений розв'язок вважається новим рекордом (найкоротшим маршрутом, відомим на ітерації  $i$ ).

Оскільки керуючі параметри алгоритму, які входять у залежності (3)–(6), впливають на ефективність його роботи алгоритму. Тому на першому етапі досліджень проводилось визначення такої комбінації керуючих параметрів алгоритму, за якої він знайде коротший маршрут за однакової кількості ітерацій (у дослідженні прийнято 50 ітерацій з трикратним повторенням).

На другому етапі дослідження для обраної комбінації параметрів для кожної з мереж встановлювалась мінімальна кількість ітерацій, потрібну для пошуку найкоротшого маршруту.

Оскільки за результатами досліджень, наведених у [6], оптимальною кількістю елітних мурах є 3, тому на третьому етапі в алгоритм було введено три елітні мурахи, які підсилюють рівень феромона на ребрах маршруту, який вважається найкоротшим на  $i$ -тій ітерації. При цьому встановлювалось, як це впливає на пришвидшення пошуку оптимального маршруту.

**5. Аналіз результатів дослідження ефективності роботи мурашиного алгоритму**

Результати дослідження впливу параметрів алгоритму на ефективність його роботи (перший етап) наведено в табл. 1 (числа показують найкоротший знайдений маршрут для трьох послідовних повторень алгоритму для тієї ж мережі). Видно, що найкращі результати досягаються за таких комбінацій параметрів:  $\alpha=1, \beta=5, \rho=0,2$  та  $\alpha=1, \beta=5, \rho=0,5$ . Збільшення значення параметра  $\beta$  покращує результати пошуку найкоротшого маршруту в той час, як збільшення значення параметра  $\alpha$  його погіршує. Оскільки значення параметра  $\rho$  неістотно впливає на результат пошуку

найкоротшого маршруту, то у подальшому використано такі значення параметрів:  $\alpha=1, \beta=5, \rho=0,2$ .

На другому та третьому етапах досліджень для обраної комбінації параметрів встановлено мінімальну кількість ітерацій, за якої алгоритм знаходить найкоротший маршрут (табл. 2). Видно, що оптимальний маршрут знаходиться за суттєво меншою кількістю ітерацій, ніж кількість усіх можливих варіантів кільцевих маршрутів у разі розв'язування задачі методом повного перебору. При цьому наявність елітних мурах дає змогу додатково зменшити кількість ітерацій для досягнення оптимального маршруту (для наведених мереж – на 30–200 ітерацій). Це вказує на те, що мурашиний алгоритм є ефективним методом для швидкого розв'язування задачі комівояжера.

Таблиця 1

Розв'язки задачі комівояжера за різних значень керуючих параметрів мурашиного алгоритму

Керуючі параметри алгоритму			Мережа з кількістю пунктів			
$\alpha$	$\beta$	$\rho$	15	20	25	30
0,5	5	0,2	1681 1703 1701	2209 2199 2205	2758 2754 2800	3350 3312 3357
0,5	5	0,5	1677 1682 1672	2218 2214 2198	2784 2766 2760	3256 3302 3293
1	1	0,2	1766 1760 1759	2268 2323 2203	2852 2894 2870	3472 3505 3443
1	1	0,5	1753 1750 1729	2340 2309 2276	2864 2897 2850	3442 3491 3432
1	2	0,2	1672 1668 1701	2197 2207 2178	2751 2773 2795	3284 3325 3320
1	2	0,5	1668 1725 1684	2204 2185 2199	2766 2744 2754	3281 3322 3278
1	5	0,2	1668 1672 1668	2134 2162 2138	2676 2713 2681	3216 3211 3168
1	5	0,5	1668 1672 1668	2142 2138 2135	2715 2665 2676	3206 3218 3189

Таблиця 2

Результати дослідження роботи мурашиних алгоритмів

Кількість пунктів мережі	Кількість ітерацій при застосуванні мурашиного алгоритму		Можлива кількість варіантів маршрутів
	класичного	з елітними мурашками	
15	100	70	$1,30 \cdot 10^{12}$
20	400	300	$2,43 \cdot 10^{18}$
25	1000	800	$1,55 \cdot 10^{25}$
30	3000	3000	$2,65 \cdot 10^{32}$

На рис. 2 наведені процедури пошуку оптимального шляху для мережі з 20-ма пунктами. Видно, що на початкових ітераціях алгоритм з елітними мурашками швидко переходить від довгих маршрутів до коротших. Проте пошук оптимального маршруту триває відносно довго. Так, маршрут довжиною 2135 був знайденим класичним алгоритмом на 69-й ітерації (рис. 2, а), оптимальний маршрут довжиною 2134 – на 307-й ітерації (рис. 2, б). З рис. 2 також видно, що мурахи не припиняють пошук нових маршрутів навіть у разі, коли оптимальний маршрут уже знайдений. Це є перевагою мурашиного алгоритму, оскільки не відбувається зациклення алгоритму на локальному мінімумі.

Однак введення елітних мурах в алгоритм не завжди призводить до підвищення ефективності його роботи. Зокрема, для мережі з 30-ма пунктами алгоритм за 3000 ітерацій не завжди знаходить опти-



мальний маршрут довжиною 3160, а інколи близький до оптимального довжиною 3175. Тобто, алгоритм швидко знаходить локально-оптимальні розв'язки і заціклюється на них, оскільки елітні мурахи на кожній ітерації постійно підвищують рівень феромона на вже відомому на той час найкоротшому маршруті, який може не бути оптимальним.

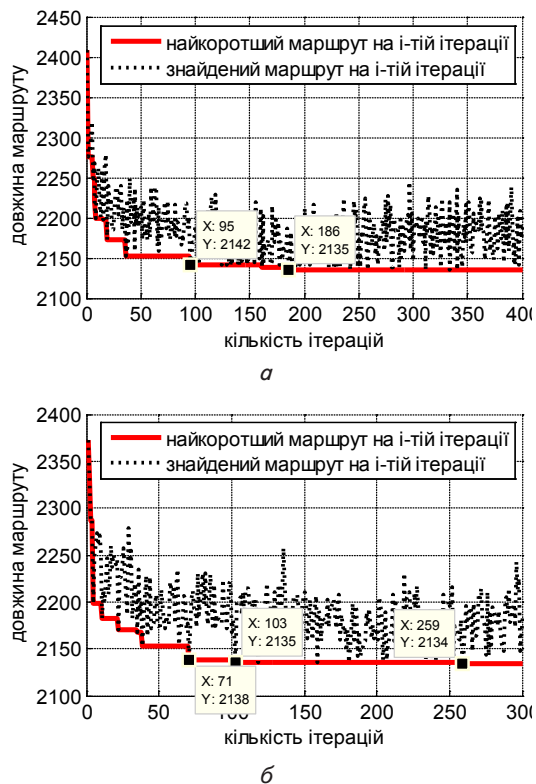


Рис. 2. Процедура пошуку оптимального маршруту на мережі з 20-ма пунктами: а – за використанням класичного мурашиного алгоритму; б – за використання елітного мурашиного алгоритму

## 6. Висновки

Мурашиний алгоритм є ефективним метаевристичним методом розв'язування задачі комівояжера. За рахунок своєї фізичної природи – моделювання поведінки мурах – він дає змогу швидко знаходити оптимальні маршрути. Проте ефективність його роботи залежить від значень керуючих параметрів, і потреба їх підбору є недоліком, що ускладнює його впровадження в програмні продукти для масового застосування.

У роботі було розроблено мурашиний алгоритм для задачі комівояжера та реалізованого його в середовищі MATLAB. У результаті дослідження його роботи у три етапи за різних значень керуючих параметрів на мережах з різною кількістю пунктів встановлено, що найкращі найкоротші маршрути досягаються за використання трьох елітних мурах та таких значень керуючих параметрів:  $\alpha = 1$ ,  $\beta = 5$ ,  $\rho = 0,2$ .

Отримані результати можна в подальшому використовувати не лише для розв'язування задачі коміво-

яжера, але і для задачі маршрутизації автомобільних перевезень, якщо мурашиний алгоритм враховуватиме додаткові умови (обмежену вантажність автомобіля, тривалість роботи на маршруті, часові вікна доставки вантажу тощо).

## Література

1. Лещеніх, О. А. Методи і моделі оптимізації транспортних процесів і систем: навч. посібник [Текст] / О. А. Лещеніх, О. Ф. Кузькін. – Запоріжжя: ЗНТУ, 2006. – 434 с.
2. Таха Хемди, А. Введение в исследование операций [Текст] / Хемди А. Таха; 7-е изд., пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 912 с.
3. Пожидаев, М. С. Алгоритмы решения задачи маршрутизации транспорта [Текст] : дис. ... канд. техн. наук: 05.13.18 / М. С. Пожидаев. – Томск: ГОУВПО «Томский государственный университет», 2010. – 136 с.
4. Штовба, С. Д. Муравьиные алгоритмы: теория и применение [Текст] / С. Д. Штовба // Программирование. – 2005. – № 4. – С. 1–16.
5. Штовба, С. Д. Мурашині алгоритми оптимізації [Текст] / С. Д. Штовба, О. М. Рудий // Вісник ВПІ. – 2004. – № 4. – С. 62–69.
6. Штовба, С. Д. Муравьиные алгоритмы [Текст] / С. Д. Штовба // Exponenta PRO. – 2003. – № 4 (4). – С. 70–75.
7. Джонс, М. Т. Программирование искусственно-го интеллекта в приложениях: пер. с англ. [Текст] / М. Т. Джонс – М.: ДМК Пресс, 2004. – 312 с.
8. Субботін, С. О. Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей [Текст] : монографія / С. О. Субботін, А. О. Олійник, О. О. Олійник; під заг. ред. С. О. Субботіна. – Запоріжжя: ЗНТУ, 2009. – 375 с.
9. Dorigo, M. Ant Colony Optimization [Text] / M. Dorigo, T. Stuzle. – Cambridge: A Bradford Book, 2004. – 305 p.
10. Кажаров, А. А. Использование шаблонных решений в муравьиных алгоритмах [Текст] / А. А. Кажаров, В. М. Курейчик // Известия ЮФУ, Технические науки. – 2013. – № 7 (144). – С. 17–22.
11. Данчук, В. Д. Оптимізації пошуку шляхів по графу в динамічній задачі комівояжера методом модифікованого мурашиного алгоритму [Текст] / В. Д. Данчук, В. В. Сватко // Системні дослідження та інформаційні технології. – 2012. – № 2. – С. 78–86.
12. Cheng, C.-B. A modified ant colony system for solving the travelling salesman problem with time windows [Text] / C.-B. Cheng, C.-P. Mao // Mathematical and Computer Modelling. – 2007. – Vol. 46, Issue 9-10. – P. 1225-1235. doi:10.1016/j.mcm.2006.11.035
13. Yang, J. An ant colony optimization method for generalized TSP problem [Text] / J. Yang, X. Shi, M. Marchese, Y. Liang // Progress in Natural Science. – 2008. – Vol. 18, Issue 11. – P. 1417–1422. doi:10.1016/j.pnsc.2008.03.028
14. Dorigo, M. Ant Colony Systems: a Cooperative Learning Approach to the Traveling Salesman Problem [Text] /

- M. Dorigo, L. M. Gambardella // IEEE Transactions on Evolutionary Computation. – 1997. – Vol. 1, Issue 1. – P. 53–66. doi:10.1109/4235.585892
15. Игнатъев, А. Л. Использование алгоритма муравьиных колоний для решения задачи маршрутизации транспортных средств [Электронный ресурс] : матер. IV междуна. науч.-прак. конф. / А. Л. Игнатъев // Современные информационные технологии и ИТ-образование. – М.: МГУ им. М. В. Ломоносова, 2009. – Режим доступа : [http://2009.it-edu.ru/docs/Sekziya\\_8/3\\_Ignat%27ev\\_Ignatyev.doc](http://2009.it-edu.ru/docs/Sekziya_8/3_Ignat%27ev_Ignatyev.doc)
  16. Jun-Mam, K. Application of an Improved Ant Colony Optimization on Generalized Traveling Salesman Problem [Text] / K. Jun-Man, Z. Yi // Energy Procedia. – 2012. – Vol. 17. – P. 319–325. doi: 10.1016/j.egypro.2012.02.101
  17. Hlaing, Z.C. Solving Traveling Salesman Problem by Using Improved Ant Colony Optimization Algorithm / Z. C. Hlaing, M. A. Khine // International Journal of Information and Education Technology. – 2011. – Vol. 1, № 5. – P. 404-409. – doi: 10.7763/ijiet.2011.v1.67.
  18. Иванова, И. Исследование работы муравьиного алгоритма на примере задачи коммивояжера [Текст] / И. Иванова // Research and Technology: Step into the Future. – 2012. – Vol. 7, no. 3. – P. 39-46.

□                      □

*Пропонується новий чисельно-аналітичний метод для керування тепловими процесами. Метод базується на розв'язанні обернених задач ідентифікації функцій управління за заданими оптимальними тепловими режимами в часі. При визначенні ступеня оптимальності заданих теплових режимів враховуються всі необхідні обмеження на розподіл температури, її градієнти, швидкість нагрівання чи охолодження*

*Ключові слова: управління, теплові процеси, чисельно-аналітичний метод, структура рішення, S-функції, функціонал, ідентифікація, модель, розпаралелювання*

□                      □

*Предлагается новый численно-аналитический метод для управления тепловыми процессами. Метод базируется на решении обратных задач идентификации функций управления по заданным оптимальным тепловым режимам во времени. При определении степени оптимальности заданных тепловых режимов учитываются все требуемые ограничения на распределение температуры, ее градиенты, скорость нагрева или охлаждения*

*Ключевые слова: управление, тепловые процессы, численно-аналитический метод, структура решения, S-функции, функционал, идентификация, модель, распараллеливание*

□                      □

УДК 536.24

# УПРАВЛЕНИЕ ТЕПЛОВЫМИ ПРОЦЕССАМИ ПРИ ТОЧНОМ УЧЁТЕ ГЕОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ С ПОМОЩЬЮ S-ФУНКЦИЙ

**А. П. Слесаренко**

Доктор физико-математических наук, профессор,  
ведущий научный сотрудник,  
Отдел моделирования  
идентификации тепловых процессов  
Институт проблем машиностроения  
им. А. Н. Подгорного НАН Украины  
ул. Д. Пожарского, 2/10, г. Харьков, Украина, 61046

**Ю. В. Журавлев**

Кандидат технических наук, доцент, профессор  
Кафедра автоматизации производственных процессов  
Харьковский национальный  
университет строительства и архитектуры  
ул. Сумская, 40 г. Харьков, Украина, 61002

## 1. Введение

В настоящее время ежегодно производится около 160 Тб уникальных данных (согласно другим источникам это число ещё больше и составляет порядка 1 экзбайта (1000000 Тб)). Процесс конвертирования такого количества данных в структурированную форму становится невозможным. Поэтому предпочтительно организовать выходную информацию при решении задач в форме, которую не нужно дополни-

тельно обрабатывать для сжатия, анализа и поиска скрытой закономерности.

Таким образом, алгоритмы, построенные на базе регионально-структурных решений задач управления тепловыми процессами, должны быть представлены в такой форме, которая позволила бы с максимальной возможной скоростью воспроизводить их и хранить в минимальном объёме машинной памяти, что давало бы возможность оперативно их анализировать.