

УДК 004.942(67)

DOI: 10.18372/2310-5461.37.12364

**С. С. Бучик**, д-р техн. наук, доц.  
Національний авіаційний університет  
orcid.org/0000-0003-0892-3494  
e-mail: s\_stbu@ukr.net;

**І. В. Пулеко**, канд. техн. наук, доц.  
Житомирський військовий інститут імені С. П. Корольова  
orcid.org/0000-0001-8875-017X  
e-mail: pulekoigor@gmail.com;

**І. В. Половніков**  
Житомирський військовий інститут імені С. П. Корольова  
orcid.org/0000-0001-9367-727X  
e-mail: piv331@gmail.com

## МЕТОДИКА ОПТИМІЗАЦІЇ ПОШУКУ ШЛЯХУ В ГРАФІ ЗА РАХУНОК ВІДКИДАННЯ ЗАЙВИХ ВУЗЛІВ

### Актуальність дослідження

На сьогодні, цифрові пристрої стають меншими, а область їх використання кожного дня — ширшою.

Саме тому на сьогодні необхідно, щоб цифрові пристрої не просто виконували свої функції оптимально швидко і якісно, наскільки це можливо для певної категорії практичних завдань.

Однією з категорій практичних завдань, є навігація та пошук оптимального шляху між двома точками на карті. Ці точки можна подати вузлами, а пошук шляху на карті звести до пошуку шляху між вузлами графу. Люди звикли користуватися навігатором у мобільному телефоні та в особистому автомобільному транспорті. Це потребує від подібних пристроїв постійного удосконалення як апаратного забезпечення за рахунок підвищення його швидкодії, так і програмного забезпечення за рахунок не тільки оновлення карт, але і оптимізації знайденого шляху.

Звісно навігацією все не закінчуються, оскільки сучасні електронні пристрої, а саме багатопланові схеми друкованих плат проектуються за допомогою алгоритмів пошуку шляху. Великі підприємства та військова сфера використовує дані алгоритми для пошуку найбезпечнішого маршруту передачі пакетів даних в прихованій мережі.

Аналізуючи загрози державним інформаційним ресурсам (ДІР) виникає питання врахування ризику проходження інформації скрізь певні вузли мережі передачі даних та оптимізації цього шляху саме за рахунок урахування всіх складових, які впливають на цей процес.

Дані питання безумовно актуальні і є складовими загальної системи захисту ДІР, яка описана в праці [1, с. 95].

Отже, задачу пошуку оптимального шляху можна зустріти в повсякденному житті практично на кожному шляху, але проблема залишається актуальною і на далі потребує все більшого різноманіття підходів і методів її вирішення. В зв'язку з чим, тематика статті, яка присвячена розробці методики оптимізації пошуку шляху в графі за рахунок відкидання зайвих вузлів є **актуальною** для сьогодення.

Таким чином **метою** статті — розробка методики для оптимізації пошуку шляху в графі за рахунок відкидання зайвих вузлів на основі транзитивного замикання матриці відношення та перевірка її ефективності.

### Аналіз останніх досліджень і публікацій

Питаннями оптимізації пошуку шляху в графі займається багато як вітчизняних, так і закордонних учених, оскільки дана тематика є доволі актуальною саме з практичної точки зору. Загалом відсікання або іншими словами відкидання зайвих вузлів графу є очевидним рішенням для оптимізації пошуку шляху і на сьогодні вже існує кілька методів зі схожою ідеєю.

Подібний підхід під назвою «Метод гілок та меж» було запропоновано вченими А. Х. Лендом та А. Г. Дойгом для вирішення задач цілочисельного програмування. Загалом метод є варіантом повного перебору з відкиданням підмножин, які точно не мають ніякої цінності для знаходження оптимального рішення для певної задачі [2].

Алгоритм альфа-бета-відсікання неодноразово і незалежно один від одного пропонували різні вчені по всьому світові. Даний алгоритм, прагне скоротити кількість вузлів, які оцінюються в дереві пошуком мінімуму. В основі алгоритму лежить ідея, що оцінювання гілки дерева

пошуку може бути достроково припинено (і не потребує повного перебору всіх варіантів), якщо було знайдено оптимальне рішення на певному кроці [3].

Подібний метод до альфа-бета-відсікання було запропоновано Дж. Перлом який називається методом пошуку по першому найкращому співпаданню. Алгоритм аналізує граф і обирає найбільш перспективні вузли відповідно до певних критеріїв [4, с. 48].

У роботі авторів В. Д. Данчука та В. В. Сватко запропоновано метод вирішення завдання пошуку мінімального шляху по графу на основі застосування модифікованого мурашиного алгоритму [5, с. 78].

Як вже було наведено вище, існує ще багато подібних методів оптимізації пошуку шляху в графі, але на відміну від них запропоновано методику оптимізації пошуку оптимального шляху в графі за рахунок відкидання зайвих вузлів на основі кластерного аналізу з використанням транзитивного замикання матриці відношення.

#### Виклад основного матеріалу

На рис. 1 представлена загальна схема методики оптимізації пошуку шляху в графі за рахунок відкидання зайвих вузлів. Вона складається з семи основних кроків. Перші три та останній крок можуть дещо змінюватись залежно від завдання.

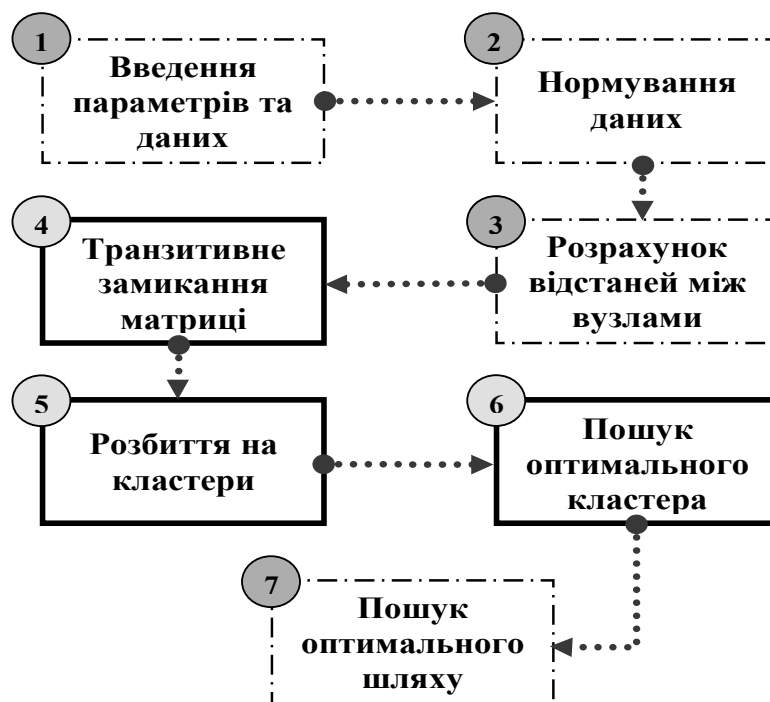


Рис. 1. Загальна схема методики оптимізації пошуку шляху в графі за рахунок відкидання зайвих вузлів

Розглянемо методику детальніше.

**Крок 1.** *Уведення параметрів та початкових даних*, а саме кількість вузлів, які позначатимемо  $N$ , а самі вузли  $S_i$ , де  $i$  — номер вузла. Параметрами вузлів можуть бути географічні координати, висота та ін.

**Крок 2.** *Нормування даних*. Оскільки введені параметри можуть сильно відрізнятися за числовим значенням, виконується їх нормування, тобто приведення їх до однакового вигляду. В даному випадку відбувається пошук максимумів по кожному з параметрів вузлів та діленням всіх значень на цей максимум:

$$x_{ij} = \frac{x_{ij}}{\text{MAX}(x_j)} \quad (1)$$

де  $x_{ij}$  — поточне значення  $j$ -го параметра  $i$ -го вузла;  $\text{MAX}(x_j)$  — максимальне значення  $j$ -го параметра серед усіх вузлів.

Нормалізація дає змогу привести значення всіх параметрів до спільного вигляду в діапазоні від 0 до 1.

**Крок 3.** *Розрахунок відстаней між вузлами*. Нормалізовані на кроці 2 дані характеризують кожен окремий вузол. Для отримання характеристики відносно інших вузлів необхідно виконати розрахунок відношення відстаней за допомогою певної міри або метрики.

У даному випадку використана класична формула евклідової відстані [6]:

$$d(x, x^*) = \sqrt{\sum_i^p (x_i - x_i^*)^2} \quad (2)$$

де  $d$  — відношення;  $p$  — кількість параметрів;  $x$  — перший вузол;  $x^*$  — другий вузол, з яким виконується порівняння.

Після розрахунку відношення між усіма  $N$  вузлами отримується матриця відношення  $R$  розмірністю  $N \times N$  вигляду:

$$R = \begin{bmatrix} x_{1j} & \dots & x_{1N} \\ \dots & \dots & \dots \\ x_{Nj} & \dots & x_{NN} \end{bmatrix}, \quad (3)$$

де  $x_{ij}$  — відстань між  $i$ -м та  $j$ -м вузлом;  $N$  — кількість вузлів.

Графічно вузли з двома параметрами можна подати у вигляді точок на площині, а відношення це геометрична відстань між цими вузлами (рис. 2).

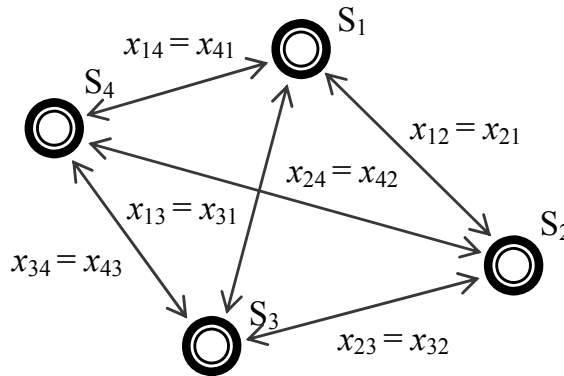


Рис. 2. Графічне представлення вузлів на площині з відстанями

Важливо пам'ятати, що відстань  $x_{ij} = x_{ji}$ .

Цим зумовлена симетричність матриці відношення, але можливе використання і несиметричної матриці.

**Крок 4.** Транзитивне замикання матриці відношення  $R$ , що відбувається за формулою [7, с. 94]:

$$R^*(x, z) = \underset{y}{\text{MIN}}[\text{MAX}(R(x, y), R(y, z))], \quad (4)$$

де  $R$  — початкова матриця відношення;  $R^*$  — транзитивно-замкнена матриця.

Транзитивне замикання виконується ітеративно, доти доки отримана матриця не буде дорівнювати попередній, тобто:

$$R_{k+1}^* = R_k^*. \quad (5)$$

**Крок 5.** Розбиття на кластери. Після отримання на 4 кроці транзитивно-замкненої [8, с. 57] матриці  $R^*$  необхідно розсортувати вузли по кластерам у вигляді множин, де членами множини кластеру є вузли, які увійшли в цей кластер. При цьому кількість кластерів заздалегідь невідома.

$$L_k = \{S_i\}, \quad (6)$$

де  $L_k$  — кластер у вигляді множини вузлів;  $k$  — номер множини;  $S_i$  — вузол;  $i$  — номер вузла.

Необхідно пам'ятати, що кожен наступний кластер може включати вузли з попередніх кластерів, тому справедливий вираз:

$$L_k \subset L_{k+1}, \quad (7)$$

де  $L_k$  — кластер у вигляді множини вузлів;  $k$  — номер множини.

Таким чином отримуємо дерево, яке можна представити у вигляді дендограми (рис. 3).

**Крок 6.** Пошук оптимального кластеру. Оптимальним можна вважати той кластер в якому вперше об'єднуються початковий і кінцевий вузол.

Усі вузли оптимального кластеру використовуються для подальших розрахунків, а всі вузли які не ввійшли в цей кластер в розрахунках не використовуються, тобто іншими словами відкидаються, як показано на рис. 4.

Такий підхід виключає вірогідність того що, цільові вузли опиняться в різних кластерах.

**Крок 7.** Пошук оптимального шляху. Для пошуку оптимального шляху між вузлами кластеру було використано класичний алгоритм Дейкстри [9], який добре справляється з подібними задачами.

**Перевірка працездатності методики та оцінка результатів**

Для перевірки працездатності та ефективності методики було розроблено програмне забезпечення (рис. 5) на мові програмування високо рівня C++. Програма дозволяє вводити данні (рис. 5), проводити кластеризацію з візуалізацією дендограми (рис. 6) та розраховувати і відображати оптимальний шлях (рис. 7).

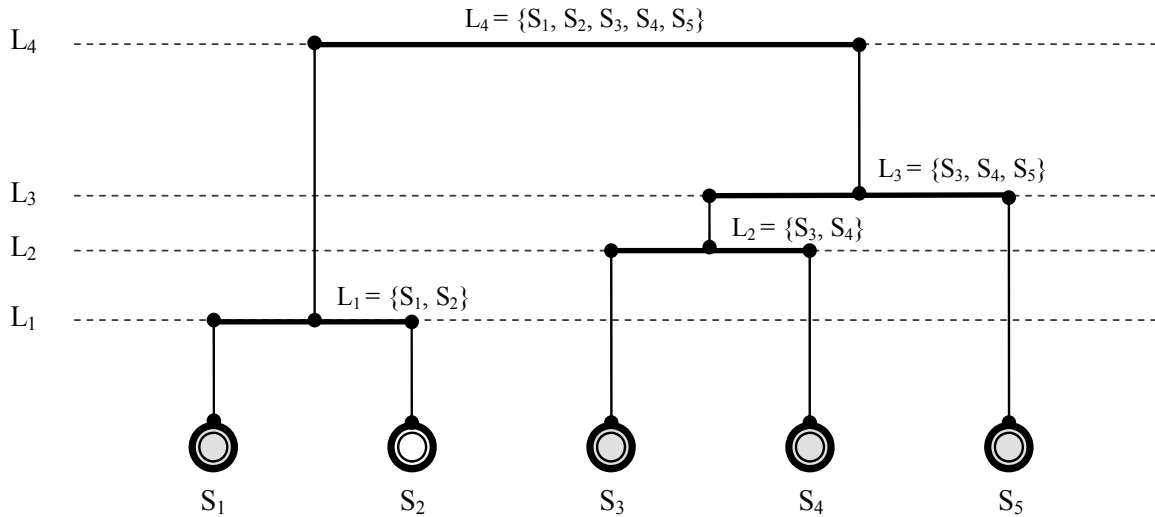


Рис. 3. Дендограма з 4 рівнями (об'єднанням 5 вузлів в 4 кластери)

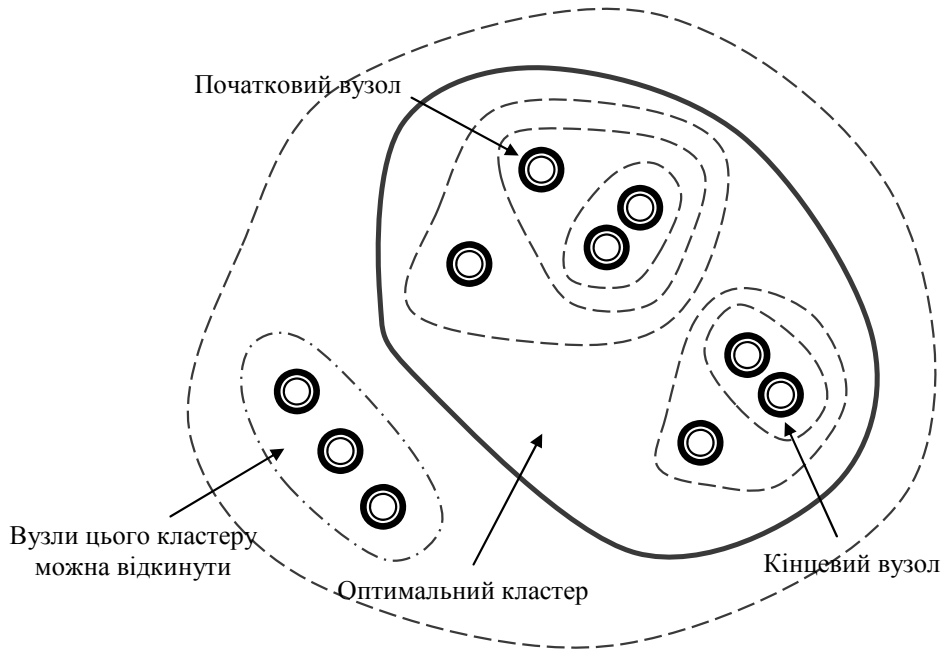


Рис. 4. Оптимальний кластер, та кластер який можна відкинути

tryonTC:0,153 AL:0,126 SP:0,01 SP:0,004 SP:0,002 SP:0,008 SP:0,005 SP:0,007

Введення Кластеризація Оптиміальний шлях

Параметри: 2 Об'єкти: 100 Створити Інтервал [ 0 1000 ] Заповнити Округлення: 3 Кластеризація

|    | Название | X0  | X1  |
|----|----------|-----|-----|
| 0  | S0       | 41  | 467 |
| 1  | S1       | 334 | 500 |
| 2  | S2       | 169 | 724 |
| 3  | S3       | 478 | 358 |
| 4  | S4       | 962 | 464 |
| 5  | S5       | 705 | 145 |
| 6  | S6       | 281 | 827 |
| 7  | S7       | 961 | 491 |
| 8  | S8       | 995 | 942 |
| 9  | S9       | 827 | 436 |
| 10 | S10      | 391 | 604 |
| 11 | S11      | 902 | 153 |
| 12 | S12      | 292 | 382 |
| 13 | S13      | 421 | 716 |
| 14 | S14      | 718 | 895 |
| 15 | S15      | 447 | 726 |
| 16 | S16      | 771 | 538 |
| 17 | S17      | 860 | 012 |

Рис. 5. Вікно введення даних програми

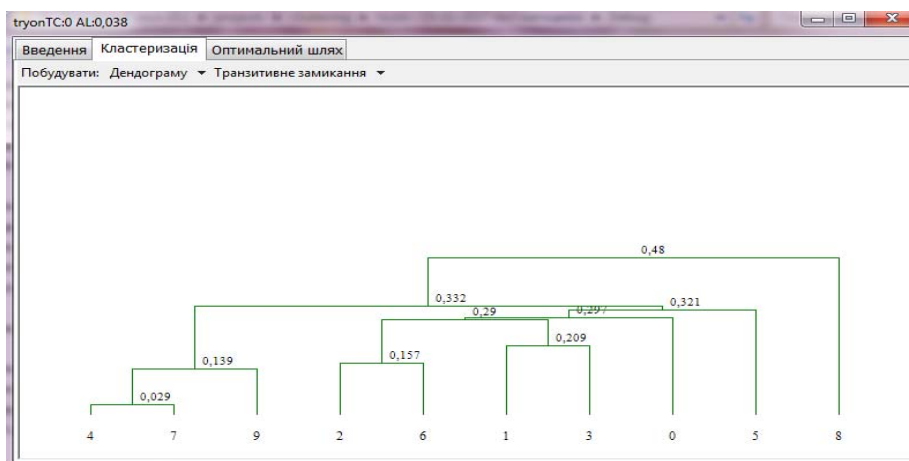


Рис. 6. Візуалізація дендрограми

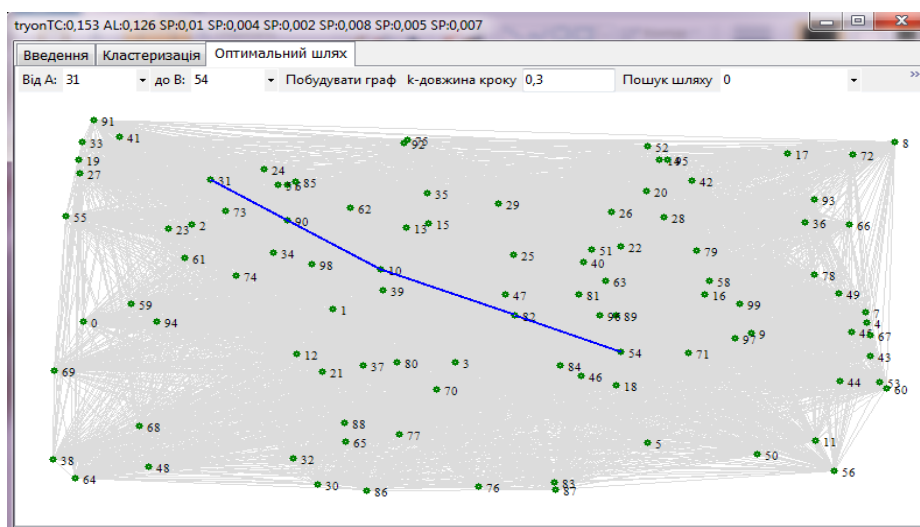


Рис. 7. Пошук та відображення оптимального шляху

Оцінка ефективності проводилась шляхом порівняння розробленої методики з відомим алгоритмом Дейкстри [10], за критерієм мінімального часу на пошук оптимального шляху. Результати досліджень подані у таблиці.

**Результати досліджень**

| Кількість вузлів      | 50       |            | 100        |            | 200       |          | 300         |            | 500         |            |
|-----------------------|----------|------------|------------|------------|-----------|----------|-------------|------------|-------------|------------|
|                       | А        | В          | А          | В          | А         | В        | А           | В          | А           | В          |
| 1                     | 1        | 0          | 6          | 3          | 21        | 9        | 5           | 5          | 33          | 16         |
| 2                     | 6        | 0          | 7          | 2          | 24        | 3        | 17          | 7          | 30          | 15         |
| 3                     | 4        | 1          | 8          | 2          | 16        | 1        | 15          | 3          | 40          | 2          |
| 4                     | 4        | 0          | 8          | 2          | 29        | 5        | 10          | 3          | 31          | 2          |
| 5                     | 7        | 3          | 11         | 3          | 17        | 5        | 15          | 9          | 25          | 15         |
| 6                     | 8        | 3          | 8          | 1          | 24        | 11       | 12          | 10         | 18          | 2          |
| 7                     | 6        | 2          | 12         | 5          | 25        | 7        | 23          | 6          | 28          | 16         |
| 8                     | 9        | 3          | 4          | 4          | 25        | 6        | 16          | 1          | 23          | 11         |
| 9                     | 8        | 2          | 12         | 4          | 23        | 7        | 14          | 3          | 32          | 4          |
| 10                    | 7        | 2          | 8          | 2          | 26        | 6        | 16          | 7          | 34          | 4          |
| Середнє значення (мс) | <b>6</b> | <b>1,6</b> | <b>8,4</b> | <b>2,8</b> | <b>23</b> | <b>6</b> | <b>14,3</b> | <b>5,4</b> | <b>29,4</b> | <b>8,7</b> |

У випадку **A** оцінювався час (мс) пошуку оптимального шляху алгоритмом Дейкстри на всьому графі та у випадку **B** — аналогічний час оптимізації за розробленою методикою.

Оптимізація шляху у кластері, відповідно до методики, також виконувалась алгоритмом Дейкстри.

Для дослідження використали різну кількість вузлів  $N$  від 50 до 500. Порівняння проводилось серіями із 10 експериментів, потім розраховувалось середнє значення для випадків **A** і **B** та відповідного  $N$ .

На графіку (рис. 5) подана апроксимація середніх значень результатів.

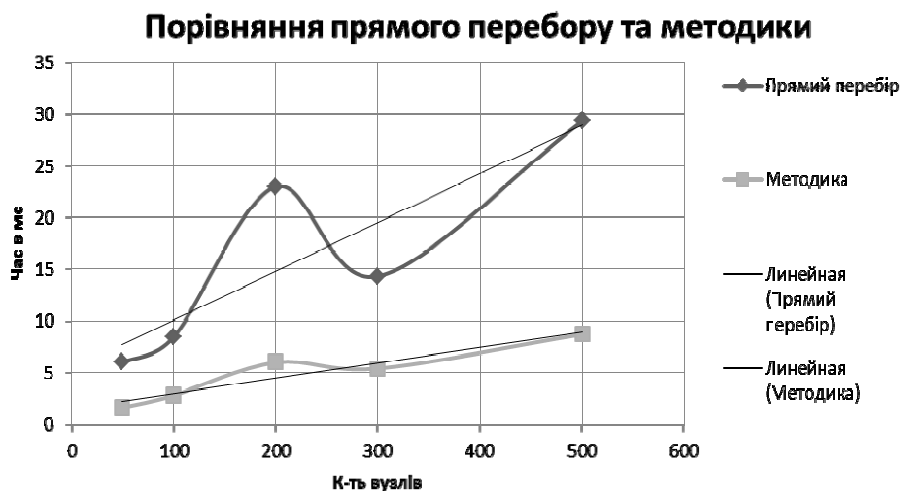


Рис. 5. Графіки залежності середнього часу пошуку оптимального шляху від  $N$

Отримані результати підтверджують ефективність методики в певних категоріях практичних задач. Зі збільшенням кількості вузлів, отримуємо більшу перевагу (менший час) порівняно з класичними підходами прямого перебору по всім вузлам графу.

Однак, треба розуміти, що дана методика добре виконує свої функції за умови проведення транзитивного замикання матриці відношення завчасно, а тому необхідно, щоб вузли, залишалися статичними і не змінювали своїх параметрів певний період часу. При розрахунках для великої кількості вузлів ( $N \geq 500$ ) доцільно провести оптимізацію процедури транзитивного замикання.

### Висновки

Таким чином, у статті розроблено методику оптимізації пошуку шляху за рахунок відкидання зайвих вузлів графу на основі транзитивного замикання матриці відношення. Перевірка працездатності довела її ефективність за критерієм мінімуму часу затраченого на пошук оптимального шляху. При збільшенні кількості вузлів методика дає більший ефект, але доцільно провести оптимізацію процедури транзитивного замикання.

Методику пропонується застосовувати в практичних задачах для пошуку оптимального шляху в умовах стаціонарних вузлів, кількість яких не змінюються тривалий час, а отже транзитивне замикання можна виконати одноразово.

Отримана методика належить до класу евристичних методів оптимізації, а її результат є на-

ближено оптимальним, оскільки точно не відомо, які саме вузли будуть відкинута і наскільки вони важливі для розрахунку оптимального шляху. В подальшому доцільно порівняти методику з іншими відомими евристичними алгоритмами оптимізації.

### ЛІТЕРАТУРА

1. Юдін О. К. Аналіз загроз державним інформаційним ресурсам / О. К. Юдін, С. С. Бучик // Проблеми інформатизації та управління. — 2013. — № 4 (44). — С. 93–99.
2. Land A. H. An Automatic Method of Solving Discrete Programming Problems / A. H. Land, A. G. Doig — [Електронний ресурс]. — Режим доступу: <http://jmvidal.cse.sc.edu/library/land60a.pdf>.
3. Newell A. Computer Science as Empirical Inquiry: Symbols and Search / A. Newell, H. A. Simon // — [Електронний ресурс]. — Режим доступу: [http://delivery.acm.org/10.1145/370000/360022/a1975-newell\\_simon.pdf](http://delivery.acm.org/10.1145/370000/360022/a1975-newell_simon.pdf).
4. Pearl J. Heuristics: Intelligent Search Strategies for Computer Problem Solving. — [Електронний ресурс]. — Режим доступу: [http://mat.uab.cat/~alseda/MasterOpt/Judea\\_Pearl-Heuristics\\_Intelligent\\_Search\\_Strategies\\_for\\_Computer\\_Problem\\_Solving.pdf](http://mat.uab.cat/~alseda/MasterOpt/Judea_Pearl-Heuristics_Intelligent_Search_Strategies_for_Computer_Problem_Solving.pdf).
5. Данчук В. Д. Оптимізації пошуку шляхів по графу в динамічній задачі комівояжера методом модифікованого мурашиного алгоритму / В. Д. Данчук, В. В. Сватко // Системні дослідження та інформаційні технології. — 2012. — № 2. — С. 78–86. — [Електронний ресурс]. — Режим доступу: <http://journal.iasa.kpi.ua/article/viewFile/71975/66948>.

6. Обзор алгоритмов кластеризации данных. — [Электронный ресурс]. — Режим доступа: <https://habrahabr.ru/post/101338/>.

7. **Кофман А.** Введение в теорию нечетких множеств / А. Кофман. — М. : Радио и связь, 1982. — 432 с.

8. **Новиков Ф. А.** Дискретная математика для программистов: учебник для вузов / Ф. А. Новиков — СПб : Питер, 2009. — 384 с.

9. Алгоритм Дейкстры. — [Электронный ресурс]. — Режим доступа:

[https://ru.wikipedia.org/wiki/Алгоритм\\_Дейкстры](https://ru.wikipedia.org/wiki/Алгоритм_Дейкстры)

10. Алгоритм Дейкстры. Поиск оптимальных маршрутов на графе. — [Электронный ресурс]. — Режим доступа: <https://habrahabr.ru/post/111361/>.

11. **Бучик С. С., Пулеко И. В., Половников И. В.** Методика оптимізації пошуку шляху в графі за рахунок відкидання зайвих вузлів.

**Бучик С. С., Пулеко И. В., Половников И. В.**

## **МЕТОДИКА ОПТИМІЗАЦІЇ ПОШУКУ ШЛЯХУ В ГРАФІ ЗА РАХУНОК ВІДКИДАННЯ ЗАЙВИХ ВУЗЛІВ**

*У статті запропоновано методику оптимізації пошуку оптимального шляху в графі за рахунок відкидання зайвих вузлів на основі кластерного аналізу з використанням транзитивного замикання матриці відношення. Проаналізовано існуючі методи оптимізації пошуку шляху, в результаті чого запропоновано методику, що реалізує ідею зменшення кількості вузлів в поточному обрахунку, як результат відбувається зменшення кількості задіяних ресурсів. Розглянуто застосування методики на практиці, та зроблено висновки щодо подальших досліджень в цьому напрямку. Для перевірки працездатності та оцінки ефективності методики розроблено програмне забезпечення на мові програмування високо рівня C++. Отримані результати свідчать про досить високу ефективність оптимізації для статичних вузлів, які не змінюють своїх властивостей певний період часу.*

**Ключові слова:** пошук шляху; оптимізація; транзитивне замикання; кластерний аналіз; оптимальний шлях в графі.

**Buchyk S. S., Puleko I. V., Polovnikov I. V.**

## **METHOD OF OPTIMIZING THE SEARCH FOR A PATH IN A GRAPH BY REJECTING UNNECESSARY NODES**

*The article proposes method optimizing the search of the optimal path in the graph by rejecting extra nodes based on cluster analysis using the transitive closure of the matrix of the relation. The existing methods of optimizing the search of the path have been analyzed, which resulted in the proposed method, which implements the idea of reducing the number of nodes in the current calculation, as a result of which there is a decrease in the number of resources. The application of the method in practice is considered, and conclusions are made regarding further research in this direction. To test performance and evaluate the effectiveness of the method the software was developed on the programming language of high-level C++. The results of the research indicate rather high efficiency of optimization for static nodes that do not change their properties for a certain period of time.*

**Keywords:** search for a path; optimization; transitive closure; cluster analysis; the best path in the graph.

**Бучик С. С., Пулеко И. В., Половников И. В.**

## **МЕТОДИКА ОПТИМІЗАЦІЇ ПОИСКА ПУТИ В ГРАФЕ ЗА СЧЕТ ОТБРАСЫВАНИЯ ЛИШНИХ УЗЛОВ**

*В статье предложена методика оптимизации поиска оптимального пути в графе за счет отбрасывания лишних узлов на основе кластерного анализа с использованием транзитивного замыкания матрицы отношения. Проанализированы существующие методы оптимизации поиска пути, в результате чего предложена методика, реализующая идею сокращения количества узлов в текущих расчетах, как результат происходит сокращение количества задействованных ресурсов. Рассмотрено применение методики на практике, и сделаны выводы о последующих исследованиях в этом направлении. Для проверки работоспособности и оценки эффективности методики разработано программное обеспечение на языке программирования высокого уровня C++. Полученные результаты свидетельствуют о достаточно высокой эффективности оптимизации для статических узлов, которые не меняют своих свойств длительный период времени.*

**Ключевые слова:** поиск пути; оптимизация; транзитивное замыкание; кластерный анализ; оптимальный путь в графе.

Стаття надійшла до редакції 07.02.2018 р.

Прийнято до друку 15.02.2018 р.

Рецензент — д-р техн. наук, проф. Юдін О. К.