Дейкстри дозволяє інженерам та розробникам створювати оптимальні маршрутизаційні системи, що забезпечують швидкий та ефективний рух у різних сферах нашого життя [3].

Список літератури

1. *Система побудови оптимальних маршрутів на основі алгоритмів найкоротших шляхів URL: http://biomedtech.kpi.ua/article/download/185416/185475/413642*
2. *Алгоритм Дейкстри URL: https://ua5.org/algorithm/1970-algorytm-dejkstry.html*
3. *Research on Optimal Path based on Dijkstra Algorithms URL: https://www.atlantis-press.com/article/55917280.pdf*

УДК 004.6

Овчар М. І., студент 1 курсу
спеціальності 122 «Комп'ютерні науки»
Науковий керівник:
Гончар В. М., асистент
кафедри інформаційних технологій

АЛГОРИТМИ ЗНАХОДЖЕННЯ КІЛЬКОСТІ ГАМІЛЬТОНОВИХ ЦИКЛІВ У ГРАФАХ

Донецький національний університет імені Василя Стуса, м. Вінниця

Теорія графів є одним із найзатребуваніших і найдосліджуваніших розділів, що вивчаються дискретною математикою. Такий простий, на перший погляд, розділ використовується в самих різноманітних сферах людського життя, наприклад: в інформаційних технологіях для побудови складних і взаємозв'язаних структур даних, в транспортних перевезеннях для побудови маршрутів перевезень, в соціально дослідницькій сфері для побудови схем соціальних зв'язків і взаємодій, та в інших [1].

Саме тому перед сучасними розробниками програмного забезпечення стоїть завдання створення таких програм, які можуть спростити та автоматизувати процес роботи з графами. Серед таких завдань є і знаходження Гамільтонових циклів, які є основою для багатьох транспортних перевезень.

Аналіз спеціалізованих публікацій показав, що проблеми дослідження Гамільтонових циклів досліджувалися Ваді Аль-Халабі, Омаром Кіттанех, Наман Гаром [3]. Також варто зауважити, що всі ці дослідження базується на основі самих перших досліджень графів Вільяма Гамільтона [2].

Основною метою даної статі є розкриття і аналіз основних алгоритмів, які допомагають в знаходженні Гамільтонових циклів, та підрахунку зальної кількості всіх можливих Гамільтонових циклів, які можна знайти в тому чи іншому графі.

В результаті дослідження була визначено, що розрахунки Гамільтонових циклів є однією із найдосліджуваніших проблем і на сьогодні. Головна проблема розрахунку таких циклів полягає у відсутності достатньо швидких алгоритмів, щоб їх можна було використовувати для графів з великою кількістю вершин у промислових масштабах.

Сучасні алгоритми знаходження кількості Гамільтонових циклів можна поділити на два основні види: знаходження кількості Гамільтонових циклів у повних графах і знаходження кількості Гамільтонових циклів у довільних графах.

Алгоритм знаходження кількості Гамільтонових циклів у повних графах є доволі простим, і представляє із себе лише одну формулу (1):

$$\frac{(n\ -\ 1)!}{2}\ (1).$$

Кількість всіх можливих Гамільтонових циклів у повному графі завжди дорівнює *(n-1)!*, а так як рахуються як прямі так і зворотні проходження одного і того самого циклу, то загальну кількість циклів прийнято зменшувати в два раза.

Проблема даного алгоритму знаходиться в його низькій функціональності. Він розрахований на аналіз лише повних графів і не здатний повертати самі цикли.

Другий спосіб знаходження Гамільтонових циклів, на відміну від попередника, може повертати самі Гамільтонові цикли і може працювати з довільними графами [3]. Цей алгоритм основується на алгоритмі обходу графа в глибину і його можна представити такими кроками:

1. Знаходимо стартову вершину;
2. Переходимо на інциденту вершину, яка ще не проходилася, з якої будемо продовжувати обхід графа;
3. Якщо більше не існує інцидентних нових вершин, то повертаємося до попередньої вершини;
4. Якщо в результаті обходу графа, були пройдені всі вершини, то слід перевірити, чи має остання пройдена вершина ребро, що пов’язує її з першою вершиною. Якщо таке ребро є, то один із Гамільтонових циклів знайдено, інакше – ні. Далі ініціюємо повернення до попередньої вершини для продовження пошуку.

Так як даний алгоритм базується на основі алгоритму обходу графа в глибину, то результат його роботи можливо записати у вигляді дерева, в якому зберігаються всі можливі цикли проходження.

Даний алгоритм є найбільш універсальним із нині існуючих, але його швидкість виконання доволі велика і дорівнює *O((n - 2)!)* [4], що не дозволяє використовувати його ефективно у промислових масштабах і з великою кількістю вершин. Також, даний алгоритм може виконуватися ще довше, у випадках, якщо граф буде орієнтований або зважений.

Отже, розрахунки Гамільтонових циклів до сьогодні є одним із найпопулярніших напрямків дослідження графів, на який покладено великі надії, так як існуючі алгоритми аналізу Гамільтонових циклів повільні та обмежені і не дають змоги використовувати їх ефективно у промислових масштабах для сфер транспортування та комп’ютерних розробок.

Список літератури

1. *Пошук Гамільтонового циклу в неорієнтованому графі, URL: https://www.mathros.net.ua/poshuk-gamiltonovgo-cyklu-v-neorijentovanomu-grafi.html.*
2. *Efficient solution for finding Hamilton cycles in undirected graphs, URL:: https://springerplus.springeropen.com/articles/10.1186/s40064-016-2746-8.*
3. *Number of Hamiltonian cycle. URL: https://www.geeksforgeeks.org/number-of-hamiltonian-cycle/.*
4. *Hamiltonian path. URL: https://en.wikipedia.org/wiki/Hamiltonian_path.*

УДК 004.6

Остапчук Д.О. студент 1 курсу

спеціальності 122 «Комп’ютерні науки»

Науковий керівник:

Гончар В.М., асистент кафедри

інформаційних технологій

АЛГОРИТМИ ЗНАХОДЖЕННЯ МАКСИМАЛЬНОГО ПОКРИТТЯ РЕБРАМИ В ГРАФАХ

Донецький національний університет імені Василя Стуса, м. Вінниця

Максимальне покриття ребрами (англ. maximal edge covering) у графі - це множина ребер графа, які покривають кожну вершину графа і до цієї множини не можна додати жодного додаткового ребра, щоб збільшити кількість покритих вершин. Інакше кажучи, це множина ребер графа, яка покриває кожну вершину графа і має максимальну вагу.

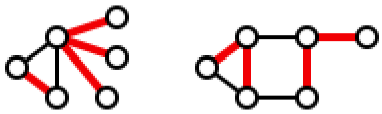

Рисунок 1. Приклади покриття ребрами графа

Існує багато алгоритмів для знаходження максимального покриття ребрами у графі. Вибір алгоритму залежить від конкретного випадку, а також від вимог до швидкодії та точності розв'язку. В цьому документі розглядаються лише декілька з них.

Модифікований алгоритм Борувки

Алгоритм Борувки будує мінімальне кістякове дерево графу, проте його можна модифікувати для знаходження максимального покриття ребрами. Для цього необхідно змінити порядок додавання ребер до кістякового дерева. Замість