

наріїв: для ідеї та натхнення (Midjourney), для експериментів і кастомізації (Stable Diffusion) або для швидкої та надійної генерації через хмару (DALL·E). З огляду на швидкий розвиток галузі, майбутні моделі, ймовірно, ще більше покращать якість і швидкість, а також розширять можливості керування процесом генерації.

Annotation. This article analyzes modern AI-based image generation technologies. We explain how models like DALL·E, Midjourney, and Stable Diffusion work (transformers, diffusion, neural architectures), including data structures (tensors, embeddings). A brief history from GANs to current diffusion models is given. In the practical part, we compare the three tools in terms of image quality, speed, accessibility, flexibility, licensing and open-source status. The strengths and weaknesses of each approach are highlighted. Keywords: image generation, artificial intelligence, generative models, DALL·E, Midjourney, Stable Diffusion, transformer, diffusion model, CLIP, open source.

Keywords: image generation, artificial intelligence, generative models, transformer, diffusion model.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. DALL·E: Creating images from text. *OpenAI*. 2021. URL: <https://openai.com/index/dall-e/>
2. Text-to-image Diffusion Models in Generative AI: A Survey / Chenshuang Zhang, Chaoning Zhang, M. Zhang, In So Kweon. 2024. URL: <https://arxiv.org/abs/2303.07909>
3. Rombach R., Blattmann A., Lorenz D. High-Resolution Image Synthesis with Latent Diffusion Models. *CVPR*. 2022. P. 10684–10695. URL: https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html
4. Stable Diffusion. *Wikipedia The Free Encyclopedia*. URL: https://en.wikipedia.org/wiki/Stable_Diffusion
5. Ticong L. Midjourney vs DALL-E: AI Art Tools Face-Off for 2025. *eWeek*. 2025. URL: <https://www.eweek.com/artificial-intelligence/midjourney-vs-dalle/>
6. Ticong L. Midjourney vs Stable Diffusion: 2025's Creative Clash. *eWeek*. 2025. URL: <https://www.eweek.com/artificial-intelligence/midjourney-vs-stable-diffusion/>
7. Self-Hosted License. *Stability AI*. URL: <https://stability.ai/license>
8. Ryan O'Connor. How DALL-E 2 Actually Works. *AssemblyAI*. 2023. URL: <https://www.assemblyai.com/blog/how-dall-e-2-actually-works>

УДК 004:005:51

АЛГОРИТМИ НА ГРАФАХ ТА ЇХ ЗАСТОСУВАННЯ

Д. О. Круцюк, Н. Р. Веселовська

Анотація. У публікації розглянуто ключові алгоритми на графах та їх фундаментальну роль у розв'язанні сучасних обчислювальних задач. Проаналізовано базові підходи до обробки графових структур, зокрема алгоритми пошуку в ширину (BFS) та глибину (DFS), а також методи знаходження найкоротших шляхів, зокрема алгоритм Дейкстри. Особливу увагу приділено практичному застосуванню цих алгоритмів у реальних системах: від GPS-навігації та оптимізації логістики до аналізу соціальних мереж.

Ключові слова: граф, алгоритм, оптимізація, пошук у ширину, Дейкстра, маршрутизація, аналіз мереж.

Вступ. У сучасному світі, що побудований на зв'язках – від соціальних мереж до глобальних логістичних маршрутів, алгоритми на графах стали ключовим інструментом для аналізу та оптимізації. Вони дозволяють розв'язувати складні задачі, де головну роль відіграє не сама сутність об'єктів, а структура відносин між ними. Саме ці алгоритми є в основі GPS-навігаторів, систем рекомендацій та інструментів для виявлення загроз у комп'ютерних мережах, які роблять наше життя більш ефективним.

Хоча теорія графів має глибоке математичне коріння, її практичне значення значно зросло з розвитком цифрових технологій. Сьогодні ці методи знаходять застосування у найрізноманітніших галузях: від біоінформатики, де графи моделюють взаємодії білків, до аналізу фінансових ринків. За допомогою графових алгоритмів можна не тільки обробляти величезні мережеві структури, але й знаходити в них приховані закономірності та найоптимальніші рішення.

У цій публікації ми розглянемо фундаментальні алгоритми на графах, їх основні принципи роботи та приклади застосування. Особливу увагу буде приділено тому, як методи обходу та пошуку найкоротшого шляху впливають на розвиток сучасних інформаційних систем.

Метою статті є висвітлення базових алгоритмів на графах та демонстрація їх ролі у розв'язанні практичних задач маршрутизації, аналізу мереж та оптимізації.

Основна частина. Алгоритми на графах становлять основу сучасних підходів до аналізу складних зв'язних структур, що відображають об'єкти та взаємодії між ними. Їх використовують для моделювання транспортних мереж, соціальних взаємозв'язків, комунікаційних систем або біологічних процесів. Граф як математична модель дозволяє описати об'єкти у вигляді вершин, а відносини між ними – у вигляді ребер, що забезпечує універсальність і гнучкість під час розв'язання аналітичних та оптимізаційних задач [1].

Базові принципи обробки графів реалізуються через три основні класи алгоритмів: обходу графа, пошуку найкоротших шляхів та оптимізації потоків. Алгоритм пошуку в ширину (BFS) послідовно відвідує вершини, що лежать на однаковій відстані від початкової точки, тоді як пошук у глибину (DFS) пересувається якомога далі вздовж одного шляху, перш ніж повернутись до розгалужень. Обидва методи широко використовуються під час розв'язання задач аналізу зв'язності, виявлення компонент графа та перевірки наявності циклів, а також під час моделювання шляхів передачі даних у комп'ютерних мережах [2].

Велике прикладне значення мають алгоритми пошуку найкоротших шляхів. Методи Дейкстри, Беллмана–Форда та A^* дозволяють віднайти найоптимальніші маршрути у графах із різними типами ваг ребер. Перший метод працює швидко за відсутності від'ємних ваг, другий – універсальніший, а A^* – застосовує евристику для пришвидшення обчислень. У транспортній логістиці та навігаційних системах саме ці алгоритми лежать в основі побудови маршрутів і прогнозування часу пересування [3]. Для опрацювання великих даних їх реалізують у паралельному виконанні в обчисленнях на GPU, хмарних серверах або кластерних середовищах [4].

До ключових завдань також належить побудова мінімального остового дерева – структури, що охоплює всі вершини графа, забезпечуючи мінімальну суму ваг ребер. Для цього використовують алгоритми Краскала та Прима, які є важливими у проектуванні мереж зв'язку, інженерних систем або енергорозподільних мереж [5]. Процес побудови відбувається у кілька кроків:

1. Обчислення ваг ребер та впорядкування їх за зростанням.
2. Покрокове додавання ребер у структуру без утворення циклів.
3. Перевірка цілісності дерева та видалення надлишкових зв'язків.
4. Оптимізація остаточного набору ребер з урахуванням вартості чи пропускної здатності.

Результатом стає основна структура, що мінімізує використання ресурсів і забезпечує стабільність мережі.

Значне практичне застосування мають алгоритми потоків у мережах. Методи Форда–Фалкерсона або Едмондса–Карпа дозволяють визначати максимальний потік у графі з обмеженнями на ребрах, розраховуючи ефективний розподіл ресурсів між вузлами [6]. Такі підходи активно впроваджуються у транспортних системах, промисловій логістиці, водопостачанні та енергетиці.

У сфері інформаційних технологій графові алгоритми стали основою пошукових і рекомендаційних систем. Класичний алгоритм PageRank, що аналізує вагу посилань між сторінками, став фундаментом роботи сучасних пошукових платформ. Удосконалені версії цього методу враховують не лише кількість, а й якість зв'язків, що дозволяє покращити релевантність результатів у глобальних інформаційних мережах [7].

У транспортній логістиці алгоритми на графах застосовуються для оптимізації маршрутів і розв'язання задачі комівояжера – визначення найкоротшого циклу, який охоплює заданий набір пунктів. Використання жадібних та евристичних алгоритмів (наприклад, Ant Colony Optimization або Tabu-Search) дає змогу зменшити витрати часу на пошук оптимального рішення у великих мережах [8].

У соціальних мережах графові методи дозволяють аналізувати структуру зв'язків між користувачами, ідентифікувати центри впливу та створювати спільноти. Алгоритми кластеризації Louvain та Girvan–Newman дають можливість розділити соціальні графи на підгрупи за рівнем інтенсивності взаємодій [9]. Подібні графові моделі використовуються також у біо-

інформатиці для аналізу взаємодій білків або генів, у фізиці – для моделювання енергетичних потоків, а в енергетиці – для балансування навантажень у розподільчих мережах [10].

Отже, алгоритми на графах поєднують математичну строгість та прикладну ефективність. Вони забезпечують можливість швидкої обробки великих обсягів мережевих даних, створюють умови для автоматизації складних процесів і залишаються фундаментальним інструментом інтелектуальних систем, штучного інтелекту та аналізу даних [11]. У розвитку сучасних інформаційних систем спостерігається тенденція до ускладнення графових структур, що вимагає підвищення швидкодії та оптимальності алгоритмів. Зі збільшенням обсягів даних загострюється проблема обчислювальної складності, тому велика увага приділяється розробленню наближених і паралельних методів обробки. Такі підходи дозволяють в обмежений час отримувати результат прийнятної точності без вичерпного перебору всіх можливих шляхів [12]. Сучасні архітектури обчислень – кластери, GPU-прискорювачі та хмарні середовища – забезпечують підтримку мільйонів вершин і ребер, що робить можливим аналіз надвеликих графів у реальному часі [13].

Дослідники продовжують активні пошуки шляхів інтеграції алгоритмів на графах із методами машинного навчання. Графові нейронні мережі (Graph Neural Networks, GNN) поєднують принципи класичних алгоритмів з адаптивністю штучного інтелекту, що дозволяє аналізувати динамічні мережі та робити прогнози для систем, які швидко змінюються [14]. Гібридні підходи з використанням GNN-моделей уже сьогодні застосовуються для передбачення станів транспортних потоків, аналізу графових баз знань та виявлення кіберзагроз [15].

Перспективним напрямом є використання квантових обчислень для розв'язання графових задач. Квантові алгоритми пошуку, зокрема модифікація алгоритму Гровера, потенційно можуть скоротити час знаходження оптимальних зв'язків у графах, а квантові підходи до задач розфарбовування та кластеризації дозволяють досліджувати нові межі обчислювальної ефективності [16].

Окрему увагу приділяють алгоритмам обробки динамічних графів – структур, що змінюються в часі. Такі алгоритми аналізують еволюцію зв'язків між вершинами, оновлюючи результати без повного перерахунку всієї мережі. Це вкрай актуально для систем моніторингу трафіку, соціальних комунікацій та потоків енергії. У таких алгоритмах поєднуються механізми кешування, імовірнісного прогнозування та локального оновлення компонент, що скорочує час обробки [17].

Застосування алгоритмів на графах дедалі більше інтегрується із суміжними технологіями – блокчейном, розподіленими базами даних, хмарними екосистемами. Моделі, побудовані на графових структурах, уже сьогодні використовуються для ефективного керування транзакціями, виявлення шахрайства та відстеження зв'язків у фінансових мережах. Графові бази даних типу *Neo4* або *TigerGraph* підтримують обчислення на рівні сотень мільйонів записів, що робить можливим оперативний аналіз пов'язаних даних навіть у високонавантажених середовищах [18].

Загалом процес еволюції графових алгоритмів можна умовно розділити на кілька етапів:

1. Створення базових методів обходу та пошуку (BFS, DFS).
2. Розроблення оптимізаційних підходів (Дейкстра, Краскал, Форда–Фалкерсона).
3. Поява спеціалізованих алгоритмів для паралельних і розподілених систем.
4. Інтеграція з машинним навчанням і штучним інтелектом.
5. Зародження квантових алгоритмів для обчислення на графах.

Кожен із цих етапів формує основу нової парадигми аналізу даних, а швидкість упровадження цих технологій визначає темпи розвитку сучасних обчислювальних наук.

Отже, сучасні алгоритми на графах не лише є ефективними інструментами для розв'язання прикладних завдань, а й перетворюються на методологічну основу побудови нових інтелектуальних систем. Їх подальший розвиток сприятиме створенню більш гнучких аналітичних платформ, здатних працювати з даними у реальному часі, масштабувати обробку на глобальному рівні та забезпечувати енергетичну й обчислювальну ефективність.

Висновки. Алгоритми на графах є одним із найважливіших інструментів сучасної інформатики та математики, оскільки саме вони забезпечують ефективне моделювання, аналіз і оптимізацію взаємопов'язаних систем. Вони лежать в основі транспортної логістики, соціальних мереж, енергетичних систем, біоінформатики та інформаційної безпеки. Розвиток графових методів дозволяє розв'язувати дедалі складніші завдання маршрутизації, аналізу структур зв'язків і керування потоками ресурсів у реальному часі.

Подальше вдосконалення алгоритмів на графах сприяє підвищенню продуктивності обчислювальних систем та створенню більш адаптивних і інтелектуальних технологій. Використання паралельних обчислень, евристичних методів і штучного інтелекту підсилює потенціал цих алгоритмів для застосування у великих мережах та системах Big Data. Отже, алгоритми на графах залишаються невід'ємною складовою сучасного технологічного розвитку, адже забезпечують аналітичну основу для оптимізації процесів, автоматизації систем і ухвалення ефективних рішень у найрізноманітніших галузях людської діяльності.

Abstract. This paper summarizes the importance of graph algorithms as a fundamental tool of modern computer science and applied mathematics. Graph-based methods provide efficient modeling, analysis, and optimization of interconnected systems. They serve as the foundation for transport logistics, communication networks, bioinformatics, energy optimization, and cybersecurity. The continuous improvement of graph algorithms increases the efficiency and adaptability of computational processes, enabling intelligent decision-making in dynamic network environments. The integration of heuristic optimization, parallel processing, and artificial intelligence expands their potential in solving large-scale analytical and engineering problems.

Keywords: graph algorithms, optimization, networks, route finding, data analysis, automation, artificial intelligence.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Introduction to Algorithms / T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Cambridge: MIT Press, 2022. 1312 p. URL: <https://mitpress.mit.edu/9780262046305>
2. Kleinberg J., Tardos E. Algorithm Design. Boston: Pearson Education, 2021. 749 p.
3. Sedgewick R., Wayne K. Algorithms. Boston: Addison Wesley, 2022. 976 p. URL: <https://algs4.cs.princeton.edu>
4. Frieze A., Karonski M. Introduction to Random Graphs. Oxford: Oxford University Press, 2023. 456 p. URL: <https://www.math.cmu.edu/~af1p/BOOK.pdf>
5. Hamilton W. L. Graph Representation Learning. Cambridge: Cambridge University Press, 2023. 420 p. URL: https://www.cs.mcgill.ca/~wlh/grl_book/
6. Graph Neural Networks: A Review of Methods and Applications / Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P. S. Yu. New York: Elsevier, 2023. 530 p. URL: <https://arxiv.org/abs/2001.00595>
7. Burachievsky O. V. Теорія графів та її застосування в інформатиці. Київ: КНЕУ, 2023. 278 с.
8. Козак С. М. Евристичні методи та алгоритми оптимізації графових структур. Київ: КПІ ім. І. Сікорського, 2022. 310 с.
9. Вороніна Л. Ю. Аналіз соціальних графів у цифрових системах. Одеса: Астропринт, 2024. 254 с.
10. Хоміч Ю. О. Хмарні обчислення та графові моделі аналізу даних. Київ: КНЕУ, 2022. 321 с.
11. Graph Algorithms Explained. *GeeksforGeeks*. 2024. URL: <https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>
12. Applications of Graph Theory in Computer Science / H. C. Martin, W. R. V. Sampaio, J. C. C. Verancio, R. S. Souza. *Revista Indicação Geográfica e Inovação*. 2023. № 7(2). P. 2069–2078. URL: <https://www.researchgate.net/publication/370515232>
13. Graph Algorithms and Their Applications in Big Data. *ScienceDirect*. 2023. URL: <https://www.sciencedirect.com/topics/computer-science/graph-algorithm>
14. Shortest Path Algorithms: Dijkstra, Bellman-Ford A. *Khan Academy*. 2024. URL: <https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation>
15. CS224W: Machine Learning with Graphs. *CS224W Course Materials*. 2024. URL: <https://web.stanford.edu/class/cs224w/>
16. Алгоритми на графах у Python. *IT-Info*. 2024. URL: <https://itinfo.ua/articles/python-graph-algorithms>
17. Graph Neural Networks: A Review of Methods and Applications Applications / Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P. S. Yu. *ScienceDirect*. 2020. URL: <https://www.sciencedirect.com/science/article/pii/S2666651021000012>
18. Kumar S., Patel N. Approximation Algorithms and Heuristics for Large-Scale Graph Problems. London: Academic Press, 2023. 388 p.