

ДОСЛІДЖЕННЯ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ АЛГОРИТМІВ ЗНАХОДЖЕННЯ ОПТИМАЛЬНОГО ШЛЯХУ НА ГЕОГРАФІЧНИХ КАРТАХ

Вінницький національний технічний університет

Анотація

Розглянуто алгоритми, що дозволяють ландшафту надавати один з можливих варіантів шляху з однієї точки в іншу на географічній карті з урахуванням особливостей прохідності місцевості. Описано методи, які умовно можна розділити на наступні класи: алгоритми пошуку найкоротшого шляху (Дейкстри); алгоритми пошуку субоптимального шляху (A^* і його модифікації, зокрема Θ^*).

Ключові слова: пошук шляху, алгоритм Дейкстри, алгоритм A^* .

Abstract

The algorithms that allow the landscape to provide one of the possible variants of the way from one point to another on a geographical map, taking into account the peculiarities of the pertinence of the terrain, are considered. Described methods that can be conventionally divided into the following classes: algorithms for finding the shortest path (Deikstri); algorithms for finding the sub-optimal path (A^* and its modifications, in particular Θ^*).

Keywords: path search, Dijkstra algorithm, A^* algorithm.

При створенні симуляторів, що мають перемішувати різні типи об'єктів по великих територіях з урахуванням поточної обстановки, виникають проблеми з вибором алгоритму пошуку оптимального шляху, так як на його використання накладаються обмеження, викликані такими факторами як великий обсяг даних реальних карт місцевості; складність уявлення ландшафту та ін. оптимальним рішенням може виявитися як пряма, так і ламана лінія.

Головна проблема задачі пошуку шляху заключається в тому, що не існує будь-якого універсального алгоритму її вирішення.

Граф представляє собою сукупність не пустої множини вершин і ребер (наборів пар вершин). Дві вершини на графі сусідні, якщо вони з'єднуються спільним ребром. Шлях в неорієнтованому графі представляє собою послідовність вершин $P = (v_1, v_2, \dots, v_n) \in V \times V \times \dots \times V$, таких, що v_i суміжна з v_{i+1} для $1 \leq i < n$. Такий шлях P називається шляхом довжиною n з вершини v_1 в v_n (i вказує на номер вершини шляху і не має ніякого відношення до нумерації вершин на графі) [1].

Для вирішення вказаної задачі можна використовувати алгоритм Дейкстри. Він знаходить найкоротшу відстань від однієї вершини графа до всіх інших.

Складність алгоритму складає $O(n^2 + m)$, де n – кількість вершин і m – кількість ребер. Недоліком алгоритму є те, що він не завжди працює з ребрами при від'ємному значенні ребер [2].

Алгоритм Флойда-Уоршелла служить для визначення найкоротших шляхів між усіма парами вершин графа.

Метод Флойда базується на тому факті, що в графі з позитивними вагами будь-якого неелементарного ребра (що містить більше 1 ребра), короткий шлях складається з інших найкоротших шляхів [3].

Цей алгоритм більш загальний порівняно з алгоритмом Дейкстри, так як він знаходить найкоротші шляхи між будь-якими двома вершинами графа [4].

Перегавою алгоритму є його простота реалізації, слабкість – в складності алгоритму. Вона дорівнює $O(n^3)$, де n – кількість вершин графа.

Перебірні алгоритми по своїй суті є алгоритмами пошуку, як правило, пошуку оптимального рішення. При цьому рішення конструюється поступово. В цьому випадку звичайно говорять про перебір вершин дерева варіантів. Для прискорення перебору з поверненням обчислень завжди намагаються організувати так, щоб була можливість якомога відмінити недоречні варіанти [5].

Алгоритм пошуку A^* належить до евристичних алгоритмів пошуку. Він використовується для пошуку найкоротшого шляху між двома вершинами графу з додатними вагами ребер. Алгоритм

використовує допоміжну функцію (евристику), аби скеровувати напрям пошуку та скорочувати його тривалість. Основним недоліком алгоритму A^* є потреба в пам'яті для збереження всіх відомих та досліджених вершин. Через це алгоритм A^* непридатний для багатьох задач.

Однією з основних проблем алгоритму A^* є те, що одержані за його допомогою шляхи не виглядають реалістичними. Ця проблема вирішується в алгоритмі Theta*. Ключовий момент, який відрізняє ці два алгоритми, в тому, що Theta* дозволяє в якості «предка» для кожної вершини вибрати будь-яку вершину, на відміну від A^* , де «предком» може бути тільки найближчий видимий сусід. Якщо потрібно отримати найбільш реалістичний субоптимальний шлях, рекомендується використовувати алгоритм Theta* [6].

Було розглянуто алгоритми, які використовуються для знаходження оптимального шляху на географічних картах, а саме: алгоритм Дейкстри, алгоритм Флойда-Уоршелла, алгоритм бектрекінгу, алгоритм A^* та алгоритм Theta*. Також було проведено короткий огляд перебірних алгоритмів, але вони не підходять для вирішення поставленої задачі через великі затрати пам'яті.

У загальному випадку необхідно будувати систему алгоритмів. Так як витрати на звернення на ландшафт являються критичними, то будемо використовувати наступну комбінацію алгоритмів:

- алгоритм A^* для отримання маршруту;
- видалимо точки, що лежать на одній прямій;
- застосуємо алгоритм перевірки наявності шляху по прямій.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Cook W. J. In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation. Princeton University Press, 2012. P. 19–39.
2. Applegate D.L., Bixby R.E., Chvátal V., Cook W.J. The Traveling Salesman Problem. Princeton University Press, 2007. P. 44–52.
3. Левитин А. Алгоритмы. Введение в разработку и анализ. Вильямс, 2006. P. 160.
4. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. Москва, Вильямс, 2005.
5. Botea A., Muller M., Schaeffer J. Near Optimal Hierarchical Path-Finding. Journal of Game Development, 2004, vol. 1, issue 1, pp. 7–28.
6. Daniel K., Nash A., Koenig S., Felner A. Theta*: Any-Angle Path Planning on Grids. Journal of Artificial Intelligence Research, 2010, vol. 39, pp. 533–579.

Решетник Вікторія Романівна — студентка групи ІКН-146, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця.

Науковий керівник – **Петришин Сергій Іванович**, – асистент кафедри комп'ютерних наук ВНТУ, Вінницький національний технічний університет, м. Вінниця.

Reshetnyk Viktoria R. — Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.

Supervisor– **Sergiy I. Petrishyn** — assistant of the Computer Sciences Chair, Vinnytsia National Technical University, Vinnytsia.