



УДК 004.02

DETERMINE THE DIRECTION OF IMPROVEMENT OF EXISTING ALGORITHMS FOR LAYING THE ROUTE TO THE DESTINATION POINT**ВИЗНАЧЕННЯ НАПРЯМКУ УДОСКОНАЛЕННЯ ІСНУЮЧИХ АЛГОРИТМІВ ПРОКЛАДАННЯ МАРШРУТУ ДО ТОЧКИ ПРИЗНАЧЕННЯ**

Melnyk Y.V. / Мельник Ю.В.

Ivano-Frankivsk National Technical University of Oil and Gas,

Ivano-Frankivsk, Karpatska 15, 76019

Івано-Франківський національний технічний університет нафти і газу,

Івано-Франківськ, вул. Карпатська 15, 76019

Анотація. Були досліджені існуючі алгоритми для прокладання маршрутів. Встановлено, що вони мають ряд недоліків, які не дозволяють їм виконувати покладені на них функції у відповідності з вимогами сьогодення. В результаті проведених досліджень встановлено, що перспективним напрямком буде створення алгоритму пошуку оптимального шляху на базі алгоритму A^* , який буде використовувати його сильні сторони та усувати його слабкі сторони за рахунок використання елементів інших існуючих алгоритмів.

Ключові слова: пошук шляху, алгоритм Дейкстри, алгоритм Флойда, алгоритм Беллмана-Форда, алгоритм A^* .

Вступ.

Обчислення шляху на множині геометричних об'єктів – це фундаментальна задача обчислювальної геометрії, що має широке застосування. Задача полягає у пошуку шляху між двома заданими точками, що оминає задані перешкоди [1].

В даний час в різних сферах діяльності людини виникає необхідність прокладання оптимального маршруту (зокрема, в робототехніці, плануванні руху, навігації, тощо). В зв'язку з вищевказаним, актуальним завданням на даний час є вдосконалення існуючих та розробка нових алгоритмів прокладання маршруту для досягнення економії часу.

Основний текст.

Все різноманіття алгоритмів для визначення маршруту, за яким можна потрапити з однієї точки в іншу, ділиться на дві групи:

- алгоритми, що дозволяють визначити оптимальний шлях [2];
- алгоритми, що дозволяють знайти субоптимальний шлях [3-5].

Алгоритми першої групи потребують повного дослідження простору станів задачі. Найпростішим способом пошуку оптимального шляху є повний перебір всіх можливих маршрутів. Це гарантує знаходження оптимального шляху. Але на практиці частіше за все такий спосіб неможливо використати із-за надмірних накладних витрат, оскільки він потребує зберігання в пам'яті усієї досліджуваної області.

Прикладом алгоритмів пошуку субоптимальних шляхів є евристичні алгоритми, які на кожному кроці наближаються до кінцевої точки. Однак при пошуку одного з близьких до оптимального шляху слід враховувати, що спочатку важко точно передбачити, який саме варіант буде обраний. У цьому



випадку можна використовувати різні алгоритми при виборі напрямку на кожному кроці. Найбільш раціональним у такій ситуації є розробка нових алгоритмів для різних класів задач на базі алгоритмів визначення оптимального шляху із застосуванням евристики, що враховують специфіку даної проблемної області.

Можна узагальнити три найбільш ефективні алгоритми на основі яких розробляються похідні [6-7]:

- алгоритм Дейкстри (використовується для знаходження оптимального маршруту між двома вершинами);
- алгоритм Флойда (для знаходження оптимального маршруту між усіма парами вершин);
- алгоритм Беллмана-Форда (для знаходження найкоротшого шляху від однієї вершини графа до всіх інших).

Алгоритм Дейкстри названий за іменем творця і був розроблений у 1959 році. У процесі виконання алгоритм перевіряє кожен з вершин графа, і знаходить найкоротший шлях до початкової вершини. Стандартна реалізація працює на зваженому графові – графові, у якого кожен шлях має вагу, тобто “вартість”, яку потрібно буде “заплатити”, щоб перейти по цьому ребру. У стандартній реалізації ваги не являються негативними. Цей алгоритм є найпростішим алгоритмом з досліджуваних. Він добре виконується в графах з невеликою кількістю вершин. З огляду на структуру схеми міста програма, написана з використанням алгоритму Дейкстри буде виконуватися повільно, а розподіл пам'яті не буде раціональним.

Алгоритм Флойда – призначений для вирішення завдання пошуку всіх найкоротших шляхів на графі. Для заданого орієнтованого зваженого графа алгоритм знаходить найкоротші відстані між усіма парами вершин за час $O(n^3)$. Алгоритм застосовуємо до графів з довільними, в тому числі з негативними, вагами. Таким чином, він є більш загальним в порівнянні з алгоритмом Дейкстри, який не працює з негативною вагою ребер. Також алгоритм розпізнає наявність негативних циклів. Але він містить три вкладених цикла, тобто має кубічну складність, тому при використанні цього алгоритму буде використовуватися великий обсяг пам'яті, що є не раціональним рішенням.

Алгоритм Беллмана-Форда призначений для вирішення завдання пошуку найкоротшого шляху на графі. Для заданого орієнтованого зваженого графа алгоритм знаходить найкоротші відстані від виділеної вершини-джерела до всіх інших вершин графа. Алгоритм Беллмана-Форда масштабує гірше інших алгоритмів вирішення зазначеного завдання (складність $O(|V||E|)$ проти $O(|E|+|V|\ln(|V|))$ у алгоритму Дейкстри), проте його відмінною рисою є можливість застосування до графів з довільними, в тому числі негативними, вагами. Алгоритм використовує повний перебір всіх вершин графа, що призведе до великої втрати часу і займе більший обсяг пам'яті.

Для задачі знаходження пошуку шляху між вершинами графів, де граф – це карта міста, а вершини – початок і кінець вулиць, також існують похідні алгоритми. Одним з таких алгоритмів є A^* (A “із зірочкою”). Даний алгоритм вперше описаний 1968 р. Пітером Хартом, Нільсом Нільсоном і Бертрамом



Рафаелем. Алгоритм A^* є розширенням алгоритму Дейкстри, в якому прискорення роботи досягається за рахунок евристики – при розгляді кожної окремої вершини перехід робиться в ту сусідню вершину, передбачуваний шлях з якої до шуканої вершини найкоротший. При цьому є безліч різних методів підрахунку довжини передбачуваного шляху з вершини. Проте у нього є теж багато недоліків. На карті міста в залежності від напрямку руху на перехресті може знадобитися різна кількість часу на його подолання. Одним з недоліків даного алгоритму в такому випадку є неможливість враховувати напрямки руху на перехресті.

Зростання виробництва мікропроцесорних пристроїв, зменшення їх розмірів, собівартості, підвищення продуктивності і надійності створили умови для покращення даних алгоритмів та їх ефективного використання в різних автоматизованих системах [8-15].

Під час дослідження було визначено, що в існуючих алгоритмах пошуку шляху (Дейкстри, Беллмана-Форда, Флойда) відрізняються сильні і слабкі сторони. Тому, доцільно створити новий похідний алгоритм який буде синтезом вищевказаних алгоритмів (двох, або більше), завдяки чому вдасться усунути слабкі сторони і використати сильні існуючих алгоритмів.

Висновки.

Були досліджені існуючі алгоритми для прокладання маршрутів. Встановлено, що вони мають ряд недоліків, які не дозволяють їм виконувати покладені на них функції у відповідності з вимогами сьогодення.

В результаті проведених досліджень встановлено, що перспективним напрямком буде створення алгоритму пошуку оптимального шляху на базі алгоритму A^* , який буде використовувати його сильні сторони та усувати його слабкі сторони за рахунок використання елементів інших існуючих алгоритмів.

Література:

1. E. W. Dijkstra. A note on two problems in connexion with graphs. // Numerische Mathematik. V.1 (1959), P.269-271
2. Nash A. Any-Angle Path Planning. Dis. ... Doctor of Philosophy (Computer Science). University of South California. August 2012.
3. Botea A., Muller M., Schaeffer J. Near Optimal Hierarchical Path-Finding. Journal of Game Development, 2004, vol. 1, issue 1, pp. 7–28.
4. 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.
5. Variants of A^* , Amit Patel's Home Page. <http://theory.stanford.edu/~amitp/GameProgramming/Variations.html> (дата обращения 16.04.2013).
6. Ананий В. Левитин Глава 8. Динамическое программирование: Алгоритм Флойда поиска кратчайших путей между всеми парами вершин // Introduction to The Design and Analysis of Algorithms. - М.: Вильямс, 2006. – С.349-353.
7. Кощєєв, І.С. Алгоритми рішення задачі маршрутизації транспорту: дис. канд. тех. наук / І.С. Кощєєв. - Уфа, 2015. - 118 с.
8. Бабчук С.М. Визначення шляхів підвищення гнучкості автоматизованих



систем управління технологічними процесами. Матеріали 17-ї Міжнародної науково-технічної конференції “Вимірювальна та обчислювальна техніка в технологічних процесах”. – 2017. – С. 159

9. Бабчук С.М. Класифікація спеціалізованих комп'ютерних мереж // Проблеми управління і інформатики. – 2016. – №5. – С. 97-103.

10. Бабчук С.М. Алгоритм вибору спеціалізованої безпроводної цифрової мережі // Научные труды SWorld : международное периодическое научное издание. – Иваново : Научный мир, 2017. – Вып. 48. - Т. 1. - С.8-13.

11. Бабчук С. М. Синтез бази знань “спеціалізовані комп'ютерні мережі” для об'єктів нафтогазового комплексу / С. М. Бабчук // Східно-європейський журнал передових технологій. – 2014. – №2. – С. 14-18.

12. Babchuk, S. Classification of Specialized Computer Networks // Journal of Automation and Information Sciences. – 2016. – Vol. 48. – P. 57-64.

13. Бабчук С. М. Класифікація спеціалізованих комп'ютерних мереж для транспортних засобів [Текст] / С. М. Бабчук // Наукові труды SWorld. – 2015. – Т. 4. – №3. – С. 39-43.

14. Мельник Ю.В. Аналіз методів і систем прогнозування сходження лавин та визначення напрямку їх подальшого вдосконалення // Научные труды SWorld. –Иваново : Научный мир, 2017. – Вып. 46. - Т. 1. - С.60-65.

References:

1. E. W. Dijkstra. A note on two problems in connexion with graphs. // Numerische Mathematik. V.1 (1959), P.269-271

2. Nash A. Any-Angle Path Planning. Dis. Doctor of Philosophy (Computer Science).University of South California. August 2012.

3. Botea A., Muller M., Schaeffer J. Near Optimal Hierarchical Path-Finding. Journal of Game Development, 2004, vol. 1, issue 1, pp. 7–28.

4. 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.

5. Variants of A*, Amit Patel's Home Page. <http://theory.stanford.edu/~amitp/GameProgramming/Variations.html> (дата обращения 16.04.2013).

6. Ananii V. Levitin (2006). Chapter 8. Dynamic programming: Floyd's algorithm for finding the shortest paths between all pairs of vertices, C.349 —353.

7. Koshcheev I.S. (2015). Algorithms for solving the problem of transport routing: Dis. Dis. ... Cand. Those. Sciences / I.S. Koshcheev, 118.

8. Babchuk S.M. (2017). Vyznachennya shlyaxiv pidvyshhennya gnuchkosti avtomatyzovanyx system upravlinnya texnologichnymy procesamy [Determination of ways to increase the flexibility of automated control systems of technological processes]. Materialy 17 Mizhnarodnoyi naukovo-texnichnoyi konferenciyi “Vymiryuvalna ta obchyslyuvalna texnika v texnologichnyx procesax” [Materials of the 17th International Scientific and Technical Conference "Measuring and Computing Techniques in Technological Processes"], pp. 159.

9. Babchuk S.M. (2016). Klassyfikatsiya spetsializovannykh kompyuternykh merezh [Classification of this specialist networks]. Problemy upravlinnya i informatyky [Problems of control and informatics], no. 5, pp. 97-103.

10. Babchuk S.M. (2017). Alhorytm vyboru spetsializovanoi bezprovodnoi tsyfrovoyi merezhi [Algorithm of selection specialized wireless digital network], Naukovi trudy SWorld [Scientific labors SWorld], no. 48, vol. 1, pp. 8-13

11. Babchuk, S. Classification of Specialized Computer Networks // Journal of Automation and Information Sciences. – 2016. – Vol. 48. – P. 57-64.



12. Babchuk, S. Classification of Specialized Computer Networks. Journal of Automation and Information Sciences. – 2016. – Vol. 48. – P. 57-64.

13. Babchuk S. M. (2015). Klasyfikatsiya spetsializovanykh komp'yuternykh merezh dlya transportnykh zasobiv [Classification specialized networks for vehicles]. Naukovi trudy SWorld [Scientific labors SWorld], no. 3, pp. 39-43.

14. Melnyk, Y. V. (2017). Analysis of existing methods and systems for forecasting the ascent of lava and determining the direction of their further improvement. Scientific Papers SWorld., 46rd ser., P. 60-65.

Abstract. Existing algorithms for laying routes were explored. It is established that they have a number of shortcomings that do not allow them to perform the functions assigned to them in accordance with the requirements of the present. As a result of the research, it was found that a promising direction will be the creation of an algorithm for finding the optimal path based on the A* algorithm, which will use its strengths and eliminate its weaknesses by using elements of other existing algorithms.

Key words: search path, Dijkstra algorithm, Floyd algorithm, Bellman-Ford algorithm, A* algorithm.

Науковий керівник: к.т.н., доц. Бабчук С.М.

Стаття відправлена: 12.10.2018 р.

© Мельник Ю.В