УДК 514.18+ 519.852.3+681.5

В.В. Ванін, д.т.н., професор
О.В. Залевська, к.т.н.
П.М. Яблонський, к.т.н., доцент

ЗАСТОСУВАННЯ ТЕОРІЇ ГРАФІВ ДЛЯ УДОСКОНАЛЕННЯ ТА ВІЗУАЛІЗАЦІЇ АЛГОРИТМУ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ В МАТЕМАТИЧНІЙ МОДЕЛІ ВІДЕО ІГРИ

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Проектування складних систем, вивчення їх властивостей і управління ними потребують розробки математичної моделі. Дослідження характеристик системи за допомогою математичних моделей часто є єдиним способом вивчення складних систем і вирішення найважливіших практичних завдань.

Застосування теорії графів для побудови математичної моделі обумовлено можливістю опису ними широкого класу об'єктів та процесів. Це дає можливість автоматизації пошуку оптимального маршруту, досяжності цілі, мережевого планування. Однією з сфер такого застосування є відеоігри, які потребують розробки інтерфейсу, візуалізації дій користувачів та персонажів, розробки алгоритму та прорахунок дій рухомих персонажів в процесі гри. Широко застосовуються ігри у жанрі Roguelike, особливостями яких є випадкове, створення рівнів, покроковий ігровий процес, плиткова (тайлова) або ASCII-графіка. [1].

В роботі запропоновано узагальнення алгоритму пошуку найкоротшого шляху А при динамічній кінцевій цілі за допомогою методу проходу всіх вершин зваженого неорієнтованого графу, а також реалізація цього алгоритму для візуалізації.*

Ключові слова: теорія графів, зважений граф, неорієнтований граф, візуалізація роботи алгоритму, комп'ютерне моделювання.

Постановка проблеми. При розробці відео ігор виникають такі задачі, як недосконалий алгоритми опису:

- пошуку найкоротшого шляху А* при динамічній кінцевій цілі;
- всіх можливих варіантів поведінки персонажів (неспроможність пройти лабіринти, обійти перепони на шляху та інше);
- взаємодії персонажа та гравців з картою гри.

В зв'язку з цим виникає потреба в удосконаленні існуючих алгоритмів пошуку найкоротшого шляху при динамічній кінцевій цілі та їх візуалізація.

Цілі статті. Удосконалення алгоритму А* з усуненням недоліків алгоритму за допомогою методу проходу всіх вершин зваженого неорієнтованого графу.

Аналіз останніх досліджень і публікацій. Всі відомі алгоритми розв'язку поставленої задачі можна розділити на 3 групи, що базуються на [2]:

• алгоритмі Дейкстри (використовується для знаходження оптимального маршруту між двома вершинами);

• алгоритмі Форда-Бельманна (для знаходження оптимального маршруту між усіма парами вершин);

• алгоритмі Флойда-Уоршалла (для знаходження найкоротшого шляху між парами вершин).

Алгоритм A* використовує допоміжну еврістичну функцію для спрямування напрямку пошуку, що суттєво скорочує час роботи алгоритму Дейкстри. Алгоритм A* покроково перебирає всі вершини графу, що забезпечує знаходження оптимального розв'язку при умові його існування [3].За алгоритмом вершини поділяються на три класи в залежності від знайденого шляху до них: невідомі вершини, відомі вершини та повністю досліджені. Відомими вважаються вершини до яких є відомим шлях, а до досліджених вершин відносять вершини до яких відомий найкоротший шлях [4].

Основна частина. На початку алгоритму всі вершини вважаються невідомими. Спершу розглядаються вузли, суміжні з початковим; вибирається той з них, який має мінімальне значення f(x), після чого цей вузол розкривається та вважається відомим. На кожному етапі алгоритм оперує з безліччю шляхів з початкової вершини до всіх ще не розкритих вершин графа , що розміщується в черзі з пріоритетом. Пріоритет шляху визначається за значенням функціоналу $f(x)=g(x)+h(x)$, де g(x) - це вага вершини, а вартість шляху від початкової вершини, а h(x) – евристична функція, що оцінює вартість шляху від вершини x до кінцевої вершини. Під вагою вершини розуміємо кількість можливих шляхів від початкової вершини до даної [5]. Приклад використання цих алгоритмів наведено на рис.1.

Всі наведені вище алгоритми працюють з невеликою кількістю вершин. При зростанні кількості вершин застосування даних алгоритмів не є доцільно, оскільки час їхньої дії зростає в експоненціальній залежності.

Одночасне врахування алгоритмом A* відстані та оцінки шляху між вершинами обумовлює його використання, як основу, в поставленій задачі.

При знаходження шляху між вершинами алгоритм A* обходить мінімальну кількість вершин, так як він працює з «оптимістичною» оцінкою шляху через вершину. Оптимістичною в тому сенсі, що при проходженні вершин він забезпечує те, що реальна вартість результату буде не менше заданого значення евристичної функції. Евристичний функціонал задаємо як евклідову відстань між двома точками на прямій.

Для реалізації алгоритму будемо використовувати інформований пошук. Інформований пошук – це стратегія пошуку рішень в просторі визначених станів, що відносяться до конкретного завдання [5].

Інформовані методи забезпечують більш ефективний пошук в порівнянні з неінформованими методами.

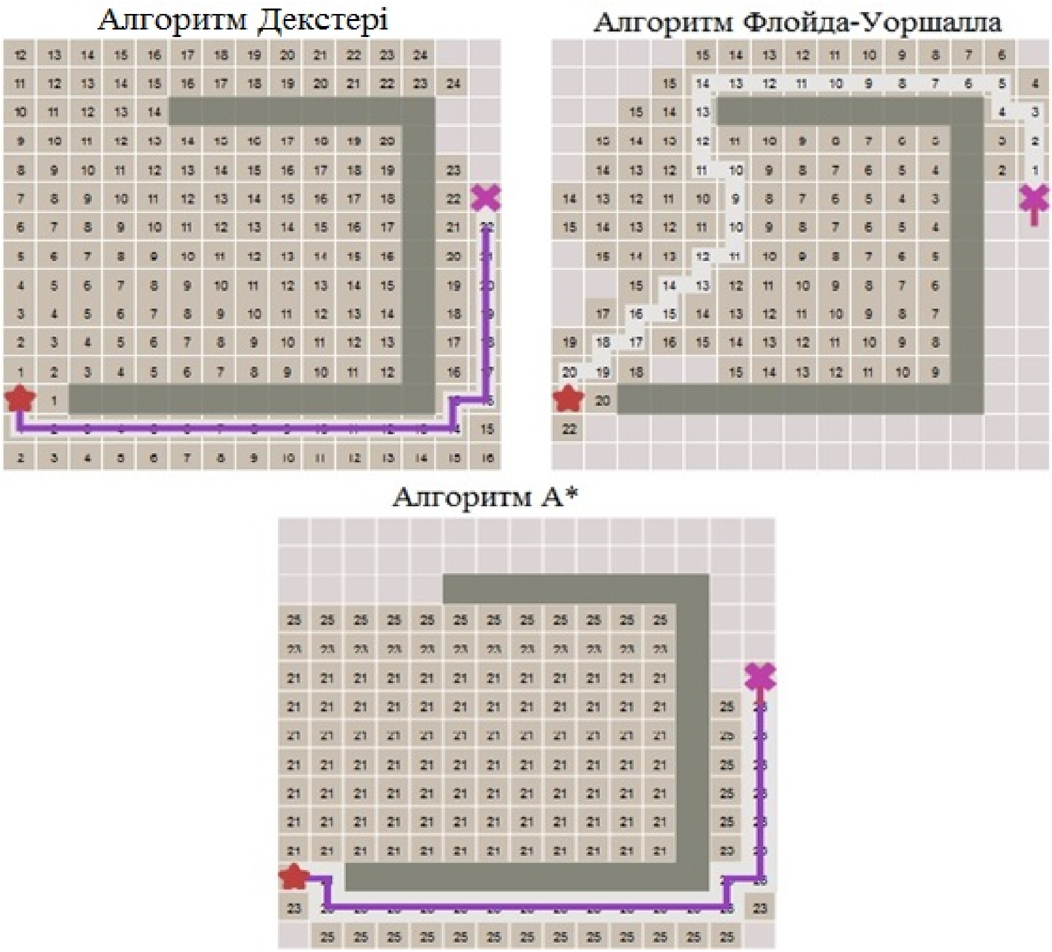

Рис. 1. Приклад роботи алгоритмів для пошуку найкоротшого шляху між вершинами зваженого неорієнтованого графу

Інформацію про конкретну задачу будемо заносити до евристичного функціоналу, за допомогою якого, на кожному кроці будемо приймати рішення про продовження алгоритму чи його завершення.

При роботі з вершинами дані зберігаємо у вигляді стеку та використовуємо його в подальших дослідженнях. Використання стеку замість стандартної черги в алгоритмі А* дозволяє швидше звернутись до будь-якої вершини в процесі дослідження.

Удосконалений алгоритм має включати в себе наступні кроки:

1. Створення двох списків вершин – ті що очікують розгляду і вже досліджені. У перший список додається вершина старту, список досліджених вершин на момент старту алгоритму є порожнім.
2. Для розрахунку кожної вершини знаходиться значення функціоналу $f(x)=g(x)+h(x)$, де $g(x)$ - відстань від старту до вершини, $h(x)$ - приблизну відстань від вершини до мети. Кожна вершина зберігає посилання на точку, з якої в неї прийшли.

3. Зі списку вершин на розгляд вибирається вершина з найменшим значенням функціоналу f(x). Позначимо її X.
4. Якщо X - мета, то маршрут знайдено.
5. Переносимо X зі списку тих що чекають на розгляд у список вже досліджених.
6. Для кожної з вершини X суміжні вершини позначимо як У. Покроково розглядаємо вершини У:
   6.1. Якщо Y вже знаходиться в списку досліджуваних вершин - пропускаємо її.
   6.2. Якщо Y ще немає в списку вершин, що очікують дослідження - додаємо до її та враховуємо зв’язок з вершиною X через змінні Y.g(x) та Y.h(x).
   6.3. Якщо ж Y є в списку на розгляд – перевіряємо: якщо X. g(x) + відстань від X до Y < Y.g(x), ми прийшли в точку Y коротшим шляхом. Тоді замінюємо Y.g(x) на X. g(x) + відстань від X до Y, а точку, з якої прийшли в Y, на X
7. Маршруту не існує, якщо список вершин на дослідження є порожнім, а поставлена мета не досягнута.

При створенні відповідного графа відстань між вершинами вважаємо нескінченною, а самі вершини ізольованими. За вагу вершини приймаємо номер ітерації, що забезпечує доступ до вершини. Граф починаємо розглядаємо починаючи з лівої верхньої вершини в напрямку правої нижньої вершини. Для кожної вершини виконуємо перевірку чи є вона кінцевою вершиною або перешкодою.

Висновки та перспективи. Удосконалено алгоритм A*, так що існує можливість дослідження динаміки кінцевої вершини з урахуванням поведінки всіх персонажів та забезпеченням взаємодії гравців з картою гри. Наведений алгоритм може бути використаний при розробці програмного забезпечення для робототехніки.

Література

1. *Stuart Russell, Peter Norvig* Artificial Intelligence: A Modern Approach, 2004, Prentice Hall, ISBN 3-8273-7089-2
2. *P. E. Hart, N. J. Nilsson, B. Raphael* A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems Science and Cybernetics SSC4 (2), pp. 100—107, 1968.
3. *P. E. Hart, N. J. Nilsson, B. Raphael* Correction to «A Formal Basis for the Heuristic Determination of Minimum Cost Paths», SIGART Newsletter, 37, pp. 28-29, 1972.
4. *Anthony Stentz* Optimal and Efficient Path Planning for Partially-Known Environments, Original D* paper, ICRA International Conference on Robotics and Automation, 1994.

5. *Amid P.* Red Blob Games. Introduction to the A* Algorithm [Електронний ресурс] / Petel Amid – Режим доступу до ресурсу: https://www.redblobgames.com/pathfinding/a-star/introduction.html.

ПРИМЕНЕНИЕ ТЕОРИИ ГРАФОВ ДЛЯ СОВЕРШЕНСТВОВАНИЕ И ВИЗУАЛИЗАЦИИ АЛГОРИТМА ПОИСКА КРАТЧАЙШЕГО ПУТИ В МАТЕМАТИЧЕСКИХ МОДЕЛЯХ ВИДЕО ИГРЫ

В.В.Ванин, О.В. Залевская, П.М. Яблонский
Национальный технический университет Украины «Киевский политехнический институт имени Игоря Сикорского»

Проектирования сложных систем, изучение их свойств и управления ими требуют разработки математической модели. Исследование характеристик системы с помощью математических моделей часто является единственным способом изучения сложных систем и решения важнейших практических задач.

Применение теории графов для построения математической модели обусловлено возможностью описания ними широкого класса объектов и процессов. Это дает возможность автоматизации поиска оптимального маршрута, досягаемости цели, сетевого планирования. Одной из сфер такого применения является видеоигры, которые требуют разработки интерфейса, визуализации действий пользователей и персонажей, разработки алгоритма и просчет действий подвижных персонажей в процессе игры. Часто применяются игры в жанре Roguelike, особенностями которых является случайное создание уровней, пошаговый игровой процесс, плиточная (тайлова) или ASCII-графика.

В работе предложено обобщение алгоритма поиска кратчайшего пути алгоритма A* при динамической конечной цели с помощью метода прохода всех вершин взвешенного неориентированного графа, а также реализация этого алгоритма для визуализации.

Ключевые слова: теория графов; взвешенный граф; неориентированный граф; визуализация работы алгоритма; компьютерное моделирование.

THE APPLICATION OF GRAPH THEORY FOR IMPROVING AND RENDERING ALGORITHM FOR FINDING THE SHORTEST PATH IN THE MATHEMATICAL MODELS OF VIDEO GAMES

V. V. Vanin, O. V. Zalewski, P. M. Jablonskiy

Design of complex systems, study their properties and management require the development of a mathematical model. Study of the characteristics of the system using *mathematical models is often the only way of studying complex systems and the solution of important practical problems.*

The application of graph theory to build mathematical models due to the possibility of describing them, a broad class of objects and processes. This gives you the ability to automate the search for optimal route, reach goals, network planning. One area of such applications is video games that require interface development, visualization of user actions and characters, development of the algorithm and the calculation of the moving actions of the characters in the game. Often used games in the genre Roguelike, the features of which is the creation of random levels, turn-based gameplay, tile (tile) or ASCII graphics.

In this paper we propose a generalization of the algorithm for finding the shortest path algorithm A with dynamic the ultimate goal of using the method of passage of all vertices in the weighted undirected graph and an implementation of this algorithm for visualization.*

Key words: graph theory; weighted graph; undirected graph; a visualization of the algorithm; computer simulation.