

УДК 514.18

НЕДОЛІКИ ТА ПЕРЕВАГИ ВИКОРИСТАННЯ ТЕОРІЇ ГРАФІВ В ВІДЕОІГРАХ

Залевська О.В., к.т.н.,

o.zalevska@kpi.ua, ORCID: 0000-0002-3163-1695

Захаркин М.С.

sytnik.akim@gmail.com, ORCID: 0000-0003-4609-8130

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

У роботі розглядаються недоліки та переваги використання теорії графів в відеоіграх. Проведено аналіз існуючих методів застосування теорії графів, таких як алгоритмів пошуку найкоротшого шляху та його модифікації у вигляді алгоритму Дейкстри, Беллмана, алгоритм A^ , жадібний алгоритм та інші. Розглянуто використання теорії графів для опису стану в якому знаходиться персонаж та перетворення таких станів, граф редуції. До загальних недоліків використання теорії графів можна віднести роботу тільки з структурами, що містять графи з невід'ємною довжиною дуг, пошук найкоротшого шляху тільки від однієї вершини, для реалізації методу необхідні повтори алгоритму для кожної вершини, не шукають рефлексивні замикання та мають досить складну в порівнянні з матричними методами програмну реалізацію.*

До переваг методу можна віднести простоту реалізації, невисоку поліноміальну обчислювальну складність, можливість машинної реалізації, знаходження розв'язку задачі, якщо такий існує.

Наведені недоліки та переваги алгоритмів, що базуються на застосуванні теорії графів в відеоіграх дозволяють зробити наступний висновок: вибір алгоритму для реалізації залежить від кількості вершин графу, його орієнтованості та кількості зв'язків між вершинами графів. Не існує єдиного оптимального алгоритму задач у відповідність яким не можливо встановити граф з визначеною кількістю кількістю вершин та зв'язків між ними.

Для невеликої кількості вершин оптимальним для застосування у відеоіграх є алгоритм a^ , але коли кількість вершин графа досягає ста, то застосування даного алгоритму є не ефективним. Це пов'язано з кількістю порівнянь, що проводить алгоритм. Для великої кількості вершин, з урахуванням недоліків та переваг, як оптимальними пропонуються методи послідовного уточнення згори та метод породження та перевірки. Дані методи володіють такими недоліками як відсутність достовірних способів оцінки методу, ризик втрати деяких можливих розв'язків та пошук всіх можливих розв'язків. До переваг даних методів можна віднести наявність в них алгоритму уточнення роз*

в'язків та зовнішній вплив на них в процесі знаходження розв'язку.

Ключові слова: теорія графів, алгоритми пошуку шляху, алгоритм A^* , відеогра.

Постановка проблеми. Під відеогрою розуміють програму, що призначена для керування та організації ігрового процесу. Під час пандемій та інших катаклізмів збільшується вплив відеоігор на суспільство, що провокує розвиток інформаційних технологій для неігрового програмного забезпечення. Досить часто персонажем відеогри керує не людина, а штучний інтелект. Поведінка таких персонажів відрізняється від поведінки персонажів якими керує людина. Ігровий штучний інтелект має містити в собі програмні методики, що створюють ілюзію поведінки живої людини. Ці методи включають в себе алгоритми теорії управління, робототехніки, комп'ютерної графіки та інформатики. Поведінка персонажу залежить від основних правил гри, основних принципів та якості процесу штучного міркування. Якість процесу штучного міркування залежить від фактів, що впливають на певну ситуацію та ефективність реалізації [1].

Головним принципом поведінки персонажа є прийняття рішень щодо подальшої поведінки персонажу. Найяскравіше це реалізується в стратегіях реального часу, де потрібно аналізувати і поточний стан персонажу, і повністю стан всієї гри. Приймавши рішення про те, щоб потрапити до певного об'єкту необхідно оцінити та вибрати оптимальний шлях до цього об'єкту, час, який необхідно витратити на цей шлях та можливі перешкоди. Це може бути реалізовано за допомогою побудови відповідного графу. Аналіз можливих рішень проблеми приводить до вибору методу реалізації подальшої поведінки.

Аналіз останніх досліджень і публікацій. В роботі [2] розглянуто використання графів для вирішення проблеми передачі даних в типовій структурі багатовимірної системи. В роботі [3] наведено можливий варіант модифікації алгоритму A^* при динамічній кінцевій цілі за допомогою методу проходження через всі вершини графу. В роботі [4] запропоновано узагальнення алгоритму A^* та візуалізація його роботи. Аналіз недоліків та переваг методів реалізації теорії графів залишається актуальним питанням.

Формулювання цілей статті. Провести аналіз існуючих методів використання теорії графів в відеоіграх. Встановити переваги та недоліки таких методів. Сформулювати умови при яких кожен з наведених методів є оптимальним. Основуючись на встановлених перевагах та недоліках надати можливі варіанти удосконалення цих методів.

Основна частина.

Розглянемо методи знаходження оптимального шляху поведінки персонажа відеогри:

- алгоритм Дейкстри, що вирішує проблему найкоротшого шляху з

одним джерелом з невід'ємною вагою ребра;

- алгоритм Беллмана – Форда вирішує проблему з одним джерелом, якщо ваги країв можуть бути від'ємними;

- алгоритм пошуку A^* вирішує найкоротший шлях для однієї пари за допомогою евристики для спроби прискорити пошук;

- алгоритм «жадібного пошуку»;

- пропонуються методи послідовного уточнення згори;

- метод породження та перевірки.

Алгоритм Дейкстри - алгоритм пошуку найкоротшого шляху між вершинами в графі. Розроблено Едсгер Дейкстрой і опублікований в 1959 році. Алгоритм Дейкстри починає роботу зі стартової точки об'єкта. Щоразу він розглядає найближчу нерозглянутих вершину, додаючи суміжні вершини до безлічі вершин для розгляду. Цей алгоритм гарантовано знайде найкоротший шлях від стартової точки до цільової, якщо в графі немає ребер з негативною вартістю

До переваги алгоритму Дейкстри можна віднести простоту реалізації, невисоку поліноміальну обчислювальну складність, можливість машинної реалізації. До недоліків віднесемо можливість працювати тільки з графовими структурами, які мають невід'ємні довжини дуг, відшукує найкоротші шляхи і їх величини тільки від однієї вершини, для визначення взаємозв'язку між усіма парами вершин, вимагає багаторазового повтору для всіх пар вершин вихідного графа, що відповідно в N раз збільшує обчислювальні витрати в цілому, не знаходження рефлексивних замикань, складне в порівнянні з матричними методами програмне уявлення.

В алгоритмі Беллмана-Форда ми починаємо з ініціалізації всіх відстаней усіх вузлів, за винятком вихідного вузла, який ініціалізується нулем. На кожному кроці ми перебираємо всі ребра всередині графіка. Для кожного ребра від вузла до вузла, за необхідності, оновлюються відстані. Нова можлива відстань дорівнює попередній плюс вага ребра між досліджуваними вершинами..

Складність часу алгоритму Беллмана-Форда дорівнює $O(V \cdot E)$, де V - кількість вершин, а E - кількість ребер усередині графа, оскільки на кожному кроці необхідно перевірити всі ребра.

Головною перевагою алгоритму Беллмана-Форда є його здатність обробляти негативні ваги ребер графу. Однак алгоритм Беллмана-Форда має значно більшу складність, ніж алгоритм Дейкстри. Тому алгоритм Дейкстри має більше застосувань, оскільки графіки з від'ємними вагами зазвичай вважаються рідкісним випадком.

Алгоритм Беллмана-Форда може обробляти спрямовані та неорієнтовані графи з невід'ємними вагами. Однак він може обробляти лише спрямовані графи з від'ємними вагами за умови відсутності від'ємних циклів.

Жадібний алгоритм використовує деяку оцінку (евристику) відстані

до цільової точки. Замість того щоб вибирати вершини поблизу стартової точки, він переглядає найближчі до цільової точки вершини.

Жадібний алгоритм не гарантує найкоротшого шляху, хоча він працює швидше, ніж алгоритм Дейкстри, за рахунок використання евристики. Проблема жадібного алгоритму в тому, що він намагається рухатися в бік цілі, навіть якщо це неправильний шлях. Через ігнорування часу, який необхідно витратити на шлях, результат може виявитися неоптимальним. Складність алгоритму в гіршому випадку $O(b^d)$, де b - середня кількість сусідів у кожній вершині, d - глибина, на якій знаходиться цільова точка

Алгоритм використовує евристику для скеровування напрям пошуку та скорочувати його тривалість. Алгоритм завжди знаходить оптимальний розв'язок, за умови його існування. До переваг алгоритму A^* належить вершини тільки з найкоротшого шляху та видалення інших шляхів.

Алгоритм Дейкстри непоганий для використання в пошуку найкоротшого шляху, але він витрачає час на дослідження всіх напрямків, навіть безперспективних. Жадібний пошук досліджує перспективні напрямки, але може не знайти найкоротший шлях. Алгоритм A^* використовує і дійсну відстань від початку, і оцінене відстань до цілі.

Алгоритм A^* гарантовано знайде найкоротший шлях, якщо евристика не перевищує дійсної відстані між вершинами. Якщо евристика стає менше, алгоритм перетворюється в алгоритм Дейкстри. Коли евристика стає більше то в жадібний алгоритм. Цей алгоритм перебирає лише мінімальну кількість вершин, завдяки тому, що він працює з «оптимістичною» оцінкою шляху через вершину. Оптимістичною в тому сенсі існування розв'язку. Коли A^* завершує пошук, він, згідно з визначенням, знайшов шлях, вага якого менше, ніж оцінка ваги будь-якого шляху через будь-який відкритий вузол.

До недоліків алгоритму можна віднести знаходження пошуку серед всіх вершин графа. У разі, якщо граф буде мати велику кількість вершин цей алгоритм буде неефективним.

Для великої кількості застосовуються методи послідовного уточнення згори та метод породження та перевірки. Дані методи володіють такими недоліками як відсутність достовірних способів оцінки методу, ризик втрати деяких можливих розв'язків та пошук всіх можливих розв'язків. До переваг даних методів можна віднести наявність в них алгоритму уточнення розв'язків та зовнішній вплив на них в процесі знаходження розв'язку.

Висновки. Розглянуто використання теорії графів для опису стану в якому знаходиться персонаж та перетворення таких станів, граф редукції. До загальних недоліків використання теорії графів можна віднести роботу тільки з структурами, що містять графи з невід'ємною довжиною дуг, пошук найкоротшого шляху тільки від однієї вершини, для реалізації методу необхідні повтори алгоритму для кожної вершини, не шукають

рефлексивні замикання та мають досить складну в порівнянні з матричними методами програмну реалізацію. До переваг методу можна віднести простоту реалізації, невисоку поліноміальну обчислювальну складність, можливість машинної реалізації, знаходження розв'язку задачі, якщо такий існує. Наведені недоліки та переваги алгоритмів дозволяють зробити висновок: вибір алгоритму для реалізації залежить від кількості вершин графу, його орієнтованості та кількості зв'язків між вершинами графів. Не існує єдиного оптимального алгоритму задач у відповідність яким не можливо встановити граф з визначеною кількістю вершин та ребер.

Література

1. Dolinskaya IS. Optimal path finding in direction, location, and time dependent environments. *Naval Research Logistics*. 2012 Aug;59(5), P.325-339.
2. Нікітіна Л.О. Нікітін С.О.. Моделі та методи штучного інтелекту у комп'ютерних іграх : [довідник модуля]. Харків : Друкарня Мадрид, 2018. 102 с.
3. Ванін В.В. Залевська О.В., Яблонський П.М. Застосування теорії графів для удосконалення та візуалізації алгоритму пошуку найкоротшого шляху в математичній моделі відео ігри. *Прикладна геометрія та інженерна графіка*. Київ, 2020. Вип. 97. С. 23-28.
4. Ваврук Є.Я. Мозіль З.Г. Вибір алгоритму пошуку оптимального шляху передавання даних у розподіленій системі. *Вісник Національного університету "Львівська політехніка". Комп'ютерні системи та мережі*. Львів, 2018. № 905. С. 42-48.

НЕДОСТАТКИ И ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ ТЕОРИИ ГРАФОВ В ВИДЕОИГРАХ

Залевская О. В., Захаркин М.С.

В работе рассматриваются недостатки и преимущества использования теории графов в видеоиграх. Проведен анализ существующих методов применения теории графов, как алгоритмов поиска кратчайшего пути и его модификации в выкладит алгоритма Дейкстри, Беллмана, алгоритм A^ , жадный алгоритм и другие. Рассмотрено использование теории графов для описания состояния в котором находится персонаж и преобразования таких состояний, граф редукции. К общим недостаткам использования теории графов можно отнести работу только со структурами, содержащих графы с неотъемлемой длиной дуг, поиск кратчайшего пути только от одной вершины, для реализации метода необходимы повторы алгоритма для каждой*

вершины, не ищут рефлексивные замыкания и имеют достаточно сложную в сравнении с матричными методами программную реализацию.

К достоинствам метода можно отнести простоту реализации, невысокую полиномиальную вычислительную сложность, возможность машинной реализации, нахождение решения задачи, если таковой существует.

Приведенные недостатки и преимущества алгоритмов, основанных на применении теории графов в видеоиграх позволяют сделать следующий вывод: выбор алгоритма для реализации зависит от количества вершин графа, его ориентированности и количества связей между вершинами графов. Не существует единого оптимального алгоритма задач в соответствии которым невозможно установить граф с определенным количеством количеством вершин и связей между ними.

Для небольшого количества вершин оптимальным для применения в видеоиграх алгоритм A^* , но когда количество вершин графа достигает ста, то применение данного алгоритма является не эффективным. Это связано с количеством сравнений, проводит алгоритм. Для большого количества вершин, с учетом недостатков и преимуществ, как оптимальным предлагаются методы последовательного уточнения сверху и метод порождения и проверки. Данные методы обладают такими недостатками как отсутствие достоверных способов оценки метода, риск потери некоторых возможных развязки и поиск всех возможных развязки. К преимуществам данных методов можно отнести наличие в них алгоритма уточнения развязки и внешнее воздействие на них в процессе нахождения решения.

Ключевые слова: теория графов, алгоритмы поиска, алгоритм A^* , видеоигра.

DISADVANTAGES AND ADVANTAGES OF USING GRAPH THEORY IN VIDEO GAMES

Olga Zalevska, Mikhailo Zakharkin

The paper discusses the disadvantages and advantages of using graph theory in video games. The analysis of existing methods of applying the theory of graphs, how the algorithm for finding the shortest path and its modification looks like the algorithm of Dijkstra, Bellman, algorithm A^ , greedy algorithm and others. The article considers the use of graph theory to describe the state in which the character is located and transformations of such states, the reduction graph. The general disadvantages of using graph theory include working only with structures containing graphs with an integral length of arcs, finding the shortest path from only one vertex, to implement the method, the algorithm needs to be repeated for each vertex, reflexive closures are not searched for,*

and are rather complicated in comparison with matrix software implementation methods.

The advantages of the method include simplicity of implementation, low polynomial computational complexity, the possibility of machine implementation, and finding a solution to the problem, if any.

The above disadvantages and advantages of algorithms based on the use of graph theory in video games allow us to draw the following conclusion: the choice of an algorithm for implementation depends on the number of graph vertices, its orientation and the number of connections between graph vertices. There is no single optimal problem algorithm according to which it is impossible to establish a graph with a certain number of vertices and connections between them.

For a small number of vertices, the A * algorithm is optimal for use in video games, but when the number of vertices in the graph reaches one hundred, then the use of this algorithm is ineffective. This is due to the number of comparisons the algorithm conducts. For a large number of vertices, taking into account the disadvantages and advantages, methods of sequential refinement from above and the method of generation and verification are proposed as optimal. These methods have such drawbacks as the lack of reliable methods for evaluating the method, the risk of losing some of the possible ties and the search for all possible ties. The advantages of these methods include the presence of an algorithm for clarifying the linkage and external influence on them in the process of finding a solution.

Key words: graph theory, search algorithms, A * algorithm, video game.

References

1. Dolinskaya, Irina (2012) *Optimal Path Finding in Direction, Location and Time Dependent Environments*, *Naval Research Logistics*, 59(5), 325-339 [in English].
2. Nikitina, L., Nikitin, S. (2018) Models and methods of artificial intelligence in computer games. Kharkiv: "Madrid Printing House" [in Ukrainian].
3. Vanin, Vladimer, Zalevska, Olha, Yablonskyi, Petro (2020) Application of graph theory to improve and visualize the shortest path search algorithm in a mathematical model of a video game. *Applied geometry and engineering graphics*. 97, 23-28, [in Ukrainian].
4. Varuk, E., Mozil, Z. (2018) Selection of the algorithm for finding the optimal way of data transmission in a distributed system. *Bulletin of the National University "Lviv Polytechnic". Series: Computer systems and networks*. 905, 42-48, [in Ukrainian].