

РЕАЛІЗАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ АЛГОРИТМУ A^* ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ В GAME DEVELOPMENT

Вінницький національний технічний університет

Анотація

Роботу присвячено реалізації алгоритму пошуку найкоротшого шляху A^* в 2D просторі. Аналізується взаємодія між генератором ігрового світу, структурами даних ландшафту та модулем пошуку шляху.

Ключові слова: алгоритм A^* , математична модель, Game Development.

Abstract

The paper is devoted to the implementation of A^* shortest path search algorithm in 2D space. The interaction between the game world generator, the terrain data structures, and the pathfinding algorithm module is analyzed.

Keywords: A^* search algorithm, mathematical model, Game Development.

Вступ

Побудова шляху між двома точками – це одна із проблем, які стоять перед кожним розробником комп'ютерних ігор. Вона полягає в знаходженні оптимальної траєкторії між початковою точкою А та кінцевою точкою В з урахуванням перешкод та особливостей пересування по різних типах поверхні.

Теоретичні основи алгоритму A^*

Математична модель алгоритму A^* – це метод пошуку найкоротшого шляху на графі, який би мінімізував функцію:

$$f(n) = g(n) + h(n),$$

де $g(n)$ – фактична вартість шляху від початкового вузла до вузла n , $h(n)$ – евристична оцінка вартості від вузла n до кінцевої точки, $f(n)$ – загальна оціночна вартість.

Покажемо реалізацію даного алгоритму у класі Pathfinder. За функцію $h(n)$ візьмемо Манхеттенську відстань, коли рух у просторі дозволяється лише в чотирьох напрямках по сітці (вверх, вниз, вліво, вправо):

$$h(n) = |x_{goal} - x_n| + |y_{goal} - y_n|.$$

Це дозволяє алгоритму ефективно знаходити оптимальний шлях, мінімізуючи сумарну вартість пройденої відстані та залишку шляху.

Структура середовища та генерація світу

Перш ніж алгоритм зможе прокласти шлях, необхідно визначити граф навігації. У нашому коді світ представлений як **сітка (Grid)** тайлів.

TileData – структура, що зберігає стан конкретного вузла (тип рельєфу, наявність об'єктів або будівель);

WorldGenerator – використовує шум Перліна (через FastNoiseLite) для створення процедурного ландшафту. Це створює природні зони (воду, траву, камінь), які безпосередньо впливають на можливість проходження.

Програмна реалізація

Вузли та пріоритетність. Структура PathNode (див рис. 1) інкапсулює координати та значення вартостей. Важливим елементом є перевантаження оператора порівняння. Завдяки цьому

priority_queue автоматично витягує найперспективніший вузол, що мінімізує кількість перевірених вузлів.

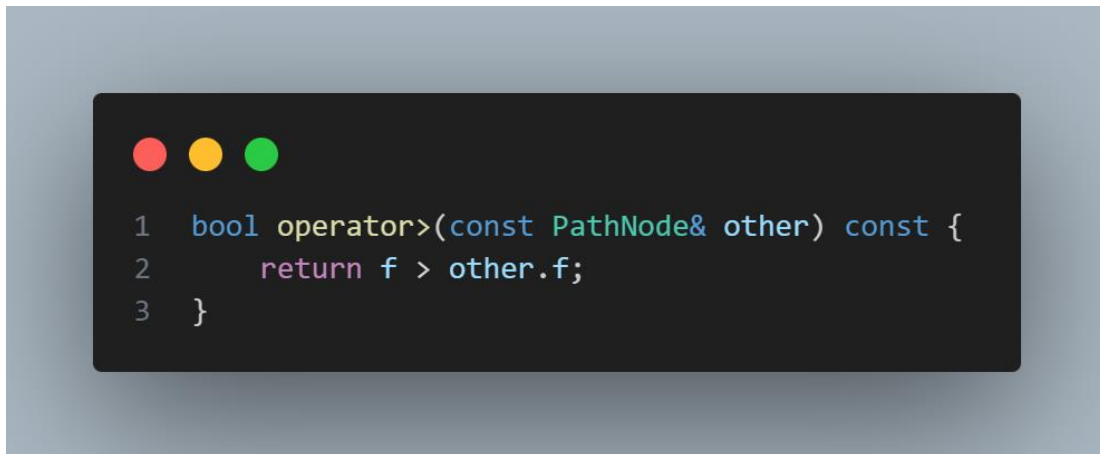


Рис. 1 Перевантаження оператора для класу PathNode

Перевірка прохідності (Walkability). Функція isWalkable виконує роль фільтра. Вона враховує:

- 1) межі карти – перевірка виходу за координати WORLD_WIDTH/HEIGHT;
- 2) тип ландшафту – якщо terrainIt->second.isWalkable є false (наприклад, глибока вода), шлях закритий;
- 3) динамічні об'єкти – наявність buildingRef (будівель) миттєво робить тайл непрохідним.

Цикл пошуку. Алгоритм використовує два списки: OpenSet – кандидати на розгляд (пріоритетна черга); ClosedSet – вже оброблені вузли (карта для швидкого доступу).

Процес ітеративно розширює «фронтір», перевіряючи чотири сусідні. Якщо знайдено коротший шлях до вже існуючого вузла, його значення $g(n)$ та батьківський вузол оновлюються.

Оптимізація та керування пам'яттю. У даній реалізації є важливий аспект очищення пам'яті (див. рис 2)



Рис. 2 Фрагмент коду очищення пам'яті

Оскільки вузли створюються динамічно через new PathNode, цей цикл запобігає витоку пам'яті після завершення пошуку шляху.

Висновки

Це досить типова, але робоча реалізація A* для ігор на сітці. Використання карти типів ландшафту та пріоритетної черги працює досить швидко.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ka Heng Chan, Aida Mustapha, Siaw Chong Lee. Shortest Pathfinding in a Standard Rectangular Maze using A Search Algorithm*, International Journal of Industrial Electronics and Electrical Engineering. Vol. 16 No. 3 (2024). Available: <https://publisher.uthm.edu.my/ojs/index.php/ijie/article/view/18232>.

2. Daohong Liu. Research of the Path Finding Algorithm A* in Video Games. (n.d.). International Journal of High-Speed Electronics and Systems. Vol. 39 (2023). Available: <https://drpress.org/ojs/index.php/HSET/article/view/6642>

Потуторівський Максим Андрійович – студент групи ЗПІ-256, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: nazarkupchishen@gmail.com.

Науковий керівник: **Прозор Олена Петрівна** – к.пед.н., доцент кафедри вищої математики, Вінницький національний технічний університет, Вінниця, email: prozor@vntu.edu.ua.

Potutorivskyi Maksim A. – student of group ЗПІ-25b, faculty of information technologies and computer engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: nazarkupchishen@gmail.com.

Supervisor: **Prozor Olena P.** – PhD (in Pedagogical Sciences), Docent, Associate Professor of the Department of Higher Mathematics, Vinnytsia National Technical University, Vinnytsia, email: prozor@vntu.edu.ua.