

ОПТИМІЗАЦІЯ ШЛЯХІВ З УРАХУВАННЯМ ЗНАЧУЩОСТІ ПРОМІЖНИХ ТОЧОК

Мазурок І. Є., Веремйов К. В.

Одеський національний університет імені Мечникова

У роботі розглядається задача побудова маршруту між двома вершинами в графі з урахуванням векторної ваги проміжних вузлів. Якщо під вузлами розуміти пункти на певній місцевості, то вагами буде задаватися їх значущість для прикладних задач. Наприклад, інтерес щодо туристичних об'єктів, наявність точок обслуговування, води, їжі і таке інше.

Ключові слова: теорія графів, багатокритеріальна оптимізація, фільтр Гауса, функція корисності, маршрут, оптимізація, подорож, туризм, цікавість.

Зараз існує велика кількість алгоритмів, які знаходять оптимальний за довжиною шлях між двома точками. Ми розглядаємо задачу пошуку шляху не за одним критерієм, наприклад часу або відстані, а з урахуванням накопичувального значення проміжних точок. Наприклад, туристам може бути цікаво не просто швидко дійти до пункту призначення, а по дорозі побачити ще якісь пам'ятки. Таким чином, ми можемо описати це завдання як задачу багатоцільової оптимізації: знайти послідовність вузлів $\{v_i\}_{i=1}^n$, таких що

$$\begin{cases} \sum_{i=1}^{N-1} distance(v_i, v_{i+1}) \rightarrow \min \\ \sum_{i=1}^N I(v_i) \rightarrow \max \end{cases},$$

Де $I(n)$ – функція цікавості вузла, показує наскільки важливий, цікавий вузол.

Щоб вирішити цю проблему, було вирішено модифікувати алгоритм пошуку короткого шляху - A^* (А зірка). Як відомо, A^* вибирає вузол з мінімальним

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

де n — вузол на шляху, $g(n)$ — вартість шляху від початкового вузла до n , а $h(n)$ — евристика, яка оцінює вартість найдешевшого шляху від n до пункту призначення. Модифікація полягала в тому, щоб обчислювати $g(n)$ не просто як довжину шляху до вузла, але включити туди цікавість попереднього шляху. Алгоритм A^* намагається мінімізувати $g(n)$, а нам потрібно максимізувати суму $I(n)$, що дорівнює мінімізації суми $-I(n)$. Отже, функція корисності, яка може описати, який вузол може бути кращим на цьому кроці алгоритму за критеріями відстані від початку, цікавості та довжини до місця призначення, може бути функція, яка враховує ці параметри, наприклад (де k_i, k_g, k_h — ваги для функцій)

$$f(n) = k_g g(n) - k_i I(n) + k_h h(n)$$

Для тестування були обрані граfi, представлені матрицями, в яких дозволено переміщення вліво, вправо, вгору і вниз, за винятком точок-бар'єрів (клітинки є вершинами графа, а ребра знаходяться між двома клітинками). На малюнку 1 показано результат роботи алгоритму. Як бачимо, алгоритм дав досить швидкий шлях, але додав до нього точки інтересу. Тож шлях не найшвидший, але досить швидкий і цікавий водночас. Це те, що ми намагалися отримати.

Але давайте подивимося на верхній лівий кут шляху. Там є вузол з досить високим значенням цікавості (дев'ять), але він не включений до шляху і навіть не розглядався. Отже, спробуємо розглядати й врахувати вузли, нехай і далекі від поточної точки. Для цього можна використовувати модифікацію фільтра Гауса:

$$G_I(x, y) = \frac{1}{2\pi\sigma^2} I(x, y) e^{-\frac{(p_1-x)^2 + (p_2-y)^2}{2\sigma^2}}$$

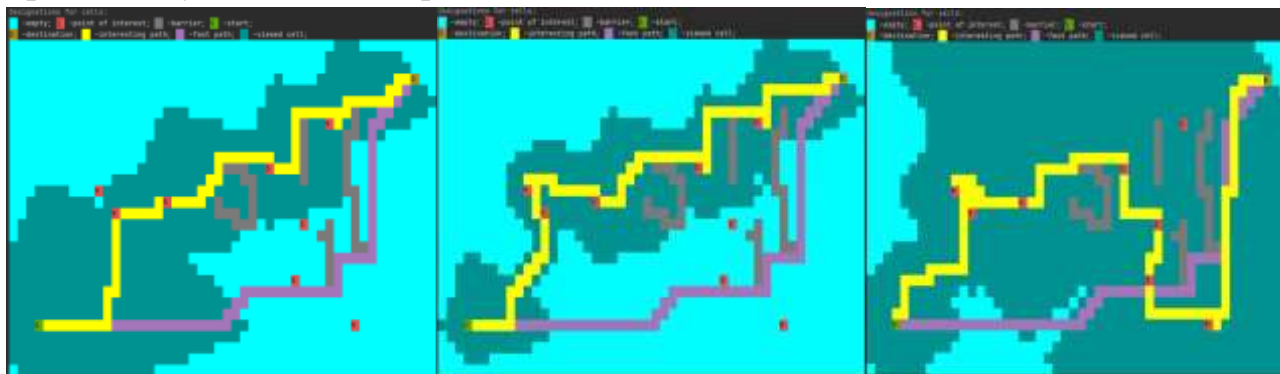
Він як раз може давати найвище значення, коли (x, y) є точкою (вузлом) інтересу (p_1, p_2) і чим далі (x, y) від (p_1, p_2) , тим нижче. Отже, ідея полягає в тому, щоб додати до $f(n)$ доданок, який враховує, наскільки далеко від цікавої точки знаходиться спостережувана точка (й наскільки та точка цікава). Таким

чином, для кожної точки цікавості ми можемо обчислити фільтр для навколишніх точок i , підсумувавши його з фільтрами інших точок, можна отримати один фільтр F . Використовуючи коефіцієнт k_F для фільтра, нова функція корисності:

$$f(n) = k_g g(n) - k_i I(n) - k_F F(n) + k_h h(n).$$

Також у такий спосіб можна сумувати значення багатьох цікавих точок й брати до уваги концентрацію пам'яток й розглядати привабливість території.

Таким чином, як ми бачимо на малюнку 2, ми додали до шляху ще одну точку з досить високою оцінкою інтересу. Варіації значення коефіцієнтів, дозволяють отримувати різні за дією алгоритми. Наприклад, зменшуючи k_h , вклад евристичної функції, можна отримати більш «гуляючий» алгоритм, де перевага надається більш цікавості, ніж швидкості. Алгоритм може розглядати більшу площу, менша тяга до кінцевої точки (мал. 3). Таким чином, враховуючи проміжні вузли, можна отримати як цікаві, так і досить швидкі шляхи.



Малюнок 1

Малюнок 2

Малюнок 3

Література

1. GitHub посилання на проект— 2023— Посилання:
https://github.com/KyrylVeremiov/a_star_modified_by_poi