УДК 004.02

Мисюков Альберт Александрович, студент, Уфимский университет науки и технологий, Российская Федерация, г. Уфа

Научный руководитель – доцент **Валиахметова Юлия Ильясовна**, Уфимский университет науки и технологий, Российская Федерация, г. Уфа

АЛГОРИТМ ПОИСКА КРАТЧАЙШЕГО ПУТИ А*

Аннотация: Статья посвящена алгоритму поиска кратчайшего пути А*. В статье кратко описывается история появления алгоритма (Нильс Нильсон, Бертрам Рафаэль, Питер Э. Харт), принцип работы алгоритма и весовая функция $f(x) = g(x) + h(x)$. Также в статье приводятся популярные виды метрик, используемые в качестве эвристик (манхэттенское, Чебышева и евклидово расстояния) и требования к ним (допустимость и монотонность). В статье также упомянута зависимость времени выполнения алгоритма от выбора эвристики.

Ключевые слова: граф; алгоритм Дейкстры; алгоритм на графе; алгоритм А*; жадный алгоритм; эвристика; обход графа; поиск кратчайшего пути.

Abstract: The article is devoted to the shortest path search algorithm A*. The article briefly describes the history of the algorithm (Nils Nilson, Bertram Raphael, Peter E. Hart), the principle of the algorithm and the weight function $f(x) = g(x) + h(x)$. The article also provides popular types of metrics used as heuristics (Manhattan, Chebyshev, and Euclidean distances) and their requirements (tolerance and monotony). The article also mentions the dependence of the algorithm execution time on the choice of heuristics.

Keywords: graph; Dijkstra's algorithm; graph algorithm; A* algorithm; greedy algorithm; heuristics; graph traversal; shortest path search.

Введение

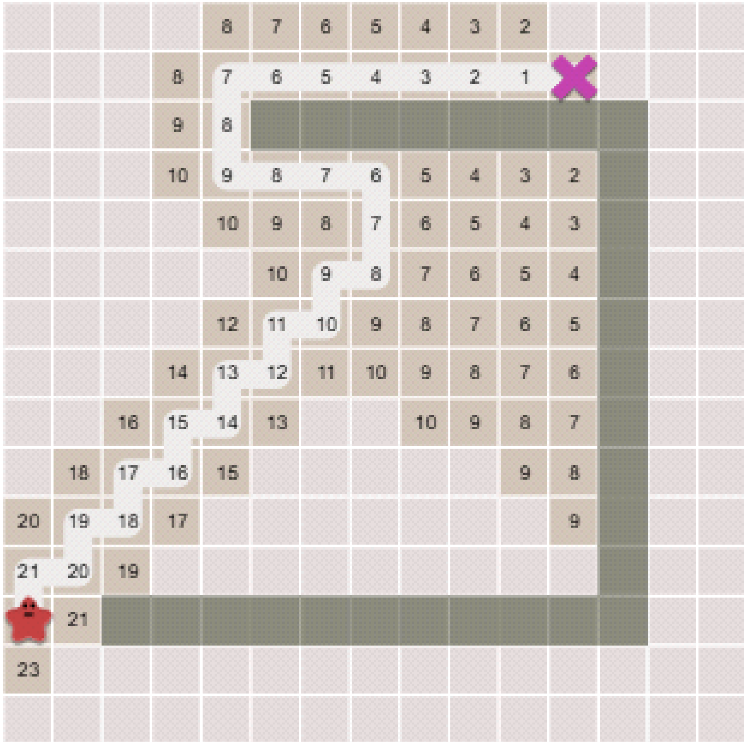
Алгоритм A* (произносится «A звезда» или «A стар», англ. A star) — алгоритм поиска оптимального пути во взвешенном графе без отрицательных весов. [1], [2]. A* объединяет в себе идеи Дейкстры и жадного алгоритма поиска по первому наилучшему совпадению. [1], [2], [3], [4] Алгоритм нашел широкое применение в сфере игровых разработок и GPS навигации. [4]

В 1964 Нильс Нильсон предложил использовать метрики в качестве эвристик для улучшения алгоритма Дейкстры. Алгоритм был назван A1. В 1967 Бертрам Рафаэль сделал значительные улучшения A1, но не смог доказать оптимальность своих изменений. Данный алгоритм был назван A2. В 1968 году Питер Э. Харт с незначительными изменениями A2 доказал его оптимальность при использовании последовательной эвристики. Алгоритм Харта был обозначен звездой (A*), которая включала в себя предыдущие числовые версии. [2]

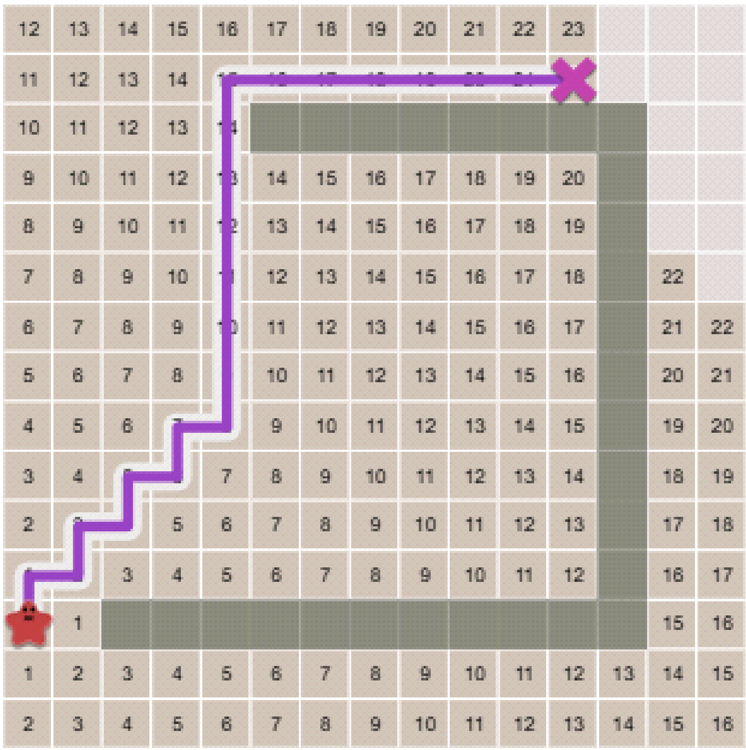
Описание алгоритма

Алгоритм Дейкстры посещает вершины в порядке возрастания стоимости пути от начальной точки, стремясь найти оптимальный маршрут до цели. Однако данный алгоритм не учитывает расположение конечной точки и рассматривает все пути, даже бесперспективные. Алгоритм поиска по первому наилучшему совпадению вводит эвристическое приближение для оценки близости вершины к конечной точке. Это направляет алгоритм в сторону финальной вершины, позволяя не обходить все пути. Данный подход работает быстрее, чем алгоритм Дейкстры, но не гарантирует оптимальность пути. Если целевая точка будет находится за препятствием, алгоритм поиска по первому совпадению поведет нас по неоптимальному маршруту. [4]

Рис. 1-2. Левый рисунок отображает работу алгоритма Дейкстры, правый рисунок отображает работу алгоритма поиска по первому наилучшему



совпадению.



Алгоритм A* использует точность Дейкстры и скорость эвристического поиска. A* обходит те вершины, до которых можно дойти за минимальное расстояние и из которых можно попасть в целевую точку за минимальное расстояние. Для этого вводится следующая весовая функция: [2], [5]

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

где $g(v)$ - наименьшая стоимость пути в v из начальной вершины, $h(v)$ - эвристическое приближение стоимости пути от v до конечной точки. Причем эвристическая функция должна обладать следующими свойствами: [2], [5]

– Эвристическая функция должна быть допустимой. Эвристическая функция допустима, если для любой вершины v значение $h(v)$ не превышает минимального расстояния от вершины v до конечной точки;

– Эвристическая функция должна быть монотонной. Эвристическая функция монотонна, если значение $h(v1)$ и $h(v2)$ у потомков v1 и v2 отличается не более чем на вес ребра между v1 и v2.

$f(v)$ показывает предполагаемую длину пути до цели через вершину v. Чем меньше $f(v)$, тем этот путь перспективнее. Принцип работы алгоритма следующий: выбирается вершина с минимальным значением весовой функции $f(v)$ (у начальной точки $f(v) = 0$). Производится релаксация ее соседей, то есть для каждой смежной вершины пересчитывается значение $f(v)$, и они добавляются в приоритетную очередь по значению весовой функции для дальнейшего извлечения. Процесс повторяется, пока не раскроется целевая вершина или все дерево не будет просмотрено. Из множества решений выбирается решение с наименьшей стоимостью. [1], [2], [3].

Виды эвристик

Поведение алгоритма сильно зависит от того, какая метрика выбрана в качестве эвристики. В свою очередь, выбор метрики зависит от постановки задачи. [1]

Если мы можем перемещаться в четырех направлениях, то в качестве эвристики стоит выбрать манхэттенское расстояние:

Расстояние Чебышева применяется, когда к четырем направлениям добавляются диагонали:

Если передвижение не ограничено сеткой, то можно использовать евклидово расстояние по прямой:

Если вычислить эвристическое приближение по данным формулам невозможно, то h(v) принимается равным 0, и алгоритм A* будет вести себя так же как и алгоритм Дейкстры. Для корректной работы алгоритма необходимо, чтобы эвристическое приближение вычислялось за O(1). [2]

Асимптотика

Временная сложность:

При h(v) = 0 алгоритм работает как Дейкстра, то есть за , где |V| — количество вершин, а |E| — количество ребер в графе.

В худшем случае, когда эвристика неэффективна, число вершин, исследуемых алгоритмом, растет экспоненциально по глубине d поиска, достигая , где b — средняя степень ветвления.

Сложность становится полиномиальной , когда пространство поиска является деревом, а эвристика удовлетворяет следующему условию: , где h* - оптимальная эвристика, то есть точная оценка расстояния из вершины x к цели. [1], [2], [5]

Пространственная сложность:

Алгоритм хранит в памяти все открытые и закрытые вершины, поэтому асимптотика также может достигать . [1], [2], [5]

Библиографический список

- Алгоритм A* [Электронный ресурс] URL: https://neerc.ifmo.ru/wiki/index.php?title=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_A*
- A* [Электронный ресурс] URL: https://ru.wikipedia.org/wiki/A*
- Introduction to A* [Электронный ресурс] URL: https://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html

- Introduction to the A* Algorithm [Электронный ресурс] URL: https://www.redblobgames.com/pathfinding/a-star/introduction.html
- A* search algorithm [Электронный ресурс] URL https://en.wikipedia.org/wiki/A*_search_algorithm