

## ПРИМЕНЕНИЕ АЛГОРИТМА A\* И ЕГО МОДИФИКАЦИЙ

Тюшкевич Ф.И., студент

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Жвакина А.В. – канд. тех. наук, доцент

**Аннотация.** В работе рассматривается алгоритм поиска пути A\* который нашёл применение в различных областях программирования, принцип его работы и преимущества перед алгоритмом Дейкстры, а также его модификации. Предложены наглядные примеры работы алгоритма A\* и его модификаций.

Поиск пути – это задача нахождения наилучшего, оптимального маршрута от одной точки до другой в графе, сетке или любом другом дискретном пространстве. Алгоритмы поиска пути определяют, как наиболее эффективно добраться из пункта А в пункт Б, учитывая при этом стоимость перемещения и препятствия. Данная задача часто встречается в современном мире, например: в навигаторах и компьютерных играх.

Один из наиболее известных алгоритмов поиска пути это – алгоритм Дейкстры, однако он не является наиболее эффективным для задач поиска в больших пространствах поскольку обходит очень большое количество вершин. Для того чтобы значительно сократить время поиска используется алгоритм A\*, который меняет приоритет вершин для изменения порядка их обработки [1].

Суть алгоритма A\* заключается в использовании двух множеств: открытого и закрытого. Под открытым множеством подразумеваются вершины, которые уже были обнаружены, но ещё не были обработаны. Именно из открытого множества выбирается следующая вершина для обработки. Под закрытым же множеством подразумеваются вершины, которые уже были обработаны. Выбор следующей вершины для обработки осуществляется с помощью функции

$$f(n) = g(n) + h(n) \quad (1),$$

где  $g(n)$  – стоимость пути минимальной стоимости от начальной вершины до некоторой вершины  $n$ ,  $h(n)$  – стоимость пути минимальной стоимости от вершины  $n$  до цели. Значение  $h(n)$  берётся как наиболее благоприятный исход (на пути нет никаких препятствий или существует прямой путь от вершины до цели) [2].

Благодаря использованию эвристики поиск пути направляется в более перспективном направлении существенно сокращая количество посещённых вершин по сравнению с алгоритмом Дейкстры, что хорошо видно на рисунке 1.

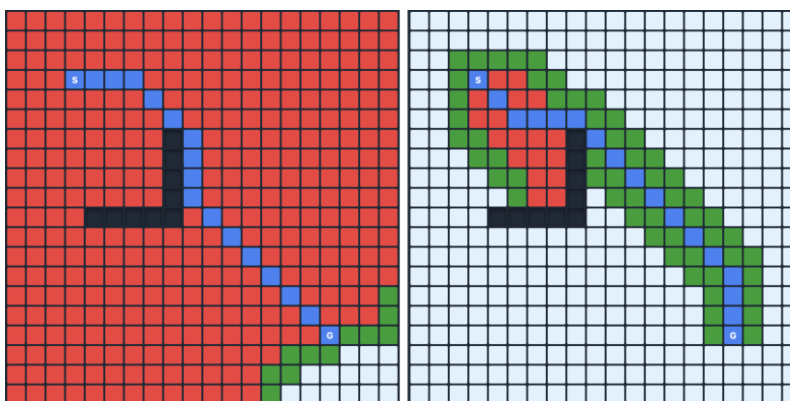


Рисунок 1 – Схема препятствий (чёрный), посещённых узлов (красный), узлов открытого множества (зелёный), пути из s в g (синий) для алгоритма Дейкстры (слева) и A\* (справа) на примере двумерной сетки

В примере из рисунка 1 количество посещённых вершин для алгоритма Дейкстры составило 365, а для A\* 34, при этом оба алгоритма нашли разные пути одинаковой стоимости (стоимость перехода по диагонали в данном примере была взята равной 1.41).

Несмотря на то, что A\* способен находить оптимальный путь при использовании допустимой эвристики, на практике часто необходимо пожертвовать оптимальностью найденного решения ради повышения скорости работы. В таких случаях применяются различные модификации A\*.

Один из вариантов такой модификации это взвешенный A\*. Если  $h(n)$  — допустимая эвристическая функция, то во взвешенной версии поиска A\* используется  $hw(n) = \epsilon \cdot h(n)$ ,  $\epsilon > 1$  в качестве эвристической функции, и поиск A\* выполняется как обычно (что в итоге происходит быстрее, чем при

использовании  $h$ , поскольку расширяется меньше вершин). Найденный алгоритмом поиска пути может иметь стоимость не более чем в  $\epsilon$  раз больше, чем стоимость пути с наименьшей стоимостью в графе [3]. Пример работы алгоритма представлен на рисунке 2.

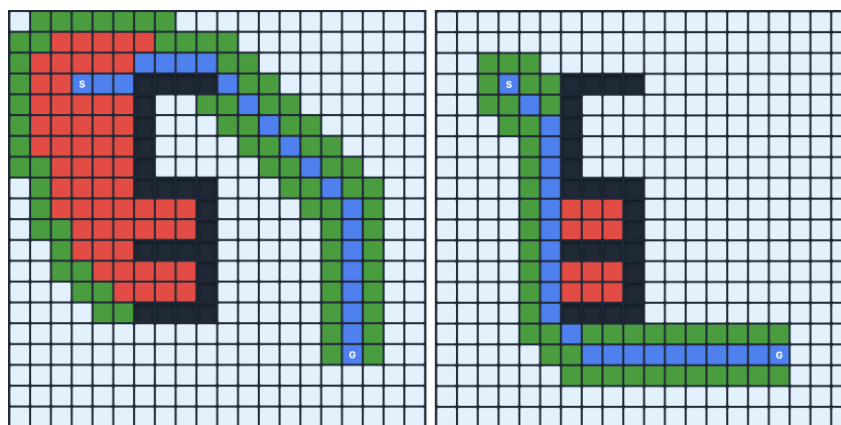


Рисунок 2 – Схема препятствий (чёрный), посещённых узлов (красный), узлов открытого множества (зелёный), пути из  $s$  в  $g$  (синий) для алгоритма  $A^*$  (слева) и взвешенного  $A^*$  при  $\epsilon = 2$  (справа) на примере двумерной сетки

Другой модификацией, которая уже гарантирует оптимальность найденного пути, является двунаправленный  $A^*$ . В случае этого алгоритма поиск ведётся сразу с двух вершин (начальной и целевой). Важным условием завершения работы алгоритма является следующий критерий: алгоритм завершает работу тогда и только тогда, когда доказано, что найденный путь имеет минимальную возможную стоимость:

$$d \leq \max(\min fs(n), \min fg(n)) \quad (2),$$

где  $d$  – стоимость текущего найденного пути минимальной стоимости, а  $\min fs(n)$  и  $\min fg(n)$  – минимальное значение функции  $f$  среди всех вершин текущего открытого множества начальной и целевой вершин соответственно. Невыполнение этого критерия может привести к нахождению не самого оптимального пути.

В итоге алгоритм  $A^*$  по праву можно считать одним из наиболее эффективных и универсальных алгоритмов поиска пути. Благодаря сочетанию точности и использования эвристической информации, он позволяет находить решения в широком классе задач. Применение  $A^*$  или его модификаций в подходящей ситуации может значительно улучшить производительность программы.

**Список использованных источников:**

1. Wagner Dorothea; Willhalm Thomas. *Speed-up techniques for shortest-path computations*. STACS. / Massachusetts Institute of Technology, 2007. — С. 27 - 29.
2. Нильсон Н. *Искусственный интеллект: методы поиска решений = Problem-solving Methods in Artificial Intelligence* / Пер. с англ. В. Л. Стефанюка; под ред. С. В. Фомина. — М.: Мир, 1973. — С. 68 - 70.
3.  $A^*$  search algorithm – Wikipedia [Электронный ресурс]. – Режим доступа: [https://en.wikipedia.org/wiki/A\\*\\_search\\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm).