

Научная статья

Статья в открытом доступе

УДК 656.022

doi: 10.30987/2782-5957-2025-9-60-69

РЕШЕНИЕ ЗАДАЧИ ПРОСТРАНСТВЕННОГО РАЗВИТИЯ НАЗЕМНОЙ ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЫ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА ПОИСКА ПУТИ A*

Дмитрий Владимирович Кузьмин

Российский университет транспорта, Москва, Россия

kuzminmiit@yandex.ru; <https://orcid.org/0000-0002-0585-393X>

Аннотация

Задача. Рассмотрен вопрос применимости алгоритма A* для решения задач поиска пути в контексте пространственного развития линейных объектов наземной транспортной инфраструктуры.

Цель исследования. Определение наиболее предпочтительной конфигурации алгоритма поиска пути для решения задачи пространственного развития линейной транспортной инфраструктуры.

Методы исследования. Алгоритм A*, широко используется для различных прикладных задач теории графов, в том числе трассирования и планирования пути.

Новизна работы. С алгоритмом проведен ряд простых экспериментов с целью определения количественных показателей его асимптотической сложности, т.е. количества выполняемых операций и времени выполнения алгоритма. Серия экспериментов имеет различную конфигурацию, определя-

емую направленностью поиска (однонаправленный и двунаправленный), метрикой поиска (Евклидова метрика, Манхэттенское расстояние, расстояние Чебышева) и способом прохода ячеек (прямой и смешанный).

Результаты исследования и выводы. В целом можно заключить, что двунаправленный поиск требует примерно на 19,17 % меньше операций чем однонаправленный. Манхэттенская метрика и расстояние Чебышева показывают схожие результаты по длине пути и времени выполнения. Евклидова метрика позволяет найти более короткий путь, но может потребовать больше времени на выполнение.

Ключевые слова: алгоритм A*, трассирование, развитие, транспортная инфраструктура, система, теория, графы.

Ссылка для цитирования:

Кузьмин Д.В. Решение задачи пространственного развития наземной транспортной инфраструктуры с использованием алгоритма поиска пути A*/ Д.В. Кузьмин // Транспортное машиностроение. – 2025. - № 9. – С.60-69. doi: 10.30987/2782-5957-2025-9-60-69.

Original article

Open Access Article

SOLVING THE PROBLEM OF SPATIAL DEVELOPMENT OF GROUND TRANSPORTATION INFRASTRUCTURE USING PATHFINDING ALGORITHM A*

Dmitry Vladimirovich Kuzmin

Russian University of Transport, Moscow, Russia

kuzminmiit@yandex.ru; <https://orcid.org/0000-0002-0585-393X>

Abstract

Problem. The issue of applying A* algorithm for solving pathfinding problems in spatial development of linear objects of land transport infrastructure is considered.

Study objective. Determining the most preferred configuration of pathfinding algorithm for solving the problem of spatial development of linear transport infrastructure.

Research methods. A* algorithm is widely used for various graph theory applications, including tracing and path planning.

The novelty of the work. A number of simple experiments were carried out with the algorithm in order to determine quantitative indicators of its asymptotic complexity, i.e. the number of operations performed and the algorithm time. The set of experiments has a different configuration, determined by the direction of the search (unidirectional and bidirectional), the search metric (Euclidean distance, Manhattan distance,

Chebyshev distance) and the method of cell passing (direct and mixed).

Research results and conclusions. In general, it can be concluded that bidirectional search requires about 19.17% fewer operations than unidirectional search. Manhattan distance and Chebyshev distance show similar results in terms of path length and execution time. Euclidean distance allows finding a shorter path, but it may take longer to complete.

Keywords: algorithm A*, tracing, development, transport infrastructure, system, theory, graphs.

Reference for citing:

Kuzmin DV. Solving the problem of spatial development of ground transportation infrastructure using pathfinding algorithm A*. *Transport Engineering*. 2025;9:60-69. doi: 10.30987/2782-5957-2025-9-60-69.

Введение

Пространственное развитие транспортной инфраструктуры является фундаментальной задачей организации работы и функционирования транспортных систем. Значительная часть подходов к решению данной задачи сводится к декомпозиции рассматриваемого полигона на отдельные территориальные единицы, совокупность которых, в дальнейшем, рассматривается, как граф. Совокупность рассматриваемых отдельных территориальных единиц, не имеющая разрывов и наложений, представляет собой растровую пространственную модель.

Под растровым моделированием пространственных данных понимают способ цифрового описания пространственных объектов и топологических отноше-

ний между ними. Описание выполняется с помощью регулярных и нерегулярных сеток, покрывающих рассматриваемый полигон. Сетка делит рассматриваемый полигон на дискретные ячейки – операционно-территориальные единицы (ОТЕ). При решении геоинформационных задач чаще используются регулярные (постоянные) сетки, в которых все ОТЕ имеют одинаковый размер, форму и тд. Все ОТЕ содержат одинаковый набор параметров, характеризующих их пространственные свойства, т.н. атрибутивные данные, которые могут содержать информацию о топологических, гидрографических и антропогеографических и других свойствах пространства. Соединение центроидов ОТЕ образует граф.

Обзор источников

Растровые сетки широко используются для решения геоинформационных задач поиска пути, например для трассировок трубопроводов [1, 2], автомобильных дорог [3, 4], железных дорог [3, 5], линий электропередач [6, 7]. Логика использования растровых сеток в этом случае заключается в присвоении всем ОТЕ одинакового набора пространственных данных т.н. атрибутов. Анализируя различными методами распределение значений атрибутов ОТЕ, исследователь может определить наилучший маршрут в рамках существующей инфраструктуры или оптимальное пространственное развитие трассы.

Растровые модели поиска имеют слабые стороны. По причине графового рассмотрения пространства (абстракция узлов и связей) возникают неизбежные искажения трассировки, например, трасса может оказаться избыточно длинной или содержать множество геометрических несовершенств. Это приводит к получению нереалистичных результатов определения пространственного развития трассы. Подробно данная проблематика рассмотрена в работе [8]. В частности отмечается, что одномерный граф является приближением к бесконечному числу трассировок в рамках рассматриваемой области пространства, поэтому неизбежны фактические расхождения между расчетной и реальной трас-

сой. Путь, прокладываемый в растровом пространстве, имеет дискретный шаг, определяемый в том числе геометрическими свойствами формы ОТЕ. Это приводит к неизбежным удлинением и геометрическим несовершенствам трассы. Особенно явно эти негативные эффекты проявляются

Постановка задачи

Пусть имеется граф $G = (V, E)$, где V – множество вершин, E – множество ребер, каждое из которых имеет вес (стои-

Материалы и методы

Алгоритм поиска A^* (« A -star»), разработанный коллективом Стенфордского исследовательского института П. Хартом, Н. Нилльсоном и Б. Рафаэлем в 1968 году является эволюцией алгоритма Дейкстры. Принципиальное отличие заключается в подходах к обходу графа. За счет использования эвристики, алгоритм не перебирает все возможные вершины графа, как алгоритм Дейкстры, а осуществляет направленный перебор.

На этапе инициализации каждой вершине x_i присваивается метка $g(x_i)$, представляющая стоимость пути от начальной вершины x_s до x_i . Начальной вершине x_s присваивается метка 0:

$$O = \{x \in V \mid \text{вершина } x \text{ была обнаружена, но еще не исследована}\}$$

Изначально множество O содержит только начальную вершину x_s . Данное множество можно представить в виде структуры данных приоритетной очереди (*open_set*). Данная структура играет центральную роль в управлении процессом

$$C = \{x \in V \mid \text{вершина } x \text{ была исследована}\}$$

Множество C можно представить в виде структуры данных приоритетной очереди (*closed_set*). Данная структура данных содержит вершины, которые уже были полностью исследованы и используется для предотвращения повторного поиска.

Объявление массива предшественников *predecessor*, который необходим для восстановления пути после того, как алгоритм достиг конечной вершины.

в неоднородных растровых пространствах. По причине существенной разницы количественных показателей атрибутивных данных путь подвержен к частым изменениям направления, тогда как в условиях однородности растра данные искажения менее выражены [9].

мость). Необходимо найти наименее дорогой путь между начальной вершиной x_s и конечно вершиной x_t .

$g(x_s) = 0$. Всем остальным вершинам присваивается метка бесконечность $g(x_i) = \infty$ для всех $x_i \neq x_s$. Каждой вершине x_i вычисляется эвристическая оценка $h(x_i)$ – оценка стоимости пути от x_i до конечной вершины x_t .

Для каждой вершины x_i вычисляется функция оценки $f(x_i) = g(x_i) + h(x_i)$, которая представляет собой оценку общей стоимости пути от начальной вершины x_s до конечной вершины x_t , проходящего через x_i .

Формируется множество открытых вершин O , которые были обнаружены, но не исследованы.

поиска, так как определяет порядок рассмотрения вершин. Приоритет вершин определяется функцией оценки $f(x)$.

Также формируется множество закрытых вершин C . Изначально множество C является пустым.

predecessor[x] – это вершина, предшествующая x на кратчайшем пути от x_s до x , которая обновляется при обнаружении алгоритмом более короткого пути от x_s до x .

Далее, до тех пор, пока *open_set* не пусто, происходит выполнение основного цикла:

1. Выбирается вершина x из *open_set* с наименьшей функцией оценки $f(x)$, которая рассматривается как наиболее пер-

спективная для достижения конечной цели. Если x является x_t , то путь найден и происходит восстановление из массива *predecessor*. В противном случае происходит перемещение вершины из *open_set* в *closed_set*.

2. Далее запускается процесс исследования соседних x вершин. Для каждой соседней вершины выполняется следующий набор операций:

- Если вершина находится в *closed_set*, то она пропускается, так как ранее уже была исследована.

- Вычисляется стоимость пути от начальной вершины до y через x :

$$temp_g = g(x) + c(x, y)$$

- Если y не находится в *open_set* или $temp_g < g(y)$:

- устанавливается предшественник у равный x : $predecessor[y] = x$;

- обновляется стоимость пути до узла y : $g(y) = temp_g$;

- Вычисляется функция оценки для y : $f(y) = g(y) + h(y)$;

- если вершина y не находится в *open_set*, то выполняется её добавление в *open_set*.

На заключительном этапе работы алгоритма происходит восстановление пути. Если конечная вершина x_t была достигнута, то путь можно восстановить, используя массив *predecessor*, двигаясь в обратном направлении от x_t до x_s . Если *open_set* пуст при не достигнутой конечной вершине, то пути не существует.

Блок-схема алгоритма A^* представлена на рис. 1.

Для получения корректного результата (кратчайшего пути), должна обеспечиваться допустимость эвристической функции $h(n)$, т.е. она никогда не должна переоценивать минимально возможную стоимость достижения узла. В качестве возможной эвристики чаще всего используются Евклидова и Манхэттенская метрики.

Манхэттенская эвристика основана на определении расстояния, формируемого только путем прямого шага при переборе соседствующих ячеек.

$$h(A, B) = |x_2 - x_1| + |y_2 - y_1|$$

где x_1, y_1 – координаты ячейки A ; x_2, y_2 – координаты ячейки B .

Преимуществом данной метрики является простота использования и высокая скорость, в силу малого объема потребляемой вычислительной мощности. Однако, если использовать данную метрику в комбинации с диагональным проходом ячеек, то она неизбежно приведет к переоценке расстояния.

Евклидово расстояние основывается на принципе вычисления прямого координатного расстояния между двумя точками (A и B) на плоскости.

$$h(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

где x_1, y_1 – координаты ячейки A ; x_2, y_2 – координаты ячейки B .

Данная метрика требует большего объема вычислительных ресурсов, на взятие квадратного корня, так как сумма квадратов разности дает квадрат расстояния. Как правило дает более точный результат оценки расстояния по сравнению с Манхэттенской метрикой.

Расстояние Чебышева учитывает расстояние по принципу движения короля по шахматной доске

$$h(A, B) = \max(|x_2 - x_1|, |y_2 - y_1|)$$

где x_1, y_1 – координаты ячейки A ; x_2, y_2 – координаты ячейки B .

В некоторых случаях расстояние Чебышева может недооценивать фактическое расстояние до цели, особенно если диагональные перемещения стоят дороже или недопустимы по условию поиска.

Многочисленные исследования, основанные на практическом приложении данных метрик в аспекте решения задачи поиска пути, показывают, что Евклидово расстояние, как правило обеспечивает более точный результат вычисления, особенно в средах со сложной геометрией. Хотя конкретные результаты определяются множеством факторов, конфигурирующих поиск: форма ячейки сетки; наличие препятствий, их геометрия и пространственное расположение; способ прохода ячейки; распределение стоимости и др.

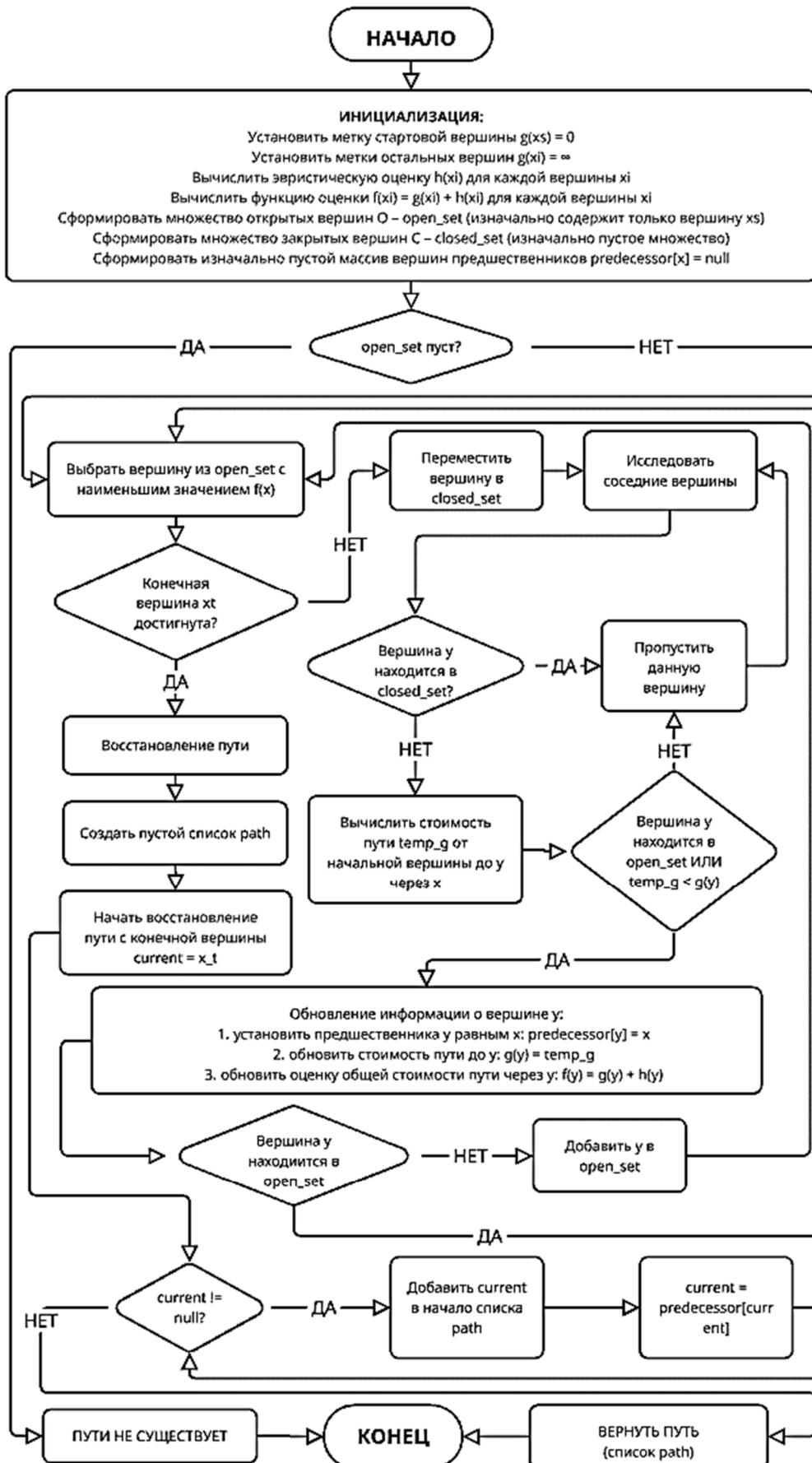


Рис. 1. Блок-схема алгоритма A*
 Fig. 1. Block diagram of algorithm A*

Широкое использование данных метрик при работе с алгоритмом A^* , многократно апробировано научным сообществом для решения разнообразных задач. Универсальность, доступность и широкая применимость алгоритма, фактически сделали поиск A^* определенным стандартом решения задач поиска пути. Выбор между евклидовой и манхэттенской метрикой определяется спецификой решаемой задачи. Если шаг перебора возможен в горизонтальном, вертикальном и диагональном направлении, то целесообразно использование евклидовой эвристики, несмотря на значительную разницу в вычислительных потребностях поиска.

Для использования алгоритма A^* необходимо выполнение следующих условий:

1. Граф, по которому осуществляется поиск должен быть взвешенным, т.е. его вершины или ребра должны иметь вес;
2. Граф должен содержать исходный и целевой узел;
3. Используемая эвристическая функция должна быть допустимой, то есть

Эксперименты и обсуждение

Для определения асимптотической сложности с алгоритмом была проведена серия экспериментов, позволяющая оценить пространственную и временную сложность A^* . Различные варианты поиска будут конфигурироваться используемой метрикой (расстояниями Евклидова, Чебышева и Манхэттенское), вариантами реализации поиска (однонаправленный / двунаправленный) и возможностью пересечения ячейки (прямое, смешанное). Среда поиска, в рамках которой осуществлялся эксперимент, с целью сравнения асимптотической сложности алгоритмов, была беспрепятственной и имела постоянные рейтинги ячеек их форму и размер. Расположение стартовой и целевой вершины, так же оставалось постоянным. Условные координаты стартовой вершины (0, 0), целевой (4, 3).

Результаты визуализации экспериментов представлены на рис. 2.

не должна переоценивать стоимость достижения цели (не быть ниже минимальной стоимости достижения цели).

Структура графа, используемая эвристика и количество узлов определяют асимптотическую (временную) сложность алгоритма A^* . В худшем случае (алгоритм должен перебрать все узлы графа) асимптотическая сложность составит $O(bd)$, где b – коэффициент ветвления, который обозначает среднее количество дочерних узлов, для каждого узла графа, d – количество узлов на пути до целевого узла. Например, если в графе, каждый из узлов содержит 3 дочерних, а глубина поиска составляет 5 узлов, то асимптотическая сложность составит $O(3^5) = O(243)$. Другими словами, в худшем случае алгоритм переберет 243 узла, прежде чем найдет решение.

Существенно снизить объем выполняемой работы позволяет эвристическая функция, приближая сложность к $O(n)$, где n – общее количество узлов в графе.

С целью повышения наглядности результатов экспериментов обобщим полученные результаты инфографически, рис. 3 и 4.

Представленные рисунки, позволяют отследить зависимость покрытия графа и времени выполнения алгоритма от используемой метрики, формата прохода ячейки и направленности поиска пути.

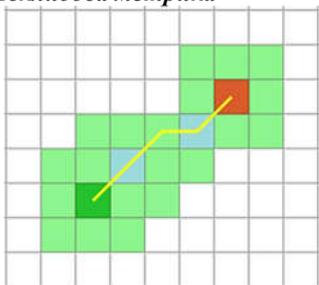
Так как использованный граф является небольшим выполнение алгоритма для всех конфигураций поиска довольно низкая – доли миллисекунд. Однако, двунаправленный поиск в сочетании с Евклидовой метрикой и смешанным проходом ячеек показал самое высокое время выполнения (0,7 мс).

Количество выполняемых операций варьируется в зависимости от конфигурации. В целом можно заключить, что двунаправленный поиск требует примерно на 19,17% меньше операций чем однонаправленный.

СМЕШАННЫЙ (ДИАГОНАЛЬНО – ПРЯМОЙ) ПРОХОД ЯЧЕЕК

Двунправленный поиск

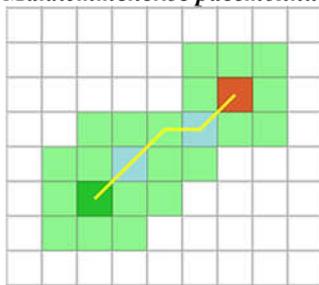
Евклидова метрика



а)

Длина пути – 5,24 ед., количество операций: 28, время выполнения алгоритма: 0,7 мс.

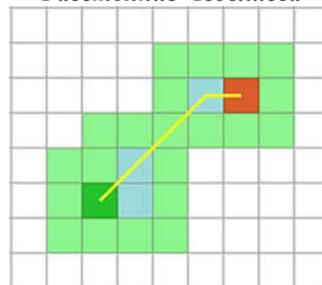
Манхэттенское расстояние



б)

Длина пути – 7 ед., количество операций: 28, время выполнения алгоритма: 0,3 мс.

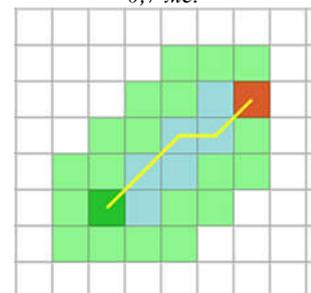
Расстояние Чебышева



в)

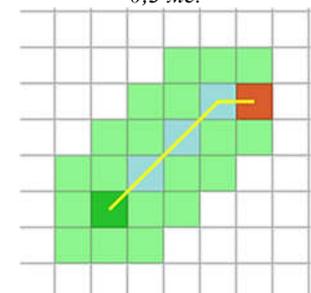
Длина пути – 7 ед., количество операций: 31, время выполнения алгоритма: 0,4 мс.

Однонаправленный поиск



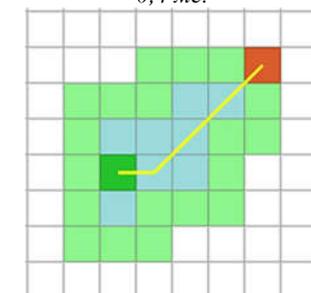
г)

Длина пути – 5,24 ед., количество операций: 35, время выполнения алгоритма: 0,5 мс.



д)

Длина пути – 5,24 ед., количество операций: 29, время выполнения алгоритма: 0,2 мс.



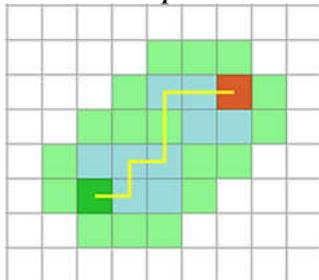
е)

Длина пути – 5,24 ед., количество операций: 39, время выполнения алгоритма: 0,2 мс.

ПРЯМОЙ ПРОХОД ЯЧЕЕК

Двунправленный поиск

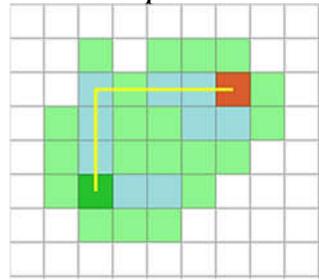
Евклидова метрика



ж)

Длина пути – 7 ед., количество операций: 39, время выполнения алгоритма: 0,2 мс.

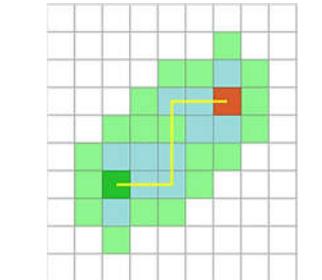
Манхэттенское расстояние



з)

Длина пути – 7 ед., количество операций: 42, время выполнения алгоритма: 0,3 мс.

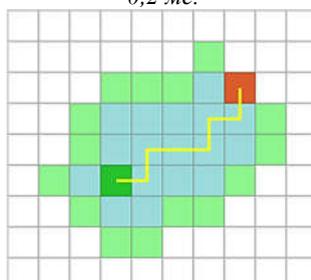
Расстояние Чебышева



и)

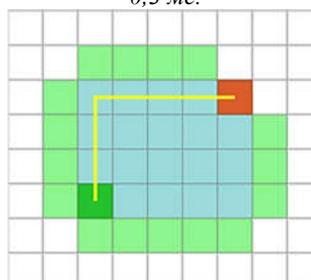
Длина пути – 7 ед., количество операций: 45, время выполнения алгоритма: 0,2 мс.

Однонаправленный поиск



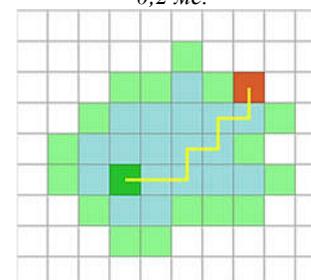
к)

Длина пути – 7 ед., количество операций: 53, время выполнения алгоритма: 0,3 мс.



л)

Длина пути – 7 ед., количество операций: 56, время выполнения алгоритма: 0,2 мс.



м)

Длина пути – 7 ед., количество операций: 56, время выполнения алгоритма: 0,2 мс.

Рис. 2. Визуализация результата поиска пути алгоритмом A*, где зеленая и красная ячейки – соответственно, стартовый и целевой узлы. Ячейки светло-зеленого цвета – ячейки, рассматриваемые в рамках текущей итерации; Ячейки голубого цвета – множество посещенных вершин

Fig. 2. Visualization of the path search result using A* algorithm, where the green and red cells are, respectively, the start and target units. Light green cells are the cells considered in the current iteration; Blue cells are the set of vertices visited

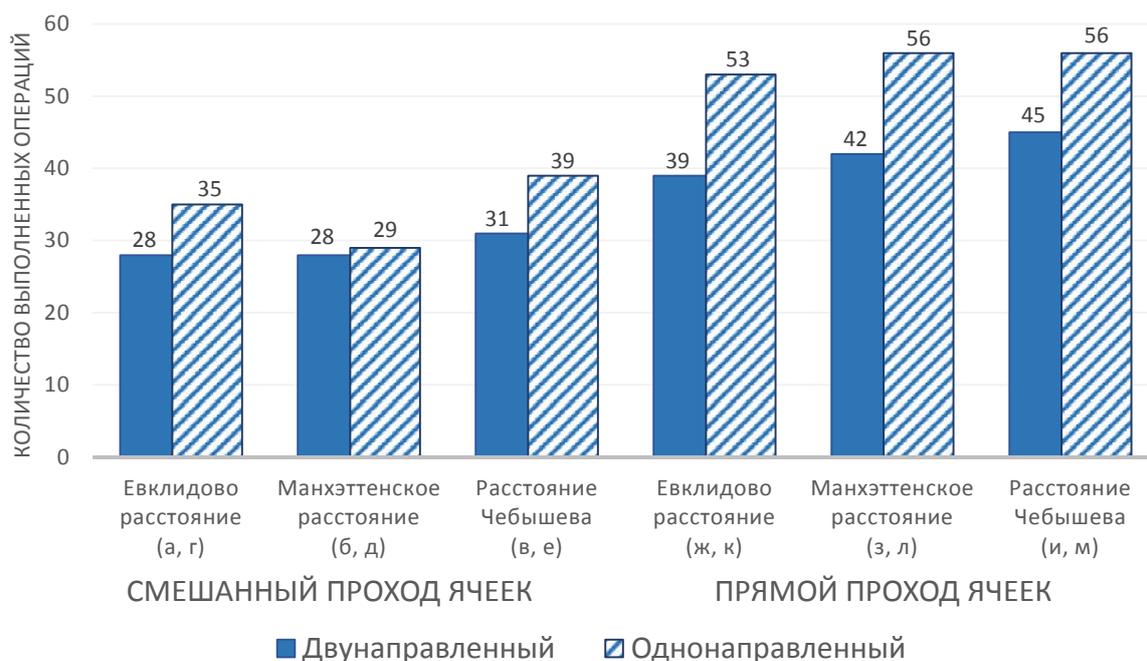


Рис. 3. Результаты экспериментов с различными вариантами реализации алгоритмом A*, по показателю «Количество выполненных операций»
 Fig. 3. Results of experiments with various types of implementing A* algorithm, according to the indicator "Number of operations performed"

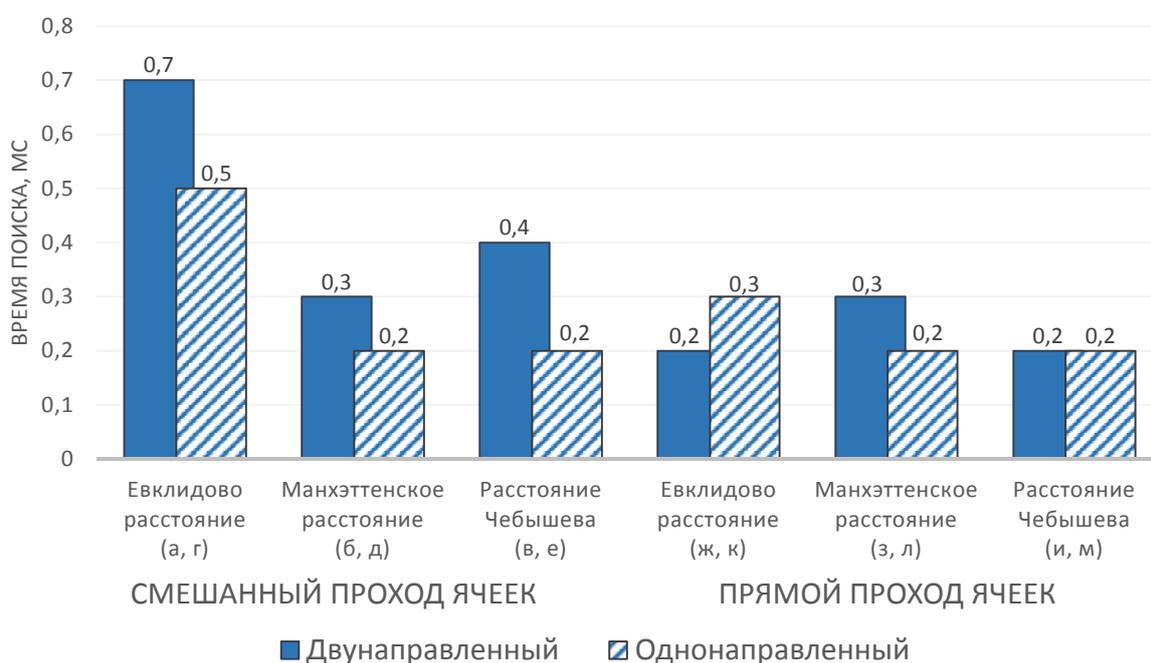


Рис. 4. Результаты экспериментов с различными вариантами реализации алгоритмом A*, по показателю «Время поиска»
 Fig. 4. Results of experiments with various types of implementing A* algorithm, according to the indicator "Search time"

Процентное уменьшение при использовании двунаправленного поиска по сравнению с однонаправленным для каж-

дой конфигурации эксперимента представлено на рис. 5.



Рис. 5. Процентное уменьшение при использовании двунаправленного поиска по сравнению с однонаправленным для каждой конфигурации эксперимента

Fig. 5. Percentage decrease when using bidirectional search compared to unidirectional for each experiment configuration

Манхэттенская метрика и расстояние Чебышева показывают схожие результаты по длине пути и времени выполнения. Евклидова метрика позволяет найти более короткий путь, но может потребовать

больше времени на выполнение. Исходя из сказанного, можно сделать вывод о том, что выбор оптимальной конфигурации использования алгоритма определяется исследователем исходя из приоритетов.

СПИСОК ИСТОЧНИКОВ

1. Automatic pipeline route design with multi-criteria evaluation based on least-cost path analysis and line-based cartographic simplification: case study of the Mus project in Turkey [Internet]. 2023 [cited 2023 Oct 23]. Available from: <https://www.mdpi.com/2220-9964/8/4/173>
2. Kang JY, Lee B. Optimisation of pipeline route in the presence of obstacles based on a least cost path algorithm and laplacian smoothing. *International Journal of Naval Architecture and Ocean Engineering*. 2017;9.
3. Scaparra MP, Church RL, Medrano FA. Corridor location: the multi-gateway shortest path model. *Journal of Geographical Systems*. 2014;16(3):287-309.
4. Yu C, Lee J, Munro-Stasiuk M. Extensions to least-cost path algorithms for roadway planning. *International Journal of Geographical Information Science – GIS*. 2003;17.
5. Jamali AA, Esmailian A, Mokhtarisabet S, He S. Path selection by topographic analysis: vector re-

- classification versus raster fuzzification as spatial multi-criteria using cost-path. *Spatial Information Research*. 2023;31.
6. Stefano B, Davide G, Francesco O. Routing of power lines through least-cost path analysis and multicriteria evaluation to minimise environmental impacts | Request PDF: *Environmental Impact Assessment Review*. 2011;31(3):234-239.
7. Monteiro C, Ramírez-Rosado I, Miranda V, Zorzano-Santamaria PJ, García-Garrido E, Fernandez-Jimenez L. GIS spatial analysis applied to electric line routing optimization. *Power Delivery, IEEE Transactions*. 2005;20:934-942.
8. Antikainen H. Comparison of different strategies for determining raster-based least-cost paths with a inimum amount of distortion. *Transactions in GIS*. 2013;17.
9. Dana T. Propagating radial waves of travel cost in a grid. *International Journal of Geographical Information Science*. 2010;24(9):1391-1413.

REFERENCES

1. Automatic pipeline route design with multi-criteria evaluation based on least-cost path analysis and line-based cartographic simplification: case study of the Mus project in Turkey [Internet]. 2023 [cited 2023 Oct 23]. Available from: <https://www.mdpi.com/2220-9964/8/4/173>
2. Kang JY, Lee B. Optimisation of pipeline route in the presence of obstacles based on a least cost path algorithm and laplacian smoothing. *International Journal of Naval Architecture and Ocean Engineering*. 2017;9.
3. Scaparra MP, Church RL, Medrano FA. Corridor location: the multi-gateway shortest path model. *Journal of Geographical Systems*. 2014;16(3):287-309.
4. Yu C, Lee J, Munro-Stasiuk M. Extensions to least-cost path algorithms for roadway planning. *International Journal of Geographical Information Science – GIS*. 2003;17.
5. Jamali AA, Esmailian A, Mokhtarisabet S, He S.

- ing. 2017;9.
3. Scaparra MP, Church RL, Medrano FA. Corridor location: the multi-gateway shortest path model. *Journal of Geographical Systems*. 2014;16(3):287-309.
4. Yu C, Lee J, Munro-Stasiuk M. Extensions to least-cost path algorithms for roadway planning. *International Journal of Geographical Information Science – GIS*. 2003;17.
5. Jamali AA, Esmailian A, Mokhtarisabet S, He S.

Path selection by topographic analysis: vector re-classification versus raster fuzzification as spatial multi-criteria using cost-path. *Spatial Information Research*. 2023;31.

6. Stefano B, Davide G, Francesco O. Routing of power lines through least-cost path analysis and multicriteria evaluation to minimise environmental impacts | Request PDF: *Environmental Impact Assessment Review*. 2011;31(3):234-239.
7. Monteiro C, Ramirez-Rosado I, Miranda V, Zorzano-Santamaria PJ, Garcia-Garrido E, Fernandez-

Jimenez L. GIS spatial analysis applied to electric line routing optimization. *Power Delivery, IEEE Transactions*. 2005;20:934-942.

8. Antikainen H. Comparison of different strategies for determining raster-based least-cost paths with a minimum amount of distortion. *Transactions in GIS*. 2013;17.
9. Dana T. Propagating radial waves of travel cost in a grid. *International Journal of Geographical Information Science*. 2010;24(9):1391-1413.

Информация об авторе:

Кузьмин Дмитрий Владимирович – кандидат технических наук, доцент кафедры «Логистика и управление транспортными системами» Российского университета транспорта, Author-ID-РИНЦ: 793979; Scopus-Author ID: 57208305058.

Kuzmin Dmitry Vladimirovich – Candidate of Technical Sciences, Associate Professor of the Department of Logistics and Management of Transport Systems at Russian University of Transport, Author-ID-RSCI: 793979; Scopus-Author ID: 57208305058.

**Статья опубликована в режиме Open Access.
Article published in Open Access mode.**

Статья поступила в редакцию 14.07.2025; одобрена после рецензирования 01.09.2025; принята к публикации 28.07.2025. Рецензент – Волохов Г.М., доктор технических наук, главный научный эксперт динамики и прочности подвижного состава и инфраструктуры Акционерного общества «Научно-исследовательский и конструкторско-технологический институт подвижного состава, член редколлегии журнала «Транспортное машиностроение».

The article was submitted to the editorial office on 14.07.04.2025; approved after review on 01.09.2025; accepted for publication on 28.08.2025. The reviewer is Volokhov G.M., Doctor of Technical Sciences, Chief scientific expert of Rolling Stock and Infrastructure Dynamics and Strength of "Research and Development and Technological Institute of Rolling Stock", member of the Editorial Council of the journal *Transport Engineering*.