

# Информатика и системы управления

УДК 51-7

В.В. Фурсов, С.Г. Самохвалова

## АНАЛИЗ АЛГОРИТМОВ ПОИСКА ПУТИ ДЛЯ РАЗЛИЧНЫХ ЗАДАЧ И ИХ ИСПОЛЬЗОВАНИЕ В МОБИЛЬНОМ ПРИЛОЖЕНИИ

*В статье проводится анализ существующих алгоритмов поиска пути для разных типов задач, история их создания и модификации. Реализация и тестирование алгоритмов на мобильном устройстве.*

*Ключевые слова: поиск пути, граф видимости, сетка навигации, путевые точки, мобильное приложение.*

## ANALYSIS PATHFINDING ALGORITHMS FOR VARIOUS PROBLEMS AND THEIR USE IN MOBILE APPLICATIONS

*The existing pathfinding algorithms for different types of tasks and history of creation and modification, implementation and testing the algorithms on a mobile device are analyzed in this paper.*

*Key words: pathfinding, search path, visibility graph, navigation grid, waypoints, mobile app.*

### Введение

Поиск пути является основой для применения в области GPS [1], видеоиграх [2], робототехнике [3], логистике и моделировании толпы [4]. Поиск пути может быть реализован также в статических, динамических и в средах реального времени. В течение последних двух десятилетий удалось повысить точность и эффективность методов поиска путей, но задача по-прежнему привлекает большое внимание исследователей. В настоящее время наиболее важными являются высокая производительность и реалистичные пути для пользователей.

Рассмотрим подробнее представленные на конференции [5] отношения между различными типами топологий местности (рис. 1).

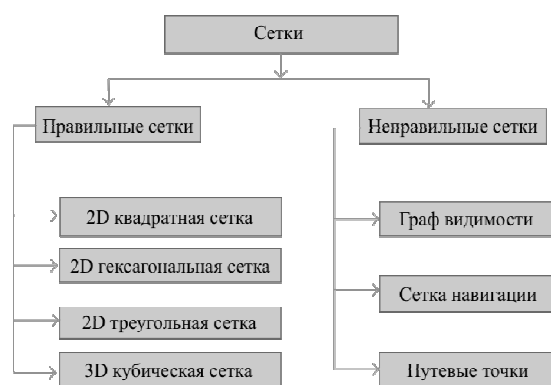


Рис. 1. Топология местности.

### Методы на основе сетки

Сетка состоит из вершин или точек, которые соединены ребрами и представляют собой граф. В большинстве алгоритмов поиска пути производительность навигации – это основной атрибут представления графа. Фундаментальной концепцией являются два популярных подхода, основанных на правильных и неправильных сетках.

## Правильные сетки

Правильные сетки – один из самых известных типов графов, их широко используют. В 2D- и 3D-средах регулярные сетки описывают мозаику правильных многоугольников (т.е. равносторонних и равноугольных полигонов). Шестиугольники, квадраты и треугольники являются единственными правильными многоугольниками, которые могут быть использованы как непрерывная мозаика в 2D (рис. 2, а, б, в) и кубические решетки в 3D (рис. 2, г).

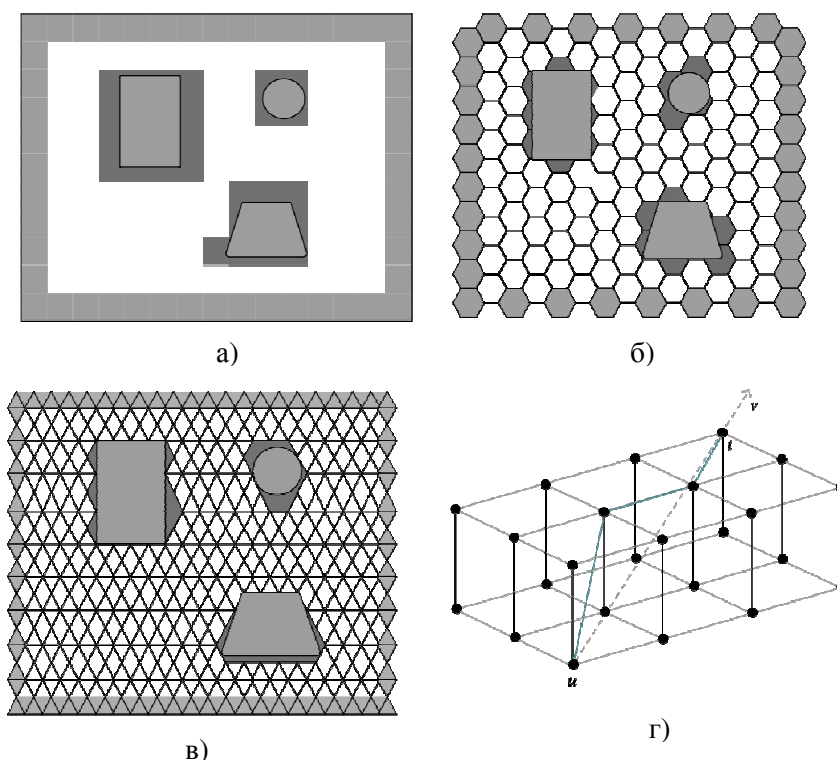


Рис. 2. Правильные сетки: а – квадратные сетки с тремя препятствиями; б – гексагональные сетки с тремя препятствиями; в – триангуляционные сетки с тремя препятствиями; г – кубическая решетка без препятствий.

## 2D-квадратная сетка (рис. 2, а)

Квадратные сетки – одни из наиболее популярных графов в компьютерных играх и навигации по карте, а также многочисленные алгоритмы были предложены для решения задачи поиска пути для этого типа сеток (рис. 2, а). Д. Харабор и А. Грастен [6] предложили алгоритм «прыжковых точек» (jump point search) с одним агентом для решения общей задачи в играх и навигации – хорошо известной «единичной стоимости сетки в статической среде». Jump point search в 10 раз быстрее, чем алгоритм «А\*», и имеет небольшие накладные расходы на память. Исследователи оценивали свою работу на основе стандартного набора карт для поиска пути, предложенного Н.Р. Стертевант [7].

Харабор и Грастен [8] улучшили алгоритм «прыжковых точек» (jump point search), используя три стратегии: во-первых, они рассматривали узлы как базовые блоки операций, а затем применили систему предварительной обработки в автономном режиме, третья стратегия представляет собой усовершенствованный набор действующих правил обрезки деревьев. Полученные алгоритмы сравнивались с методом Ураса.

В отношении изменяющихся графов в присутствии «динамических препятствий» К. Андерсон [9] представил отличное дополнение эвристики для поиска с единственным агентом на 4-направленной квадратной сетке. Этот метод эффективно сокращает время поиска в неясных графах, единственный его недостаток – он реализован с алгоритмом «А\*», излишне расширяющего число узлов во время поиска.

Для мультиагентной задачи Х. Джин и др. [10] использовали алгоритм «А\*» в ХНА-игре, чтобы решить проблему поиска пути. Они использовали алгоритм «А\*» для обработки динамических препятствий с 2D-квадратной картой сетки, несмотря на 3D-реализацию. Недостаток их работы – то, что «А\*» не всегда является оптимальным для статических карт сетки.

*2D-гексагональная сетка (рис. 2, б)*

Ю. Бьорнссон и др. [11] показали, что гексагональные сетки (рис. 2, б) имеют много свойств квадратных сеток. Кроме того, у шестиугольных сеток меньше время поиска и потребление памяти, чем у сложных сеток графов, построенных из квадратов. П.Ж. Кихано и Л. Гарридо [12] использовали 6 алгоритмов гексагональной сетки для моделирования разведки робота. Они доказали, что алгоритмы на гексагональных сетках превосходят квадратные сетки для одно- и мультиагентных задач.

М.Ф. Осман и др. [13] обсудили проблемы, с которыми сталкиваются исследователи, использующие алгоритмы поиска пути в робототехнике. Одна из главных проблем в мобильной робототехнике – избегание реальных объектов. Алгоритм  $D^*$  был применен к методам моделирования 2D-карты с помощью квадратной плитки, шестиугольников и усовершенствованных шестиугольников. Результаты показывают, что усовершенствованные шестиугольники дают лучшие пути, но требуют больше времени для поиска в больших картах. Приемлемые результаты достигаются для малых и средних карт. Алгоритмы поиска пути были реализованы в основном на квадратных сетках, а мультиагентная задача на шестиугольной сетке еще не была исследована в динамических средах или в средах реального времени.

*2D-треугольная сетка (рис. 2, в)*

Треугольные сетки (рис. 2, в) используются не так широко, как квадратные и гексагональные, но у них есть некоторые хорошие свойства. Д. Димайном и М. Буро [14] был предложен способ, который уменьшает затраты на поиск с использованием ограничений триангуляции Делоне. Изменяя размеры объектов, авторы исследовали влияние окружающей среды на движение во время навигационной задачи. При использовании их алгоритмов  $TA^*$  и  $TRA^*$  обнаружено, что они безупречно работают на больших картах.

Наги [15] получил новую систему координат, состоящую из двух гексагональных и треугольных сеток графов. Эта новая сетка может применяться в различных областях, – например, таких как компьютерная графика и обработка изображений. Считается, что новую топологию можно использовать с детерминированными методами поиска пути – такими как  $A^*$  и его варианты.

*3D-кубическая сетка (рис. 2, г)*

В отличие от сеточных графов, рассмотренных выше, кубическая решетка (рис. 2, г) является привычным графом для 3D-среды. Кылыч и Ялчин использовали простой алгоритм для вычисления движения трех типов волн в 3D-среде и применили его, чтобы решить проблему поиска пути для робота. Позже авторы улучшили алгоритм для среды 3D. В предложенной модели робот имеет всего 6 соседних клеток, которые могут быть использованы для навигации. Это явный недостаток данной модели, число соседних клеток должно быть увеличено, чтобы включать все 26 соседних ячеек для улучшения получившегося пути. Не так давно А. Нэш и С. Кениг [16] опубликовали отличную статью под названием «Any-Angle Path Planning». Они ввели три различных алгоритма –  $A^*$ ,  $Theta^*$  и Ленивая  $Theta^*$  – для квадратных, шестигранных, треугольных и кубических сеток. Цель этих алгоритмов – нахождение плавного реального пути для любой топологии местности, без учета размера агента или формы препятствий. На основании наблюдаемой производительности они определили кратчайшие пути, а не самый короткий путь на сетке. К сожалению, алгоритм «Any-Angle Path Planning» потребляет больше времени, чем обычные алгоритмы поиска пути.

### Неправильные сетки

Аномальные сетки используются во многих задачах и областях.

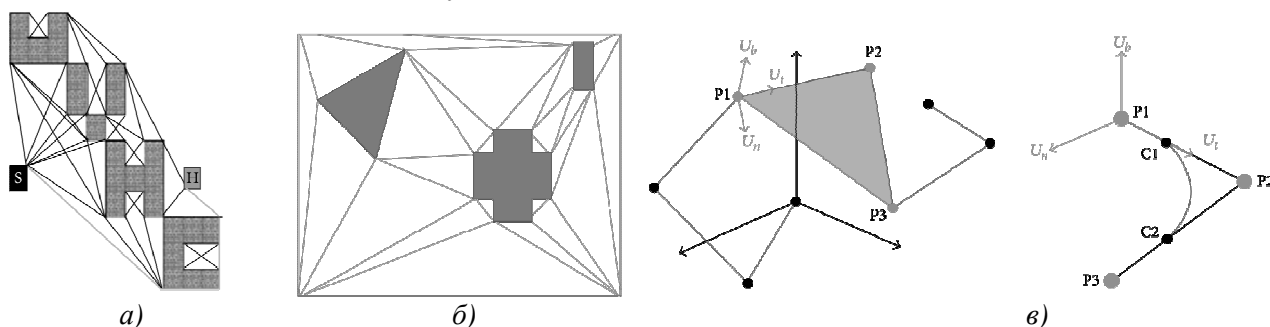


Рис. 3. Неправильные сетки: а – граф видимости; б – сеточная система навигации с треугольной сеткой; в – 3D пути сглаживания.

#### Граф видимости (рис. 3, а)

Граф видимости считается фундаментальной структурой в различных областях геометрической теории графов и в вычислительной геометрии, его привлекают к изучению различных задач. Граф видимости недавно был использован при вычислении евклидовых кратчайших путей при наличии препятствий [17].

Чтобы решить проблему с одним агентом, М. Надзан-Тахан и Т. Манзури-Шалмани [18] предложили новый генетический алгоритм для поиска эффективного пути через ряд расширяющихся препятствий. Кроме того, они использовали стратегию ускорения скорости сходимости путем создания подходящей начальной популяции. Авторы применили общую вычислительную геометрию на основе подхода суммы Минковского, а не на основе методов ячеек – таких, как сетки. Этот метод показал, что он быстрее, чем другие подходы вычислительной геометрии, но производит неоптимальные решения.

#### Сетка навигации (рис. 3, б)

Как показано на рис. 3, б, проходимые участки карты представляют собой навигационную сетку. Эта сетка может быть представлена разными способами, – например, с помощью треугольников или многоугольников. Ведь сеточные графы и графы видимости очень похожи, но в практическом плане графы видимости являются более сложными, чем сеточные. Чаще всего сетки графов применяются в видеоиграх.

Сислак и др. представили алгоритм, вдохновленный  $A^*$ . Их алгоритм  $AA^*$  (Accelerated  $A^*$ ) позволяет планировать траекторию полета на основе задачи поиска пути с единственным агентом. Алгоритм  $AA^*$  был использован в динамичной среде для моделирования движения самолета. Основные его преимущества – способность рассматривать самолет ненулевого размера и планировать путь для направленного вектора вверх от горизонтальной плоскости.

С целью мультиагентного поиска пути Кападия и др. предложили структуру режима реального времени для мультиагентной навигации, которая использует несколько неоднородных проблемных областей в больших, сложных и динамических виртуальных средах.

#### Путевые точки (рис. 3, в)

Путевые точки широко используются в компьютерных играх и робототехнике. Нидерберг и др. предложили алгоритм на основе  $A^*$  для поиска пути в статических местностях с полигональными препятствиями. Алгоритм генерирует путь в соответствии с минимальным числом точек пути, но характеризуется длительным временем вычислений и значительными накладными расходами памяти.

Фергюсон и Стентз разработали алгоритм Field  $D^*$  с интерполяцией на основе планирования и описали алгоритм перепланировки для генерации малозатратных путей через однородные и неод-

народные сетки. Есть два недостатка в поиске пути на основе сетки: во-первых, планирование на основе сетки имеет ограниченную способность в нахождении оптимального пути, потому что путь должен проходить через соседние точки; во-вторых, требования к памяти на основе планирования пути часто неприемлемо высоки.

### Использование алгоритма поиска пути для навигации по зданию в мобильном приложении

После проведенного исследования алгоритмов и модификаций было решено протестировать их производительность на мобильном устройстве. Для этого было написано соответствующее программное обеспечение с использованием кроссплатформенного фреймворка Xamarin. Для теста использовалась карта, созданная по плану третьего этажа главного корпуса АмГУ (рис. 4).

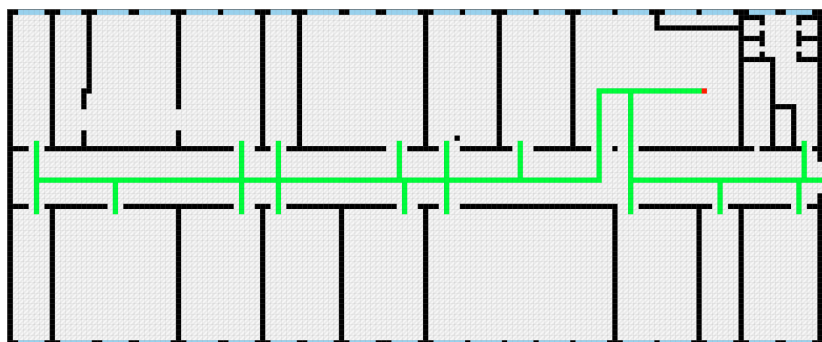


Рис. 4. Частичный вид карты третьего этажа главного корпуса АмГУ.

Для поиска пути по зданию было решено использовать в качестве карты правильную 2D-квадратную сетку как наиболее подходящую для текущей задачи.

### Тестирование

При тестировании учитывались такие параметры как количество итераций, время расчета пути. Все замеры воспроизвелись на устройстве iPhone 5 с двухъядерным процессором на 1,3 ГГц и оперативной памятью в 1 Гигабайт.

### Результаты тестов

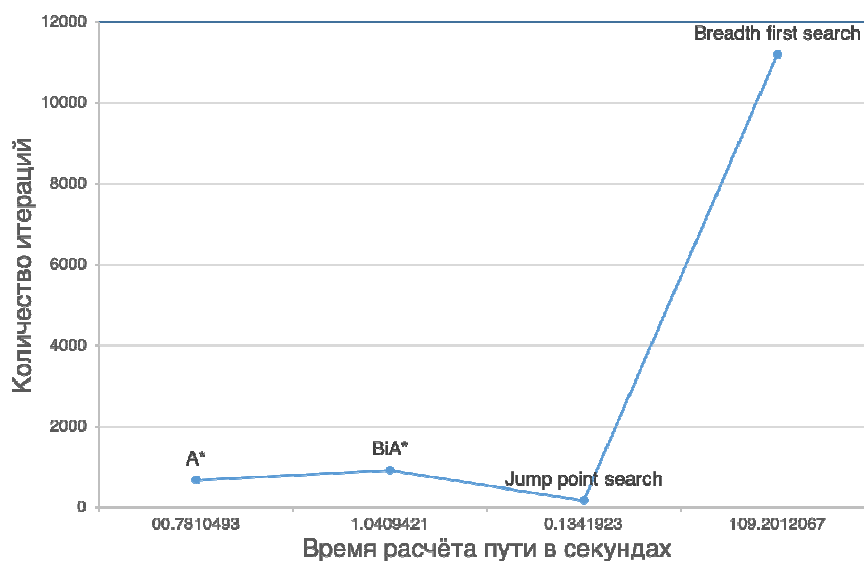


Рис. 5. Замеры для неоптимизированной карты.

Таблица 1

## Замеры для неоптимизированной карты

Алгоритм	Время, сек.	Количество итераций
A*	00.7810493	686
BiA*	01.0409421	927
Jump point search	00.1341923	182
Breadth first search	109.2012067	11203

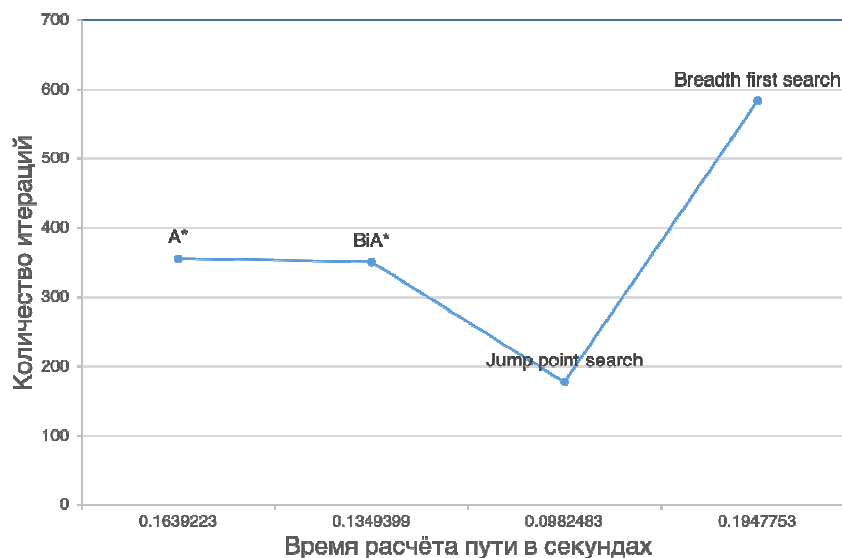


Рис. 6. Замеры для оптимизированной карты.

Таблица 2

## Замеры для оптимизированной карты

Алгоритм	Время, сек.	Количество итераций
A*	00.1639223	356
BiA*	00.1349399	351
Jump point search	00.0982483	178
Breadth first search	00.1947753	583

## Заключение

В результате проведенного анализа были изучены различные алгоритмы поиска путей для разных типов задач. В итоге лучшим алгоритмом для поиска путей на правильной 2D-квадратной сетке, на мобильном устройстве, с минимальным временем расчета пути стал алгоритм «прыжковых точек» (jump point search), который и используется для создания навигационного мобильного приложения. Разработанное мобильное приложение поможет гостям и студентам в навигации внутри зданий университета.

1. Стертевант, Н.Р., Гайсбергэ, Р. Сравнение на высоком уровне подходов для ускорения поиска пути // Труды 6 АААІ конференции по искусственному интеллекту и интерактивным цифровым развлечениям (АИДЕ '10), 2010. – С. 76-82.

2. Коливанд, Х., Сунар, М. С. Обзор алгоритмов теневого объема в компьютерной графике // IETE Technical Review. – 2010. – Т. 30, № 1. – С. 38-46.

3. Ван Ден Берг, Й., Шах, Р., Хуан, А., Гольдберг, К. ANA \*: всегда непараметрический A // Труды XXV АААІ конференции по искусственному интеллекту (АААІ '11), 2011. – С. 105-111.

4. Бонет, Б., Гефнер, Х. Планирование в виде эвристического поиска // Искусственный интеллект. – 2001. – Т. 129, № 1-2. – С. 5-33.

5. Фурсов, В.В., Самохвалова, С.Г. Обзор алгоритмов поиска пути для мобильного приложения // XXXVI Международная научно-практическая конференция «Приоритетные научные направления: от теории к практике». – Новосибирск: Изд-во ЦРНС, 2017. – С. 245-251.
6. Харабор, Д., Грастен, А. Обрезка графа для поиска пути на сетке // Труды XXV национальной конференции по искусственному интеллекту (AAAI '11). – Сан-Франциско, штат Калифорния, США, 2011.
7. Стертевант, Н.Р. Критерии для поиска пути на основе сетки // IEEE Transactions по вычислительному и искусственному интеллекту в играх. – 2012. – Т. 4, № 2. – С. 144-148.
8. Харабор, Д., Грастен, А. Улучшение jump point search // Труды XXIV Международной конференции по вопросам автоматизированного планирования и составления расписаний, 2014.
9. Андерсон, К. Добавление эвристики для соединенных четырех сеток // Труды III ежегодного симпозиума по комбинаторному поиску, 2010.
10. Джин, Х., Вэй, У., Зиян, Л. Многоагентная система поиска пути, реализованная на XNA // Труды IV Международной конференции по вычислительному интеллекту и коммуникационным сетям (CICN '12). – Матхура, Индия, 2011. – С. 651-655.
11. Бьорнссон, Ю., Ензерберг, М., Холтэ, Р., Шафер, Ж., Яп, П. Сравнение различных абстрактных сеток для поиска пути на картах // Труды XVIII Международной совместной конференции по искусственному интеллекту (IJCAI '03). – Акапулько, Мексика, 2003. – С. 1511-1512.
12. Кихано, Х.Ж., Гарридо, Л. Совершенствование объединения робота с использованием шестиугольного мирового представления // Труды конференции по электронике, робототехнике и автомеханике (CERMA '07), 2007. – С. 450-455.
13. Осман, М. Ф., Самади, М., Асл, М. Х. Моделирование динамического планирования пути в режиме реального времени, базовое зрение роботов // Интеллектуальные робототехнические системы: вдохновляя в БУДУЩЕЕ . 2013.– С. 1-10.
14. Димайн, Д., Буро, М. Эффективный поиск пути на базе триангуляции // XXI национальная конференция по искусственному интеллекту и XVIII конференция по новаторскому использованию искусственного интеллекта. 2006. – С. 942-947.
15. Наги, Б. Клеточная топология и топологические системы координат на гексагональных и на треугольных сетках // Ежегодник по математике и искусственному интеллекту. – 2014. – С. 1-18.
16. Нэш, А., Кениг, С. Алгоритм «Any-angle path planning» // Журнал по искусственному интеллекту. – 2013. – Т. 34, № 4. – С. 9.
17. Лозано-Переса, Т., Уэсли, М. А. Алгоритм для планирования пути без столкновений между многогранными препятствиями // ACM. – 1979. – Т. 22. – С. 560-570.
18. Надэан-Тахан, М., Манзури-Шалмани, Т. Эффективное и безопасное планирование пути для мобильного робота с использованием генетического алгоритма // IEEE-конгресс по эволюционным вычислениям (CEC '09), 2009. – С. 2091-2097.