|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ***

***НА ТЕМУ:***

***Алгоритм поиска кратчайшего безопасного пути на электронной карте местности***

Студент \_\_ИУ7-84\_\_\_\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_**Д.И. Монахов**\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель ВКР **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_**М. Ю. Барышникова\_

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

Нормоконтролер **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_**Ю.В. Строганов**\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2020 г.*

Введение

Проблема определения маршрута была актуальной еще в древности и является актуальной по сей день. Раньше человеку приходилось использовать карту для поиска оптимального по времени пути, чтобы достичь желаемого пункта назначения. В наши дни мало что изменилось, нам по-прежнему нужно перемещаться и очевидное желание сэкономить время и топливо никуда не исчезло. К тому же начинают появляться роботизированные средства передвижения, такие как беспилотные автомобили, беспилотные летательные аппараты (БПЛА). В военной промышленности также производятся крылатые ракеты с рельефометрическим наведением, которое использует карты для прокладывания маршрута полета таким образом, чтобы не попасть под огонь враждебных средств противоракетной обороны. Согласно проведенным экспериментам, данный тип наведения показывает отличные результаты в преодолении средств защиты и позволяет ракете обладать высокими показателями живучести и помехозащищенности [1]. Наведение по рельефу в комплексе с инерциальными средствами навигации позволяет аппарату быть практически невосприимчивым к средствам радиоэлектронной борьбы, т.к. исключает использование радиосвязи в навигации. Автоматизация процесса прокладывания маршрута произошла и в гражданской сфере: практически на каждом мобильном компьютере (смартфоне) имеются программно-аппаратные средства для отображения картографических данных, геопозиционирования пользователя и помощи в прокладывании маршрута. Также технология используется в бизнесе. Самый яркий пример – это логистика. Оптимизация пути является основной задачей логистики, от успешности выполнения которой зависит прибыль компании. Стоимость маршрута может зависеть от множества параметров, таких как: время, расстояние, расход топлива и т.д. А сами эти параметры могут зависеть от характеристики ландшафта, по которому прокладывается путь. [2]

Все рассмотренные выше приложения требуют программных средств поиска кратчайшего пути на электронной карте местности, который удовлетворяет некоторым заданным пользователем требованиям. Например, гражданин хочет избежать использования нерегулируемых перекрестков, логистическая компания не готова оплачивать платные дороги, а крылатая ракета не должна попасть в зону покрытия вражеской противоракетной обороны. Исходя из этого, данная работа предлагает рассмотреть применение алгоритмов поиска кратчайшего пути с возможностью исключения неблагоприятных участков пути. Под безопасностью пути будет пониматься оценка пользователя программного продукта о допустимости включения участка в конечный план маршрута.

И так, целью работы является модификация алгоритма поиска кратчайшего пути для решения проблемы прокладывания маршрута на карте местности с учетом требований пользователя по исключению специфических участков.

Для достижения поставленной цели необходимо решить следующие задачи:

* исследовательская (выбор существующего алгоритма поиска кратчайшего пути)
* конструкторская (модификация выбранного алгоритма для решения проблемы)
* технологическая (выбор средства отображения картографических данных, выбор среды и языка программирования, описание основных моментов разработки).

Аналитический раздел

Рассмотрим существующие алгоритмы поиска кратчайшего пути. Все приведенные ниже алгоритмы основываются на представлении пространства в виде графа, в котором вершины являются точками пространства, а рёбра отображают наличие возможности переместится из одной точки в другую.

Сеточная карта (дискретное рабочее поле)

Для начала следует рассмотреть алгоритмы, работающие с сеточным представлением карты местности.

Карту можно представить в виде сетки (или таблицы), состоящей из проходимых и непроходимых ячеек. Каждая ячейка соответствует некоторой области в реальном мире и имеет две координаты (x, y), совпадающие с положением ячейки в таблице.

Рассмотрим построение такой сетки на примере фрагмента карты. Непроходимыми будем считать те ячейки, которые более чем на 50% заполнены какими-либо объектами реального мира. Эти ячейки закрасим красным цветом, а затем выделим получившуюся таблицу как отельное изображение (рис 1).



Рисунок . Пример создания сеточной карты

Такое представление карты уже достаточно легко анализировать. Очевидно, что в данном примере использовалась слишком низкая частота дискретизации пространства, в конечном программном продукте следует повысить точность.

**Алгоритм Ли**

Существуют алгоритмы, которые работают с форматом входных данных в виде дискретного рабочего поля. Самый известный такой алгоритм – это алгоритм Ли [[3](http://users.eecs.northwestern.edu/~haizhou/357/lec6.pdf)]. Рассмотрим принцип работы данного алгоритма. Он основан на алгоритме поиска в ширину и его сильной стороной является гарантия нахождения кратчайшего пути между двумя точками, если такой путь существует.

Входные данные: дискретное рабочее поле с помеченными непроходимыми ячейками, пара точек (старт и финиш).

Выходные данные: последовательность ячеек, представляющая кратчайший путь из точки старта в точку финиша, если такой путь существует.

Работа алгоритма делится на три этапа: инициализацию, распространение волны и восстановление пути.

1) Инициализация: построение множества ячеек, запоминание стартовой и финишной ячейки.

2) Распространение волны: порождается шаг в соседнюю ячейку с проверкой проходимости и принадлежности к уже рассмотренным ячейкам.

3) Соседняя ячейка помечается как рассмотренная и получает в соответствие число, равное количеству шагов от стартовой ячейки.

4) Из каждой рассмотренной на данном шаге ячейки порождается новая волна, которая трактует данную ячейку как стартовую.

5) При достижении конечной ячейки проводится восстановление пути. Начиная с конечной ячейки ищется соседняя ячейка, имеющая в соответствие число на единицу меньшее чем у текущей. Последовательность этих ячеек и становится результатом работы алгоритма (после инверсии составленного списка).

Ниже приведен псевдокод алгоритма:

// Инициализация

Пометить стартовую ячейку

d := 0

// Распространение волны

ЦИКЛ

ДЛЯ каждой ячейки loc, помеченной числом d

пометить все соседние свободные непомеченные ячейки числом d + 1

КЦ

d := d + 1

ПОКА (финишная ячейка не помечена) И (есть возможность распространения волны)

// Восстановление пути

ЕСЛИ финишная ячейка помечена

ТО

перейти в финишную ячейку

ЦИКЛ

выбрать среди соседних ячейку, помеченную числом на 1 меньше числа в текущей ячейке

перейти в выбранную ячейку и добавить её к пути

ПОКА текущая ячейка — не стартовая

ВОЗВРАТ путь найден

ИНАЧЕ

ВОЗВРАТ путь не найден

Следует заметить, что соседство ячеек может быть двух вариантов: в смысле окрестности Мура и окрестности фон Неймана. В первом случае, в отличие от второго, рассматриваются не только ближайшие вертикальные и горизонтальные ячейки, но также диагональные. Принцип работы алгоритма никак не зависит от определения соседства.

Однако у алгоритма Ли есть недостаток - данный алгоритм является очень требовательным ко времени и памяти. Временная сложность и требование к памяти для сетки с размерами M x N составляет O(MxN), т.е. с увеличением размеров исследуемой области требования алгоритма растут в квадратической зависимости.

Способы сократить время выполнения:

1) выбирать в качестве начальной точки ту, которая находится дальше от центра

2) распространять волну и из начальной точки и из конечной

3) исследовать как можно меньшую прямоугольную область, включающую входные точки

Можно сказать, что данный алгоритм слабо подходит к поставленной задаче. Не смотря на гарантированность результата, он является слишком медленным и требовательным к памяти. Следует рассмотреть другие алгоритмы поиска кратчайшего пути, которые будут более эффективны по этим параметрам.

Преобразование в граф

Теперь рассмотрим способ превращения карты местности в графовое представление. Реальный мир является не дискретным пространством, что исключает возможность бесконечно точного хранения его реплики в памяти компьютера. Однако для решения задач прокладывания маршрутов это не понадобится. Достаточно разбить карту на равномерную сетку, где каждая проходимая ячейка представляется вершиной графа, а непроходимые участки (например, стены) удаляют соответствующие рёбра между вершинами (см. рис 1)



Рисунок . Графовое представление пространства

Такое представление пространства позволит применить классические алгоритмы обхода графов, которые и будут рассмотрены ниже.

Поиск в ширину

Поиск в ширину – метод обхода графов, который последовательно обходит каждый уровень графа, начиная с исходной вершины. Рассмотрим последовательность действий данного алгоритма.

1. Поместить исходную вершину в очередь

2. Извлечь вершину x из очереди и пометить как посещенный

а. Если узел x является целевым, то завершить поиск как успешный

б. Иначе, добавить в очередь все неотмеченные вершины, которые связаны с x и не находятся в очереди

3. Если очередь пуста, то завершить поиск как неудачный

4. Перейти в п. 2

Список литературы

1. Prepared Statement of Capt. Walter M. Locke, U.S. Navy, Tomahawk Cruise Missile Project Manager. Hearings before the Subcommittee on Research and Development of the Committee on Armed Services, United States Senate, March 31, 1977. — Washington, D.C.: U.S. Government Printing Office, 1977. — 6454 p.   
   Ссылка: <https://babel.hathitrust.org/cgi/pt?id=mdp.39015013772390&view=1up&seq=349>
2. Engineering Route Planning Algorithms. Daniel Delling, Peter Sanders, Dominik Schultes, and Dorothea Wagner. Universit ̈at Karlsruhe (TH), 76128 Karlsruhe, Germany  
   Ссылка: <https://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=82FD69BD60193407E0159A0844367A21?doi=10.1.1.164.8916&rep=rep1&type=pdf>