

Об одном методе построения оптимального маршрута

Гилева Е.А., Пономарев И.В.

Алтайский государственный университет, г. Барнаул

molniy1111@gmail.com, igorpon@mail.ru

Аннотация

Построение оптимальной логистической сети имеет несомненный интерес как с точки зрения минимизации транспортных расходов, так и относительно времени доставки грузов. Данная работа посвящена построению программного комплекса для получения оптимального транспортного пути между заданными точками (магазин – склад). Основой данного комплекса является сеть Штейнера, что придает исследованию строгое математическое обоснование.

Ключевые слова: сеть Штейнера, решетка Ханана, транспортная сеть, математическое моделирование.

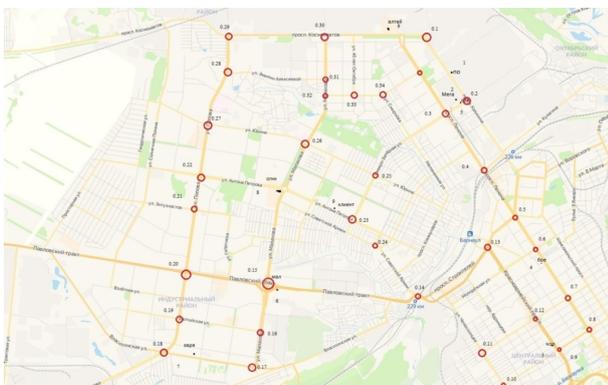
Классическая постановка задачи Штейнера формулируется следующим образом: на плоскости есть конечное число точек, которые нужно соединить с системой отрезков с наименьшей общей длиной так, чтобы из всех точек по этим отрезкам можно было попасть в другую. Поиском решений данной задачи занимались такие известные ученые, как П. Ферма, Э. Торричелли, Т. Симпсон, В. Ярник и О. Кесслер [1]. Результатом данных исследований стало большое число разнообразных алгоритмов решения данной задачи, каждый из которых имеет как свои достоинства, так и ряд недостатков.

В основу нашего программного комплекса был положен алгоритм сетки Ханана. В геометрии сетка Ханана $H(S)$ получается ограниченным набором S точек на плоскости путем проведения вертикальных и горизонтальных линий через каждую точку. Приведем краткий алгоритм данного метода:

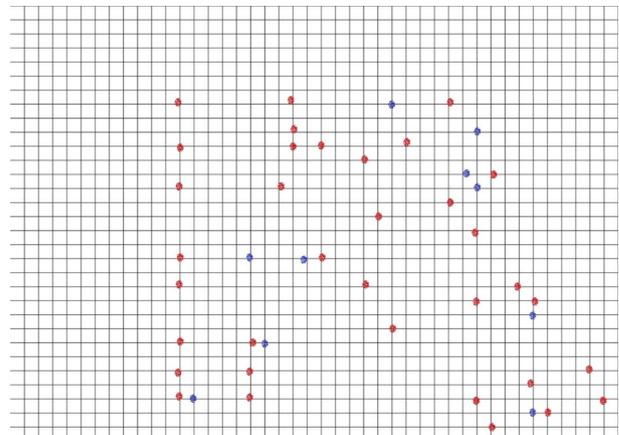
1. Все множество вершин $x_i \in X$ с координатами $(s_i; t_i)$ разбивается на классы S_0, S_1, \dots, S_m в порядке возрастания первой координаты так, чтобы у всех вершин одного класса была одинаковая координата s_i .
2. Анализируется множество вершин класса S_0 и соединяется между собой прямой.
3. Образуется множество I , состоящее из вершин, включенных в деревья Штейнера и точек, принадлежащих построенной прямой.
4. Выбирается следующее множество, имеющее наименьшую координату. Для вершины $x_k \in S_{i+1}$ определяется точка $x_j \in X$, для которой $d_{rj} = \min |d_{kg}|, x_g \in I$. Вершина X_k соединяется двухзвенной ломаной линией с точкой $x_j \in I$.
5. Для очередной вершины класса S_{i+1} операция подсоединения к дереву Штейнера выполняется в соответствии с пунктами 3, 4.
6. Пункты 3, 4, 5 повторяются для всех множеств S_j до построения полного дерева Штейнера.

Данный алгоритм относится к классу адаптивных алгоритмов с управляемой точностью решения и использованием параметрической адаптации к внешним условиям его выполнения: размерности задачи, ресурса времени, отведенном для выполнения проектной операции, а также быстродействия компьютера, на котором выполняется программа алгоритма, направлен на получение наиболее точного решения за отведенное время на основе выбора значения параметра адаптации [2].

Поставим задачу построения оптимального маршрута между целевыми точками с расчетом времени и подсчет километража. В роли целевых точек выступали адреса магазинов, складов и постоянных покупателей одной из крупных торговых сетей г. Барнаула. Для обеспечения соблюдения правил дорожного движения были добавлены 50 вспомогательных точек, указывающих на основные транспортные развязки и ограничения на передвижения крупногабаритных транспортных средств. Заметим, что точность построения маршрута напрямую зависит от количества вспомогательных точек, а также от правильного отслеживания новых дорожных знаков. Эти данные были перенесены на решетку Ханана (см. рисунок 1).



а)



б)

Рисунок 1. Выбранные для моделирования целевые и связующие точки: а) на карте г. Барнаула; б) на решетке Ханана

Для написания программного комплекса будем использовать язык программирования Pure Basic, который предназначен для создания автономных консольных и графических приложений [3].

Для вычисления нам понадобится широта и долгота каждой точки, чтобы результат приблизить к более достоверному. Эти данные мы записали в отдельном файле, пример записи: [{"lat":53.380567,"long":83.751057,"EndPoint":true,"ID":1,"Name":"Магазин 1"}]. Загружаем список связанных точек. Для расчета среднего времени мы использовали средний коэффициент плотности трафика на отдельных отрезках дороги по данным сервиса Яндекс.Пробки. В результате работы программы пользователь получает результат в виде текстового сообщения (рисунок 2).

```

Оптимальный маршрут построен
1->46->47->12->48->13->14->15->16->17->22->49->5
Длина - 7.11 км
Время - 601.23 сек
Затраты топлива - 0.57 л
Введите 1 если нужно показать маршрут на карте

```

Рисунок 2. Пример выходного сообщения программы

Имеется возможность визуализировать рассчитанный маршрут на карте (рисунок 3).

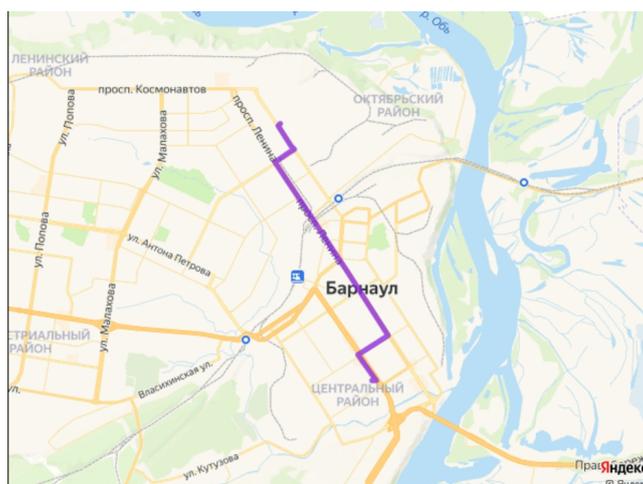


Рисунок 3. Результат отображения на карте полученного маршрута

К разработанному программному комплексу был проявлен интерес со стороны торговой компании, которая заинтересована во внедрении подобной технологии в свою практику перевозок. В ближайшее время планируется расширить количество целевых и связующих точек, а также учесть возможность наличия временных ограничений на маршруте следования.

Список литературы

1. Протасов В.Ю. Максимумы и минимумы в геометрии. — М. : МЦНМО, 2009. — С. 56.
2. Иванов А.О., Тужилин А.А. Задача Штейнера на плоскости или плоские минимальные сети // Матем. сб. — 1991. — Т. 182, № 12. — С. 1813–1844.
3. Willoughby G. Purebasic: A Beginner's Guide to Computer Programming. — NY : Aardvark Global Publishing, 2006.