

ний, агентств и университетов, участвующих в разработке публично доступных концептуальных решений, которые могут быть полезны со всеми видами прикладных программ, которые управляют пространственными данными.

Пространственные данные в MySQL

Модель пространственных данных OGIS может быть встроена в большое количество различных языков программирования, например C, Java, SQL и т. д. Эта модель данных состоит из базового класса GEOMETRY, который является абстрактным и задает пространственную систему координат, применимую ко всем производным от него классам. Четыре основных класса, порожденных от предка GEOMETRY – это Point, Curve, Surface и GeometryCollection. С каждым из этих классов связан набор операций, выполняемых над экземплярами классов.

В MySQL имеются типы данных, которые соответствуют классам OpenGIS. Некоторые из этих типов хранят единственное геометрическое значение (GEOMETRY, POINT).

Некоторые типы данных могут хранить коллекции значений (MULTIPOINT, GEOMETRYCOLLECTION).

MySQL обеспечивает стандартный способ создания пространственных столбцов для типов геометрии, например, через CREATE TABLE или ALTER TABLE: CREATE TABLE geom (g GEOMETRY).

MySQL обеспечивает набор функций, чтобы выполнить различные операции на пространственных данных. Эти функции могут быть сгруппированы в четыре главных категории согласно типу операции, которую они выполняют:

- функции, которые преобразовывают конфигурации между различными форматами;

- функции, которые обеспечивают доступ к качественным или количественным реквизитам геометрии;

- функции, которые описывают отношения между двумя конфигурациями;

- функции, которые создают новые конфигурации из существующих.

Пространственные функции анализа могут использоваться во многих контекстах:

- любая интерактивная программа SQL, типа mysql или MySQL Query Browser;

- прикладные программы, написанные на любом языке, который поддерживает клиентский MySQL API [2].

Заключение

Изучены разновидности ГИС, их назначение и сферы применения. Были рассмотрены основные принципы работы некоторых ГИС. Была разработана схема проекта, а также изучены необходимые для её реализации компоненты. В настоящий момент ведётся активная разработка данной системы.

Литература

1. Компьютер пресс [Электронный ресурс]. - Режим доступа: <http://www.compress.ru/article.aspx?id=10532&iid=431>, свободный.

2. Пространственные расширения [Электронный ресурс]. – Режим доступа: <http://www.rldp.ru/mysql/mysqlpro/spat.htm>, свободный.

3. Коновалова Н.В., Капралов Е.Г. Введение в ГИС: учебное пособие. – М., 1997. – 160 с.

4. Демерс М.Н Географические информационные системы. Основы. – М., 2006. – 246 с.

5. Ковин Р.В, Н.Г.Марков Геоинформационные системы. Учебное пособие, 2008. – 175 с.

МОДИФИКАЦИЯ АЛГОРИТМА ПОИСКА «A STAR» ДЛЯ ПЛАНАРНОЙ ДОРОЖНОЙ СЕТИ С ЦЕЛЬЮ УЧЕТА НАПРАВЛЕНИЯ ДВИЖЕНИЯ НА ПЕРЕКРЕСТКЕ

Максимова Е.И., Хаустов П.А.

Томский политехнический университет

634050, Россия, г. Томск, пр-т Ленина, 30

E-mail: yelenamaksimova@yandex.ru

Введение

Все чаще в современном мире используются системы поиска оптимального маршрута для дорожных сетей. Такие системы нашли широкое применение среди водителей транспортных средств и применяются для нахождения маршрута до некоторой конечной точки, требующего минимального количества времени на его преодоление. Большинство алгоритмов нахождения оптимального пути не являются общедоступными. Однако, существуют и общеизвестные алгоритмы, на ко-

торых основывается большинство подобных систем. Одним из таких алгоритмов является алгоритм «A star».

Алгоритм «A star»

Алгоритм «A star» применяется для поиска маршрута наименьшей стоимости от начальной вершины до выбранной конечной во взвешенном графе.

Для вычисления оптимального пути «A star» использует два параметра рассчитываемых для

каждого узла: наименьший путь $G(v)$ от заданной вершины до вершины v и эвристическую оценку расстояния $H(v)$ от текущей вершины до конечной. Порядок обхода вершин в графе определяется итоговой функцией $F(v) = G(v) + H(v)$, поведение которой зависит от выбранной эвристической оценки $H(v)$. Для планарной дорожной сети в качестве $H(v)$ выбирается евклидово расстояние.

Стоит отметить, что при эвристической оценке $H(v) = 0$ данный алгоритм полностью совпадает с алгоритмом Дейкстры.

В реальной дорожной сети в зависимости от направления движения на перекрестке может потребоваться различное количество времени на преодоление этого перекрестка. Одним из недостатков данного алгоритма, в таком случае, является невозможность учитывать направление движения на перекрестке.

Модификация алгоритма «A star» для учета направления движения на перекрестке

В ходе работы алгоритма «A star» учитываются лишь кратчайшие расстояния до каждого из перекрестков. В данном алгоритме не учитывается, по какой дороге достигнут текущий перекресток. Казалось бы, возможна простая модификация алгоритма, при которой запоминалось бы не только минимальное время пути до каждого из перекрестков, но и дорога, по которой был проложен этот путь. Однако подобная модификация не является корректной. На практике возможна ситуация, при которой можно достичь некоторого перекрестка за минимальное время в одном из направлений, но затем потребуется осуществить поворот налево, что является более затратным по времени, нежели движение прямо или поворот направо. Такая ситуация показана на рисунке 1. Перекресток В можно проехать в различных направлениях. Если двигаться по указанному маршруту, то для достижения перекрестка А потребуются меньше времени. Однако для дальнейшего движения потребуется повернуть налево.

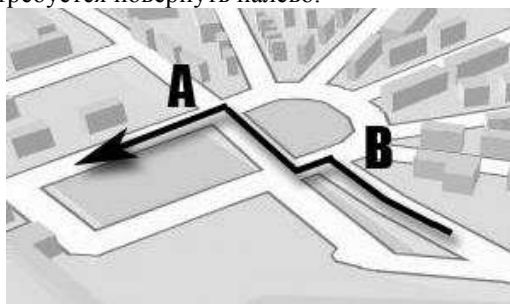


Рис. 1. Пример маршрута с левым поворотом на перекрестке А

Добраться до перекрестка А можно, используя другой маршрут. При движении в таком направлении может потребоваться больше времени, но для продолжения маршрута не придется осуществлять левый поворот. Такая ситуация представлена на рисунке 2. Если проехать перекресток

В в указанном направлении, то на перекрестке А не потребуется осуществлять поворот, а, следовательно, на преодоление этого перекрестка уйдет намного меньше времени.

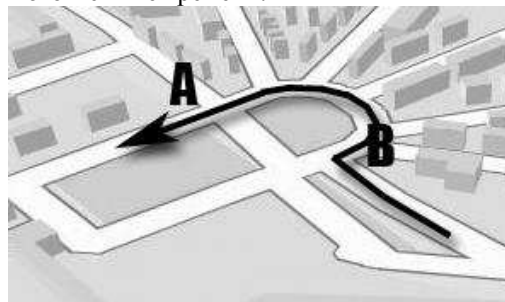


Рис. 2. Пример маршрута без левого поворота на перекрестке А

Как можно заметить, в подобной ситуации недостаточно учитывать только минимальное время, за которое можно достичь тот или иной перекресток. Важно так же учитывать время необходимое для достижения перекрестка в каждом из возможных направлений. Для этого следует несколько модифицировать граф, которым задается дорожная сеть. Каждая вершина в таком графе теперь должна задавать не только некоторый перекресток, но и направление движения, по которому он достигается. То есть каждая вершина графа будет представлять собой пару {идентификатор перекрестка; направление движения до этого перекрестка}. При использовании графа с такой структурой кратчайшее расстояние будет рассчитываться не просто до всех перекрестков, а для каждого направления подъезда к каждому перекрестку. Данная модификация не приведет к потере возможности нахождения оптимального маршрута, ведь теперь каждому возможному положению транспортного средства (с учетом текущего направления движения) на дорожной сети в графе соответствует единственная вершина. До любой вершины, а, соответственно, и любого положения будет найден кратчайший путь. Следовательно, такой подход позволяет найти оптимальный маршрут с учетом направления движения на перекрестках.

Очевидно, что для перекрестка существует столько различных направлений движения до него, сколько участков дороги инцидентно ему. Каждому участку дороги инцидентно ровно два перекрестка, а, следовательно, в новом графе будет всего $2 \cdot M$ вершин, где M – количество участков дороги между двумя перекрестками в исходной дорожной сети. Количество ребер в полученном графе можно оценить, как $P_a \cdot M$, где P_a – среднее количество участков дорог, инцидентных перекрестку в исходной дорожной сети. Для реальных дорожных сетей значение P_a находится в интервале от 3 до 4, что говорит о том, что количество ребер в новом графе будет больше, чем в исходном не более чем в 4 раза.

Более того, для реализации алгоритма «A star» на таком графе не обязательно заранее задавать структуру графа в явном виде. Достаточно рассматривать только те пары {идентификатор перекрестка; направление движения до этого перекрестка}, которые будут посещены в ходе работы данного алгоритма. Таким образом, количество вершин в полученном графе для реальной дорожной сети будет существенно меньше, чем $2 \cdot M$. Поэтому количество ребер, которые будут рассмотрены в ходе работы алгоритма, так же будет намного меньше приведенной ранее оценки.

Очевидно, что для реализации данного алгоритма необходим способ определения типа маневра (движение прямо, поворот налево или поворот направо) для каждого возможного направления движения. Для этого достаточно учесть координаты трех перекрестков: текущего перекрестка C_1 , предыдущего перекрестка C_2 , и перекрестка C_3 , в направлении которого выполняется дальнейшее движение. Рассмотрим два вектора (C_2, C_1) и (C_1, C_3) , анализируя их взаимное расположение можно определить тип маневра. В случае, когда перекресток имеет 4 инцидентных ему участка дороги, среди всех возможных направлений дальнейшего движения можно выбрать то, в котором векторное произведение данных векторов является наименьшим, такое движение можно считать поворотом налево. Аналогично, если выбрать направление, для которого векторное произведение имеет наибольшее значение, то такой маневр является поворотом направо. Оставшееся направление соответствует движению прямо. В случае если перекрестку инцидентно 3 участка дороги, требуется определить направление, для которого угол отклонения первого вектора от второго наименьший. Такое направление соответствует движению прямо. Оставшийся тип маневра определяется в зависимости от векторного произведения вектора для движения прямо и вектора этого направления. Можно предположить, что если оба угла отклонения больше некоторого порогового

значения, то ни один из маневров не является движением прямо.

Оценка вычислительной сложности алгоритма

Вычислительная сложность алгоритма «A star» совпадает с вычислительной сложностью алгоритма Дейкстры, оценка которой $O(E \cdot \log E)$, где E – количество ребер в графе. Такая асимптотическая оценка достигается при использовании в реализации бинарной кучи (очереди с приоритетами). Как уже было замечено, количество ребер в новом графе равняется $P_a \cdot M$, что в P_a раз больше чем в исходном графе. Учитывая, что количество ребер умножается на некоторую константу P_a , итоговая асимптотическая оценка не изменяется. Следовательно, для дорожной сети предложенная модификация алгоритма «A star» так же имеет вычислительную сложность $O(M \cdot \log M)$, как и аналогичный алгоритм без модификации.

Заключение

В результате проделанной работы была предложена модификация алгоритма «A star» для дорожных сетей, позволяющая учитывать направление движения на перекрестке. Было установлено, что данная модификация обладает той же вычислительной сложностью, что и оригинальный алгоритм.

Литература

1. Кристофидес Никос. Теория графов: алгоритмический подход – 2-е изд., испр. – М.: Мир, 1978. – 430 с.
2. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ / Т.Кормен – М.: Издательский дом «Вильямс», 2011. – 1293 с.
3. Delling, D. and Sanders, P. and Schultes, D. and Wagner, D. "Engineering route planning algorithms". Algorithmics of large and complex networks – «Springer», 2009. – 376 p.

АЛГОРИТМ СЖАТИЯ ГИПЕРСПЕКТРАЛЬНЫХ АЭРОКОСМИЧЕСКИХ ИЗОБРАЖЕНИЙ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ

Саринова А.Ж., Замятин А.В.*

Инновационный Евразийский университет
140003, Республика Казахстан, г. Павлодар, ул. Ломова, 45

*Томского политехнического университета
634050, Россия, г. Томск, пр-т Ленина, 30

E-mail: assiya_prog@mail.ru

Введение

Современные центры космического мониторинга и системы дистанционного зондирования Земли (ДЗЗ) позволяют оперативно принимать, обрабатывать, архивировать и распространять объемы данных, составляющие порой десятки и сотни гигабайт [1, 2]. При этом одной из ключевых является задача сжатия данных ДЗЗ с целью

повышения эффективности передачи данных по каналам связи ограниченной пропускной способности и их архивирования в подсистемах хранения данных ДЗЗ ограниченной емкости. Наибольшую ценность представляет сжатие без потерь, не допускающее искажений статистических яркостных характеристик восстановленных данных.