Петрозаводский государственный университет
Математический факультет
Кафедра информатики и математического обеспечения

Промежуточный отчет

Идентификация маршрута мобильного пользователя по набору GPS координат.

Выполнил:
студент II курса
Сало А. Ю.

Руководитель:
кандидат технических
наук, доцент кафедры
Богоявленская О. Ю.

Петрозаводск
2002

Идентификация маршрута мобильного пользователя происходит с использованием координат, предоставляемых GPS (Global Positioning System), или просто GPS координат. Во время движения пользователя через некоторые промежутки времени происходит вычисление его текущих GPS координат. Таким образом, мы получаем последовательность точек в пространстве, которые в какой-то степени отображают перемещение пользователя. Задача заключается в том, чтобы с заданной точностью представить маршрут пользователя в виде ломаной. Эта задача подразделяется на две части. Во-первых, необходимо получить набор точек с использованием GPS. При этом точки должны находиться в удобной для дальнейших вычислений системе координат. Во-вторых, необходимо вычислить и оставить точки, являющиеся точками поворота в маршруте пользователя, который надо с некоторой точностью представить в виде ломаной, а остальные точки отбросить.

GPS предоставляет координаты положения пользователя в двух стандартных системах координат: в декартовой трехмерной системе координат, зафиксированной относительно Земли (Cartesian Earth-Centered, Earth-Fixed (ECEF) X, Y, Z), и географические координаты пользователя — широта, долгота и высота над поверхностью Земли (Geodetic Latitude, Longtitude and Height). Существуют формулы для перевода ECEF XYZ координат в географические и обратно. Однако для нас было бы удобнее использовать координаты точки в двумерной декартовой системе координат (с учетом того, что идентифицировать маршрут пользователя предполагается на ограниченной территории, т. е. поверхность Земли можно рассматривать как плоскость, и, кроме того, предполагается, что перепадами высот в данной местности можно пренебречь). В принципе, можно использовать непосредственно ECEF XYZ, т. е. трехмерные декартовы координаты. При этом усложнится процедура определения расстояния между точками, станет труднее определять точки поворота. С другой стороны, не потребуется дополнительного времени на перевод из одной системы координат в другую, а главное, можно будет избежать грубых ошибок (blunders), связанных с такими переводами. Возможно также использование географических координат без учета высоты над поверхностью Земли.

Важным является то, что GPS приемники (receivers) могут предоставлять координаты положения мобильного пользователя в иной, требуемой самим пользователем системе координат (user-required). Остается определить правила построения системы координат для данного мобильного пользователя на данной локальной территории.

Вторая часть проблемы идентификации маршрута мобильного пользователя по его GPS координатам тесно связана с проблемой приближения или упрощения линии (line simplification problem) в картографии. Лучшим в плане понимания сути проблемы считается алгоритм Дугласа-

Пекера для приближения линии (Douglas-Peucker line-simplification algorithm).

Базовый алгоритм Дугласа-Пекера довольно прост и легко выполним с использованием рекурсии. Идея его заключается в следующем: пусть необходимо приблизить сегмент $\overline{V_i V_j}$ с точностью ϵ. Если расстояние самой удаленной от этого сегмента вершины не превосходит ϵ, тогда сегмент упрощен. В противном случае, берем эту вершину в качестве разделяющей и рекурсивно приближаем два получившихся сегмента.

Дан массив вершин V, вызов DPbasic(V, i, j) приближает сегмент от V_i до V_j.

Procedure DPbasic(V, i, j)

1. Найти вершину V_f, максимально удаленную от $\overline{V_i V_j}$. Присвоить *dist* значение расстояния до этой точки.
2. if $dist > \epsilon$ then
   - 3a. DPbasic(V, i, f)
   - 3b. DPbasic(V, f, j)

   else
   - 4. Output$(\overline{V_i V_j})$

Базовый алгоритм приближения линии.

Оценка сложности этого алгоритма дает $O(n^2)$ для количества точек n. Это, конечно, неудовлетворительно, т. к. при большом количестве точек время работы существенно возрастает. Есть несколько другой алгоритм, основное различие которого с алгоритмом Дугласа-Пекера состоит в ином способе нахождения точки разделения, т. е. точки поворота. Он основывается не на непосредственном последовательном вычислении расстояний до каждой точки с последующим определением наиболее удаленной, а на утверждении, что максимально удаленная точка находится на выпуклом многоугольнике (convex hull), который содержит все точки ломаной. Сначала определяются точки, которые являются вершинами этого многоугольника, затем из них выбирается та, которая является максимально удаленной. Навание алгоритма (path hulls based algorithm) можно перевести как алгоритм основанный на построении сопутствующих многоугольников. Используется специальная структура данных (path hull data structure). Время работы этого алгоритма в худшем случае пропорционально $O(n \log_2 n)$, что равняется времени работы базового алгоритма Дугласа-Пекера в лучшем случае. Такое время работы является удовлетворительным.

Однако, на мой взгляд, алгоритм Дугласа-Пекера, равно как и усовершенствованный алгоритм, обладает некоторыми недостатками, которые делают его непосредственное применение непригодным для идентификации маршрута мобильного пользователя.

Во-первых, самым главным недостатком алгоритма яляется то, что он применим толькл для ломаных, не имеющих самопересечений. Авторы случаи с самопересечениями просто не рассматривают, только оговаривают, что верный результат гарантирован лишь при их отсутствии. Очевидно, что на практике такие случаи не только могут, но и будут появляться, поэтому алгоритм будет приводить к неверным результатам.

Во-вторых, приходится при добавлении каждой новой точки пересчитывать заново все точки поворота. Несмотря на то, что сложность усовершенствованного алгоритма $O(n \log_2 n)$, затраты по времени могут оказаться значительными. Кроме того, приходится по многу раз пересчитывать уже найденные точки поворота, т. е. выполнять по нескольку раз уже выполненную до этого работу. Для того, чтобы избежать такого нерационального использования памяти и времени, возможен следующий вариант: применяем алгоритм каждый раз с получением новой координаты до тех пор, пока не найдем вую точку поворота; запоминаем эту точку; снова применяем алгоритм, но теперь не рассматриваем те точки, которые отсекаются запомненной точкой; находим следующую точку поворота, запоминаем ее и т. д. Однако, в некоторых случаях это решение не будет оптимальным. Возможно также комбинирование приведенного выше метода с усовершенствованным алгоритмом Дугласа-Пекера.