

基于 GIS 路网的公交线路轨迹算法^①

钟会玲, 金红达, 沈建惠, 沈 斌, 徐 梦

(浙江浙大中控信息技术有限公司, 杭州 310052)

摘 要: 为解决公交线路轨迹偏移路网以及在 GIS 路网信息缺失尤其是乡村道路情况下的公交轨迹描绘. 论文首先通过深入分析公交车辆 GPS 数据, 分别聚类出线路上下行轨迹点; 其次, 轨迹点清洗并排序; 再次, 结合 GIS 路网基础信息进行地图匹配; 最后, 根据改进的 Dijkstra 算法解决路网拓扑结构缺失情况下制作出公交线路轨迹. 将该算法实施在 A 市 35 条公交线路, 线路匹配成功率为 85%, 未匹配成功线路由于样本缺失或者路网基础信息错误导致, 可见该算法具有较好的准确率和实用性.

关键词: Dijkstra 算法; 公交; GIS; 地图匹配; 轨迹偏移

引用格式: 钟会玲, 金红达, 沈建惠, 沈斌, 徐梦. 基于 GIS 路网的公交线路轨迹算法. 计算机系统应用, 2017, 26(11): 182-186. <http://www.c-s-a.org.cn/1003-3254/6054.html>

Algorithm of Bus Route Trajectory Based on GIS Road Network

ZHONG Hui-Ling, JIN Hong-Da, SHEN Jian-Hui, SHEN Bin, XU Meng

(Zhejiang Supcon Information Technology Co. Ltd., Hangzhou 310052, China)

Abstract: In order to solve the bus trail problem that bus route trajectory offset road network and GIS road network information is missing, especially on the rural roads, this paper proposes an algorithm of bus route trajectory based on GIS road networks. Firstly, it makes an in-depth analysis of bus GPS data, clustering line up and down track points respectively. Secondly, it cleans the track points and sort. Thirdly, it combines with GIS road network information for map matching. Finally, according to the improved Dijkstra algorithm, it solves the bus trail problem that GIS road network information is missing. The algorithm is applied in City A with 35 bus lines. The successful match rate is 85%. Unsuccessful matches are due to missing samples or wrong road network information. It can be seen that the algorithm has good accuracy and practicability.

Key words: Dijkstra algorithm; bus; GIS; map matching; trajectory offset

如今, 大部分城市的公交车辆安装了 GPS 定位终端设备, 从而能够实现实时对公交车辆进行位置监控. 有学者利用 GPS 定位数据聚类研究公交线路轨迹, 也有将轨迹点根据已有 GIS 路网信息匹配路网, 董俊等人利用改进 Dijkstra 算法在 GIS 导航应用中最短路径搜索研究, 该改进的算法虽然效率较传统算法有所提高, 但能在 GIS 导航中应用前提是路网基础信息完整, 而且针对 GIS 路网信息缺失尤其是乡村道路的情

况未给出具体解决方案, 得出公交车辆的行驶轨迹, 能提取出公交线路, 从而能够对公交线路尤其是新开公交线路进行快速数字化, 公交线路轨迹有助于公交线路车辆上下行、进出站时间和班次计算, 公交轨迹数据是路网的拓扑结及完善路网信息的终于参考依据, 同时也能直观反映公交路网结构供决策者决策使用.

因此本文利用大量的历史 GPS 轨迹数据, 对 GPS 轨迹数据聚类, 并结合路网基础信息进行地图匹配, 提

^① 基金项目: 浙江省科技计划项目 (2017C01016)

收稿时间: 2017-02-23; 修改时间: 2017-03-09; 采用时间: 2017-03-20

高公交线路精度.

1 方法与算法论述

1.1 方法

1.1.1 K-means 算法

K-means 算法是 MacQueen(1967) 在 Cox(1957)、Fisher(1958)、Sebestyen(1962) 等人的研究基础上提出的, k-means 算法是一种较为简单也最为常用的迭代聚类算法, 迭代过程中不断更新簇的聚类中心, 每个簇的聚类中心用该簇中对象的平均值表示. 经过 k-means 算法得到的簇, 相同簇中对象的相似度高, 不同簇中对象的相似度较低. GPS 轨迹聚类的处理过程如下:

① 选取初始站点经纬度作为初始聚类中心 $\mu_1, \mu_2, \dots, \mu_k$, 聚类样本为 $x^1, x^2, \dots, x^m$. 其中, $\mu_k = (\text{longitude}, \text{latitude})$, $x^m = (\text{longitude}, \text{latitude})$.

② 每次迭代将 x^i 分配到最近的聚类中心 μ_j , 同时将 μ_j 所在的聚类簇选择新的聚类中心点 μ_j .

③ 迭代达到以下收敛条件是终止.

对每个经纬度样本 i ,

$$c^i = \arg \min_j \|x^i - \mu_j\|^2 \quad (1)$$

$$\mu_j = \frac{\sum_{i=1}^m I\{c^i = j\} x^{(i)}}{\sum_{i=1}^m I\{c^i = j\}} \quad (2)$$

其中, μ_j 为每次迭代聚类中心位置点.

如图 1 为三次迭代中每次聚类中心点位置, 图 1(a) 中聚类中心点离样本点平均距离未达到收敛, 因此继续迭代, 图 1(b) 中为第二次迭代聚类中心点位置, 图 1(c) 为三次迭代后, 各聚类簇平均距离达到预设阈值, 且所有数据单元都收敛迭代终止.

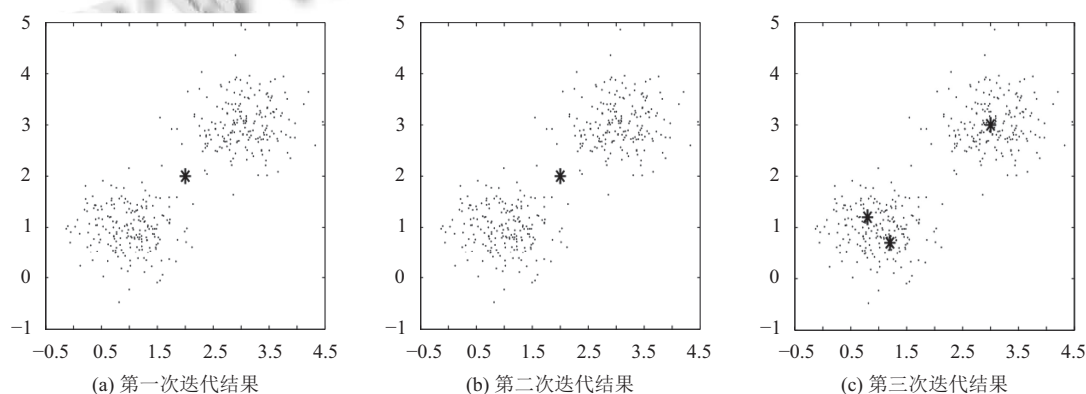


图 1 三次迭代聚类中心点位置

1.1.2 改进的 Dijkstra 算法

Dijkstra 算法是典型的单源最短路径算法, 用于计算一个节点到其他所有节点的最短路径. 主要特点是以起始点为中心向外层层扩展, 直到扩展到终点为止.

Dijkstra 算法是典型的单源最短路径算法, 用于计算一个节点到其他所有节点的最短路径. 主要特点是以起始点为中心向外层层扩展, 直到扩展到终点为止, 按路径长度递增次序产生最短路径算法.

Dijkstra 算法不能计算中间线段截断的情况, 将轨迹点匹配到路网时, 通常情况下如果路段 a 与路段 b 相邻, 路段 a 的终点应为路段 b 的起点, 进而形成路网拓扑结构, 当由于路网基础信息缺失导致路段 a 与路段 b 未连接, 且中间无其他任何路段信息, 此时仅仅利

用 Dijkstra 算法不能解决问题.

改进的算法伪代码如下.

Function()=updijkstra()

for(; ;)

Vertex s=smallest unknown distance vertex;

if(s==not_a_vertex)

% 如果不连通 dijkstra 会跳出算法

% 针对不连通调用 updijkstra 函数

w=func_updijkstra(s, v);

s.flag=1;

for each Vertex w adjacent to s

if (!w.known)

if(s.length+cvw<w.length)%cvw 为起点到当前点

最短路径长度

```
% 更新 w
    decrease(w, length to s.length+cvw);
    w.path=s;
end
end
end
end
End
updijkstra 算法中 func_updijkstra
w=func_updijkstra(s, v);
v.falg=false;% 初始化全未标记
if i.falg==false% 未标记
    for i in v
        for j in v
            % 计算路段 i 的 e_v 到 j 所有路段 s_v 距
            离及夹角
            d(j)=distance(i.e_v, j.s_v);
            an(j)=angle(i.e_v, j.s_v);
            ppd=w1*d(j)+w2*an(j);%w1 和 w2 为距离
            和夹角权重
            i.falg=true;
        end
        % 选择可信度最大的路段为连通边
        j= max(ppd);
        j.flag=true;
        vertex s=add[s,j];
        vertex v=delete[v,j];
    end
end
```

其中 Vertex v 为线路轨迹从起点到终点匹配成功路段集 V.

Vertex v=[s_longitude, s_latitude, e_longitude, e_latitude, roadid, length, flag]

s_v=[s_longitude, s_latitude];e_v=[e_longitude, e_latitude];

因此本文改进 Dijkstra 算法, 寻找路网拓扑节点与寻找邻近路段同时进行, 从起点找到终点完成路网匹配.

1.2 算法步骤

聚类后样本点结合线路上下行起点和终点, 得出

带顺序的轨迹点, 该轨迹点可初步作为线路轨迹, 轨迹点匹配路网, 将距离和角度赋值不同的权重, 选择可信度最大的作为被匹配路段, 最后根据改进的 Dijkstra 算法, 起点路段选择邻近路段, 选择依据是寻找有拓扑结构的所有相邻路段计算距离, 如果无相邻路段, 根据匹配路段与最近路段夹角和被匹配路段轨迹数二者结合起来, 作为下一个路段选择依据. 包括 5 个步骤.

步骤一. 利用 k-means, 将同一个站点名称对应两个以上不同经纬度信息数据聚类, 设置聚类半径阈值 a, 超出阈值 a 范围部分单独聚类形成站点, 目的是线路基础站点信息统一化.

步骤二. 选择各线路分方向一周 GPS 样本数据, 利用 k-means, 设置聚类半径阈值 b 和迭代次数阈值 c, 达到阈值 b 或者迭代次数达到迭代次数阈值 c 为迭代终止条件, 得到聚类后数据集 A

步骤三. 根据起点找到数据集 A 中最近的点, 根据夹角和距离选择邻近轨迹点, 直到找到终点附近一定距离范围结束. 得到带序号的轨迹点集 B

步骤四. 轨迹点匹配路段, 如果投影到路段 (且非路段延长线) 距离超过距离设置范围, 轨迹点作为路段样本集保留, 得到候选路段集 C

步骤五. 用轨迹点起点从候选路段集 C 中匹配最近的路段作为线路起点路段, 起点路段根据改进 Dijkstra 算法寻找有拓扑结构的所有相邻路段计算距离, 如果无相邻路段, 根据匹配路段与最近路段夹角和被匹配路段轨迹数二者结合起来, 作为下一个路段选择依据, 直到被匹配到的路段在终点站附近一定范围终止, 最终得到带顺序的路段, 即为路线轨迹.

2 实验分析

2.1 数据

2.1.1 公交 GPS 数据

为了验证所提算法的有效性, 以 2017 年 1 月份浙江省 A 市 35 条公交线路 GPS 数据为实验对象, 正常情况下, 每条线路每天数据量约 3 万行, 选择一条线路一周 GPS 数据, 每辆车 GPS 数据, 与对应线路站点起点和终点比对分离出线路上下行然后分方向清洗异常点, 得到带方向的样本数据.

2.1.2 线路基础站点

线路基础站点经纬度信息是随公交车采集, 会存在多条线路同过一个站点但出现不同经纬度情况, 因

此需要采用聚类方法,将多站点信息同一化,聚类后得到所有站点基础信息为一个站点对应一个经纬度,上下行为两个经纬度。

2.1.3 A市地图数据

根据高德地图底图信息,地图信息存在节点信息未显示,利用 arcgis 处理后将道路以及路段中所有节点经纬度信息全部导入到数据库。

2.2 实验结果

基于 GIS 路网的公交线路轨迹算法,在各阶段实验结果如图 2。以线路 K1 路上行方向为例,公交 gps 样本点聚类,初始聚类中心 $\mu_1, \mu_2, \dots, \mu_k$ 为上行方向 31 个站点经纬度,聚类样本 $x^1, x^2, \dots, x^m$ 为线路 K1 所有上行 gps 点;每次迭代将样本 gps 点 x^i 分配到最近的聚类中心 μ_j ,同时将 μ_j 所在的聚类簇重新选择新的聚类中心点 μ_j ,当聚类簇 c 聚类中心为 μ_j 时,重新计算聚类半径 c^i ,如果 c^i 大于半径阈值(此处设置为 100 米)则增加一个聚类中心 μ_{j+1} ,簇 c 内部迭代寻找新的 μ_j ,以此类推,直到所有 $c^i \leq 100$,为达到算法效率设置各簇内迭代 ≤ 10 次。最终聚类中心 μ_j 个数为 166,结果如图 2(a),图中 166 个聚类中心 μ_j 基本展现聚类后线路轨迹,此时聚类中心点为无序轨迹样本点,图 2(a)中能清晰看出 μ_j 中出现个别异常离散点;根据 μ_j 点与左右邻近点间形成的向量夹角 angle ,如果 $\text{angle} > 110^\circ$,去除 μ_j 点,并根据邻近距离原则将轨迹点排序,形成带顺序轨迹点结果如图 2(b),图 2(b)为 arcgis 地图软件打开的路网信息,然后在该基础上导入带顺序轨迹点,且相邻轨迹点用直线连接后的效果图,路网信息坐标与轨迹点坐标均为 WGS84 坐标,图 2(c)为图 2(b)放大后拐角效果图,虽然路网信息坐标与轨迹点坐标均为 WGS84 坐标,但是叠加显示时对比有明显误差,原因由于存在 GPS 数据本身会出现 30-50 米精度误差,而且路网信息为道路中心线与公交车实际行走轨迹必然存在不重合情况,由图 2(c)可见在拐角处明显出现偏离道路。因此需要与 GIS 底图数据匹配, μ_j 与底图匹配根式 (3):

$$\text{ppd} = w_{\text{len}} * \text{len_fi} + w_{\text{dir}} * \text{dir_fi} \quad (3)$$

其中 ppd 为 μ_j 与底图路段匹配度, len_fi 为 μ_j 到路段标准化后垂直距离, w_{len} 为距离权重, dir_fi 为 μ_j (带方向角) 与路段标准化夹角, w_{dir} 为夹角权重。

根据 ppd 选择 k1 路上行方向所有轨迹点 μ_j 匹配

的候选路段集 Vertex v , 结果如图 2(d) 所示,直线路段已被拆分为间隔不大于一定距离范围的短路段,路网轮廓清晰,尤其拐角处,通过轨迹点匹配到的候选路段包括周边一定距离范围平行及交叉路段,且也出现路段断开情况。

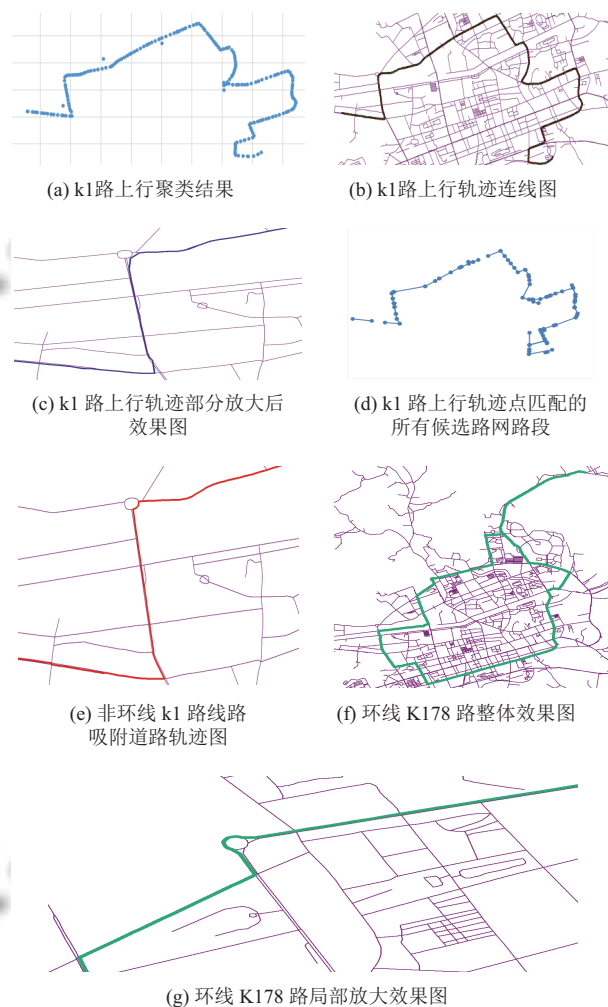


图2 基于 GIS 路网的公交线路轨迹算法各阶段实验结果

对路段集 Vertex v 利用上文改进的 Dijkstra 算法,从起点站点所在路段 s 寻找邻近路段 w ,如果路段 w 与 s 连通,则直接用 Dijkstra 算法计算 w 的 s 到当前点最短路径长度 cvw ,如果未连通,计算路段 s 的 e_v 到所有路段 s_v 距离及夹角 $d(j)$, $\text{an}(j)$ 以及各路段集 v 可信度 ppd ,选择可信度最大的路段为连通边,然后对连通边使用 Dijkstra 算法,经过基于 GIS 路网的公交线路轨迹算法后, k1 线路轨迹结果如图 2(e),图 2(e)为线路吸附道路轨迹,可见路网信息坐标与轨迹

点坐标重合,未出现轨迹点偏移路网道路,图 2(f) 为环线吸附道路轨迹效果图,环线与非环线均能很好适应该算法,实现轨迹点与路网信息完全匹配。

3 讨论

首先聚类后样本点结合线路上下行起点和终点,得出带顺序的轨迹点,该轨迹点可初步作为线路轨迹,当然在拐角出现漂移轨迹情况,原因是聚类粒度导致,但是聚类过程中如果聚类粒度太小会导致轨迹离散点过多,清洗困难,且需要消耗机器大量内存,浪费资源。

其次在处理环线过程中将复杂问题简单化,将环线根据站点信息从中间站点处截断,上半部分为上行,下半部分为下行,能有效处理重复轨迹相反方向以及绕圈情况。

再次根据改进 Dijkstra 算法寻找有拓扑结构的所有相邻路段计算距离,如果无相邻路段,根据匹配路段与最近路段夹角和被匹配路段轨迹数二者结合起来,作为下一个路段选择依据,该方法能解决路网信息丢失以及路段断开情况。

4 结语

本文介绍了 GPS 轨迹聚类算法,在轨迹点、路段夹角和被匹配路段轨迹数信息数据基础上结合改进的 Dijkstra 算法,最终得到与路网匹配的线路轨迹数据。

在 A 市交通项目中取得较好效果,该算法结果是公交实时到站算法的基础,为后续公交相关算法提供完整的线路信息。

本文研究的线路适合在少量路网信息缺失情况下,可利用轨迹点完善路网信息,但如果出现大量基础路网信息缺失,算法准确率将降低,后续研究将利用所有线路轨迹点完善路网基础信息。

参考文献

- 1 李清泉,黄练.基于 GPS 轨迹数据的地图匹配算法.测绘学报,2010,39(2): 207-212.
- 2 王祖超,袁晓如.轨迹数据可视分析研究.计算机辅助设计与图形学学报,2015,(1): 9-25.
- 3 李桃迎.交通领域中的聚类分析方法研究[博士学位论文].大连:大连海事大学,2010.
- 4 董俊,黄传河.改进 Dijkstra 算法在 GIS 导航应用中最短路径搜索研究.计算机科学,2012,39(10): 245-247, 257. [doi: 10.3969/j.issn.1002-137X.2012.10.055]
- 5 姜波.一种基于 GPS 轨迹的路线规划方法.软件工程师,2012,(3): 43-46.
- 6 周建武,杨彬.一种公交班次行驶轨迹精确计算算法的研究及应用.计算机应用与软件,2014,31(8): 100-103, 117.
- 7 王佳,符卓,杜靖毅.基于遗传算法的城市公交骨架线网优化设计.计算机应用研究,2012,29(12): 4518-4521. [doi: 10.3969/j.issn.1001-3695.2012.12.030]
- 8 张扶桑,金蓓弘,汪兆洋,等.基于轨迹挖掘的公交车自组织网络路由机制.计算机学报,2015,38(3): 648-662.