

车辆行驶轨迹丢失检测以及补偿应用研究

罗健飞¹, 邢亚东¹, 张卫庆¹, 夏代江¹, 吴仲城²

¹安徽中科美络信息技术有限公司智能终端研发中心, 安徽 合肥

²中国科学院强磁场科学中心, 安徽 合肥

Email: jfluo@mail.ustc.edu.cn

收稿日期: 2021年4月26日; 录用日期: 2021年5月21日; 发布日期: 2021年5月28日

摘 要

针对轨迹应用类系统中, 因为多种原因导致车辆行驶轨迹丢失的问题, 本文提出一种车辆行驶轨迹丢失检测以及补偿应用方法, 首先对行驶轨迹进行丢失检测, 然后采用地图资源方的接口以及根据该车辆历史轨迹的相似性匹配相结合的方法对丢失轨迹进行补偿, 实验部分结合具体的车辆数据, 验证了本文方法的有效性, 能有效地解决车辆轨迹丢失问题, 具有很强的实用价值, 目前本文提出的方法已经应用在实际场景中, 为60,000多台北斗定位终端提供轨迹检测和补偿服务。

关键词

北斗车载定位终端, 轨迹丢失, 自动化检测, 路径补偿

Application Research on Vehicle Trajectory Loss Detection and Compensation

Jianfei Luo¹, Yadong Xing¹, Weiqing Zhang¹, Daijiaing Xia¹, Zhongcheng Wu²

¹iTerm R&D Centre, AnHui ZhongKeMeiLuo Info-Tech Co. Ltd., Hefei Anhui

²High Magnetic Field Laboratory, Chinese Academy of Sciences, Hefei Anhui

Email: jfluo@mail.ustc.edu.cn

Received: Apr. 26th, 2021; accepted: May 21st, 2021; published: May 28th, 2021

Abstract

In the trajectory application system, the vehicle travel trajectory is lost due to various reasons. This paper presents a method of vehicle trajectory loss detection and compensation application. Firstly, the trajectory loss is detected, and then the lost trajectory is compensated with the map interface and the similarity matching method based on the historical trajectory of the vehicle. The

文章引用: 罗健飞, 邢亚东, 张卫庆, 夏代江, 吴仲城. 车辆行驶轨迹丢失检测以及补偿应用研究[J]. 计算机科学与应用, 2021, 11(5): 1563-1571. DOI: 10.12677/csa.2021.115161

experimental part combines the specific vehicle data to verify the effectiveness of the method and effectively solve the problem of vehicle trajectory loss. It has a strong practical value. The method proposed in this paper has been applied in the actual scene to provide trajectory detection and compensation services for more than 60,000 BeiDou-based positioning terminals.

Keywords

Beidou Vehicle Positioning Terminal, Trajectory Loss, Automatic Detection, Trajectory Compensation

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

车辆行驶轨迹作为车联网应用的一个基础服务,为车联网多种业务应用提供支撑。例如,被广泛使用的打车软件,基于车辆行驶轨迹的里程进行计费,物流公司的油费管理,公务用车的费用计算与透明化监督等。因此,如何实现车辆行驶轨迹的完整性有着很高的应用价值。传统的车辆行驶轨迹计算方法主要有曲线拟合法、拓扑法和图元计算法等,但它们都具有算法复杂度较高的缺点,并不适用于当前车联网技术与云计算、大数据融合应用场景下。相比传统技术,基于图像处理的视觉技术具有使用场景更多、计算更精准等特点,根据系统所配置相机数其可以分为单目视觉和立体视觉路线,但视觉里程计的装备成本较高,暂时并不适合于大规模部署推广。因此,回归到车辆行驶轨迹的本质,需要解决卫星定位存在的问题。

北斗车载定位终端发送定位是否正常受到多个方面因素影响:北斗卫星信号(高楼、隧道、恶劣天气等)、网络通讯信号以及设备状态(设备老化、天线接触不良等)等等。这些因素的发生从而导致定位轨迹异常,最终体现在产品服务中便是车辆的轨迹不完整(部分路段的轨迹表现为一条直线),以及里程低于实际里程[1]。这一点在客户端(设备)层面难以避免或者说优化、更替成本较高。所以,急需从服务端层面将该问题最大化的解决。相对于从客户端层优化,从服务端进行优化能够更大化地覆盖更多类型的设备,也就是使用一套解决方法来解决所有类型终端的此类问题[2] [3] [4]。

2. 系统设计

要实现轨迹自动化补偿,首先需要能够精确地识别车辆的哪些轨迹存在丢失可能,其次就是在发现丢失的情况后如何更好地补偿(最大化的接近实际轨迹)。所以本文从两个方向探讨[5] [6] [7]:轨迹丢失自动检测方法以及轨迹丢失自动补偿方法。

系统框图如图 1 所示,各模块功能介绍如下:

1) 终端发送的定位感知数据通过 adapt 对数据增强后(终端、车牌对应关系)适配程序,采用 MySQL 触发器实时感知订单表(apply_car)变化。

2) 后台的存储程序实时读取 Kafka 消息,并写入 NoSql 数据库 HBase;轨迹数据解析与数据转发、存储各个模块相互分离,后续涉及轨迹查询、里程计算、分析统计等功能,直接通过缓存或 HBase 服务器中获取数据。

3) 通过消息中间件,一方面接收轨迹信息,一方面监听订单信息,采用 storm 集群实时运算车辆轨

迹状态。

4) storm 集群分析计算的结果, 采用数据共享平台实现分析结果从 kafka 消息中间件实时写入 mysql 数据库, 以供前台查询使用。

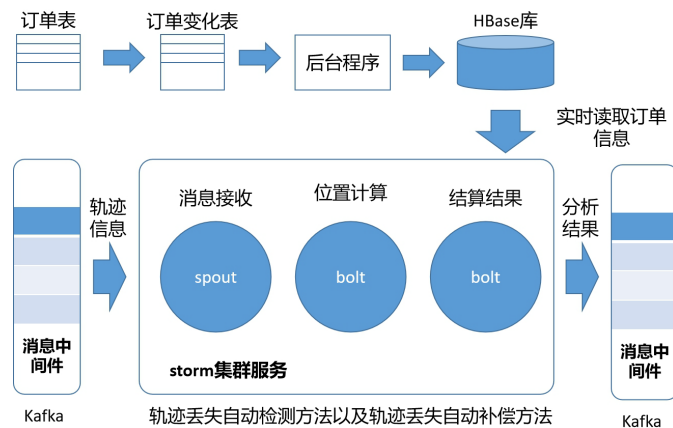


Figure 1. System architecture diagram

图 1. 系统架构图

2.1. 轨迹丢失自动检测方法

这里选取一辆车一天的行驶轨迹, 对一天内的所有轨迹进行轨迹丢失的检测。

为了方便实验的开展以及效果验证, 我们编写两类工具: 轨迹获取工具和轨迹展示工具。其中, 轨迹展示工具分为三个版本: 原始轨迹展示、原始轨迹并显示标出轨迹丢失段展示和自动补偿后轨迹展示。

轨迹获取工具从 Hbase 中获取车辆指定时间段内的所有轨迹并保存到文本文件中, 如图 2。这里选取了 20 条轨迹有丢失的轨迹样本。每条样本包括一行元数据信息: 定位时间、经度、纬度以及速度。

9Z015_20181101000000_20181102000000.txt	文本文档	2018/11/12 16:10	125 KB
99E91_20181111000000_20181112000000.txt	文本文档	2018/11/13 9:24	30 KB
10385_20181106000000_20181107000000.txt	文本文档	2018/11/13 17:05	31 KB
38812_20181107000000_20181108000000.txt	文本文档	2018/11/13 17:08	32 KB
A0578_20181112000000_20181113000000.txt	文本文档	2018/11/12 16:11	184 KB
A3743-30010_20181105000000_20181106000000.txt	文本文档	2018/11/12 16:07	159 KB
W0776_20181111000000_20181112000000.txt	文本文档	2018/11/12 16:13	24 KB
W0971_20181112000000_20181113000000.txt	文本文档	2018/11/12 16:08	8 KB
W9095_20181105000000_20181106000000.txt	文本文档	2018/11/13 17:20	18 KB
W9096_20181112000000_20181113000000.txt	文本文档	2018/11/12 16:14	15 KB
W9110_20181107000000_20181108000000.txt	文本文档	2018/11/12 16:14	8 KB
W9396_20181105000000_20181106000000.txt	文本文档	2018/11/12 16:12	87 KB
W9500_20181107000000_20181108000000.txt	文本文档	2018/11/12 16:12	8 KB
W9502_20181105000000_20181106000000.txt	文本文档	2018/11/12 16:13	4 KB
40065_20181113000000_20181114000000.txt	文本文档	2018/11/14 8:55	160 KB
40089_20181109000000_20181110000000.txt	文本文档	2018/11/12 16:23	76 KB
98085_20181109000000_20181110000000.txt	文本文档	2018/11/12 16:29	169 KB
GW936-23615_20181109000000_20181110000000.txt	文本文档	2018/11/12 16:29	57 KB
B085_20181105000000_20181106000000.txt	文本文档	2018/11/12 16:14	10 KB
B085_20181108000000_20181109000000.txt	文本文档	2018/11/12 16:15	13 KB

Figure 2. Trajectory samples with missing trajectories

图 2. 有丢失轨迹现象的轨迹样本

具体的轨迹丢失检测流程如图 3 所示, 算法如下:

- 1) 首先计算定位集合中每个相邻点的时间间隔。
- 2) 计算这些相邻点中哪种时间间隔最常见(众数)。因为各种类型的终端发送频率不同, 并且相同终端在不同情况下发送频率也会上下波动, 所以计算出众数相比于平均数更能反应该终端的正常发送频率。
- 3) 判断众数*比率是否大于时间的设定阈值。其中比率为大于 1 的数字, 它的意义是可以允许相邻点之间可以丢失多少个点, 例如比率为 10 时, 众数*10 表示丢失了 10 个定位点的这个区间的长度。比率和时间的设定阈值可以作为参数根据情况进行调整, 这里时间的设定阈值我们设置为 10 分钟, 也就是如果大于 10 分钟我们认为有可能丢失轨迹, 距离阈值则一般根据不同路段的限速来测算两点的阈值。
- 4) 对于第 3 步中认为有可能丢失轨迹的相邻定位点, 计算这两点之间的距离, 并与距离的设定阈值进行比较。
- 5) 对于第 4 步中相邻距离低于距离设定阈值的相邻点, 可进行下一个相邻点的判断。如果不是, 则认为丢失了轨迹。

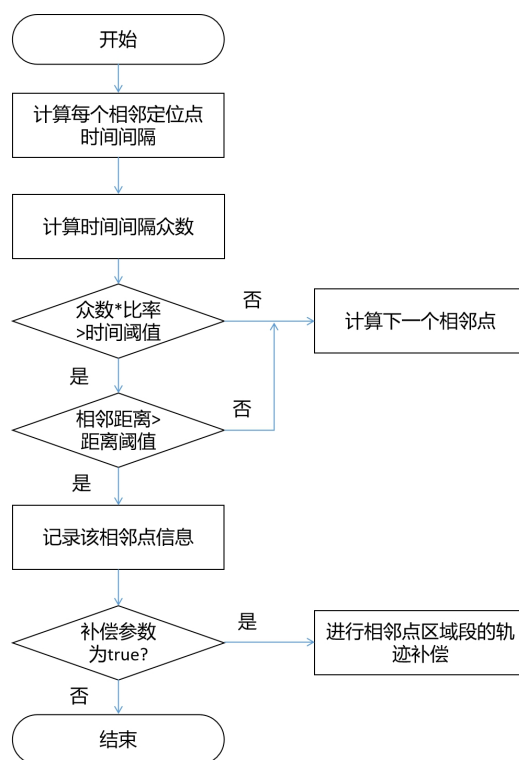


Figure 3. Track loss detection process

图 3. 轨迹丢失检测流程

2.2. 轨迹丢失自动补偿方法

轨迹丢失自动补偿方法目前考虑的有两种方法: 实时地图 GIS 数据资源接口服务以及基于车辆历史轨迹相似性匹配。

2.2.1. 实时地图 GIS 数据资源接口服务

根据已有的轨迹丢失的起点以及终点, 使用百度路线规划接口, 调用该起终点的轨迹规划, 获取这条规划轨迹的定位点集合, 然后补充到原集合中。需要稍微注意的是这些补充进行的集合, 需要跟原集合进行较好的匹配, 比如时间上的连续性以及速度的合理设置。

2.2.2. 基于车辆历史轨迹相似性匹配

本方法是从该车辆的历史轨迹中寻找该丢失段最有可能的行驶轨迹，然后进行补偿，主要采用机器学习的方法，通过样本训练，建立车辆行驶特征集，构建相似性匹配算法。实现本方法需要解决两个核心问题：1) 数据的合理存储以及相对应的查询；2) 相似性匹配算法。在此场景下(与轨迹查询的场景不同，轨迹查询的场景对存储的设计要求较低)，合理的存储是为了高效地查询、方便且准确地匹配。因为定位点是由经度、纬度，如果加上车辆、时间、速度就是一个五维的数据，对于维度高数据的存储以及检索，是非常耗费性能(以及存储容量)，尤其在数据量很大的情况下。所以需要降维存储、检索以及匹配，这也是本文采用 Storm 以及 Kafka 技术来构建大数据平台的缘由。

3. 数据实验

针对收集的 20 个样本进行轨迹丢失的自动检测和补偿效果展示。如表 1 所示，每个样本包括上文说的三个版本。

3.1. 样本数据集

Table 1. Sample set table

表 1. 样本集合表

序号	车配号	开始时间	结束时间
1	xx99E91	2018-11-11 00:00:00	2018-11-12 00:00:00
2	xx 10385	2018-11-06 00:00:00	2018-11-07 00:00:00
3	xx38812	2018-11-07 00:00:00	2018-11-08 00:00:00
4	xxA0578 黄	2018-11-12 00:00:00	2018-11-13 00:00:00
5	xxA3743	2018-11-05 00:00:00	2018-11-06 00:00:00
6	xxW0776	2018-11-11 00:00:00	2018-11-12 00:00:00
7	xxW0971	2018-11-12 00:00:00	2018-11-13 00:00:00
8	xxW9095	2018-11-05 00:00:00	2018-11-06 00:00:00
9	xxW9096	2018-11-12 00:00:00	2018-11-13 00:00:00
10	xxW9110	2018-11-07 00:00:00	2018-11-08 00:00:00
11	xxW9396	2018-11-05 00:00:00	2018-11-06 00:00:00
12	xxW9500	2018-11-07 00:00:00	2018-11-08 00:00:00
13	xxW9502	2018-11-05 00:00:00	2018-11-06 00:00:00
14	xx40065	2018-11-13 00:00:00	2018-11-14 00:00:00
15	xx40089	2018-11-09 00:00:00	2018-11-10 00:00:00
16	xx98085	2018-11-09 00:00:00	2018-11-10 00:00:00
17	xxGW936	2018-11-05 00:00:00	2018-11-06 00:00:00
18	xxB085 警	2018-11-08 00:00:00	2018-11-09 00:00:00
19	xxB085 警	2018-11-01 00:00:00	2018-11-02 00:00:00
20	xx9Z015	2018-11-01 00:00:00	2018-11-02 00:00:00

3.2. 算法效果

以下展示三个车辆的三个轨迹效果图：原始轨迹图、丢失轨迹自动检测标注图以及丢失轨迹补偿之后的轨迹图。其中，第二类图是自动检测出丢失轨迹段后在地图上使用红色定位点以及直线标注出来。

图 4 为原始及产出文件，.txt 为原始轨迹文件，.out 为自动检测生成文件，autoCompensation.out 为轨迹补偿后文件。

99E91_20181111000000_20181120000000.txt	2018/11/13 9:24
99E91_20181111000000_20181120000000.txt.autoCompensation.out	2018/11/15 11:11
99E91_20181111000000_20181120000000.txt.out	2018/11/13 15:56
10385_20181106000000_2018110700000000.txt	2018/11/13 17:05
10385_20181106000000_2018110700000000.txt.autoCompensation.out	2018/11/19 11:22
10385_20181106000000_2018110700000000.txt.out	2018/11/13 17:12
8812_20181107000000_2018110800000000.txt	2018/11/13 17:08
8812_20181107000000_2018110800000000.txt.autoCompensation.out	2018/11/15 11:44
8812_20181107000000_2018110800000000.txt.out	2018/11/13 17:08

Figure 4. Original and output documents

图 4. 原始及产出文件

这里选择车牌号 xx99E91 和 xx10385 进行说明。

xx99E91 的原始轨迹图如图 5(a)所示, 图中红色框内标识的为轨迹丢失的路段;

通过轨迹丢失自动检测方法, 对丢失轨迹进行自动标识, 主要标识丢失路段的起点、终点位置, 如图 5(b)所示。

针对丢失的轨迹, 利用本文介绍的轨迹丢失自动补偿方法, 根据丢失路段的起点、终点位置, 以及前后数据属性, 进行补偿, 补偿效果如图 5(c)所示。



Figure 5. (a) Original trajectory (xx99E91); (b) Automatic detection and labeling of missing tracks (xx99E91); (c) Trajectory after missing trajectory compensation (xx99E91)

图 5. (a) 原始轨迹图(xx99E91); (b) 丢失轨迹自动检测标注图(xx99E91); (c) 丢失轨迹补偿之后的轨迹图(xx99E91)

xx10385 的原始轨迹图如图 6(a)所示, 图中红色框内标识的为轨迹丢失的路段;

通过轨迹丢失自动检测方法, 对丢失轨迹进行自动标识, 主要标识丢失路段的起点、终点位置, 如图 6(b)所示。

针对丢失的轨迹, 利用本文介绍的轨迹丢失自动补偿方法, 根据丢失路段的起点、终点位置, 以及前后数据属性, 进行补偿, 补偿效果如图 6(c)所示。

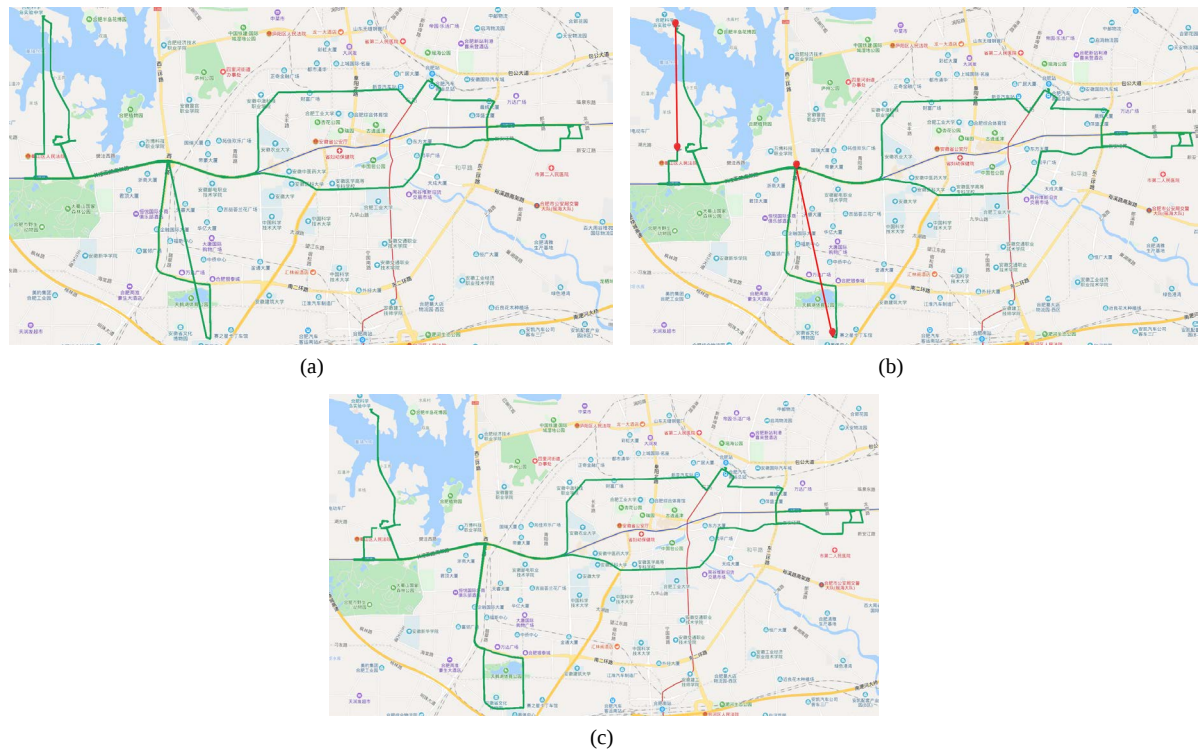


Figure 6. (a) Original trajectory (xx10385); (b) Automatic detection and labeling of missing tracks (xx10385); (c) Trajectory after missing trajectory compensation (xx10385)

图 6. (a) 原始轨迹图(xx10385); (b) 丢失轨迹自动检测标注图(xx10385); (c) 丢失轨迹补偿之后的轨迹图(xx10385)

通过对比车牌号 xx99E91 和 xx10385 的轨迹数据, 可以看到, 补偿后的轨迹显示合理, 也符合实际的行驶情况。

3.3. 执行效率

本文提出的方法最终需要应用在实际业务中, 所以算法的执行效率以及执行结果是衡量算法能够适用于大规模系统平台的基本条件, 本文以算法执行流程和执行时间作为衡量算法执行效率的指标。为此, 选择 4 个车牌号, 选择不同码表数值, 然后对比本文方法与差值法(本文采用的插值法为滑动平均窗口法, 既一个列表中的第 i 个位置数据为缺失数据, 则取前后窗口数据的平均值, 作为插补数据)的实验结果, 如表 2 所示。从表格数据可见: 对照实际结果, 本文方法比较插值方法, 更与实际结果接近。

3.3.1. 执行时间

从执行时间上, 本文进一步统计在不同里程下, 插值法和本文方法的时间差, 插值法会随着里程数增加, 插值的点也会随之增加, 本文方法结合历史轨迹以及地图数据补正, 会提高整体的运算效率, 这种效率会随着里程的增加而日益明显, 具体如图 7 所示。

Table 2. Comparison of experimental results
表 2. 算法执行流程和时间测试结果

车辆编号		NO.1	NO.2	NO.3	NO.4
采集间隔(s)		5	5	5	5
码表里程(KM)		189	256	362	493
执行里程(KM)	插值法	185.3	251.9	260.6	489.5
	本方法	187.8	255.1	361.5	492.1
执行时间(ms)	插值法	356	501	716	963
	本方法	334	471	679	911

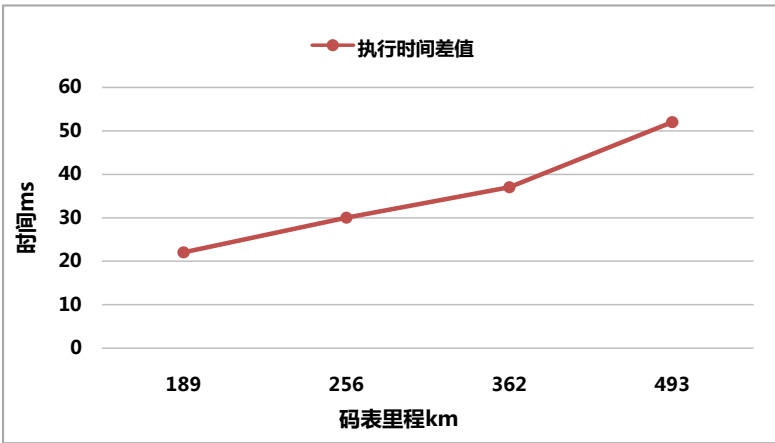


Figure 7. Method execution time difference under different mileage
图 7. 不同里程下方法执行时间差值

3.3.2. 众数分析

另一方面，本文方法中，众数的选择也是关键内容，众数的设置，与终端类型、发送频率、数据类型都有关系，不同众数下，插值法和本文方法的里程差异以及执行时间差异，也会有很大区别，具体如图 8 所示，本文结合目前已经运营的实际系统，选择众数为 5，得到的结果，无论与真实里程偏差，还是执行效率上，都能够满足系统的实时性要求。

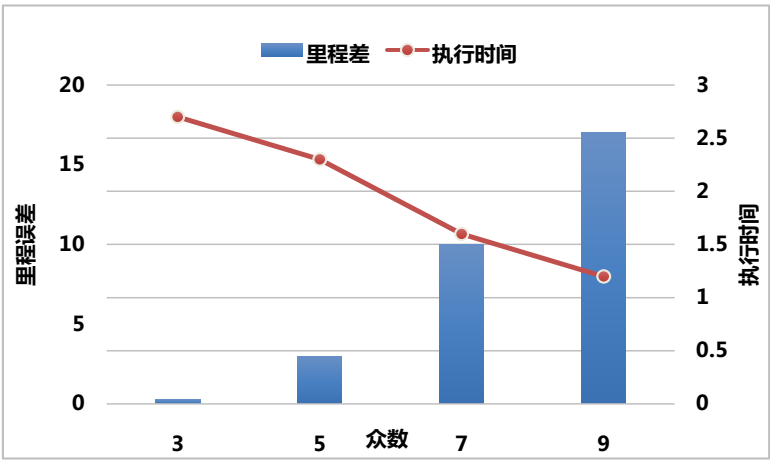


Figure 8. Comparison of execution results under different modes
图 8. 不同众数下方法执行结果比较

4. 结论

在强业务的应用系统中,例如轨迹里程与费用关联,或者对车辆行驶轨迹敏感的监管层面,因为北斗定位信号受干扰等情况,导致轨迹丢失会产生严重的影响,基于此,本文提出一种基于北斗的车辆行驶轨迹丢失检测以及补偿应用方法,该方法可以有效地解决由于北斗信号弱、车辆转弯、丢点等问题导致的轨迹丢失问题。实验结果表明该方法在保证算法的同时能有效地解决车辆轨迹丢失问题,具有很强的实用价值,目前本文提出的方法已经应用在实际场景中,为 60,000 多台北斗定位终端提供轨迹检测和补偿服务。后续的工作将围绕: 1) 轨迹丢失场景深度分析[8]; 2) 轨迹补偿算法优化,建立轨迹库[9][10]; 3) 服务稳定性等工作中做深入的探讨。

本文是在安徽省车联网共享数据中心与运营管理云服务平台(新能源汽车暨智能网联汽车产业技术创新工程项目)项目资助下开展,相关研究成果已经应用在实际产品中,获得良好的效果。

参考文献

- [1] 郝汪洋, 申飞, 张俊, 罗东锋, 刘静. 基于卫星定位信号的车辆行驶里程计算方法[J]. 计算机系统应用, 2017, 26(10): 201-206.
- [2] 刘小洋, 伍民友. 车联网: 物联网在城市交通网络中的应用[J]. 计算机应用, 2012, 32(4): 900-904.
- [3] Gerla, M., Lee, E.K., Pau, G., et al. (2014) Internet of Vehicles: From Intelligent Grid to Autonomous Cars and Vehicular Clouds. *Proceedings of 2014 IEEE World Forum on Internet of Things (WFIoT)*, Seoul, 6-8 March 2014, 241-246. <https://doi.org/10.1109/WF-IoT.2014.6803166>
- [4] 王保云. 物联网技术研究综述[J]. 电子测量与仪器学报, 2009, 23(12): 1-7.
- [5] 黄刚, 翟丽红. 智能车弯道行进算法的研究与实现[J]. 仪器仪表学报, 2019(2): 122-131.
- [6] 孙玉砚. 车辆轨迹分析关键技术研究[D]: [硕士学位论文]. 北京: 中国科学院大学, 2015.
- [7] 罗成, 陈淑荣, 剧季. 基于 GPRS 传输的车辆远程监控及行驶轨迹优化[J]. 微型机与应用, 2017, 36(11): 60-63.
- [8] 毛嘉川. 弯道宽度对车辆行驶轨迹和速度的影响规律[J]. 交通科学与工程, 2011(4): 78-84.
- [9] 郑剑炜, 顾晶晶, 庄毅. 基于生存分析的 GPS 轨迹缺失规律挖掘[J]. 计算机科学, 2018, 45(5): 192-196.
- [10] 龚旭东. 轨迹数据相似性查询及其应用研究[D]: [博士学位论文]. 合肥: 中国科学技术大学, 2015.