(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103699680 A
(43)申请公布日 2014.04.02

(21)申请号 201310753688.6

(22)申请日 2013.12.31

(71)申请人 中国科学院深圳先进技术研究院
地址 518055 广东省深圳市南山区西丽大学城学苑大道 1068 号
申请人 深圳市易行网交通科技有限公司
中科文讯科技（深圳）有限公司

(72)发明人 邹瑜斌 张帆 张昕 胡斌 须成忠

(74)专利代理机构 深圳市科进知识产权代理事务所（普通合伙）44316
代理人 沈祖锋 郝明琴

(51)Int.Cl.
G06F 17/30(2006.01)
G06F 9/46(2006.01)

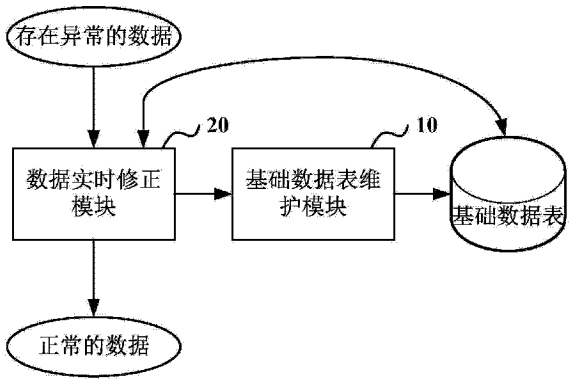
权利要求书2页 说明书6页 附图3页

(54)发明名称

公交车实时地理信息数据清洗方法及系统

(57)摘要

本发明涉及智能公交技术领域，提供了一种公交车实时地理信息数据清洗方法及系统。其中公交车实时地理信息数据清洗方法包括如下步骤：初始状态下，根据实时数据建立基础数据表，所述基础数据表保存实时数据中各类数据记录的历史信息；基于所述基础数据表实时修正不断到达的数据记录；动态更新所述基础数据表。本发明实现了对实时公交车地理信息数据的可靠实时清洗，为公交车数据的实时计算提供基础。



CN 103699680 A

1. 一种公交车实时地理信息数据清洗方法，其特征在于，包括如下步骤：

初始状态下，根据实时数据建立基础数据表，所述基础数据表保存实时数据中各类数据记录的历史信息；

基于所述基础数据表实时修正不断到达的数据记录；

动态更新所述基础数据表。

2. 如权利要求 1 所述的公交车实时地理信息数据清洗方法，其特征在于，所述基于所述基础数据表实时修正不断到达的数据记录的步骤具体包括：

使用基础数据表修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表以保证数据一致性。

3. 如权利要求 1 所述的公交车实时地理信息数据清洗方法，其特征在于，所述根据实时数据建立基础数据表的步骤、基于所述基础数据表实时修正不断到达的数据记录的步骤和动态更新所述基础数据表的步骤均由多个线程或进程根据特定的机制协同执行，所述线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。

4. 如权利要求 3 所述的公交车实时地理信息数据清洗方法，其特征在于，所述顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，保证各个线程或进程所接收的数据量大小一致；

所述哈希分配方法根据数据的多个指定属性，计算所述数据的哈希值，根据哈希值 %m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

5. 如权利要求 1 所述的公交车实时地理信息数据清洗方法，其特征在于，所述基础数据表中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，所述历史信息用于修正该类特定属性的数据记录的异常，所述基础数据表有多个。

6. 如权利要求 1 所述的公交车实时地理信息数据清洗方法，其特征在于，所述初始状态下，根据实时数据建立基础数据表的步骤具体为：用户使用已有知识建立基础数据表；

所述动态更新所述基础数据表的步骤具体为：根据实时的正常数据记录动态更新所述基础数据表，所述正常数据记录包括已被修正的异常数据记录。

7. 一种公交车实时地理信息数据清洗系统，其特征在于，包括：

基础数据表维护模块，所述基础数据表维护模块根据实时数据建立并动态更新基础数据表，所述基础数据表保存实时数据中各类数据记录的历史信息；

数据实时修正模块，所述数据实时修正模块基于基础数据表实时修正不断到达的数据记录。

8. 如权利要求 7 所述的公交车实时地理信息数据清洗系统，其特征在于，所述数据实时修正模块使用基础数据表修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表以保证数据一致性。

9. 如权利要求 7 所述的公交车实时地理信息数据清洗系统，其特征在于，所述基础数据表维护模块和数据实时修正模块由多个线程或进程根据特定的机制协同执行，每个模块内部的线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。

10. 如权利要求 9 所述的公交车实时地理信息数据清洗系统，其特征在于，所述顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，保证各个线程或进程所接收的数

据量大小一致；

所述哈希分配方法根据数据的多个指定属性，计算所述数据的哈希值，根据哈希值 %m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

11. 如权利要求 7 所述的公交车实时地理信息数据清洗系统，其特征在于，所述基础数据表中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，所述历史信息用于修正该类特定属性的数据记录的异常，所述基础数据表有多个；

初始状态时，基础数据表由用户使用已有知识建立，后续过程中基础数据表维护模块根据实时的正常数据记录动态更新所述基础数据表，所述正常数据记录包括已被数据实时修正模块修正的异常数据记录。

公交车实时地理信息数据清洗方法及系统

【技术领域】

[0001] 本发明涉及智能公交技术领域，特别是涉及一种公交车实时地理信息数据清洗方法及系统。

【背景技术】

[0002] 近年来城市机动车的数量急速增长，引发了许多诸如堵车、停车难、打车难等严重影响老百姓出行质量的问题。同时，城市的交通网络也日趋复杂，对一个完善的管理系统的要求越来越高。在未来构建智能城市的蓝图中，智能交通可谓是重中之重。

[0003] 大数据时代的到来为智能交通的建立提供了一个契机，其中公交车数据的使用价值巨大。由于公交车是城市的主流交通工具之一，公交车实时数据的计算可以为市民提供到站预测，耗时预测的多种服务。进一步地，公交车行驶路线固定，公交车实时数据可以用于计算道路的实时路况。然而，由于设备缺陷和网络信号不稳定等客观原因，公交车数据常常包含了很多异常的位置信息和状态信息，极大地干扰了公交车数据的实时计算的准确性，因此，公交车地理信息数据的实时地正确计算很大程度上取决于如何有效地并且实时地进行数据清洗。

[0004] 公交车地理信息数据主要有以下特点：

[0005] (1)数据量大，由于公交车数量众多，每秒产生的数据量巨大，因此要求能够正确快速地清洗这些数据；

[0006] (2)异常数据复杂，全球定位系统(Global Positioning System，GPS)设备种类众多，受到卫星定位的精度、定位设备的限制、网络信号等多种客观并且不可预知的因素影响，导致数据中存在大量不可预知的数据，异常数据的种类繁多，因此要求能够正确识别并且修正这些异常。

[0007] 现有的常用方法把接收的数据保存在存储介质中，由后台处理单元在固定的时间间隔进行批量地修正。由于现有技术使用批量处理实时接收到的数据，有如下缺陷：1、无法充分利用实时计算能力，导致后期计算量巨大；2、无法使用动态的信息组合成先验知识对数据进行智能修正。

[0008] 鉴于此，克服上述现有技术所存在的缺陷是本技术领域亟待解决的问题。

【发明内容】

[0009] 本发明要解决的技术问题是提供一种公交车实时地理信息数据清洗方法及系统，以实现对实时公交车地理信息数据的可靠实时清洗，为公交车数据的实时计算提供基础。

[0010] 本发明采用如下技术方案：

[0011] 一种公交车实时地理信息数据清洗方法，包括如下步骤：

[0012] 初始状态下，根据实时数据建立基础数据表，所述基础数据表保存实时数据中各类数据记录的历史信息；

[0013] 基于所述基础数据表实时修正不断到达的数据记录；

[0014] 动态更新所述基础数据表。

[0015] 进一步地，所述基于所述基础数据表实时修正不断到达的数据记录的步骤具体包括：

[0016] 使用基础数据表修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表以保证数据一致性。

[0017] 进一步地，所述根据实时数据建立基础数据表的步骤、基于所述基础数据表实时修正不断到达的数据记录的步骤和动态更新所述基础数据表的步骤均由多个线程或进程根据特定的机制协同执行，所述线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。

[0018] 进一步地，所述顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，保证各个线程或进程所接收的数据量大小一致；

[0019] 所述哈希分配方法根据数据的多个指定属性，计算所述数据的哈希值，根据哈希值 %m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

[0020] 进一步地，所述基础数据表中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，所述历史信息用于修正该类特定属性的数据记录的异常，所述基础数据表有多个。

[0021] 进一步地，所述初始状态下，根据实时数据建立基础数据表的步骤具体为：用户使用已有知识建立基础数据表；

[0022] 所述动态更新所述基础数据表的步骤具体为：根据实时的正常数据记录动态更新所述基础数据表，所述正常数据记录包括已被修正的异常数据记录。

[0023] 本发明还提供了一种公交车实时地理信息数据清洗系统，包括：

[0024] 基础数据表维护模块，所述基础数据表维护模块根据实时数据建立并动态更新基础数据表，所述基础数据表保存实时数据中各类数据记录的历史信息；

[0025] 数据实时修正模块，所述数据实时修正模块基于基础数据表实时修正不断到达的数据记录。

[0026] 进一步地，所述数据实时修正模块使用基础数据表修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表以保证数据一致性。

[0027] 进一步地，所述基础数据表维护模块和数据实时修正模块由多个线程或进程根据特定的机制协同执行，每个模块内部的线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。

[0028] 进一步地，所述顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，保证各个线程或进程所接收的数据量大小一致；

[0029] 所述哈希分配方法根据数据的多个指定属性，计算所述数据的哈希值，根据哈希值 %m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

[0030] 进一步地，所述基础数据表中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，所述历史信息用于修正该类特定属性的数据记录的异常，所述基础数据表有多个；

[0031] 初始状态时，基础数据表由用户使用已有知识建立，后续过程中基础数据表维护模块根据实时的正常数据记录动态更新所述基础数据表，所述正常数据记录包括已被数据

实时修正模块修正的异常数据记录。

[0032] 与现有技术相比，本发明的有益效果在于：本发明可以对海量实时公交车地理信息数据进行实时地并且可靠地清理和修正，以满足数据计算的正确性要求，为公交车数据的实时计算提供基础，其计算规模可扩展，鲁棒性强。

【附图说明】

[0033] 图 1 是本发明实施例的公交车实时地理信息数据清洗方法流程图；

[0034] 图 2 是本发明实施例的公交车实时地理信息数据清洗系统结构框图；

[0035] 图 3 是 line_id 的两种修正方法示意图；

[0036] 图 4 是 bus_pos 的一种修正方法示意图；

[0037] 图 5 是 cur_station 的五种修正方法示意图；

[0038] 图 6 是 next_station 的四种修正方法示意图。

【具体实施方式】

[0039] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0040] 此外，下面所描述的本发明各个实施方式中所涉及到的技术特征只要彼此之间未构成冲突就可以相互组合。

[0041] 如图 1 所示，本发明实施例提供了一种公交车实时地理信息数据清洗方法，包括如下步骤：

[0042] 步骤 S10：初始状态下，根据实时数据建立基础数据表(Base Data Table，BDT)，基础数据表 BDT 保存实时数据中各类数据记录的历史信息；

[0043] 具体地，基础数据表 BDT 存储在一数据库中，其中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，该历史信息用于修正该类特定属性的数据记录的异常，基础数据表 BDT 有多个。用户可使用已有知识建立基础数据表 BDT。

[0044] 步骤 S20：基于基础数据表 BDT 实时修正不断到达的数据记录；

[0045] 具体地，使用基础数据表 BDT 修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表 BDT 以保证数据一致性。

[0046] 步骤 S30：动态更新基础数据表 BDT。

[0047] 具体地，根据实时的正常数据记录动态更新基础数据表 BDT，正常数据记录包括已被修正的异常数据记录。

[0048] 步骤 S10-S30 均可由多个线程或进程根据特定的机制协同执行，线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，保证各个线程或进程所接收的数据量大小一致；哈希分配方法根据数据的多个指定属性，计算数据的哈希值，根据哈希值 %m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

[0049] 如图 2 所示，本发明实施例还提供了一种公交车实时地理信息数据清洗系统，其

包括基础数据表维护模块 10 和数据实时修正模块 20。基础数据表维护模块 10 根据实时数据建立并动态更新基础数据表 BDT，基础数据表 BDT 保存实时数据中各类数据记录的历史信息；数据实时修正模块 20 基于基础数据表 BDT 实时修正不断到达的数据记录，具体地，数据实时修正模块 20 使用基础数据表 BDT 修正不断到达的数据记录，抛弃最终仍无法修正的数据记录，并适当修改基础数据表 BDT 以保证数据一致性。

[0050] 基础数据表维护模块 10 和数据实时修正模块 20 由多个线程或进程根据特定的机制协同执行，每个模块内部的线程或进程之间通过顺序分配方法或哈希分配方法进行数据分配。顺序分配方法根据数据的到达顺序依次分发给各个线程或进程，最大限度地保证各个线程或进程所接收的数据量大小一致；哈希分配方法根据数据的多个指定属性，计算数据的哈希值 HashValue，根据 HashValue%m 的计算结果来决定数据所被分发的线程或进程，其中 m 为线程数或进程数。

[0051] 在一优选实施例中，基础数据表维护模块 10 使用哈希分配方法，分发所接收到的数据给本模块的各个线程或者线程，数据实时修正模块 20 使用顺序分配方法，分发所接收到的数据给本模块的各个线程或者线程。

[0052] 基础数据表 BDT 中每个表项为一个 key-value 结构，其中 key 记录了一种特定数据记录属性，value 保存了该类特定属性的数据记录的历史信息，该历史信息用于修正该类特定属性的数据记录的异常，根据具体的修正算法，可以有多个基础数据表 BDT。初始状态时，基础数据表 BDT 可由用户使用已有知识建立，后续过程中基础数据表维护模块 10 根据实时的正常数据记录动态更新基础数据表 BDT，正常数据记录包括已被数据实时修正模块 20 修正的异常数据记录。

[0053] 下面通过一优选实施例对本发明进行进一步地详细说明：

[0054] 在本实施例中，为了简要描述，把多维的公交车实时地理信息数据记录 bus_record 简化为一个五元组 {bus_id, line_id, bus_pos, cur_station, next_station}，其中：

[0055] bus_id 是公交车 id；

[0056] line_id 是该公交车当前正在执行的线路 id；

[0057] bus_pos 为该公交车的 GPS 坐标；

[0058] cur_station 为该公交车当前所在的站点 id 或者上一个站点 id；

[0059] next_station 为该公交车将要前往的站点 id。

[0060] 如果把全部公交车的行驶路线抽象成一张无向图 TransportGraph，则每个站点是 TransportGraph 中的一个节点 N，N_i 和 N_j 相邻意味着存在一条公交路线依次驶经 N_i 和 N_j，且称 N_i 和 N_j 之间存在着连接 edge。

[0061] 在系统初始化节点，基础数据表 BDT 将会被建立，包括线路信息表 lineinfotb，站点信息表 stationinfotb，相邻站点表 neighborstb，这三张表逻辑上构建了 TransportGraph。公交车日志表 buslogtb，站点日志表 stationlogtb，路径日志表 edgelogtb。其中：

[0062] 线路信息表 lineinfotb 的每个表项为一个 key-value 结构，key 为线路 id，value 为该线路所途经的站点 id。

[0063] 站点信息表 stationinfotb 的每个表项为一个 key-fields 结构，key 为站点 id，

fields 为 {GPS 坐标,站点名 }。

[0064] 相邻站点表 neighborstb 的每个表项为一个 key-set 结构,key 为站点 id,set 为该站点相邻的其他站点 id 集合

[0065] 公交车日志表 buslogtb 的每个表项为一个 key-list 结构,key 为公交车 id,list 为该公交车已经途经的站点 id。

[0066] 站点日志表 stationlogtb 的每个表项为一个 key-list 结构,key 为站点 id,list 为该站点已经发出的公交车 id。

[0067] 路径日志表 edgelogtb 的每个表项为一个 key-list 结构,key 为“站点 id1 - 站点 id2”,list 为正在从 id1 行驶到 id2 的公交车 id。

[0068] 实时数据记录将会被不断地输入数据实时修正模块 20,模块内部使用顺序分配的方式分发数据记录给不同的线程或者进程。对于一个数据记录 {bus_id,line_id,bus_pos,cur_station,next_station},如果存在不合法的字段,则数据实时修正模块 20 会使用如图 3- 图 6 的方法修正这些字段。如图 3 所示,line_id 的两种修正方法表示 line_id 可由两种方式推导得出,第一种方式是用 cur_station 和 next_station,基于基础信息表 lineinfotb 来推导得出 ;第二种方式是用 bus_id 基于基础信息表 buslogtb 来推导得出,图 4- 图 6 中的推导方法与图 3 中的方法类似,此处不再赘述。如果字段修正最终失败,则该数据记录会被丢弃。

[0069] 数据实时修正模块 20 将会向基础数据表维护模块 10 输出正常的数据记录(包括已修正的正常数据记录),基础数据表维护模块 10 会使用哈希分配的方法分配所接收到的数据给本模块的各个线程或者进程,各个线程和进程会根据数据记录的信息动态更新公交车日志表 buslogtb,站点日志表 stationlogtb,路径日志表 edgelogtb,其中 :

[0070] 公交车日志表 buslogtb :更新该表 key 为 bus_id 的表项,把 cur_station 放在该表项 list 的尾部。

[0071] 站点日志表 stationlogtb :更新该表 key 为 cur_station 的表项,把 bus_id 放在该表现 list 的尾部。

[0072] 路径日志表 edgelogtb :更新该表 key 为“cur_station-next_station”的表项,把 bus_id 放入该表项 list 的尾部。

[0073] 实验验证 :

[0074] 使用 Storm 流式计算工具实现了本发明,所处理的实时数据来源于深圳市 5000 辆实时的公交车地理位置数据,每秒接收 500 条数据记录,采用 2 个物理节点,结构中各个模块使用的线程数为 :数据实时修正模块(4 个)、基础数据表维护模块(4 个),存储了基础数据表的数据库使用了 redis 内存数据库。

[0075] 本发明实施例可以对海量实时公交车地理信息数据进行实时地并且可靠地清理和修正,以满足数据计算的正确性要求,为公交车数据的实时计算提供基础,其计算规模可扩展,鲁棒性强。

[0076] 值得说明的是,上述装置和系统内的模块、单元之间的信息交互、执行过程等内容,与本发明的处理方法实施例基于同一构思,系统和方法的具体内容可相互参考实施例中的叙述。

[0077] 本领域普通技术人员可以理解实施例的各种方法中的全部或部分步骤是可以通

过程序来指令相关的硬件来完成，该程序可以存储于一计算机可读存储介质中，存储介质可以包括：只读存储器（ROM，Read Only Memory）、随机存取存储器（RAM，Random Access Memory）、磁盘或光盘等。

[0078] 以上所述仅为本发明的较佳实施例而已，并不用以限制本发明，凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等，均应包含在本发明的保护范围之内。

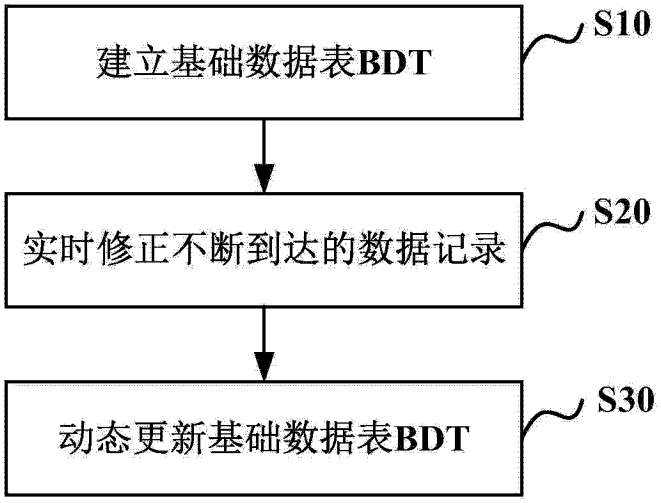

建立基础数据表BDT
S10
实时修正不断到达的数据记录
S20
动态更新基础数据表BDT
S30

图 1

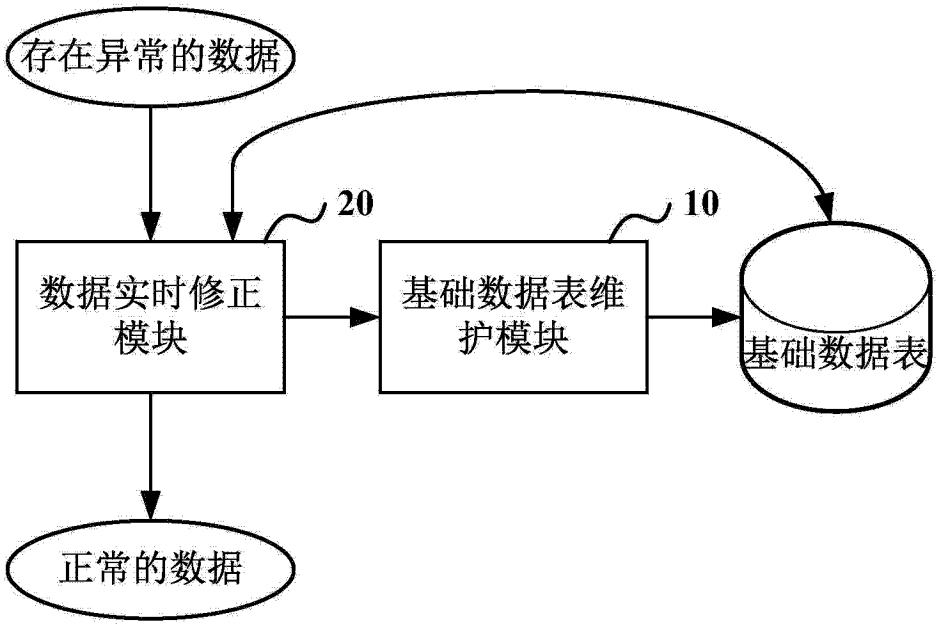

存在异常的数据
20
10
数据实时修正模块
基础数据表维护模块
基础数据表
正常的数据

图 2

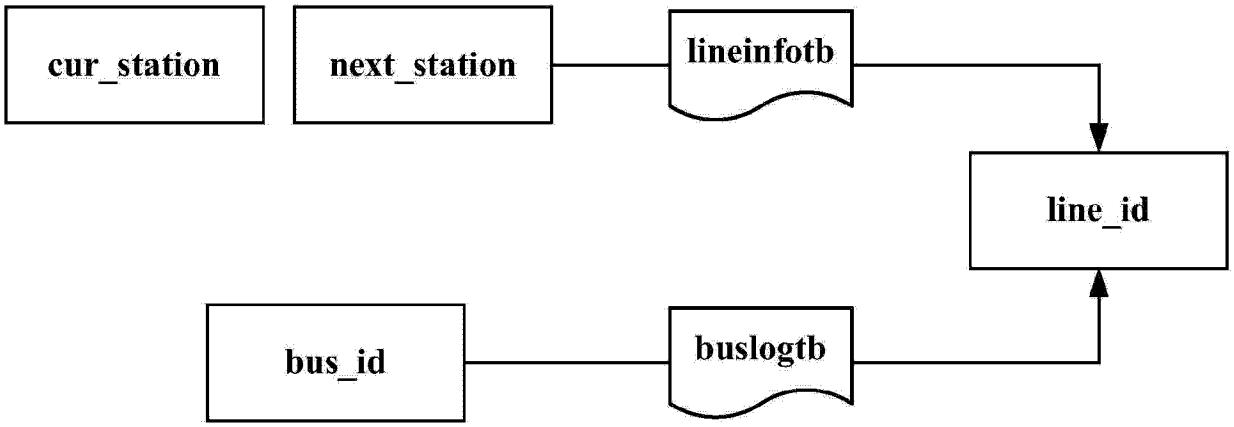

cur_station
next_station
lineinfotb
line_id
bus_id
buslogtb

图 3

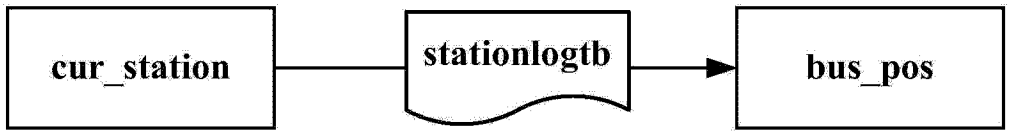

cur_station
stationlogtb
bus_pos

图 4

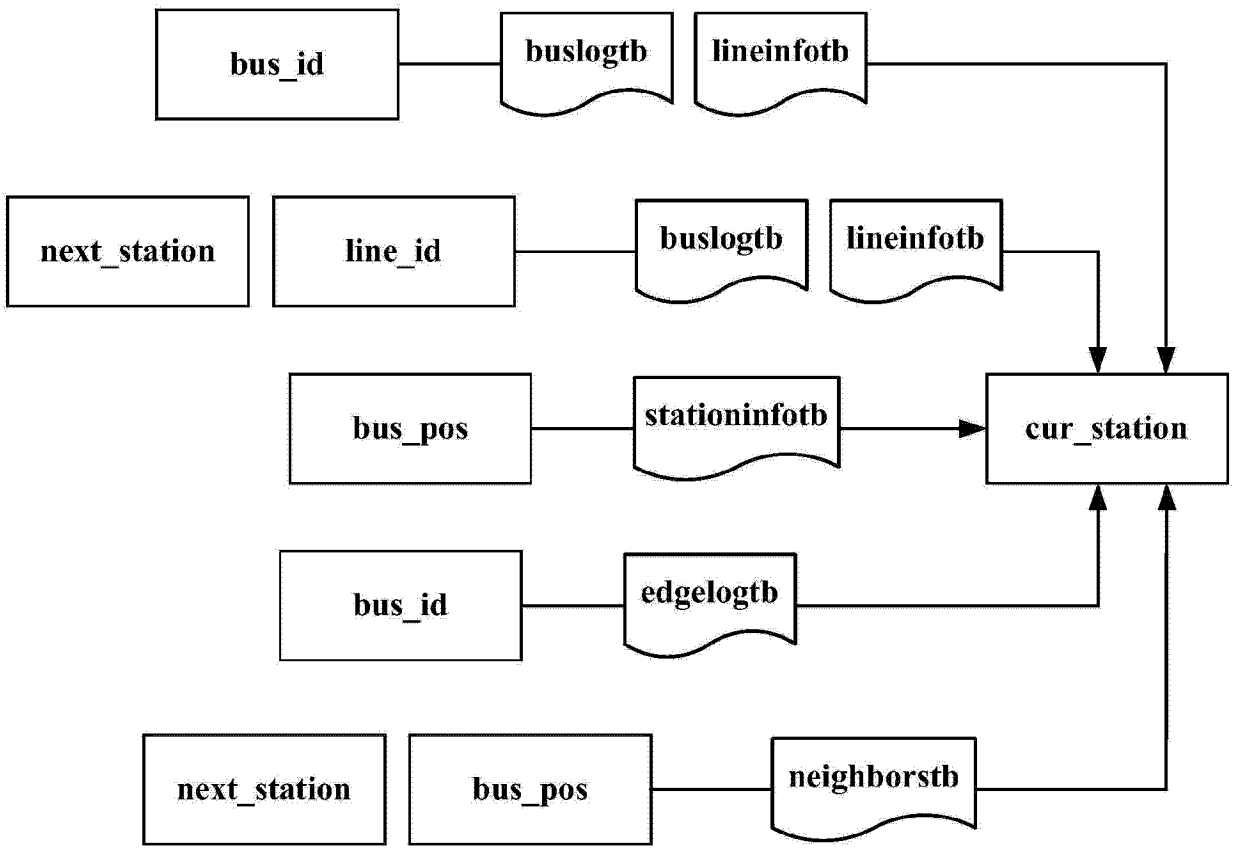

bus_id
buslogtb
lineinfotb
next_station
line_id
buslogtb
lineinfotb
bus_pos
stationinfotb
cur_station
bus_id
edgelogtb
next_station
bus_pos
neighborstb

图 5

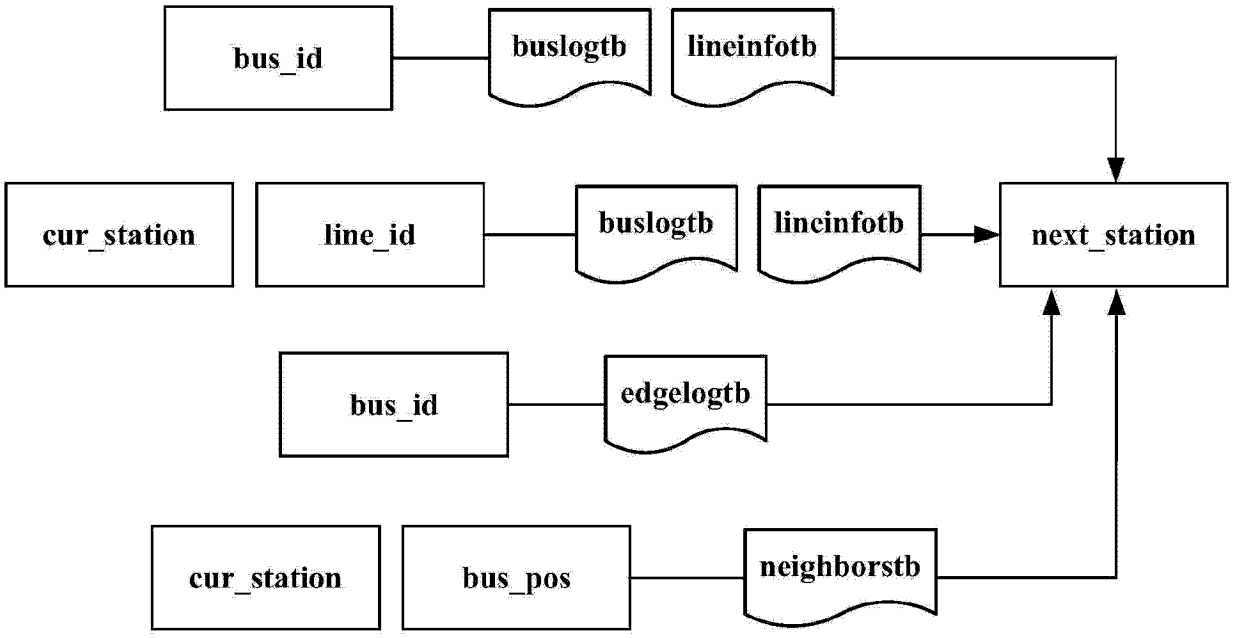

bus_id
buslogtb
lineinfotb
cur_station
line_id
buslogtb
lineinfotb
next_station
bus_id
edgelogtb
cur_station
bus_pos
neighborstb

图 6