

车辆导航系统高速检索技术研究

孙明媚, 张代远

(南京邮电大学 计算机学院, 江苏 南京 210003)

摘要:在车辆安全行驶中,危险区域与慢速行驶区域的自动检测是最为关键的问题,也是安全行驶的基本保证。传统方法通过卫星通讯将危险区域与慢速行驶区域信息发送到外出车辆,不仅应答速度较慢,并且数据在通信过程中存在风险。文中研究并提出了一种在车辆导航系统内部实现的高速数据检索技术,将区域数据压缩保存在车辆导航设备,以R树数据结构压缩保存实现区域数据的本地查询,这样数据库内容保存在车辆本身固件中,车辆导航系统每次检索只从服务器端更新数据库内容,使得检索可以在车辆内部实现,从而提高了检索效率,具有良好的实用价值。

关键词:车辆导航系统;区域信息数据;R树数据结构;高速数据检索

中图分类号:TP391.3

文献标识码:A

文章编号:1673-629X(2010)11-0193-04

High Speed Search Technology on Vehicle Navigation System

SUN Ming-mei, ZHANG Dai-yuan

(Coll. of Computer, Nanjing Univ. of Posts and Telecommunications, Nanjing 210003, China)

Abstract: In the safe driving of the vehicle, automatic detection of the danger zone and slow moving area is the most critical issue, but also the basic guarantee for safe driving. Traditional methods sent information of hazardous area and slow moving zone through satellite communication to vehicles out, it is not only have a slow response, and the data in the communication process are at risk. In this paper, present a navigation system within the vehicle to achieve high-speed data retrieval technology, data what are compressed and stored in the vehicle navigation equipment, in order to achieve local search by compressing using R-tree data structure, by this way the content of the database can be stored in the vehicle itself, each search of the vehicle navigation system can update the database content from the server-side, search can be achieved in the vehicle itself, thereby enhancing the retrieval efficiency, so it has a very good value.

Key words: vehicle navigation system; data of regional information; R-tree data structure; high-speed data retrieval

0 引言

车载检索系统是智能交通系统的一个重要组成部分,这一智能系统不仅能实现安全、通畅的驾驶,缓解或避免多余的交通堵塞,而且能使车辆的移动和使用变得更加方便,促进了汽车消费和使用环境的完善,有着广阔的前景。在车载业务中,地图上往往会预先设定一些区域,比如危险区域、慢速行驶区域。这些区域信息被保存在数据库中,随着车载业务的不断复杂,区域设置也会更复杂,检索需求越多。运行中地车辆必须实时、动态地根据自己的位置信息检索相关数据库,从而判断自己在哪些区域,根据结果采取行动。

而一些传统的做法效率并不高,它是把数据存放在数据库端,检索在数据库端执行,因此导航系统每次

查询都给服务器发送查询请求,这样以来,服务器压力就过大,同时检索程序必须支持大规模并发,不仅应答速度较慢,而且数据在通信过程中存在风险。文中提出了一种新的方法,把数据库内容保存在车载机固件中,从而车载机每次启动都从服务器端更新数据库内容,以便于每次检索都在车载机内部实现,这样就大大减少了检索所需要的时间,减轻了服务器的压力,从而提高了检索的效率。

1 问题建模

这个需求可以建模成检索问题^[1](见图1),可以看成在二维空间中有若干矩形,给定点 P ,判断 P 属于哪些矩形,这个问题在计算几何中也称作 Retangle stabbing problem。由于,每个矩形 (x_1, y_1, x_2, y_2) 有四个坐标值,需要做的就是如何对这些矩形进行排序。

2 算法理论

解决算法—^[2]:假设有 N 个矩形区域,每次检索

收稿日期:2010-01-22;修回日期:2010-04-22

作者简介:孙明媚(1984-),女,硕士,研究方向为计算机在通信中的应用;张代远,教授,博士,研究方向为神经网络、人工智能、计算机体系结构、计算机应用等。

遍历所有矩形进行判断,这样初始化时间为 $O(1)$,查找时间为 $O(N)$ 。这样查找速度是很慢的,以前是在高性能服务器端查找,如果在车载机中如此实现,速度是绝对不够的。

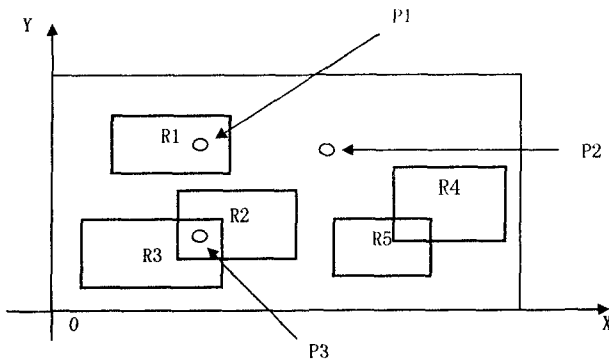


图 1 问题建模

解算法二^[3]:同样假设有 N 个区域,可以用区间树有效地对一组 $\text{interval}(x_1, x_2)$ 进行排序。对于 N 个区间,建树时间为 $N * O(\log N)$,每次查询时间 $O(\log N)$ 。但是区间树主要用于一维空间,如果要扩展到二维或者高维,则必须在每个一维区间树节点中再嵌套一个区间树,这样实现起来就不太可取。

根据算法二中区间树的思想,根据一维空间中的线段建立线段数,可以解决判断点属于哪些线段。对于二维矩形,分 X, Y 方向分别建立两颗线段数,对于 $P(x, y)$,分别判断 x, y 属于哪些矩形再合并结果。如果直接根据矩形区域划分二维空间,能不能找到类似空间树这样的结构呢?

Antonin Guttman (University of California Berkeley) 1984 在论文《R-trees: a dynamic index structure for spatial searching》中提出 R 树(Rectangle Tree)概念^[4]。R 树是一种用于空间检索的动态索引结构,类似于 B 树,适用于处理多维数据查找。2 维 R 树广泛应用于解决 Rectangle Stabbing Problem,高维 R 树可用于解决数据挖掘系统^[5]中的多关键字查询。

R 树常用的几个概念:

1. MBR:若干空间的最小外接矩形(MBR: Mini-

mum Boundary Rectangle): (x_1, y_1, x_2, y_2) 。

2. 节点:相当于数据库中的一个 page。

3. 叶节点:包含 m 个以上,最大 M 个记录。 $m \in [0, M/2]$ 。包含该节点所有矩形空间的 MBR,矩形的位置信息,矩形 id。

4. 非叶节点:包含 m 个以上,最大 M 个记录。 $m \in [0, M/2]$ 。包含该节点所有子节点空间的 MBR,指向子节点的指针。

5. 根节点:至少包含两个记录。

R 树图例^[6]如图 2 所示。

P 点查询^[7]示例如图 3 所示。

R 树的建树:追加矩形 x 。

从根节点开始顺序检索,对于每个非叶节点 y ,选择使其 mbr 面积^[8]增加最小的 y ,即: $\text{mbr}(\text{mbr}(x), y. \text{mbr}) - y. \text{mbr}$ 最小。到达叶节点时,如果叶节点未满足,则插入该矩形,并修改此叶节点的 mbr。如果叶节点已经满了,则要分割成两个节点,并且重新分配节点所指的矩形对象。

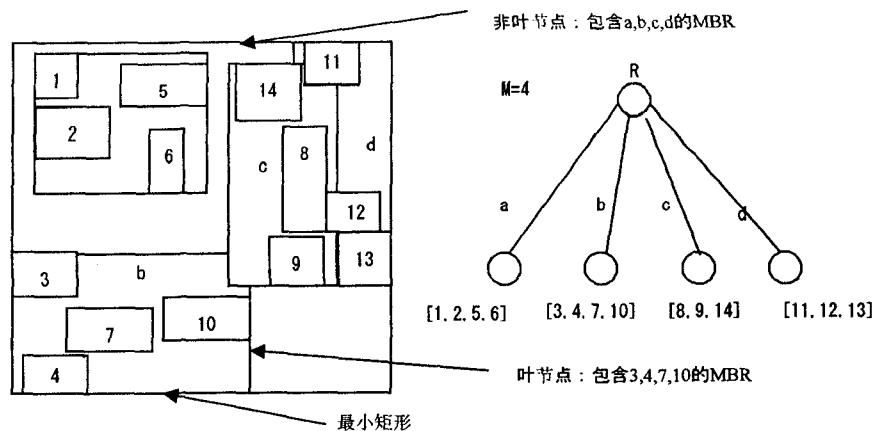


图 2 R 树图例

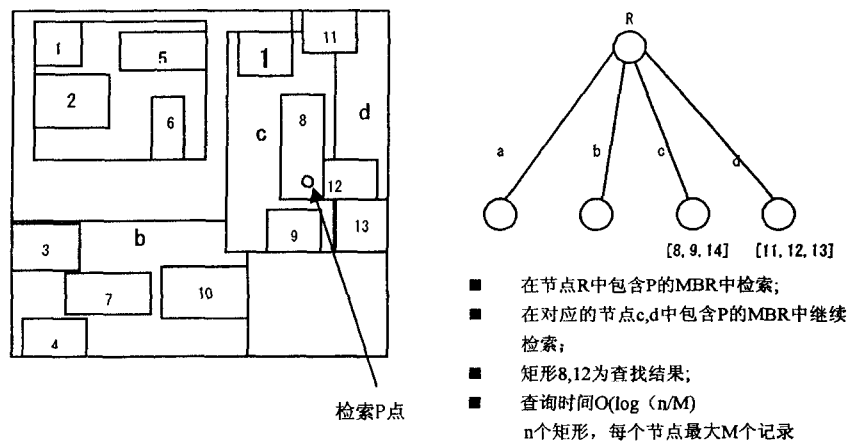


图 3 P 点查询示例

把节点分割成 v 和 v' ,其中 v 有 $m + i$ 个记录, v' 有 $m + 1 - i$ 个记录($0 \leq i \leq M$)。如果采用综合面积最小原则分割节点,就存在如下(a)、(b)两种分割标

准,但是这两种标准不可能同时满足(见图 4):

- (a) 两个节点对应的 mbr 的面积的综合最小:如果有 n 个矩形,要划分 2^n 次;
- (b) 两个节点对应的 mbr 的重合部分最小:2 次分割,不能保证最优,时间 $O(n^2)$ 。

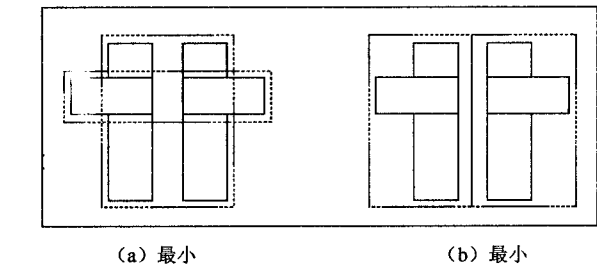


图 4 分割标准

这时继续追加矩形^[9](见图 5、图 6):

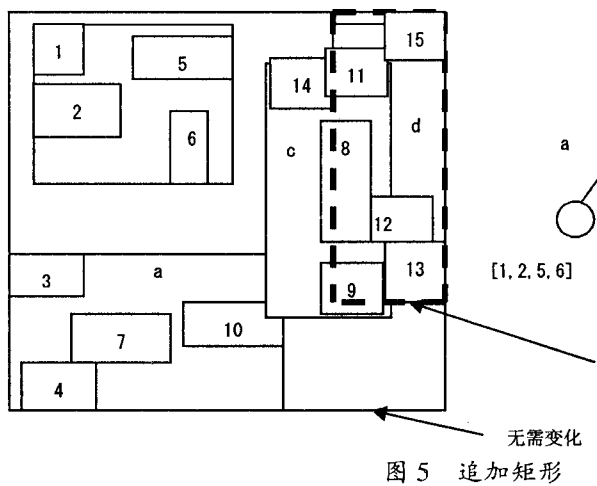


图 5 追加矩形

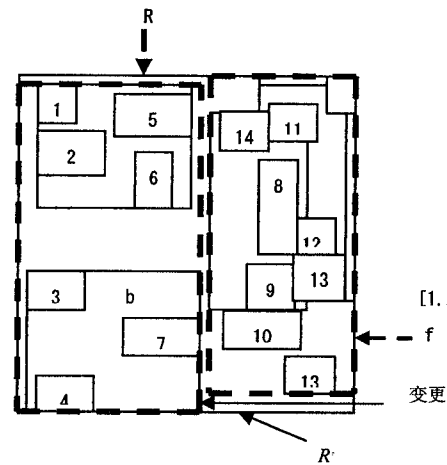


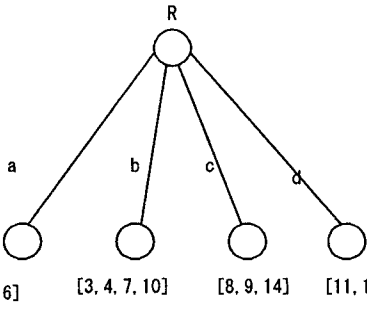
图 6 继续追加矩形

但是作为嵌入式系统,有严格的空间限制。

(2)内存管理^[11]:在 Rtree 的建树过程中,需要生成新的节点,而在车载系统中,malloc()一类的函数不能够使用,应该尽量实现简单的内存管理,避免内存碎片的产生。

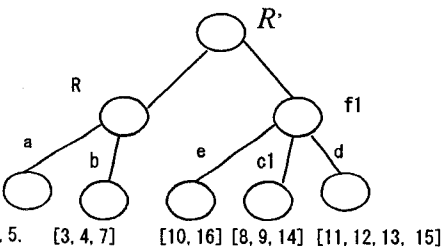
具体实现结构为:

```
struct Rect
{
    int boundary[NUMSIDES]; /* xmin,ymin
};
struct Node;
struct Branch
{
    struct Rect * rect;
    struct Node * child;
};
struct Node
{
    short count;
    short level; /* 0
    is leaf,
    struct Branch
    branch[MAXCARD];
};
struct Database
{
    int index; /* index for the
data */
    int * addr; /* memory ad-
dress */
    int rd_len;
    int rd_cnt;
    struct Node * root;
    int result_cnt;
    int result[100];
};
```



新的节点d

无需变化



- 节点b满了,先被分割b,e
- 节点R满了,再次被分割R,f
- 生成新的根节点R'

在 R 树的应用中,每次分配的内容主要是用于存放 Node 或者 Rect,分配的 Size 是固定的,所以可以开辟大数组,用 stack 的方式^[12]实现一个简单版本的内存管理。

```
struct Node memory_n[COUNT_N];
struct Rect memory_r[COUNT_R];
struct ListNode memory_ln[COUNT_LN];
```

3 车载应用

该算法在理论上可行,查询速度快,但是应用嵌入式系统中如下实际问题:

(1)空间限制^[10]:算法的高效往往是空间换时间,

```

short stack_n[COUNT_N]; /* store the
int stack_r[COUNT_R];
short stack_ln[COUNT_LN];

short top_n = COUNT_N - 1; /* the top
int top_r = COUNT_R - 1;
short top_ln = COUNT_LN - 1;

struct Node * allocateNode()
{
Short pos = pop_n();
return &memory_n[pos]; //start from the first el-
eme
}
Void freeNode(struct Node * node)
{
Short pos = (node - memory_n) / sizeof ( struct
Node);
memset(node, 0, sizeof(struct Node));
Push_n(pos);
}

```

实际运行结果为:

具体步骤	运行时间 (单位: μs)
tree_index	15m998
search: succeeds	33
search: fails	20

4 结束语

提出了在车辆检索技术中,通过把数据库内容保存在车载机固件中,每次检索都在车载机内部实现,这

样就大大提高了检索的效率,并提供了相应算法,给出了运行结果,与传统方法相比,运行时间大大缩短,并且算法易于实现。

参考文献:

- [1] Aranha R F M, Venkatesh G. Implementation of a Real-time Database System[J]. Information System, 1996, 21(1): 55 - 74.
- [2] Kam - Yiu L, Tei - Wei K. Evaluation of concurrency control strategies for mixed soft realtime database[J]. Information Systems, 2002, 27: 123 - 149.
- [3] Arulampalam S, Maskell S, Gordon N, et al. A tutorial on particle filters for online nonlinear/non Gaussian Bayesian tracking [J]. IEEE Trans on Signal Processing, 2002, 50(2): 174 - 188.
- [4] 杨述斌, 张 阳. 复杂车辆图像中的车牌快速形态定位法[J]. 计算机技术与发展, 2008, 18(6): 50 - 53.
- [5] 张文芊, 廖惠敏. 嵌入式网络连通检测器的实现[J]. 计算机技术与发展, 2008, 18(1): 240 - 242.
- [6] 吴 哲, 孙 涵. 基于快速直线段提取的道路标识线识别算法[J]. 计算机技术与发展, 2009, 19(5): 48 - 51.
- [7] 刘 俊, 杨全盛. 一种基于定位应用的无线传感器网络 MAC 层方案[J]. 计算机技术与发展, 2009, 19(1): 204 - 206.
- [8] 陈黄焱, 郑洪源. 基于 WebGIS 高速公路应急联动系统设计[J]. 计算机技术与发展, 2009, 19(8): 220 - 223.
- [9] 李华贵, 项志华, 何 伟, 等. 基于 GPS 和 GPRS 车载导航定位系统的实现[J]. 计算机技术与发展, 2006, 16(11): 241 - 242.
- [10] 潘成源, 徐 勇, 李鑫. 嵌入式 Linux 在车载导航系统中的应用[J]. 计算机技术与发展, 2006, 16(5): 223 - 225.
- [11] 陶玉贵. 车载 GPS 组合测速系统数据融合算法研究[J]. 计算机技术与发展, 2009, 19(1): 200 - 203.
- [12] 田 凯, 孟志军, 王武宏. 基于 Windows CE.net 的车载平台的定制开发[J]. 计算机技术与发展, 2006, 16(3): 45 - 46.
- [6] 刘 阳, 李 欣. 3D GIS 中空间数据可视化的研究与应用[J]. 计算机工程与设计, 2006, 27(6): 1090 - 1092.
- [7] 马照亭, 李成名, 王继周, 等. 海量地形可视化的研究现状与前景展望[J]. 测绘科学, 2006, 31(1): 134 - 136.
- [8] 戴晨光, 邓雪清, 张永生. 海量地形数据实时可视化算法[J]. 计算机辅助设计与图形学报, 2004, 16(11): 1605 - 1606.
- [9] Watt A. 3D 计算机图形学[M]. 包 宏译. 北京: 机械工业出版社, 2005.
- [10] David A, Ponce F J. Computer vision: modern approach[M]. Beijing: Publishing House of Electronics Industry, 2004.
- [11] Schneider P J, Eberly D H. Geometric Tools for Computer Graphics[M]. [s.l.]: Morgan Kaufmann, 2002.
- [12] 黄 桦, 吴健平. 基于 ArcScene 的建筑物日照分析模块的设计与实现[J]. 测绘与空间地理信息, 2008, 31(1): 86 - 89.
- [13] 吴 颖. 数码城市 GIS 中的日照分析研究[J]. 测绘通报, 2007(12): 62 - 65.
- [14] 王立言, 孙培巨, 阎 峰, 等. 日照分析软件在规划管理中的应用[J]. 青岛理工大学学报, 2007, 28(3): 111 - 113.

(上接第 192 页)

Computer Graphics, 1977, 11(2): 242 - 248.