基于网格索引的时空轨迹伴随模式挖掘算法

杨 阳 吉根林 鲍培明
(南京师范大学计算机科学与技术学院 南京 210023)

摘 要 时空轨迹伴随模式是数据挖掘领域的一项重要研究内容。CMC(Coherent Moving Cluster)算法是一种经典的时空轨迹伴随模式挖掘算法,该算法引入了DBSCAN算法以挖掘出任意形状的簇。但是,DBSCAN聚类算法极耗时,导致CMC算法的时间效率较低。因此提出了一种基于网格索引的时空轨迹伴随模式挖掘算法MAP-G(Mining Adjoint Pattern of spatial-temporal trajectory based on the Grid index)。实验表明,MAP-G算法不仅比CMC算法具有更高的时间效率,而且能够过滤掉部分不正确的结果,因此结果也更加准确。

关键词 伴随模式,时空轨迹挖掘,网格索引
中图法分类号 TP391.1 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.1.025

Algorithm for Mining Adjoint Pattern of Spatial-Temporal Trajectory Based on Grid Index

YANG Yang JI Gen-lin BAO Pei-ming
(School of Computer Science and Technology, Nanjing Normal University, Nanjing 210023, China)

Abstract In the field of data mining, adjoint pattern of spatial-temporal trajectory is an important research direction. CMC(Coherent Moving Cluster) algorithm is a classical algorithm for mining adjoint pattern, and it is applied to mine clusters of arbitrary shape. However, it reduces the efficiency of the algorithm. We presented an algorithm for mining adjoint pattern of spatial-temporal trajectory called MAP-G(Mining Adjoint Pattern of spatial-temporal trajectory based on the Grid index). The experimental results demonstrate that the proposed algorithm is more efficient compared to the CMC algorithm, and the accuracy is higher as our algorithm can filter some wrong results.

Keywords Adjoint pattern, Mining of spatial-temporal trajectory, Grid index

1 引言

随着卫星定位技术、无线通信以及跟踪检测设备的快速发展,人们能够方便地以低廉的代价获得时空轨迹数据[1]。移动对象的位置、属性都可能随着时间的推移而发生变化,人们不仅需要知道某一对象的属性和空间信息,更想要了解该对象的来龙去脉,以便对其形成原因作出评估,对未来情况进行预测。时空轨迹数据能够有效地表达移动对象的这些特性。通过分析各种不同对象的时空轨迹数据,有助于研究人类行为模式、交通物流、动物习性以及市场营销等。随着大数据时代的到来,从这些海量的时空轨迹数据中发现隐藏的知识和"有趣"的轨迹模式有重要意义。例如,挖掘出具有相同轨迹模式的卡车有助于物流规划;挖掘上班族上班路线的共同路段有助于公共交通的规划等[2,3]。

挖掘伴随模式是一种有效的分析时空轨迹模式的方法,即挖掘一起运动的超过设定时间长度阈值的移动对象群体[5]。时空轨迹伴随模式是时空数据轨迹模式中重要的组成部分,其在挖掘具有相同或相似运动模式的移动对象群体等方面有着广泛的应用。

本文第2节对现有的伴随模式挖掘算法进行概要介绍,并对目前的算法普遍存在的不足进行论述;第3节主要介绍本文提出的基于网格索引的时空轨迹伴随模式挖掘算法MAP-G的思想;第4节通过对比实验说明本文提出的MAP-G算法相较CMC算法[8]不仅在时间效率上有较大的提升,同时结果的准确性也更高;最后对全文所做的工作进行总结。

2 相关研究工作

2002年Patrick Laube等人提出了Group Concurrence和Flock的概念[6],并提出了基于运动属性矩阵的挖掘算法(运动属性主要指移动对象的运动方向)。Group Concurrence被定义为:给定一个对象集合O及阈值σ(σ通常大于50%),如果O中有超过σ个对象在连续k个采样时刻之间表现出相同或相似的运动趋势,那么这群对象形成的运动模式被称为Group Concurrence模式。Group Concurrence模式可以被视为伴随模式的雏形。在随后的十来年间人们不断地提出了改进的伴随模式的定义及相应的挖掘算法。其中,最具代表性的伴随模式包括Moving Cluster[7]、Convoy[2,8]、Swarm[5]以及Travelling Companion[9,10]等。纵观已有的伴随模式挖掘

到稿日期:2015-03-15 返修日期:2015-07-21 本文受国家自然科学基金项目(41471371)资助。
杨 阳(1988—),男,硕士,主要研究方向为数据挖掘及其应用,E-mail:yyang0108@163.com;吉根林(1964—),男,博士,教授,博士生导师,CCF高级会员,主要研究方向为数据挖掘及其应用,E-mail:glji@njnu.edu.cn(通信作者);鲍培明(1966—),女,硕士,副教授,硕士生导师,主要研究方向为数据库技术和智能算法。

算法，其大体上可以分为：基于运动属性[6,11,12]、基于 DBSCAN[2,3,5,8]和基于模型[9,10]这 3 类。

基于 DBSCAN 的方法是目前主流的伴随模式挖掘方法，CMC 算法是其中最常用的方法之一，因此提高该算法的时间效率具有重要意义。

CMC 算法主要包括聚类和取交集两个过程。其规定若至少 m 个移动对象伴随运动至少连续 k 个时刻，则这群移动对象构成伴随模式。图 1 所示的是 CMC 算法挖掘伴随模式的过程：(1)$t1$ 时刻聚类得到簇 C11，并将其作为候选存入 $V1$ 中；(2)$t2$ 时刻 O3 对象没有采样点，根据线性插值法为 O3 在 $t2$ 时刻插入虚拟采样点，最终聚类得到簇 C21，由于簇 C21 的候选簇 V1 中移动对象集合的交集数为 2，满足 $m=2$ 的要求，因此 $V1$ 被更新为 C11 和 C21 的交集；(3)在 $t3$ 时刻，聚类得到簇 C31 和 C32，$V1$ 与这两个簇分别取交集，只有 $V1$ 和簇 C32 的交集满足 $m=2$，且此时满足 $k=3$ 的要求，C31 作为新的候选簇存入 $V2$ 中。由于仅有 3 个采样时刻，因此输出满足要求的伴随模式即 C11、C21 和 C32 3 个簇中共同的移动对象集合 O1 和 O2 形成的时空轨迹模式构成了伴随模式。

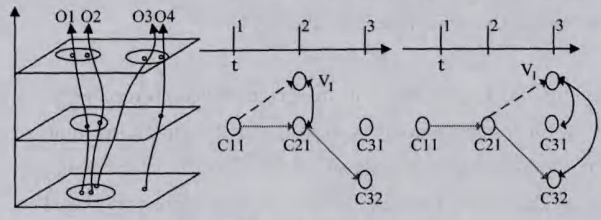

图 1　算法 CMC 挖掘过程演示

我们在实验中发现，CMC 算法中聚类过程的耗时占整个算法耗时的 95%以上，因此从降低聚类耗时入手来提高算法的效率。

根据 CMC 算法的思想，一群“滞留”于某个相对固定的区域内的移动对象群体构成的模式会被作为伴随模式的结果输出，但该结果并不正确。因此我们针对 CMC 算法的这一弊端提出相应的改进方法，以提高挖掘结果的准确性。

3　MAP-G 算法

3.1　相关概念

定义 1　设移动对象 O 的时空轨迹 $TR=\{P_1,P_2,\cdots,P_i,\cdots,P_n\}$，其中 $P_i=(t_i,x_i,y_i)$，t_i 表示采样时刻，x_i 表示经度，y_i 表示纬度。

定义 2　时空轨迹集合 $TS=\{TR_1,TR_2,\cdots,TR_n\}$，如果其中超过 m 个移动对象连续伴随运动时长超过 α，则这 m 个移动对象形成的时空轨迹模式便可以称为伴随模式。

定义 3　网格序列表示为 $GS=\{G_1,G_2,\cdots,G_i,\cdots,G_n\}$，其中 $G_i=(gid,t_s,t_e)$。gid 表示网格标识，t_s 表示移动对象 O 进入网格 gid 时后第一个采样点的采样时刻，t_e 表示移动对象离开该网格前最后一个采样点的采样时刻。

3.2　算法思想

MAP-G 算法主要包括网格索引构建、轨迹转换以及伴挖掘随模式等过程。图 2 所示为 MAP-G 算法的执行流程。

通过网格索引不仅可以将原始轨迹点序列转化为网格序列，从而大大降低需要处理的数据量，有效地提高算法的时间效率；而且可以提高算法搜索候选伴随模式对象集合的效率，从而提高算法的时间效率。

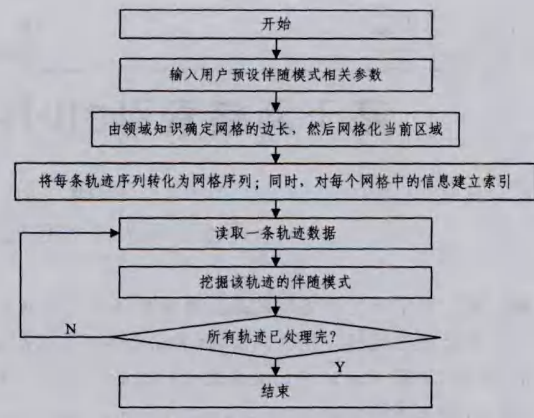

图 2　算法 MAP-G 执行流程

现有的多数算法可能会将部分非伴随模式的结果作为伴随模式输出，原因如下：现有的算法通常是先依次对每个时刻点的采样数据进行聚类，然后与候选项集合中的簇取交集并更新候选项集合，再然后对下一个时刻点的采样数据进行聚类，再与候选项集合中的簇进行取交集操作，最后对结果进行相应的处理。但是，现有的算法只在聚类阶段考虑到了采样点的地理位置的属性，在挖掘伴随模式阶段只考虑了簇之间的交集关系，忽略了地理位置属性。假设被挖掘的对象集合在连续的采样时刻“滞留”于某区域内，即保持在原地不动或者在很小的范围内运动，那么该情况实际上不能认定为是伴随模式的。本文提出的 MAP-G 算法能去除这种不正确的结果，因此结果的准确性也相对较高。

3.2.1　网格索引的建立

如图 3 所示，整个矩形可以看作是挖掘区域，假设经过实验得出该区域应该按 3×4 的粒度划分。首先分别对每个网格进行编号，则图中的一条原始轨迹点序列便可以转化为网格序列：{(1,1),(2,1),(2,2),(2,3),(3,3),(3,4),(2,4)}，从而降低了算法需要处理的数据量的大小，因此提高了时间效率。

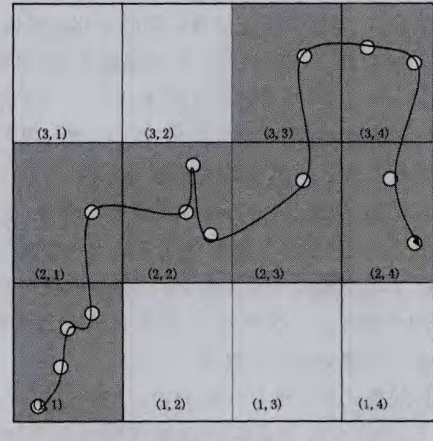

图 3　网格索引示例

在构建网格索引的过程中，网格大小的确定是核心问题。因为如果单个网格的面积过小，势必会导致丢失很多本该属于同一簇的候选对象的情况发生，最终导致伴随模式结果的正确性大大降低；与此相反，如果单个网格的面积过大，虽然最终结果的正确性能够得到保证，但是由于每个网格中的移动对象数量太多，因此在轨迹转换阶段以及伴随模式挖掘阶段，算法的时间效率较低。

3.2.2 轨迹转换处理

轨迹转换处理是 MAP-G 算法的重要步骤，关系到算法输出结果的准确性及时间效率。由于硬件的限制或者网络延时等原因，同时也可能由于每个移动对象的采样间隔不一致等原因，因此对于某个移动对象的轨迹序列来说，可能会出现在某些网格中只有一个采样点甚至没有采样点的情况，这两种异常情况是轨迹转换算法需要处理的关键问题。

算法 1 轨迹转换

输入：轨迹序列 TR={P_1,P_2…P_n}，时间增量 α(α 等于平均采样间隔时间)

输出：网格序列 GS={G_1,G_2,…,G_j,…,G_k}

过程：

```
1. for( i=0;i<n;i++)
2.    gid=compute(Pi. x,Pi. y);//计算轨迹点 Pi 所在网格 id 号
3.    flag=judge(gid);//如果当前采样点进入不同的网格，则 flag 为 true，否则为 false
4.    if(flag)
5.       output(Gj);//输出与网格 gid 有关的信息，其中 Gj=(gid, ts-α,te+α)
6.    else
7.       te= Pi. t;
8. endfor
```

算法 1 描述了将一条移动对象时空轨迹点序列 TR 转化为对应的网格序列 GS 的过程。

移动对象的时空轨迹序列转换为网格序列的主要步骤如下：(1)读取 TR 的第一个采样点 P_1，计算得到 P_1 所在网格的编号 *gid*；(2)如果下一个采样点 P_2 仍在编号为 *gid* 的网格中，则更新 P_1 在网格 gid 中的最后采样时刻 t_e；(3)如果 P_2 进入新的网格 *newgid* 中，则输出与网格 *gid* 有关的信息，其中包括网格 *id*、进入该网格的时刻 $t_s-\alpha$ 以及离开该网格的时刻 $t_e+\alpha$。由于移动对象几乎不可能恰好在进入和离开网格 *gid* 边界的时刻发射出采样信息，因此 t_s 要减去时间增量 α，而 t_e 要加上时间增量 α。

有两种特殊情况需要处理：(1)如果出现无采样点的网格，则直接跳过该网格，从下一个网格重新执行轨迹序列转换步骤；(2)如果网格中只有一个采样点，设采样点 P_i 的采样时刻为 t，则 t_s 和 t_e 分别为 $t-\alpha$ 和 $t+\alpha$；最终，这条时空数据采样点序列 TR 便转化成了对应的网格序列 GS。

3.2.3 挖掘伴随模式

经过以上步骤后，程序便开始挖掘伴随模式的步骤。该步骤的功能是对预处理完成后的时空轨迹数据进行时空轨迹伴随模式的挖掘。

算法 2 挖掘伴随模式

输入：移动对象 O_j 的网格序列 GS={G_1,G_2,…,G_k}，区域范围参数{X1,Y1,X2,Y2}，阈值 m,dur 以及 len

输出：O_j 的伴随模式 AP

过程：

```
1. for(i=0;i<k;i++)
2.    gid=Gi. gid;
3.    num=get_num(8 近邻);//获得网格 gid 及其 8 近邻网格中符合要求的对象数量，要求对象的采样时刻 t 满足 Gi. ts≤t≤Gi. te
4.    if(num≥m)
5.       cand_ap=候选对象集合
6.       save(cand_ap);//将候选伴随模式保存到 cand_ap 中
7.    else
8.       flag=judge(cand_ap);//判断候选伴随模式是否符合伴随模式条件，符合则 flag 为 true，否则为 false
9.       if(flag)
10.         output(AP);//输出伴随模式 AP
11.         sign(AP);//将 AP 结果中的所有轨迹对象 TR 标记为 RE，程序遇到有该标记的 TR 则跳过，不作任何处理，减少重复计算量
12. endfor
```

算法 2 描述了挖掘伴随模式的过程。输入移动对象的网格序列 GS 以及区域的范围参数，以及条件阈值 *m*、*dur* 和 *len*，最终输出符合要求的伴随模式结果 AP。假如一群数量大于 *m* 的移动对象群体伴随运动时长超过 *dur*，且这群移动对象伴随运动至少经过 *len* 个网格区域，则这群移动对象构成的时空轨迹模式称为伴随模式。

在挖掘伴随模式的过程中，有一种特殊情况需要单独考虑，即网格序列可能会出现间断。本文提出的算法进行如下处理：首先对出现间断之前得到的候选伴随模式进行判断，符合要求则输出，否则清空候选集，然后从间断处的下一个网格重新执行伴随模式挖掘程序。实验表明，该处理方法对算法结果的正确性和算法的效率都无影响。

为了减少不必要的重复处理，已输出的伴随模式结果中的对象集合都会被标记，当程序遇到被标记的移动对象后，不作任何处理，因为该移动对象的伴随模式已经输出。因此减少了计算量，提高了算法的时间效率。

本文引入“8-近邻”的概念，与网格相邻的 8 个网格即为该网格的“8-近邻”网格。与仅仅考虑单个网格中的对象集合相比，考虑 8 近邻网格中的对象集合将有效提高候选伴随对象集合的完整性，保证伴随模式结果的准确性。

与 CMC 等算法不同的是，本文提出的 MAP-G 算法不仅要求 *m* 个移动对象伴随运动持续时长超过设定阈值 *dur*，同时还要求这些移动对象在伴随运动时间区间内至少穿越 len 个网格区域，此举可以提高结果的准确性，因为 CMC 及其它采用 DBSCAN 算法的伴随模式挖掘算法会将一群移动对象长时间滞留在某个区域时形成的运动模式错误地判断为伴随模式。

4 实验与结果分析

实验运行于单机环境，处理器为 Intel(R) Core i3 2350m @2.3GHz，操作系统为 64 位 Windows7 sp1，Eclipse 版本是 4.2.0，数据库采用 Microsoft SQL Server 2008 R2。

实验使用两组轨迹数据集：第一组轨迹数据集为合成数据，包括 200 条轨迹序列以及随机生成的采样点，共计 5000 个采样点，该组数据集用于验证 MAP-G 算法的正确性，合成轨迹数据集包含两组预设的伴随模式结果；第二组轨迹数据集是 2012 年 11 月 30 日北京 2 万辆出租车 24 小时内采集到的时空轨迹数据，数据集总大小为 1.82GB。

MAP-G 算法基于网格索引实现，因此网格大小的设定直接影响到算法的时间性能。如果网格宽度过大，则会导致每个网格中采样点数量太多，降低了搜索候选伴随模式对象集合的效率；如果网格宽度过小，则可能导致轨迹点序列对应的

网格序列过长，降低了挖掘伴随模式的效率。

图 4 所示为第二组轨迹数据集中部分出租车采样数据映射到地图上的实际位置，可见采样点并未能准确地映射到道路上，说明采样数据的精度并不高。为了准确地挖掘出租车数据中隐含的伴随模式，同时根据相关领域知识以及该数据集的平均采样间隔时间等因素，设定网格的宽度为 200 米，即 MAP-G 算法中网格的宽度参数 WD 值设定为 200。

图 4　采样点映射图

图 5 所示为第一组轨迹数据集中部分采样点，其中包含 3 条轨迹。执行 MAP-G 算法后，图 5 对应的伴随模式如图 6 所示。

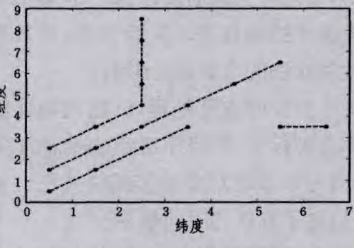

图 5　第一组数据集中的部分采样点

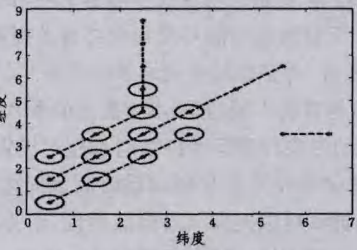

图 6　图 5 所示轨迹的伴随模式

实验表明，MAP-G 算法在第一组轨迹数据集上可以正确挖掘出两组预设的伴随模式结果，从而验证了本文提出的 MAP-G 算法的正确性。

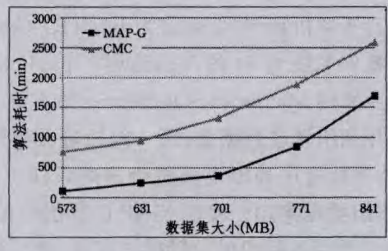

图 7　算法执行时间比较

以第二组轨迹数据作为实验数据，分别执行 MAP-G 算法和 CMC 算法，这两种算法的执行时间如图 7 所示。实验表明 MAP-G 算法比 CMC 算法具有更高的效率。因为 MAP-G 算法采用网格索引提高了伴随模式挖掘的效率，同时利用“8-近邻”网格获取候选伴随模式对象集合，替代了耗时巨大的基于密度的聚类算法，省去了大量重复的计算，并且 MAP-G 算法考虑到了移动对象群体的“滞留”情况，所以其产生的伴随模式挖掘结果更加准确。

结束语　本文提出了一种基于网格索引的时空轨迹伴随模式挖掘算法 MAP-G。该算法利用网格索引提高了挖掘伴随模式候选集合的时间效率。引入了“8-近邻”的概念来挖掘伴随模式候选项集合，避免了耗时的聚类操作，因此 MAP-G 算法具有较高的效率。此外 MAP-G 算法还对移动对象是否一直“滞留”于某区域内进行了判断，使伴随模式挖掘结果更加准确。

参 考 文 献

[1] Benkert M, Gudmundsson J, Hübner F, et al. Reporting flock patterns[J]. Computational Geometry, 2008, 41(3): 111-125

[2] Jeung H, Shen H T, Zhou X. Convoy queries in spatio-temporal databases[C]// 24th International Conference on Data Engineering (ICDE). IEEE, 2008: 1457-1459

[3] Jeung H, Yiu M L, Zhou X, et al. Discovery of convoys in trajectory databases[J]. Proceedings of the VLDB Endowment, 2008, 1(1): 1068-1080

[4] Giannotti F, Nanni M, Pinelli F, et al. Trajectory pattern mining [C]// Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2007: 330-339

[5] Li Z, Ding B, Han J, et al. Swarm: Mining relaxed temporal moving object clusters[J]. Proceedings of the VLDB Endowment, 2010, 3(1/2): 723-734

[6] Laube P, Imfeld S. Analyzing relative motion within groups of trackable moving point objects[M] // Geographic Information Science. Springer Berlin Heidelberg. 2002: 132-144

[7] Kalnis P, Mamoulis N, Bakiras S. On discovering moving clusters in spatio-temporal data[M]// Advances in Spatial and Temporal Databases. Springer Berlin Heidelberg. 2005: 364-381

[8] Jeung H, Yiu M L, Zhou X, et al. Discovery of convoys in trajectory databases[J]. Proceedings of the VLDB Endowment, 2008, 1(1): 1068-1080

[9] Tang L A, Zheng Y, Yuan J, et al. On discovery of traveling companions from streaming trajectories[C]// 28th International Conference on Data Engineering (ICDE). IEEE, 2012: 186-197

[10] Tang L A, Zheng Y, Yuan J, et al. A framework of traveling companion discovery on trajectory data streams [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2013, 5(1): 992-999

[11] Laube P, van Kreveld M, Imfeld S. Finding REMO—detecting relative motion patterns in geospatial lifelines[M] // Developments in Spatial Data Handling. Springer Berlin Heidelberg. 2005: 201-215

[12] Laube P, Imfeld S, Weibel R. Discovering relative motion patterns in groups of moving point objects[J]. International Journal of Geographical Information Science, 2005, 19(6): 639-668