面向不确定移动对象的连续 K 近邻查询算法*

于彦伟[1] 齐建鹏[1] 宋 鹏[1] 张永刚[2]

[1](烟台大学 计算机与控制工程学院 烟台 264005)
[2](吉林大学 符号计算与知识工程教育部重点实验室 长春 130012)

摘 要 近年来,位置服务等领域急需解决的一个难点问题是不确定移动对象连续 K 近邻查询.基于此情况,文中提出高效的面向不确定移动对象的连续 K 近邻查询算法.首先提出 2 种预测移动对象可能区域算法 MaxMin 与 Rate,利用最近一段时间窗口内的位置采样、速度和方向预测移动对象在查询时刻到未来 I 区间可能的位置区域.同时使用最小距离与最大距离区间描述移动对象到查询对象的距离.然后采用优化的基于模糊可能度判定的排序方法查找查询对象的 K 近邻.最后在真实和合成的大规模移动对象数据集上验证文中方法的有效性.

关键词 移动对象,K 近邻查询,可能度判定排序
中图法分类号 TP 391 **DOI** 10.16451/j.cnki.issn1003-6059.201611010
引用格式 于彦伟,齐建鹏,宋 鹏,张永刚.面向不确定移动对象的连续 K 近邻查询算法.模式识别与人工智能,2016,29(11):1048-1056.

Continuous K-Nearest Neighbor Queries for Uncertain Moving Objects

YU Yanwei[1], QI Jianpeng[1], SONG Peng[1], ZHANG Yonggang[2]
[1](*School of Computer and Control Engineering, Yantai University, Yantai* 264005)
[2](*Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun* 130012)

ABSTRACT

An urgent problem in location-based services is continuous K-nearest neighbor (KNN) queries for uncertain moving objects. An efficient algorithm for continuous K-nearest neighbor queries for uncertain moving objects is proposed. Firstly, two solutions, MaxMin and Rate, are proposed to predict the possible location range of the moving object in the time interval by utilizing the sampling points with velocities in the recent time window. A closed interval of minimum and maximum distances is employed to represent the distance between the query object and the moving object. Secondly, an optimized ranking method based on vague possibility decision is proposed to quickly find KNNs of the query object. Finally, experimental results on real and synthetic large-scale datasets demonstrate the effectiveness of the

* 国家自然科学基金项目(No. 61572419,61403328,61302065)、山东省自然科学基金项目(No. ZR2014FQ016,ZR2013FM011)、山东省重点研发计划项目(No. J2015GSF115009)、吉林大学符号计算与知识工程教育部重点实验室开放基金项目(No. 93K172014K13)资助
Supported by National Natural Science Foundation of China (No. 61572419,61403328,61302065), Natural Science Foundation of Shandong Province (No. ZR2014FQ016,ZR2013FM011), Key Program for Research and Development of Shandong Province (No. J2015GSF115009), Open Project of Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education of Jilin University (No. 93K172014K13)
收稿日期:2016-03-02;**修回日期**:2016-05-03;**录用日期**:2016-05-23
Manuscript received March 2, 2016; revised May 3, 2016; accepted May 23, 2016

proposed algorithm.

Key Words　Moving Object, K-Nearest Neighbor Query, Possibility Decision Ranking

Citation　YU Y W, QI J P, SONG P, ZHANG Y G. Continuous K-Nearest Neighbor Queries for Uncertain Moving Objects. Pattern Recognition and Artificial Intelligence, 2016, 29(11): 1048 - 1056.

随着 GPS 终端、Pad、智能手机等位置感知设备的广泛普及,智能导航、兴趣点查询推荐、打车软件等新型信息查询应用的不断涌现,基于位置的服务已成为独具特色的新兴产业,尤其在交通调度与控制、安防监控、位置感知广告服务等领域具有广泛的应用前景,得到越来越多的关注[1]. 时空查询技术是面向移动对象的位置信息服务在实际应用中的重要技术支撑,而 K 近邻查询是时空数据领域广泛应用的重要查询类型之一,受到学术界与工业界长期的关注与研究[2].

Song 等[3] 提出面向移动点的连续 K 近邻查询方法,节省计算成本. Kolahdouzan 等[4] 提出在空间数据库中连续查询 K 近邻兴趣点的方法,即上边界算法(Upper Bound Algorithm, UBA),在查询请求的位置上执行快照 K 近邻查询,通过减少 K 近邻数量评估,提高查询性能.

在动态环境下的面向移动对象不断查询的问题称为连续 K 近邻查询. Mouratidis 等[5] 提出面向移动对象的连续增量监测算法(Incremental Monitoring Algorithm, IMA),以处理在对象发生位置更新时的重复评估查询. 然而 IMA 初始生成扩展树时计算开销较大,当连续查询时间间隔较大时,正确率大幅降低. 孙圣力等[6] 针对 IMA 的不足,提出内结构迭代变更法和数据对象树,弥补 IMA 在数据更新频繁和扩展树生成时的性能缺陷. Huang 等[7] 提出基于路网的移动对象连续查询方法,通过剪枝和精炼两个阶段获取查询点的 K 近邻. 为了提高道路网中增量 K 近邻查询效率,文献[8]使用任务并行方法,文献[9]使用数据并行方法优化连续查询过程. Yu 等[10] 利用网格索引提出基于对象索引和查询索引确定移动对象的 K 近邻. Xiong 等[11] 提出基于网格索引的查询算法,对于新查询并未给出初始结果计算方法,而是重点研究当移动对象位置变化后如何维护查询结果.

大多数研究都需要对移动对象进行相同频率的同步数据采样. 但是,在真实世界系统中,移动对象的运动状态动态变化,位置采集频率一般不一致,在解决传输冲突问题时,一般也不同步,而且还存在数据丢失、数据传输延迟等问题. 此外,用户随时可发起查询请求,经常发生在非采样时刻,这就需要研究不确定的移动对象数据下的 K 近邻查询处理技术.

Huang 等[12] 考虑不确定速度的移动对象连续 K 近邻查询问题,提出有效代价的基于概率的可能 KNN(Probability-Based Possible-KNN, P^2KNN),在给定的查询时间区间内找出每个时刻基于可能度的 K 近邻对象. Li 等[13] 提出基于路网的连续不确定 KNN(Continuous Uncertain KNN, CUKNN),连续查找不定速度对象的 K 近邻问题,使用可能线段描述在一个时间区间内移动对象与查询点之间的可能距离. Fan 等[14] 考虑移动对象的运动状态,利用路网的距离区间模型计算移动对象与查询点的最大、最小距离,然后采用 vague 模糊集计算 K 近邻结果. Sistla 等[15] 考虑位置不确定下的 K 近邻查询问题,提出基于上下界的概率查询方法,但是在考虑移动对象位置不确定区域时仅考虑圆形区域. 上述算法大多是基于道路网络下的查询问题,在道路网络中,对象的移动范围及方向限定较严格,容易预测与判断,而在任意场景下的不确定查询对有效的连续 K 近邻查询处理技术研究更具挑战性.

位置感知设备除定位到准确的位置信息外,大多还能够采集移动对象的运动状态信息. 在多数 K 近邻查询应用中,查询用户大多还期望能预测在未来一小段时间内的查询结果. 因此,本文针对不确定移动对象下的连续 K 近邻查询问题,结合采集到的运动速度大小与方向,使用最近一段时间内位置采样预测移动对象在未来 I 时间区间内的可能区域,采用最大、最小距离区间衡量查询对象与可能区间的距离. 采用优化的基于模糊可能度判定的排序方法查找 K 近邻. 大规模真实与合成数据的综合实验验证本文方法的有效性与查询性能.

1　问题定义

首先给出一些重要的定义和表示,然后给出面向不确定移动对象的连续 K 近邻查询问题.

定义 1 移动对象　一个移动对象由唯一的标识符表示,记为 o_i,o_i 的移动轨迹数据为一个多维时

空数据点序列，每个数据点表示为五元组 (t,x,y,v_x,v_y)，其中，t 为采样时间，x、y 分别为 o_i 在 t 时刻的位置坐标，v_x、v_y 分别为 o_i 在该位置时 x 和 y 上的速度分量.

对于固定采样频率的移动对象，轨迹数据可表示为等时间间隔的序列

$$s_i = \{(1,x_1,y_1,v_{x1},v_{y1}),(2,x_2,y_2,v_{x2},v_{y2}),\cdots,(k,x_k,y_k,v_{xk},v_{yk})\}.$$

$s_{i,j}$ 表示移动目标 o_i 在时刻 j 时的位置，相应地，用 $s_{i,(j,k)}$ 表示 o_i 在时刻 j 到时刻 k 这段时间中采样到的轨迹序列集. 对于采样频率不固定的移动对象，轨迹数据为不确定时间间隔的序列，如

$$s_i = \{(1,x_1,y_1,v_{x1},v_{y1}),(2,x_2,y_2,v_{x2},v_{y2}),(5,x_5,y_5,v_{x5},v_{y5}),\cdots,(k,x_k,y_k,v_{xk},v_{yk})\}.$$

包含 n 个移动对象的集合表示为

$$DB_o = \{o_1,o_2,\cdots,o_n\},$$

轨迹数据集表示为

$$S = \{s_1,s_2,\cdots,s_n\}.$$

由于移动对象位置数据按时间采集并存储，所有移动对象的轨迹数据按时间序列存放与处理.

本文采用时间窗口 W 处理移动对象位置数据，窗口长度标记为 w，这里将最小的时间刻度单位定义为一个时间点，则 $|W| = w$.

给定移动对象集合 DB_o，不确定轨迹数据序列 S，时间窗口 W，当前查询时刻 now 和查询对象 q，$Query(q,I)$ 查询返回 q 在接下来时间区间 I 内最邻近的 K 个移动对象.

2 不确定数据的 K 近邻预测

本文通过采样到的轨迹数据中的速度与时间的变化关系对移动对象进行位置预测.

一般是通过位置数据拟合一条曲线，进而预测 o_i 的位置. 如图 1 所示，时间窗口 W 为从时刻 1 到 now 的时间区间（即 $w = now - 1$）. 通过拟合之前采样到的移动对象 o_i 的位置信息集合 $s_{i,(1,now)}$ 预测未来 $now + I$ 时刻移动对象 o_i 可能的位置.

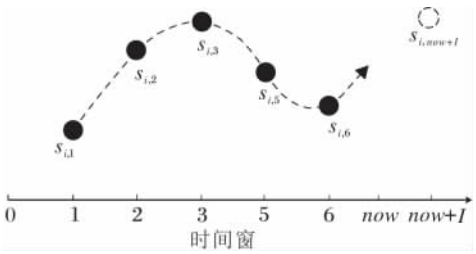


图 1 位置数据拟合预测

Fig. 1 Position prediction using data fitting

当面向海量数据时，数据拟合预测的时间及空间消耗过大，因此设计最大最小速度 MaxMin 及速度变化率 Rate 的预测.

从图 1 中可看到，$s_{i,now+I}$ 受窗口 W 内临近几个采样的影响较大，因此应考虑时间窗口 W 内最近采样到的位置信息. 使用 $\beta \in (0,1)$ 表示采样因子，则 $size = \lceil \beta w \rceil$ 表示使用的采样位置点个数. 令 w_1、w_n 分别表示 o_i 在窗口 W 内第一次和最后一次采样时刻，使用的位置点集合 s_i：

$$s_i = \begin{cases} s_{i,(w_{n-size+1},w_n)}, & |s_{i,(w_1,w_n)}| > size \\ s_{i,W}, & \text{其它} \end{cases}$$

当时间窗口 W 内采集到的 o_i 位置个数 $|s_{i,(w_1,w_n)}|$ 小于 $size$ 时，使用窗口 W 内所有采集的位置.

2.1 基于最大最小速度的位置预测

为了便于描述，给出一个实例示意该预测方式的过程. 假设时间窗口 $W = [t_1,t_{18}]$，即当前时刻 $now = 18$，考虑移动对象 o_i 在 W 内出现的轨迹集合 $s_{i,(1,18)} = \{(5,10,10,10,10),(10,20,30,10,20),(13,40,40,20,10),(16,50,50,10,10)\}$，预测在未来时刻 $now + I = 20(I = 2)$ 时移动对象可能到达的位置 $s_{i,20}$.

基于最大最小速度的预测计算方式是通过寻找移动目标 o_i 在时间窗口 W 中的速度在每个分量上的最大最小值确定，即

$$s_{i,now+I} = \begin{cases} s_{i,w_n} + \max v_i(I + now - t_{w_n}) \\ s_{i,w_n} + \min v_i(I + now - t_{w_n}) \end{cases}$$

$$\max v_i = \max\{v_{i,w_1},v_{i,w_2},\ldots,v_{i,w_n}\}, \quad (1)$$

$$\min v_i = \min\{v_{i,w_1},v_{i,w_2},\ldots,v_{i,w_n}\},$$

$$n > size,\ w_1 = n - size.$$

以 $s_{i,(1,18)}$ 数据为例，水平速度方向上 $\min v_{x,16} = 10$，垂直方向上 $\min v_{y,16} = 10$. 同时可以计算

$$\max v_{x,16} = 20,\ \max v_{y,16} = 20,$$

因此

$$\min v_{x,y} = (10,10),\ \max v_{x,y} = (20,20).$$

代入式(1) 后得到

$$\min s_{i,20} = (20,90,90,10,10),$$

$$\max s_{i,20} = (20,130,130,20,20).$$

以其为顶点表示的区域如图 2 所示，在速度以任何 min-max 范围内变化的运动都会落在该区域内. 可见最大最小方式将会对移动目标 o_i 预测一个范围区域.

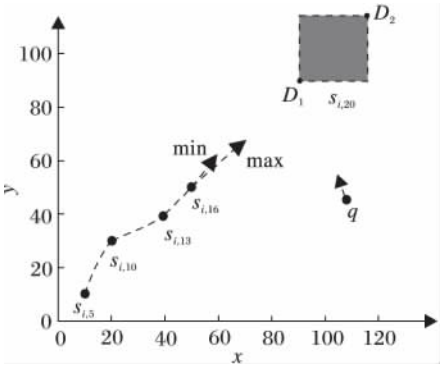


图 2　基于最大最小速度的移动目标位置预测

Fig. 2　Location prediction of moving object based on MaxMin velocity

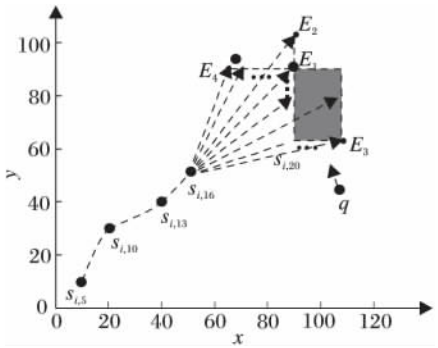


图 3　基于速度变化率的预测

Fig. 3　Location prediction of moving object based on of velocity variation

最大最小速度方法确定的预测范围是由临界值得出,然而实际上物体的运动到达临界值的频率并不高,同时该方法未考虑物体运动与采样时间的关系,且当前速度的影响权重不高,这使物体运动同样不具有连续性,因此当采样频率不定或速度动态变化较大时,该方法预测准确度不高.

2.2　基于速度变化率的位置预测

当移动对象采样频率不定时,需要考虑速度变化与时间的关系. 本文探索通过速率变化,即采样点速度大小与方向随时间变化的关系进行预测:

$$s_{i,now+I} = s_{i,w_n} + v_{i,w_n}(I + now - t_{w_n}) + \frac{1}{2}a(I + now - t_{w_n})^2, \quad (2)$$

$$a_{i,w_k} = \frac{\Delta v}{\Delta t} = \frac{v_{i,w_{k+1}} - v_{i,w_k}}{w_{k+1} - w_k},$$

其中,a 为速度在每个方向分量上的变化率,即 $a = \Delta v/\Delta t$. 在二维空间中,采用向量的形式表示 a,即 a 带有大小和方向.

同样以 $s_{i,(1,18)}$ 为例,首先计算得到

$$a_{i,w_1} = \Delta v_{i,10} = (0,2),$$

$$a_{i,w_2} = \Delta v_{i,13} = (10/3, -10/3),$$

$$a_{i,w_3} = \Delta v_{i,16} = (-10/3, 0).$$

同时考虑到对象可能按照当前的速度继续运动,因此新增一个变化率 $a_i = (0,0)$. 从而根据式(2)依次计算图3所示的 $E_1 \sim E_4$ 各点,考虑移动对象速度在以上变化率范围内变化,o_i 会落在如图 3 所示的阴影区域或虚线上.

由于考虑速度随采样时间间隔的变化和当前速度的权重,基于多速率变化的可能区域比 MaxMin 能够更好地刻画移动对象在未来时刻 $now + I$ 时可能存在的位置.

当给定查询移动 q 时,需要计算查询对象在未来时刻 $now + I$ 与其它移动对象的距离,以便进行 K 近邻查询. 由于最大最小及速度变化率预测刻画的是一个区域,因此,本文采用一个距离区间衡量查询对象 q 与该移动对象的可能距离范围,具体算法实现步骤如下.

算法 1　Query(q, I)

输入　DB_o,S,W,q,K,α

输出　*top K nearest objects*

1 $S_w \leftarrow S(W)$; // locations sequence in W
2 $DB_{o,w} \leftarrow DB_o(W)$; // moving objects in W
3 compute $pos_{q,now+I} = pos_{q,now} + q.v_{now} \cdot I$;
4 FOREACH $o_i \in DB_{o,w}$ DO
5　FOREACH $s_{i,w} \in \{\beta S_w\}$ DO
6　$a_{i,w} \leftarrow \Delta v/\Delta t$;
7　compute possible position pos_i by eq. 2;
8　put pos_i into POS set;
9　FOREACH $pos_j \in POS$ DO
10　$dis_j = distance(pos_j, pos_{q,now+I})$;
11　put dis_j into DIS set;
12　get distance interval $di_i = [d,D]$ by DIS;
13　put di_i to DI;
14 FOREACH $di_j \in DI$ DO
15　compute probability pro_j by possibility vague set;
16　put pro_j into PRO set;

17 select KNN by top-K in PRO.

算法 1 描述在未来时刻 $now + I$ 时,根据速率变化 Rate 搜索查询对象 q 可能的 K 近邻的过程. 首先,提取时间窗口 W 内的数据,记轨迹集合为 $S_{o,w}$、对象集合为 $DB_{o,w}$,如 1、2 行. 对于每个移动对象 o_i,根据采样因子 β,获取窗口 W 中采样到的 $size$ 个位置点,通过式(2)预测 o_i 的可能位置,并存储在 POS 集合中,如 5 ~ 8 行描述. 根据对象 q 和对象 o_i 的可能位置集 POS 计算 q 到 o_i 之间的距离,并使用得出的距离上下界表示成距离区间,存储在 DI 集中. 之后采用第 3 节描述的模糊可能度方法排序移动对象,从而得出前 K 个距离 q 最近的移动对象,如 12 ~ 15 行.

3 基于模糊可能度判定的 K 近邻排序

MaxMin 和 Rate 两种方法预测查询对象到移动对象间可能的距离空间,使用最近距离与最远距离的区间 $[d,D]$ 表示. 要获取到查询对象 q 的 K 近邻,就需要对查询对象到其它移动对象的距离进行排序对比. 对于 2 个距离区间的对比,本文参考文献[15]的 vague 模糊集方法.

对于查询点 q 到 2 个移动对象 o_1、o_2 的距离区间,分别设为 $[d_1,D_1]$ 和 $[d_2,D_2]$. 使用 $o_1 \prec o_2$ 表示查询点 q 到 o_1 的距离更近于到 o_2 的距离,概率 $p(o_1 \prec o_2)$ 表示 q 到 o_1 距离更近于 o_2 的可能度,因此

$$p(o_1 \prec o_2) \in [0,1].$$

使用该概率表示 q 到 o_1 更近的可能性,至少需要表达 3 种状态:1) q 到 o_1 距离近于到 o_2 距离;2) q 到 o_1 距离等于到 o_2 距离;3) q 到 o_1 距离远于到 o_2 距离.

若设定 $p(o_1 \prec o_2) = 0.5$ 表示 q 到 o_1 距离等于到 o_2 距离,$p(o_1 \prec o_2) > 0.5$ 表示 q 到 o_1 距离近于 o_2 的可能度更高,$p(o_1 \prec o_2) < 0.5$ 表示 q 到 o_1 距离近于 o_2 的可能性较低. 根据 vague 模糊集,可由 2 个距离区间的交集部分占两区间总长度的比重衡量 $p(o_1 \prec o_2)$. 文献[15]给出相应计算方法:

$$p(o_1 \prec o_2) = \frac{\max(0, D_1 - d_1 + D_2 - d_2 - \max(0, D_1 - d_2))}{D_1 - d_1 + D_2 - d_2}. \quad (3)$$

图 4 为 2 个距离区间相对位置.

如图 4(a)所示,两距离区间的总长度为 $D_1 - d_1 + D_2 - d_2$,交集部分为 $D_1 - d_2$. 由于在交集部分,两距离区间无差异,该部分可忽略. (b)为交换两距离区间的相对位置,

$$p(o_2 \prec o_1) = \frac{D_1 - d_2}{D_1 - d_1 + D_2 - d_2},$$

反而表示除交集以外长度占总长度的比重,也确保一定大于 0.5.

$$p(o_1 \prec o_2) = \frac{D_2 - d_1}{D_1 - d_1 + D_2 - d_2}$$

表示交集部分占总长度的比重,显然 $p(o_1 \prec o_2)$ 也小于 0.5. 同样,式(3)也适用于图 4 中(c) ~ (e)所示特殊情况.

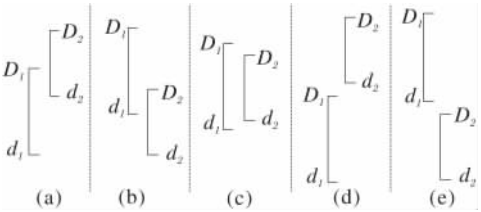


图 4 两个距离区间相对位置示例

Fig. 4 Examples of relative positions of two distance intervals

可以验证该概率计算公式满足上述 4 个要求. 此外,还能得出如下重要性质.

性质 1 $p(o_1 \prec o_2) \geq 0.5$,当且仅当 $D_2 - d_1 \geq D_1 - d_2$.

利用式(3)可计算查询点 q 到对象 o_1 的距离比对象 o_2 的距离更近的可能概率. 对于 m 个移动对象,文献[15]使用

$$P_i = \frac{2}{m^2}\sum_{j=1}^{m} p_{ij}$$

计算对象 o_i 是查询点 q 的一个最近邻居的可能度 P_i,表示为对象 o_i 与其它所有对象之间的概率之和占所有两两对象间概率总和的比重. 通过对所有对象的可能度排序,可找出查询点 q 的 K 近邻.

本节提出利用性质 1 的简单高效的 K 近邻排序方法. 利用性质 1,仅验证 $D_2 - d_1 \geq D_1 - d_2$ 是否成立,便可判断查询点 q 到对象 o_1 是否更近于对象 o_2. 若对象 o_1 与所有其它对象都验证一遍,可确定对象 o_1 是查询点 q 的第几近邻. 根据这种判断,设计基于模糊可能度判定(Possibility Decision)的 K 近邻排序方法.

对于 m 个移动对象,将查询点 q 相对于 m 个对象两两之间的可能度存放在一个 $m \times m$ 的矩阵 $\boldsymbol{R}$ 中. 若 $p(o_i \prec o_j) \geq 0.5$ 成立,$R_{ij} = 0$;否则 $R_{ij} = 1$. 这样 $\sum_{j=1}^{m} R_{ij}$ 为对象 o_i 在查询点 q 的最近邻居排序中

的序号(从 0 开始).

表 1 给出 5 个移动对象的示例,设定到查询点 q 距离远近的排序为

$$o_3 \prec o_2 \prec o_1 = o_4 \prec o_5.$$

根据 $p(o_i \prec o_j) \geqslant 0.5$ 是否成立,填充矩阵,每行计算 1 的个数,即为对象 o_i 的序号. 可以看出,对象 o_1、o_4 到 q 的距离相等,它们的序号也相等,表示并列第 2(从 0 开始). 若设定 $K = 3$,根据最后一列的结果,找出小于 3 的对象结果$\{o_3, o_2, o_1\}$ 或$\{o_3, o_2, o_4\}$.

表 1　基于可能度判定的 K 近邻排序

Table 1　K neighbors ranking based on possibility decision

q	o_1	o_2	o_3	o_4	o_5	排序
o_1	×	1	1	0	0	2
o_2	0	×	1	0	0	1
o_3	0	0	×	0	0	0
o_4	0	1	1	×	0	2
o_5	1	1	1	1	×	4

相比基于 vague 集概率排序方法,本文方法既不需要计算模糊概率,也不需要计算移动对象属于查询点 q 的 K 近邻的可能度,可较大幅度提高 K 近邻的查找效率.

4　实验及结果分析

4.1　实验数据与测试方法

算法由 Java 实现,实验平台配置为 2.2 GHz 至强 E5-2660 处理器,16 GB 内存,Windows Sever 2012 操作系统.

实验采用 2 个数据集:合成数据集,真实数据集. 合成数据集 MOD 由移动对象生成器合成,在 1 000 × 1 000 区域内,生成 5 000 个移动对象在 500 个时间点上的位置数据. 真实数据集 Taxi 来自微软亚洲研究院的 T-Drive 项目[17],包含 10 357 辆出租车在北京市区一周内的 GPS 位置数据,位置采样点数量达 150 000 000,数据大小达 1.5 GB. 为了模拟不确定移动对象数据,从 MOD 数据集中随机删除 20% 的采样点.

本文采用准确率评估方法,

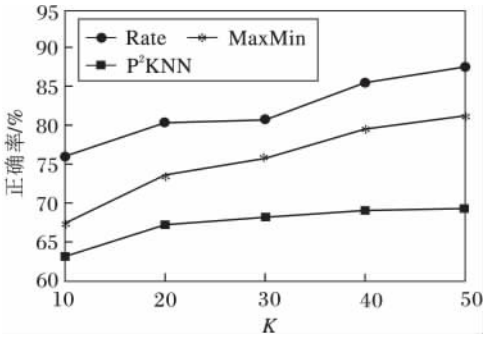
$$Precision = \frac{R \cap D}{R},$$

其中,D 为在查询时刻从完整 MOD 数据中查询的真实 K 近邻结果,R 为本文算法在删除数据后的 MOD 上获得的 K 近邻结果. 对于效率评估,通过变化各重要参数,测量单次查询所执行的 CPU 时间指标以评估算法的性能.

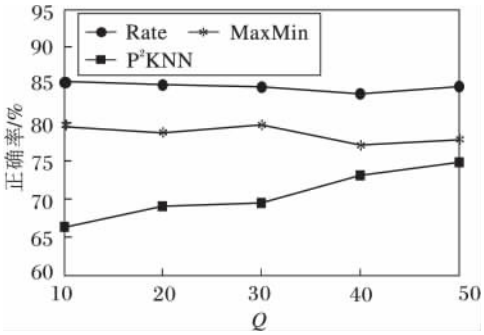
对比本文的 MaxMin、Rate、优化的 K 近邻排序方法及 P^{2}KNN[13]. MaxMin 采用 vague 模糊集排序,采用 vague 模糊集排序的 Rate 记为 PVRate,采用优化排序方法的 Rate 记为 PDRate(在 4.2 节,PVRate 与 PDRate 统称 Rate). 算法实现都采用 PR-tree 索引.

4.2　正确性评估实验

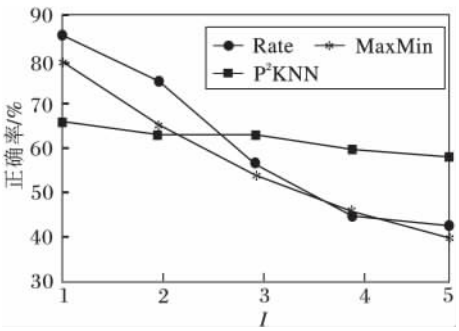
在 MOD 数据集评估 MaxMin 与 Rate 的正确率,默认参数设置如下:$W = 10$,$\beta = 0.4$,$Q = 20$,$I = 1$,$\Delta t = 2$,$K = 40$. MaxMin、Rate 和 P^{2}KNN 随各参数变化的正确率结果如图 5 所示.



(a) K



(b) Q



(c) I

图 5　各算法在不同参数下的正确率结果

Fig. 5　Precision results of algorithms with different parameters

如图 5(a)所示,随着 K 的增加,MaxMin 和 Rate 的正确率都逐渐提升. 这主要是由于当 K 较小时,基数较小,对处于第 K 附近邻居的错误预测对正确率计算影响较大,而随着 K 的增加,该影响逐渐减少. 相比 P^2KNN,MaxMin 和 Rate 的正确率分别提高 12% 和 18%. 同时可以发现,当 $K=40$ 时,算法的正确率趋向于平稳. (b)给出算法正确率随 Q 变化的结果,Rate 正确率平均达到 85%,比 P^2KNN 提高 14%,而 MaxMin 也达到 79% 的正确率. 随着 Q 的增加,算法的正确率并未受到明显影响,说明正确率与查询对象的数量无直接联系,这也符合预期目标. (c)为 I 对算法正确性的影响,随着 I 的增加,算法的正确率逐渐降低,这符合预期结果. 因为移动对象运动动态变化较大,在越远的未来时间,使用最近窗口内的采样位置预测的可能区域与真实位置差异越大,正确率也就越低. 在 $I\leqslant 2$ 时,Rate 正确率比 P^2KNN 平均提高 24%. 然而,P^2KNN 在更远的未来时刻正确率优于 Rate,这是因为 P^2KNN 预测的未来时刻查询点 q 范围较大,在未来很长一段时间里包含的可能性较多,同时也带来可信度的降低. 针对移动对象的位置预测,本文通常关注的是最近几个未来时刻.

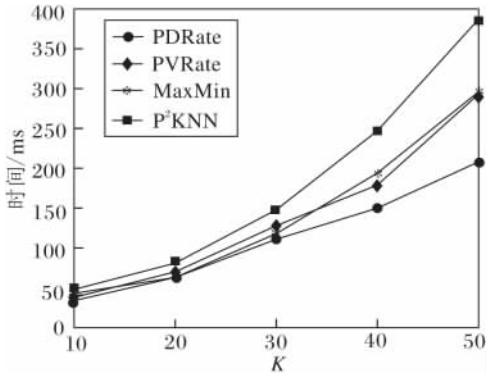
总之,Rate 在大多数情况下达到 85% 的正确率,足以验证 Rate 能够较有效地预测未来一段时间内查询对象的前 K 个近邻.

4.3 性能评估

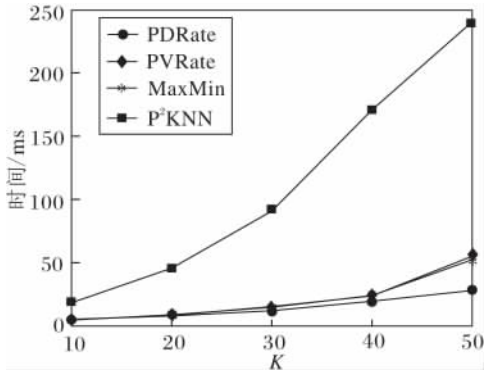
本节在 2 个数据集上对算法性能进行综合评估,保持 MOD 默认参数设置不变,Taxi 上默认参数设置为: $W=10$ min, $I=2$ min, $\Delta t=2$ min. 实验结果如图 6 ~ 图 8 所示.

由图 6 可看出,随着 K 的增加,4 种算法的平均查询时间都随之增加. 这是因为算法都采用 PR-tree 索引移动对象,随着 K 的增加,每次查询都需要检测更多的候选移动对象,增加距离计算数量,模糊可能度排序阶段也相应增加计算成本.

由图 6 还可看出,在 Taxi 上的查询时间慢于 MOD 上的查询时间,这是由于 Taxi 中移动对象数量多于 MOD,在构建 PR-Tree 索引时消耗较多时间. 但是,随着 K 的增加,优化的基于可能度判定的排序方法表现出最好的性能,当 $K\geqslant 40$ 后,相比 P^2KNN,效率平均提高 66%,相比 vague 模糊集排序,PVRate 的查询效率也提升 33%.



(a) MOD

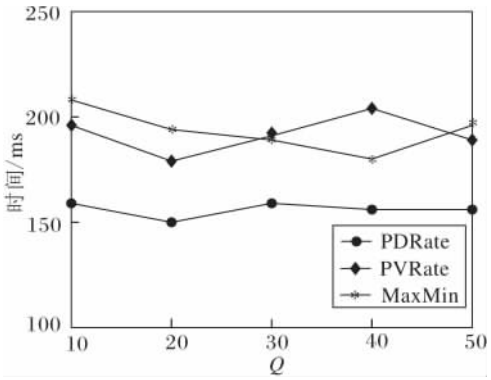


(b) Taxi

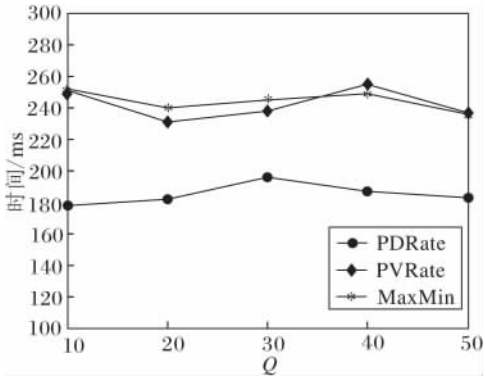
图 6 K 对算法性能的影响

Fig. 6 Influence of K on performance of algorithms

图 7、图 8 给出在 MOD 和 Taxi 上变化 Q 和 I 对 MaxMin 和 Rate 效率的影响. 通过图 7 和图 8 可以看出,当固定 K 时,变化 Q 和 I 对 3 种算法的平均查询时间影响都不大. 这是因为在单次查询中,参数变化对距离计算及排序过程的计算量基本无影响. 同样可以看到,在所有测试中,PDRate 都使用最少的查询时间,相比 PVRate,效率平均提升约 30%.



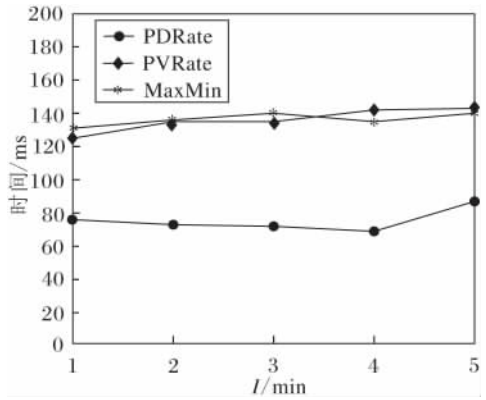
(a) MOD



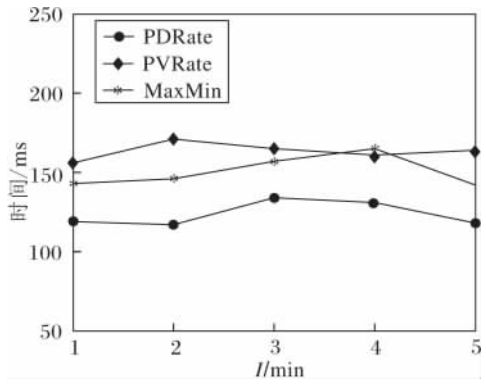
(b) Taxi

图 7　Q 对算法性能的影响

Fig. 7　Influence of Q on performance of algorithms



(a) MOD



(a) Taxi

图 8　I 对算法性能的影响

Fig. 8　Influence of I on performance of algorithms

5　结 束 语

本文实现面向不确定移动对象的连续 K 近邻查询算法. 首先,利用移动对象的运动状态(速度大小与方向)和最近一段时间内位置采样预测未来时间区间 I 内的可能区域,并使用最大、最小距离区间表示查询点与可能区间的距离. 然后,在模糊集理论的基础上设计优化的 K 近邻排序方法. 最后通过实验综合评估算法的有效性. 下一步将对大量查询点高并发过程中计算复用模型展开研究与分析.

参 考 文 献

[1] 周傲英,杨 彬,金澈清,等. 基于位置的服务:架构与进展. 计算机学报, 2011, 34(7): 1155 – 1171.
(ZHOU A Y, YANG B, JIN C Q, *et al*. Location-Based Services: Architecture and Progress. Chinese Journal of Computers, 2011, 34(7): 1155 – 1171.)

[2] TAO Y F, PAPADIAS D, SHEN Q M. Continuous Nearest Neighbor Search // Proc of the 28th International Conference on Very Large Databases. New York, USA: VLDB Endowment, 2002: 287 – 298.

[3] SONG Z X, ROUSSOPOULOS N. *K*-Nearest Neighbour Search for Moving Query Point // Proc of the International Symposium on Advances in Spatial and Temporal Databases. Berlin, Germany: Springer, 2001: 79 – 96.

[4] KOLAHDOUZAN M R, SHAHABI C. Alternative Solutions for Continuous *K* Nearest Neighbor Queries in Spatial Network Databases. GeoInformatica, 2005, 9(4): 321 – 341.

[5] MOURATIDIS K, YIU M L, PAPADIAS D, *et al*. Continuous Nearest Neighbor Monitoring in Road Networks // Proc of the 32nd Very Large Databases Conference. New York, USA: VLDB Endowment, 2006: 43 – 54.

[6] 孙圣力,林 硕. 一个高效的连续 k 近邻查询改进算法. 计算机研究与发展, 2013, 50(Z): 80 – 89.
(SUN S L, LIN S. An Improved Algorithm For Efficient Continuous KNN Queries. Journal of Computer Research and Development, 2013, 50(Z): 80 – 89.)

[7] HUANG Y K, CHEN Z W, LEE C. Continuous *K*-Nearest Neighbor Query over Moving Objects in Road Networks // Proc of the Joint International Conferences on Advances in Data and Web Management. Berlin, Germany: Springer, 2009: 27 – 38.

[8] 赵 亮,陈 荦,景 宁,等. 道路网中的移动对象连续 K 近邻查询. 计算机学报, 2010, 33(8): 1396 – 1404.
(ZHAO L, CHEN L, JING N, *et al*. Continuous *K* Nearest Neighbor Queries of Moving Objects in Road Networks. Chinese Journal of Computers, 2010, 33(8): 1396 – 1404.)

[9] 赵 亮,景 宁,陈 荦,等. 面向多核多线程的移动对象连续 K 近邻查询. 软件学报, 2011, 22(8): 1805 – 1815.
(ZHAO L, JING N, CHEN L, *et al*. Continuous *K* Nearest Neighbor Queries over Moving Objects Based on Multi-core and Multi-threading. Journal of Software, 2011, 22(8): 1805 – 1815.)

[10] YU X H, PU K Q, KOUDAS N. Monitoring *k*-Nearest Neighbor Queries over Moving Objects // Proc of the 21st International Con-

ference on Data Engineering. Washington, USA: IEEE, 2005: 631 – 642.

[11] XIONG X P, MOKBEL M F, AREF W G. SEA-CNN: Scalable Processing of Continuous *K*-Nearest Neighbor Queries in Spatio-Temporal Databases // Proc of the 21st International Conference on Data Engineering. Washington, USA: IEEE, 2005: 643 – 654.

[12] HUANG Y K, CHEN C C, LEE C. Continuous *K*-Nearest Neighbor Query for Moving Objects with Uncertain Velocity. GeoInformatica, 2009, 13(1): 1 – 25.

[13] LI G H, LI Y H, SHU L C, *et al*. CkNN Query Processing over Moving Objects with Uncertain Speeds in Road Networks // Proc of the 13th Asia-Pacific Web Conference on Web Technologies and Applications. Berlin, Germany: Springer, 2011: 65 – 76.

[14] FAN P, LI G H, YUAN L, *et al*. Vague Continuous *K*-Nearest Neighbor Queries over Moving Objects with Uncertain Velocity in Road Networks. Information Systems, 2012, 37(1): 13 – 32.

[15] SISTLA A P, WOLFSON O, XU B. Continuous Nearest-Neighbor Queries with Location Uncertainty. The VLDB Journal, 2015, 24(1): 25 – 50.

[16] ARGE L, DE BERG M, HAVERKORT H J, *et al*. The Priority R-Tree: A Practically Efficient and Worst-Case Optimal R-tree. ACM Trans on Algorithms, 2008, 4(1). DOI: 10.1145/1328911.1328920.

[17] YUAN J, ZHENG Y, XIE X, *et al*. T-Drive: Enhancing Driving Directions with Taxi Drivers´ Intelligence. IEEE Trans on Knowledge and Data Engineering, 2013, 25(1): 220 – 232.

作者简介

于彦伟(通讯作者),男,1986 **年生,博士,讲师,主要研究方向为数据挖掘、分布式计算**. E-mail:yuyanwei@ ytu. edu. cn.

(**YU Yanwei**(Corresponding author), born in 1986, Ph. D., lecturer. His research interests include data mining and distributed computing.)

齐建鹏,男,1992 **年生,硕士研究生,主要研究方向为数据挖掘**. E-mail:jianpengqi@ 126. com.

(**QI Jianpeng**, born in 1992, master student. His research interests include data mining.)

宋 鹏,男,1983 **年生,博士,讲师,主要研究方向为机器学习、数据挖掘**. E-mail:pengsongseu@ gmail. com.

(**SONG Peng**, born in 1983, Ph. D., lecturer. His research interests include machine learning and data mining.)

张永刚,男,1974 **年生,博士,副教授,主要研究方向为数据挖掘、知识工程**. E-mail:zhangyg@ jlu. edu. cn.

(**ZHANG Yonggang**, born in 1974, Ph. D., associate professor. His research interests include data mining and knowledge engineering.)