ISU-Tree：一种支持概率 k 近邻查询的不确定高维索引

庄 毅
（浙江工商大学计算机与信息工程学院 杭州 310018）

摘 要 文中提出一种支持概率 k 近邻查询的不确定高维索引结构——ISU-Tree. 在高维空间，首先对 n 个不确定数据对象进行 k 平均聚类，然后分别对每个不确定超球进行初始“切片”，并对其进行多特征编码得到对应的统一化索引键值，并且用 B^+ 树建立索引. 这样，高维空间的概率查询就转变成对一维空间的启发式的范围查询及求精运算. 理论及实验分析表明 ISU-Tree 索引能更有效地缩小搜索空间，减少积分计算的代价. 在查询效率方面要明显优于其它的索引方法，尤其适合海量高维不确定数据的概率查询.

关键词 初始距离；概率 k 近邻查询；不确定超球；初始片；概率密度函数
中图法分类号 TP301 **DOI 号**：10.3724/SP.J.1016.2010.01934

ISU-Tree: A Uncertain High-Dimensional Indexing Algorithm for Probabilistic k Nearest Neighbor Query

ZHUANG Yi
(*School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou* 310018)

Abstract This paper proposes an uncertain high-dimensional indexing algorithm called ISU-Tree to facilitate probabilistic k nearest neighbor query. In the high-dimensional spaces, all (n) uncertain data objects are first grouped into some clusters by a k-Means clustering algorithm. Then each object's corresponding uncertain sphere is "sliced" in terms of the initial-distance. Finally a unified key of each data object is computed by adopting multi-attribute encoding scheme, which is inserted by a B^+-tree. Thus, given a query object, its probabilistic range search in high-dimensional spaces is transformed into the search in the single dimensional space with the aid of the ISU-Tree. Theoretical and experimental analysis both indicate the effectiveness and efficiency of the proposed scheme.

Keywords initial distance; probabilistic k nearest neighbor query; uncertain sphere; initial-slice; probabilistic density function

1 引 言

无线传感器网络[1]和 RFID(Radio Frequency Identification)[2]等技术在环境、水资源监测、移动查询和军事监控等领域有着重要的应用前景. 这些应用时刻会产生大量采样数据，由于采用的数据模型不同，这些数据呈现 4 种特性：海量(large-scale)、异构(heterogeneous)、高维(high-dimensional)和不确定(uncertain). 目前，不确定数据，特别是高维不确定数据的查询处理正受到越来越多的关注[3]，已成为国内外学术界关注的一个重要研究课题，具有

收稿日期：2010-08-22. 本课题得到国家自然科学基金(60003047，60533090，60903053)、浙江省自然科学基金(Z1100822，Y1080148，Y1090165)、浙江省科技厅重大科技项目(2008C13082)、浙江工商大学青年人才基金(Q09-07)资助.庄 毅，男，1978 年生，博士，副教授，主要研究方向为不确定数据库、多媒体数据库和云计算等. E-mail：zhuang@mail.zjgsu.edu.cn.

较强的理论研究价值及现实应用前景.

与传统高维数据相似查询[4](similarity query)不同，高维不确定数据的概率查询(probabilistic query)[3]研究将概率引入到高维数据模型中来衡量不确定对象成为结果集中元素的可能性，即每个不确定对象可表示为具有一定概率密度函数(probability density function)的记录. 因此对高维不确定数据采用传统多维索引方法(如 R-Tree[5]、VA-File[6]等)难以对其进行有效处理，往往会导致查询结果出现偏差. 同时，由于其概率查询存在大量积分运算，因此处理代价非常之高[3,7]. 而且随着数据量的增长，其查询效率往往并不理想.

面对高维概率查询中的高计算代价的挑战，本文提出一种基于初始分片的不确定高维索引方法——ISU-Tree(Initial-Slice-based Uncertain High-Dimensional Indexing Tree)，以支持高效的概率范围查询. 该方法通过对每个不确定超球进行基于初始距离的“切片”，再将得到的初始片进行连续邻接组合，将分片组合编码与每个分片对应的存在概率表达成一个统一的索引键值，将高维空间的范围概率查询转化成一维空间的基于启发式的范围查询及求精运算. 与传统查询方法相比，如顺序检索、U-Tree[8]，ISU-Tree 能显著缩小搜索空间，从而更有效地快速过滤掉不相关的对象. 理论和实验分析表明该方法能有效提高查询效率，尤其适合海量不确定高维数据的概率查询.

2　相关工作

与确定性高维索引[4-6,9-11]不同的是，不确定高维索引所针对的数据对象是随着时间推移而变化的、不确定的，如移动对象的运动轨迹[10]、传感器采集得到的数据[1]等. 由于不确定高维概率查询计算代价非常之大. 为了加快其检索效率，需要对其建立索引机制. Cheng[3]等人较早提出了一种基于 R-tree 的最近邻概率查询(PNN)的索引方法. 文献[12]提出另一种加速执行 PNN 查询的方法，其中每个对象表示为一组从该对象对应的连续的概率密度函数采样得到的点构成. 最近，文献[13-14]分别提出采用一个对象存在于数据库中的概率(称为存在概率)来推导出下界和上界，从而对求得其对应的最近邻对象进行有效的“裁剪”. 另外，U-Tree[8]作为一种多维不确定索引，其原理是在概率范围查找中将不满足条件的数据事先过滤，但该方法对高维概率查询效果不十分理想. 为了提高 PNN 查询的判定概率的计算，Cheng[15]等人又提出 PNN 查询的变种，但不太适合 k 近邻概率查询(k-PNN)(其中 $k\geq 1$). 针对不确定数据的 k 近邻概率查询(k-NN)，Soliman[12]等人提出一种新型的查询类型，它会对每个对象作为查询对象 q 的最近邻的概率进行排序，返回出现概率最高的 k 个对象. 在文献[16]中，Ljosa 等人提出一种加速 k-NN 查询的高效的索引结构——APLA-tree. 最近几年国内外相继开发出一些不确定数据管理系统，如美国斯坦福大学的 Trio 系统[17]、华盛顿大学的 MystiQ 项目①、普渡大学的 Orion 项目②和牛津大学的 MayBMS③等. 国内对不确定数据的研究尚处于起步阶段，文献[18]提出移动环境的不确定移动对象索引方法. 谷峪[2]等人提出了在移动阅读器上的基于 RFID 的概率查询算法.

3　ISU-Tree 索引结构

3.1　预备工作

表 1 首先给出本文将要用到的符号.

表 1　常用符号

符号	意义
Ω	高维不确定数据库且 $\Omega=\{U_1,U_2,\cdots,U_n\}$
n	不确定对象个数
U_i	第 i 个不确定对象
pdf	概率密度函数
$Prob$	概率
$Vol(\cdot)$	$\cdot$ 的体积
$d(U_i,U_j)$	相似距离
$\Theta(q,r)$	查询超球，其中 q 为查询对象，r 为查询半径

定义 1(高维不确定对象).　高维不确定对象 U_i 是一个 D 维空间中的数据点，并且满足以下两个条件：(1) U_i 存在一个概率密度函数(pdf_i)；(2) U_i 对应一个不确定区域，即其在该不确定区域内活动，出现概率满足 pdf_i，其中 $U_i\in\Omega$ 且 $i\in[1,n]$.

定义 2(高维不确定区域).　给定高维不确定对象 U_i，其对应不确定区域可表示为以 U_i 为球心，ε 为半径的超球，记作 $UR(V_i)=\Theta(U_i,\varepsilon)$，其中 ε 是对应不确定区域的活动半径且 $U_i\in\Omega$ 同时 U_i 在 $\Theta(U_i,\varepsilon)$ 里的出现满足一个概率密度函数(pdf_i).

① http://www.cs.washington.edu/homes/suciu/project-mystiq.html
② http://www.cs.purdue.edu/probdb/
③ http://www.comlab.ox.ac.uk/projects/MayBMS/

根据定义 2,在图 1 中,不确定对象 V_i 会在阴影区域内按照一定概率密度函数(pdf_i)分布出现. 简单起见,假设高维空间任意对象 U_i 在虚线圆区域的出现概率满足均匀分布,则其概率密度函数为 $pdf_i=1/Vol(\Theta(U_i,\varepsilon))$.

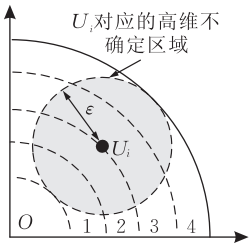


图 1 $\Theta(U_i,\varepsilon)$对应的初始片举例

定义 3(高维不确定数据库). 高维不确定数据库(Ω)由 n 个高维不确定对象 U_i 构成,记作 $\Omega=\{U_1,U_2,\cdots,U_n\}$且 $U_i\in\Omega$.

图 2 中用虚线圆表示的为 4 个不确定对象(如 V_1,V_2,V_3和 V_4)对应的不确定区域.

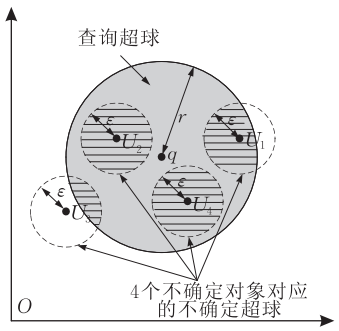


图 2 基于阈值 T 的概率范围查询

定义 4(高维 k 近邻概率查询). 给定查询对象 q、阈值 T 和 k,其 k 近邻概率查询返回 k 个对象 U_i,使得其出现在以 q 为中心 r 为半径的区域中的概率大于 T,记作 $Prob(U_i$出现在$\Theta(q,r)$中$)>T$,其中 $U_i\in\Omega$ 且 r 为虚拟半径.

假设 U_i 在不确定区域($\Theta(U_i,\varepsilon)$)出现概率满足均匀分布,即其概率密度函数为 $pdf_i=1/Vol(\Theta(U_i,\varepsilon))$,则出现概率表示为

$$\begin{aligned}Prob(U_i\text{落入}\Theta(q,r))&=\int_{\Theta(U_i,\varepsilon)\cap\Theta(q,r)}pdf_i\cdot \mathrm{d}U_i\\&=\int_{\Theta(U_i,\varepsilon)\cap\Theta(q,r)}\frac{1}{Vol(\Theta(U_i,\varepsilon))}\cdot \mathrm{d}U_i\\&=\frac{Vol(\Theta(U_i,\varepsilon)\bigcap\Theta(q,r))}{Vol(\Theta(U_i,\varepsilon))}\end{aligned}\tag{1}$$

根据式(1),如图 2 所示,其出现概率可表示为 U_i 出现在不确定超球 $\Theta(U_i,\varepsilon)$ 与查询超球 $\Theta(q,r)$ 相交部分(用栅格表示)的概率.

3.2 基于初始片的不确定超球编码

由于概率查询涉及大量积分运算,为了提高查询效率,首先提出一种基于初始距离的分片编码方式. 通过预先计算不同初始片的概率,使得尽可能减少查询中的概率计算量.

定义 5(初始距离). 给定不确定对象 U_i,其初始距离(记为 Initial-Distance,ID)表示为它到原点 O 的距离,记作 $ID(V_i)=d(U_i,O)$,其中 $O=\{0,0,\cdots,0\}$.

由于 U_i 可表示为一高维向量,因此预先对 n 个对象 U_i 通过 K 平均聚类得到 M 个类. 对于任意一个类 C_j,其中 $j\in[1,M]$,该类中对象的个数记为 $|C_j|$ 且满足 $\sum_{j=1}^{M}|C_j|=n$.

定义 6(类半径). 给定任意类 C_j,其质心 O_j 与该类中距离其最远的对象的距离,称为类半径,记作 CR_j,其中 $j\in[1,M]$.

定义 7(类超球). 给定任意一个类 C_j 和类半径 CR_j,类超球表示为 $\Theta(O_j,CR_j)$.

定义 8(初始片,Initial-Slice). 对于任意不确定超球 $\Theta(U_i,\varepsilon)$,按照初始距离将其均匀切分成 τ 片,将该超球中的第 λ 个初始片表示为 $IS(\lambda,i)$,其中 $\lambda\in[1,\tau]$且 $i\in[1,n]$.

由于可以用不确定区域 $\Theta(U_i,\varepsilon)$ 中的随机对象 X_t 来模拟 U_i 在不确定区域中的出现且 $t\in[1,n_1]$,因此对于随机对象 $X_t\in\Theta(V_i,\varepsilon)$,其对应初始片的编号随着其初始距离的增加而增加. 如图 1 所示,包含 4 个初始片,编号从 1～4.

对于每个不确定超球 $\Theta(U_i,\varepsilon)$ 来说,将其均匀切分成 τ 个初始片,则该不确定超球中的任意一随机对象 X_t 可以用三元组表示:

$$X_t::=\langle t,Cid,IS\_id\rangle\tag{2}$$

其中 Cid 表示 X_t 所在类的编号,IS_id 表示 X_t 所在的初始片的编号且

$$IS\_id(X_i)=1+\left\lceil\frac{ID(X_i)-ID(U_i)+\varepsilon}{\varepsilon/\tau}\right\rceil.$$

一般来说,对于 τ 个分片,其连续邻接编码的组合共有 $\tau(1+\tau)/2$ 个. 由于查询超球与不确定超球相交部分的初始分片是连续的,假设其相交部分对应的初始分片范围为$[LB,UB]$,则其对应的组合编码为

$$Code(LB,UB)=LB^3+UB^3\tag{3}$$

表 2 为不确定超球 $\Theta(U_i,\varepsilon)$的编码举例. 如表 2 所示，将不确定超球分成 4 个初始片，共有 10 组连续邻接组合. 基于以上的编码规则，得到高维数据的编码算法. 如图 3 所示.

表 2　初始片编码举例

ID_id	连续邻接组合	*LB*	*UB*	编码值(*Code*)
1	{1}	1	1	2
2	{1,2}	1	2	9
3	{1,2,3}	1	3	28
4	{1,2,3,4}	1	4	65
	{2}	2	2	16
	{2,3}	2	3	35
	{2,3,4}	2	4	72
	{3}	3	3	54
	{3,4}	3	4	91
	{4}	4	4	128

基于初始片的编码算法.
输入：Ω：高维不确定对象集
输出：H (1 to τ)：τ 个初始片对应的编码表示
1. for $U_i\in\Omega$ and do
2. for $X_t\in\Theta(U_i,\varepsilon)$ do
3. 计算 X_t的初始距离；
4. 由公式(3)获得 X_t的编码值；

图 3　初始分片编码算法

3.3　索引键值表达

在分片编码基础之上，为了能有效地将不确定区域(超球)中分片的统一编号($Code(LB,UB)$)与存在概率值($Prob(U_i)$)结合起来组成一个有效的索引键值，提出一种索引键值的统一表达方法. 该方法将 $Code(LB,UB)$与 $Prob(U_i)$通过线性组合表达成一个统一的索引键值，如下所示：

$$key(U_i)=Code(LB,UB)+Prob(U_i) \quad (4)$$

其中 $Code(LB,UB)$为大于 1 的整数. $Prob(U_i)$表示为随机点 X_i落入从第 LB 个初始片到第 UB 个初始片的不确定区域的概率且 $Prob(U_i)<1$.

为了得到 $Prob(U_i)$，假设在该区域随机产生 n_1 个点(X_t)，则当前的$Prob(U_i)$可表示为$\sum_{t=1}^{n_1}pdf(X_t)$且 $t\in[1,n_1]$.

式(4)不包含任何对象及其对应类信息，为了将这些信息包含其中，将上式改写为式(5)所示：

$$\begin{aligned}key(U_i)&=c_1\times Cid+c_2\times i+\\&\quad Code(LB,UB)+Prob(U_i)\\&=c_1\times Cid+c_2\times i+LB^3+\\&\quad UB^3+\sum_{t=1}^{n_1}pdf(X_t)\end{aligned} \quad (5)$$

其中常数 c_1和 c_2分别是两个较大的整数，使得每个类中的对象对应的键值进行进一步线性放大，使其值域不重叠，其中 $c_1\gg c_2$. ISU-Tree 索引的创建步骤如图 4 所示.

索引生成算法.
输入：Ω：高维数据库
输出：bt：ISU-Tree 不确定高维索引
1. cluster n high-dimensional uncertain objects U_i by using K-Means algorithm
2. for each uncertain sphere $\Theta(U_i,\varepsilon)$ in M clusters
3. τ initial-slices are equally divided;
4. the encoding values of $\tau(1+\tau)/2$ combination of slices are obtained by 基于初始片的编码算法;
5. The unified index key of such $\tau(1+\tau)/2$ combination of slices are obtained by Eq. (5), which are inserted by B^+ tree;
6. return ISU-Tree bt;

图 4　ISU-Tree 索引生成算法

4　不确定概率范围查询算法——PkNNQ

本节提出一种基于 ISU-Tree 索引的高维不确定概率 k 近邻查询算法——PkNNQ. 不失一般性，首先假设查询超球 $\Theta(q,r)$与不确定超球 $\Theta(U_i,\varepsilon)$相交，研究该超球 $\Theta(U_i,\varepsilon)$中的哪些分片与 $\Theta(q,r)$相交.

定义 9(初始片下界编号，Low Bound Id of Initial-Slice).　给定两个相交的超球 $\Theta(q,r)$和 $\Theta(U_i,\varepsilon)$，$\Theta(U_i,\varepsilon)$对应初始片的下界编号是指离 O 最近的初始片编号，表示为 $LB(i)$.

定义 10(初始片上界编号，Upper Bound Id of Initial-Slice).　给定两个相交的超球 $\Theta(q,r)$和 $\Theta(U_i,\varepsilon)$，$\Theta(U_i,\varepsilon)$对应初始片的上界编号是指离 O 最远的初始片编号，表示为 $UB(i)$.

在图 5 中，不确定超球 $\Theta(U_i,\varepsilon)$被切分成 4 个初始片. 与 $\Theta(q,r)$相交的初始片共 2 个，是第 3 和第 4 个初始片，表示为 $LB(i)=3$，$UB(i)=4$.

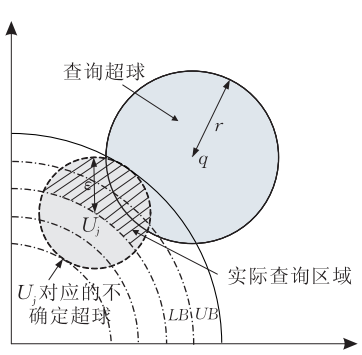


图 5　基于初始片的 $\Theta(q,r)$对应的搜索区域

不失一般性，假设不确定超球 $\Theta(U_i,\varepsilon)$与查询超球 $\Theta(q,r)$相交，且其被切分成 τ 个初始片，则两

球相交区域所包含的初始片一定是连续的(如从第 $LB(i)$ 个初始片到第 $UB(i)$ 个初始片,其中 $LB(i)\leq UB(i)$. 该上下界值如式(6)、(7)所示:

$$LB(i)=\begin{cases}\left\lceil\dfrac{ID(q)-r-ID(U_i)+\varepsilon}{2\varepsilon/\tau}\right\rceil+1, \\ \qquad ID(U_i)-\varepsilon<ID(q)-r<ID(U_i)+\varepsilon \\ 1, \quad ID(q)-r\leq ID(U_i)-\varepsilon\end{cases} \tag{6}$$

$$UB(i)=\begin{cases}\left\lceil\dfrac{ID(q)+r-ID(U_i)+\varepsilon}{2\varepsilon/\tau}\right\rceil+1, \\ \qquad ID(q)+r<ID(U_i)+\varepsilon \\ \tau, \quad ID(q)+r\geq ID(U_i)+\varepsilon\end{cases} \tag{7}$$

其中 $\lceil\cdot\rceil$ 表示 $\cdot$ 的整数部分.

由于 PkNNQ 查询通过依次扩大查询半径执行范围概率查询来得到最终查询结果. 因此先讨论范围概率查询. 如图 6 所示,查询前,由于已经对 n 个不确定对象进行聚类,得到 M 个类超球. 因此可以先判断查询超球与这 M 个类超球是否相交. 如不相交,则可以很快排除这些类中包含的不确定对象. 否则,将每个与查询超球相交的类超球中不确定超球与查询超球判断是否相交,对相交的不确定对象,通过 ISU-Tree 索引快速进行得到其相交部分的近似概率. 具体如图 6 所示,由于相交部分小于或等于从第 LB 到第 UB 初始片的部分,所以概率表示为

$$Prob=\int_{\Theta(U_i,\varepsilon)\cap\Theta(q,r)} pdf_i\cdot \mathrm{d}U_i\simeq\sum_{i=1}^{n_1} pdf(X_i) \tag{8}$$

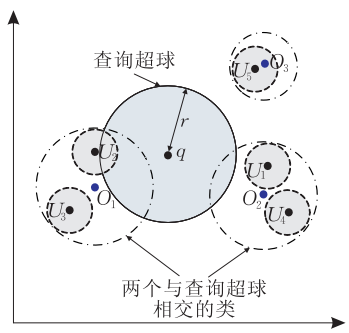


图 6 $\Theta(q,r)$的概率范围查询

对任意不确定超球 $\Theta(U_i,\varepsilon)$,当 $\sum_{i=1}^{n_1} pdf(X_i)<T$ 时,则放弃概率计算. 否则进一步进行求精计算(refinement),即精确求得出现概率. 假设查询超球与不确定超球相交部分的初始分片为 $[LB,UB]$,则其对应的编码为 LB^3+UB^3. 又因为出现概率取值范围为 $[0,1]$,所以求索引键值的取值范围为 $[c_1\times Cid+c_2\times i+LB^3+UB^3,\ c_1\times Cid+c_2\times i+LB^3+UB^3+1]$. 图 7 为以 q 为中心和 T 为阈值的 k 近邻概率查询函数,其中 $|S|$ 表示查询结果的个数;函数 $PRQ(q,r,T)$ 为范围概率查询;$Refinement(q,r,U_i)$ 用于对候选对象精确求得其出现概率;$Farthest(S,q)$ 用于返回结果集中距离 q 最远的对象 $Ufar$;$BRSearch(left,right,j)$ 用于对第 j 个子索引进行标准的范围查询.

```
查询算法. PkNNQ(q,r,T,k).
输入:查询对象 q,r,T
输出:查询结果 S
1.  S←Φ, S1←Φ, r←0; /*初始化*/
2.  while (|S|<k) /*当|S|<k,继续循环*/
3.      r←r+Δr;
4.      S←PRQ(q,r,T);
5.      if (|S|>k) then
6.          for count := 1 to |S|−k−1
7.          U_far←Farthest(S,q);
8.          S←S−U_far; /*从 S 中删除 U_far*/

PRQ(q,r,T)
9.  S←∅, S1←∅;  /*初始化*/
10. for each cluster sphere Θ(O_j,CR_j) do
11.   if Θ(O_j,CR_j) dose not intersect with Θ(q,r) then
12.     break;
13.   else if Θ(O_j,CR_j) is contained by Θ(q,r) then
14.     return the all objects in Θ(O_j,CR_j) to S1;
15.     break;
16.   else
17.     for each uncertain sphere Θ(U_i,ε)∈Θ(O_j,CR_j) do
18.       if Θ(U_i,ε) intersects with Θ(q,r) then
19.         S1←GetProb(q,r,Cid);
20.         if S1<T then end loop
21.         else Refinement(q,r,U_i);
22.   S←S∪S_1;
23. return S;

GetProb(q,r,j)
24. LB(j),UB(j)由公式(6)、(7)获得;
25. left←c_1×Cid+c_2×j+LB^3+UB^3;
26. right←c_1×Cid+c_2×j+LB^3+UB^3+1;
27. S3←BRSearch[left,right,j];
28. return S3;

Refinement(q,r,U_i)
29. prob=0;
30. for each X_j∈Θ(q,r) and X_j∈Θ(U_i,ε) do
31.     prob=prob+Σ_{j=1}^{n_1} pdf(X_j)
32. return prob
```

图 7 基于阈值 T 的概率 kNN 查询算法

5 性能分析

假设 f 表示 B$^+$ 树中每个节点的平均出度,T_S 为磁盘寻道时间,T_L 为延迟时间,T_T 为数据传输时间,T_{TOTAL} 为范围查询时间.

假设聚类个数为 K 个,其中 k 个与查询超球相交,同时由于 B$^+$ 树为 ISU-Tree 的基本索引结构,因此该 B$^+$ 树高度 h、每个节点的平均出度 f 和元素

个数 $n\tau(1+\tau)/2$ 近似满足式(9)：

$$f\times(f+1)^{h-1}=\frac{n\tau(1+\tau)}{2} \tag{9}$$

求解式(9)，得到该树的高度为

$$h=\left\lceil\frac{\lg n+\lg\tau+\lg(1+\tau)-\lg 2-\lg f}{\lg(f+1)}\right\rceil+1 \tag{10}$$

假设查询与 n_1 个不确定超球相交，对于概率范围查询，整个查询分为两部分：首先是从根节点到叶节点，共访问 h 个节点；其次为在叶节点上的范围查询. 由于总共需要进行 n_1 次范围查询，因此其总查询代价近似为

$$T_{\mathrm{FIL}}=n_1\times\left(\left\lceil\frac{\lg n+\lg\tau+\lg(1+\tau)-\lg 2-\lg f}{\lg(f+1)}\right\rceil+1+\left\lceil\frac{1}{f}\right\rceil\right)\times(T_{\mathrm{S}}+T_{\mathrm{L}}+T_{\mathrm{T}}) \tag{11}$$

对候选对象集的求精运算距离运算代价满足式(12)：

$$T_{\mathrm{REF}}=\sum_{j=1}^{n_1}num(j)\times T_{\mathrm{C}} \tag{12}$$

其中 $num(j)$ 为第 j 个不确定超球需要计算随机对象 X_i 出现的个数且 T_{C} 为 CPU 执行一个计算操作的时间.

又因为整个范围查询的代价可以表示为

$$T_{\mathrm{TOTAL}}=T_{\mathrm{FIL}}+T_{\mathrm{REF}} \tag{13}$$

合并式(9)～式(13)得到式(14)：

$$T_{\mathrm{TOTAL}}=n_1\times\left(\left\lceil\frac{\lg n+\lg\tau+\lg(1+\tau)-\lg 2-\lg f}{\lg(f+1)}\right\rceil+1+\left\lceil\frac{num(j)}{f}\right\rceil\right)\times(T_{\mathrm{S}}+T_{\mathrm{L}}+T_{\mathrm{T}})+\sum_{j=1}^{n_1}num(j)\times T_{\mathrm{C}} \tag{14}$$

该查询代价正比于数据对象个数且反比与索引平均出度.

6　实　验

6.1　实验准备

本节通过实验来验证该算法的有效性，同时与其它索引，如 U-Tree 和顺序检索作比较. 我们用 C 语言实现了基于初始片的不确定高维索引——ISU-Tree，同时实现了 U-Tree 等高维索引算法. 采用 B^+ 树作为单维索引结构. 所有实验的运行环境为 Pentium IV CPU 2.0GHz，2GB 内存，硬盘大小为 500G 且 7200 转/分，同时索引页大小设为 4096Bytes.

实验中的测试不确定高维数据分为两类：(1) 对 UCI 提供的 68040 个 32 维的颜色直方图数据[9] 进行改进，在每一维中加入“噪声”信息，使得满足正态分布；(2) 计算机随机产生的 100000 个 64 维的均匀分布的合成数据，其中每一维值的范围也在 0 和 1 之间且每个数据对象对应一个不确定区域，其中不确定半径 $\varepsilon=0.35$. 以下实验分别将索引磁盘块访问数及 CPU 运算开销作为衡量查询性能的两个指标.

6.2　聚类数对查询的影响

首先研究聚类个数 M 对 PkNNQ 查询性能的影响. 从图 8(a)和(b)看出，随着 M 的增加，查询效率(包括 I/O 代价和 CPU 代价)开始是缓慢减少. 因为聚类数增加会导致平均搜索区域减少，但减少的幅度是缓慢的. 当 M 超过一定数目(60)时，会使得各个类超球相互重叠导致查询的 I/O 和 CPU 代价提高. 因此可以将 M 作为一个查询性能优化的调整因子. 因此在下面的实验中将 M 设为 60.

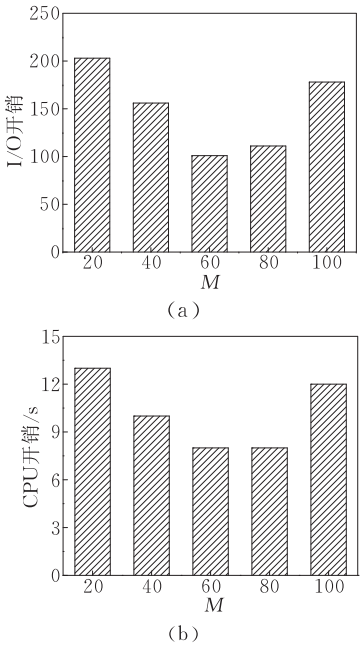


图 8　聚类数对查询的影响

6.3　分片数(τ)对查询的影响

在第 2 组实验中，研究初始分片数对查询性能的影响. 实验采用 100000 个合成数据作为测试数据，其中维数为 32. 从图 9 中看出随着分片数的增加，ISU-Tree 索引的查询效率逐步提高，当分片数超过 8 时，查询效率就不再提高了. 这是因为分片数的增加会使实际查询区域越接近理想的查询区域(即查询超球与不确定超球相交部分)，因此在下面的实验中将分片数 τ 设为 8.

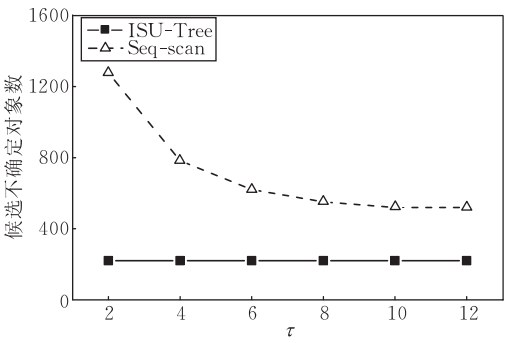
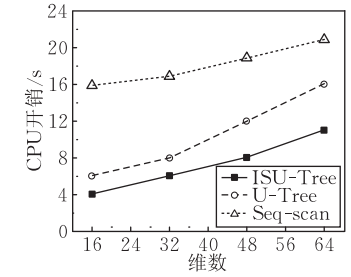


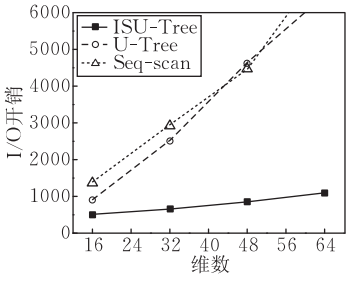
图 9 分片数对查询的影响

6.4 维数对查询的影响

本次实验研究维数对查询性能的影响. 实验采用 100000 个合成数据作为测试数据,其中维数从 16～64. 从图 10 中看出,较 ISU-Tree 而言,随着维数的增加,U-Tree 的 CPU 计算代价的增加幅度明显变大. 这是因为 ISU-Tree 能够在较高维度下有效地缩减查询过程中的搜索空间,使得其查询时间及磁盘块的访问次数大大减少. 同时,维数对其查询效率影响相对较小.



(a) CPU代价与维数比较

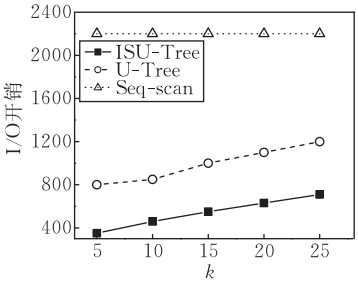


(b) I/O代价与维数比较

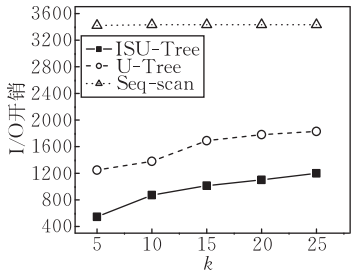
图 10 查询效率与维数比较

6.5 k 对查询的影响

本次实验采用两类数据作为测试数据集研究不同半径值对查询性能的影响. 从图 11 和图 12 可以看出,当 k 从 5～25 时,ISU-Tree 无论在 I/O 还是 CPU 计算代价方面都要明显优于其它方法. 在这些索引中,尽管 U-Tree 索引采用区域分片方法来缩小高维不确定搜索空间,其查询代价仍然非常高,仅次于顺序检索查询. 同时也可以看出,ISU-Tree 对于真实数据具有更好的过滤效果,使得它优于其它索引方法.

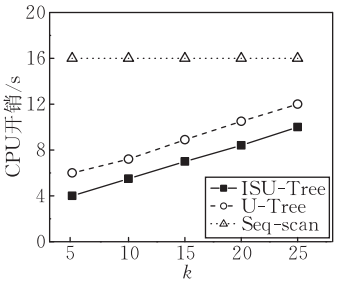


(a) 真实数据

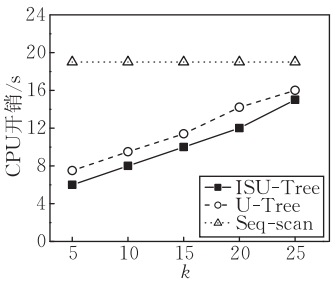


(b) 合成数据

图 11 k 与 I/O 代价比较



(a) 真实数据

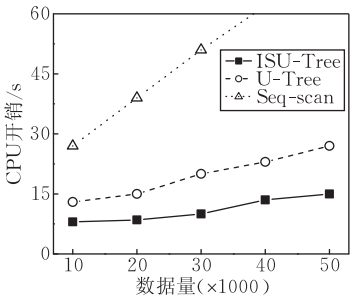


(b) 合成数据

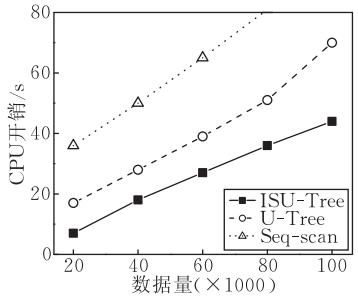
图 12 k 与 CPU 代价比较

6.6 数据量对查询的影响

最后实验研究数据量对查询性能的影响. 采用两类数据作为测试数据集执行范围概率查询. 图 13 从 CPU 开销方面比较了它们各自在查询性能上的差异. 实验表明顺序检索要远远高于 U-Tree 和 ISU-Tree,因为它在查询过程中需要进行 CPU 密集运算的概率积分操作. 同时从图 14 可以看出在 I/O 的开销方面,ISU-Tree 要优于其它两种方法.

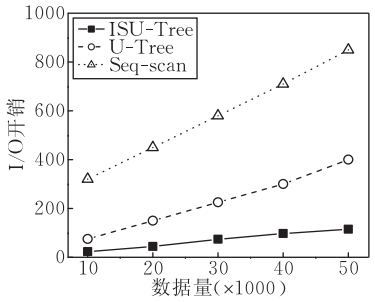


(a) 真实数据

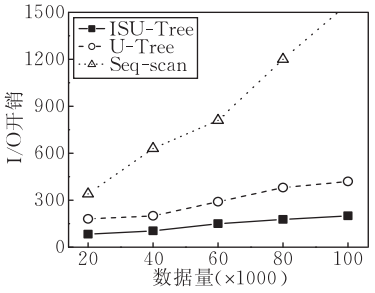


(b) 合成数据

图 13　CPU 代价与数据量比较



(a) 真实数据



(b) 合成数据

图 14　I/O 代价与数据量比较

7　结语及下一步工作

本文对高维不确定数据查询进行研究，提出一种基于初始分片的高维不确定索引方法——ISU-Tree. 该方法首先对高维不确定对象进行预处理聚类，然后对每个类中的对象进行基于初始编码及存在概率的统一编码，同时结合其对应的类信息生成统一索引键值，并采用 B^+ 树对其建立索引. 较其它方法，理论和实验都表明 ISU-Tree 能够有效地缩小搜索空间，从而明显减少概率积分运算的代价，优于其它同类索引方法，如 U-Tree 和顺序检索等.

为了进一步提高海量不确定高维查询效率，下一步将单机环境下的查询扩展到云计算环境，利用其强大的并行计算的优势，提高不确定查询性能.

参　考　文　献

[1] Deshpande A, Guestrin C, Madden S, Hellerstein J, Hong W. Model-driven data acquisition in sensor networks//Proceedings of the VLDB'04. Toronto, 2004: 588-599

[2] Gu Yu, Guo Na, Yu Ge. Study on processing probabilistic RFID spatial range query based on mobile readers. Chinese Journal of Computers, 2010, 32(10): 2052-2065 (in Chinese)
(谷峪，郭娜，于戈. 基于移动阅读器的 RFID 概率空间范围查询技术研究. 计算机学报，2010，32(10)：2052-2065)

[3] Cheng R, Xia Y, Prabhakar S, Shah R, Vitter J S. Efficient indexing methods for probabilistic threshold queries over uncertain data//Proceedings of the VLDB'04. Toronto, 2004: 876-887

[4] Bohm C, Berchtold S, Keim D. Searching in high- dimensional spaces: Index structures for improving the performance of multimedia databases. ACM Computing Surveys, 2001, 33(3): 322-373

[5] Guttman A. R-tree: A dynamic index structure for spatial searching//Proceedings of the ACM SIGMOD'84. Boston: ACM Press, 1984: 47-54

[6] Weber R, SChek H, Blott S. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces//Proceedings of the VLDB'98. New York: Morgan Kaufmann Publishers, 1998: 194-205

[7] Qi Y, Singh S, Shah R, Prabhakar S. Indexing probabilistic nearest-neighbor threshold queries//Proceedings of the Workshop on Management of Uncertain Data. New Zealand, 2008: 87-102

[8] Tao Y, Cheng R, Xiao X, Ngai W K, Kao B, Prabhakar S. Indexing multi-dimensional uncertain data with arbitrary probability density functions//Proceedings of the VLDB'05. New York, 2005: 922-933

[9] Beckmann N, Kriegel H-P, Schneider R et al. The R*-tree: An efficient and robust access method for points and rectangles//Proceedings of the ACM SIGMOD'90. Atlantic City: SIGMOD Record 19(2), 1990: 322-331

[10] Berchtold S, Bohm C, Kriegel H P et al. Independent quantization: An index compression technique for high-dimensional data spaces//Proceedings of the ICDE'00. USA: IEEE Computer Society, 2000: 577-588

[11] Jagadish H V, Ooi B C, Tan K L et al. iDistance: An adaptive B^+-tree based indexing method for nearest neighbor search. ACM Transactions on Data Base Systems, 2005, 30(2): 364-397

[12] Ding X, Lu Y. Indexing the imprecise positions of moving objects//Proceedings of the ACM SIGMOD 2007 Ph. D. Workshop on Innovative Database Research. Beijing, 2007: 45-50

[13] Beskales G, Soliman M, Ilyas I. Efficient search for the top-k probable nearest neighbors in uncertain databases//Proceedings of the VLDB. New Zealand, 2008: 326-339

[14] Kriegel H, Kunath P, Renz M. Probabilistic nearest- neighbor query on uncertain objects//Proceedings of DASFAA'07. India, 2007: 337-348

[15] Cheng R, Chen J, Mokbel M, Chow C. Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data//Proceedings of the ICDE'08. Mexico, 2008: 973-982

[16] Ljosa V, Singh A K. APLA: Indexing arbitrary probability distributions//Proceedings of the ICDE'07. Turkey, 2007: 946-955

[17] Widom J. Trio: A system for integrated management of data, accuracy and lineage//Proceedings of the CIDR'05. USA, 2005: 262-276

[18] Ding Xiao-Feng, Lu Yan-Sheng, Pan Peng, Hong Liang, Wei Qiong. U-tree-based uncertain mobile object indexing. Journal of Software, 2008, 19(10): 2696-2705(in Chinese)
(丁晓锋, 卢炎生, 潘鹏, 洪亮, 魏琼. 基于U-tree的不确定移动对象索引策略. 软件学报, 2008, 19(10): 2696-2705)

ZHUANG Yi, born in 1978, Ph. D., associate professor. His current research interests include uncertain data management, multimedia database and cloud computing, etc.

Background

Recently, uncertain data management has become a hot topic in database research community. Since the uncertain data, especially for the high-dimensional uncertain data is based on a "possible world" model. Moreover, due to the "Curse of Dimensionality", the query cost of high-dimensional uncertain data is very high. The paper proposes an index-support fast pruning uncertain region. Theoretical analysis and experimental evaluation both indicate that our proposed method achieves better performance comparing with the state-of-the-art algorithms.

This work is partially supported by the Program of National Natural Science Foundation of China under grant Nos. 60003074, 60873022, 60903053; The Program of Natural Science Foundation of Zhejiang Province under grant Nos. Z1100822, Y1080148, Y1090165; and the Science Fund for Young Scholars of Zhejiang Gongshang University under grant No. G09-7.