

# 基于 MOST 模型的移动对象全轨迹建模

何云斌<sup>1</sup>, 樊守德<sup>1</sup>, 郝忠孝<sup>1,2,3</sup>

(1. 哈尔滨理工大学计算机科学与技术学院, 哈尔滨 150080; 2. 齐齐哈尔大学计算机科学与技术学院, 齐齐哈尔 161006;

3. 哈尔滨工业大学计算机科学与技术学院, 哈尔滨 150001)

**摘要:** 针对以往建模方法的不足, 在移动对象时空(MOST)模型基础上结合离散建模的思想提出支持过去、现在、未来信息处理的全轨迹模型。MOST 模型不能处理历史信息, 可以在基于点的建模方法之上采用线性插值的思想来解决。它可以处理当前、短暂的未来信息, 可采用速度预测方法处理较长未来信息。给出全轨迹模型上 3 种速度预测方法及 2 种位置更新策略的比较结果。

**关键词:** 全轨迹建模; 移动对象时空模型; 速度预测; 位置更新

## Whole Trajectory Modeling of Moving Objects Based on MOST Model

HE Yun-bin<sup>1</sup>, FAN Shou-de<sup>1</sup>, HAO Zhong-xiao<sup>1,2,3</sup>

(1. College of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080;

2. College of Computer Science and Technology, Qiqihar University, Qiqihar 161006;

3. College of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001)

**【Abstract】** Because of the limitations of the previous modeling methods, this paper combines the ideas of the discrete modeling to propose the whole trajectory model, which is on the basis of the Moving Objects Spatio-Temporal(MOST) model and supports the past, present and future information processing. Adopting the linear interpolation of the point-based modeling method, the MOST model disposes the historical information. It can query the present and near future information. However, it adopts the efficiently predicted-speeds to dispose the further future information queryings. Experiment shows the comparison results of three predicted-speeds and two location update policies of the whole trajectory modeling.

**【Key words】** whole trajectory modeling; Moving Objects Spatio-Temporal(MOST) model; speed prediction; location update

### 1 概述

移动对象数据库是近年来的研究热点。移动对象在空间的位置随着时间发生变化, 如果用传统数据库管理系统来管理移动对象数据类型, 必须频繁地更新数据库。这不仅严重影响了数据库系统性能, 而且加重了无线带宽的开销。因此, 需要一种新的建模思路解决目前存在的问题。

移动对象数据库建模领域存在着 2 个并行发展的方向: 基于位置服务的管理与时空数据库。

以美国 Illinois 大学芝加哥分校的 O.Wolfson 为代表于 1997 年开发的用来存储随时间变化的位置信息模型<sup>[1-2]</sup>, 该模型不直接存储需要频繁更新的位置而是存储运动矢量。仅当移动对象的运动矢量(如速度)达到预定的门限时, 才更新数据库所存储的信息。其中的移动对象时空(Moving Objects Spatio-Temporal, MOST)模型提出了动态属性的概念, 介绍了瞬时、连续与持续查询的思想, 给出了相关的查询语言。此模型有一定的局限性, 即只处理移动对象的当前及不久将来的运动信息。

以德国哈根大学项目研究组 CHOROCRONOS<sup>[3]</sup> 于 1999 年开发的时空数据库模型, 目标是管理随时间变化的几何体。此模型带着相关的查询语, 完全可以处理历史信息, 也给出了数据类型与运算符的定义, 开发出了支持移动点、线、区域的处理算法。它对历史查询有很好的支持, 但对未来查询支持不足。

以法国国家科研中心为代表于 1998 年提出的时空数据库模型的约束表示模式, 主要集中在移动点的处理上, 原型系统是 DEDALE。

以上建模方法都不能很好地处理过去、现在、未来的信息并都存在片面性。本文在时空模型 MOST 基础之上, 结合离散建模的思想提出了支持过去、现在、未来信息处理的全轨迹模型。

### 2 相关问题

#### 2.1 时空模型 MOST

时空模型 MOST 引入了动态属性的概念, 将移动对象的位置表示为时间的函数, 移动对象的动态属性是其位置的属性。在移动对象正常运动时无须对其位置信息反复更新, 仅在发生异常情况(如速度突然变化或运动线路发生改变)时才进行数据库更新。对于曲折运动的对象而言, 这并非是很好的处理方法。为了使其不能支持历史信息的查询, 在此用折线段来近似模拟移动对象运动的全轨迹。

#### 2.2 离散模型的建模思想

移动对象离散数据模型将复杂的空间对象及移动轨迹分

**基金项目:** 黑龙江省自然科学基金资助项目(F200601)

**作者简介:** 何云斌(1972 - ), 男, 博士, 主研方向: 时空数据库; 樊守德, 硕士; 郝忠孝, 教授、博士生导师

**收稿日期:** 2007-09-20 **E-mail:** hybha@163.com

割为相对简单的离散片段。要利用计算机来处理信息必须将抽象数据模型离散化。在进行抽象数据模型处理时，常把离散模型作为它的近似表示。

### 3 全轨迹的定义

**定义 1** 在三维  $X \times Y \times T$  空间里，一个移动对象的轨迹  $Tr$  是由序列点  $(x_1, y_1, t_1), (x_2, y_2, t_2), \dots, (x_n, y_n, t_n)$  ( $t_1 < t_2 < \dots < t_n$ ) 构成的折线，其中，折线中的每一段都是直线段。轨迹  $Tr$  在二维  $X \times Y$  平面上的投影叫做路径  $Rr$ 。

轨迹把移动对象的位置定义为时间的隐式函数，移动对象在时刻  $t_i$  的位置坐标是  $(x_i, y_i)$ ，在时间区间  $[t_i, t_{i+1}]$  上从点  $(x_i, y_i)$  到点  $(x_{i+1}, y_{i+1})$  以速度  $v_i$  进行匀速直线运动。速度  $v_i$  的值可以由

公式  $v_i = \frac{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}{t_{i+1} - t_i}$  计算得到。在时间区间

$[t_i, t_{i+1}]$  ( $1 \leq i < n$ ) 上移动对象在轨迹  $Tr$  上  $(x_i, y_i)$  与  $(x_{i+1}, y_{i+1})$  2 点之间的位置可以通过线性插值方法得到。

**定义 2** 在三维  $X \times Y \times T$  空间里，一个移动对象的将来轨迹  $Tr$  由七元组序列  $(MOID, FUTNUMBER, LOC_0, T_0, DIRECTION, V_0, UNCERTAINTY)$  构成的折线组成，其中， $MOID$  是移动对象表示符； $FUTNUMBER$  是将来轨迹段表示符； $LOC_0$  是更新初始位置； $T_0$  是更新初始时间； $DIRECTION$  是轨迹方向； $V_0$  是更新初始速度； $UNCERTAINTY$  是更新门限。这里  $V_0$  的值由下面的速度预测方法求出。

**定义 3** 在三维  $X \times Y \times T$  空间里，一个移动对象过去轨迹  $Tr$  由六元组序列  $(MOID, PASNUMBER, STARTLOC, STARTT, ENDLOC, ENDT)$  构成的折线组成，其中，邻近的 2 个点由直线段相连接。轨迹  $Tr$  在二维  $X \times Y$  平面上的投影叫做路径  $Rr$ 。2 个六元序列之间任何时刻的位置可以通过线性插值得到。其中， $MOID$  是移动对象表示符； $PASNUMBER$  是过去轨迹段表示符； $STARTLOC$  是移动对象开始位置； $STARTT$  是移动对象开始时刻； $ENDLOC$  移动对象结束位置； $ENDT$  是移动对象结束时刻。

### 4 速度预测方案

为了得到移动对象的将来轨迹，往往采用速度预测技术，速度预测一般有 2 种预测模型<sup>[4]</sup>：延时模型与滑动平均模型。将  $t_{n+1}$  设为未来采样时刻，在  $t_{n+1}$  时刻到来之前需要对速度  $V_n$  进行预测。由于  $V_n$  为矢量，其预测包括标量值预测和方向预测。

#### (1) 延时模型

$$\hat{v} = v_{n-1}$$

即未来时刻速度标量值和方向都与最后一个历史速度  $V_{n-1}$  的标量值和方向一致。

#### (2) 滑动平均模型

$$|\hat{v}_n| = \frac{1}{p} \sum_{i=1}^p v_{n-i}$$

即未来时刻速度标量值是前  $P$  个历史速度的标量均值，其方向与最后一个历史速度  $V_{n-1}$  的方向一致。

显然，延时模型对移动对象运动状态的改变适应得更快，而滑动平均模型对速度标量值的预测比较准确。虽然 2 种速度预测方法在一定程度上对速度作了预测，但预测精度值不高。为了对未来速度作出准确的预测，下面给出带有滑动因子<sup>[5]</sup>的速度预测模型。

#### (3) 滑动因子预测模型

此模型也是采用历史的速度去预测未来速度，关键是滑

动因子  $\alpha$  的选择，滑动因子决定了速度的预测，可以由以下公式得出：

$$\begin{cases} \alpha = \frac{\alpha_1 + \alpha_2}{2} & 0 < \alpha < 1 \\ \alpha = 1 & \alpha > 1 \\ \alpha = 0 & \alpha < 0 \end{cases} \quad (1)$$

其中，

$$a_1 = \frac{\sum_{i=3}^n (v_i - v_{i-2})(v_{i-1} - v_{i-2})}{2 \sum_{i=3}^n (v_{i-1} - v_{i-2})^2} \quad (2)$$

$$a_2 = \frac{3}{4} \left[ \frac{\sum_{i=3}^n (v_i - v_{i-1})(v_{i-1} - v_{i-2})}{\sum_{i=3}^n (v_{i-1} - v_{i-2})^2 + 2 \sum_{i=4}^n (v_i - v_{i-1})(v_{i-2} - v_{i-3})} + 1 \right] \quad (3)$$

其中， $V_i$  是当前时刻  $t$  的速度； $V_{i-1}$  ( $i=1, 2, 3$ ) 是当前时刻  $t$  之前的第  $i$  时刻速度； $n$  是计算滑动因子  $\alpha$  的时刻总数。

历史速度计算可以由上面提到的计算方法得出，假定采样时刻的历史位置为  $(x_{t-3}, y_{t-3}), \dots, (x_{t-2}, y_{t-2}), (x_{t-1}, y_{t-1}), (x_t, y_t)$ ，其中， $0 \leq i < n$ ，则有

$$v_{i+1} = \sqrt{(x_{t+i+1} - x_{t+i})^2 + (y_{t+i+1} - y_{t+i})^2} / \Delta t \quad (4)$$

其中， $\Delta t$  是 2 个连续采样点的时间间隔。假定要预测未来时刻  $t+1$  的速度  $\hat{v}_{t+1}$ ，则有

$$\hat{v}_{t+1} = \alpha v_t + \alpha(1-\alpha)v_{t-1} + \alpha(1-\alpha)^2 v_{t-2} + \dots + \alpha(1-\alpha)^N v_{t-N} \quad (5)$$

其中， $N$  ( $0 < N < t$ ) 是历史记录的长度，由式(6)得出。

$$N = \frac{(2-\alpha)}{\alpha} \quad (6)$$

如果要预测未来时刻  $i$  ( $i > t+1$ ) 的速度，先计算  $\hat{v}_{t+2} = \alpha v_{t+1} + (1-\alpha)\hat{v}_{t+1}$ ，然后用  $\hat{v}_{t+2}$  的值估算  $\hat{v}_{t+3}$  的值， $\hat{v}_{t+3} = \alpha v_{t+2} + (1-\alpha)\hat{v}_{t+2}$ ，再用  $\hat{v}_{t+3}$  的值估算  $\hat{v}_{t+4}$  的值， $\hat{v}_{t+4} = \alpha v_{t+3} + (1-\alpha)\hat{v}_{t+3}$ ，...，按此逐步计算未来时刻  $i$  的速度  $\hat{v}_i$ 。

### 5 位置更新策略

位置更新策略通过选择合适的更新方式，降低移动对象位置的不确定性、不精确性与系统资源的占用率，并保证查询结果具有一定的准确性。本文给出 2 种位置更新策略的比较。

#### (1) 速度更新预测策略

在行程开始移动对象  $M$  发给移动对象数据库管理系统一个不确定门限值。这个门限值被存储在  $UNCERTAINTY$  中并在整个行程中保持不变。一旦  $M$  的偏差超过  $UNCERTAINTY$  就会更新数据库。每次更新只需要包含移动对象当前的位置、速度和方向。门限值是随机给定的。

#### (2) 自适应预测策略<sup>[1]</sup>

在行程的开始时，移动对象  $M$  发送给移动对象数据库管理系统一个随意选择的初始化门限值  $th_1$ 。当偏差达到  $th_1$  时，移动对象  $M$  发送更新给数据库。每次更新都包含移动对象的当前位置、速度、方向和一个新的门限  $th_2$ 。数据库把这个新门限值放进  $UNCERTAINTY$  子属性中。 $th_2$  是这样得到的： $t_1$  表示从行程的开始到移动对象的偏差第 1 次达到  $th_1$  所经过的时间单元数。 $I_1$  表示在同样时间间隔里的偏差代价，令  $a_1 = \frac{2I_1}{t_1}$ 。

那么  $th_2$  可以取为  $\sqrt{\frac{2a_1 C_1}{1+2C_2}}$ ，其中， $C_1$  是更新代价； $C_2$  是单位不确定代价。当偏差到达  $th_2$  的时候也会发送一个类似的更新， $th_3$  取为  $\sqrt{\frac{2a_2 C_1}{1+2C_2}}$ ，其中， $a_2 = \frac{2I_2}{t_2}$  ( $I_2$  是从第 1 次更新到第 2 次更新之间的偏差代价； $t_2$  表示从第 1 次位置更新以来经过的时间

单元数)。由于 $a_2$ 可能与 $a_1$ 不同， $th_3$ 也可能与 $th_2$ 不同，因此当移动对象的偏差达到 $th_3$ 后，应当发送另一个包含 $th_4$ 的更新， $th_4$ 的计算方法与前面相同。

自适应预测策略在移动对象每次更新时都会提供一个新的门限。它使得直至下次更新之前单位时间内的总信息代价最小。移动对象需要估计下次更新的产生时刻，而未来的偏差在位置更新时刻进行下次更新时刻估计时是不知道的。这就需要自适应预测策略根据过去的偏差来预测将来的偏差。门限每次更新时计算得出的值也是不同的，从而保证了预测的精度。

## 6 实验分析

实验环境为 2.93 GHz 奔腾 4 处理器、512 MB 内存、80 GB 硬盘、Windows XP 操作系统、Visual C++6.0 编程工具与 Oracle8i 数据库管理系统。

在全轨迹建模实验中，移动对象 $M$ 在未来时刻 $T$ 的预测位置为 $PredLoc$ ，实际位置为 $LacLoc$ ，更新时刻 $T_0$ 移动对象的位置取 $LOC_0$ ，预测初始速度为 $V_0$ ，更新的门限为 $UNCERTAINTY$ ，在离上次更新的时间间隔为 $t=T-T_0$ 时，移动对象 $M$ 的预测位置为 $PredLoc=LOC_0+V_0t$ 。由于 $M$ 在任何时候都能确切地知道自己所在的实际位置 $LacLoc$ ，因此在该时刻的偏差 $Devi$ 为 $Devi=|LacLoc-PredLoc|$ ，当有 $Devi>UNCERTAINTY$ ， $M$ 就向数据库服务器发送更新信息，使数据库服务器更新当前的位置、速度和方向。

下面就未来 30 min 里在速度更新与自适应预测策略的条件下，分别对延时、滑动平均与滑动因子预测模型 3 种速度预测方法时 CPU 上的耗时和位置的预测精确性作了比较，结果如图 1~图 4 所示。其中，速度更新预测策略的初始门限值 $UNCERTAINTY=2.5$ ；自适应预测策略的初始门限值 $UNCERTAINTY=3$ ；初始速度值 $V_0=10$ ；初始位置 $LOC_0=0$ ；初始时刻 $T_0=0$ 。对速度更新预测策略与自适应预测策略在未来不同时刻 $T$ 对将来位置查询结果的影响如图 5 所示。

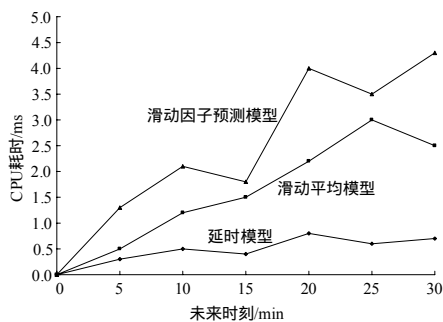


图 1 基于速度更新预测策略计算未来不同时刻速度的 CPU 耗时

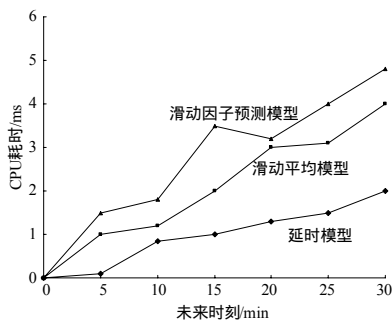


图 2 基于自适应预测策略计算未来不同时刻速度的 CPU 耗时

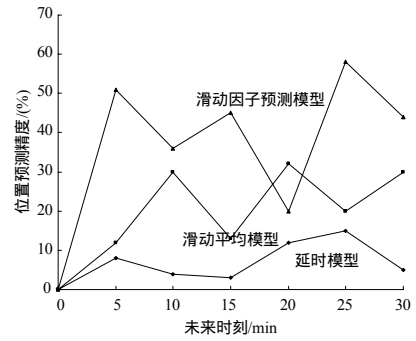


图 3 基于速度更新预测策略的位置预测精度

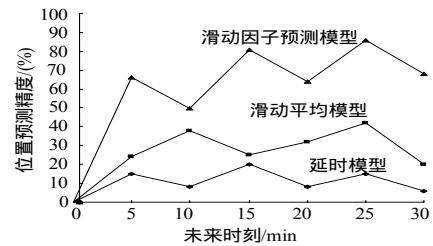


图 4 基于自适应预测策略的位置预测精度

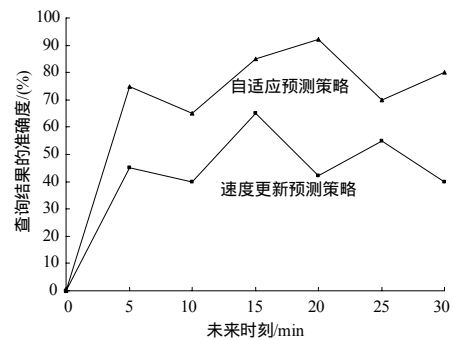


图 5 速度预测策略在未来不同时刻 $T$ 对将来位置查询结果的影响

## 7 结束语

本文在时空模型 MOST 基础之上结合离散建模的思想提出了支持过去、现在、未来信息处理的全轨迹模型。采用 3 种速度预测方案与 2 种位置更新策略，以提高此模型对未来位置预测的准确性，减少网络的通信开销。

## 参考文献

- [1] Wolfson O, Sistla A P, Chamberlain S, et al. Updating and Querying Databases That Track Mobile Units[J]. Distributed and Parallel Databases, 1999, 7(3): 257-387.
- [2] Trajcevski G, Wolfson O, Hinrichs K, et al. Managing Uncertainty in Moving Objects Databases[J]. ACM Trans. on Database Syst., 2004, 29(3): 463-507.
- [3] Erwig M, Güting R H, Schneider M, et al. Spatio-temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases[J]. GeoInformatica, 1999, 3(3): 265-291.
- [4] Saltenis S, Jensen C S, Leutenegger S T, et al. Indexing the Positions of Continuously Moving Objects[C]//Proc. of SIGMOD'00, 2000: 331-342.
- [5] Tang Yansen. The New Method of Getting the Value of Smoothing Constant[J]. Statistics and Information, 1997, 3(1): 15-18.