

一种启发式的球面点定位策略*

吴勇⁺, 何援军, 蔡鸿明

(上海交通大学 计算机科学与工程系, 上海 200030)

A Heuristic Strategy for Spherical Point Location

WU Yong⁺, HE Yuan-Jun, CAI Hong-Ming

(Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China)

+ Corresponding author: Phn: +86-21-62933840, E-mail: wuyong916@sjtu.edu.cn, http://www.sjtu.edu.cn

Received 2004-11-02; Accepted 2005-01-07

Wu Y, He YJ, Cai HM. A heuristic strategy for spherical point location. *Journal of Software*, 2005,16(11): 1983-1991. DOI: 10.1360/jos161983

Abstract: A heuristic strategy is presented to solve the point location problem in spherical triangulation mesh. Firstly, a spherical mesh with regular subdivision connectivity is constructed to partition the spherical domain into some small regions. Then the region, which contains the query point p , is found according to the position of p and selected as the search area for locating p . During the point location, the barycentric coordinates are used to extract local heuristic information about the location of p so as to find the shortest path from the start triangle to the target one containing p . In comparison with traditional algorithms, it is found that the heuristic strategy has better time and space performances.

Key words: spherical mesh; point location; heuristic strategy; subdivision mesh; barycentric coordinate

摘要: 提出了一种适用于球面三角形网格的启发式点定位策略. 通过构造规则子分网格将原始球面网格区域划分成若干具有子分层次关系的查询小块; 在进行查询前, 根据查询点 p 的位置找到它所在的小块作为搜索区域, 从而极大地缩小了查询范围; 在查询过程中, 根据重心坐标所包含的启发信息, 选择一条从初始搜索三角形到目标三角形的最短查询路径. 分析表明, 启发式点定位策略比传统算法具有更优的运算性能.

关键词: 球面网格; 点定位; 启发式策略; 子分网格; 重心坐标

中图法分类号: TP391 文献标识码: A

平面点定位是计算几何方面的一个基本问题. 它在计算机图形学、计算机辅助设计、地理信息系统等许多领域有着广泛的应用. 国内外学者对其进行了大量研究^[1-4]. 近年来, 随着球面参数化应用的深入^[5-8], 球面点定位问题逐渐受到学者们的关注. 给定球面网格 C 及其上一点 p , 球面点定位问题要解决的就是如何在 C 上找到包含 p 的网格面片. 已有的球面点定位算法^[9,10]往往是对传统的平面点定位算法的一种扩展, 这种扩展虽然不

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2003AA411310 (国家高技术研究发展计划(863))

作者简介: 吴勇(1977 -), 男, 湖北鄂州人, 博士, 主要研究领域为网络虚拟设计, 三维几何处理, 计算机动画; 何援军(1945 -), 男, 教授, 博士生导师, 主要研究领域为 CAD&CG, 几何计算; 蔡鸿明(1975 -), 男, 博士, 讲师, 主要研究领域为产品创新设计, 知识工程.

会增加算法的时空复杂度,但是也没有充分利用球面这一特殊空间域来提高算法性能.针对这一问题,本文提出一种适用于球面网格的启发式点定位策略,通过构造规则球面网格和运用基于重心坐标的启发式搜索策略,有效地降低了算法的空间和时间复杂度.为了便于处理,我们假设 C 是二维流形的球面三角形网格,对于多边形网格,可以通过三角化算法将其转换为三角形网格后再处理.

1 相关工作

对于网格上一个给定点,最简单的定位方法是顺序遍历并检测每个三角形,直到找到目标三角形为止.假设网格有 n_f 个三角形,则定位每一个点需要执行 $n_f/2$ 次三角形检测,当 n_f 很大时,这种搜索方法的效率很低.为了便于区分,我们称这种依次检测所有三角形的点定位方法为盲目式方法.在前面的时间复杂度公式中, λ 表示计算算法运行时间的常系数.虽然它不会决定复杂度的大小,但是能够影响算法运行的实际时间.当两个算法的时间复杂度相同时, λ 可以用来评估它们的优劣.

最早的有效点定位算法是由 Dobkin 和 Lipton^[1]提出的.在预处理阶段,他们过平面网格 M 的每个顶点作直线将平面分成若干带状区域(slabs),然后依次排序这些 slabs;在定位阶段,他们利用二分搜索法来确定包含 p 的目标三角形.由于二分搜索法的时间复杂度是 $O(\log n_f)$,因此 slab 方法的单点查询时间复杂度为 $O(\log n_f)$,但是该方法在执行搜索操作前的预处理阶段需要额外花费 $O(n_f^2)$ 的空间和 $O(n_f^2 \log n_f)$ 的时间来构造特殊的数据结构,因此,如果给定 n_v 个查询点,则该算法的额外空间和总体时间复杂度分别为 $O(n_f^2)$ 和 $O(n_f^2 \log n_f + n_v \log n_f)$.

Kirkpatrick^[2]构造了一种层次数据结构来解决平面点定位问题.分析发现,该算法的单点查询时间复杂度为 $O(\log n_f)$,而预处理的空间和时间复杂度分别为 $O(n_f)$ 和 $O(n_f \log n_f)$.因此,对于有 n_v 个查询点的定位问题,总体时空复杂度分别为 $O(n_f)$ 和 $O(n_f \log n_f + n_v \log n_f)$.虽然该算法比 slab 方法的复杂度要低,但是在计算实际运行时间时所涉及到的常系数 λ 的值比较大.另外,算法实现起来也比较困难.

针对上述算法需要为预处理操作增加额外开销的问题,Sundareswara 等人提出了一种不需要进行预处理的基于重心坐标的启发式点定位算法^[3].该算法虽然无须花费额外的空间和时间来构造特殊的数据结构,但是对初始搜索三角形 $f(p_0, p_1, p_2)$ 的选择是盲目的、随机的,其单点查询时间复杂度为 $O(\sqrt{n_f})$.当查询点的数目 $n_v > \sqrt{n_f} \log n_f$ 时,该方法的时间复杂度将高于 Kirkpatrick 算法^[2].

上述几种算法虽然是基于平面点定位问题提出来的,但是可以推广到球面,而且推广后的时空复杂度不变.在本文后续章节的讨论中,如果不作特殊说明,所涉及到的这些算法都是指扩展到球面后的点定位算法.

2 基本数据结构

球面网格 C 可以表示为一个二元组 $C = (P_C, K_C)$,其中 $P_C = \{p_1, \dots, p_n\}$ 是 C 的顶点几何属性集合(n 表示顶点数目),每个顶点的球面几何位置表示为 $p_i = (\theta_i, \varphi_i)$; K_C 是包含 C 中所有拓扑关系的集合,其元素分为 3 种类型:点、边、面,其中点有指针指向 P_C 中的属性;另外,每个点还有指针指向相邻边,每个边有指针指向相邻三角形面片,每个面片有指针指向所包含的 3 个顶点.根据上述定义的拓扑关系可知,通过 K_C 中的任何一个元素都可以在 $O(1)$ 时间内访问到拓扑相邻的其他元素.在本文的算法中还将用到一个球面规则网格 S ,其数据结构的定义与 C 类似,可以用一个二元组 $S = (P_S, K_S)$ 来表示.

3 启发式点定位策略

本文所提出的启发式点定位策略包括预处理和搜索两个阶段.在预处理阶段,我们首先在原始球面网格 C 所在区域(一般为单位球面)构造另一个具有规则拓扑关系的球面网格 S ,将球面区域划分为若干规则小块,然后为每一个规则小块 f_s 从 C 中找到一个位于 f_s 所决定的球面区域的三角形 f_c ;在查询阶段,我们首先根据给定点 p 的位置,在规则球面 S 上找到包含 p 的规则小块 f_s ,然后将与 f_s 对应的原始网格 C 上的三角形 f_c 作为初始搜索对象,利用重心坐标方法(见第 3.4 节)对 C 进行搜索,直到找到包含 p 的目标三角形为止.为了与

Sundareswara 的基于重心坐标的点定位算法^[3]相区别,我们称这种选择初始搜索三角形的基于重心坐标的球面点定位算法为启发式点定位策略。

根据上述策略可知, S 和 C 是位于同一球面域上的两个具有不同拓扑关系的网格, S 上的每一个三角形 f_s 与 C 上的一个三角形 f_c 对应,且后者位于前者所决定的球面区域中。

3.1 规则球面网格的构造

构造规则球面网格的目的是将单位球面划分成若干均匀小块,然后根据点 p 的位置将搜索范围限定在一个较小的区域中,从而提高搜索效率。规则网格的构造方法有许多,为了保证小块的均匀性,我们选择球的内接正八面体作为初始网格,然后将其上所有的边都用球面上连接该两端点的大弧替换,则替换后的网格就形成了球面基网格 S^0 ;之后,我们采用 1-4 子分操作对基网格的 8 个面进行逐层细化,最终得到具有子分拓扑关系的规则球面网格(如图 1 所示)。在构造 S^l 的过程中,子分层次 l 的大小是决定点定位算法效率的一个关键,它的确定方法将在第 4 节加以讨论。

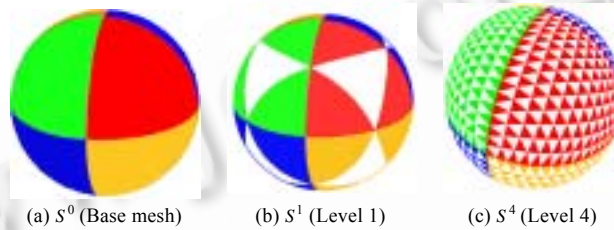


Fig.1 Regular spherical meshes at several different subdivision levels

图 1 几个不同子分层次的规则网格

3.2 代表三角形的确定

在构造了 S^l 之后,我们需要为其上的每一个子分三角形 f_s 从原始球面网格 C 中找到一个代表该区域的三角形 f_c ,如果给定点 p 位于 f_s 所在区域中,则对应的 f_c 就被选为初始搜索三角形。为了便于描述,我们称 f_c 为 f_s 的代表三角形。

在为每个规则子分三角形 f_s 确定对应的 f_c 时,我们首先求得 f_s 的重心 q ,然后从 C 中找到包含 q 的三角形作为 f_c 。对于球面三角形,由于其具有与平面三角形类似的重心性质:重心既是三条中线大弧的交点,同时又将每条中线分割成比例为 2:1 的两段小弧,所以我们可以利用球面几何知识求得 q 点的位置。

3.3 初始搜索三角形的确定

在定位每一个查询点 p 之前,需要根据 p 的位置在 C 上选择一个初始搜索三角形。我们的思路是:首先为 p 点在 S^l 上找到一个包含它的三角形 f_s^p ,然后选择对应于 f_s^p 的代表三角形 f_c^p 作为定位 p 点的初始搜索对象。对于按照第 3.1 节的方法构造的规则球面网格序列 S^0, S^1, \dots, S^l ,由于相邻两个网格之间具有层次细分关系,我们不再直接采用第 3.4 节的基于重心坐标的搜索方法,而是利用四分搜索法确定 f_s^p :首先确定基网格 S^0 上包含 p 的三角形 $f_{S^0}^p$,将搜索范围缩小到 $f_{S^0}^p$ 所确定的球面区域;然后从第 1 层子分网格 S^1 上对应于 $f_{S^0}^p$ 所在区域的 4 个子分三角形中找到一个包含 p 的三角形 $f_{S^1}^p$ 作为下一层子分网格的搜索对象,我们逐层执行上述搜索操作,直到找到 S^l 上包含 p 的三角形 $f_{S^l}^p$,然后将对应于 $f_{S^l}^p$ 的代表三角形 f_c^p 选为初始搜索三角形。在上述搜索过程中,除了第 1 层之外,每一层网格上的操作被执行后,搜索区域都缩小为原来的 1/4,因此我们称其为四分搜索法。

3.4 基于重心坐标的搜索策略

在确定了初始搜索三角形之后,我们利用基于重心坐标的搜索方法查询包含给定点 p 的球面三角形。

3.4.1 重心坐标的计算

对于二维平面上的一个给定点 p 和一个给定三角形 $f(p_0, p_1, p_2)$,由于两者共面,我们可以直接应用下述

重心坐标公式求得 p 相对于 f 的重心坐标.

$$\begin{cases} p = \alpha p_0 + \beta p_1 + \gamma p_2 \\ \alpha + \beta + \gamma = 1 \end{cases} \quad (1)$$

但是如果 $f(p_0, p_1, p_2)$ 是球面三角形,而 p 是球面网格上的一点,则公式(1)不能直接用来求解这种空间曲面三角形的重心坐标.为了解决这一问题,我们放弃直接考虑 $f(p_0, p_1, p_2)$,转而研究由相同 3 点 p_0, p_1, p_2 所确定的空间平面三角形 $f'(p_0, p_1, p_2)$.显然, p 点一般不和平面三角形 f' 共面,为了能够应用公式(1),我们考虑 p 点在 f' 上的投影点 p' ,本文中投影点 p' 被设定为 op 与三角形 f' 所在平面的交点,其中 op 表示球心 o 与点 p 所确定的直线.图 2 给出了点在平面三角形上的两种可能的投影结果.

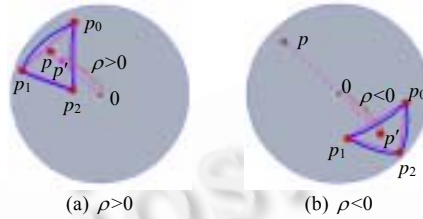


Fig.2 Projection of p onto the planar triangle $f'(p_0, p_1, p_2)$

图 2 p 点在平面三角形 $f'(p_0, p_1, p_2)$ 上的投影

在图 2 中,我们假设:球心 o 的坐标为 $(0,0,0)$,矢量 \vec{op} 的方向为 (l, m, n) ,三角形 f' 所在平面的一般方程为 $Ax + By + Cz + D = 0$,则投影点 p' 的位置可用下式计算.

$$p' = (\rho l, \rho m, \rho n) \quad (2)$$

其中,系数 $\rho = -\frac{D}{Al + Bm + Cn}$.在求得平面三角形 $f'(p_0, p_1, p_2)$ 和投影点 p' 后,我们可以利用公式(1)求得 p' 相对于 f' 的重心坐标 (α, β, γ) ,然后根据 $\rho, \alpha, \beta, \gamma$ 的取值来确定下一搜索三角形.

3.4.2 下一搜索三角形的确定

在得到 $\rho, \alpha, \beta, \gamma$ 之后,根据 ρ 的不同取值,我们需要讨论 3 种情况.

首先,当 $Al + Bm + Cn = 0$ 时,矢量 \vec{op} 与三角形 f' 法向量正交,此时 $\rho = \infty$,点 p 在 f' 所在的平面上无投影点,于是我们从球面三角形 f 的邻居三角形集合中任意选取一个未被访问的三角形作为下一搜索对象.

其次,当 $\rho \neq \infty$ 且 $\rho > 0$ 时,从图 2(a)可以看到矢量 \vec{op}' 与 \vec{op} 有相同的方向.在计算出 p' 相对于 f' 的重心坐标 (α, β, γ) 之后,根据 (α, β, γ) 的取值可以将 p' 与 f' 之间的位置关系划分为图 3 所示的 3 种情况.如图 3(a)所示的第 1 种情况,如果 $\alpha \geq 0, \beta \geq 0, \gamma \geq 0$,则 p' 位于 f' 内部,我们可以立刻得出 f 就是包含 p 点的目标三角形;如果 3 个重心坐标中仅有一个小于 0,比如 $\alpha < 0, \beta \geq 0, \gamma \geq 0$ (如图 3(b)所示),此时我们能够判断 p' 一定位于顶点 p_0 的对边 $\{p_1, p_2\}$ 的右边(假设三角形 $f'(p_0, p_1, p_2)$ 的顶点遍历方向是逆时针),因此我们选择球面网格上与球面三角形 f 通过大弧 $\{p_1, p_2\}$ 相连的邻居三角形 f_1 作为下一搜索对象;图 3(c)给出了第 3 种情况,此时有两个重心坐标小于 0,比如 $\alpha < 0, \beta < 0, \gamma \geq 0$,根据重心坐标的特性,我们可以判断 p' 位于边 $\{p_1, p_2\}$ 的右半平面与边 $\{p_2, p_0\}$ 右半平面的相交区域中(如图 3(c)的交叉阴影部分),此时下一搜索三角形应该从交叉区域选取,因此我们首先找到 p_2 的 1 环邻居球面三角形集合 $T(p_2)$,然后从 $T(p_2)$ 中去除 f 以及与 f 通过大弧 $\{p_1, p_2\}, \{p_2, p_0\}$ 相连的两个邻居三角形,得到剩余三角形集合 $T^*(p_2)$,然后从 $T^*(p_2)$ 中任选一个未被访问的元素作为下一搜索对象.

最后,当 $\rho \neq \infty$ 且 $\rho < 0$ 时,从图 2(b)可以看到矢量 \vec{op}' 与 \vec{op} 的方向相反.此时,下一搜索三角形的选择策略与图 2(a)完全相反:如果 $\alpha \geq 0, \beta \geq 0, \gamma \geq 0$,则 p 点位于单位球面上与 f 相对的半球面区域中,于是我们任意选择一个与 f 相连的邻居三角形作为下一搜索对象;如果 3 个重心坐标中只有一个小于 0,如 $\alpha < 0, \beta \geq 0, \gamma \geq 0$,则令 $\alpha \geq 0, \beta < 0, \gamma < 0$,并运用前述 $\rho > 0$ 情况下处理图 3(c)的方法来确定下一搜索对象;如果 3 个重心坐标中有两个小于 0,如 $\alpha < 0, \beta < 0, \gamma \geq 0$,则令 $\alpha \geq 0, \beta \geq 0, \gamma < 0$,并运用前述 $\rho > 0$ 情况下处理图 3(b)的方法来确定下一搜索对象.

3.5 算法要点

综合上述算法细节,我们的启发式点定位策略有如下一些要点:

Step 1. 利用第 3.1 节的方法构造具有子分拓扑结构的球面规则网格 S^l ;

Step 2. 利用第 3.2 节的方法为 S^l 上的每一个三角形从原始球面网格 C 找到一个代表三角形;

Step 3. 对于给定的查询点 p ,利用第 3.3 节的方法确定初始搜索三角形 f_C^p ;

Step 4. 利用第 3.4 节的基于重心坐标的搜索方法,从 f_C^p 开始在原始球面网格 C 上查询包含 p 的目标三角形.如果还有其他查询点,则反复执行 Step 3 和 Step 4.

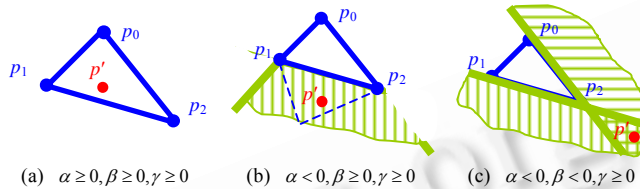


Fig.3 Three kinds of possible relationships between the point p' and the triangle $f'(p_0, p_1, p_2)$

图 3 点 p' 与 $f'(p_0, p_1, p_2)$ 之间位置关系的 3 种可能情况

4 算法性能分析

首先假设原始球面网格 C 的三角形个数为 n_f ,需要进行定位的点的数目为 n_v ,为了便于比较,我们定义一个表示 n_v 与 n_f 之间大小关系的系数 $\varepsilon = \log_{n_f}(n_v)$,于是 n_v 可以表示为 $n_v = n_f^\varepsilon$.另外,对于本文所研究的球面网格,其三角形数目 n_f 与顶点数目 n 之比大约等于 2,所以我们在计算复杂度时直接用 n_f 代替 n .

对于盲目式搜索算法,定位每个点时都需要顺序遍历并检测 C 上的三角形,因此为 n_v 个查询点定位的时间复杂度可以表示为 $O(n_v \cdot n_f) = O(n_f^{1+\varepsilon})$.由于不用进行预处理,该算法不需要花费任何额外的存储空间.

接下来我们分析本文所提出的启发式点定位策略的平均时间和额外空间复杂度.首先假设规则网格的子分层次为 l ,根据第 3.5 节的描述,整个算法的计算时间 T 由 4 部分组成:基网格的构造时间 T_b 、代表三角形的确定时间 T_r 、初始搜索三角形的确定时间 T_s 以及最后的查询时间 T_h ,用公式表示如下:

$$T = T_b + T_r + T_s + T_h \tag{3}$$

在构造球面基网格时,由于除了 S^l 之外在每一子分网格上都要对每一个三角形进行子分,所以我们有:

$$T_b = a_0(n_{S^0} + n_{S^1} + \dots + n_{S^i} + \dots + n_{S^{l-1}}) = \frac{a_0}{3}(4^l - 1) \tag{4}$$

其中, a_0 是常系数, n_{S^i} 表示 S^i 所包含的三角形数目,对于第 3.1 节构造的网格, $n_{S^i} = 8 \times 4^i$.

在为 S^l 上的所有三角形从原始球面网格 C 中寻找代表三角形时,我们采用的是先计算三角形重心,再利用扩展后的重心坐标方法进行搜索的策略.根据前面的分析,代表三角形的确定时间:

$$T_r = b_0 n_{S^l} + b_1 n_{S^l} \sqrt{n_f} = 8 \times 4^l (b_0 + b_1 \sqrt{n_f}) \tag{5}$$

其中, b_0, b_1 是常系数.

利用四分搜索法为 n_v 个查询点确定初始搜索三角形的时间为

$$T_s = c_0 n_v (8 + l) \tag{6}$$

其中, c_0 是常系数.

在计算查询时间 T_h 之前,我们先来分析利用重心坐标方法为单个点定位的时间.由于原始球面网格 C 的三角形分布是不规则的,我们无法精确计算 T_h ,只能给出近似估算值.在这里,首先假设 C 的三角形分布完全均匀规则,则 S^l 上的每一个三角形区域 f_s 所覆盖的 C 上的三角形个数的平均值为 $\frac{n_f}{8 \times 4^l}$,并且这些三角形的分布情况可近似地用图 4 表示.根据启发式点定位策略可知, C 上的查询点 p 和对应的初始搜索三角形 f_C^p (如图 4 所



Fig.4 Uniform distribution of spherical triangles in C covered by $f_S(p_0, p_1, p_2)$

图4 $f_S(p_0, p_1, p_2)$ 中所包含 C 中球面三角形的规则分布情况

要搜索时间为

$$T_h = d_0 n_v m_p \leq d_0 n_v \left(\sqrt{\frac{n_f}{2 \times 4^l}} - 1 \right) \tag{7}$$

其中, d_0 是常系数. 这里需要注意一个极限情况, 即当子分层次 l 大到使得 $8 \times 4^l > n_f$ 时会出现 $m_p < 1$, 这似乎说明为单个查询点定位时所需访问的三角形个数可以小于 1, 然而实际情况是, 当前的启发式策略在执行每次查询时至少要对一个三角形进行重心坐标计算和判断, 这就是说, 当 l 增加到一定程度时, 继续增加 l 对降低搜索时间 T_h 毫无意义. 鉴于此, 我们规定 l 必须满足不等式 $8 \times 4^l \leq n_f$, 于是得到 l 的取值范围如下:

$$0 \leq l \leq \frac{1}{2} (\log_2 n_f - 3) \tag{8}$$

上式中, 为了保证 l 的有效取值, n_f 至少必须取 8, 对于 $n_f < 8$ 的球面网格, 启发式策略虽然也能够处理, 但是本节性能分析结果对其不再有效.

综合式(3)~(7), 我们可以得到启发式点定位策略的期望运行时间的计算公式如下:

$$T \leq \frac{a_0}{3} (4^l - 1) + 8 \times 4^l (b_0 + b_1 \sqrt{n_f}) + c_0 n_v (8 + l) + d_0 n_v \left(\sqrt{\frac{n_f}{2 \times 4^l}} - 1 \right) \tag{9}$$

由于 a_0, b_0, b_1, c_0, d_0 都是常系数, 只有 l, n_f, n_v 是变量, 因此启发式点定位策略的期望时间复杂度为

$$O \left(4^l \sqrt{n_f} + n_v l + n_v \sqrt{\frac{n_f}{4^l}} \right) \tag{10}$$

上式中, n_f 和 n_v 是事先给定的, 需要选择的是子分层次 l , 显然, l 的取值应该使得时间复杂度达到最小值.

于是, 我们令 $f(l) = 4^l \sqrt{n_f} + n_v l + n_v \sqrt{\frac{n_f}{4^l}}$, 则导数方程 $f'(l) = 0$ 的实数解就是满足上述条件的最优值. 我们

将 $n_v = n_f^\varepsilon$ 代入, 并令 $n_f \rightarrow \infty$, 同时考虑到式(8), 则可求得 l 的最优取值如下:

$$l = \begin{cases} \frac{\varepsilon}{3} \log_2(n_f), & \varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f} \\ \frac{1}{2} (\log_2 n_f - 3), & \varepsilon \geq \frac{3}{2} - \frac{9}{2 \log_2 n_f} \end{cases} \tag{11}$$

将式(11)代入式(10)后, 可以得到启发式点定位策略的理论时间复杂度:

$$\begin{cases} O \left(n_f^{\frac{1+\varepsilon}{3}} \right), & \varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f} \\ O(n_f^\varepsilon \log n_f), & \varepsilon \geq \frac{3}{2} - \frac{9}{2 \log_2 n_f} \end{cases} \tag{12}$$

由于在预处理阶段只需构造规则球面网格 S^l , 因此启发式点定位策略所需的额外空间复杂度为

示)应该位于规则网格 S^l 上的同一个三角形区域中, 因此利用第 3.4 节的重心坐标方法从 f_C^e 开始搜索时, 所需访问的三角形个数 m_p 不会超过图 4 中灰白三角形带所包含的三角形数目 m_{\max} , 即 $m_p \leq m_{\max}$. 如果灰色三角形的个数为 m_g , 则根据均匀规则网格的特性可知, $m_{\max} = 2m_g - 1$ 且 $(m_g)^2 = \frac{n_f}{8 \times 4^l}$, 综合上述 3 式可得 $m_p \leq \sqrt{\frac{n_f}{2 \times 4^l}} - 1$. 因此, 为 n_v 个查询点寻找所属三角形需

$$O(8 \times 4^l) = \begin{cases} O\left(n_f^{\frac{2}{3}\varepsilon}\right), & \varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f} \\ O(n_f), & \varepsilon \geq \frac{3}{2} - \frac{9}{2 \log_2 n_f} \end{cases} \quad (13)$$

表 1 给出了几种定位算法的时间和额外空间复杂度.可以发现:当 $\varepsilon < 1$ 时,Dobkin 等人的算法的时间复杂度和额外空间复杂度最大;当 $\varepsilon \geq 1$ 时,盲目式搜索算法的时间复杂度最大.gn Sundareswara 等人提出的重心坐标算法相比,我们的启发式策略虽然要花费额外的存储空间,但是时间复杂度却要小得多.gn Kirkpatrick 的算法相比,当 $\varepsilon \geq \frac{3}{2} - \frac{9}{2 \log_2 n_f}$ 时,本文的启发式策略与 Kirkpatrick 算法具有相同的空间和时间复杂度.当 $\varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f}$ 时,需要讨论两种情况:(1) 如果 $\varepsilon < \frac{3}{4}$,我们的启发式策略无论是在时间复杂度还是在额外空间复杂度上都要比 Kirkpatrick 算法小一些;(2) 如果 $\frac{3}{4} \leq \varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f}$,启发式策略的时间复杂度是 Kirkpatrick 算法的 $n_f^{\frac{1}{3}}$ 倍,但是对于一般的三角形数目 $n_f \leq 1000000$ 的网格, $n_f^{\frac{1}{3}} \leq 10$ 比起 Kirkpatrick 算法的时间复杂度系数 λ 来说要小得多,因此对于普通的球面网格,启发式策略具有更高的实际运算效率.另外,启发式策略所需的额外空间复杂度也比 Kirkpatrick 算法要小.

Table 1 Theoretical performance of several point location algorithms with n_f triangles and $n_v = n_f^\varepsilon$ query points

表 1 在网格三角形和查询点数目分别为 n_f 和 $n_v = n_f^\varepsilon$ 的条件下,几种点定位算法的时空复杂度

Algorithm	Preprocessing stage		Search stage time	The whole time T
	Time	Space		
Scanning algorithm	—	—	$O(n_f^{1+\varepsilon})$	$O(n_f^{1+\varepsilon})$
Dobkin's algorithm	$O(n_f^2 \log n_f)$	$O(n_f^2)$	$O(n_f^\varepsilon \cdot \log n_f)$	$O(n_f^2 \log n_f + n_f^\varepsilon \log n_f)$
Kirkpatrick's algorithm	$O(n_f \log n_f)$	$O(n_f)$	$O(n_f^\varepsilon \cdot \log n_f)$	$O(n_f \log n_f + n_f^\varepsilon \log n_f)$
Sundareswara's algorithm	—	—	$O\left(n_f^{\frac{1}{2}+\varepsilon}\right)$	$O\left(n_f^{\frac{1}{2}+\varepsilon}\right)$
Our algorithm	$\varepsilon < \frac{3}{2} - \frac{9}{2 \log_2 n_f}$	$O\left(n_f^{\frac{1+2\varepsilon}{2}}\right)$	$O\left(n_f^{\frac{2\varepsilon}{3}}\right)$	$O\left(n_f^{\frac{1+2\varepsilon}{3}}\right)$
	$\varepsilon \geq \frac{3}{2} - \frac{9}{2 \log_2 n_f}$	$O\left(n_f^{\frac{3}{2}}\right)$	$O(n_f)$	$O(n_f^\varepsilon \log n_f)$

5 实验结果

原始空间模型来自于 Stanford 大学的图形实验室,球面网格利用 Praun 的方法^[11]对原始模型参数化得到.实验的微机性能如下:1.4GHz Pentium4 PC,256M 内存.

我们利用几种三角形数目不同的球面网格对点定位算法的性能进行测试,其中查询点数目 $n_v = 10000$,且均匀分布于单位球面上: $\left\{(\theta_p, \varphi_p) \mid \theta_p = \frac{2\pi}{100}i, \varphi_p = \frac{\pi}{100}j, i = 1, 2, \dots, 100, j = 1, 2, \dots, 100\right\}$.进行测试的算法包括:盲目式点定位算法、扩展到球面网格的 Sundareswara 算法和本文提出的启发式点定位策略.这里,我们之所以不对 Kirkpatrick 算法进行测试,是因为该算法是针对平面网格提出的,要扩展到球面网格比较困难.另外,在上一节的性能分析中,我们已经对其时空复杂度进行了详细的讨论.

图 5 给出了利用本文算法为 Bunny 球面网格($n_f = 69630$)上一给定点 $p\left(\theta = \frac{\pi}{4}, \varphi = \frac{\pi}{4}\right)$ 定位的过程.在预处理阶段,我们根据式(11)计算 l ,然后利用向下取整公式 $l_o = \lfloor l \rfloor$ 求得最优的子分层次 $l_o = 4$,于是启发式策略首先

构造规则球面网格 S^4 (如图 5(b)所示),然后为 S^4 中的每一个三角形 f_s 根据其重心位置 q 从原始球面网格(a)中 找到一个代表三角形 f_c (具体过程见第 3.2 节);在定位阶段,对于给定点 $p\left(\theta = \frac{\pi}{4}, \varphi = \frac{\pi}{4}\right)$,我们首先从 S^4 的网格 中找到包含 p 的三角形 f_s^p ,然后根据预处理阶段的结果找到对应的代表三角形 f_c^p (在图 5(d)中, f_c^p 就是标号 为 0 的三角形,它包含 f_s^p 的重心点 q),最后我们从 f_c^p 开始,利用第 3.4 节的重心坐标方法找到包含 p 的目标 三角形,图 5(d)中的着色三角形序列 (0,1,2,3)就是相应的启发式搜索路径,它所访问的三角形个数仅为 4.

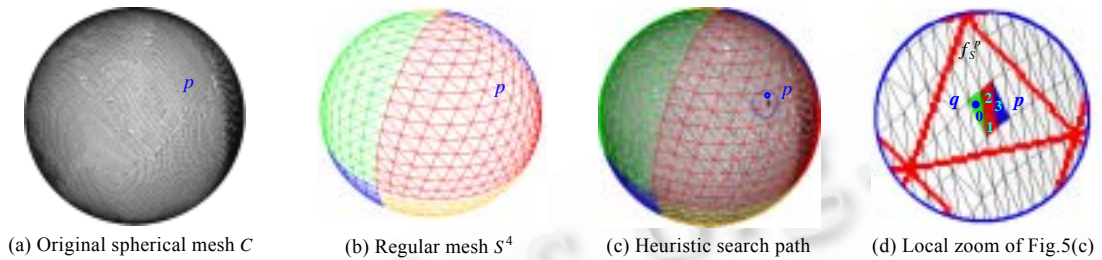


Fig.5 Spherical point location process of heuristic strategy

图 5 启发式策略的球面点定位过程

表 2 给出了 3 种不同点定位算法的运行时间,可以发现启发式策略的计算时间要比其他两种算法小得多. 另外,原始球面网格的三角形分布特性也会影响搜索时间,观察 Gargoyle 和 Tyra 两个球面网格可以发现,两者的 三角形数目虽然相同,但是运算时间并不相同,这主要是由原始球面网格 C 上三角形分布的不均匀性引起的,一 个有效的解决途径是构造顶点分布情况与 C 一致的规则球面网格.

Table 2 Running time of three different algorithms ($n_v = 10000$, T is the whole running time for locating all the query points)

表 2 3 种点定位算法的运行时间($n_v = 10000$, T 表示定位所有查询点的运行时间)

Spherical meshes	Pawn	Cow	Bunny	Venus	Gargoyle	Tyra
Number of triangles (n_f)	304	23 216	69 630	100 000	200 000	200 000
Number of query points (n_v)	10 000	10 000	10 000	10 000	10 000	10 000
Exponent $\varepsilon = \log_{n_f}(n_v)$	1.61	0.92	0.83	0.80	0.75	0.75
Optimal subdivision level (l_o)	2	4	4	4	4	4
Scanning algorithm (T) (s)	8.81	365.61	1 937.52	2 618.78	4 587.51	6 225.21
Sundareswara's algorithm (T) (s)	1.17	4.80	16.95	17.66	28.61	30.31
Our algorithm (T) (s)	0.39	0.66	0.92	0.94	1.30	1.48

6 结论和展望

为了能够有效地解决球面网格的点定位问题,本文提出了一种启发式策略,首先通过构造规则子分网格将 整个球面区域划分成若干具有子分层次关系的查询小块;然后在进行查询之前,根据查询点 p 的位置找到它所 位于的小块作为搜索区域,从而极大地缩小了查询范围;最后在查询过程中,我们利用 p 相对于当前搜索三角形 的重心坐标值来选择下一搜索三角形,从而找到一条从初始搜索三角形到目标三角形的最短查询路径.分析表 明,启发式点定位策略比传统算法具有更优的运算性能.实验结果进一步验证了算法的有效性.

本文虽然提出了一种适用于球面网格的启发式点定位策略,但是有一个假设条件:原始球面网格 C 的顶点 分布均匀,然而实际的网格无法满足这一条件,在最坏情况下,三角形的不均匀分布可能导致启发式点定位算 法的时间复杂度变为 $O(n^{1+\varepsilon})$,因此如何构造顶点分布与 C 一致的规则子分网格是将来的一个研究方向.另外,当前 算法仅适用于球面网格,如何将其推广到任意的空间三维网格亦是将来的一个研究方向.

References:

[1] Dobkin D, Lipton RJ. Multidimensional searching problems. SIAM Journal on Computing, 1976,5(2):181-186.

- [2] Kirkpatrick DG. Optimal search in planar subdivisions. *SIAM Journal on Computing*, 1983,12(1):28–35.
- [3] Sundareswara R, Schrater PR. Schrater: Extensible point location algorithm. In: *Proc. of the Int'l Conf. on Geometrical Modeling and Graphics*. London, 2003. 84–89.
- [4] Mucke E, Saias I, Zhu B. Fast randomized point location without preprocessing in two and three dimensional delaunay triangulations. In: *Proc. of the 12th Annual Symp. on Computational Geometry*. Philadelphia: ACM Press, 1996. 274–283.
- [5] Certain A, Popovic J, DeRose T, Duchamp T, Salesin D, Stuetzle W. Interactive multiresolution surface viewing. In: *Computer Graphics, Proc. of the SIGGRAPH'96*. New Orleans: ACM Press, 1996. 91–98.
- [6] Zorin D, Schroder P, Sweldens W. Interactive multiresolution mesh editing. In: *Computer Graphics, Proc. of the SIGGRAPH'97*. Los Angeles: ACM Press, 1997. 259–268.
- [7] Schroder P, Sweldens W. Spherical wavelets: Efficiently representing functions on the sphere. In: *Computer Graphics, Proc. of the SIGGRAPH'95*. Los Angeles: ACM Press, 1995. 161–172.
- [8] Labsik U, Kobbelt L, Schneider R, Seidel HP. Progressive transmission of subdivision surfaces. *Computational Geometry*, 2000, 15(1):25–39.
- [9] Kobbelt L, Vorsatz J, Labsik U, Seidel HP. A shrink wrapping approach to remeshing polygonal surfaces. In: *Computer Graphics Forum, Proc. of the EUROGRAPHICS'99*. Milan, 1999. 119–130.
- [10] Andrade MVA, Barros WF, Stolfi J. An exact algorithm for point location on spherical maps. In: *Proc. of the IV Brazilian Symp. on Geoinformatics—GEOINFO 2002*. Minas Gerais, 2002. 99–107.
- [11] Praun E, Hoppe H. Spherical parametrization and remeshing. In: *Computer Graphics, Proc. of the SIGGRAPH 2003*. San Diego: ACM Press, 2003. 340–349.

SPW/Prosim 2006 国际研讨会

征文通知

随着信息技术的发展,软件的开发方法和质量管理面临越来越多新的挑战。快速、高质量地交付软件产品、有效地进行风险控制一直软件工程追求的目标。新的软件过程技术以及软件复用、软件演化等技术日益发展并受到重视。2006年,两个成功的国际研讨会 SPW 和 ProSim 将首次携手共同探讨这些问题。会议将于 2006 年 5 月 20 - 21 日,在中国上海,与 ICSE2006 同时召开。

会议的主题是“软件过程变革——应对挑战”(Software process change – meeting the challenge)。会议内容包括:世界著名软件过程领域研究者的特邀报告、针对软件过程挑战与解决方法的论文报告、工具演示、关于软件过程研究方向的专题讨论会。

一、征文范围(包括但不限于)

会议将由 Springer LNCS 出版正式论文集,优秀论文还将推荐到国际重要学术刊物 *International Journal of Software Process: Improvement and Practice*。

欢迎有关软件过程的经验、描述和方法等各相关研究领域的论文。例如,过程内容(文档驱动的、变化驱动的、体系结构驱动的、风险驱动的、涉众驱动的...);过程表示与分析;过程工具和度量;过程中的人为因素;过程建模;过程模拟等等。

二、征文要求

1. 论文未被其他会议、期刊录用或发表;
2. 论文需用英文书写,长度为 8 页(含 8 页)以内;
3. 投稿采用本会议电子投稿系统,格式为 PDF 或 Word 文档;
4. 详见会议主页: <http://www.cnsqa.com/~spwprosim2006>, <http://www.iscas.ac.cn/~spwprosim2006>。

三、重要日期

论文提交:2006 年 1 月 6 日

录用通知:2006 年 2 月 17 日

最终论文:2006 年 3 月 12 日

四、联系方式

联系人:舒风笛

电话:+86-10-62612440

传真:+86-10-62550138

电子邮件: spwprosim2006@iscas.ac.cn

地址:北京中关村南四街 4 号,中国科学院软件研究所