

# 基于 MMP 三角曲面测地线算法研究

周 刚<sup>1)</sup> 邬义杰<sup>2)</sup> 宋德玉<sup>1)</sup> 李 岸<sup>1)</sup>

<sup>1)</sup>(浙江科技学院机械与汽车工程学院, 杭州 310023) <sup>2)</sup>(浙江大学, 杭州 310027)

**摘 要** 测地线的计算在计算机图形处理等方面有着广泛的应用。采用基于 MMP (Mitchell, Mount, Papadimitriou) 方法, 实现了三角曲面上测地线的计算, 修正了 Vitaly Surazhsky 等采用的测地线算法中的误差。该方法首先在窗口传播上摒弃了原有的近似结束条件, 采用光源射线法。特别在窗口相交处理过程中采用多种情况的分层枚举, 补充了 Vitaly Surazhsky 讲述的单一情况, 窗函数多交点时的测地线偏差情况, 并且提供简洁的回溯方法。实验结果表明, 该方法所需时间相当于 Vitaly Surazhsky 算法, 可以代替 Vitaly Surazhsky 采用的算法。

**关键词** 测地线计算 MMP 算法 窗函数 三角曲面

**中图法分类号:** TP391.6 **文献标志码:** A **文章编号:** 1006-8961(2010)08-1260-09

## Geodesics on Triangular Meshes Surface Based on MMP

ZHOU Gang<sup>1)</sup>, WU Yijie<sup>2)</sup>, SONG Deyu<sup>1)</sup>, LI An<sup>1)</sup>

<sup>1)</sup>(Mechanical & Automotive Engineering Department, Zhejiang University of Science and Technology, Hangzhou 310023)

<sup>2)</sup>(Zhejiang University, Hangzhou 310027)

**Abstract** Geodesics are used widely in computer graphics and other field. MMP is a sophisticated method while geodesics are calculated on triangular meshes. We achieved a precision geodesics algorithm based on MMP, and modified the error which occurred when adopting the Vitaly Surazhsky's method. At the window propagating, lamp-house-radial method was used instead of using an approximate type. In dealing with window intersection, the complex and multiplex sorts were treated with layered enumerate type. A clear and comprehensible algorithm used in backtracing. At last, the result was presented, the run time is little more than Vitaly Surazhsky's to some exerd, but the proposed method is precise.

**Keywords** geodesics, MMP algorithm, window function, triangular meshes

## 0 引 言

测地线计算在计算机图形学、图像处理、计算几何和计算机视觉等方面有着广泛的应用。如利用测地线计算分割网格曲面, 在曲面的变形、映射, 曲面图形绘制、边界重构, 机器人行走和导航等方面都有着丰富的应用。

曲面上测地曲率定义为给定弧长参数曲线的曲率在运动标架单位法向量和切线的正交方向的投影, 并认为在曲面上测地曲率为零的曲线称为测地

线, 可以由基于标架运动的测地线方程求解。而对于多面体曲面, 特别是三角曲面, 关于测地线的算法相当丰富, 基本定义为多面体曲面上最短的路线为测地线, 其后关于测地线求解的算法大都基于这一思想。算法研究大都采用最短路径算法 (Dijkstra), 其中, 文献[1]采用窗口传播的方法, 一般称为 MMP (Mitchell, Mount, Papadimitriou) 法 (算法时间复杂度为  $n^2 \log n$ ), 文献[2]把 Voronoi 图算法引入到测地线的计算中, 文献[3]又把 MMP 算法时间复杂度降低为  $O(n^2)$ 。文献[4]采用在三角面片中添加相应边来不断分解三角面, 通过迭代的方法达到给定

**基金项目:** 国家自然科学基金项目 (50575205); 浙江省自然科学基金项目 (Y105686)

**收稿日期:** 2008-12-08; **改回日期:** 2009-03-10

**第一作者简介:** 周 刚 (1980—), 男, 浙江科技学院机械工程专业博士研究生。主要从事数控技术、STEP-NC 方向及曲面加工研究。

E-mail: zghly@zju.edu.cn

的精度,文献[5]同样采用了类似的方法,该类方法最大的问题在于初始迭值需处于最终正确解的附近区域,否则将引起测地线计算结果的不可修正。文献[6-7]应用变量方式计算近似测地线,算法时间复杂度降低到 $n \log n$ ,但是当测地线通过脐点时处理方法复杂,并可能带来较大误差。文献[8]采用MMP方法实现了测地线的精确和近似计算,但是其在窗口传播和窗口间求交及判断时并不能够实现完全精确,在特别情况下甚至会导致测地线求解的偏移。

本文采用基于MMP方法求解精确的测地线,能够在复杂形状的三角曲面上求解精确的测地线,该方法在窗口相交求解时的划分种类较多,但是其分类主要是判断运算,相对于文献[8]在算法的执行时间上并没有带来较多的开销,但却保证了测地线计算的精度。

## 1 MMP 算法概述

### 1.1 算法思想

MMP算法<sup>[6]</sup>是基于点光源光线直线传播原理求解三角曲面上的测地线,其过程为:首先展开三角

曲面(图1(b)为图1(a)空间三角面的平面展开,由于展开的曲面在同一平面内,3维空间问题就转化为平面问题),然后以测地线的起点为光源,起点的两条相邻边为约束,照向远方。当照射过程中遇到顶点的阻挡,光线束一分为二(如图1中所示,光线经过C点折射分为I区和II区),并且根据阻挡顶点是否可折射性质决定是否开始以该阻挡顶点为光源进行新的光线传播。每一条被照耀到的边上都有一个窗函数表示光线到达当前边的距离(如图1(b)中边上的弧形阴影部分,该函数包含了光源到该窗函数所需的距离数据和窗函数在该边中的位置数据,窗函数定义在1.3中给出)。当光线在照耀过程满足以下两个条件之一,传播结束。1)光线照射到结束顶点所在面的任一边上;2)所传播光线的长度超过给定的值,初始给定值由离散曲面的边为路径采用最短路径算法得到(Dijkstra法)。当所有光源传播的光线照射完毕时,再从测地线结束点开始根据各边的窗函数数据以路径最短条件回溯到相邻三角面的边界,直至测地线的开始点,如图1(c)所示。最后在空间中连接各边的回溯点,三角曲面上的测地线得到求解,如图1(d)所示。

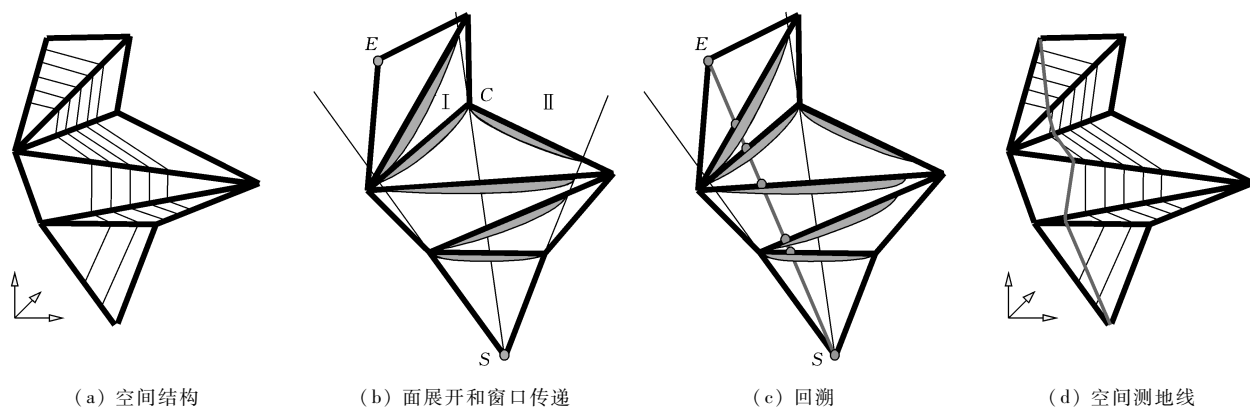


图1 基于MMP的测地线计算原理

Fig. 1 The principle of geodesic based on MMP

### 1.2 可折射顶点的判断

MMP算法的关键是窗函数的计算和光线的传播。由于光线是直线传播,所以在传播之前必须对三角曲面进行展开,也就使得光线在平面区域内传播。光线在传播过程中,如果遇到以下两种顶点时需要在顶点进行折射传播,即1)边界点;当点位于边界上时,2)当该顶点所有相邻面的内角和超过 $2\pi$ 时(如图2所示),称之为衍生顶点。上述两种情况的顶点称为可折射点。

### 1.3 窗函数定义

对于每一个窗函数,数据结构定义为 $W(b_0, b_1, d_0, d_1, \lambda)$ 。如图3所示,该窗口的所有数据为非负数, $\lambda$ 为起点传播到该窗口光源位置需要经过的距离,称之为光源值; $b_0, b_1$ 为在半边方向上的窗口大小, $b_0, b_1 \in [0, \|e\|]$ 其中 $e$ 为该窗口所在的边; $d_0, d_1$ 为当前窗口光源分别到 $b_0, b_1$ 的距离, $b_0, b_1$ 与 $b_1 - b_0$ 构成三角形或在同一直线上,且 $|b_0 - b_1| \leq b_1 - b_0 \leq |b_0 + b_1|$ 。

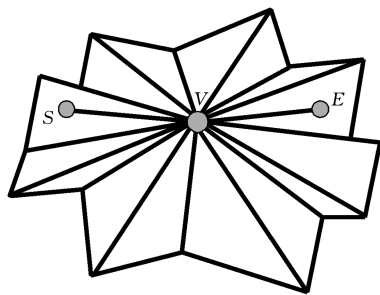


图 2 衍生点的表示  
Fig. 2 Expression of propagation vertex

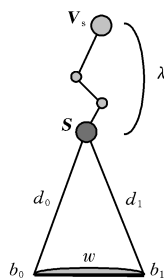


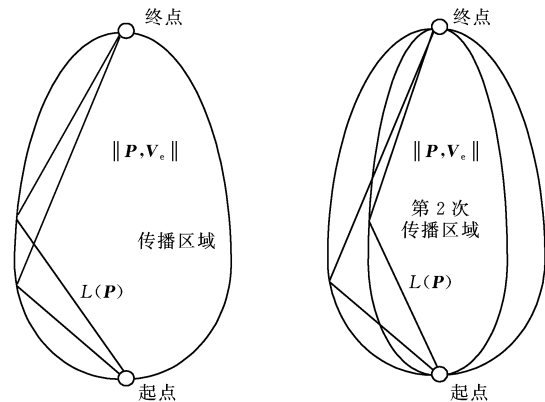
图 3 基于 MMP 的窗口定义  
Fig. 3 Definition of window in MMP

2 精确测地线算法

2.1 窗口传播算法

MMP 算法中对于窗口传播是采用“连续的 Dijkstra”算法。参照基于边的 A\* 启发式搜索的方法,与以往的测地线计算相比,其速度很快。本算法在用于比较的最短路径的长度上做了相应地修改,特别是当基于边的计算得到的最短路径与实际测地线相差较多时,可以减少窗口的传播范围(图 4 所示),图中,  $D(s)$  是用于比较的最短路径长度;  $L(p)$  是开始点到当前窗口上一点  $P$  的曲面距离;  $\|P, V_e\|$  是终点  $V_e$  到窗口上一点的直线距离。算法步骤如下:

- 1) 根据边的长度,利用离散的最短路径算法,计算起点到终点的最短路径长度  $D(s)$ 。
- 2) 根据起点开始传播,其中允许继续传播的条件为  $L(p) + \|P, V_e\| < D(s)$ 。当不满足该条件时,该支窗口光线传播结束。
- 3) 当窗口传播到的顶点为边界点和衍生点时,比较该顶点当前的光源值与先前的光源值的大小,如果目前的光源值小于先前的光源值,把先前光源



(a) 文献[8]的 A\* 算法 (b) 改进后的 A\* 算法

图 4 改进 A\* 启发式搜索

Fig. 4 Approval search method based on A\*

值赋为当前光源值,等当前窗口传播完毕,则开始以该顶点为光源的窗口传播。

4) 当窗口传播到终点所处的平面,则该窗口传播完毕,并计算终点到该窗口的测地距离  $L(s) = \min\{L(P) + \|P, V_e\|\}$ ,如果  $L(s) < D(s)$ ,则  $D(s) = L(s)$ 。

5) 当所有递归传播结束后,从终点开始进行回溯,并计算得到回溯路径。

该算法效率的关键是步骤 2),经过该判断,可以极大地缩小窗口传播的范围,步骤 4)在步骤 2)基础上,特别是  $D(s)$  与最终的  $L(s)$  相差较大时,可以部分地减少传播路径。

2.2 基于点光源的窗口传播策略

光线的传播由几个相关的参数组成,分别为光源点、左右射线、传播面,传播边(如图 5(a)所示),也就是随着给定上述参数,光线就能够根据既定的条件传播,最终投影于半边,形成窗函数。如图 5(b),为了方便计算,根据传播边对应的三角面顶点相对于左右射线的位置不同,把光线的传播分为 3 类:左传播、右传播和中心传播,所有光线传播的窗口计算都包含在这 3 类之中。在中心传播中,如果该顶点为衍生顶点或边界点,则计算该次传播到达该顶点的距离,并且与原来的距离比较,如果该次传播所需的距离较少,则存储该距离为衍生顶点的光源值,称之为衍生顶点的光源距离刷新。对于起点和衍生顶点开始传播光线时,其侧面需要补偿窗口,并且在同一面中的另外两个顶点如果是衍生顶点或边界点,需要刷新该点的光源值(如图 5(c)所示)。

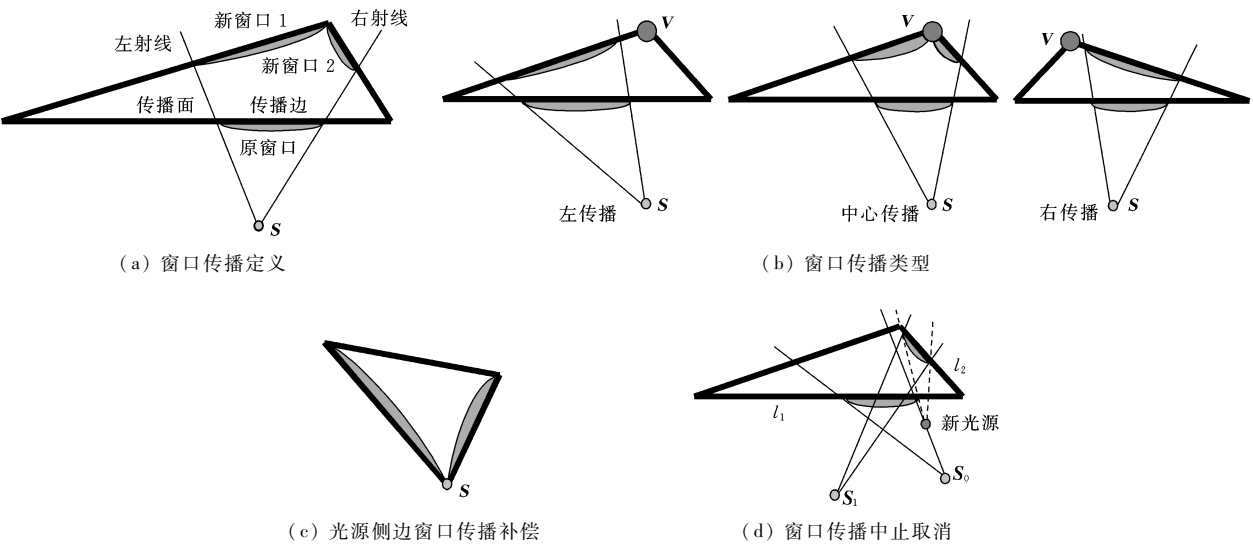


图 5 窗口传播的定义和类型

Fig.5 Definition and type of window propagation

光线的传播是根据左右射线确定,而不是窗口确定,因为在传播过程中,在特定的一条边中,由于在该次传播之前,已经存在窗口,而且存在的窗口优于目前的窗口,按照文献[8]的方法,该次传播就此结束,但是对于下一条边,该次传播就可能优于前次的传播,如图 5(d)所示,在  $l_1$  上,光源  $S_1$  所传播的窗口劣于  $S_0$  传播的窗口,但是在  $l_2$  上由于  $S_0$  传播到该边需要折射,所以在  $l_2$  上,光源  $S_1$  所传播的窗口存在优于光源  $S_1$  所传播的窗口的情况。当传播类型为中心传播时,由于曲面是沿着边按照原来的平面进行旋转得到,所以光线需要分为两支不同的路径传播。

2.3 相交窗口处理

在光线的传播过程中,由于新光源的产生和传播的路线不同,经过传播以后到达某一边的窗口可能有多个,并且存在相交的情况。当两窗口相交时,需要保留到达该边时的经过路径较少的窗口,并且当一个窗口不能完全覆盖另外一个窗口时,需要区分窗口的区域。相交窗口处理是窗口传播的关键。

2.3.1 相交窗口的数学表达

在同一条边上,当两个窗口相交时它们的参数  $[b_0, b_1]$  范围的交集非空。在文献[8]中,其窗口的相交判断做了近似处理,其处理方法为:令该边为  $x$  轴,原点为该边的起点,建立平面直角坐标系,根据窗函数参数,可以确定两窗口各自光源在该坐标系中的位置坐标(横坐标为非负),然后根据窗函数的参数求出  $x$  轴上一点到两窗口光源距离相等的点

$(P_x, 0)$ ,根据该等距点的位置划分相交窗口的覆盖区域。该方法对于窗口相交的命题是完全正确和充分的,但是在求解过程中,却做了极大的简化,只是考虑较多情况中的一种可能,所以在较复杂的条件下,其测地线求解的结果可能发生偏移(如图 6 所示)。其原因是由于错误划分窗口区域而导致该边中有部分区域的光照距离变长,而当该距离变长区域是实际测地线通过的区域时,就出现了如图 5 所示的结果(原因如图 7 所示)。

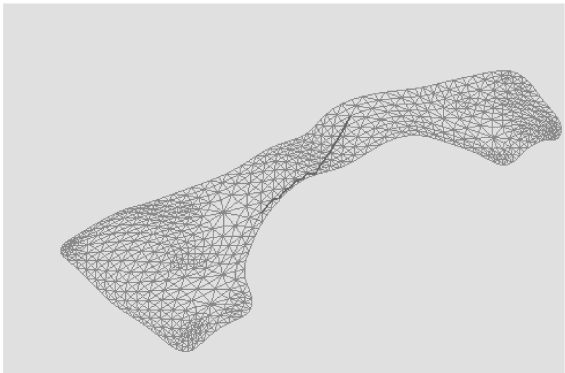


图 6 窗口相交简化出现的测地线偏差

Fig.6 Geodesic error if simpler of windows intersection

相交窗口的区域划分点实际上是  $x$  轴上到两光源的在直角坐标系中的距离差等于光源值差的点。也就是以两光源为焦点,光源值差为定值的双曲线和  $x$  轴的交点(如图 8 所示)。那么问题就转化为在窗口相交区域内,即为  $x$  轴上的点落在双曲线不

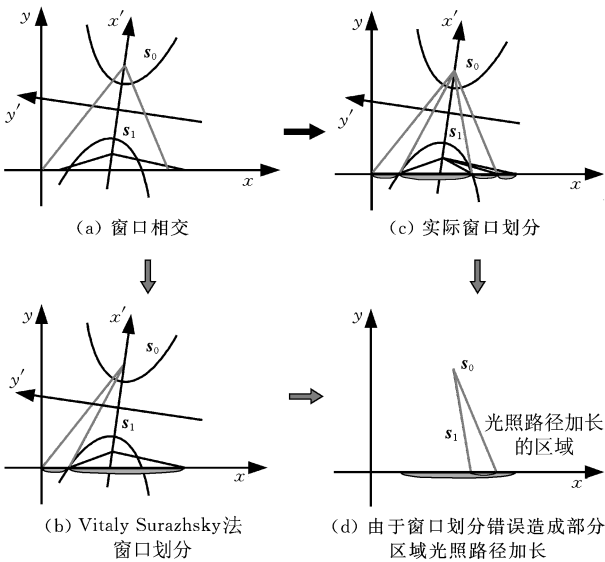


图 7 窗口划分简化引起的光照路径加长

Fig. 7 Longer light length if simpler of windows intersection

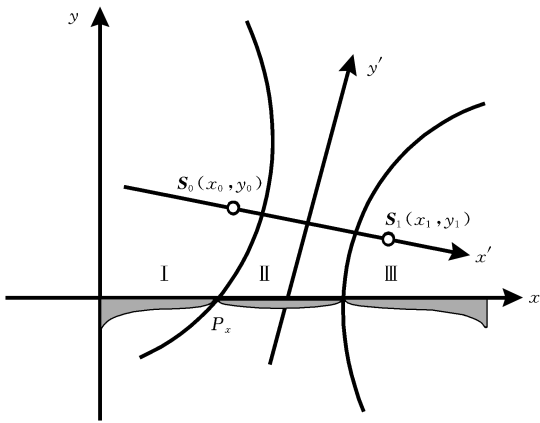


图 8 窗口拥有的区域划分

Fig. 8 Windows region partition

同区域来确定窗口的划分。把坐标平面按照双曲线划分为 3 个区域,如图 8 所示, I 区为光源  $S_0$  区域, III 区为光源  $S_1$  区域,而 II 区则根据光源  $S_0$  和光源  $S_1$  的光源值决定。当  $\lambda(S_0) - \lambda(S_1) > 0$  时, II 区为  $S_1$  区域,否则为  $S_0$  区域。

在计算过程中,首先进行坐标系转换,令向量  $S_1 - S_0$  为  $x'$  方向,  $\frac{S_1 + S_0}{2}$  为坐标原点,建立平面直角坐标系。经过坐标变换,原坐标系  $x$  轴与双曲线的交点( $P_x$ )计算方程表示为  $Ax^2 - Bx + C = 0$ ,其中,  $A = b^2 - a^2k^2$ ,  $B = 2a^2kI$ ,  $C = -a^2I^2 - c^2$ 。 $a, b, c$  为双曲线的参数,  $a = \|S_0 - S_1\|$ ,  $c = \|S_1 - S_0\|/2$ 。 $k$  为向量  $S_1 - S_0$  斜率的相反数。 $I$  为转换坐标系原点与  $x$

轴交点距离的相反数。

但是在窗口相交过程中,并不是所有情况下,上述计算方程都适用,其中两种情况需要另外判断:

1) 当双曲线的参数中  $a > c$  时,也就是说在根据边创建的坐标平面中,不存在到两光源点的距离等于两光源值的差的绝对值。这种情况只需取光源值小的窗口为完整窗口,另一窗口根据相交情况确定。

2) 当  $x$  轴和双曲线没有交点(判别式小于零),也就是说在  $x$  轴位于双曲线两条渐近线中间时,这种情况的判定与第 1 种相同。

所以如果简单的根据二次方程求解结果作为相交窗口的区域划分是缺乏足够精确性。

2.3.2 窗口相交类型分类与区域划分

在  $x$  轴与双曲线有交的情况下,由于窗口范围、交点  $P_x$  和坐标平面划分的不同,其判断的结果也不相同。对于类型的划分根据影响条件的不同划分为 3 个层次,分别为坐标平面的划分、窗口范围的划分、交点状态的划分。

坐标平面的划分主要依据之前所述的 3 个区域的划分有两种情况,同时考虑两个对应光源在坐标平面中的关系,所以该层为 8 种情况,图 9 表示光源在坐标平面中的 4 种关系。窗口范围划分根据类似于卷积的推移关系,可以表示为 4 种情况,两种相交和两种包含关系,如图 10 所示。交点状态的划分情况较前两层复杂,其又可以分为单侧相交和双侧相交。这是针对直线和双曲线相交关系定义的,当直线和双曲线中的一支相交时为单侧相交,与两支分别相交为双侧相交,其判断由所求得的交点在转换坐标系中的  $x$  值确定,如图 11 所示。其中双侧相交的判断类型根据窗口范围的划分类型的不同而不同,每种窗口相交范围对应 3 种情况。单侧相交同样与窗口范围划分相关,其情况对应图 10 中的次序为 3,4,4,3 种。

如图 12(a) 所示。首先  $\lambda(S_0) - \lambda(S_1) > 0$ , 所以 II 区为  $S_1$  区域,只需对左边的双曲线进行处理即可;判断窗口划分范围类型如图 10(c) 所示;然后根据在转换坐标系中交点的  $x$  值判定为双侧相交,属于双侧相交的第 2 种情况,根据最终的判定就可以划分两个相交窗口的控制区域了,如图 12(b) 所示。对于窗口相交类型的分类情况比较复杂,一共达到了 204 种,但是由于采用了分层讨论,一次相交计算所需的判断次数最多为 13 次。而且判断极为简单,其带来的附加时间消耗基本可以忽略。

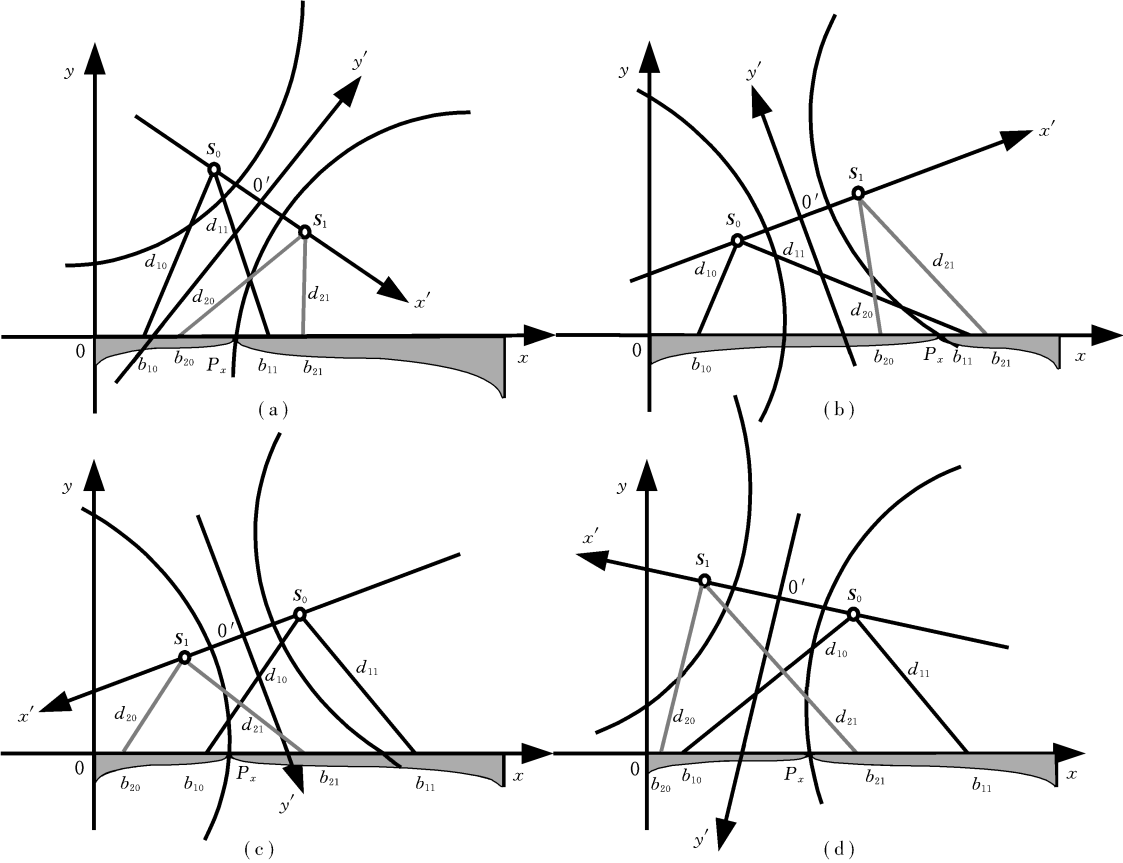


图 9 根据光源在转换坐标系中的位置划分坐标平面类型  
Fig.9 Coordination type partition based on propagation vertex in translation

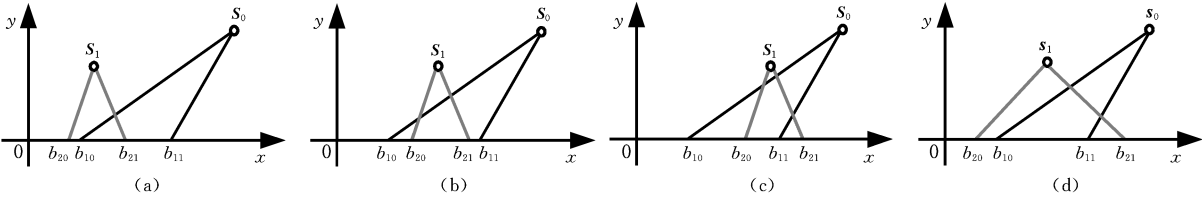


图 10 窗口相交范围划分  
Fig.10 Windows intersection region partition

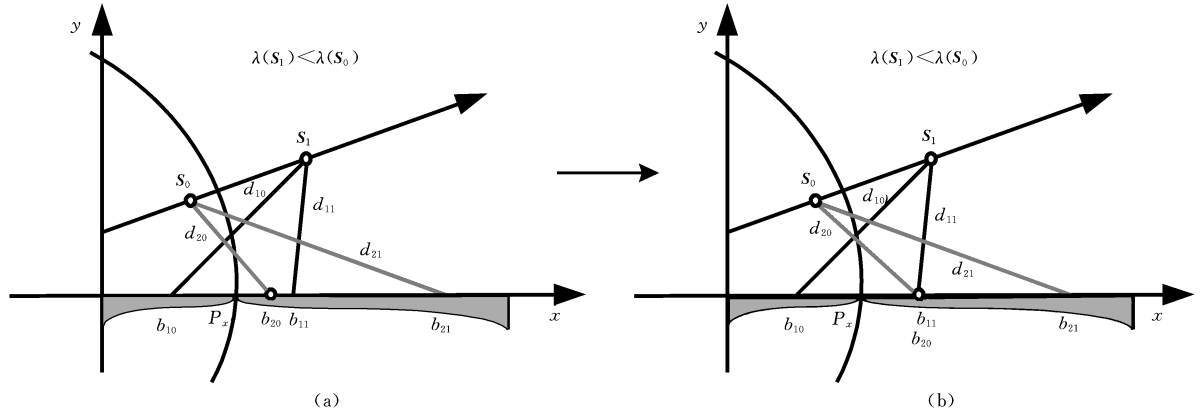


图 11 根据  $x$  坐标位置判断双曲线的相交情况  
Fig.11 Type of hyperbola intersection based on  $x$  axis

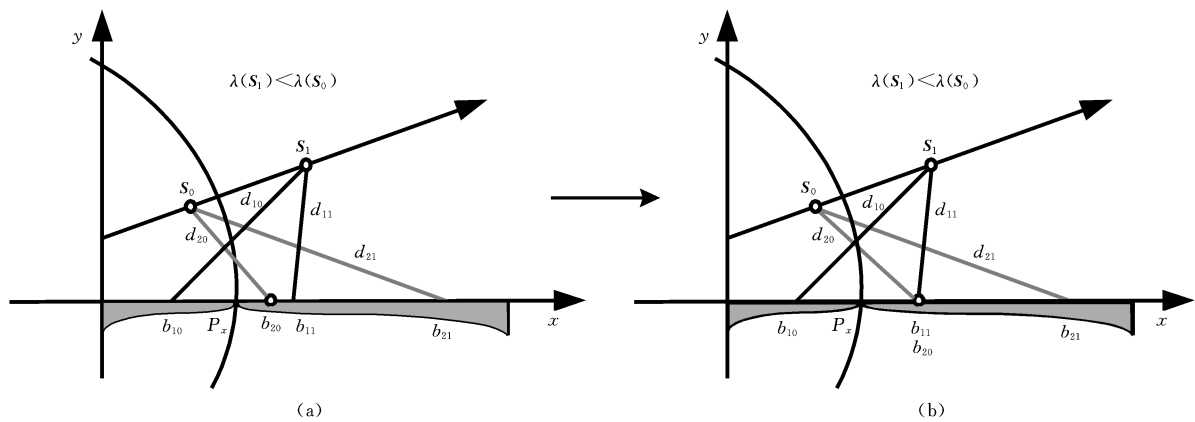


图 12 根据类型分类处理窗口相交情况

Fig. 12 Windows intersection dealing based on their type

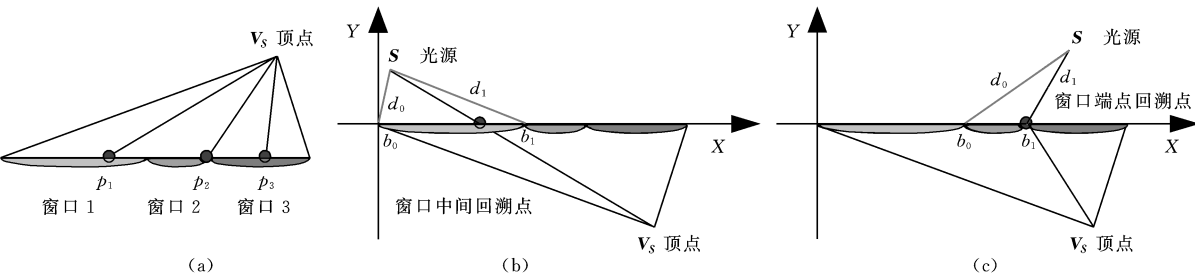
2.4 路径回溯算法

回溯算法实际上是求解在对应边的所有窗口中的一点  $P$ ,使得  $L(s) = \min \{ L(p) + \|P, V_e\| \}$ ,由于根号存在,求解该函数极值比较繁琐,可把它转换为平面上两点在约束条件下的最短距离来解决。

根据 MMP 算法,在同三角平面内,不存在两条线段同为测地线,其回溯点必定位于对边上,把回溯分为两类:1)基于点的回溯,2)基于边的回溯。基于点的回溯只存在一条对边,而基于边的回溯则有两条对边,但是基于顶点的回溯需要判断与该顶点相邻的所有面,而基于边的回溯只需处理当前面。如图 13(a)所示,  $V_s$  为终点,其对边上存在多个窗口,由于在传播过程中,对于窗口相交做了区域划

分,所以窗口之间不相交。通过计算每一个窗口内到终点的最短测地距离,最终比较所有的窗口之间的最短测地距离,选择测地距离最短的窗口中的最短点为该边的回溯点。

单一窗口的最短距离计算如图 13(b)、图 13(c)所示,其窗口分布对应于图 13(a)。把该边对应的三角形顶点位于  $x$  轴下方,窗口光源位于  $x$  轴上方,这样计算在  $x$  轴上一点到这两点的最短距离的关系就显而易见了,只需求解经过三角形顶点和窗口光源的直线和  $x$  轴交点,并比较其所有窗口范围,就可以获取该窗口到  $V_s$  的最短测地距离和对应点。比较该边上所有的窗口回溯距离,取最短的回溯路径即可。其中回溯的时间占用整个算法的时间比例极少。



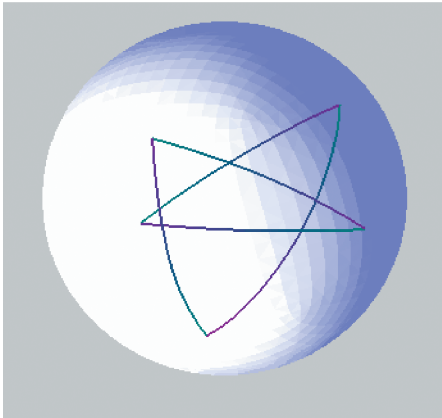


图 14 球面上测地线求解结果  
Fig. 14 Geodesic on sphere

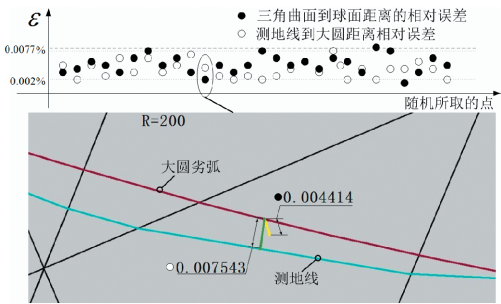


图 15 球面上测地线求解结果  
Fig. 15 Geodesic result on sphere

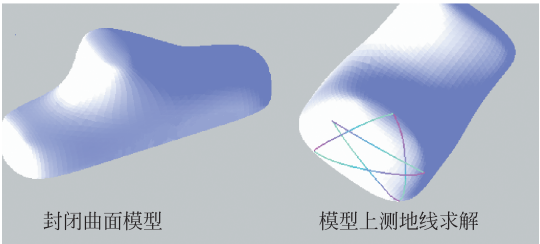


图 16 封闭曲面上测地线求解结果  
Fig. 16 Geodesic result on closed surface

关。表明传播时间的真正决定因素是从起点到终点所需传播面的个数,而不是模型所包含面的个数。与文献[8]的算法时间比较,表明本文的算法相当,只是在判断方面增加了少许时间。图 17—图 19 为各种模型下或不同端点下的测地线求解结果。

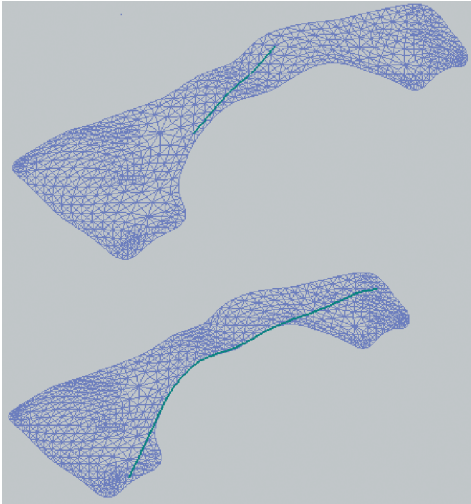


图 17 图 6 所示的图形正确的测地线求解  
Fig. 17 Correct geodesic result on Fig. 6 surface

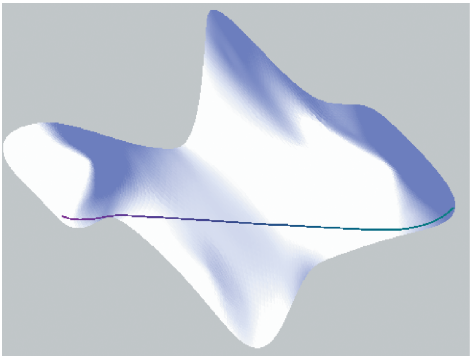


图 18 复杂曲面的测地线求解  
Fig. 18 Geodesic result on Example1 surface

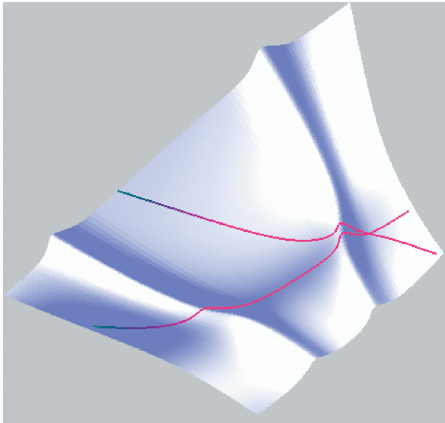


图 19 另一复杂曲面的测地线求解  
Fig. 19 Geodesic result on example 2 surface

表 1 本文方法和文献[8]方法在传播时间和面数、长度关系上的对比

Tab.1 Comparison of relationship on time, surface number and geodesic length between the article and literature

|       |   | 模型面数   | 传播长度   | 传播面数   | 时间/ms  |
|-------|---|--------|--------|--------|--------|
| 本文方法  | 1 | 1 536  | 28.562 | 964    | 186    |
|       | 2 | 1 536  | 12.013 | 337    | 38     |
|       | 3 | 6 114  | 28.317 | 3 451  | 1 481  |
|       | 4 | 98 304 | 28.524 | 42 246 | 95 000 |
| 文献[8] | 1 | 1 536  | 28.571 | 972    | 163    |
|       | 2 | 1 536  | 12.015 | 342    | 36     |
|       | 3 | 6 114  | 28.432 | 3 485  | 1 069  |
|       | 4 | 98 304 | 28.535 | 42 321 | 77 000 |

4 结 论

本文给出了基于 MMP 的三角曲面精确测地线算法,该方法算法时间能够满足测地线计算的要求,具有和文献[8]同等的应用价值,并且保证了不会出现如图 5 的误差偏移情况。是基于 MMP 测地线算法应用的有益补充。

参考文献 (References)

[1] Mitchell J S B, Mount D M, Papadimitriou C H. The discrete geodesic problem [ J ]. Society for Industrial and Applied Mathematics, 1987, 16(4): 647-668.

[2] Baltsan A, Sharir M. On the shortest paths between two convex polyhedral[ J ]. Journal of ACM, 1988, 35(2): 267-287.

[3] Chen Jin-dong, Han Yi-jie. Shortest paths on a Polyhedron[ C ]// Proceedings of the 6th Annual Symposium on Computational Geometry. Berkley, CA, USA: ACM Press, 1990: 360-369.

[4] Kanai T, Suzuki H. Approximate shortest path on a polyhedral surface and its applications[ J ]. Computer-Aided Design, 2001, 33(11): 801-811.

[5] Martinez D, Velho L, Carvalho P C. Geodesic paths on triangular meshes [ C ]//Proceedings of SIBGRAPI/SIACG. Curitiba, Brazilian: IEEE Press, 2004: 210-217.

[6] Kimmel R, Sethial J A. Computing geodesic paths on manifolds [ J ]. Proceedings of the National Academy of Sciences, 1998, 95(15): 8431-8435.

[7] Novotni M, Klein R. Computing geodesic distances on triangular meshes [ C ]//Proceedings of WSCG ' 2002, [ s. n. ]. WSCG Press, 2002: 341-347.

[8] Surazhsky V, Surazhsky T, Kirsanov D. Fast exact and approximate geodesics on meshes [ C ]//Proceedings of SIGGRAPH05. Los Angeles, USA: ACM Press, 2005: 553-560.