

doi: 10.7690/bgzdh.2015.12.014

三维空间中两个三角形求交测试的改进算法

刘 渊, 高玲玲, 陈秀丽
(西南科技大学城市学院, 四川 绵阳 621000)

摘要: 针对三维建模中最常见的三角网格模型, 采用基于代数学的方法来改进常见的基于几何学的三角形与三角形求交测试算法。该算法利用暴力策略中的线性方程组。分别概述常见的和改进的三角形与三角形求交测试算法, 并给出算法的具体实施阶段。结果表明: 与几种经典算法相比, 改进的算法效率有一定优势, 可通过重用不同方程中的公共变量来加快求解的速度。

关键词: 碰撞检测; 求交测试; 基于代数学的方法; 线性方程组; 公共变量
中图分类号: TP206 **文献标志码:** A

Improved Algorithm for Triangle to Triangle Intersection Test in Three Dimensional Space

Liu Yuan, Gao Linlin, Chen Xiuli
(City College of Southwest University of Science & Technology, Mianyang 621000, China)

Abstract: Aiming at the usual triangle grid model in three dimensional modeling, use methods based on algebra to improve usual triangle to triangle intersection test method based on geometry. The algorithm uses linear equations of violence strategy. Separately summarize the usual and improved triangle to triangle intersection test algorithm, then introduces the concrete implementation stage. The result shows that the improved algorithm efficiency is better than several typical algorithms, use public variables in different equations to accelerate resolution speed.

Keywords: collision detection; intersection test; methods based on algebra; linear equations; public variables

0 引言

碰撞检测^[1]是创建具有真实性的三维场景^[2]的必要步骤。其最后也是最关键的一个步骤就是基元间的求交测试^[3]。根据构成三维模型的基元类型的不同, 基元间的求交测试也有不同的类型^[4]。笔者主要针对三维建模中最常见的三角网格模型^[5], 讨论三角形与三角形的求交测试算法的改进策略。具体来说, 就是用基于代数学的方法来改进 Möller、Held、Guigue 与 Devillers 等提出的基于几何学的方法^[6-8], 以提高求交效率。

1 算法和相关原理

1.1 常见的三角形与三角形求交测试算法

确定三维空间中 2 个三角形是否相交的暴力策略^[9]需要解 6 个线性方程组, 每一个方程组对应于一个三角形的边与另一个三角形表面的交叠。文献[6-8]中提出了一种利用 2 个三角形对应平面的交线的改进算法。文献[6]中, Möller 提出了一种依赖于三角形各顶点在这条交线上的标量投影的算法。文献[7]中, Held 论述了一种首先计算一个三角形与交线上的一个线段的交叠再检测该线段与另一个三角形的边的交叠算法。2 种算法中都为了快速、可靠

地获得变量而做了相应的改进。Guigue 与 Devillers 在文献[8]中提出了速度最快的一种算法, 该算法是文献[6]中算法的改进。

1.2 改进的三角形与三角形求交测试算法概述

以上算法都是立足于几何学视角, 而文中的改进算法则基于代数学。算法利用了暴力策略中的线性方程组。算法的核心观点是方程组之间是强关联的; 因此, 可以通过重用不同方程中的公共变量来加快求解的速度^[10]。

设 A 与 B 为三维空间中的 2 个三角形。若 A 与 B 相交, 则一个三角形的一条边与另一三角形的表面交叠。理论上说, 分别测试所有可能的边——三角形组合就可以确定是否相交。不失一般性的, 设 p_1 与 p_2 为三角形 B 上共同一顶点 P 的两条边, q_i ($1 \leq i \leq 3$) 为三角形 A 上由顶点 Q_i 出发的边 (如图 1)。为了找出 p_1 与 p_2 确定的平面与边 q_i 之间的交点, 需求解以下方程组:

$$P + \alpha_1 \times p_1 + \alpha_2 \times p_2 = Q_i + \beta_i \times q_i \quad (1)$$

为了使相交点局限在三角形内, 方程组的解需满足以下约束条件: $0 \leq \beta_i \leq 1$, $\alpha_1, \alpha_2 \geq 0$ 并且 $\alpha_1 + \alpha_2 \leq 1$ 。

收稿日期: 2015-07-22; 修回日期: 2015-09-01

作者简介: 刘 渊(1981—), 男, 四川人, 硕士, 助教, 从事计算机图形学研究。

共需进行 6 次这样的求交测试：三角形 B 的三条边与三角形 A 共需进行 3 次测试，剩下的 3 次是三角形 A 的三条边与三角形 B ；因此，共有 6 个方程组。

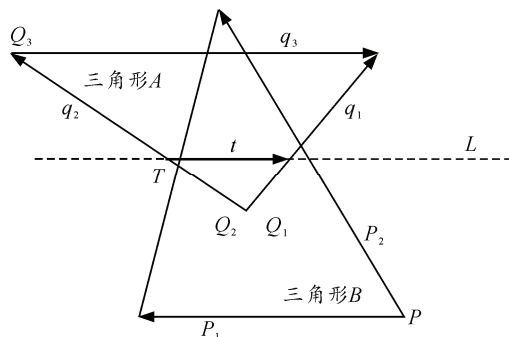


图 1 问题定义

算法的关键是通过重用公共变量和利用矩阵运算的线性来减少算术运算的次数。此外，此算法只需解 3 个方程组，然后利用方程的解通过其他方式获取求交测试的结果。

算法始于求解决定 $\beta_i (1 \leq i \leq 3)$ 的 3 个方程组。这些方程组对应于三角形 A 的三条边与三角形 B 所决定平面的求交测试，见式 (1)。三角形 A 与三角形 B 的角色是任意的(即可以互换)。求解 β_i 不仅是因为本阶段的需要， β_i 的值还可以快速确定哪些组合不会相交，否则， β_i 可以用于构建三角形 A 与 B 所在平面的交叠线段(图 1 中的 t)。很明显，2 个三角形之间的任何交叠都必然会位于这条线段上；因此，问题变成了此线段与三角形 B 之间的平面求交测试。文中的改进算法由以下 5 个阶段组成：

- 1) 利用行列式求出参数 $\beta_i (1 \leq i \leq 3)$ 的值；
- 2) 如果不存在合法的 β_i ，则认为不会发生交叠而退出算法；
- 3) 利用 β_i 构建三角形 A 与 B 所在平面的交叠线段；
- 4) 如果此线段与三角形 B 相交，则 2 个三角形相交；
- 5) 如果需要，利用解出的参数的线性组合构建 2 个三角形的交叠线段。

1.3 算法的具体描述

1) 第 1 阶段。

在执行第 1 个阶段之前，先把式 (1) 改写为矩阵形式。令 $r_i = Q_i - P$ ，则式 (1) 变形为

$$(p_1 \ p_2 \ q_i) \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ -\beta_i \end{pmatrix} = (r_i) \quad (2)$$

式 (2) 中 3×3 矩阵的前两列对于不同的 i 值是相同的。定义矩阵 $A(v) = (p_1, p_2, v)$ ，式 (2) 可以重写为标准形

$$A(q_i)x_i = r_i \quad (3)$$

其中 $x_i = (\alpha_1, \alpha_2, -\beta_i)$ 。

为保证相交点在三角形 A 内部，方程组的解应满足 $0 \leq \beta_i \leq 1$ 。满足此条件的 β_i 称为合法的 β_i 。

方程组是利用行列式来求解的。即

$$\beta_i = -\frac{|A(q_i)|}{|A(r_i)|}$$

$A(q_i)$ 与 $A(r_i)$ 的行列式的值通过按第 3 列展开而求得，这样所有行列式中的子式都是相同的，这是由于所有矩阵的前两列都是 p_1 和 p_2 ；因此，只需计算一次子式即可。当 $i=3$ 时

$$\left. \begin{aligned} q_3 &= q_2 - q_1 \\ Q_3 &= Q_1 + q_1 \\ r_3 &= P - Q_3 = P - (Q_1 + q_1) = r_1 + q_1 \end{aligned} \right\} \quad (4)$$

由于用按列展开的方式计算行列式的过程是线性的， $i=3$ 时的行列式可以用以下子式计算：

$$\left. \begin{aligned} |A(q_3)| &= |A(q_2 - q_1)| = |A(q_2)| - |A(q_1)| \\ |A(r_3)| &= |A(r_1) + q_1| \end{aligned} \right\} \quad (5)$$

由于 $i=1, 2$ 时等式右边的行列式大都已经计算过了， $i=3$ 时的计算量就很小。

总结：第 1 阶段用到的是 2 个方程组而不是 3 个，每个方程组只需计算 1 个变量，子式只需计算 1 次。

2) 第 2 阶段。

若不存在合法的 β_i ，则三角形 A 的所有顶点都位于平面 B 的同侧，也即是说 2 个三角形不会交叠。

若所有 $A(q_i)$ 的行列式均为零，则 2 个三角形位于同一平面上。只需进行平面内的三角形间的求交测试^[6]。

3) 第 3 阶段。

本阶段的目的是构建三角形 A 与平面 B 之间的交叠线段。交叠线段位于 2 个三角形所在平面的交线上，三角形 A 的边与平面 B 交于两点。这两点由 2 个合法的 β_i 给出并位于 $Q_i + \beta_i \times q_i$ 。

在三角形 A 与平面 B 的交叠线段上，用 T 表示

三角形 A 的一条边与平面 B 的交点, 用 t 表示连接 T 与另一个交点的矢量(如图 1)。 T 与 t 可以由 2 个合法的 β_i 、 q_i 及 Q_i 的线性组合来表示。例如, 如果 β_1 与 β_2 都是合法的, T 与 t 可以表示为

$$\begin{cases} T = Q_1 + \beta_1 q_1 \\ t = \beta_2 q_2 - \beta_1 q_1 \end{cases} \quad (6)$$

4) 第 4 阶段。

2 个三角形的相交有 2 种可能性。要么线段 $(T, T+t)$ 至少与三角形 B 的一条边相交, 要么该线段在三角形 B 内部。为了确定该线段与三角形的边是否相交需求解以下方程组:

$$\begin{cases} P + \delta_1 p_1 = T + \gamma_1 t \\ P + \delta_2 p_2 = T + \gamma_2 t \\ (P + p_1) + \delta_3 (p_2 - p_1) = T + \gamma_3 t \end{cases} \quad (7)$$

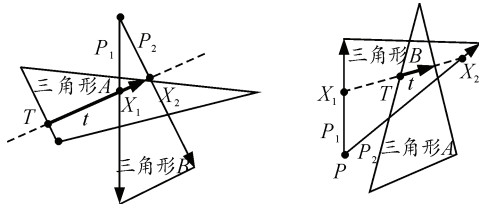
第 3 个等式是针对从顶点 $P+p_1$ 出发的边 $p_3=p_2-p_1$ 。利用行列式来解方程组并利用行列式的线性可以利用类似于式 (5) 的方法节省计算量。

解方程组之后可以验证解集是否合法。首先确定 t 是否与 B 的一条边相交。方程组的解可以确定边由 t 与 p_i 定义的边上的相交点, 相交点局限在边 p_i 内当且仅当方程组的解满足 $0 \leq \beta_i \leq 1$, 此外相交点局限在线段 t 上当且仅当 $0 \leq \gamma_i \leq 1$ 。当 2 个条件都满足时, t 与 p_i 相交。定义 t 与 p_i 的交点为 X_i 。 X_i 的值由等式

$$X_i = P_i + \delta_i p_i \quad (8)$$

给出。其中 P_i 是边 p_i 的起点。例如, 图 2(a) 中 X_1 是 t 与 P_1 的交点, 而 X_2 则是 t 与 P_2 的交点。其中 t 与 P_2 的交叠发生在线段 $[T, T+t]$ ($\gamma_2 > 1$) 的外部, 故 X_2 是不合法的。

剩下的工作就是要确定 t 是否被包含在三角形 B 内部(图 2(b))。这样的情况发生在三角形 B 与 t 所定义的直线在点 T 的两侧都相交的时候; 因此, 只需确定最后一步中能否发现 2 个合法的 δ_i 对应着符号不同的 γ_i 。例如, 图 2(b) 中 X_1 与 X_2 在线段 t 的两侧则交叠线段位于三角形 B 的内部。



(a) t 与 B 的一条边交于 X_1 (b) t 完全包含在 B 内部

图 2 不同的交叠状况

5) 第 5 阶段。

算法的最后一步通过矢量的线性组合及已经计算出的参数获得准确的交叠点。2 个三角形交叠线段的端点位于由 T 与 t 定义的线段上。图 2(b) 中 T 与 $T+t$ 就是所求的交叠线段的端点, 而图 2(a) 中 X_1 与 $T+t$ 为所求端点。一般来说, 两端点为集合 $\{T, T+t, X_i\}$ 的子集, 并可以通过式 (6) 与式 (8) 中描述的线性组合来计算。

1.4 避免除法运算

通常来说, 除法运算需要消耗其他算术运算 4~8 倍的时间。以下将讲解文中所改进算法是如何避免除法运算的。

算法通过 $\beta_i = -\frac{|A(q_i)|}{|A(r_i)|}$ 来获取 β_i 的值。为了使交叠发生在边 q_i 的内部, β_i 需要满足 $0 \leq \beta_i \leq 1$ 。这一条件数学上等价于 $0 \leq \beta_i \cdot |A(r_i)|^2 \leq |A(r_i)|^2$ 。可以通过直接测试是否满足 $0 \leq -|A(q_i)| \cdot |A(r_i)| \leq |A(r_i)|^2$ 来避免除法运算。

检测 γ_i 与算法第 3 阶段的约束条件 δ_i 时也可以用类似的技巧, 用乘法运算来代替除法运算。

β_i 用于计算 T 与 t 。若 β_i 合法, 则 q_i 与另一三角形的平面相交于

$$T = Q_i + \beta_i q_i \quad (9)$$

等式两边都左乘 $|A(r_i)|^2$, 则式 (9) 变形为

$$|A(r_i)|^2 T = |A(r_i)|^2 Q_i + (|A(r_i)|^2 \beta_i) \cdot q_i \quad (10)$$

将式 (10) 中的 $|A(r_i)| \beta_i$ 替换为 $-|A(q_i)|$, 则式 (10) 变形为

$$|A(r_i)|^2 T = |A(r_i)|^2 Q_i + (-|A(q_i)| |A(r_i)|) \cdot q_i$$

这样, 也可以避免除法运算。

2 结束语

用文中所改进算法和 3 种经典算法编写的程序对 3 个模型进行碰撞检测, 并统计各种运算的次数。3 个模型分别为: 一个由 5 000 个三角形构成的花托, 一个由 6 084 个三角形构成的 X 翼战机模型以及 3 752 个三角形构成的茶壶, 每个测试进行 1 万次随机放置和碰撞检测。实验结果如表 1 所示, 文中所改进的三角形与三角形求交测试算法需进行 95~97 次加法、乘法及比较运算, 且不需要进行除法运算。文献[8]中需进行 114~144 次运算, Möller 在文献[6]中的算法虽然也不需要进行除法运算, 运算次数却是 126~148 次; 因此, 文中所改进的算法

在效率上有一定优势。

表 1 不同算法算术运算次数的比较

算法	加(减)	乘	比较	除
Held	74~94	35~45	33~50	1
Möller	54	57	12~28	—
Guigue	62~76	43~52	9~16	—
文中	26~27	56~57	13	—

参考文献:

- [1] 王晓荣. 基于 AABB 包围盒的碰撞检测算法的研究[D]. 武汉: 华中师范大学, 2007: 23-24.
- [2] Canny J. Collision detection for moving polyhedral[R]. Cambridge, MA, USA: Massachusetts Institute of Technology, 1984: 34-36.
- [3] Chazelle B. An optimal algorithm for intersecting three-dimensional convex polyhedral[J]. SIAM Journal on Computing, 1992, 21(4): 671-696.
- [4] 魏迎梅. 虚拟环境中碰撞检测问题的研究[D]. 长沙:

(上接第 51 页)

笔者采取基于浏览器/服务器(B/S)模式建立远程维修保障资源信息系统。客户端基于 Web 方式与服务器进行动态交互。除采用脚本语言, 还通过 COM 对象, 在网页中嵌入 ActiveX 插件, Java Applet 小程序等方式来增强用户与服务器的交互功能^[4-5]。服务器端放置于维修信心支持中心, 包括应用服务器和数据服务器。应用服务器是处理用户决策要求的核心, 根据用户要求向知识库及数据库提取知识和数据, 经处理后以 Web 页面形式传送给用户; 数据服务器用于存放维修保障资源, 以 XML 数据模式存储于数据库中, 以方便用户查找和使用, 数据库数据内容的组织采用 IETM 技术。使用过程中, 系统管理员仅需维护服务器端, 保证数据的格式统一性、数据唯一性和及时更新。

3.3 信息安全技术

设备认证管理软件结合电子钥匙外部设备, 利用存储的用户身份 ID 号、主机编号和加解密密钥等信息, 通过通用的认证算法, 共同完成对使用者的身份认证, 达到设备安全保密的需要。外部的认

国防科学技术大学, 2000: 58-60.

- [5] Terzopoulos D, Fleischer K. Modeling inelastic deformation: viscoelasticity, plasticity, fracture[J]. Computer Graphics, 1988, 22(8): 269-278.
- [6] Moller T. A fast triangle-triangle intersection test[J]. Journal of Graphics Tools, 1997, 2(2): 25-30.
- [7] Held M. ERIT: a collection of efficient and reliable intersection tests[J]. Journal of Graphics Tools, 1997, 2(4): 25-44.
- [8] Guigue P, Devillers O. Fast and robust triangle-triangle overlap test using orientation predicates[J]. Journal of Graphics, 2003, 8(1): 25-42.
- [9] Dobkin D P, Kirkpatrick D G. A linear algorithm for determining the separation of convex polyhedral[J]. Journal of Algorithms, 1985, 6(3): 381-392.
- [10] Cameron S. Collision detection by four-dimensional intersection testing[J]. IEEE Transaction on Robotics and Automation, 1990, 6(3): 291-302.

证模块与主机应用程序采用双向的数据交互。

4 结束语

应用结果表明: 该系统能及时地为现场维修人员提供丰富的维修信息, 把装备维修专家与现场维修人员联系起来, 快速准确地确定故障排除方案, 极大地提高了装备维修效率。当前, 在“以网络为中心”的维修体系建设中, 该系统有力地促进了装备维修的信息化建设。

参考文献:

- [1] 宋建社, 曹小平, 曹耀钦, 等. 装备维修信息化工程[M]. 北京: 国防工业出版社, 2005: 170-189.
- [2] 徐泳龙, 张晓阳, 刘娟. 复杂装备远程服务与维护系统研究[J]. 北京: 计算机集成制造系统, 2006, 12(4): 616-620.
- [3] 张波, 姜震, 孟上. 某型导航雷达便携式维修辅助装置(PMA)设计[J]. 计算机测量与控制, 2013, 21(7): 1855-1857.
- [4] 陈俞龙, 鲁冬林, 王小龙, 等. 基于 Web 的便携式维修辅助系统[J]. 兵工自动化, 2014, 33(3): 56-58.
- [5] 包江奇, 黄炜. 便携式国产大飞机维修辅助设备的设计与实现[J]. 电子技术应用, 2011, 37(1): 28-39.