

非凸多面体碰撞检测简化算法

Simplified Collision Detection Algorithm for Non-convex Polyhedron

臧若兰 曹其新

(上海交通大学机械与动力工程学院, 上海 200240)

摘要: 针对虚拟机器人模型的非凸特点, 根据虚拟机器人的 VRML(virtual reality modeling language)数据格式, 对其中的非凸多面体模型按照 VRML 的标准层次结构分解成相应的子模型, 为每个子模型构造其 AABB 包围盒, 并利用 Vclip 实现 AABB(axis aligned bounding box)包围盒间的碰撞计算, 解决了传统碰撞检测方法处理非凸模型困难的问题。从算法复杂度和存储空间两个方面分析了该算法的实时性和有效性。试验结果表明, 对于处理非凸物体的碰撞检测该文提出的算法比表面凸分解算法效率更高。

关键词: 非凸多面体 碰撞检测 AABB 包围盒 Vclip

Abstract: As for the the non-convex characteristic of the virtual robot model, the paper decompose the non-convex model into sub-model according to the standard hierachical structure of VRML(Virtual Reality Modeling Language) based on the VRML data format of the virtual robot. AABB bounding box is built for every sub-model. Then Vclip was used to do the collision-computation for the AABB bounding-boxes, which solves the difficulty of dealing with non-convex model for traditional collision-detection method. The analyses of the time and storage complexity demonstrates the practical applicability and efficiency. The experiment result shows that the algorithm was more efficient than the convex surface decomposition algorithm dealing with non-convex objects.

Key words: non-convex model collision detection AABB bounding box Vclip

0 引言

机器人与虚拟环境间的碰撞检测是机器人三维仿真的关键问题, 是机器人执行任务和路径规划的基础。在实际的虚拟场景中, 大多数物体均是非凸模型。处理非凸物体的碰撞检测, 一般是将非凸物体进行凸分解, 将物体分解为凸体的集合。凸分解方法分为两种: 一是基于实体的凸分解^[1,2]; 二是基于表面的凸分解。基于实体的凸分解是将非凸物体分解为若干凸实体的集合; 而基于表面的凸分解则是把非凸物体的表面分解成一些凸面片的集合, 对复杂的非凸物体进行凸表面分解, 得到一组凸片集合, 并针对每一凸片构建与其相应的凸块。CHAZELLE 等指出凸分解中如何找到最小个数的凸片集合是一个 NP 问题, 并提出了将一非凸模型分解为较小数目的凸片集合的几种方法。EHMANN 等^[3]以 Chazelle 方法为基础, 提出了一种面向多面体表示物体的凸表面分解方法。采用表面凸分

解技术处理非凸复杂物体的代价较大, 需要较长的预处理时间^[4,5]。

Vclip 算法能处理刺穿情况, 不需要用容错阈值来调整, 而且不会有死循环的现象, 同时由于特例情况少而实现起来更加简单, 是目前所有算法中处理凸体之间的碰撞检测最快速有效的算法之一。本仿真系统中的虚拟机器人模型表达为 VRML 格式, 每个 VRML 模型有 n 个 Shape 节点, 根据 VRML 的标准层次结构将相应的模型分解成 n 个子模块, 对每个子模块构造 AABB 包围盒, 利用 Vclip^[6]实现了 AABB^[7]包围盒间的碰撞检测计算。

1 算法设计

机器人模型很多都是复杂的非凸多面体, 需将其中的非凸多面体进行分解以有效地处理其碰撞检测。根据虚拟机器人的 VRML 数据表达提出一种非凸多面体分解算法, 该算法依据

作者简介: 臧若兰 女, 硕士。研究方向为智能机器人控制、服务机器人三维仿真。

VRML的组织结构将原始模型分解成 n 个子模型,对每个子模型构造 ABB 包围盒,以此作为 Vclip 的核心数据结构 PolyTree 的 n 个节点,将一般非凸多面体间的碰撞检测转化为凸多面体的碰撞检测。图 1 显示了如何由 VRML 模型构造 PolyTree。PolyTree 是表示凸或者非凸几何信息的层次结构,可以是一个简单的凸多面体,也可以表示多个简单的凸多面体组成的复杂模型(包括凹多面体)。Vclip 使用 PolyTree 表示复杂的多面体模型, PolyTree 的叶结点与子模型的 ABB 包围盒相对应。定义好模型的 PolyTree 后, Vclip 就可以有效地处理这个层次结构。

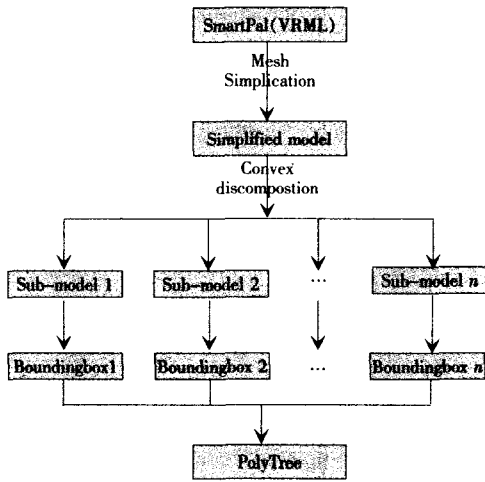


图 1 构造 PolyTree 流程

1.1 模型分解

描述虚拟机器人模型的 VRML 场景图包含许多面集,各面集的结构相同,都对应一个 Shape 节点;每个 Shape 节点描述了一组三角面片,其几何尺寸和外观由 Appearance 和 Geometry 定义,描述了虚拟对象的其中一部分特征。根据 Shape 节点的个数把 VRML 模型相应地分解为 n 个子模块。图 2 显示

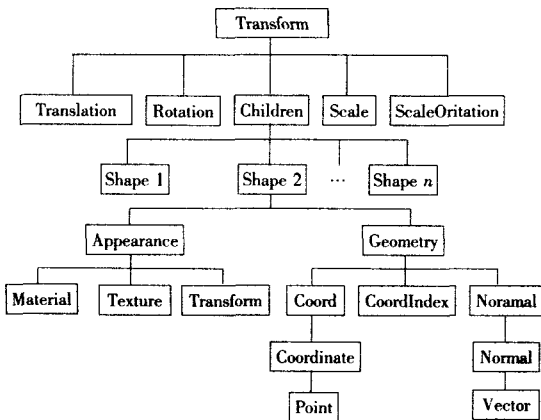


图 2 VRML 模型结构图

了 VRML 模型结构图。

根据 VRML 的标准层次化结构,把每个 Shape 节点作为一个子模型,将一个复杂的非凸体剖分成若干个子模型的集合。

1.2 ABB 包围盒构造

本文采用的 ABB 包围盒是包含子模型且各边平行于坐标轴的最小六面体。通过遍历子模型中的所有顶点,求得顶点的 X 坐标、 Y 坐标和 Z 坐标的最大值和最小值,构造子模型的 ABB 包围盒。描述一个 ABB 仅需 6 个标量。以虚拟机器人模型中包含凹点最多的手指模型为例,手指模型共包含 62 个 Shape 节点,其 ABB 包围盒的 8 个顶点坐标如下: $(X_{max}, Y_{max}, Z_{max}), (X_{min}, Y_{max}, Z_{max}), (X_{max}, Y_{min}, Z_{max}), (X_{max}, Y_{max}, Z_{min}), (X_{min}, Y_{min}, Z_{max}), (X_{max}, Y_{min}, Z_{min}), (X_{min}, Y_{max}, Z_{min}), (X_{min}, Y_{min}, Z_{min})$ 。图 3 是手指模型其中的 4 个节点 ABB 包围盒。

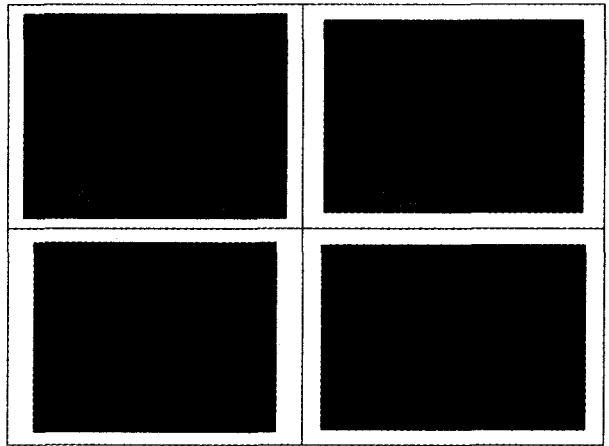


图 3 手指模型其中 4 个子节点 ABB 包围盒

1.3 算法实现

算法分为预处理和动态检测两个阶段。预处理阶段的主要工作是将手指等非凸模型按照 VRML 的标准层次结构分解为相应的子模型,然后为每个子模型构造 ABB 包围盒,利用子模型的 ABB 包围盒构造 Vclip 的 PolyTree 层次结构。动态检测阶段根据物体的运动学得到虚拟机器人和虚拟环境的相对变换矩阵,同时利用对象 PolyTree 和运动变换矩阵计算两个物体之间的距离。

2 算法复杂度分析

下面从 VRML 模型分解和 ABB 包围盒构造的时间开销和对存储空间的要求进行比较分析。

2.1 时间复杂度分析

VRML 模型分解时间复杂度为 $O(n)$,其中 n 为 Shape 节点的个数。构造 ABB 包围盒需要遍历 Shape 节点中所有顶点的坐标。构造单个 ABB 包围盒的时间复杂度为 $O(m_i)$, m_i 为第 i 个 Shape 节点的顶点坐标的个数。基于表面凸分解需要将非凸物体分解为一组凸片集合,对每一凸片构建与其相对应的凸

块,然后构造凸块的 AABB 包围盒;其中分解凸块的时间复杂度为 $O(n_1 \log n_1 + j n_1)^{[8]}$, n_1 为单个 VRML 模型顶点的个数, $n_1 = \sum_{i=1}^n m_i$, j 为模型中凹点的个数。可见时间复杂度远大于 $O(n)$, 并且 Chazelle 等指出凸分解中如何找到最小个数的凸片集合是一个 NP 问题,其复杂度远大于按照 VRML 自身结构分解的复杂度。

2.2 存储空间分析

根据以上的模型分解和 AABB 包围盒构造方法,引入存储需求函数 P :

$$P = \sum_{i=1}^n B I_i + \sum_{j=1}^n B L_j$$

其中: P 为总的内存需求字节数; $B I_i$ 为存储对象 AABB 包围盒所需要的字节数; $B L_j$ 为存储 PolyTree 的面索引多需要的字节数; n 为对象 Shape 节点的个数。

由于每个 AABB 包围盒包含了不止一个三角面片,减少了存储 VRML 模型所需要的字节数,从存储空间上提高了算法的效率。

3 实验结果与对比分析

本文的虚拟场景在 Java3d 平台上实现见图 4,包括虚拟机器人、桌子、杯子。将虚拟机器人各关节 VRML 模型导入到 Java3d 中,环境中的桌子和杯子直接用 Java3d 的类创建。程序预处理阶段对手指模型等非凸物体分别采用本文算法和基于表面凸分解算法进行模型分解,然后构造子模型的 AABB 包围盒,对杯子、桌子、手臂等凸体直接构造其 AABB 包围盒。动态阶段根据虚拟机器人手臂运动学,计算虚拟机器人抓手和杯子的相对运动矩阵,实现端杯的任务。



图 4 虚拟场景和端杯试验

实验是在系统内存 2G, CPU 1.86G 的 PC 机上进行,记录碰撞过程中手指模型不同的面片数 N 与对应的碰撞时间 t 对应关系见图 5。因虚拟机器人手臂在场景中随机移动,其结果有一定的随机性。采用多次测试取平均值减少测试误差,非凸模型在相同的面片数测量 10 次,去掉最大值、最小值,其他 8 次的平均值作为该面片数的碰撞检测时间。实验结果表明,随

着面片数的增加,基于表面凸分解算法所需要的碰撞时间要明显高于本算法的时间。

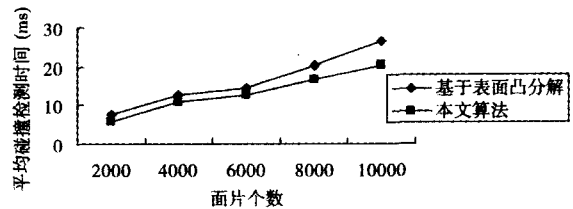


图 5 本文算法和基于表面凸分解算法平均碰撞时间

4 结束语

通过试验数据,比较了本文算法、表面凸分解方法的平均碰撞检测时间。本文算法效率高的原因主要有以下几个方面:

- (1) 提出算法简单、效率高的 VRML 结构模型分解法,应用于非凸体的碰撞检测;
- (2) 对分解后的子模型采用最简单的 AABB 包围盒,建构简单快捷,内存开销少,以提高算法的效率;
- (3) 利用效率高的 Vclip 算法,计算 AABB 包围盒的相交情况。

本文提出的算法简单易行,能有效地处理非凸物体的碰撞检测,程序实现方便,达到了在普通 PC 机上进行实时检测的要求。

参考文献

- [1] CHAZELLE B, DOBKIN D, SHOURABOURA N, et al. Strategies for polyhedral surface decomposition: an experimental study [J]. Computational Geometry: Theory and Applications, 1997, (7): 327-342.
- [2] CHAZELLE B, PALLOS L. Decomposing the boundary of a non-convex polyhedron [J]. Algorithmica, 1997, (17): 245-265.
- [3] EHMANN S, LIN M C. Accurate and fast proximity queries between polyhedra using convex surface decomposition [J]. Eurographics Conference, Manchester, 2001: 500-510.
- [4] 范昭炜, 万华根. 基于图像的快速碰撞检测算法 [J]. 计算机辅助设计与图形学学报, 2002, 14(9): 805-809.
- [5] 范昭炜. 实时碰撞检测技术研究 [D]. 杭州: 浙江大学计算机学院, 2003.
- [6] MIRTICH B. VClip: fast and robust polyhedral collision detection [J]. ACM Transaction on Graphics, 1998, 17(3): 177-208.
- [7] 刘晓东, 姚兰. 一种基于混合层次包围盒的快速碰撞检测算法 [J]. 西安交通大学学报, 2007, 41(2): 141-144.
- [8] 刘海涛. 相容三角剖分及网格优化的算法研究 [D]. 杭州: 浙江大学计算机学院, 2006.