

基于 B+ 树存储的 AABB 包围盒碰撞检测算法

杨帆

电子科技大学 成都 611731

摘要 对于碰撞检测算法,使用传统的 AABB 包围盒来构建包围盒层次树时,其包围盒层次树的层数、叶子节点的个数和各结点的存储字节数是影响碰撞检测效率的主要因素。为了减少结点存储容量对碰撞检测效率的影响,提高碰撞检测的效率,文中采取 B+ 树的存储结构来存储包围盒等信息。在包围盒相交测试之前,使得各结点存储索引有序,不需要再对各结点进行额外的排序,减少了内存开销,并且避免了不必要的包围盒测试。此外 B+ 树的非叶子结点不存储具体的数据信息,从而减少了整棵树的存储空间。实验表明,在检测环境和检测对象相同的条件下,使用 B+ 树存储的 AABB 包围盒碰撞检测算法的检测时间明显比传统的 AABB 算法短。

关键词: 碰撞检测;AABB;层次包围盒;B+ 树;相交测试

中图法分类号 TP391.41

Collision Detection Algorithm of AABB Bounding Box Based on B+ Tree

YANG Fan

University of Electronic Science and Technology of China, Chengdu 611731, China

Abstract For the collision detection algorithm, when using a traditional AABB bounding box to construct a bounding box hierarchy tree, the number of bounding box hierarchy trees, the number of leaf nodes, and the number of bytes stored in each node are the main factors that affect the efficiency of collision detection factor. In order to reduce the impact of node storage capacity on collision detection efficiency and improve the efficiency of collision detection, this paper adopts a B+ tree storage structure to store information such as bounding boxes. Before the bounding box intersection test, the storage index of each node is ordered, no additional sorting of each node is required, which reduces the memory overhead and avoids unnecessary bounding box testing. In addition, the non-leaf nodes of the B+ tree do not store specific data information, thereby reducing the storage space of the entire tree. Experiments show that the detection time of the AABB bounding box collision detection algorithm using B+ tree storage is significantly shorter than the traditional AABB algorithm under the same detection environment and detection object.

Keywords Collision detection, AABB, Hierarchical bounding box, B+ tree, Intersection test

1 引言

碰撞检测在工业制造、虚拟现实、计算机动画和机器人路径规划等应用环境中扮演着至关重要的角色^[1]。碰撞问题在现实生活中无处不在,当两个物体的空间距离接近于 0 的时候,我们认为这个两个物体发生了碰撞。由于碰撞的存在,才使得生活呈现出多样性。在工业制造中,传感器与物体间的碰撞检测是让整个制造流水线稳定有序生产的基础。若物体与传感器发生了碰撞,传感器会记录碰撞点、碰撞深度等,同时触发碰撞后的回调而迅速进行下一步操作。由于制约碰撞检测精度的因素是实时性的,即在很短的时间内完成检测给出反馈,因此如何快速准确地完成碰撞检测对工业制造的稳定性和敏捷性起着巨大的作用。这也是本文研究的主要内容。

目前公认的碰撞检测算法都采用层次包围盒技术来封装存储对象信息。常见的包围盒类型有球包围盒(Sphere)、轴对齐包围盒 AABB(Axis-Aligned Bounding Box)、方向包围盒 OBB(Oriented Bounding Box)和 k-DOPs(Discrete Orientation Polytopes)^[3],图 1 是上述几种包围盒的简单图例,表 1 是它们的综合性能对比。

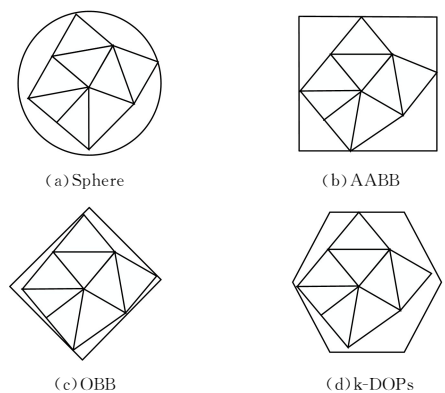


图 1 4 种常见的包围盒类型

Fig.1 Four common types of bounding box

表 1 常见的 4 种包围盒功能的比较

Table 1 Comparison of four common functions of bounding box

包围盒类型	构造难度	紧密程度	更新速度
AABB	较简单	较紧密	快
Sphere	简单	不紧密	不需要更新
OBB	复杂	紧密	慢
k-DOPs	较复杂	紧密	慢

AABB 包围盒是众多碰撞检测算法的首选存储结构,因此研究者们对其进行了各种优化改进让其适应于不同的应用领域。AABB 包围盒采用轴对齐的方式,因此它的构造较为简单,且可以用于刚体和软体的碰撞检测。本文在已有研究的基础上得知,影响 AABB 层次包围盒碰撞检测速度的因素主要有 3 个:AABB 层次树的深度、非叶子结点的个数和叶子结点中存储的几何元素相交测试的次数。对于这 3 个影响因素,国内学者提出了一些优化办法。为了减少 AABB 层次树的深度,潘振宽等^[4]对 AABB 树的叶子结点进行压缩存储,减少了包围盒结点的个数,从而减少了 AABB 树存储所需要的内存空间。为了提高包围盒之间相交检测的效率,沈学利等^[5]提出了混合空间分割和包围盒的碰撞检测算法,该算法根据对象的不同特征来选取不同的包围盒,使得包围盒层次树更加灵活。为了减少叶子结点中基本几何图形的相交次数,Li 等^[6]对 AABB 包围盒的相交测试做出了优化。

众多的结点优化算法都是对传统二叉树的叶子结点进行优化,也即删除叶子结点,而将叶子结点的信息存储到其父结点,但是在删除叶子结点时,每次都要从根结点逐个遍历到叶子结点。若有 n 个叶子结点,那么需要遍历 n 次,因此给算法带来了额外的开销。

基于此,本文采用 B+ 树的结构代替了传统二叉树来存储 AABB 包围盒结点,在不删除叶子结点的情况下,使得存储每个结点所需要的字节数在一定程度上减少,从而使得一次调入 CPU 的结点数增加,减少了 IO 的交换次数,即降低了整个算法所需要的内存空间,提高了碰撞检测的效率。

2 算法实现

2.1 B+ 树介绍

B 树是 1970 年提出的一种平衡的多叉查找树,因其具有高性能的查找能力,后来被定义成一种数据结构,目前被广泛应用在数据库系统和文件索引系统中。B+ 树作为 B 树的一种变形结构^[7],除了每个非叶子结点可以有多个孩子结点外,B+ 树和二叉树的结构一样,拥有根结点、内部结点和叶子结点 3 种类型结点。图 2 是 B+ 树内存结构的简单示意图。

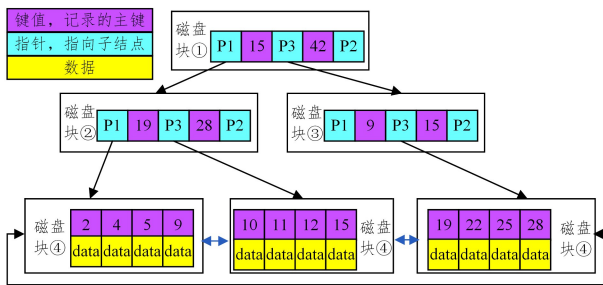


图 2 B+ 树内存结构简单示意图

Fig. 2 Simple diagram of B+ tree memory structure

B+ 树的特征如下:

- 1) 根结点至少包含两个孩子结点(多叉树);
- 2) 非叶结中存在一个索引值,通过索引可以迅速找到存储数据的结点;
- 3) 非叶子结点中的索引值是有序的,结点间的查找迅速;
- 4) 叶子结点中的数据信息构成了一个有序的链表。

由于 B+ 树的性质,非叶子结点不存储具体的数据而只存放索引,因此每一个磁盘块能够存放更多的结点,在数据量足够的情况下,能够明显减少内存与磁盘的 I/O 次数。每个非叶子结点都存储了子节点的地址信息,而叶子结点维护一个有序链表使得数据不需要额外的排序。对 N 个结点的 B+ 树而言,查找某个关键字的时间复杂度为 $O(\log N)$ 。

2.2 基于 B+ 树存储的 AABB 数据结构

2.2.1 构建 AABB 包围盒层次树

将上述 B+ 树的存储结构运用到 AABB 包围盒中,将整个被检测物体构成的 AABB 包围盒作为层次树的根结点,然后采用自顶向下的方法逐层分解根结点的包围盒。采用自顶向下方法构建 AABB 包围盒层次树的关键是:如何将包含物体对象所有信息的根结点划分成最小不相交的子对象,使得生成的整棵包围盒层次树保持平衡。

本文采用基于分裂平面^[8]的方法将根结点的 AABB 包围盒进行划分,为了保持 AABB 包围盒层次树的平衡,如何选择分裂平面的法线轴和分裂点是划分的关键。根据空间立体几何知识,在 3 个维度中选取最大的点组成的轴线在二维投影中一定最长。因此在 x, y, z 坐标轴中选择值最大的点作为包围盒的跨度最大的轴,该轴就是分裂平面的法线轴^[10],然后选取所有投影在法线轴上且值为最大 max 和最小 min 的两个点作为分裂点,将投影分为 S 和 W 两组, $min < x \leq (max + min) / 2$ 的点归入 S 组; $max > x \geq (max + min) / 2$ 的点归入 W 组; $x = (max, min)$ 的点归入含投影点较少的一组。根据 B+ 树的性质易知,每个组内存在一个到孩子结点的索引且各结点间有序排列。

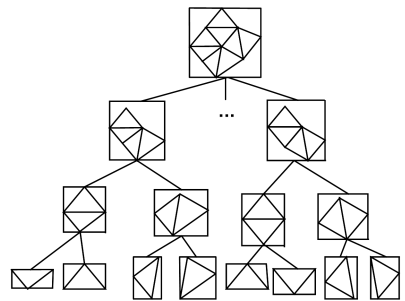


图 3 基于 B+ 树实现的包围盒层次树示意图

Fig. 3 Hierarchical tree diagram of bounding box based on B+ tree

2.2.2 优化 AABB 包围盒层次树的存储结构

在传统的 AABB 包围盒层次树中,每个结点存储着两部分数据信息,分别为孩子结点的索引和该结点的包围盒信息。为了优化各结点的存储结构,Akenine-Möller^[9]提出任何叶子结点的相交测试都可以不用包围盒,即叶子结点不用存储包围盒信息,而叶子结点不存在孩子结点,因此不必使用一个单独的结点表示叶子结点。对于包含 n 个叶结点的二叉树总共有 $2 \times n - 1$ 个结点,在删除叶子结点后使得整棵树只剩下 $n - 1$ 个结点,以这种方式达到减少内存的需求。

本文使用 B+ 树来优化 AABB 包围盒层次树的存储结构,用 B+ 树优秀的索引性能和存储特点来替代二叉树。使用 B+ 树实现的包围盒层次树不再删除叶子结点,而是减少每个结点的存储的字节数,使 AABB 包围盒层次树的

存储空间大大降低。

由于 AABB 包围盒中每个点在法线轴的投影被归为两个不同的组,并且父节点与孩子节点最多存在 6 个不同的范围值。根据包围盒的划分原理,孩子节点的极值(最大值和最小值)是由父节点提供,即父节点中存储着孩子节点包围盒的最大值和最小值,因此父节点和孩子节点信息重复,冗余度高,导致了包围盒层次树的层数增加。采用 1 个字节的标志位使得冗余信息只在父节点中存在一次,这样只需要存储 3

个浮点数(x, y, z)共 12 个字节来表示包围盒信息^[10]。由于包围盒中点的顺序按照分裂点排序存放,因此只需要 1 个字节存放孩子节点的简单索引。表 2、表 3 和表 4 是使用 B+ 树优化后的各结点的存储结构。表中每一位标志位采用二进制 0 或 1 表示, Bxmin, Bxmax, Bymin, Bymax, Bzmin 和 Bzmax 表示该节点的孩子节点的包围盒取值范围; Blchild 和 Brchild 标识该节点的孩子节点是否为叶子节点, 1 表示叶子节点, 0 表示非叶子节点。

表 2 根节点的存储结构
Table 2 Storage structure of root node

包围盒取(3×4=12 字节)									
左子节点索引 (1 个字节)	右子节点索引 (1 个字节)	Bxmin	Bxmax	Bymin	Bymax	Bzmin	Bzmax	Blchild	Brchild

表 3 叶子节点的父节点的存储结构
Table 3 Storage structure of parent node of leaf node

包围盒取(3×4=12 字节)									
左子节点索引 (1 个字节)	右子节点索引 (1 个字节)	Bxmin	Bxmax	Bymin	Bymax	Bzmin	Bzmax	1	1

表 4 叶子节点的存储结构
Table 4 Storage structure of leaf nodes

单个三角面片的包围盒信息(3×4=12 字节)							
Bxmin	Bxmax	Bymin	Bymax	Bzmin	Bzmax	1	1

一棵含有 n 个叶节点的 AABB 包围盒层次树,采用 B+ 树的存储方式优化过后可以节省大约 $32n - (n - 1) \times 15 - 12n = 5n + 15$ 个字节。这样整棵包围树所占的内存空间大大减少,使得一次可以载入更多包围盒结点到 CPU 中,加快了算法的执行效率。

2.3 包围盒间的相交测试

包围盒相交测试是从根节点开始两两比较,若不相交,则两个对象不发生碰撞;若相交,则继续逐层遍历子节点来判断子节点的包围盒是否相交。因此可以说碰撞检测是一个初步求精到精确求交的过程^[11]。图 4 是整个包围盒碰撞检测的流程图;图 5 是包围盒碰撞检测的伪代码;图 6 是在二维平面上两个包围盒发生碰撞的示意图;图 7 是在二维平面上两个包围盒未发生碰撞的示意图。由图 6 和图 7 可知,当两个包围盒发生碰撞时,遍历到叶子节点最终所得到的包围盒是一个相交的包围盒,而当两个包围盒未发生碰撞时,到叶子节点时会得到两个不相交的包围盒。

```
bool sat(polygon a, polygon b){
    for(int i=0; i<a.edges.length; i++){
        vector axis=a.edges[i].direction;//获取对象 a 包围盒的一边
        axis= vec_normal(axis);//标准化法向量轴
        //计算两个对象在该法线轴向量上的投影
        segment proj_a=project(a,axis),proj_b=project(b,axis);
        //如果投影不重叠返回 false,不相交
        if(!seg_overlap(proj_a,proj_b)) return false;
    }
    ...
    //对象 b 也做如此操作
    //遍历对象 a 和 b 所有满足条件的边,只有当投影重叠时才相交,即发生碰撞
    return true;
}
```

图 5 包围盒碰撞检测的伪代码

Fig. 5 Pseudocode for bounding box collision detection

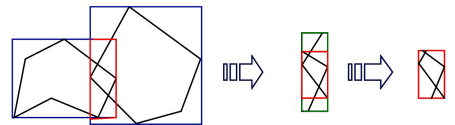


图 6 包围盒发生碰撞的示意图

Fig. 6 Schematic diagram of collision of bounding box

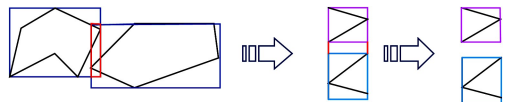


图 7 包围盒未发生碰撞的示意图

Fig. 7 Schematic diagram of no collision of bounding box

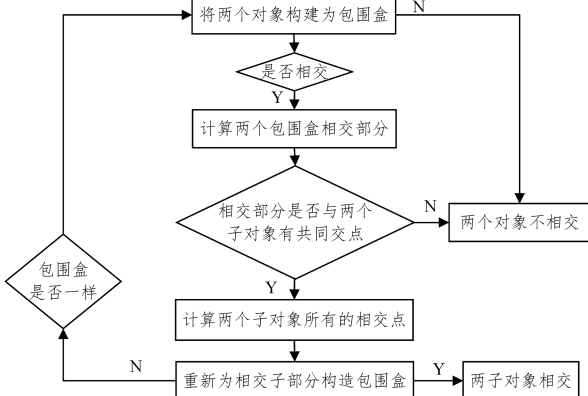


图 4 包围盒碰撞检测算法的流程图

Fig. 4 Flowchart of bounding box collision detection algorithm

3 实验条件及结果

为了验证使用 B+ 树改进后的层次包围盒碰撞检测效率,本文对此进行了实验验证。该实验在 Intel(R)Core(TM) i7-7700HQ CPU@2.80 GHz, 16 GB 内存、编译环境为 Visual Studio2017 的 PC 机上进行,并使用 C++ 语言为编码语言编

Operational Research, 2001, 129(1):1-47.

- [8] GRECO S, MATARAZZO B, SLOWINSKI R. Rough approximation by dominance relations[J]. International Journal of Intelligent Systems, 2002, 17(2):153-171.
- [9] CAO B R, LIU Y. Variable Precision Rough Set Model Based on Set Pair Situation Dominance Relationship[J]. Computer Engineering, 2015, 41(11):35-40.
- [10] LI Y, ZHANG L, WANG X J, et al. Attribute Reduction for Sequential Three-way Decisions Under Dominance-Equivalence Relations[J]. Computer Science, 2019, 46(2):242-248.
- [11] ANDERBERG M R. Cluster Analysis for Applications[M]. New York: Academic Press, 1973.
- [12] SUN J G, LIU J, ZHAO L Y. Clustering algorithms research [J]. Journal of Software, 2008, 19(1):48-61.
- [13] LIU Y H, MA H F, LIU H J, et al. An overlapping subspace K-Means clustering algorithm[J]. Computer Engineering, 2020, 46(8):58-63.
- [14] HUANG Z. Extensions to the K-means algorithm for clustering large data sets with categorical values[J]. Data Mining and Knowledge Discovery, 1998, 2(3):283-304.
- [15] HUANG Z, NG M. Fuzzy K-modes algorithm for clustering categorical data [J]. IEEE Transactions on Fuzzy Systems, 1999, 7(4):446-452.

- [16] CHEN Y, SONG J J, YANG X B. Accelerator for finding reduct based on attribute group[J]. Journal of Nanjing University of Science and Technology, 2020, 44(2):216-223.
- [17] CHEN Y, ZENG D S, XIE C. A Method of Attribute Reduction Based on Clustering[J]. Computer Systems Applications, 2009, 18(5):173-176.
- [18] LU J, ZHANG T, REN H L. Reduction of attribute in decision table based on clustering rate[J]. Computer Engineering and Application, 2012(28):135-138, 233.
- [19] CHEN Y C, LI O, SUN Y. Attribute reduction based on clustering discretization and variable precision neighborhood entropy [J]. Control and Decision, 2018, 33(8):1407-1414.
- [20] ZIARKO W. Variable precision rough set model[J]. Journal of Computer and System Sciences, 1993, 46(1):39-59.
- [21] UCI Machine Learning Repository[OL]. <https://archive.ics.uci.edu/ml/index.php>.



LI Yan, born in 1976, Ph.D, professor, master supervisor, is a member of China Computer Federation. Her main research interests include Granular computing and knowledge discovery and machine learning.

(上接第 333 页)

写基于 B+ 树存储的碰撞检测代码。实验的最终结果如图 8 所示。

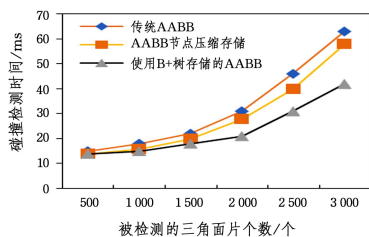


图 8 实验环境同等条件下 3 种算法的性能比较

Fig. 8 Comparison of performance of three algorithms

从图 8 可以看出,在相同的实验环境和空间中,针对相同数量的碰撞物体的碰撞检测效率而言,使用 B+ 树存储改进的算法相比于传统算法和结点压缩存储算法在检测时间上明显缩短,且随着被检测物体的增加,该算法的优势更加明显。

结束语 本文主要针对传统 AABB 包围盒空隙大和层次树各节点的存储空间大等问题,结合学者们对 AABB 包围盒结点压缩存储的思想,提出使用 B+ 树存储结点的概念,对可能发生碰撞的两个对象分别构造平衡的 AABB 层次树,通过降低每个结点存储的字节数进一步减少 AABB 树所须的内存空间大小,降低了遍历 AABB 树所需的时间,加快了算法的执行效率,最终使得碰撞检测效率得到了很大的提高。但是,在生产环境中,物体对象往往是多种形态存在,在本实现环境中涵盖的物体对象的形状有限而可能导致不同的环境下结果有略微差异。为了达到生产环境要求,后续还需要不断地进行实验对比以得出更为精确的结果。

参 考 文 献

- [1] 王志强,洪嘉振,杨辉. 碰撞检测问题研究综述[J]. 软件学报,

1999(5):98-104.

- [2] GARICA-ALONSO A, SERRANO N. Solving the Collision Detection Problem[J]. IEEE Computer Graphics and Application, 1994, 13(3):36-43.
- [3] 王伟. 轴对齐包围盒算法的研究[J]. 网络安全技术与应用, 2013(10):127-128.
- [4] 潘振宽, 李建波. 基于压缩的 AABB 树的碰撞检测算法[J]. 计算机科学, 2005, 33(2):213-215.
- [5] 沈学利, 吴琼. 基于包围盒和空间分割的混合碰撞检测算法[J]. 计算机工程, 2012, 38(6):256-258.
- [6] LI C F, FENG Y T, OWEN D R J. SMB: Collision detection based on temporal coherence[J]. Computer Methods in Applied Mechanics and Engineering, 2005, 195(19):2252-2269.
- [7] 施恩, 顾大权, 冯径, 等. B+ 树索引机制的研究及优化[J]. 计算机应用研究, 2017, 34(6):1766-1769.
- [8] 王崴, 周诚, 杨云, 等. 面向虚拟维修的碰撞检测算法[J]. 计算机应用与软件, 2016, 33(4):235-238.
- [9] AKENINE-MÖLLSER T. Fast 3D Triangle-Box Overlap Testing[J]. Journal of Graphics Tools, 2001, 6(1):29-33.
- [10] 金汉均. 虚拟环境中物体碰撞检测算法研究[D]. 武汉:华中科技大学, 2006.
- [11] 孙黎阳, 毛少杰, 林剑柠, 等. 基于 XYZ/ADL 的网络中心化仿真运行支撑平台体系结构形式化描述[J]. 计算机科学, 2012, 39(S1):365-369.



YANG Fan, born in 1995, postgraduate. His main research interests include artificial intelligence and big data.