

# Проверка проходимости полигональной модели по заданной траектории<sup>1</sup>

В.А. Ромакин,  
с.н.с., к.т.н., insight.ru@gmail.com,  
ИПУ РАН, г. Москва

В работе рассматривается задача проверки проходимости полигональной модели по заданной траектории в пространстве с препятствиями.

The article examines the problem of checking the passability of polygon mesh along given path in the obstructed space.

## Введение

В настоящее время разработаны различные методы построения пути в пространстве с препятствиями, рассматривающие перемещаемый объект как точку или, в лучшем случае, как сферу [1]. Однако, во многих приложениях требуется строить траектории для более сложных объектов, в том числе, трехмерных полигональных моделей. И если построить такую траекторию, рассматривая объект как точку, в дальнейшем требуется проверить, сможет ли по построенному пути пройти не точка, а сам объект.

Траектория геометрической фигуры представляет собой конечную последовательность перемещений объекта, направленную на достижение поставленной цели. В этой связи ее можно рассматривать как технологическую цепочку, а задачу проверки проходимости фигуры по заданной траектории — как задачу оценки качества технологической цепочки [2].

В работе рассматривается задача проверки проходимости полигональной модели по заданной траектории в пространстве с препятствиями. Обозначим  $M$  - перемещаемая полигональная модель,  $V$  - полигональная модель препятствий,  $T(M)$  - траектория движения модели  $M$ . Под траекторией понимается последовательность геометрических преобразований (переносов и поворотов)  $M$  в пространстве. Требуется проверить, может ли  $M$  пройти по заданной траектории без пересечений с  $V$ .

Пусть в траектории движения  $T(M)$  можно зафиксировать  $N$  промежуточных положений модели  $M$ , между которыми модель либо перемещается по прямому отрезку, либо поворачивается в пространстве вокруг некоторой точки. Обозначим  $M_i$  - модель  $M$  в  $i$ -м промежуточном положении ( $i=1..N$ ) и  $T_i$  - участки траектории между  $i$  и  $i+1$  промежуточным положением ( $i=1..N-1$ ). Для решения задачи сначала проверим пересечение всех  $M_i$  и  $V$ , а затем — пересечение  $M$  и  $V$  при проходе по каждому из участков траектории  $T_i$ .

## Проверка модели в промежуточных положениях

Полигональные модели  $M_i$  и  $V$  представляют собой наборы треугольных граней. Чтобы узнать, пересекаются ли два таких набора, необходимо проверить пересечение всех пар треугольников, взятых по одному из каждого набора. Для повышения эффективности этой операции обычно используют иерархии ограничивающих объемов.

Ограничивающий объем (bounding volume) объекта (множества объектов) — это замкнутый объем, который полностью содержит объект (множество объектов). Ограничивающие объемы используются для повышения эффективности геометрических операций за счет использования простых геометрических фигур вместо более сложных объектов, например, при трассировке лучей или обнаружении столкновений. На практике используется множество различных ограничивающих объемов: сферы, эллипсоиды, ориентированные по координатным осям ограничивающие параллелепипеды (AABB - axis-aligned bounding box) [3], произвольно ориентированные ограничивающие параллелепипеды (OBB - oriented bounding box) [4], дискретно-ориентированные многогранники с  $k$  плоскостями ( $k$ -DOP - discrete oriented polytopes) [7], ограничивающие капсулы (swept sphere), цилиндры и др. (см. рис.1). Выбор фигуры определяется конкретным приложением.

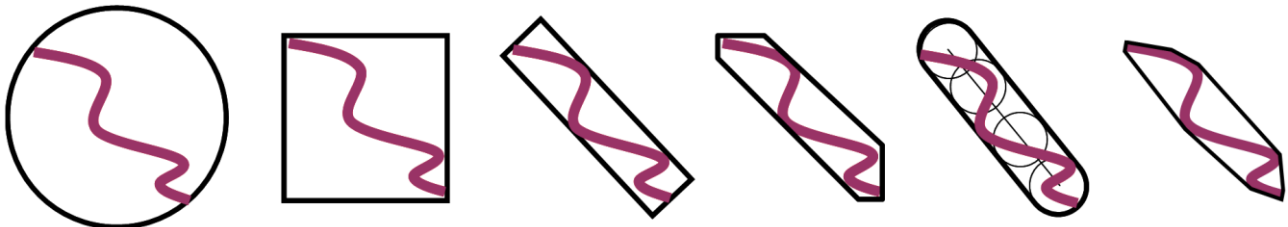


рис. 1 Распространенные ограничивающие объемы. Слева направо: сфера, ограничивающий параллелепипед AABB, ограничивающий параллелепипед OBB, многогранник k-DOP, ограничивающий капсула, выпуклая оболочка [9]

Использование ограничивающих объемов позволяет упростить проверку на пересечение объектов, однако при сложно структурированных объектах один ограничивающий объем дает недостаточное приближение исходного объекта. Для оптимизации задачи проверки пересечений объекта с препятствиями используют иерархии ограничивающих объемов. Иерархия ограничивающих объёмов (BVH – bounding volume hierarchy) — это дерево, каждый узел

<sup>1</sup> Работа выполнена при финансовой поддержке проекта РФФИ № 16–29–12839–офи\_м.

которого ассоциирован с ограничивающим объемом (см. рис. 2). Ограничивающий объем в корне иерархии содержит весь объект целиком, а ограничивающие объемы потомков любого узла иерархии совокупно содержат ту часть объекта, которую содержит ограничивающий объем этого узла. Совокупность ограничивающих объемов на каждом уровне иерархии содержит объект целиком, а каждый следующий уровень иерархии все плотнее и плотнее охватывает объект [9].

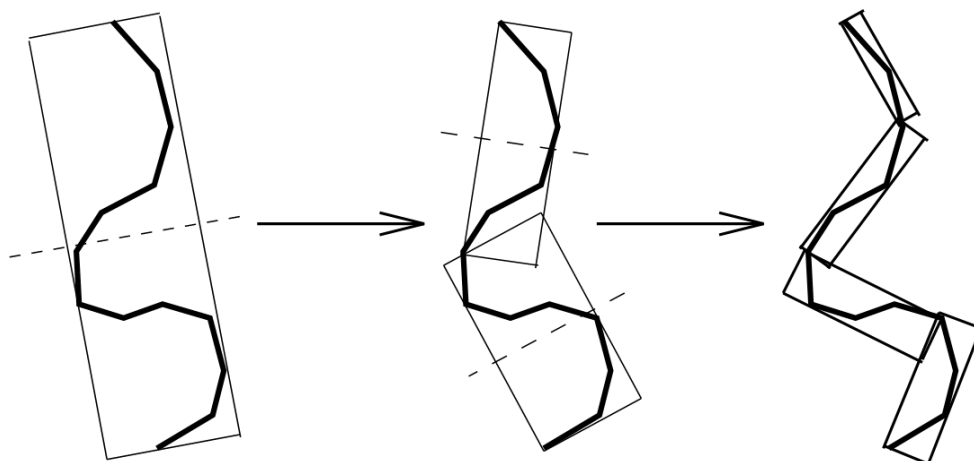


рис. 2 Иерархия ограничивающих параллелепипедов ОБВ полилинии [4].  
Ограничивающий параллелепипед верхнего уровня это грубое приближение объекта, а совокупность ограничивающих параллелепипедов в листьях иерархии обеспечивает наилучшую аппроксимацию

Иерархия ограничивающих объемов полигональной модели может быть построена либо сверху вниз, либо снизу вверх. При построении сверху вниз сначала вычисляется ограничивающий объем для всего множества примитивов модели (корень иерархии), затем это множество разбивается на два (или более) подмножества, и для каждого из них вычисляется ограничивающий объем. Эти разбиения продолжаются до тех пор, пока все полученные подмножества не станут неделимыми. При разбиении очередного множества примитивов на дочерние требуется определить, в какое подмножество отнести тот или иной примитив. Наиболее простым способом является простое деление пространства на две части: все примитивы с координатами выше заданной относятся к правому поддереву, все остальные — к левому. При построении снизу вверх сначала вычисляются ограничивающие объемы каждого примитива, затем примитивы близлежащих ограничивающих объемов объединяются для построения ограничивающего объема более высокого уровня, и так до тех пор, пока не останется один ограничивающий объем.

При проверке на пересечение объектов иерархии ограничивающих объемов позволяют быстро определить множество пар непересекающихся элементов (в нашем случае - треугольников). Пусть заданы два объекта и их иерархии ограничивающих объемов. Если ограничивающие объемы в корне иерархии не пересекаются, тогда сами объекты также не пересекаются. В противном случае объекты могут пересекаться, и тогда в одной из иерархий (например, в которой размер пересекающегося ограничивающего объема больше) следует спуститься на уровень ниже и повторить проверку на пересечение. Эти действия следует продолжать до тех пор, пока на очередном уровне иерархии пересечения будут отсутствовать, либо пока не будет достигнут самый нижний уровень иерархии. При спуске на более низкий уровень нужно двигаться только в том направлении, где есть пересечение ограничивающих объемов.

Если проверяемые объекты расположены далеко друг от друга, тогда зачастую при проверке пересечений не приходится опускаться на нижние уровни иерархий. Если же объекты пересекаются или расположены близко друг к другу, тогда приходится проверять всю иерархию до самого нижнего уровня, а затем выполнять проверку на пересечение примитивов, принадлежащих пересекающимся ограничивающим объемам нижних уровней иерархий.

Эффективность того или иного класса ограничивающего объема в иерархии зависит от его способности плотно охватывать объекты, используемые в конкретном приложении, а также от скорости проверки пересечений между двумя ограничивающими объемами этого класса. С одной стороны, чем лучше ограничивающий объем приближает исходный объект, тем меньше будет ошибочных срабатываний при проверке пересечений объектов (случаев, когда два ограничивающих объема пересекаются, а соответствующие части объектов, принадлежащие этим объемам — нет), и тем меньше потребуется уровней иерархии для приближения объекта с заданной ошибкой. С другой стороны, чем проще форма ограничивающего объема, тем меньше вычислительная сложность проверки пересечения двух таких объемов.

Проще всего проверить пересечение двух сфер, однако иерархия сфер очень медленно приближает геометрию объектов (за исключением случаев, когда эта геометрия разреженная и относительно равномерная).

Параллелепипеды AABB, а тем более ориентированные параллелепипеды OBV зачастую гораздо лучше подходят в качестве ограничивающего объема, чем сферы, так как быстрее приближают геометрию объектов.

Еще лучше могут приближать геометрию объектов дискретно-ориентированные многогранники k-Dop, которые представляют собой объемы, ограниченные заданным количеством (k) плоскостей с заданной ориентацией. Такие многогранники описываются минимальными и максимальными интервалами, заданными на k/2 направляющих осях. Параллелепипеды AABB представляют собой частный случай 6-DOP, а проверка пересечений k-Dop аналогична проверке пересечений AABB: многогранники пересекаются, если пересекаются все k/2 интервала.

При выборе вида ограничивающего объема следует учитывать, что часть из них (ограничивающие параллелепипеды AABB, многогранники k-Dop) требуется пересчитывать при повороте объекта, а другие (сферы, ориентированные параллелепипеды OBV, выпуклые оболочки) необходимо всего лишь повернуть вместе с объектом. Поскольку в рассматриваемой задаче модель M может вращаться, остановимся на иерархии ориентированных параллелепипе-

дов ОВВ, обеспечивающей хорошее приближение геометрии объектов. Алгоритм построения ОВВ полигональной модели и алгоритм пересечения двух ОВВ представлены в [4].

Если после проверки пересечения иерархий ОВВ (или ОВВ-деревьев) окажется, что они пересекаются, тогда следует проверить на пересечение все треугольники, принадлежащие пересекающимся ОВВ в листьях ОВВ-деревьев. Для быстрой проверки пересечений двух треугольников можно использовать алгоритм Моллера [6].

### Проверка модели между промежуточными положениями

Будем называть следом фигуры тот объем пространства, который она занимает при перемещении или вращении в пространстве. Например, при перемещении по прямому отрезку следом точки будет прямой отрезок, следом прямого отрезка — параллелограмм (в вырожденном случае — прямой отрезок), следом треугольника — треугольная призма (в вырожденном случае — четырехугольник), следом сферы - вышеупомянутая ограничивающая капсула. Очевидно, что если фигура при перемещении пересекается с препятствиями, тогда и ее след также пересекается с ними, и наоборот.

След полигональной модели равен объединению следов ее граней. Обозначим  $S_i$  – след полигональной модели на участке траектории  $T_i$ , а  $S_i^1$  и  $S_i^2$  - объединение следов соответственно вершин и ребер полигональной модели на участке траектории  $T_i$ . Очевидно, что  $S_i \supset S_i^2 \supset S_i^1$ . При проверке пересечения  $S_i$  и  $B$  можно сделать допущение и приблизить  $S_i$  одним из двух его подмножеств  $S_i^2$  или  $S_i^1$ . Однако в этом случае проверка на пересечение не всегда даст точный результат: если пересечение подмножеств  $S_i^2$  или  $S_i^1$  с препятствиями  $B$  не будет обнаружено, то сам след  $S_i$  все же может пересекаться с препятствиями.

Для выбранного множества ( $S_i$ ,  $S_i^2$  или  $S_i^1$ ) следует построить иерархию ограничивающих объемов и сначала проверять пересечение этой иерархии с иерархией ограничивающих объемов модели препятствий. В случае, если они пересекаются, тогда нужно искать пересечения подмножества следов (вершин, ребер или граней) модели и подмножества треугольников модели препятствий.

В случае перемещения модели по прямому отрезку след вершины представляет собой прямой отрезок. Для проверки его пересечения с треугольником из  $B$  можно использовать, например, алгоритм Моллера-Трумбора определения пересечения луча и треугольника в трехмерном пространстве [5]. Этот алгоритм определяет расстояние от начала луча до точки пересечения с треугольником (если она существует), и если это расстояние меньше длины следа вершины, тогда след вершины (а следовательно, и вся полигональная модель) пересекается с треугольником.

След ребра представляет собой четырехугольник, который можно представить в виде двух треугольников. След грани представляет собой треугольную призму, поверхность которой можно представить в виде 8-ми треугольников. Для проверки пересечения этих фигур (после разбиения на треугольники) и треугольников из  $B$  можно использовать алгоритм Моллера [6].

В случае поворота модели в пространстве след вершины представляет собой дугу окружности. Алгоритм проверки пересечения дуги окружности с треугольником может быть следующим.

- 1) Найти линию пересечения плоскости окружности и плоскости треугольника (если она существует).
- 2) Найти точки пересечения этой линии и дуги окружности (если таковые существуют).
- 3) Проверить, лежит ли хотя бы одна точка пересечения в треугольнике.

Следы ребер и граней при повороте достаточно сложны для непосредственной проверки пересечения.

### Заключение

Рассмотрена задача проверки проходимости полигональной модели по заданной траектории и предложен алгоритм решения этой задачи.

### Литература

1. В.А. Ромакин Автоматизация разборки трехмерных моделей машиностроительных конструкций // Информационные технологии в проектировании и производстве. – 2016. №4. – С. 24-28.
2. А.В. Латышев, В.А. Ромакин, В.М. Хачумов, М. В. Хачумов. «Методы и модели автоматического синтеза технологических процессов, основанного на знаниях», Программные системы: теория и приложения, 2016, 7:3(30), с. 25–43.
3. J.D. Cohen, M.C. Lin, D. Manocha, and M. K. Ponamgi. I-collide: An interactive and exact collision detection system for large-scale environments. In Proc. ACM Interactive 3D Graphics Conf., pp.189–196, 1995.
4. S. Gottschalk, M. Lin and D. Manocha. OBBTree: A Hierarchical Structure for Rapid Interference Detection. In Proc. of ACM Siggraph'96, pp.171-180, 1996.
5. T. Moller, B. Trumbore. Fast, minimum storage ray-triangle intersection. Journal of Graphics Tools, v.2 n.1, pp.21-28, 1997.
6. T. Moller. A fast triangle-triangle intersection test. Journal of Graphics Tools, 1997; 2(2): 25–30.
7. J. Klosowski, M. Held, Joseph S. B. Mitchell, K. Zikan, and H. Sowizral. Efficient collision detection using bounding volume hierarchies of k-DOPs. IEEE Trans. Visualizat. Comput. Graph., 4(1):21–36, 1998.
8. T. Moller. Fast 3D Triangle-Box Overlap Testing. Journal of Graphics Tools, vol. 6, no. 1, pp. 29–33, 2001.
9. A. Nguyen. Implicit bounding volumes and bounding volume hierarchies. Doctoral dissertation, Stanford University, 2006.