

Анализ столкновений

11, ноябрь 2014

Горячкина А. Ю.

УДК: 004.92

Россия, МГТУ им. Н.Э. Баумана

agoryachkina@mail.ru

Введение

Специалистам в области автоматизации проектирования и программистам часто приходится сталкиваться с задачей анализа геометрических препятствий при моделировании трехмерных сцен. Качественная постановка этой задачи проста. Задана конфигурация трехмерной сцены, которая может включать в себя статические и динамические элементы. Требуется переместить заданный подвижный элемент сцены из стартовой в финальную позицию, избегая столкновений с остальными элементами сцены. Часто при этом необходимо получить некоторую дополнительную информацию, связанную с таким перемещением, например траекторию движения, описание геометрии препятствий, времена и количество столкновений и др.

Основным генератором постановок и потребителем методов решения этой задачи являются компьютерные игры и симуляторы. Однако ее прикладное значение выходит далеко за пределы шутеров и аркадных игр. Она применяется в автоматизированном проектировании, для синтеза траекторий движения режущего инструмента в системах числового программного управления, для программирования движения роботов в среде с препятствиями, в системах виртуального прототипирования, при компьютерном моделировании физических процессов и др. Невозможно собрать техническую систему (машину или прибор) без учета геометрических ограничений, которые накладывает на каждый устанавливаемый элемент собранный фрагмент изделия.

В многочисленных публикациях на английском языке данная проблема называется Collision Detection, что на русский обычно переводят как анализ или обнаружение столкновений [3]. В системах автоматизированного проектирования процессов сборки она называется геометрическим доступом или геометрической разрешимостью [8].

Все модели, предназначенные для анализа геометрических столкновений, можно разделить на две принципиально различные группы: априорные и апостериорные [5]. В апостериорных моделях анализ сцены на геометрическую разрешимость выполняется че-

рез небольшие промежутки времени после движения объекта. Каждый отдельный расчет заключается в проверке условий критической близости или пересечения объекта и остальной части сцены. Факт столкновения идентифицируется с некоторой погрешностью, поскольку в расчетах используются упрощенные геометрические модели сцены. После этого создается список пересекающихся тел и принимается решение о корректировке траекторий с учетом установленного столкновения. Апостериорные модели не учитывают сложные физические процессы (например, трение, упругие и неупругие деформации), происходящие при столкновении тел и влияющие на конфигурацию сцены после этого события. Они отличаются вычислительной эффективностью, сравнительно невысокой точностью и применяются, главным образом, в компьютерных играх, промышленных симуляторах и виртуальном прототипировании.

Априорными называются такие модели, которые способны рассчитать траекторию движения объекта и предсказать его столкновение со статическими элементами сцены. Это событие вычисляется до изменения пространственной конфигурации сцены (априори). Некоторые типы априорных моделей обладают способностью имитировать физические эффекты, возникающие при столкновении тел. Как правило, модели этого типа отличаются высокой точностью и большими вычислительными затратами. Априорное моделирование трехмерных сцен имеет и принципиальные ограничения. В общем случае, эта задача описывается системой дифференциальных уравнений и может не иметь точного аналитического решения, а ее численное решение требует громадных вычислительных затрат. В этой работе рассмотрим апостериорные модели, поскольку они широко используются в современных программных системах.

Ключевая проблема любой апостериорной модели – это повышение эффективности алгоритмов, вычисляющих пересечение объектов сцены. В основе таких алгоритмов лежат тесты, определяющие пересечение двух геометрических объектов в статике. Поскольку эти тесты выполняются в процессе анализа сцены многократно, то совокупная трудоемкость алгоритмов анализа пересечений может быть очень высока. В общем случае надлежит проверить множество всех упорядоченных пар, состоящих из примитивов обоих объектов. Если учесть, что данный анализ применяется ко всем парам потенциально опасных объектов и выполняется в каждом кадре динамично меняющейся сцены, то совокупная трудоемкость такой процедуры очень быстро растет. Рассмотрим основные приемы, которые используются в современных алгоритмах анализа пересечений для повышения их эффективности.

Геометрическая аппроксимация

Первые компьютерные реализации алгоритмов анализа пересечений основывались на аппроксимации сложных геометрических форм простыми фигурами. Для этого применяются параллелепипеды (а), сферы (б), выпуклые оболочки (в), политопы (г) и другие двумерные и трехмерные примитивы, описанные вокруг геометрических объектов (рис. 1).

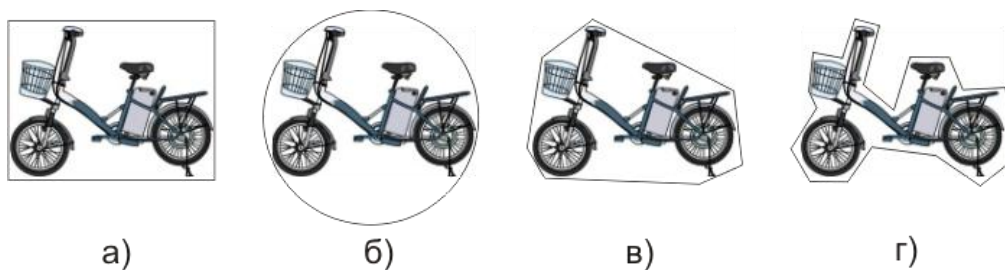


Рис. 1. Виды аппроксимации геометрических объектов

Фигуры, которые выполняют функции оболочек, должны обладать следующими свойствами: допускать простую проверку на пересечение, давать достаточно точную аппроксимацию формы, иметь эффективный способ хранения в памяти компьютера и не терять своих свойств при повороте и линейном перемещении. Среди трехмерных примитивов этим требованиям удовлетворяют: сферы, параллелепипеды, полиэдры, дискретно-ориентированные политопы (*k*-DOPs), выпуклые оболочки, конусы, эллипсоиды и др [4]. В это число не входят цилиндры, поскольку для этих простых фигур тесты на статическое пересечение выполняются со значительными вычислительными издержками.

Если оболочки не пересекаются, то, очевидно, свободны от пересечений и заключенные в них объекты. В противном случае ситуация требует дополнительного исследования. Основное преимущество геометрической аппроксимации – вычислительная простота тестов на статическое пересечение. Например, для двух сфер это неравенство $R_1 + R_2 \leq L$, где R_1 , R_2 – это радиус сфер, а L – расстояние между их центрами. Для прямоугольников, политопов и выпуклых оболочек требуется решить простую систему неравенств для вершин, заданных своими декартовыми или барицентрическими координатами.

Главные недостатки геометрической аппроксимации следующие:

1. высокая погрешность приближения формы для сложных геометрических объектов;
2. пересечение аппроксимирующих оболочек представляет собой необходимое условия для пересечения самих объектов, но не достаточное. То есть в некоторых случаях этот тест может давать ошибочные результаты;
3. невозможность расчета геометрических характеристик области пересечения, что требуется в некоторых приложениях.

Иерархическая аппроксимация геометрических объектов

Для анализа пересечений успешно применяются различные геометрические модели, основанные на иерархическом разбиении объекта на примитивы: восьмеричные деревья, конструктивная твердотельная геометрия, *BSP*-деревья и др. [5]. Иерархические модели имеют два фундаментальных преимущества:

1. если на некотором уровне геометрической иерархии установлен факт пересечения (или непересечения) элементов двух объектов, то это заключение может быть распространено на все верхние (нижние) уровни модели.

2. для локализации пересекающихся элементов и уточнения их геометрии следует проанализировать только поддеревья, которые исходят из пересекающихся узлов. Остальные фрагменты деревьев не содержат полезной информации для решения этой задачи.

Во многих случаях эти эвристические правила позволяют ограничить пространство поиска и значительно увеличить эффективность процедуры анализа.

В публикациях по компьютерной графике принято различать два класса геометрических иерархических моделей: иерархия пространственного разбиения (space partitioning hierarchy) и иерархии разбиения объекта (object partitioning hierarchy, ОВН или bounding volume hierarchy, BVH). Модели первого класса основаны на разбиении пространства на элементарные ячейки (квадраты или кубы), модели второго класса используют иерархическое деление самих геометрических объектов. Самым распространенным представителем первого класса являются восьмеричные деревья, основанные на воксельном представлении трехмерной геометрии. Хорошо известны их отличительные особенности в решении задачи анализа пересечений. Они позволяют быстро рассчитать приблизительную геометрию пересечения двух тел не очень сложной конфигурации. Увеличение точности анализа достигается за счет разбиения ячеек пространства (вокселей) на ячейки меньшего размера. Это влечет за собой стремительный рост числа ячеек и, как следствие, вычислительных затрат на расчет геометрии пересечения двух объектов.

Класс геометрических моделей, использующих иерархическое разбиение объекта, обладает большим разнообразием по критериям точность и вычислительная эффективность. Существуют модели, по которым можно быстро получить приближенное решение, а также модели, генерирующие решение высокой точности за приемлемые сроки.

Наиболее известным и цитируемым примером моделей этого класса является представление объекта в конструктивной твердотельной геометрии (Constructive Solid Geometry – CSG) (рис. 2).

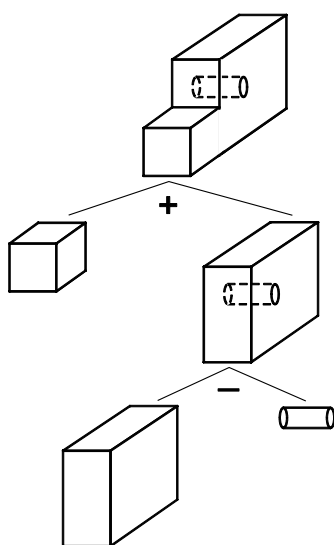


Рис. 2. Представление геометрического объекта в виде CSG-дерева

Модель *CSG* широко применяется в системах твердотельного геометрического моделирования и автоматизированного проектирования. В играх и симуляторах геометрические модели – это сетки и полиэдры, описание которых в виде *CSG*-дерева избыточно, а часто и невозможно. Анализ столкновений в этих приложениях выполняется, как правило, при помощи *BVH*-моделей, основанных на иерархической аппроксимации элементов сцены простыми геометрическими формами: сферами, параллелепипедами и др.

На рис.3,а показана *BVH*-модель треугольника, в которой для аппроксимации границы используются окружности. На рис. 3,б изображено дерево этой модели.

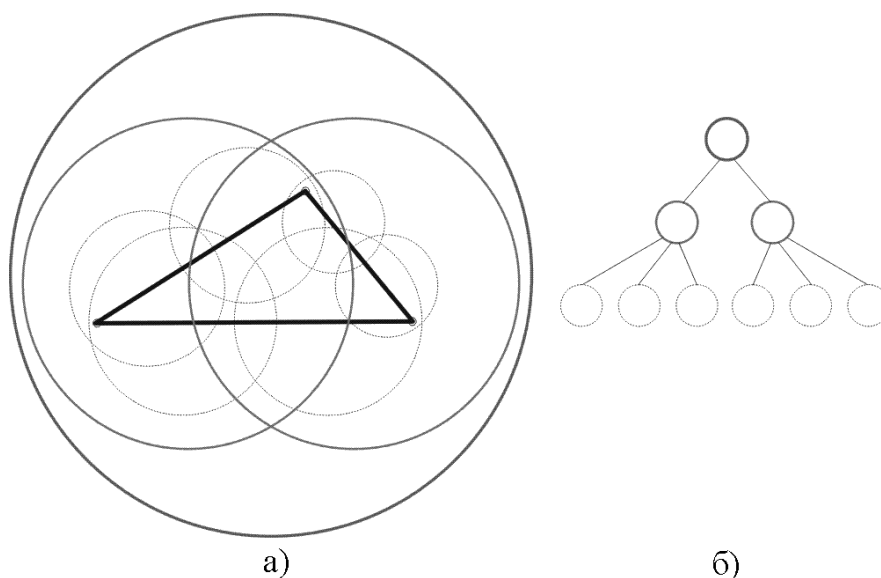


Рис. 3. Иерархическая аппроксимация треугольника при помощи окружностей

Рисунки 4,а и 4,б иллюстрируют способ анализа пересечений двух объектов, представленных этой геометрической моделью. Пересечение описанных окружностей – это необходимое, но не достаточное условие пересечения самих объектов. Для верификации данного события построена система вложенных окружностей (рис. 4,б), которые локализовали геометрию пересечения. Чтобы не загромождать изображения избыточными подробностями вложенные окружности на рис 4,б представлены в виде секторов.

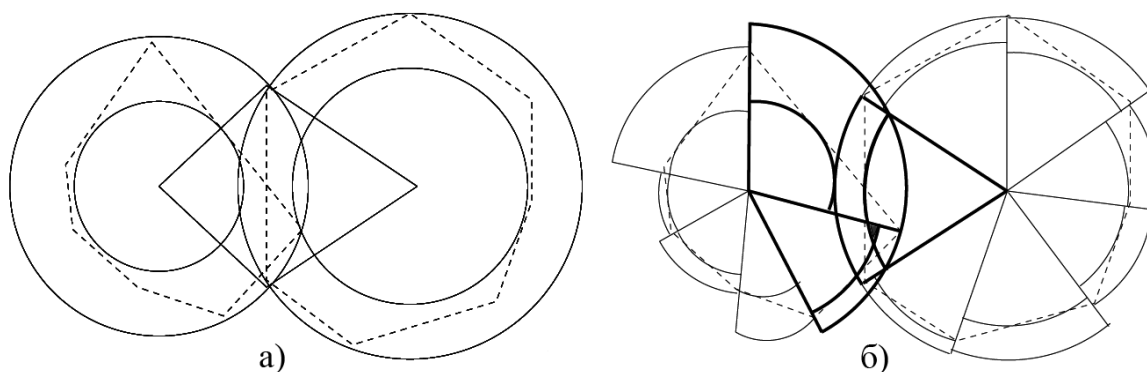


Рис. 4. Анализ пересечений двух многоугольников в *BVH*-модели

Узлами *BVH*-дерева могут быть не только окружности или сферы. В различных геометрических приложениях для этих целей используются: прямоугольники, параллелепипеды, выпуклые оболочки, многоугольники, полиэдры и другие элементарные геометрические формы, имеющие простые способы расчета пересечений. Так на рис. 5 показана *BVH*-представление, в котором элементарными формами являются прямоугольники, а на рис. 6 изображено дерево этой модели. Это важный частный случай иерархических геометрических моделей в публикациях на английском языке называется *OBB*-деревом (oriented bounding box) [2, 7].

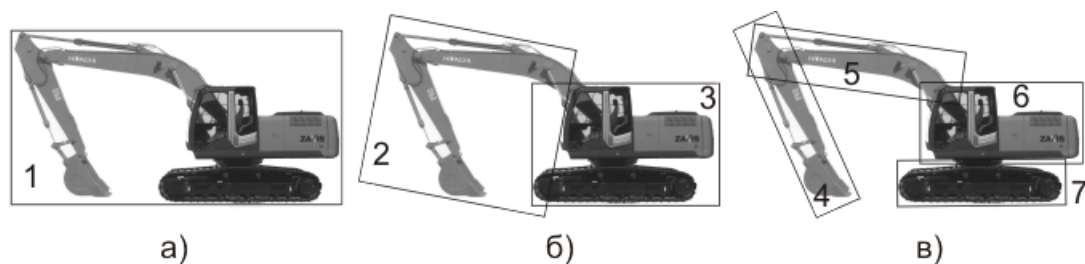


Рис. 5. Построение *OBB*-модели

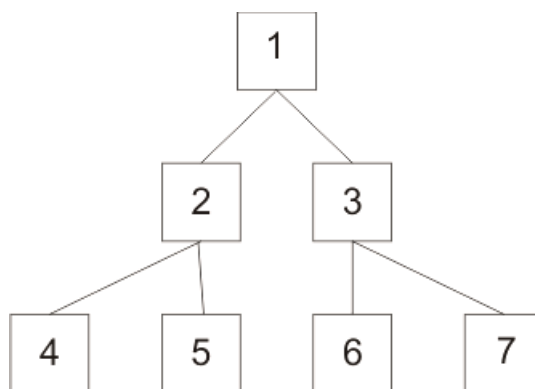


Рис. 6. Дерево *OBB*-модели

BVH-модели принято разделять не только по типу базовых геометрических форм, выбранных для приближения границ трехмерного тела или двумерной фигуры. Важными критериями, от которых зависит эффективность процедура анализа пересечений и трудоемкость создания модели являются: структура, высота и сбалансированность *BVH*-дерева; точность аппроксимации границы, степень перекрытия узлов одного уровня.

Одна из ключевых проблем, активно обсуждаемых в публикациях по компьютерной графике, – выбор рациональной структуры *BVH*-дерева. Какая структура имеет преимущество в процессе обработки: дерево ограниченной высоты или ограниченного ветвления? В самом деле, *BVH*-дерево, имеющее маленькую высоту (h), но высокую степень вершин (d), отличается небольшими вычислительными затратами на свой обход. Но, с другой стороны, для этой структуры значительно усложняется проверка пересечений в узлах и задача покрытия элементов объекта d примитивами выбранной *BVH*-модели (ок-

ружностями или прямоугольниками). Этот вопрос не может быть решен теоретически, а только на основе вычислительных экспериментов. Современная практика отдает предпочтение различного типа сбалансированным *BVH*-деревьям (бинарным, чаще всего) [1].

Свойства иерархической модели в значительной степени зависят от стратегии построения *BVH*-дерева. Существует множество способов синтеза таких структур, но среди них принято выделять две базовые стратегии: нисходящую (сверху-вниз, рис.7,а) и восходящую (снизу-вверх, рис. 7,б).

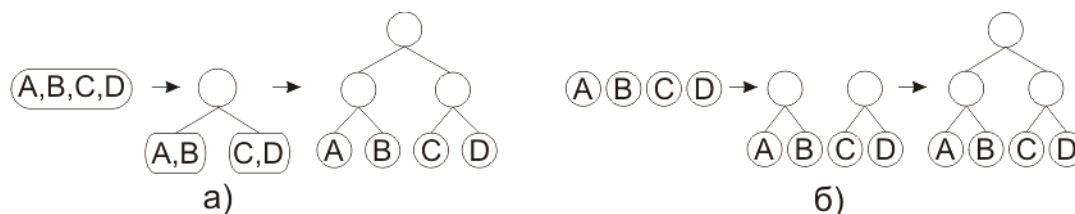


Рис. 7. Стратегии построения *BVH*-деревьев: а) – нисходящая (сверху-вниз), б) – восходящая (снизу-вверх)

Нисходящая стратегия проще в реализации. Ее работу можно описать в виде простой рекурсивной процедуры. Для геометрического объекта (или множество объектов) строится покрывающая оболочка (bounding volume). Затем объект делится на совокупность подобъектов, для каждого из которых создается собственная оболочка выбранного типа и т.д. Критерием остановки этой процедуры является точность аппроксимации, достигнутая для объектов нижнего уровня, или геометрическая простота этих объектов.

Восходящая стратегия создания *BVH*-дерева является намного более сложной в реализации, но отличается высокой точностью геометрической аппроксимации. Множество геометрических объектов сцены объявляются листьями *BVH*-дерева. На следующем уровне иерархии следует определить объекты, которые в некотором фиксированном смысле, являются близкими друг к другу и образуют вершины второго уровня и так далее до корневой вершины. Это значит, что на каждом уровне *BVH*-дерева приходится решать задачу кластеризации, то есть поиск близких объектов и объединения их в группу подобных – кластер. Кластеризация – это классическая проблема анализа данных и математики и для ее решения предложено большое число методов и алгоритмов. Даже самые простые методы ее решения, например метод ближайшего соседа, требует значительных затрат машинного времени.

Пространственное разбиение объектов – это комбинаторная процедура, которая может продуцировать принципиально различные результаты даже для объектов простой формы. Рассмотрим основные приемы и критерии разбиения геометрических объектов на элементы.

Минимизация общего числа дочерних вершин в BVH-дереве. Этот критерий используется для создания двоичных сбалансированных деревьев, для которых существуют эффективные алгоритмы перебора и экономичные структуры данных. Аппроксимация объекта при помощи небольшого числа геометрических примитивов означает, что они должны

иметь небольшую погрешность приближения. Из множества простых геометрических форм таким свойством обладают лишь выпуклые оболочки и политопы.

Минимизация пересечений дочерних вершин BVH-дерева. Этот критерий сложен в реализации даже для простых геометрических примитивов сфер и параллелепипедов.

Минимизация максимального объема (площади) вершин BVH-дерева.

Применяются также различные смешанные стратегии разбиения, в которых перечисленные критерии – это составляющие некоторого аддитивного генерального критерия.

Итак, тест на проверку статического пересечения – это ключевая задача для анализа столкновений объектов, принадлежащих статической или динамической сцене. Однако эффективность этой проверки не гарантирует низкие вычислительные затраты самого алгоритма анализа столкновений, поскольку в динамических сценах ее приходится выполнять многократно в критические моменты времени для всех потенциально опасных пар.

Ограничение перебора

В большинстве сцен существуют конфигурации тел, положение которых не меняется или меняется с небольшой скоростью относительно смены кадров или состояний. Алгоритмы расчета пересечений исключают из анализа объекты, пересечение которых маловероятно или невозможно, а рассматривают только потенциально опасные конфигурации. Некоторые современные алгоритмы способны делать предсказания относительно возможного пересечения объектов на основе данных полученных из предыдущих кадров. Это свойство принято называть временной когерентностью [3, 6].

Ключевая проблема в анализе столкновений динамических объектов – это временная дискретизация сцены. Точность анализа решающим образом зависит от частоты применения тестов на статическое пересечение объектов. Если она невелика, то существует опасность пропуска критических столкновений. В противном случае стремительно растут вычислительные затраты на анализ сцены.

В некоторых приложениях применяется достаточно грубый критерий, который не учитывает направления движения элементов сцены, а ее временную дискретизацию выполняют по нижней оценке расстояний между объектами и верхней оценке относительных скоростей. Рассмотрим более точный критерий дискретизации, который учитывает не только скорость сближения, но и направление перемещения объектов (рис. 8).

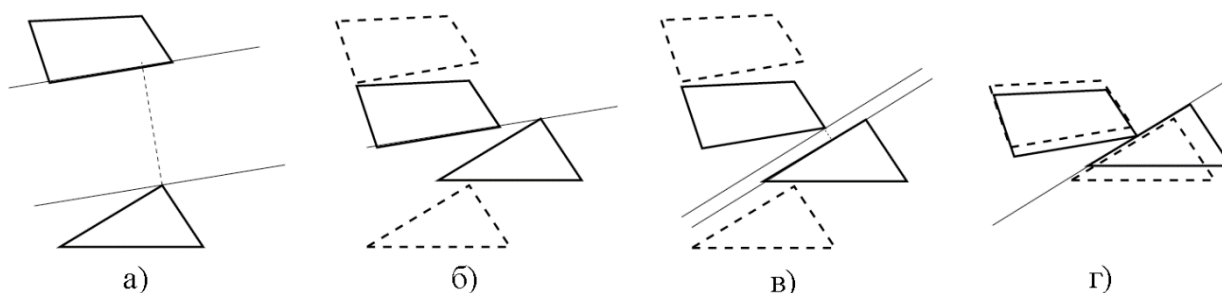


Рис. 8. Учет направления движения объектов

На рис. 8,а показана начальная позиция двух объектов. В этой позиции определяют точки объектов, расположенные на кратчайшем расстоянии друг от друга, и соединяют их прямой линией (на рис. 8,а показана пунктиром). По относительной скорости и длине этой линии рассчитывается следующий момент временной дискретизации и положение объектов в этот момент (рис. 8,б). После этого процедура повторяется в новой ситуации, которая показана на рис. 8,в. На рис. 8,г представлено положение объектов в момент третьей выборки, которая фиксирует действительное пересечение многоугольников. Описанный способ расчета времен для проверки пересечений часто применяется для объектов, представленных граничной моделью (boundary representation, B-rep).

Множество точек, которые занимает объект в процессе своего движения, называется заметанием или заметаемым объемом. Если заметаемые объемы двух тел не пересекаются, то эти тела не сталкиваются в течении заданного отрезка времени. Однако это условия является только достаточным. Легко привести примеры, когда заметаемые объемы пересекаются, а порождающие их тела не сталкиваются. На рис. 9 показаны заметаемые объемы трех многоугольников. Заметания многоугольников 1 и 3 пересекаются, а фигуры, их порождающие сталкиваются. Тогда как фигуры 2 и 3 не сталкиваются, но их заметания пересекаются.

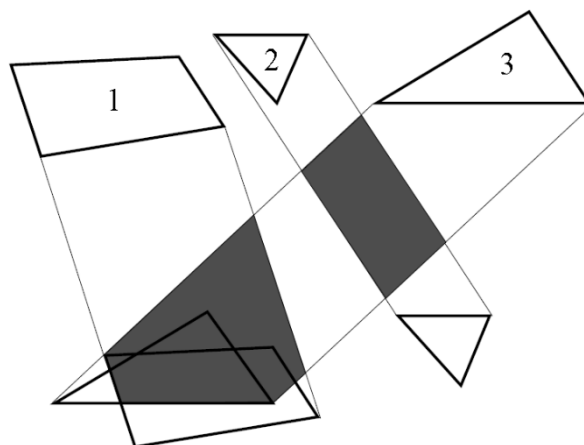


Рис. 9. Заметаемые объемы многоугольников

Чтобы сделать это тест результативным, необходимо рассмотреть заметания, которые создаются относительным движением пар объектов и сделать для каждой пары объектов сцены. Понятно, что в общем случае такая проверка потребует чрезвычайно высоких вычислительных затрат.

Список литературы

1. Bentley J., Ottmann T. «Algorithms for reporting and counting geometric intersections», IEEE Trans. Comput., vol. 28, no. 9, pp. 643–647, 1979.

2. Gottschalk S., Lin M. C., Manocha D. «OBBTree: A Hierarchical Structure for Rapid Interference Detection», Proc. SIGGRAPH'96, ACM Computer Graphics, New York, NY, USA, pp. 171-180, 1996.
3. Ericson Christer Real-time Collision Detection. Elsevier Science, 2004, 632 p.
4. «Efficient Collision Detection Using Bounding Volume Hierarchies of k-DOPs» / Klosowski J.T., Held M., Mitchell J.S.B, Sowizral H., Zikan K., IEEE Trans on Vis and Computer Graphics, 4/1, pp. 21-36, 1998.
5. Van den Bergen G. Collision Detection in Interactive 3D Environments. Morgan Kaufmann, ISBN: 155860801X, 2003, 299 p.
6. Yong Lu, Rajit Gadh. «Constrained and Aggregated Half Space Volume Decomposition: Generating Cutting Patches with Internal and External Extending» Proceedings of the Eighth Pacific Conference on Computer Graphics and Applications (PG 00), pp. 262–271, 2000. <http://csdl.computer.org/comp/proceedings/pg/2000/0868/00/08680262abs.htm>
7. Zachmann G. «The Box Tree: Exact and Fast Collision Detection of Arbitrary Polyhedra», First Workshop on Simulation and Interaction in Virtual Environments (SIVE 95), University of Iowa, July 1995. <http://web.informatik.uni-bonn.de/II/ag-klein/people/zach/papers/>
8. Божко А.Н. Геометрическая разрешимость трехмерных сцен // «Вестник МГТУ» – 3(92) – 2013