

8.1 Подсекция “ПЕРЕДАЧА И ОБРАБОТКА ДАННЫХ”

УДК 004.032.6

Н.А. Жилияк, доц., канд. техн. наук
(БГТУ, г. Минск)

ПРИМЕНЕНИЕ ФРАКТАЛЬНЫХ ПРОЦЕДУРНЫХ ОБЪЕКТОВ В АЛГОРИТМЕ ВЫЧИСЛЕНИЯ КОЛЛИЗИЙ

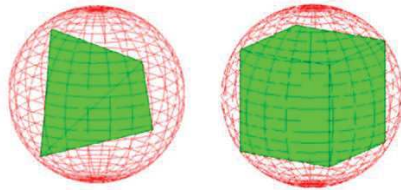
Разрабатываемый алгоритм вычисления коллизий должен предоставлять определенные преимущества, которые отсутствуют в существующих алгоритмах вычисления коллизий, либо не развиты в полной мере [1-3].

Существующие на данный момент алгоритмы вычисления коллизий значительно оптимизированы под основной подход представления геометрических форм объектов: примитивы, полигональные сетки, выпуклые объекты. Более нишевые примеры применения алгоритмов коллизий недостаточно развиты, поэтому для разработки собственного алгоритма имеет смысл ориентироваться на них.

Изначальная идея заключалась в представлении частей крупных объектов и примитивов в качестве сфер, в которые вписаны в правильные тетраэдр или куб. Координаты правильных примитивов, вписанных в сферу, могут быть вычислены через данные сферы – центр и радиус, что становится проще сделать, чем для оригинальных объектов (рис. 1).

Примитивы в такой системе по умолчанию получают в распоряжение ограничивающий объем в виде описывающей их сферы. Подобный метод может предоставить альтернативный способ создания полигональных сеток, где используются сферы вместо плоских объектов.

Также, вогнутые объекты, использующие полигональные сетки в традиционных алгоритмах, решаются с помощью разбиения объекта на несколько выпуклых либо использованием специализированных сложных алгоритмов, что создает нагрузки. С использованием описанного подхода сложный объект выражает свою форму с помощью объемных фигур вместо плоскостей, отчего вогнутые объекты не должны вызывать проблем. Также, благодаря изначальной объемности объектов, проще реализуются мягкие тела, которые по умолчанию требуют вычислений внутренней части тела, а не только его оболочку.



**Рисунок 1 – Описанные сферой примитивы
(тетраэдр слева, куб справа)**

Проблемы у данного метода представления объектов заключаются в ограниченности потенциальных форм объектов, в отличие от полигональных сеток, где формы могут быть какие угодно. К тому же, алгоритм крайне ограничен в трансформациях исходных объектов: вращения и увеличения размеров.

При вращении необходимо производить дополнительные сложные расчеты координат вписанных объектов, что потенциально убивает всю пользу от использования метода.

При увеличении размеров требуются значительные вычислительные способности при трансформациях объектов, особенно при увеличении размеров объекта по одной оси, требуя вычислений эллипсоида, что значительно сложнее сделать, чем со сферой.

В результате, данный алгоритм становится крайне ситуативным, при этом давая несопоставимо малые преимущества в скорости.

Тем не менее, есть иные способы выражения сложных объектов – в частности, параметрические модели. Их форма определяется формулами и вставляемыми в них параметрами, нежели полигональными сетками (рис. 2).

Преимущества параметрических моделей заключаются в значительной оптимизации используемой памяти по сравнению с полностью сохраняемыми полигональными сетками. Благодаря этому, особенности принимать невероятно сложные формы при использовании определенных формул, а также возможности анимировать сам коллайдер посредством привязки параметров коллайдера ко времени, делает параметрические модели эффективным объектом для применения в физических движках.

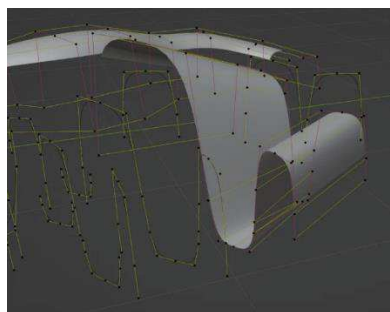


Рисунок 2 – Параметрическая поверхность, использующая кривые NURBS

Недостатки параметрических моделей заключаются в необходимости сложных вычислений формы объекта каждый раз, когда выполняются проверки на коллизию. Таким образом, получаем, что системы на основе примитивов быстры, но ограничены в своих формах. Напротив, параметрические модели имеют свободные формы, но крайне сложно вычисляются, чтобы применять их в алгоритме вычисления коллизий. Поэтому, для реализации алгоритма использована модель объектов, которые представляют собой множество простых объектов, при этом определяют свои формы с помощью формул с изменяемыми параметрами – фрактальные объекты.

Задачей нового алгоритма вычисления коллизий является использование фрактальных объектов в качестве поверхности движения по ним объектов.

Фрактальные объекты представляют собой примитивы, такие как сфера, тетраэдр или куб, которые подверглись множеству рекурсивных трансформаций. Таким образом, вычисление коллизий объектов сводится к вычислению столкновения примитива с другими объектами.

Разработанный алгоритм вычисления коллизий, основанный на принципах фрактального процедурного объекта позволяет создавать сложные и реалистичные формы объектов, а также обеспечивает гибкость и масштабируемость в процессе разработки игровых приложений. Алгоритм предоставляет возможность создания разнообразных фрактальных объектов с использованием различных параметров и правил. Это позволяет адаптировать алгоритм под конкретные требования и задачи игровых приложений

ЛИТЕРАТУРА

1. 3D Collision Detection: A Survey [Электронный ресурс]. – Режим доступа : https://www.researchgate.net/publication/220252331_3D_Collision_Detection_A_Survey.
2. Mdn Web Docs [Электронный ресурс]. – Режим доступа : https://developer.mozilla.org/en-US/docs/Games/Techniques/2D_collision_detection.
3. Collision Detection: Algorithms and Applications [Электронный ресурс]. – Режим доступа : <http://gamma-web.iacs.umd.edu/papers/COLLISION/wafr.pdf>.