

**СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ И АЛГОРИТМОВ,
ИСПОЛЬЗУЕМЫХ ДЛЯ КОМПЬЮТЕРНОГО СИНТЕЗА
ИЗОБРАЖЕНИЙ СЛОЖНЫХ КОСМИЧЕСКИХ СИСТЕМ**

Иванов Н.М, Олейник А. Г. у Цветков А. Б.

(ЦНИИМАШ, г. Королев)

В настоящее время для построения трехмерных изображений технология компьютерной графики основывается на использовании двух принципиально разных методов: метода трассировки лучей и метода Z-буфера.

В результате сравнительного анализа данных методов и их модификаций с учётом специфических требований, возникающих при проектировании и эксплуатации космических систем, в статье рассмотрены перспективы применения методов с учетом мировых тенденций развития программно-аппаратных средств компьютерной графики.

Показано, что современные требования к построению изображений сложных космических систем в реальном времени при существующих возможностях вычислительных ресурсов персональной вычислительной техники (в том числе бортовых ПК), требуют отказа от достижения максимальной реалистичности изображения в пользу повышения скорости построения изображения.

Применение метода трассировки лучей, а также метода Z-буфера, модифицированного с помощью метода излучательности, оценивается как наиболее перспективное направление максимально реалистичного синтеза изображений. Однако сделан вывод о невозможности реализации этих методов в настоящее время вследствие недостаточной вычислительной мощности — современных персональных компьютеров. Показано, что на данный момент получение приемлемых результатов на ПК возможно только с использованием аппаратных ускорителей, реализующих метод Z-буфера и буфер шаблона.

**СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ И АЛГОРИТМОВ,
ИСПОЛЬЗУЕМЫХ ДЛЯ КОМПЬЮТЕРНОГО СИНТЕЗА
ИЗОБРАЖЕНИЙ СЛОЖНЫХ КОСМИЧЕСКИХ СИСТЕМ**

Иванов Н.М, Олейник А.Г., Цветков А.Б.

(ЦНИИМАШ, г. Королев)

Методы компьютерного синтеза изображений являются одним из перспективных направлений использования современных информационных технологий для решения задач проектирования и эксплуатации сложных космических систем (КС). Ключевой проблемой внедрения компьютерных систем синтеза изображений является создание высокоскоростных алгоритмов построения изображения, обеспечивающих необходимые параметры работы системы в условиях ограниченности вычислительных ресурсов наземной и бортовой вычислительной техники.

В настоящее время для построения трехмерных изображений используются два принципиально разных метода: метод трассировки лучей и метод Z-буфера. Поэтому для решения задачи создания алгоритмов синтеза изображений, учитывающих специфические требования, возникающие при проектировании и эксплуатации КС, необходимо рассмотреть оба метода и выбрать один из них или их сочетание в качестве основы системы генерации изображений.

Для анализа и сравнения алгоритмов необходимо определить основные требования, предъявляемые к ним. Наиболее существенными являются следующие требования:

- работа в реальном времени;
- работа в условиях ограниченности вычислительных ресурсов;
- получение реалистичного изображения.

Расшифруем эти требования.

Под *работой в реальном времени* для алгоритма синтеза изображений подразумевается возможность создания динамического изображения, дающего представление о взаимном положении объектов и их движении в реальном масштабе времени.

Под *ограниченностью вычислительных ресурсов* подразумевается использование вычислительных ресурсов персонального компьютера, что позволяет значительно

расширить круг пользователей систем синтеза изображений. Как известно, главными ограничениями вычислительных ресурсов в задачах построения трехмерных изображений на ПК является: количество оперативной памяти и загрузка центрального процессора, так как при использовании больших рабочих станций или суперЭВМ проблема нехватки оперативной памяти или недопустимая загрузка процессора решаются путем добавления недостающих ресурсов.

Под *реалистичным изображением* подразумевается получение изображение по качеству близкое к оригиналу. Это требование является существенным при необходимости качественной оценки моделируемой ситуации. Кроме того, на реалистичность будет влиять количество точек в получаемом изображении и глубина цвета.

Таким образом, алгоритм синтеза изображений на персональном компьютере должен позволять строить динамические изображения близкие по качеству к оригиналу. Сюда же можно добавить требование о максимальном использовании аппаратных ускорителей, возможности которых растут с каждым годом.

Метод трассировки лучей

Метод трассировки лучей (ray tracing) [1,2] заключается в моделировании прохождения лучей в рамках геометрической оптики. Обычно для синтеза реалистичных изображений используется алгоритм Витеда, известный под названием алгоритма обратной трассировки лучей [3]. Алгоритм работает следующим образом. Из глаза наблюдателя через каждый пиксель картинной плоскости в сцену пропускается луч, и затем отслеживается в обратном направлении. Когда луч наталкивается на поверхность, интенсивность соответствующего пикселя картинной плоскости определяется освещенностью ближайшей точки пересечения луча и поверхности. Освещенность этой точки складывается из отраженной и преломленной энергии, непосредственно полученной от источников света, и отраженной и преломленной энергии, идущей от других объектов сцены. Для определения освещенности, порождаемой непосредственно освещением, из точки пересечения выпускаются лучи ко всем источникам света, и определяется вклад тех источников, которые не заслонены другими объектами. Для определения остальных величин выпускаются вторичные лучи (отраженный и преломленный) и определяется приносимая ими энергия. Для диффузных поверхностей необходимо выпускать пучок вторичных лучей, равномерно распределенный по всем направлениям.

Метод трассировки лучей наиболее подходит для сцен, содержащих точечные источники света и идеально зеркальные или прозрачные тела. В случае диффузных поверхностей метод работает неэффективно. Но даже в случае идеально зеркальных или

прозрачных тел один луч, выпущенный из глаза наблюдателя, может породить дерево, содержащее сотни или даже тысячи лучей. Причем лучи отдельно строятся для каждой из трех компонент цвета. Чтобы ограничить количество вторичных лучей вводят ограничение на глубину дерева, что сказывается на качестве изображения. При этом часть лучей будет уходить в окружающее пространство, не наталкиваясь ни на одну поверхность, что приведет к бесполезным вычислениям и неоправданному расходу памяти.

Таким образом, учитывая специфику синтеза изображений КС, а именно неограниченность окружающего их космического пространства, можно предположить, что большая часть вычислений, связанная с трассировкой лучей, не будет реально влиять на получаемое изображение. Следовательно, нельзя говорить об эффективности алгоритма. Другим недостатком, который можно считать основным, является зависимость от положения наблюдателя. Зависимость от положения наблюдателя объясняется тем, что процесс построения изображения происходит одновременно с расчетом освещенности. При увеличении размера изображения соответственно увеличивается объем вычислений и количество требуемой оперативной памяти, необходимых для построения изображений, так как луч из глаза наблюдателя проводится через каждую точку.

Например, для построения методом трассировки лучей изображения, состоящего из 10000 элементов размером 800x600 потребуется построить $48 \cdot 10^4$ первичных лучей. Затем определить какие элементы изображения пересекает каждый луч, что составит $48 \cdot 10^4$ проверок. Далее из каждой точки пересечения построить, по крайней мере, два вторичных луча и т.д.

Таким образом, к достоинствам метода трассировки лучей можно отнести большую степень реализма получаемых изображений и простоту физической интерпретации метода. Недостатком является неэффективность работы с диффузными поверхностями и зависимость от положения наблюдателя, что требует большого количества вычислительных ресурсов. Такой метод может быть реализован на многопроцессорных ЭВМ с параллельными вычислениями. Для ПК такой метод не будет соответствовать требованиям ограниченности ресурсов.

Метод Z-буфера

Метод Z-буфера ввиду простоты своего алгоритма и используемого в нем набора операций часто реализуется аппаратно. Временные характеристики этого метода линейно зависят от количества точек раstra и «глубины сложности сцены», т.е. усредненного числа граней, взаимно закрывающих друг друга. Для реализации метода используется две области памяти: буфер глубины (Z-буфер) и буфер кадра, хранящий информацию о состоянии

пикселей экрана компьютера. Буфер глубины используется для хранения координаты z (глубины). В буфере кадра запоминаются атрибуты соответствующего пикселя (цвет и прозрачность). Объекты сцены разлагаются в растр. После чего анализируется глубина каждого пикселя. Если она меньше значения в Z -буфере, то атрибуты пикселя заносятся в буфер кадра, а координата z в буфер глубины.

Метод Z -буфера при аппаратной реализации не требует дополнительной оперативной памяти ПК, так как для хранения буфера глубины и буфера кадра используется специальная память, являющаяся частью аппаратного ускорителя. Нагрузка на центральный процессор во время синтеза изображения также уменьшается, так как в аппаратном ускорителе используется специализированный процессор для трехмерной графики.

Главным недостатком Z -буфера является отсутствие возможности построения теней и отражений, что не позволяет получить реалистичного изображения.

Модифицированные методы Z -буфера

Одним из решений задачи получения реалистичного изображения является модификация метода Z -буфера с помощью метода излучательности (radiosity) [4, 5, 6, 7] или использования «буфера шаблона».

В основе метода излучательности лежит закон сохранения энергии в замкнутой системе. При построении изображения все объекты разбиваются на фрагменты, и для этих фрагментов составляется уравнения баланса энергии.

$$B_i = E_i + \rho_i \sum_{j=1}^n F_{ij} B_j, \quad i = 1, \dots, n \quad (1)$$

где

B_i - энергия, отбрасываемая i -м фрагментом сцены;

E_i - собственная излучательность фрагмента;

F_{ij} - доля энергии j -го фрагмента, попадающая на i -й фрагмент (коэффициенты формы, форм фактор, фактор формы);

ρ_i - коэффициент отражения.

В результате получается система линейных алгебраических уравнений (1), которая может решаться итерационными методами.

Для определения цвета фрагмента соответствующие системы записываются для каждой из трех цветовых составляющих, причем коэффициенты формы зависят только от геометрии сцены и не зависят от цвета.

Такой подход устраняет основные недостатки метода трассировки лучей, обеспечивая высокую точность при работе с диффузными поверхностями и отдельное вычисление освещенности независимо от положения наблюдателя.

Следовательно, система, построенная с использованием Z-буфера и метода излучательности, разделяет процесс расчета освещенности и построения изображения. Этот факт позволяет, предварительно рассчитав освещенность, менять точку наблюдения без дополнительных вычислений. Но только в случае неподвижности всех объектов сцены (к объектам необходимо отнести источники света). Далее возможны два случая: изменение оптических характеристик или взаимного положения объектов. В первом случае необходимо повторное решение системы уравнений (1) для каждой из трех компонент цвета. Во втором случае необходимо выполнить пересчет коэффициентов формы, зависящих от взаимного расположения объектов, что является наиболее трудоемким процессом. После чего решать систему уравнений (1).

Например, для построения изображения, состоящего из 10000 элементов размером 800x600, потребуется определение и хранение 10^8 коэффициентов формы и решение трех систем уравнений 10000x10000.

Пересчет коэффициентов формы и решение системы уравнений (1) является узким местом такого подхода, так как они требуют большого количества вычислительных ресурсов. В связи с этим, поиску алгоритмов нахождения коэффициентов формы и решения систем алгебраических линейных уравнений в рамках метода излучательности было посвящено большое количество исследований [8, 9], но на данный момент известные алгоритмы могут работать в реальном времени только на мощных многопроцессорных ЭВМ. Таким образом, метод Z-буфера совместно с методом излучательности может использоваться только в статических сценах, где объекты неподвижны. Для динамически меняющихся сцен такой способ построения изображений не приемлем из-за ограниченности вычислительных ресурсов.

Более простым вариантом является отказ от расчета глобальной освещенности объектов и построение только падающих теней и отражений. Основой такого подхода является использование «буфера шаблона» (stencil buffer), в котором хранится шаблон наложения тени или отражения. Существует возможность построения нескольких видов теней: плоской тени, объемной тени и гибридной тени. Наиболее простой является плоская тень, которая получается путем проецирования объекта-источника тени на плоскость. Такая

тень будет видна только на плоскости. В случае, когда между объектом-источником и тенью будет находиться другой объект, на него тень попадать не будет. Объемная тень устраняет этот недостаток, объединяя тень и источник тени в одно целое. Любой объект, попадающий между тенью и источником тени (т.е. в объем тени), оказывается затененным. Гибридная тень объединяет оба подхода и строится в зависимости от ситуации как плоская или объемная, что позволяет оптимизировать процесс построения тени. Для получения отражения объекта необходимо симметрично отобразить объект и источники света относительно плоскости отражения, и построить полученное отражение в буфере шаблона.

Использование буфера шаблона позволяет значительно упростить и ускорить процесс построения тени, так как буфер шаблона реализуется аппаратно, и основная нагрузка ложится на ускоритель.

Основной задачей при построении тени с использованием буфера шаблона будет определение ее границ. Задача возникает из-за того, что для построения тени достаточно определить ее контур, а при проецировании используются все точки объекта, что приводит к дополнительному расходу памяти и большим вычислительным затратам. (Для построения контура тени используется приблизительно 10-20% от общего количества точек, описывающих объект). Следовательно, для наиболее эффективной работы требуется оптимизация процесса построения тени, что возможно потребует разработки специальных алгоритмов, позволяющих быстро (в реальном времени), точно и с минимальными затратами памяти строить контур тени. В настоящее время широко используется метод построения выпуклой оболочки, которая используется в качестве контура объекта. Такой подход неэффективен, если исходный объект не выпуклый, что приводит к значительным искажениям получаемого изображения.

Такой подход является очень простым и может быть реализован за счет использования аппаратного ускорителя. Таким образом, использование аппаратных средств позволяет в реальном времени получать достаточно реалистичное изображение при допустимых затратах памяти и минимальной загрузке процессора во время непосредственной генерации изображения. Хотя процесс создания сцены, т.е. размещения объектов, производится с использованием центрального процессора. Но в ближайшем будущем предполагается, что и эта часть вычислений будет перенесена на аппаратный ускоритель трехмерной графики.

Оценки роста производительности центрального процессора строятся на основании закона Мура. Согласно этому закону, производительность процессора при неизменной стоимости удваивается каждые полтора два года. Закон Мура работает уже свыше тридцати лет. И хотя десятилетиями обозреватели предупреждали о снижении экспоненциального

роста производительности, по оценкам большинства экспертов закон Мура будет выполняться, по крайней мере, ближайшие 10 лет. Мур в своем законе имел в виду микропроцессоры общего назначения. Производительность специализированных графических процессоров, массовое применение которых началось в 1995 году, каждые полтора года уже дважды увеличивалась в восемь раз. Конечно, такой темп роста производительности не сможет долго сохраняться, но, очевидно, акценты смещаются в пользу графических ускорителей.

Проведенный выше анализ и сравнение методов, а также данные из других источников позволяют сделать следующие выводы. В настоящее время построение реалистичных изображений в реальном времени, используя только вычислительные ресурсы персонального компьютера, требует отказа от реалистичности в пользу скорости.

Применение наиболее реалистичного метода трассировки лучей и метода Z-буфера совместно с методом излучательности являются наиболее перспективными, так как основываются на физической модели распределения освещенности, но их реализация пока невозможна по причине недостаточной вычислительной мощности современных персональных компьютеров. На данный момент получение приемлемых результатов на ПК возможно только с использованием аппаратных ускорителей, реализующих метод Z-буфера и «буфер шаблона». Согласно прогнозам роста производительности ПК, использование метода трассировки лучей и метода Z-буфера совместно с методом излучательности для создания реалистичных изображений в реальном времени будет невозможно еще несколько ближайших лет в связи с недостаточной скоростью роста вычислительных ресурсов. В то же время более высокий темп роста производительности аппаратных графических ускорителей позволяет сделать вывод, что в ближайшее время, учитывая специфические требования проектирования и эксплуатации КС, для построения системы компьютерного синтеза изображений в реальном времени наиболее подходит модифицированный метод Z-буфера, использующий «буфер шаблона».

Список литературы

1. Вопросы кибернетики. Моделирование сложных систем и виртуальная реальность. 1995.
2. В.П. Иванов, А.С. Батраков. Трехмерная компьютерная графика. Радио и связь. 1995.

3. Whitted T. An improved illumination model for shaded display. Comm. of the ACM. 1980. V.23 #6. P.343-349.
4. Cohen, Michael, Donald P. Greenberg, "The Hemi-Cube: A Radiosity Solution for Complex Environments"
5. Chen, Shenchang Eric, "Incremental Radiosity: An Extension of Progressive Radiosity to an Interactive Image Synthesis System"
6. Cohen, Michael, Donald P. Greenberg, Dave S. Immel, Philip J. Brock, "An Efficient Radiosity Approach for Realistic Image Synthesis"
7. Baum, Daniel R., John R. Wallace, Michael F. Cohen, Donald P. Greenberg, "The Back-Buffer Algorithm: An Extension of the Radiosity Method to Dynamic Environments"
8. David W. George, A Francois X. Sillion Donald P. Greenberg, "Radiosity Redistribution for Dynamic Environments"
9. Baum, Daniel R., Holly E. Rushmeier, James M. Winget, "Improving Radiosity Solutions Through the Use of Analytically Determined Form-Factors", Computer Graphics (SIGGRAPH '89 Proceedings), vol. 23, no. 3, pp. 325-334, July 1989
10. Baum, Daniel R., James M. Winget, "Real Time Radiosity Through Parallel Processing and Hardware Acceleration", Computer Graphics (1990 Symposium on Interactive 3D Graphics), vol. 24, no. 2, pp. 67-75, March 1990.
11. Fujimoto A., T. Tanaka, K. Iwata, "ARTS: Accelerated Ray Tracing", IEEE Computer Graphics and Applications, vol. 6, no. 4, pp. 16-26, 1986.