

УДК 004.92

О. Л. Макарова, Ижевский государственный технический университет имени М. Т. Калашникова

АЛГОРИТМ АНАЛИЗА ЯЧЕЕК ТЕТРОИДНОЙ РЕГУЛЯРНОЙ СЕТИ ДЛЯ РЕАЛИЗАЦИИ РЕНДЕРИНГА 3D-ИЗОБРАЖЕНИЙ

Приводится алгоритм нахождения площади тени объемного тетраоида на опорную плоскость, что обеспечивает точную и эффективную реализацию рендеринга 3D-изображений пространственных объектов и сцен.

Ключевые слова: тетраоид, тень, TRN-модель.

Одной из основных проблем 3D-моделирования пространственных объектов и сцен в компьютерной графике является построение их моделей, которые обеспечивали бы эффективность оперирования ими, как в плане объемов потребляемой памяти и производительности вычислительных процессов, так и расширения функциональных возможностей при их использовании. Как установлено в работах [1, 2], одной из наиболее эффективных моделей является модель TRN [Tetroid Regular Network – тетраоидная регулярная сеть]. Однако при этом необходимо найти корректное решение ряда задач, одной из которых является расчет параметров элементов сети, определяющих результат их визуализации в 3D-изображении, что обеспечивает полноценное решение задачи рендеринга 3D-объектов и сцен.

Согласно [3] тетраоидом называется упорядоченная четверка $\langle P_1, P_2, P_3, P_4 \rangle$ точек пространства R^3 . Точки P_1, P_2, P_3, P_4 называются вершинами тетраоида, отрезки $L(P_1, P_2)$, $L(P_2, P_3)$, $L(P_3, P_4)$, $L(P_4, P_1)$ – ребрами, а отрезки $L(P_1, P_3)$ и $L(P_2, P_4)$ – диагоналями тетраоида. Замкнутая ломаная $L(P_1, P_2, P_3, P_4, P_1)$ образует контур тетраоида. Параллельные плоскости α и β , в которых лежат диагонали, называются опорными плоскостями тетраоида (рис. 1).

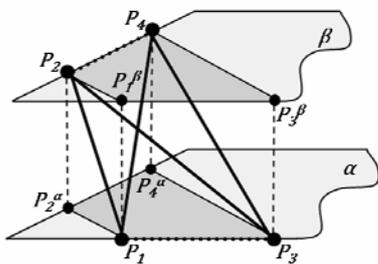


Рис. 1. Тетраоид

В дальнейшем используются следующие обозначения: α – плоскость, содержащая диагональ $L(P_1, P_3)$; β – плоскость, содержащая диагональ $L(P_2, P_4)$; P_1^β , P_3^β – проекции точек P_1 и P_3 на

плоскость β ; P_2^α , P_4^α – проекции точек P_2 и P_4 на плоскость α .

В зависимости от пространственного расположения точек P_1, P_2, P_3, P_4 различают следующие виды тетраоидов:

- точечные (нульмерные), контур вырождается в точку;
- линейные (одномерные), контур вырождается в отрезок, т. е. в пространстве R^3 существует прямая, на которой лежат все вершины тетраоида;
- планарные (двумерные), диагонали лежат в одной плоскости, т. е. $\alpha = \beta$, но тетраоид не является ни точечным, ни линейным;
- объемные (трехмерные), диагонали тетраоида лежат в параллельных плоскостях, т. е. $\alpha \neq \beta$.

В общем случае различают 86 видов тетраоидов, полная номенклатура которых описана в [3].

Тень тетраоида на опорную плоскость в общем случае представляет собой часть плоскости, ограниченную четырехугольником (тетрагоном). На рис. 1. это тетрагоны $P_2^\alpha P_4^\alpha P_3 P_1$ и $P_2 P_4 P_3^\beta P_1^\beta$. В случае вырождения тетраоидов тетрагоны могут вырождаться в тригоны, квазитригоны, ломаные, отрезки или точки.

Имеют место следующие очевидные утверждения.

Утверждение 1. Тенью любого точечного тетраоида на опорную плоскость является точка.

Утверждение 2. Тенью линейного тетраоида на опорную плоскость является точка или отрезок.

Следствие. Площадь тени любого точечного или линейного тетраоида на опорную плоскость равна 0.

Сложнее обстоит дело с тенями планарных и объемных тетраоидов.

Тень планарного тетраоида на опорную плоскость совпадает с соответствующим ему тетрагоном или может являться ломаной, тригоном, квазитригоном. Примеры теней вырожденных планарных тетраоидов приведены на рис. 2, невырожденных – на рис. 3.

Очевидно, что тетраоид вырождается в ломаную только в двух ситуациях: $P_3 = P_1$ или $P_2 = P_4$. В этом случае одна из диагоналей вырождается в точку. Имеет место еще одно очевидное утверждение.

Утверждение 3. Площадь тени вырожденного планарного тетраоида вида «ломаная» равна 0.

Тетроид вырождается в тригон, когда склеиваются две соседние вершины (диагонали имеют общую вершину) или одна из вершин лежит между двумя другими на прямой, образованной смежными ребрами (одна из диагоналей содержит вершину другой). В квазитригон тетроид вырождается, когда одна из вершин лежит на несмежном ребре (вершина одной из диагоналей лежит на продолжении другой диагонали).

Если ребра тетроида имеют общую точку, то имеет место случай невырожденного самопересекающегося тетроида (диагонали не пересекаются), иначе имеет место выпуклый (диагонали пересекаются) или невыпуклый (диагонали не пересекаются) тетроид.

Для объемных тетроидов список разновидностей теней будет совпадать со списком разновидностей теней планарного тетроида за исключением случая ломаной тени (рис. 4).



Рис. 2. Разновидности теней вырожденных планарных тетроидов



Рис. 3. Разновидности теней невырожденных планарных тетроидов

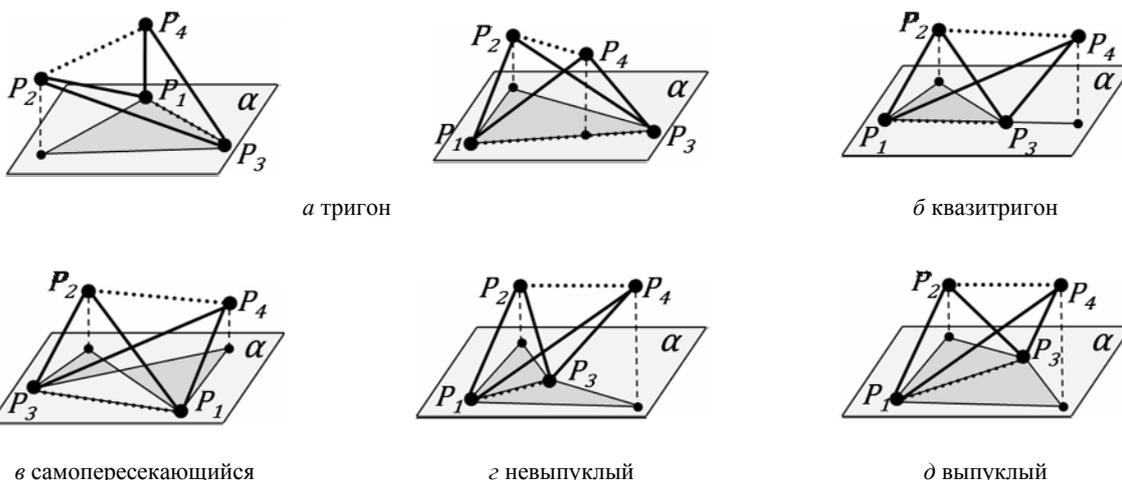


Рис. 4. Разновидности теней объемных тетроидов

Таким образом, задача нахождения площади тени сводится к нахождению универсального алгоритма, позволяющего вычислять площадь как плоского четырехугольника общего вида, так и тригона, и квазитригона.

Алгоритм реализуется в несколько этапов.

1-й этап. Нахождение проекций вершин тетроида на плоскость α .

В дальнейшем используются следующие обозначения (см. рис. 1). Координаты вершин тетроида и их проекций на плоскость α :

$$P_1(x_1, y_1, z_1), P_2(x_2, y_2, z_2), P_3(x_3, y_3, z_3),$$

$$P_4(x_4, y_4, z_4), P_2^\alpha(x_2^\alpha, y_2^\alpha, z_2^\alpha), P_4^\alpha(x_4^\alpha, y_4^\alpha, z_4^\alpha).$$

Направляющие векторы прямых, содержащих диагонали тетроида:

$$\overline{P_1P_3}(a_1, b_1, c_1) = \{x_3 - x_1, y_3 - y_1, z_3 - z_1\};$$

$$\overline{P_2P_4}(a_2, b_2, c_2) = \{x_4 - x_2, y_4 - y_2, z_4 - z_2\}.$$

Тогда вектор нормали к плоскости α есть

$$\overline{\Delta} = \overline{P_2^\alpha P_4^\alpha} = \{\Delta_1, -\Delta_2, \Delta_3\},$$

$$\text{где } \Delta_1 = \begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix}, \Delta_2 = \begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}, \Delta_3 = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}.$$

Координаты точек P_2^α и P_4^α определяются следующими выражениями (в числителе параметра t – скалярное произведение векторов):

$$P_2^\alpha : \begin{cases} x_2^\alpha = x_2 + \Delta_1 t, \\ y_2^\alpha = y_2 - \Delta_2 t, \\ z_2^\alpha = z_2 + \Delta_3 t, \end{cases}$$

где

$$t = \frac{(\overline{\Delta}, \overline{P_2 P_1})}{\overline{\Delta}^2} = \frac{\Delta_1(x_1 - x_2) - \Delta_2(y_1 - y_2) + \Delta_3(z_1 - z_2)}{\Delta_1^2 + \Delta_2^2 + \Delta_3^2},$$

$$P_4^\alpha : \begin{cases} x_4^\alpha = x_4 + \Delta_1 t, \\ y_4^\alpha = y_4 - \Delta_2 t, \\ z_4^\alpha = z_4 + \Delta_3 t, \end{cases}$$

где

$$t = \frac{(\overline{\Delta}, \overline{P_4 P_1})}{\overline{\Delta}^2} = \frac{\Delta_1(x_1 - x_4) - \Delta_2(y_1 - y_4) + \Delta_3(z_1 - z_4)}{\Delta_1^2 + \Delta_2^2 + \Delta_3^2}.$$

2-й этап. Преобразование системы координат таким образом, чтобы аппликаты всех вершин тени тетраоида обратились в 0, а вся тень целиком лежала выше оси абсцисс. Для этого необходимо, чтобы плоскость α совпала с одной из координатных плоскостей, например, $x'O'y'$.

Преобразование выполняется в несколько шагов.

Шаг 1. Параллельный перенос системы координат на вектор $\overline{OP_1}$:

$$\begin{cases} x' = x - x_1, \\ y' = y - y_1, \\ z' = z - z_1. \end{cases}$$

Таким образом, система $Oxyz$ преобразуется в систему $P_1x'y'z'$.

Шаг 2. Преобразование системы координат $P_1x'y'z'$ в систему $P_1x''y''z''$ (рис. 5).

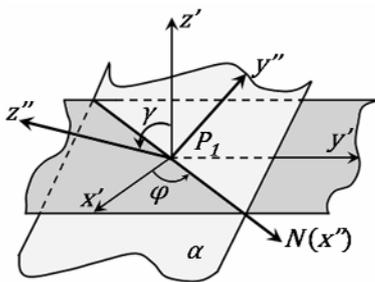


Рис. 5. Преобразование системы $P_1x'y'z'$: поворот на угол φ , поворот на угол γ

Прямая P_1N есть линия пересечения плоскостей α и $x'P_1y'$. Выбор направления оси P_1z'' определяется вектором нормали к плоскости α с учетом того, чтобы система оставалась правой. Первым выполняется поворот системы вокруг оси P_1z' на угол $\varphi = \angle x'P_1x''$, затем поворот вокруг оси P_1N на угол $\gamma = \angle z'P_1z''$. Каждый из поворотов определяется соответствующей ему матрицей:

$$M_\varphi = \begin{pmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad M_\gamma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{pmatrix}.$$

После выполнения поворотов аппликаты вершин тени тетраоида обнуляются.

Шаг 3. Параллельный перенос полученной системы координат на вектор $\overline{P_1b}$, где $b = \min\{y_i^n\}_{i=1}^3$. В результате получается новая система координат $O''x''y''z''$, в которой вся тень тетраоида лежит в координатной плоскости $x''O''y''$ выше оси $O''x''$.

3-й этап. Непосредственное вычисление площади.

Площадь невыпуклого многоугольника, заданного координатами своих вершин, вычисляется следующим образом.

Так как площади трапеции $S = \frac{1}{2}(a+b)h$, где a ,

b – основания трапеции, h – высота трапеции, то площадь многоугольника $ABCDE$ (рис. 6) равна $S = S_{ABNK} + S_{BCMN} - S_{CDM} + S_{DEL} - S_{EAKL}$.

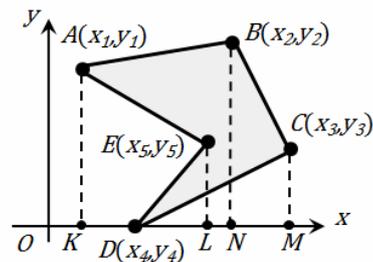


Рис. 6. Площадь многоугольника

Суммирование площадей производится по правилу: начиная и заканчивая в одной из вершин многоугольника, совершается последовательный обход по всем его вершинам, две последовательные вершины i и $i+1$ определяют трапецию, площадь имеет знак плюс, если $x_{i+1} \geq x_i$, и знак минус в противном случае. Отсюда площадь произвольного невыпуклого многоугольника: $S = \frac{1}{2} \left| \sum_{i=0}^n x_i (y_{i-1} - y_{i+1}) \right|$. При этом $(x_0, y_0) = (x_n, y_n)$.

Данный алгоритм позволяет вычислить площадь любого тетрагона, в том числе вырожденного (исключением является самопересекающийся тетрагон).

В случае самопересекающегося тетрагона достаточно найти точку пересечения диагоналей и добавить ее дважды в список вершин, изменив порядок обхода вершин (рис. 7), т. е. вычисляется площадь невыпуклого гексагона.

Таким образом, при выполнении описанных выше несложных преобразований системы координат и использовании алгоритма вычисления площади решается задача нахождения тени тетраоида на опорную плоскость без выяснения вопроса о ее разновидности.

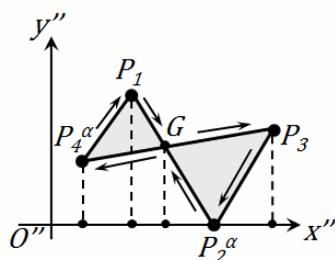


Рис. 7. Площадь самопересекающегося тетрагона

Для реализации эффективной схемы рендеринга трехмерных пространственных объектов и сцен необходимо в дальнейшем определять проекции теней тетраидов на плоскость наблюдателя, что позволит быстро вычислять параметры хроматизации соответствующих участков наблюдаемого изображения. В конечном счете, этим обеспечивается качественная, точная и эффективная реализация

рендеринга 3D-изображений пространственных объектов и сцен.

Библиографические ссылки

1. Елкин С. Л. Построение тетрагональной регулярной пространственно-деформируемой сетевой модели трехмерных объектов // Математическое моделирование и интеллектуальные системы : сб. науч. тр. ИжГТУ. – 2004. – № 1(3). – С. 27–29.
2. Макарова О. Л., Мурынов А. И., Сенилова Е. М. Тетраидная регулярная сеть как модель пространственных объектов размерности 3 // Материалы XXXI Междунар. конф. «Информационные технологии в науке, социологии, экономике и бизнесе». Украина, Крым, Ялта – Гурзуф // Прилож. к журн. «Открытое образование». – 2006. – С. 181–182.
3. Мурынов А. И., Левицкая Л. Н., Кошель О. Н. Анализ элементов тетраидной регулярной сети для формирования трехмерных моделей пространственных объектов. // Вестник ИжГТУ. – 2012. – № 4(56). – С. 135–142.

O. L. Makarova, Kalashnikov Izhevsk State Technical University

Algorithm of Analysis of Cells of Tetroid Regulated Network for Realization of Visualization of 3D-Images

This article describes algorithm for the calculation of area of tetroid shade on a supporting plane, that provides exact and effective visualization of 3d-images of spatial objects and scenes.

Keywords: tetroid, shade, TRN-model.

Получено 17.06.2014

УДК 004.8.681.3.06

А. Н. Афанасьев, доктор технических наук, Ульяновский государственный технический университет
В. С. Хородов, аспирант, Ульяновский государственный технический университет

ТЕХНОЛОГИИ РАСПРЕДЕЛЕННОГО ПРОЕКТИРОВАНИЯ VHDL-ОБЪЕКТОВ

Предложена мультиагентная распределенная система проектирования структурно-функциональных моделей, представленных на языке VHDL. Описаны роли агентов. Разработана реализационная структура системы на основе технологий SaaS и CORBA.

Ключевые слова: проектирование, распределенная система, мультиагентная система, технологии SaaS, CORBA.

В настоящее время проектирование аппаратных средств на базе высокоуровневых языков Hardware Description Language (HDL) в связи с усложнением устройств сталкивается с проблемой сложности проектов. Так, например, при разработке современных СБИС микропроцессоров объем кода на VHDL достигает нескольких сотен тысяч строк [1]. Решение проблемы основано на использовании парадигмы коллективного проектирования, перспективным направлением реализации которой является применение распределенных систем.

Место и роль описываемых в данной статье технологий, используемых при построении системы распределенного проектирования, показаны на рис. 1. Выделены два основных направления создания системы: разработка облачного сервиса (CLOUD_SFLM_CAD) и разработка локальной системы проектирования

структурно-функциональных моделей, представленных на языке VHDL [2] (SFLM_CAD). Отражено взаимодействие распределенных пользователей как с CLOUD_SFLM_CAD, так и с SFLM_CAD.

Слабым местом технологии Software as a Service (SaaS) являются вопросы безопасности и возможной утечки информации со стороны поставщика этих услуг [3]. Вопросы безопасности данных ограничивают использование концепции SaaS для систем, в которых обрабатывается конфиденциальная информация, например, продукты интеллектуальной собственности. В рассматриваемой системе таким продуктом является проектное решение. Обеспечение безопасности и конфиденциальности связано с созданием частного облака, которое вместе с локальной сетью является аппаратно-технологической основой корпоративной системы распределенного