

Министерство высшего и среднего специального
образования РСФСР

Куйбышевский ордена Трудового Красного Знамени
авиационный институт имени академика С.П.Королева

В.И.Александров

ГЕОМЕТРИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Учебное пособие

Куйбышев 1988

УДК 621.002:681.3

А.П.А л е к с а н д р о в. Геометрическое моделирование: Учебное пособие /Куйб. авиац. ин-т. - Куйбышев, 1988. - 95 с.

В учебном пособии излагаются основы геометрического моделирования объектов; рассматриваются виды геометрических моделей, способы их проектирования и представления в памяти ЭВМ, алгоритмы визуализации результатов геометрического моделирования средствами машинной графики.

Пособие предназначено для студентов специальностей 0646, 0647, изучающих курсы "Машинная графика", "Применение моделирования в САПР", а также может быть использовано студентами и специалистами, имеющими дело с системами автоматизированного проектирования машиностроительных объектов.

Ил. 40. Табл. 4.

Рецензенты: доц. И.В.В о р о н ц о в, доц. А.А.С о л о в о в

Печатается по решению редакционно-издательского совета института

© Куйбышевский авиационный институт, 1988.

ПРЕДИСЛОВИЕ

Методы геометрического моделирования имеют широкую область приложения. С геометрическими моделями приходится иметь дело математику и физику, инженеру и конструктору, ученому и рабочему, врачу и художнику. Особенно большое значение геометрические модели имеют в таких бурно развивающихся областях современной науки и техники, как автоматизированное проектирование, машинная графика, обработка изображений, машинное зрение, робототехника. Однако до сих пор не существует какого-либо систематического руководства, служащего введением в теорию геометрического моделирования. Объясняется это прежде всего тем, что сам термин *геометрические моделирование* и связанные с ним общие представления сравнительно недавно пришли в специальную литературу.

Сведения, относящиеся к области геометрического моделирования, рассматриваются, как правило, с узко специальной точки зрения во многих научных, технических и других изданиях и, чаще всего, недоступны широкому кругу читателей, к числу которых следует отнести и студентов. Поэтому одной из основных целей предлагаемого учебного пособия является рассмотрение в одном издании, хотя и в кратком изложении, возможно более широкого круга вопросов, связанных с построением и использованием геометрических моделей. Основной акцент сделан не на теоретическом обосновании применяемых методов, а на рассмотрении особенностей и возможностей их применения. Предпочтение отдавалось методам геометрического моделирования, описание которых либо вообще отсутствует в отечественной литературе, либо приведено только в периодических изданиях. В случаях, когда по рассматриваемому вопросу имеется специальная литература материал излагался кратко, и для более полного ознакомления читателя отсылался к соответствующему изданию.

В главе I собран материал по геометрическому моделированию в целом. Дается классификация геометрических моделей, приводятся краткие сведения из теории параметризации, рассматриваются условия правильного формирования геометрических объектов. 2-я глава содержит описание геометрических моделей с использованием функции принадлежности — рецепторная модель, октодеревя, симметричные и ассиметричные бинарные деревья. 3-я глава посвящена теоретико-графовой модели, когда объект представляется своими границами, состоящими из конечного числа граней, а каждая грань описывается набором ребер и вершин. Рассмотрены структуры данных для хранения модели и поисковые процедуры, позволяющие организовать поиск геометрических элементов в объекте. Дается краткое описание оператора Эйлера, используемых при синтезе новых объектов.

4-я глава содержит некоторые сведения об аналитических моделях. Поскольку это направление хорошо представлено в нашей научной литературе, в данную главу включены лишь сведения о B -сплайнах и вопросы описания бикубической поверхности по 16 точкам на ней. В 5 главе рассматривается параметрическая геометрическая модель объекта, когда описание объекта ведется через указание образца или аналога. Указано, как может быть преобразована форма образца. Для этого используется 4 типа операций: геометрические (аффинные) преобразования, теоретико-множественные операции, скругления углов и деформации.

Интересный вид геометрической модели, разработанной в последние годы и носящей название "о б о б щ е н н ы й ц и л и н д р", рассмотрен в 6-й главе.

В главе 7 рассматриваются методы визуализации результатов геометрического моделирования, которые включают приемы и алгоритмы построения изображений с различной степенью реалистичности — анализ видимости, построение теней, создание полутонных изображений.

Каждая глава сопровождается некоторым числом задач, предназначенных для проверки понимания читателем содержания главы. В пособии широко использованы материалы опубликованных работ различных авторов, в том числе и непереведенных на русский язык. Жанр учебного пособия заставил отказаться от ссылок, несущественных для хода изложения материала. Автор надеется, что это не будет воспринято как неуважение разработчиков или недооценка заимствуемых идей. В основу пособия положены лекции, читаемые автором студентам специальнос-

ти 0647 "Прикладная математика" со специализацией "Математическое обеспечение САПР".

Автор выражает благодарность коллегам по кафедре технической кибернетики за критические замечания и полезные обсуждения.

1.1. Основные определения и допущения

Расширенное использование вычислительной техники стимулировало растущий интерес к геометрическому моделированию при проектировании, производстве, контроле и исследовании объектов с помощью ЭВМ. Вычислительная машина становится средой, заменяющей традиционные инженерные чертежи и научные описания, поскольку в ней непосредственно могут храниться данные о трехмерной структуре физических объектов в отличие, например, от чертежей, где эти данные должны выбираться из двумерных проекций. При этом единственная модель может объединить всю информацию, обычно требующуюся для представления многих чертежей, страниц описаний и других инженерных данных.

Геометрические модели являются частным случаем моделей, необозримое число и разнообразие которых применяется в науке и технике. Особенностью геометрических моделей является то, что они доставляют и н ф о р м а ц и ю о форме и размерах предметов, их относительном положении в пространстве, и, если они движутся, о их траекториях в пространстве. Такая информация необходима во многих областях человеческой деятельности, однако, наиболее важно, если не центральное значение она имеет в системах автоматизированного проектирования (САПР), система машинного зрения (СМЗ), автоматизированном производстве.

Другой особенностью геометрических моделей является а б с т р а к т н о с т ь. Геометрическая модель воссоздает абстрактный образ объекта, наделенный свойствами, эквивалентными реальным свойствам отображаемого объекта. Геометрическая модель строится таким образом, чтобы отношение между ее элементами описывалось некоторыми уравнениями, физический смысл которых соответствовал бы отношениям реальных объектов. Изучение геометрических моделей дает возможность вскрыть такие взаимосвязи между объектами, которые трудно изучить на реальном объекте или физической модели.

Хранение геометрических данных в памяти ЭВМ дает целый ряд дополнительных преимуществ:

форма объекта находится в памяти машины в виде чисто цифровых данных, при этом отсутствуют сложности, связанные с деформацией чертежной бумаги или неточностями выполнения чертежей;

с помощью ЭВМ можно легко вычислить геометрические характеристики объекта, например, объемы его частей, очертания поперечных сечений или их площади;

с помощью ЭВМ легко выполняется переход от одних единиц измерения к другим;

с помощью графических устройств (дисплей, графопостроители) легко получить изображение, показывающее, как математически описанный объект будет выглядеть, если его рассматривать из любой интересующей нас точки пространства;

по данным, хранящимся в памяти ЭВМ, можно получить управляющую ленту для станка с числовым управлением. Это позволяет объединить посредством ЭВМ весь процесс производства изделий, начиная с проектирования и анализа и кончая получением готового изделия.

Многообразие форм и геометрических свойств реальных объектов и процессов не позволяет предположить единый, универсальный метод представления этих данных в форме геометрической модели.

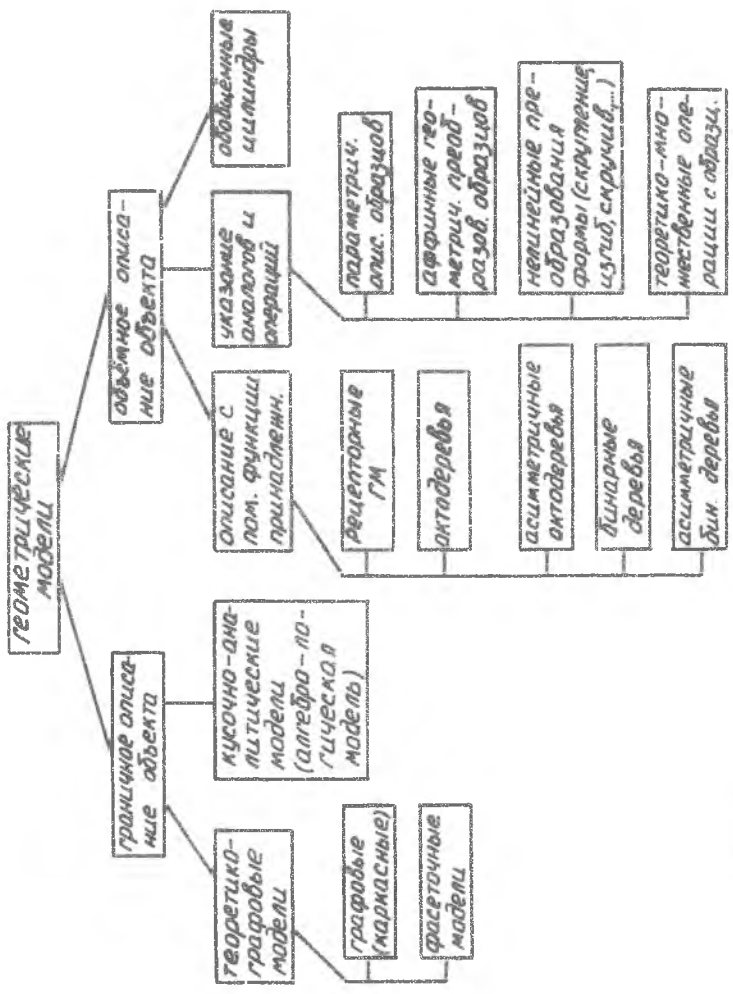
На рис. 1.1 дана классификация видов геометрических моделей, используемых в настоящее время в практике автоматизированного проектирования. Для каждого конкретного случая необходимо подбирать свой, наиболее подходящий для решения поставленной задачи способ представления геометрических данных. Наиболее важными аспектами здесь являются вид геометрической модели и уровень проектных возможностей. Вид геометрической модели определяет способ хранения данных и манипулирования ими при работе с системой геометрического моделирования. Подробное рассмотрение особенностей построения различных видов моделей дается в следующих разделах:

Уровень проектных возможностей указывает, что именно может быть изменено у объекта в процессе работы с системой геометрического моделирования. Обычно выделяется пять таких уровней [12]:

1. Уровень геометрических примитивов. Здесь основными "строительными блоками" являются вершины, ребра и грани, которые с помощью специальных операций комбинируются в единый объект.

2. Уровень поверхностей. Здесь каждая поверхность (каждая грань объекта) проектируется независимо от других.

3. Уровень трехмерных объектов. На этом уровне каким-либо об-



Р и с. I. I. Классификация геометрических моделей

разом описанная форма может изменяться преобразованием масштаба, поворота и сдвига в трехмерном пространстве.

4. Уровень операций с формами. Здесь пары форм могут комбинироваться в более сложные объекты с помощью, например, пространственных теоретико-множественных операций объединения, пересечения и разности.

5. Уровень сложных изделий. На этом уровне множества форм могут группироваться вместе в сложные иерархические структуры.

Следует добавить, что если геометрические параметры изменяются во времени, то уместнее говорить не об объектах, а о процессах, в которых более важным моментом является движение моделируемых объектов или изменение их формы.

Геометрическая модель описывает объект с помощью значений параметров. Формально можно представить объект следующей шестеркой [11]:

$$\{T, D, O, V, R, C\},$$

где T - множество топологических параметров;

D - множество размеров;

O - множество допустимых операций;

V - множество переменных;

R - набор отношений между V и элементами множеств T, O, D ;

C - множество ограничений.

Топологические параметры определяют качественные отношения между геометрическими элементами объекта или различными объектами. Примерами топологических высказываний являются: "поверхности ограничивают тело", "объект находится внутри другого объекта".

Размеры в геометрической модели дают информацию о длине ребер, координатах вершин, величине углов и т.п. Размеры определяют конкретную форму в рамках определенной топологии. Часть размеров может быть задана неявно.

Переменные включают в себя те параметры, которые могут изменяться при работе с геометрической моделью. В число переменных могут быть включены все или часть размеров топологических параметров.

Множество операций определяет последовательность допустимых преобразований существующих форм. Это могут быть простые геометрические преобразования (поворот, масштабирование, сдвиг), или более сложные теоретико-множественные операции (пересечение, объединение, разность).

Ограничения используют для приближения свойств геометрической модели к свойствам реального объекта. Наиболее часто используются ограничения на область определения переменных и на значения размеров.

Наконец, связи переменных друг с другом определяются с помощью набора отношений. Часто геометрическая модель включает некоторую область трехмерного пространства во всеми попадающими в нее объектами. В этом случае уместно разделить все параметры на две группы:

- параметры формы, которые характеризуют форму объекта и не изменяются при изменении положения объекта в пространстве;
- параметры положения, определяющие положение фигуры в пространстве независимо от ее формы.

Общее число параметров фигуры равно сумме числа параметров формы и параметров положения. Вопросами описания фигур с помощью наборов параметров занимается новый раздел геометрии — теория параметризации. Появление теории параметризации связано с трудами академика Н.Ф.Четвертухина (1964–1965 гг.) и развита И.И.Котовым, Н.Н.Рыжовым и др.

1.2. Условия правильного образования геометрических форм

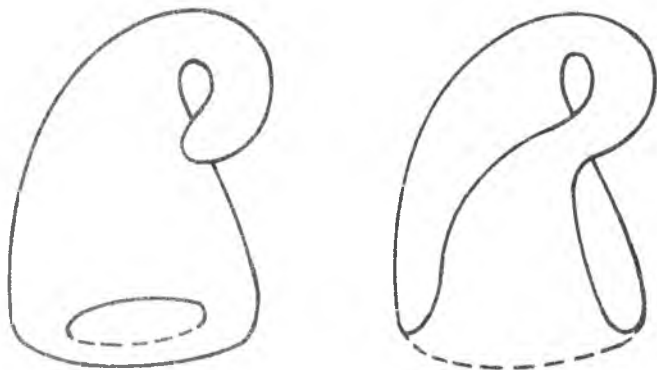
При конструировании трехмерных объектов необходима уверенность в том, что они правильно сформированы. Важно различать объемное и граничное описание. Объемное описание формы включает все точки пространства, принадлежащие объекту. Граничное представление имеет дело только с подмножеством полного множества точек объекта, лежащих на его поверхности.

Правильно сформированные трехмерные объекты должны удовлетворять условиям замкнутости, ориентированности, отсутствию самопересечений, ограниченности и связности /1/.

Замкнутые поверхности являются непрерывными, не имеют изломов, хотя это не исключает наличия отверстий через тело объекта. Это условие не допускает "открытых ящиков" и графовых сетей.

Замкнутость соответствует некоторому условию, налагаемому на связи между соединениями и топологическими элементами, а именно — каждое ребро связывает точно две грани и две вершины. Замкнутость ведет к тому, что контур каждой грани включает равное число ребер и вершин и, аналогично, периметр каждой вершины включает равное число ребер и граней (кроме специального случая для вершины конуса, когда одна вершина окружена одной гранью, и наоборот).

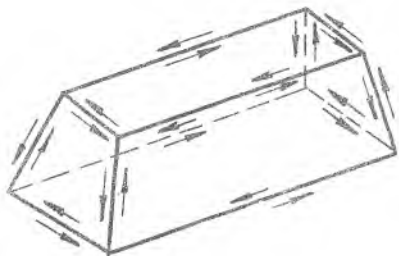
О р и е н т и р о в а н н о с т ь поверхности означает, что она имеет две различные стороны (возможно помеченные как внутренняя и внешняя). Замкнутость не является достаточным условием для ориентированности). Некоторые поверхности, рассматриваемые в математической топологии, например "бутылка Клейна (рис. I.2) не имеют изломов (хотя и самопересекаются), но они не ориентированы.



Р и с. I.2. Бутылка Клейна — пример самопересекающейся, но не ориентированной поверхности

В XIX столетии Мебиусом предложено правило для определения того, ориентирована или нет замкнутая поверхность многоугольника. Ребра каждой грани помечаются стрелками так, что внутренность грани остается справа при движении в направлении, указанном стрелкой. В замкнутой фигуре поверхность ориентирована, если только каждое ребро будет помечено двумя стрелками, направленными в разные стороны (рис. I.3).

Отсутствие самопересечения является, очевидно, желательным свойством поверхностей, ограничивающих реальные объекты.



Р и с. 1.3. Пример к правилу Мебиуса определения ориентированной поверхности

Топологические поверхности такие, как "бутылка Клейна", не могут быть реализованы в трехмерном евклидовом пространстве без самопересечений. Следует отметить, что не существует строгого топологического условия, предотвращающего самопересечение. Для некоторых систем автоматизированного проектирования самопересекающиеся формы запрещаются, в то время как в других они допустимы.

Замкнутые, ориентированные и несамопересекающиеся поверхности будут ограничены, что задает деление трехмерного пространства на 2 несвязные области, одна из которых конечна (если отбросить условие отсутствия самопересечения, то возможно более чем две результирующие области). Хорошее правило определения ограниченности ориентированной поверхности состоит в проведении луча из произвольной точки в бесконечность. Двигаясь по линии в бесконечность, прибавляем единицу всякий раз, когда линия пересекает поверхности изнутри наружу, и вычитаем единицу в противном случае. Положительно ориентированная поверхность делит пространство на две области — внутреннюю, точки которой дают результат "+1", и внешнюю, для точек которой результат равен "0".

Поверхность может быть замкнутой, но включать две или больше несвязных частей. Форма полого объекта имеет две части, внешнюю поверхность и внутреннюю полость. Связность поверхности объекта нарушена. Такой объект может рассматриваться как результат логической комбинации двух объектов. Если такая ситуация нежелательна, то добавлением искусственного разреза можно связать внутреннюю и внешнюю поверхности, чтобы получить единственную поверхность. Связность, несамопересекаемость и ориентированность являются общими свойствами, зависящими от всех частей поверхности.

Полезное для практических применений правило было предложено Эйлером, оно устанавливает взаимосвязь между элементами многогранника: $F + V - E = 2 + R - 2H$, где E — число ребер, F — число граней, V — число вершин, R — число ребер во внутренних кольцах, H — число отверстий через тело.

Практическое значение рассматриваемых здесь различных условий состоит в придании уверенности в том, что создаваемые формы правильно спроектированы и имеют смысл как самостоятельные объекты. Конечно, для некоторых целей необходимо ослабить эти условия, например, можно допускать неполные тела такие, как "открытые ящики" (просто набор граней, которые не образуют замкнутой поверхности). Однако эти наборы должны отличаться от тел с замкнутыми объемами, чтобы не допускать нежелательных операций с ними.

Большинство сформулированных условий справедливо для многогранников с плоскими гранями. Естественно, что не всегда криволинейные грани удовлетворяют всем условиям для плоских граней. Например, можно рассматривать вершину конуса как изолированную вершину без ребра, и основание конуса или цилиндра как ребро без вершин. Однако для большинства практических целей вероятно полезней преобразовать их путем включения дополнительных компонентов так, чтобы они удовлетворяли топологическим условиям.

Система геометрического моделирования обязательно должна обеспечить проверку правильности формы всех создаваемых объектов. Наиболее важной причиной этого является возможность итеративной обработки форм без участия человека. Предварительная, до проведения каких-либо операций с формой, проверка правильности поможет избежать потенциальных ошибок, которые могут возникнуть на любом шаге, и избежать распространения этих ошибок на последующие операции с формами.

2. ОПИСАНИЕ ОБЪЕКТОВ С ПОМОЩЬЮ ФУНКЦИИ ПРИНАДЛЕЖНОСТИ

2.1. Рецепторная модель

Наиболее общим формальным представлением твердого тела является непрерывное множество точек трехмерного пространства. Объект в этом случае задается простым перечислением принадлежащих ему точек.

На практике трехмерное непрерывное пространство представляется целочисленным массивом ячеек $(j\sigma, k\sigma, l\sigma)$, где j, k, l - целые числа в интервале от 0 до $N-1$ и σ - единичное расстояние (размер ячейки). Ячейка $V = (V_1, V_2, V_3)$ из этого массива иногда называется *цифровой точкой* или *воксе* -

лем. С каждой цифровой точкой V связываются те точки $X=(X_1, X_2, X_3)$ непрерывного пространства, которые удовлетворяют неравенствам $V_i - \delta/2 \leq X_i \leq V_i + \delta/2$, $i = 1, 2, 3$.

Трехмерный объект определяется с помощью характеристической функции $f(V)$, областью определения которой является множество всех вокселей, а область значений — множество $\{0, 1\}$. Множество вокселей $S = \{V | f(V) = 1\}$ называется объектом, а множество $\bar{S} = \{V | f(V) = 0\}$ — фоном. Геометрическая модель такого типа называется рецепторной или однородной [7].

Размер ячейки δ определяет точность представления объекта. Если грани объекта не параллельны осям координат или являются криволинейными, то представляются "ступеньками", величина которых равна размеру ячейки.

Рецепторная модель имеет два важных достоинства в представлении формы объекта:

легкость пространственной адресации, т.е. выделения данной точки пространства;

гарантированная пространственная единственность, т.е. исключаются случаи одновременного расположения двух объектов в одной области пространства.

Имеются также два недостатка:

нет явной информации об отношениях между частями объекта (о связности объекта);

необходим большой объем памяти для хранения данных об объекте в рамках рецепторной модели (при хорошей разрешающей способности); для описания формы, например, стола размером $1200 \times 600 \times 700$ мм с точностью 1 мм потребуется память $1200 \times 600 \times 700 \approx 5 \cdot 10^8$ бит или 63 мегабайт, что вряд ли приемлемо на практике. По этой причине рецепторные модели в настоящее время редко используются.

2.2. Геометрические модели с дзевовидными структурами данных

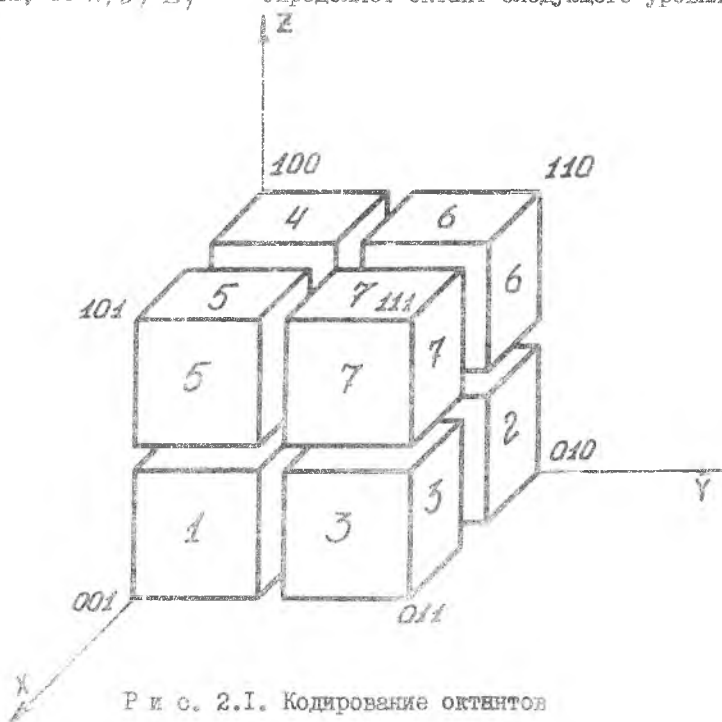
Октодеревья. В этом методе используется деление трехмерного пространства на восемь октантов равного объема. Каждый из таких октантов может быть либо однородным, если все его точки одного типа (т.е. либо все точки принадлежат объекту, либо все точки не принадлежат объекту), либо неоднородным, когда только часть точек октанта принадлежит объекту. Неоднородные октанты, в свою очередь, делятся на октанты меньшего объема. Эта процедура повторяется до

тех пор, пока не будут получены области, возможно из единственного voxеля с однородными свойствами. Поскольку метод может быть представлен моделью в виде дерева степени 8, то он назван октодеревом [21].

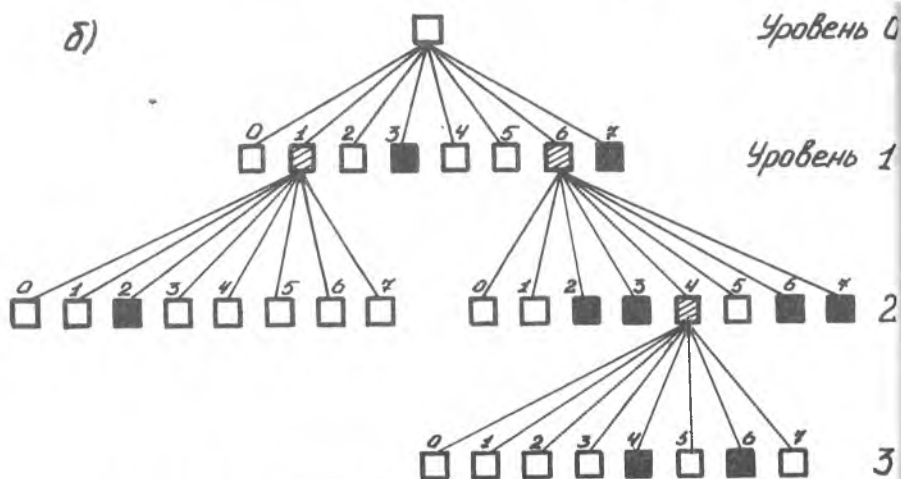
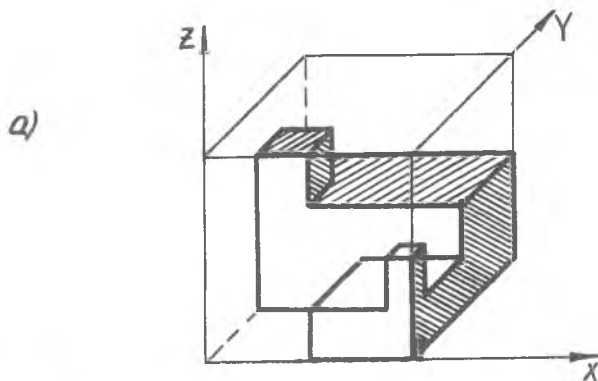
Каждый нетерминальный узел октодерева имеет восемь дочерних узлов, листья октодерева соответствуют элементам данных о дереве. Для того чтобы указать отдельную точку объекта (X, Y, Z) необходимо получить двоичное представление координат X, Y и Z :

$$z_0 z_1 z_2 \dots z_{M-1}, y_0 y_1 y_2 \dots y_{M-1}, x_0 x_1 x_2 \dots x_{M-1}$$

где $M = \log_2 N$. Верхний уровень дерева представлен массивом из восьми элементов, каждый из которых соответствует определенному октанту (рис. 2.1). Индекс для этого массива, иногда называемый дочерним типом, определяется старшими битами x_0, y_0, z_0 двоичного представления координат X, Y, Z . Таким образом, число $z_0 y_0 x_0$ определяет наибольший октант. Если этот октант далее делится, то x_1, y_1, z_1 определяет октант следующего уровня и т.д.



Р и с. 2.1. Кодирование октантов



Р и с. 2.2. Пример описания трехмерного объекта с помощью октодеревя: а - исходный объект; б - октодеревя

На рис. 2.2. показан пример описания многогранника с помощью октодеревя.

Асимметричные октодеревья. Если пространство делить таким образом, что октанты получаются неравного объема (не кубы, а параллелепипеды), то такая модель называется асимметричным октодеревом [21]. Плоскости деления по-прежнему перпендикулярны осям координат, однако расстояние от начала координат (т.е. центра разделяемого объема) до плоскости деления различно

для каждой координатной оси. Умелый выбор положения плоскостей деления может сильно сократить требуемый для хранения данных объем памяти. Например, если очень маленький объект расположен в средней области большого пустого объема, то модель с равномерным делением (т.е. симметричное октодереве) потребует большого числа делений и, соответственно, многих уровней в дереве, прежде чем дело дойдет до мелких деталей объекта. С другой стороны, в случае асимметричного октодерева можно использовать только две плоскости деления вдоль каждой координатной оси, вплотную расположенную по бокам объекта.

Каждый нетерминальный узел асимметричного октодерева требует трех дополнительных полей для записи положения плоскостей деления вдоль осей X , Y и Z . Пусть (X_{jk}, Y_{jk}, Z_{jk}) являются индексами плоскостей деления K -го нетерминального узла на уровне j , где $0 < K < 7$ и $0 < j < \log_2 N - 1$. Тогда точка (X, Y, Z) может быть найдена, если при движении в каждом нетерминальном узле выбирать ветвь (путь), определяемую битами числа $Z_j Y_j X_j$, в котором

$$x_j = \begin{cases} 1 & ; \text{ если } X_{jk} < X \\ 0 & ; \text{ в противном случае} \end{cases}$$

$$y_j = \begin{cases} 1 & ; \text{ если } Y_{jk} < Y \\ 0 & ; \text{ в противном случае} \end{cases}$$

$$z_j = \begin{cases} 1 & ; \text{ если } Z_{jk} < Z \\ 0 & ; \text{ в противном случае} \end{cases}$$

Бинарное дерево. Геометрическая модель в виде бинарного дерева получается при делении заданного объема — прямоугольного параллелепипеда — только на две равные половины с помощью плоскости, перпендикулярной одной из осей. Деление производится до тех пор, пока каждый результирующий параллелепипед не станет однородным. Каждый нетерминальный узел бинарного дерева имеет индекс, указывающий ось координат, вдоль которой производится деление $/2I/$. Такая структура данных удобна при поиске в многомерном пространстве.

Следует отметить, что, в отличие от октодерева, одному и тому же объекту может соответствовать несколько различных бинарных деревьев.

Асимметричное бинарное дерево. В асимметричном бинарном дереве

плоскость деления может располагаться на любом расстоянии от края параллелепипеда (параллельно грани), а не только в его центре. Описание асимметричного бинарного дерева подобно обычному (симметричному) бинарному дереву, но дополняется полем, в котором хранится информация о положении плоскости деления $/2I/$.

Действия с октодеревьями. Структура данных в виде октодеревьев хорошо согласуется с такими геометрическими преобразованиями, как поворот на 90° . и масштабирование с коэффициентом, равным 2^n , $n = \pm 1, \pm 2, \dots$.

Поворот объекта на 90° вокруг одной из осей координат приводит к перестановке дочерних узлов в октодереве, что можно быстро выполнить путем изменения значений указателей в структуре данных. Увеличение (или уменьшение) объекта в два раза приводит к переносу корневого узла в одну из дочерних вершин (или соответственно, с одной из дочерних вершин в родительский узел). Увеличение в четыре раза приводит к переносу корня на два уровня ниже и т.д.

Рассмотрим движение блока единичного размера параллельно оси x на единичное расстояние в направлении увеличения координаты $/29/$. Тогда, блоки, занимающие положение октантов с кодами 0, 2, 4, и 6, перейдут в октанты с кодами соответственно 1, 3, 5 и 7. (см. рис. 2.1). Таким образом, новые коды получаются прибавлением единицы к старым кодам. Аналогично изменяются узлы, соответствующие октантам 1, 3, 5, 7 при движении в обратном направлении на единичное расстояние. Это означает, что новые коды получаются прибавлением 7 к старым кодам, причем сложение выполняется по модулю 8. Однако, новые октанты имеют уже другие "родительские узлы", так как родительский узел в октодереве тоже двигается. В зависимости от положения в октодереве узла, который начинает двигаться, изменения кодов могут затронуть узлы на любой высоте дерева.

Сдвиг параллельно оси y отличается лишь тем, что коды изменяются на 2 или 6 (сложение по модулю 8). Аналогично, сдвиг вдоль оси z изменяет коды на 4. В табл. 2.1 перечислены необходимые изменения и новые значения кодов узлов, когда узел смещается на единичное расстояние в положительном направлении вдоль каждой из трех осей (сложение ведется по модулю 8).

Когда новый код становится больше 7, т.е. 10, 11, ..., то единица переноса в старший разряд показывает, что родительский узел должен быть также передвинут в соседний октант в направлении движения исходного элемента.

Т а б л и ц а 2.1

Первоначальный код	Величина и новое значение кода при движении на единичное расстояние вдоль оси		
	+ x	+ y	+ z
0	+ 1 = 1	+ 2 = 2	+ 4 = 4
1	+ 7 = 10	+ 2 = 3	+ 4 = 5
2	+ 1 = 3	+ 6 = 10	+ 4 = 6
3	+ 7 = 12	+ 6 = 11	+ 4 = 7
4	+ 1 = 5	+ 2 = 6	+ 4 = 10
5	+ 7 = 14	+ 2 = 7	+ 4 = 11
6	+ 1 = 7	+ 6 = 14	+ 4 = 12
7	+ 7 = 16	+ 6 = 15	+ 4 = 13

Вопросы и упражнения

1. Построить октодереву пирамиды, вершины которой находятся в точках с координатами $(0,0,0)$, $(16,0,0)$, $(0,16,0)$, $(0,0,16)$. Нарисовать объекты, соответствующие промежуточным уровням октодерева, считая частично заполненные октанты ("серые") пустыми.

2. Построить октодереву треугольной прямой призмы, вершины которой находятся в точках с координатами $(0,0,0)$, $(32,0,0)$, $(0,16,0)$, $(32,16,0)$, $(0,0,16)$, $(32,0,16)$. Нарисовать объекты, соответствующие промежуточным уровням октодерева, считая частично заполненные октанты ("серые" узлы) полными.

3. Определить величину изменения кода для каждого узла октодерева при сдвиге объекта на единичное расстояние вдоль осей x, y или z в отрицательном направлении.

4. Определить изменения кодов узлов октодерева при поворотах объекта против часовой стрелки на 90 , 180 и 270° вокруг осей x, y или z .

5. Объект задан октодеревом (белые узлы опущены)

0. Серый

7. Серый

0. Серый

2. Черный

6. Серый

0. Серый	0	}	Черный
	1		
	2		
	3		
I. Серый	0	}	Черный
	1		
	2		
	3		
2. Серый	0	}	Черный
	1		
	2		
	3		
3. Серый	0	}	Черный
	1		
	2		
	3		

Определить октодереву того же объекта после его сдвига на $(44, -5, 1)$.

3. ТЕОРЕТИКО-ГРАФОВОЕ ОПИСАНИЕ ОБЪЕКТОВ

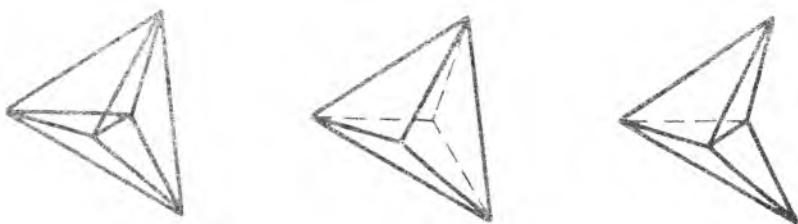
3.1. Определения и допущения

В рамках этой модели объект представляется своей границей, которая состоит из конечного числа граней и каждая грань описывается набором ребер и вершин. В графовой модели четко разделяется геометрическая и топологическая информация. Топологическая информация определяет связи между элементами объекта (вершинами, ребрами грани и т.д.). Геометрическая информация определяет форму, положение и ориентацию элементов объекта.

Графовые модели имеют несколько важных качеств. Они являются полными, так как определяют объект единственным образом. Модели применимы для всех ограниченных замкнутых и регулярных объектов. Кроме

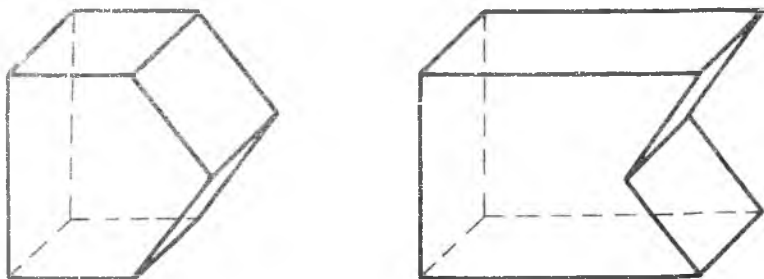
того, для визуализации объекта можно использовать различные простые и эффективные алгоритмы машинной графики.

Геометрическая и топологическая информация в равной степени важна для описания объектов. Если мы имеем дело с выпуклым объектом, то задание геометрической информации определяет однозначным образом форму. Однако, если объект не является выпуклым, то этой информации недостаточно для однозначного описания фигуры. Например, пять вершин могут быть соединены двумя способами (рис. 3.1). Аналогично семь плоскостей определяют две различные формы (рис. 3.2).



Р и с. 3.1. Неоднозначность соединения вершин многогранника

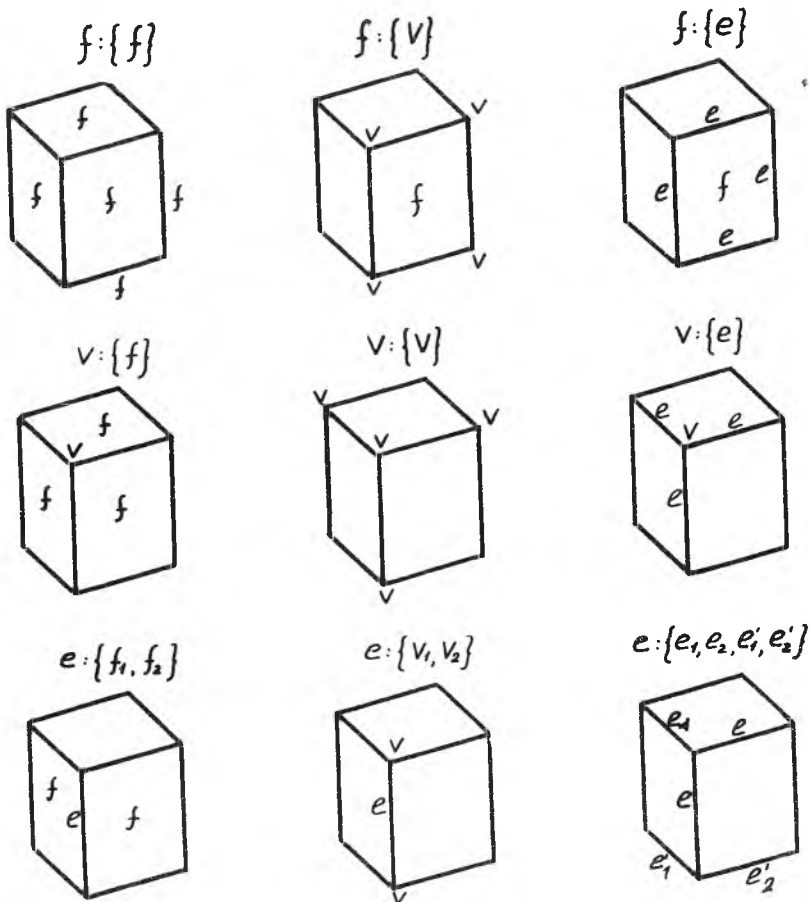
Таким образом, для объектов общего вида помимо геометрической информации необходимо указание на то, каким образом грани вершины и ребра соединяются между собой.



Р и с. 3.2. Неоднозначность описания многогранника при перечислении плоскостей

Обычно топологическая информация задается списком ребер, а каждое ребро определяется номерами инцидентных вершин. Номера вершин берут из списка вершин объекта, в который обычно включаются и

координаты вершин (геометрическая информация). Описание объекта такого типа обозначим как $e: \{v_1, v_2\}$. Однако, это не единственный способ задания топологической информации в каркасной модели. Существует девять вариантов описания связей между элементами трехмерных форм. Они представлены на рис. 3.3.



Р и с. 3.3. Варианты описания топологии многогранников

Очевидно, что хранение всех этих связей дает большую избыточность. Достаточно хранить данные о топологических связях только одного типа, а из них могут быть получены любые другие описания. Какой тип описания является лучшим, зависит от применения. Для векторной графики предпочтительными являются описания типов $v: \{v\}$, $e: \{v\}$ или $f: \{v\}$, поскольку необходимо знать, как соединяются вершины. Операции с формами (преобразования) обычно имеют дело с множеством граней, окружающих вершину, поэтому здесь более удобными окажутся описания типа $v: \{f\}$ или $f: \{f\}$. Описание типа $f: \{f\}$ позволяет без использования какой-либо геометрической информации определить число вершин, ребер, граней, отверстий в объекте.

Часто оказывается желательным иметь одновременно описания нескольких различных типов.

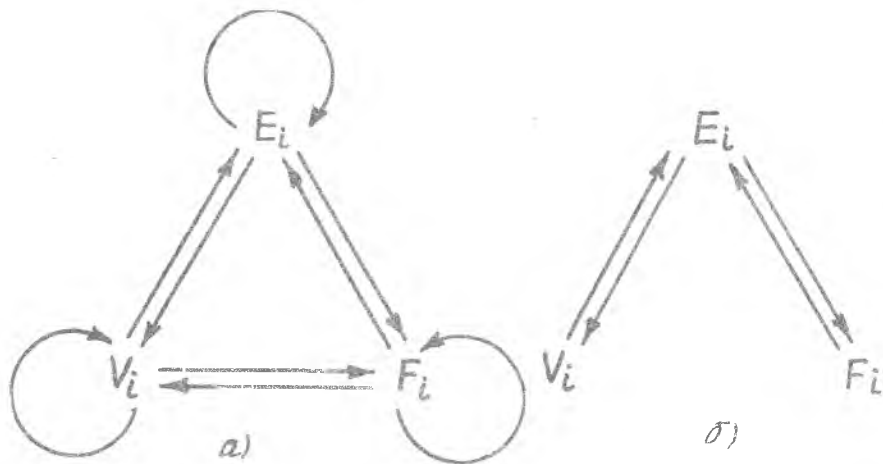
3.2. Структура данных для хранения описания объекта

Структура данных об объекте включает в себя записи с информацией об элементах объекта, т.е. его вершинах, ребрах и гранях, а также набор указателей, связывающих эти описания между собой. В обобщенной форме такую структуру данных можно представить в виде графа с тремя узлами V_i , E_i и F_i [5]. Каждый из этих узлов включает геометрическую информацию о положении, величине и кривизне. Узлы связаны между собой направленными дугами, как показано на рис. 3.4.

Каждая дуга определяет топологическую информацию о связности, соседстве и граничности. Например, дуга от E_i к V_i соответствует указателям от двух вершин некоторого ребра. Дуга от E_i к F_i соответствует указателем от данного ребра ко всем смежным ребрам. Учитывая двойственность отношения между V_i и F_i , можно определить симметричную структуру данных, показанную на рис. 3.4,б.

Дугу от узла X к узлу Y , т.е. $X \rightarrow Y$, можно интерпретировать как "дано X , найти Y ". Например, $E_i \rightarrow V_i$ соответствует задаче: "дано ребро, найти концевые вершины", а $V_i \rightarrow VG_i$ - задаче "дана вершина, найти инцидентные ей ребра".

Назовем группой вершины VG_i группу упорядоченных ребер, инцидентных вершине V_i . Например, для каждой вершины куба группа вершин содержит три ребра, а группа

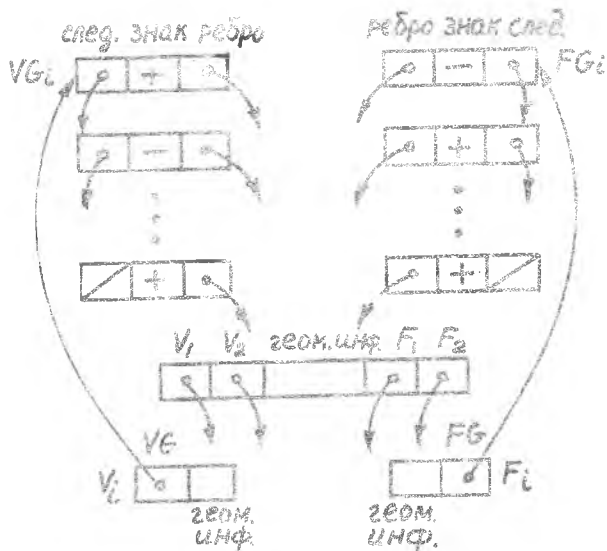


Р и с. 3.4. Обобщенная схема для структуры данных о трехмерном объекте: а — общая схема, б — симметричная форма

n -сторонней пирамиды содержит n ребер. Аналогично, назовем группой грани FG_i упорядоченный список ребер, вокруг данной грани F_i . Грань с отверстиями имеет более одной группы.

Рассмотрим пример реализации данной схемы в структуре данных, показанной на рис. 3.5. Она включает записи пяти типов, содержащих информацию о вершинах V_i , гранях F_i , ребрах E_i , группах вершин VG_i и группах граней FG_i .

Записи для вершин V_i имеют два поля, одно из которых содержит указатель на группу вершины VG_i , а другое — на запись, содержащую геометрическую информацию, т.е. координаты x_i, y_i, z_i . Запись для группы вершины VG_i включает три поля: первое содержит указатель к следующей записи из списка VG_i , второе указывает направление перехода от вершины V_i к другой вершине V_j (по данному ребру), и третье поле содержит указатель на запись из списка ребер E_i . Использование знака перехода позволяет избежать двойного считывания одной и той же вершины при поиске всех вершин, инцидентных данной вершине. Записи для ребер содержат пять полей: два поля являются указателями на концевые вершины дан-



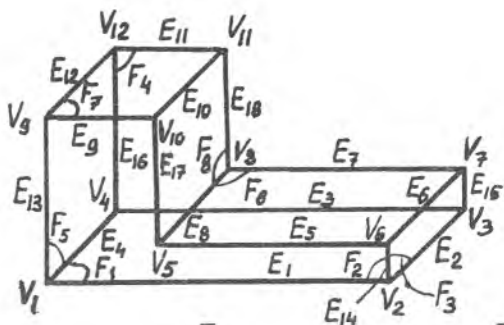
Р и с. 3.5. Вариант реализации симметричной структуры данных

ного ребра в списке V_i , два поля - указателями на две грани в списке F_i , и одно поле содержит указатель к записи с геометрической информацией по данному ребру. Эта информация может включать параметрические уравнения прямолинейного или криволинейного ребра. Записи для граней F_i и групп граней FG_i аналогичны записям для V_i и VG_i .

На рис. 3.6 дан пример содержания структуры данных конкретного объекта.

3.3. Поисковые процедуры в структуре данных

Рассмотрим сначала пример выделения трехгранной пирамиды в объекте, показанном на рис. 3.6. Так пирамида (V_6, V_2, V_7, V_5) может быть выделена соединением выпуклой трехгранной вершины V_6 с соседними вершинами V_2, V_5 и V_7 . Эта задача может быть решена следующим образом:



V_i

1	1	
2	4	
3	7	
4	10	
5	13	
6	16	
...	...	
12	34	

E_i

1	1	2		1	2
2	2	3		1	3
3	3	4		1	4
4	4	5		1	5
5	5	6		2	6
6	6	7		3	6
...
18	8	11		4	8

F_i

1	1	
2	5	
3	11	
4	15	
5	21	
6	25	
...
8	33	

VG_i

1	2	+	1
2	3	+	4
3	/	+	13
4	5	-	1
5	6	+	2
6	/	+	6
7	8	-	2
8	9	+	3
...
36	/	-	16

FG_i

1	1		2
2	2		3
3	3		4
4	4		/
5	1		6
6	4		7
7	5		8
8	17		9
...
15	3		16

Р и с. 3.6. Пример описания трехмерного объекта

Процедура ПОСТРОЕНИЯ ПИРАМИДЫ (V_1)

- Для вершины V_6 найти инцидентные ей ребра $\{E_i\}$.
- Для каждого ребра из $\{E_i\}$
 - найти две концевые вершины
 - исключить V_1 из пары
 - сформировать список вершин $\{V_j\}$

Для каждой вершины из $\{V_i\}$
соединить их попарно

Конец процедуры

В процессе построения пирамды можно выделить элементарные процедуры, часто встречающиеся при проектировании трехмерных объектов: "найти все ребра, инцидентные заданной вершине" и "найти концевые вершины данного ребра". Эти процедуры соответствуют дугам $V_i \rightarrow VG_i$ и $E_i \rightarrow V_i$ в симметричной структуре данных (см. рис. 3.4,б).

Обращаясь к рис. 3.4,б можно выделить девять основных процедур:

- P_1 : задана вершина, найти соседние вершины;
- P_2 : "-" найти инцидентные ребра;
- P_3 : "-" найти инцидентные грани;
- P_4 : задано ребро, найти концевые вершины;
- P_5 : "-" найти инцидентные ребра;
- P_6 : "-" найти две инцидентные грани;
- P_7 : задана грань, найти все вершины на ней;
- P_8 : "-" найти все ребра на ней;
- P_9 : "-" найти все соседние грани.

Процедуры P_2, P_4, P_6, P_9 выполняются путем простого обращения к структуре данных:

- P_2 : $V_i \rightarrow VG_i$
- P_4 : $E_i \rightarrow V_i$
- P_6 : $E_i \rightarrow F_i$
- P_8 : $E_i \rightarrow FG_i$

Другие процедуры могут быть выполнены за несколько обращений к структуре данных. Например, процедуре P_1 соответствует переход $V_i \rightarrow V_i$, который может быть сделан с помощью переходов (и соответствующих обращений к структуре данных) $V_i \rightarrow VG_i, VG_i \rightarrow E_i$ и $E_i \rightarrow V_i$. Аналогично, для P_3 необходимо выполнить $V_i \rightarrow VG_i \rightarrow E_i, E_i \rightarrow V_i$. Более подробно:

Процедура $P_1(V_i)$

$\{VG_i\} \leftarrow$ вызов $P_2(V_i)$.

Для каждого ребра E_i из $\{VG_i\}$ выполнить
 $\{V_j\}$ — вызов $P_4(E_i)$

Конец цикла

Возврат $\{V_j\}$

Конец процедуры P_1

Процедура $P_3(V_i)$

$\{VG_i\}$ — вызов $P_2(V_i)$

Для каждого ребра E_i из $\{VG_i\}$ выполнить

$\{F_j\}$ — вызов $P_6(E_i)$

Конец цикла

Возврат $\{F_j\}$

Конец процедуры P_3

Процедура $P_5(E_i)$

$\{V_i\}$ — вызов $P_4(E_i)$

Для каждой V_i из $\{V_i\}$ выполнить

$\{E_j\}$ — вызов $P_2(V_i)$

Конец цикла

Возврат $\{E_j\}$

Конец процедуры P_5

Процедура $P_7(F_i)$

$\{FG_i\}$ — вызов $P_8(F_i)$.

Для каждого E_i из $\{FG_i\}$ выполнить

$\{V_i\}$ — вызов $P_4(E_i)$

Конец цикла

Возврат $\{F_i\}$

Конец процедуры P_7

Процедура $P_9(F_i)$

$\{FG_i\}$ — вызов $P_8(F_i)$.

Для каждого E_i из $\{FG_i\}$ выполнить

$\{F_i\}$ — вызов $P_6(E_i)$

Конец цикла

Возврат $\{F_i\}$

Конец процедуры P_9

3.4. Операторы Эйлера

До сих пор мы анализировали объекты уже созданные, увидев объекты и не затрагивали вопросов их создания и редактирования. В данном разделе эта задача решается с использованием специальных операторов, позволяющих создать теоретико-графовые модели объектов.

Объектами являются тела, образованные многогранными поверхностями.

В классе моделируемых объектов включаются также незамкнутые многогранные поверхности, для чего вводится понятие о б о б щ е н н о с и т р а н и - грани с незамкнутым периметром. Будем считать, что все ребра ломаной принадлежат обобщенной грани.

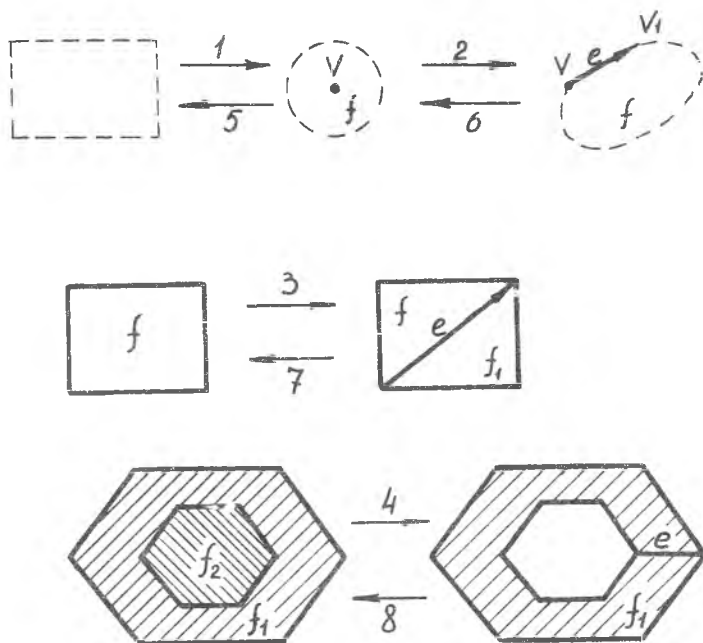
Конечной целью является создание графовой модели трехмерного объекта: объект задается набором граней, каждая из которых в свою очередь представляется набором ограничивающих ребер и вершин. Обычно используется пошаговый метод построения графовой модели с помощью набора операторов. На первом шаге формируется точечный объект. Далее с помощью операторов моделирования добавляются или удаляются элементы таким образом, что за конечное число шагов строится графовая модель исходного объекта. При этом на каждом шаге создается корректная с точки зрения выбранного множества графовых моделей структура, и процесс создания графовой модели значительно упрощается.

Для построения моделей объектов выделенного класса предлагается множество из восьми операторов моделирования, называемых о п е р а т о р а м и Э й л е р а /I/, /II/, /III/:

- 1 - создать точечный объект;
- 2 - добавить вершину и ребро;
- 3 - добавить ребро и грань;
- 4 - удалить грань, добавить ребро, топологический род поверхности увеличить на "1";
- 5 - удалить точечный объект;
- 6 - удалить вершину и ребро;
- 7 - удалить ребро и грань;
- 8 - добавить грань, удалить ребро, топологический род поверхности уменьшить на "1".

При этом элементы топологической структуры объекта добавляются и удаляются не произвольным образом, а по определенным правилам, исходя из условия сохранения топологической целостности графовой модели.

Результаты действия операторов моделирования на плоских моделях показан на рис. 3.7.



Р и с. 3.7. Результаты действия операторов Эйлера

4. АНАЛИТИЧЕСКИЕ ГЕОМЕТРИЧЕСКИЕ МОДЕЛИ

4.1. Определения и допущения

Аналитические модели описывают форму объектов с помощью набора поверхностей. В математике под поверхностью понимается непрерывное множество точек, координаты которых связаны некоторыми зависимостями в виде уравнений $F(x, y, z) = 0$. Функция $F(x, y, z)$ является либо многочленом n -й степени, либо какой-либо трансцендентной функцией. В первом случае поверхности называются алгебраическими, а во втором — трансцендентными.

Описание формы через поверхность широко используется в начертательной геометрии. Поскольку в начертательной геометрии геометрические фигуры задаются графически, то поверхность чаще всего рассматривается как совокупность последовательных положений некоторой перемещающейся в пространстве линии. В этом случае поверхность представляет собой непрерывное двухпараметрическое множество точек, а линия является непрерывным однопараметрическим множеством точек.

Многообразие форм поверхностей создает большие трудности при их изучении. Более того, поскольку число видов поверхностей и способов их получения неограничено, то создать строгую систему классификации поверхностей не представляется возможным, так как с геометрической точки зрения классификация поверхностей не имеет научного обоснования. Тем не менее, с целью систематизации материала можно использовать некоторые признаки, позволяющие выделить характерные классы, подклассы, группы, подгруппы поверхностей. Так, например, все поверхности делятся на два класса — нелинейчатые и линейчатые — по виду линии, образующей поверхность. В зависимости от способа образования эти поверхности делятся на три подкласса:

поверхности вращения, образованные вращением линии;

винтовые поверхности, образованные винтовым перемещением линии;

поверхности параллельного переноса, которые образуются поступательным перемещением линии.

Для более детального деления поверхностей используют и другие признаки. Например, линейчатые поверхности могут быть разделены по числу направляющих линий. Подробное рассмотрение типологии поверхностей выходит за рамки данного пособия. Эти данные можно получить, например, из /15/, /26/. Существует также обширная литература по методам проектирования поверхностей и кривых. Хорошим руководством по всем вопросам этого направления является книга /24/, а также /25, гл. 13/. О проектировании кривых методом Безье—Бернштейна можно узнать из /18/. Использованию сплайнов для аппроксимации зависимостей и в автоматизированном проектировании кривых посвящены работы /4/, /8/-/10/, /14/.

4.2. *B*-сплайны

Любой сплайн формируется как линейная комбинация некоторых полиномиальных базисных функций. В случае *B*-сплайна базисными

функциями являются четыре кубических полинома. В матричных обозначениях каждый кусок (часть) сплайна $C_i(t)$ можно представить как

$$C_i(t) = [t^3 t^2 t \ 1][C][V]_i^T, \quad t \in [0, 1], \quad i=1, \dots, n-1,$$

где

$$[C] = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & 6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix},$$

и

$$[V]_i = [V_{i-1} \ V_i \ V_{i+1} \ V_{i+2}], \quad i=1, \dots, n-1. \quad (4.1)$$

Здесь кривая $C(t)$ составляется из кусков $C_i(t)$, соединяющих точки интерполяции P_1, \dots, P_n . Точки V_i являются управляющими (рис. 4.1). В пределах одного куска $C_i(t)$ кривой параметр

t изменяется от 0 до 1 при движении от точки P_i к точке P_{i+1} , поэтому $C_i(0) = P_i$, $C_i(1) = P_{i+1}$, $i=1, \dots, n-1$, откуда

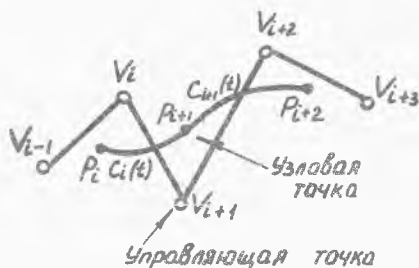
$$P_i = \frac{1}{6} (V_{i-1} + 4V_i + V_{i+1}). \quad (4.2)$$

Нетрудно перейти от кусочно-криволинейного описания ко всей кривой. Пусть

необходимо вычислить значение сплайна в точке t_0 всей кривой $C(t_0)$ причем параметр t всей кривой $C(t)$ также изменяется от 0 до 1. Для этого необходимо вычислить i и t_0^i :

$$\begin{cases} i = t_0(n-1) + 1, \\ t_0^i = t_0(n-1) - i + 1, \end{cases} \quad (4.3)$$

тогда



Р и с. 4.1. В -сплайн

$$C(t_0) = C_i(t_0^i), \quad 1 \leq i \leq n-1, \quad t_0^i \in [0, 1].$$

Для одномерного случая величины $C(t)$, $C_i(t)$, V_i и P_i в уравнениях (4.1) и (4.2) являются скалярными. При большей размерности каждый скаляр становится вектором-строкой: в двумерном случае эти столбцы содержат x и y координаты, в трехмерном случае — координаты x, y и z и т.д. Соседние сегменты $C_i(t)$ кривой имеют одинаковые, кроме одного, коэффициенты $[V]_i$, поэтому для интерполирующих точек P_i и $n-1$ куска $C_i(t)$ кривой необходимо $n+2$ коэффициента $V_i, i=0, \dots, n+1$. Однако, имеется только n уравнений типа (4.2). Эта проблема решается по-разному, в зависимости от того, замкнута кривая или разомкнута. Если кривая разомкнута, то обычно вводят для нее граничные условия, например: кривая в конечных точках имеет нулевую кривизну. Тогда:

$$\begin{cases} V_0 = 2V_1 - V_2, \\ V_{n+1} = 2V_n - V_{n-1}. \end{cases} \quad (4.4)$$

Для замкнутой кривой, проходящей через n точек, $P_{n+1} = P_1$. Следовательно, можно записать два дополнительных уравнения

$$V_{n+1} = V_1, \quad V_n = V_0. \quad (4.5)$$

Связь точек $\bar{P} = [P_1, \dots, P_n]$ и $\bar{V} = [V_1, \dots, V_n]$, следовательно, может быть представлена для разомкнутых кривых как

$$\bar{P} = [D]\bar{V},$$

где

$$[D] = \frac{1}{6} \begin{bmatrix} 6 & & & & & \\ & 4 & & & & \\ & & \ddots & & & \\ & & & i & & \\ & & & & 4 & \\ & & & & & 6 \end{bmatrix}$$

и для замкнутых кривых

$$\bar{P} = [D_c]\bar{V},$$

где

$$[D_c] = \frac{1}{6} \begin{bmatrix} 1 & 4 & & & & 1 \\ 1 & 4 & & & & \\ & & \ddots & & & \\ & & & 1 & & \\ & & & & 4 & 1 \\ 1 & & & & 1 & 4 \end{bmatrix}. \quad (4.6)$$

Вычисление значений B -сплайнового куска в m равноудаленных точках $t_j = j\sigma, j = 1, \dots, m, \sigma = 1/m$ связано с довольно большими усилиями по вычислению $[T] = [t^3 t^2 t 1]$. Вместо этого используют следующий метод:

$$\begin{aligned}
 [T_j] &= [(j\sigma)^2 (j\sigma)^2 j\sigma 1] = \\
 &= [0 0 0 1] \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} -1 & 2 & & \\ 1 & -1 & & \\ & 0 & 0 & 1 \end{bmatrix}^j \begin{bmatrix} \sigma^3 & & & \\ & \sigma^2 & & \\ & & \sigma & \\ & & & 1 \end{bmatrix} = \quad (4.7) \\
 &= [0 0 0 1] [L]^j [N] [\Delta].
 \end{aligned}$$

При вычислении значений B -сплайна на разных кусках в точках с одинаковыми значениями m и σ произведение $[N][\Delta][C]$ в уравнении (4.1) вычисляется один раз для всех $m(m-1)$ уравнений. Оставшиеся вычисления проводятся очень быстро.

Касательная к параметрической кривой совпадает с вектором ее первой производной. Кривизна кривой определяется ее второй производной. Таким образом для i -го сегмента кривой $\vec{c}(t) = [x(t) y(t)]$ касательная и кривизна равны:

$$\frac{\partial \vec{c}_i(t)}{\partial t} = \left[\frac{\partial x_i(t)}{\partial t}, \frac{\partial y_i(t)}{\partial t} \right] = [3t^2 2t 1 0][C][V]_i^T \quad (4.8)$$

и

$$\frac{\partial^2 \vec{c}_i(t)}{\partial t^2} = \left[\frac{\partial^2 x_i(t)}{\partial t^2}, \frac{\partial^2 y_i(t)}{\partial t^2} \right] = [6t 2 0 0][C][V]_i^T \quad (4.9)$$

Площадь, заключенная между началом координат и параметрической кривой $C(t) = [x(t), y(t)]$ на параметрическом интервале $[t_0, t_1]$, равна

$$A(t_0, t_1) = \int_{t_0}^{t_1} \frac{\partial A(t)}{\partial t} dt = \int_{t_0}^{t_1} (x(t) \frac{\partial y(t)}{\partial t} - y(t) \frac{\partial x(t)}{\partial t}) dt. \quad (4.10)$$

В случае B -сплайна интегрирование ведется по частям, соответствующим кускам кривой, а параметр изменяется в пределах интервала $[0, 1]$. Если $[V_x]$ - вектор коэффициентов для $x(t)$ и $[V_y]$ - вектор коэффициентов для $y(t)$, то

$$\begin{aligned}
 x(t) \frac{\partial y(t)}{\partial t} &= [t^3 t^2 t 1][C][V_x]^T [3t^3 2t 10][C][V_y]^T = \\
 &= [V_x][C]^T [t^3 t^2 t 1]^T [3t^3 2t 10][C][V_y]^T = \quad (4.11) \\
 &= [V_x][C]^T [M(t)][C][V_y]^T.
 \end{aligned}$$

Так, каждое выражение (4.11) является линейной комбинацией элементов $[N(t)]$, можно выполнить интегрирование только этих членов:

$$[N(L)] = \int_0^L [M(t)] dt = \int_0^L \begin{bmatrix} 3t^5 & 2t^4 & t^3 & 0 \\ 3t^4 & 2t^3 & t^2 & 0 \\ 3t^3 & 2t^2 & t & 0 \\ 3t^2 & 2t & 1 & 0 \end{bmatrix} dt = \quad (4.12)$$

$$= \begin{bmatrix} \frac{1}{2}L^6 & \frac{2}{5}L^5 & \frac{1}{4}L^4 & 0 \\ \frac{3}{5}L^5 & \frac{1}{2}L^4 & \frac{1}{3}L^3 & 0 \\ \frac{3}{4}L^4 & \frac{2}{3}L^3 & \frac{1}{2}L^2 & 0 \\ L^3 & L^2 & L & 0 \end{bmatrix}$$

Когда $L = 1$, имеем

$$Q = [N(1)] = \int_0^1 [M(t)] dt = \begin{bmatrix} \frac{1}{2} & \frac{2}{5} & \frac{1}{4} & 0 \\ \frac{3}{5} & \frac{1}{2} & \frac{1}{3} & 0 \\ \frac{3}{4} & \frac{2}{3} & \frac{1}{2} & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad (4.13)$$

Возвращаясь к уравнению (4.10) для одного сегмента B -сплайновой кривой, получаем

$$A = [V_x][C]^T[Q][C][V_y]^T - [V_y][C]^T[Q][C][V_x]^T = \quad (4.14)$$

$$= [V_x][C]^T([Q] - [Q]^T)[C][V_y]^T = [V_x][R][V_y]^T,$$

где

$$[R] = \begin{bmatrix} 0 & 31/10 & 14/5 & 1/10 \\ -31/10 & 0 & 183/10 & 14/5 \\ -14/5 & -183/10 & 0 & 31/10 \\ -1/10 & -14/5 & -31/10 & 0 \end{bmatrix}.$$

Если вместо $[Q]$ в уравнении использовать $[N(L)]$, то можно вычислить только ту компоненту, которая зависит от L :

$$[N(L)] - [N(L)]^T = \begin{bmatrix} 0 & \frac{1}{5}L^5 & -\frac{1}{2}L^4 & -L^3 \\ \frac{1}{5}L^5 & 0 & -\frac{1}{3}L^3 & -L^2 \\ \frac{1}{2}L^4 & \frac{1}{3}L^3 & 0 & -L \\ L^3 & L^2 & L & 0 \end{bmatrix} \quad (4.15)$$

Другая интересная формула определяет центр тяжести замкнутой планарной кривой или центроид $[x_c, y_c]$. Если A - площадь области, ограниченной кривой, то x_c определяется как

$$\bar{x}_c = \frac{1}{3A} \sum_i \int_0^1 x_i(t) \frac{\partial A_i(t)}{\partial t} dt = \sum_i \int_0^1 (x_i(t))^2 \frac{\partial y_i(t)}{\partial t} - x_i(t) y_i(t) \frac{\partial x_i(t)}{\partial t} dt. \quad (4.16)$$

Аналогичную формулу можно записать и для y_c .

Точное вычисление длины B -сплайнового куска довольно сложно, поэтому обычно прибегают к аппроксимации. Например, длина $l(t_0)$ двумерной кривой $C(t) = [x(t) y(t)]$ от $C(0)$ до $C(t_0)$ равна:

$$l(t_0) = \int_0^{t_0} \sqrt{\left(\frac{\partial x(t)}{\partial t}\right)^2 + \left(\frac{\partial y(t)}{\partial t}\right)^2} dt. \quad (4.17)$$

Для аппроксимации могут использоваться различные методы, например метод трапеций. Если B -сплайн вычисляется в $m-1$ равноотстоящих точках, то аппроксимация длины

$$l(t_0) \sim \sum_{j=1}^{m-1} \sqrt{[x(t_{j+1}) - x(t_j)]^2 + [y(t_{j+1}) - y(t_j)]^2}. \quad (4.18)$$

Для вычисления длины по формуле (4.18) полезно использовать соотношение (4.1), а для вычислений по уравнению (4.17) следующее разностное уравнение, соответствующее непрерывному соотношению (4.8):

$$[3t_j^2 \ 2t_j \ 1] = [0 \ 0 \ 1] \begin{vmatrix} 1 & 2 & j & 3\sigma^2 \\ 1 & 1 & -1 & 2\sigma \\ 1 & 1 & 0 & 1 \end{vmatrix} = [0 \ 0 \ 1] [L]^j [N] [\Delta], \text{ для } \sigma = \frac{t_0}{m}. \quad (4.19)$$

Можно сразу вычислить произведение $[N][\sigma][\dot{C}]$, где $[\dot{C}]$ получается из $[C]$ в уравнении (4.8) путем вычеркивания последнего столбца и последней строки.

Радиус кривизны $\rho(s)$ трехмерной B -сплайновой кривой в векторном параметрическом обозначении $A(s) = [x(s), y(s), z(s)]$ равен

$$\rho = \rho(s) = \frac{(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)^{3/2}}{[(\dot{x}\ddot{y} - \dot{y}\ddot{x})^2 + (\dot{y}\ddot{z} - \dot{z}\ddot{y})^2 + (\dot{z}\ddot{x} - \dot{x}\ddot{z})^2]^{1/2}}. \quad (4.20)$$

где $\dot{\alpha} = \frac{\partial}{\partial S} \alpha(S)$ и $\ddot{\alpha} = \frac{\partial^2}{\partial S^2} \alpha(S)$, для $\alpha = x, y, z$.

В точках, где $S = 0$, вычисление уравнения (4.20) становится довольно простым, так как

$$\begin{aligned} \dot{\alpha} &= [-3 \ 0 \ 3 \ 0] [V_{\alpha}] \\ \ddot{\alpha} &= [3 \ -6 \ 3 \ 0] [V_{\alpha}], \end{aligned} \quad (4.21)$$

где V_{α} - четырехэлементный вектор - столбец коэффициентов.

Таким образом, если определить $[\dot{C}] = [-3 \ 0 \ 3 \ 0]$ и $[\ddot{C}] = [3 \ -6 \ 3 \ 0]$, то для $\alpha, \beta = x, y, z$, имеем:

$$\begin{aligned} \dot{\alpha}\dot{\beta} &= [C][V_{\alpha}][\dot{C}][V_{\beta}] = [V_{\alpha}]^T[\dot{C}]^T[\dot{C}][V_{\beta}] = \\ &= [V_{\alpha}]^T[N][V_{\beta}], \quad \ddot{\alpha} = [\ddot{C}][V_{\alpha}][\dot{C}][V_{\alpha}] = \\ &= [V_{\alpha}]^T[N][V_{\alpha}], \end{aligned} \quad (4.22)$$

где

$$[N] = [\dot{C}]^T[\ddot{C}] = \begin{bmatrix} -9 & 18 & -9 & 0 \\ 0 & 0 & 0 & 0 \\ 9 & -18 & 9 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[N] = [\dot{C}]^T[\dot{C}] = \begin{bmatrix} -9 & 0 & 9 & 0 \\ 0 & 0 & 0 & 0 \\ 9 & 0 & -9 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.23)$$

4.3. Параметрические бикубические поверхности

Поверхность описывается как непрерывная функция двух параметров u, v :

$$\begin{aligned} x &= F_1(u, v), \\ y &= F_2(u, v), \\ z &= F_3(u, v). \end{aligned} \quad (4.24)$$

Чаще всего в практике автоматизированного проектирования в качестве таких функций используются полиномы третьей степени от u, v .

v . Тогда матричные формы записи для куска бикубической поверхности /2/:

$$\bar{W}(\mu, \nu) = (\mu^3 \mu^2 \mu 1) \bar{M} \bar{B} \bar{M}^T \begin{bmatrix} \nu^3 \\ \nu^2 \\ \nu \\ 1 \end{bmatrix},$$

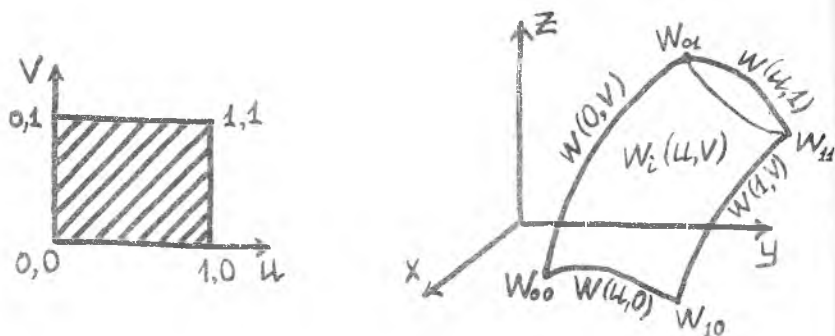
где $\bar{W}(\mu, \nu) = (x(\mu, \nu), y(\mu, \nu), z(\mu, \nu))$,

$$\bar{M} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \quad (4.25)$$

и \bar{B} — так называемая геометрическая матрица, поскольку ее элементы управляют формой куска поверхности. Обозначим элементы матрицы B следующим образом:

$$\bar{B} = \begin{bmatrix} W_{00} & W_{01} & W_{00V} & W_{01V} \\ W_{10} & W_{11} & W_{10V} & W_{11V} \\ W_{00\mu} & W_{01\mu} & W_{00\mu V} & W_{01\mu V} \\ W_{10\mu} & W_{11\mu} & W_{10\mu V} & W_{11\mu V} \end{bmatrix} \quad (4.26)$$

На рис. 4.2 показана связь между трехмерным пространством и плоскостью параметров μ, ν



Р и с. 4.2. Связь трехмерного и параметрического пространства

Элементы матрицы B определяют следующие свойства куска поверхности:

положение угловых точек куска

$$W_{ij} = [W(\mu, \nu)] \text{ в точке } \mu = i, \nu = j;$$

наклон в углах $u = i, v = j$ куска

$$W_{ij u} = \left[\frac{\partial W(u, v)}{\partial u} \right];$$

изгиб в углах $u = i, v = j$ куска

$$W_{ij uv} = \left[\frac{\partial^2 W(u, v)}{\partial u \partial v} \right].$$

Существуют различные способы создания параметрических бикубических поверхностей, однако почти все они требуют задания 48 коэффициентов (по 16 коэффициентов для x, y, z). Инженер легко может задать положение углов куска поверхности, однако определение и работа с производными, зависящими от параметров u, v , является для него намного более трудным делом. Изменение одного или нескольких параметрических наклонов или изгибов может давать непредсказуемые изменения формы поверхности.

Один из возможных путей решения этой проблемы состоит в определении поверхности только через положение точек. Рассмотрим, например, метод описания куска поверхности с помощью задания 16-ти точек.

Используя более компактную форму уравнения (4.25), получим

$$\bar{W}(u, v) = (u^3 u^2 u 1) \bar{S} \begin{bmatrix} v^3 \\ v^2 \\ v \\ 1 \end{bmatrix}, \quad 2\partial\partial\bar{S} = \bar{M} \bar{B} \bar{M}^T. \quad (4.27)$$

В обычной форме это уравнение имеет вид

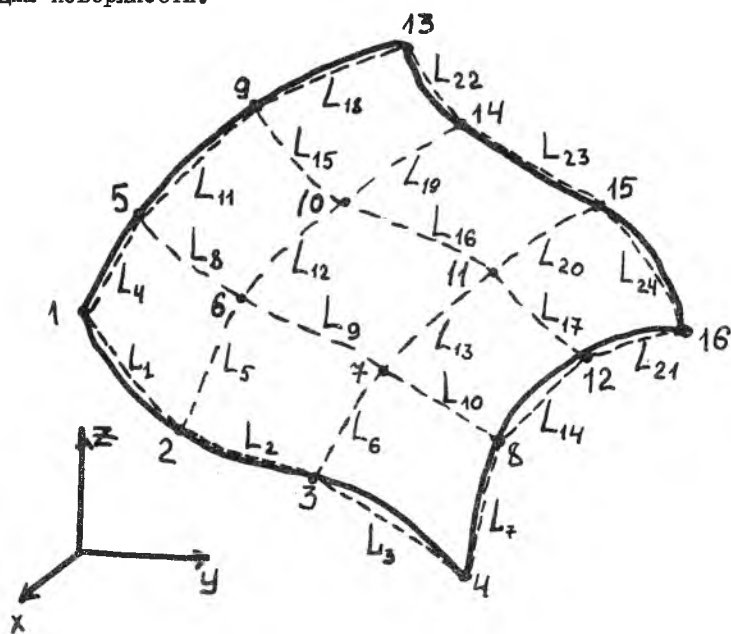
$$\begin{aligned} & (u^3 v^3) S_{11} + (u^3 v^2) S_{12} + (u^3 v) S_{13} + (u^3) S_{14} + \\ & + (u^2 v^3) S_{21} + (u^2 v^2) S_{22} + (u^2 v) S_{23} + (u^2) S_{24} + \\ & + (u v^3) S_{31} + (u v^2) S_{32} + (u v) S_{33} + (u) S_{34} + \\ & + (v^3) S_{41} + (v^2) S_{42} + (v) S_{43} + S_{44} = W(u, v) \end{aligned}$$

или

$$\bar{R} \cdot \bar{F} = \bar{G}.$$

(4.28)

Здесь \bar{R} является матрицей 16×16 , определяющей произведения степеней параметров u и v , \bar{F} является вектором 16×1 неизвестных элементов \bar{S} и \bar{G} - вектор 16×1 заданных 16 -ти точек. Следовательно, результат может быть получен решением системы линейных уравнений. Удобно использовать значения u, v , равные $0; 0,25; 0,5; 0,75$. Рассмотрим, например, кусок поверхности с 16 -ю точками, расположенными на ней (рис. 4.3). Точки сначала соединяются отрезками (показаны пунктиром), затем вычисляется длина этих отрезков и по табл. 4.1 определяются значения u, v . От значений, вычисленных с помощью табл. 4.1, можно вернуться к уравнению (4.28), чтобы определить элементы матрицы \bar{R} . После того, как 48 коэффициентов матрицы \bar{S} вычислены (матрице \bar{R} соответствуют три матрицы \bar{S} для x, y и z), можно получить координаты x, y и z для произвольной точки, лежащей на куске поверхности. Если зафиксировать один из параметров, например v , то изменяя другой, получим кривую, которую легко отобразить на экране графического дисплея при визуализации поверхности.



Р и с. 4.3. Бикубические поверхности

Таблица 4.1

Номер точки	Значения параметров	
	u	v
I	0	0
2	$L_1 / (L_1 + L_2 + L_3)$	0
3	$(L_1 + L_2) / (L_1 + L_2 + L_3)$	0
4	1	0
5	0	$L_4 / (L_4 + L_{11} + L_{18})$
6	$L_8 / (L_8 + L_9 + L_{10})$	$L_5 / (L_5 + L_{12} + L_{19})$
7	$(L_8 + L_9) / (L_8 + L_9 + L_{10})$	$L_6 / (L_6 + L_{13} + L_{20})$
8	1	$L_7 / (L_7 + L_{14} + L_{21})$
9	0	$(L_4 + L_{11}) / (L_4 + L_{11} + L_{18})$
10	$L_{15} / (L_{15} + L_{16} + L_{17})$	$(L_{12} + L_{15}) / (L_5 + L_{12} + L_{19})$
11	$(L_{15} + L_{16}) / (L_{15} + L_{16} + L_{17})$	$(L_{13} + L_6) / (L_6 + L_{13} + L_{20})$
12	1	$(L_{14} + L_7) / (L_7 + L_{14} + L_{21})$
13	0.0	1.0
14	$L_{22} / (L_{22} + L_{23} + L_{24})$	1.0
15	$(L_{22} + L_{23}) / (L_{22} + L_{23} + L_{24})$	1.0
16	1.0	1.0

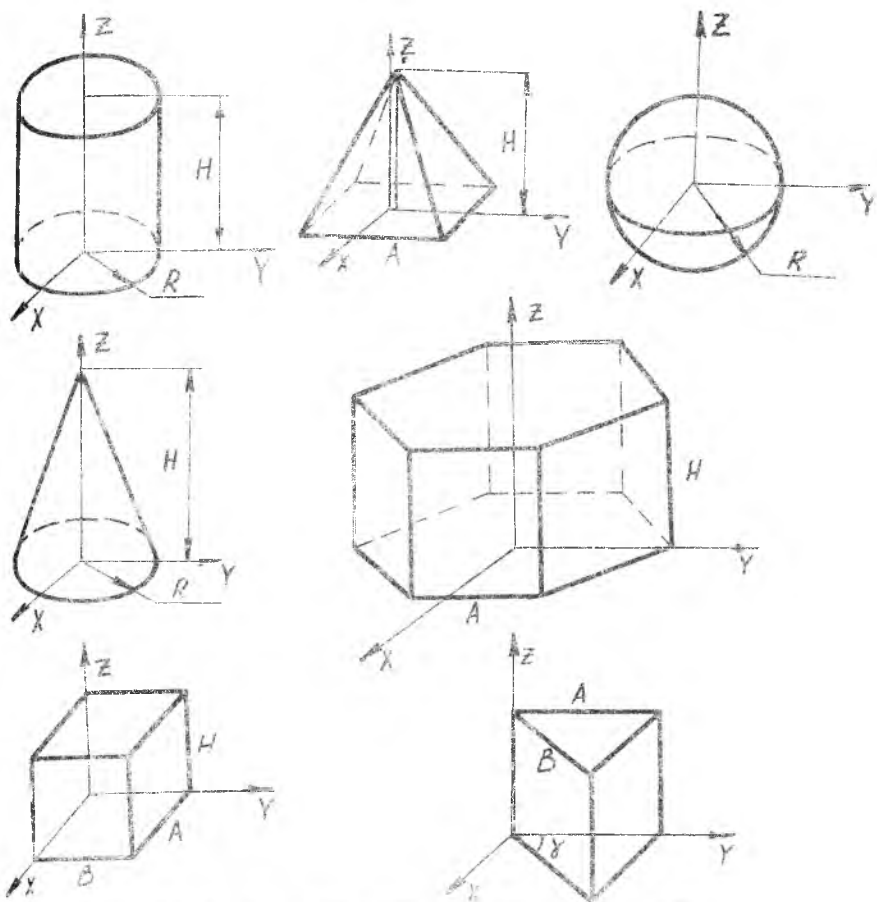
5.1. Параметрические модели

Во многих применениях имеются часто используемые модели (части), размеры которых задаются стандартными рядами (таблицами), а форма остается примерно одинаковой. Таковы, например, металлические уголковые конструкции, трубопроводы, шарикоподшипники, крепеж (винты, болты, гайки, шайбы) и т.д. Форма таких изделий может быть описана заранее с помощью перечисления геометрических отношений и условий для элементов фигуры. Каждая форма определяется несколькими параметрами, по которым с помощью стандартной процедуры вычисляются другие параметры, необходимые для описания формы.

Описание новой формы с помощью параметрической геометрической модели заключается в указании типа или номера модели и значений описывающих ее параметров. При этом происходит линейное преобразование хранимой модели, т.е. изменение размеров и пропорций при сохранении топологических свойств фигуры. Такой способ описания формы фигуры называется также описанием по образцу или описанием с использованием аналогов.

Стандартные формы называют также *непроизводными формами* или *элементарными геометрическими формами*. Обычно используется набор из нескольких стандартных форм (класс форм, "алфавит" форм). Рассмотрим алфавит непроизводных форм, включающих 3 вида призм, пирамиду, цилиндр, конус и шар (рис. 5.1).

Недостатком параметрической поверхности геометрической модели является ограниченное число классов форм, которые могут быть описаны таким образом. Стандартный прием расширения набора форм — указание рядов для значений параметров, описывающих форму. Другой метод состоит в указании некоторого геометрического преобразования для формы, значения параметров которой зафиксированы. Третий метод заключается в образовании составных форм с помощью теоретико-множественных операций. Наконец, возможны нелинейные преобразования формы типа скругления углов или скручивания всего объекта. Эти методы изменения формы объекта рассматриваются в последующих разделах.



Р и с. 5.1. "Алфавит" производных форм

5.2. Геометрические преобразования образов

Каждый трехмерный объект, входящий в число образов ("алфавит" форм), описывается в своей местной системе координат (СК). Когда пользователь создает новый примитив, значения геометрических параметров должны быть как можно проще, чтобы облегчить его использование. Так например, для всех параллелепипедов, независимо от их размера, ориентации и положения, образцом будет простейший, с точки зрения описания, параллелепипед - куб единичного размера, одна из вершин которого совпадает с началом местной СК, а стороны параллельны (или перпендикулярны) осям СК.

Использование примитива в составе реального трехмерного объекта (в частном случае — одного примитива) требует изменений размеров, положения и ориентации и описывается матрицей геометрического преобразования M . Любые изменения размеров, положения и ориентации сводится к суперпозиции преобразования масштабирования сдвига и поворота, каждое из которых может быть определено своей матрицей преобразования. Результирующему преобразованию соответствует матрица, равная произведению матриц всех составляющих преобразований. Тогда, точка с координатами (x, y, z) переходит в точку (x', y', z') , где

$$(x', y', z', 1) = (x, y, z, 1)M.$$

Аналогично, вектор $V = (D_x, D_y, D_z)$ переходит в вектор $V' = (D'_x, D'_y, D'_z)$:

$$(D'_x, D'_y, D'_z, 0) = (D_x, D_y, D_z, 0)M.$$

Для преобразования прямой линии (или луча) используется фиксированная точка (x_0, y_0, z_0) и направляющий вектор (D_x, D_y, D_z) . Тогда $(x_0, y_0, z_0)(D_x, D_y, D_z)M = [(x_0, y_0, z_0, 1)M]$.

$$[(D_x, D_y, D_z, 0)M] = (x'_0, y'_0, z'_0)(D'_x, D'_y, D'_z).$$

Подробнее о геометрических преобразованиях можно узнать из книг по машинной графике (см. /6/, /25/ и др.).

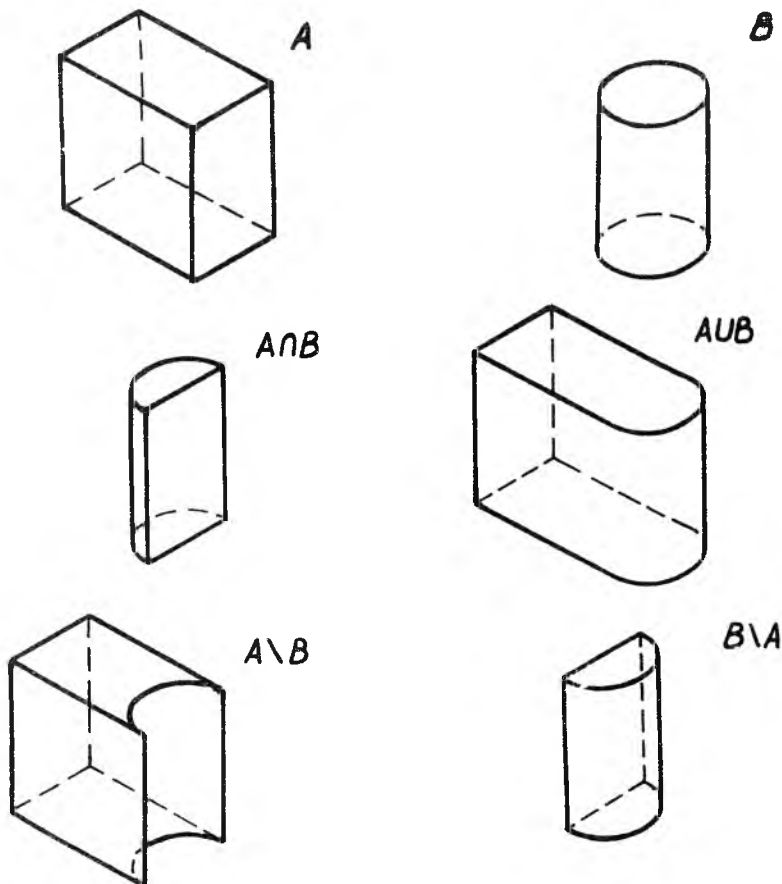
Следует отметить, что, хотя использование геометрических преобразований образцов не дает большого простора проектировщику, формы трансформированных элементов могут выглядеть совершенно различно. Например, плоская шайба и труба определяются одним и тем же цилиндром с отверстием.

5.3. Теоретико-множественные операции

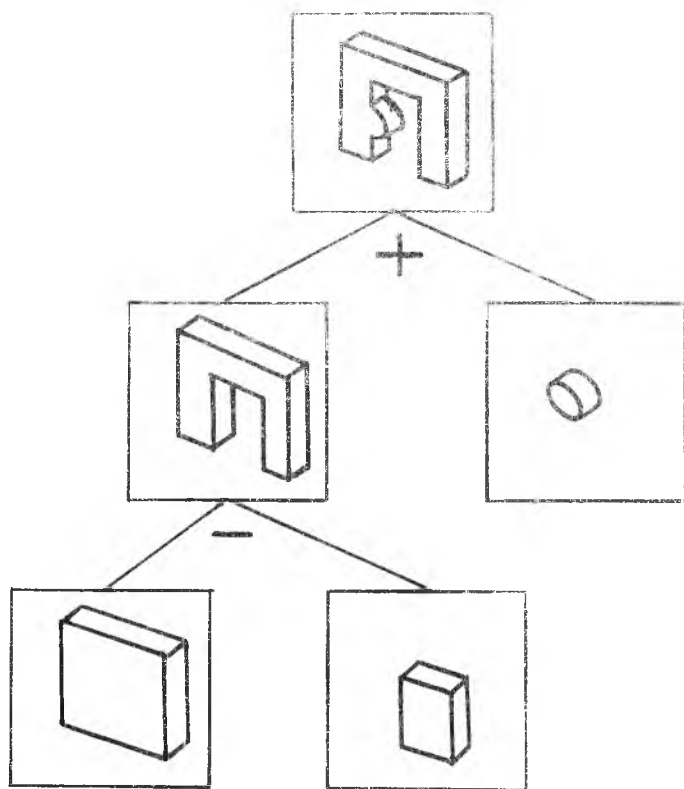
Теоретико-множественные операции (ТМО) являются мощным средством синтеза новых форм из ранее созданных примитивов и составных форм (см., например, /1/, /16/, /19/, /20/, /22/, /23/). На рис. 5.2 показаны результаты применения ТМО — объединения, пересечения и разности — к двум объектам — кубу и цилиндру. Операция объединения дает новый объект, точки которого ранее принадлежали к одному из участвующих в операции объектов. Операция пересечения дает объект, точки которого расположены в пространстве там, где одновременно присутствуют точки исходных объектов. Наконец, операция разности дает объект, составленных из

точек пространства, принадлежащих одному из исходных объектов и одновременно не принадлежащих второму.

Результат применения ТМО к двум объектам может стать операндом в новой операции. Обычно первые операции выполняются над простейшими объектами – примитивами, и путем повторного применения операций к результатам предыдущего этапа удается построить довольно сложные объекты. Последовательность применения ТМО часто хранится в памяти ЭВМ в виде двоичного дерева, в котором каждый родительский узел образован из двух дочерних с помощью одной из ТМО (рис. 5.3).



Р и с. 5.2. Теоретико-множественные операции с трехмерными объектами



Р и с. 5.3. Бинарное дерево описания трехмерного объекта с помощью теоретико-множественных операций

5.4. Преобразования формы объекта путем скругления его граней и вершин

Общие сведения. Использование геометрических преобразований примитивов и теоретико-множественных операций над ними дает возможность, как мы видели в предыдущих разделах, спроектировать довольно сложные объекты. Однако, поскольку примитивы обычно являются многогранниками или простейшими криволинейными объектами типа цилиндров или конусов, то синтез объектов сложной формы связан с аппроксимацией криволинейных поверхностей поверхностями первого и второго по-

рядка. Такую аппроксимацию очень сложно выполнить, используя только геометрические преобразования и теоретико-множественные операции над примитивами, так как человеку трудно представить, какие именно преобразования и операции следует выбрать. С другой стороны, если использовать примитивы со сложными криволинейными поверхностями, то становится трудно реализовать теоретико-множественные операции с ними.

Один из вариантов решения этой задачи связан с использованием процедур локального изменения формы путем добавления, исключения или передвижения локальных элементов объекта — граней, ребер и вершин. При этом сначала создается грубая модель многогранника путем его локальных модификаций, затем генерируется точная криволинейная модель скруглением углов и ребер. На заключительном этапе вычисляются уравнения поверхностей, интерполирующих сгенерированную модель.

Такое проектирование форм близко и понятно инженеру, оно легко реализуется и выполнение его на вычислительной машине не требует значительного времени. По этим причинам данный метод идеально подходит для интерактивного проектирования формы.

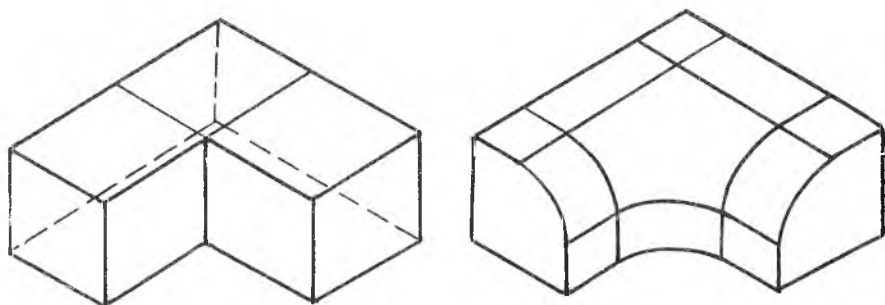
Процесс проектирования включает следующие три фазы:

генерация трехмерного объекта путем локальных изменений многогранника. Ребра синтезированного объекта являются отрезками прямых линий, однако грани не обязательно плоские. Этот объект является основой для последующих операций, определяющих топологическую информацию, т.е. данные о том, как соединяются между собой грани, ребра и вершины. Поскольку модель не содержит уравнений поверхности, то можно генерировать объекты с изогнутыми гранями;

генерация трехмерной криволинейной модели путем скручивания базового объекта. Чтобы выполнить скругление, пользователь должен сообщить, какие углы и ребра изменяются. Например, если он хочет, чтобы ребро было скруглено, то оно помечается "0", а если это ребро должно оставаться прямолинейным, то оно помечается "1". Разметка производится для всех ребер объекта.

В соответствии с этой информацией генерируются новые прямолинейные и криволинейные ребра взамен ребер и вершин исходного объекта. На рис. 5.4 показан пример изменений формы объекта с помощью

скруглений. Ребра криволинейной модели являются кривыми Безье и отрезками. Топологическая информация по криволинейной модели близка к топологии исходного объекта;

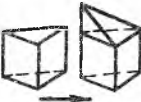

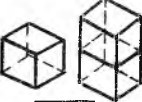


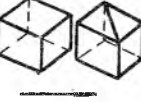
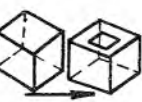
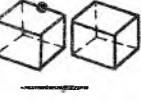
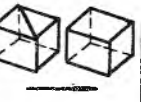
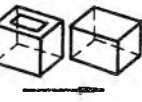
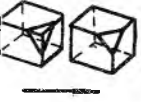


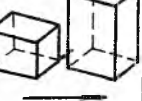


Р и с. 5.4. Пример скругления формы объекта

автоматическая генерация описания синтезированного объекта в форме набора кусков поверхности. Каждый кусок определяется только своими граничными кривыми. На этой фазе пользователь уже не может изменять форму объекта, однако он может проверить гладкость поверхности путем визуализации полученных данных. Визуализация может быть выполнена либо вычерчиванием поперечных сечений объекта ("проволочное" изображение), либо генерацией полутонного изображения. Если формат объекта в чем-то не удовлетворяет пользователя, тот может вернуться к этапам проектирования базисного объекта или создания криволинейной модели.

Локальные изменения многогранников. Локальные изменения многогранников можно представить в виде четырех процедур, характер воздействия которых на различные элементы объекта показан на рис. 5.5.

Следует учесть, что локальные изменения автоматически сохраняют топологическую согласованность частей объекта, однако геометрическая согласованность не всегда сохраняется. Поэтому если проектировщик ошибается при выборе локальных элементов или величины сдвигов в процедурах сдвига и лифта, то иногда может получиться модель объекта с самопересекающимися гранями. Простейший способ избежать таких ситуаций – визуальный контроль самим проектировщиком резуль-

	Вершина	Ребро	Грань	Множество граней
лифт				
Добавление				
Исключение				
Сдвиг				

Р и с. 5.5. Процедуры локального редактирования многогранников

татов всех локальных модификаций объекта. Однако более предпочтительным является автоматический контроль правильности синтезированного объекта (см. также п. 1.3).

Покажем сначала, как генерируются новые ребра в соответствии с ребрами базисного объекта. На рис. 5.6 показана часть базисного объекта, включающего ребро E_0 , разделяющего грани F_1 и F_2 и соединяющего вершины V_1 и V_2 . Ребра E_1, E_2, E_3, E_4 инцидентны ребрам E_0 . Точки P_0, G_1 и G_2 являются, соответственно, центрами ребра E_0 и граней F_1 и F_2 . Цент-

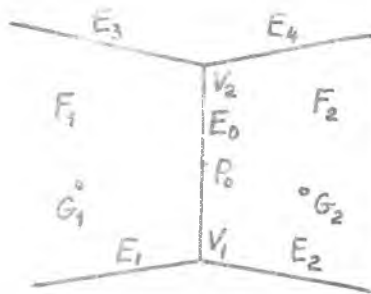
рами ребер и граней считаются точки, координатами которых равны средним значениям координат вершин, принадлежащих ребру или грани.

Если ребро E_0 помечено символом "0", то генерируется криволинейный сегмент (рис. 5.7,а) Для этого сегмента концевыми точками являются G_1 и G_2 , а касательные в точках G_1 и G_2 являются отрезками $G_1 P_0$ и $P_0 G_2$

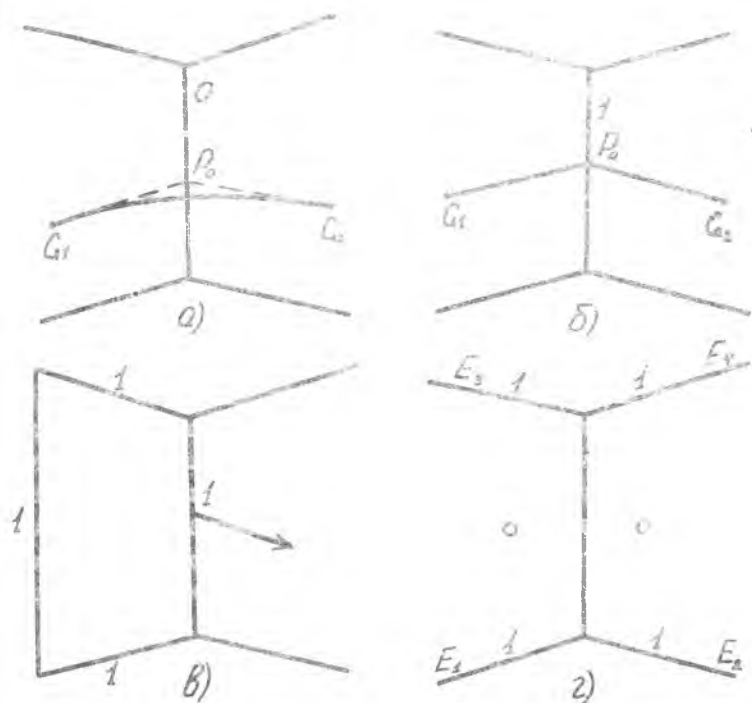
Если ребро помечено как "1", то генерируются два отрезка $G_1 P_0$ и $P_0 G_2$ (рис. 5.7,б). Такое правило иногда генерирует ненужные ребра, поэтому, когда все ребра грани базисного объекта помечены единицами, новое ребро не формируется (рис. 5.7,в). Аналогично, если ребра E_1, E_2, E_3 и E_4 помечены "1", то требование скругления ребра E_0 игнорируется, и новое ребро не формируется (рис. 5.7,г).

Следует отметить, что, когда, грань имеет границу в форме невыпуклого многоугольника, при формировании криволинейной модели иногда возникает самопересечение объекта, так как новые вершины вводятся вне грани. Для того, чтобы избежать таких случаев, проектировщик должен провести дополнительные ребра в грани, разбивая многоугольники на выпуклые части.

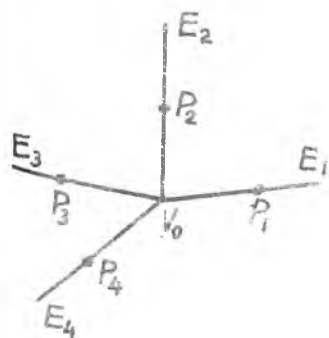
Второе правило скругления связано с вершинами исходного объекта. На рис. 5.8 показана вершина V_0 исходного объекта, из которой выходят ребра $E_i (i=1, \dots, n)$. Точки $P_i (i=1, \dots, n)$ являются центрами этих ребер. Способ скругления вершины V_0 выбирается в зависимости от вида разметки ребер E_i . Когда все ребра помечены "0" или одно ребро помечено "1", то никаких новых ребер не проводится (рис. 5.9,а,б). Если два ребра E_K и $E_M (1 < K, M < n)$ помечены "1", а остальные "0", то проводится криволинейное ребро (рис. 5.9,в), концевыми точками которого являются P_K и P_M , а отрезки $P_K V_0$ и $V_0 P_M$ являются касательными к кривой в точках P_K и P_M соответственно. Когда имеется более двух ребер, помеченных "1", то проводятся отрезки от центров этих ребер к вершине V_0 (рис. 5.9,г.).



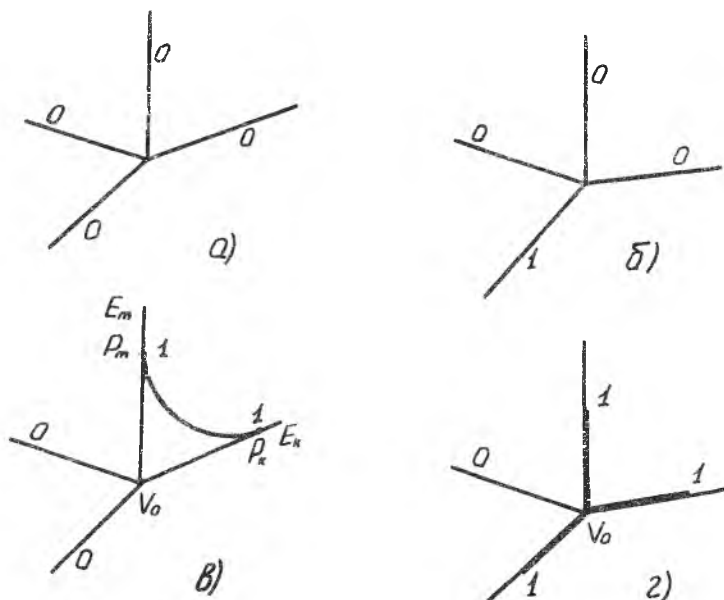
Р и с. 5.6. Ребро базового объекта



Р и с. 5.7. Новые ребра, генерируемые на ребре базового объекта



Р и с. 5.8. Вершина базового объекта



Р и с. 5.9. Новые ребра, генерируемые на вершинах базового объекта

При формировании кривой скругления всегда заданы три точки P_0, P_1, P_2 . Точки P_0 и P_1 являются концами кривой, а отрезки P_0P_1 и P_1P_2 являются касательными к кривой в точках P_0 и P_2 соответственно. Этим условиям удовлетворяет квадратичная кривая Безье с управляющими точками P_0, P_1 и P_2 .

5.5. Операция по деформации примитивов

Основные определения. Разработка глобальных и локальных операций по деформации трехмерных объектов является новым направлением в автоматизированном проектировании /3/. Деформации хорошо согласуются с интуитивными представлениями об изготовлении объектов путем, например, прессования или проката. Для описания деформаций существуют простые математические методы, широко используемые в моделировании.

Глобальная информация трехмерного объек-

та является математической функцией F , изменяющей глобальные (мировые) координаты точек. В дальнейшем будем обозначать координаты точек объекта до деформации через \bar{x} (строчная), а точки деформированного объекта — через X (прописная), т. е.

$$\bar{X} = F(\bar{x}) \quad (5.1)$$

Компоненты координат будем обозначать как $\bar{x} = (x, y, z)$ или как $\bar{x} = (x_1, x_2, x_3)$. Аналогично и для \bar{X} .

Локальная деформация изменяет часть объекта. Объект в этом случае описывается набором векторов, обычно касательных и нормалей. Касательные и нормали недеформированного пространства преобразуются в касательные и нормали деформированной поверхности. Математически это преобразование сводится к единственному умножению на якобиан от функции преобразования $\bar{X} = F(\bar{x})$. Якобиан является функцией от \bar{x} и выполняется с помощью частных производных относительно координат x_1, x_2 и x_3 :

$$J_i(\bar{x}) = \frac{\partial F(x)}{\partial x_i} \quad (5.2)$$

Когда поверхность объекта задана параметрически относительно переменных u и v , т. е. $\bar{x} = x(u, v)$, то любая касательная к поверхности может быть получена линейной комбинацией частных производных от x относительно u и v . Вектор нормали может быть получен как векторное произведение двух линейно независимых векторов, касательных к поверхности.

Преобразование касательных векторов в матричной форме записывается как

$$\frac{\partial \bar{X}}{\partial u} = J \frac{\partial \bar{x}}{\partial u} \quad (5.3)$$

Эта форма записи эквивалентна

$$X_{i,u} = \sum_{j=1}^3 J_{ij} x_j u \quad (5.4)$$

Преобразование нормалей выполняется с помощью обратной транспонированной матрицы якобиана

$$\bar{n}(\bar{X}) = (\det J)(J^{-1})^T n(x).$$

Поскольку для наших целей важно только направление нормали, то нет необходимости вычислять значение определителя $\det J$. Тем не менее, иногда важно сохранять соотношение объемов исходного и деформированного объектов, и в этом случае определитель должен быть вычислен.

Примеры деформаций

Масштабирование. Одной из наиболее простых деформаций является изменение длины объекта вдоль одной из координатных осей, что достигается ортогональной масштабирующей операцией

$$\begin{aligned} X &= a_1 x \\ Y &= a_2 y \\ Z &= a_3 z. \end{aligned} \tag{5.5}$$

Компоненты якобиана определяются как $J_{ij} = \frac{\partial X}{\partial x_j}$ поэтому

$$J = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{bmatrix} \tag{5.6}$$

Определитель этой матрицы, равный $a_1 a_2 a_3$ показывает степень изменения объема.

Матрица преобразования векторов-нормалей является обратной транспонированной матрицей и равна

$$(\det J)(J^{-1})^T = \begin{bmatrix} a_2 a_3 & 0 & 0 \\ 0 & a_1 a_3 & 0 \\ 0 & 0 & a_1 a_2 \end{bmatrix}$$

Без учета определителя матрица для преобразования нормалей равна

$$(J^{-1})^T = \begin{bmatrix} 1/a_1 & 0 & 0 \\ 0 & 1/a_2 & 0 \\ 0 & 0 & 1/a_3 \end{bmatrix}$$

Рассмотрим, например, точку (x_1, x_2, x_3) лежащую на примерно сферической поверхности с центром в начале координат и нормальным вектором $(n_1, n_2, n_3)^T$. Трансформированная поверхность имеет эллипсоидальную форму.

Точка будет иметь координаты $(a_1 x_1, a_2 x_2, a_3 x_3)^T$ и нормальный вектор параллелен вектору $(n_1/a_1, n_2/a_2, n_3/a_3)$.

Масштабирующее преобразование является частным случаем аффинного преобразования, в котором якобиан является постоянной матрицей. Аффинные преобразования включают сдвиг, поворот и масштабирование. Если преобразование состоит из чистого поворота, то обратная матрица равна ее транспонированной. Это означает, что касательные и нормали преобразуются одной и той же матрицей.

Общее сужение объекта вдоль оси z . Сужение объекта подобно масштабированию. Однако, здесь масштабный коэффициент не является постоянной величиной, а изменяется вдоль двух из трех координатных осей, т.е. является функцией $f(z)$. Когда $f(z)=1$, объект не деформируется. При $f'(z) > 0$ размеры объекта увеличиваются при увеличении z . Математически преобразование записывается в виде

$$z = f(z)$$

$$X = z x \tag{5.7}$$

$$y = z y$$

$$z = z$$

Матрица преобразования касательных равна:

$$J = \begin{bmatrix} z & 0 & f'(z)x \\ 0 & z & f'(z)y \\ 0 & 0 & 1 \end{bmatrix} \tag{5.8}$$

Локальное изменение объема определяется коэффициентом

Матрица преобразования равна:

$$z^2 (J^{-1})^T = \begin{bmatrix} z & 0 & 0 \\ 0 & z & 0 \\ -z f'(z)x & -z f'(z)y & z \end{bmatrix}$$

Обратное преобразование

$$z(z) = f(z),$$

$$x = X/z,$$

$$y = Y/z,$$

$$z = z. \tag{5.9}$$

Осевое скручивание. Это преобразование часто используется на практике. Скручивание можно аппроксимировать поворотом с переменным углом. Скручивание вокруг оси Z выполняется следующим преобразованием:

$$\begin{aligned} \theta &= f(z), \\ C\theta &= \cos(\theta), \\ S\theta &= \sin(\theta), \\ X &= xC\theta - yS\theta, \\ Y &= xS\theta + yC\theta, \\ Z &= z. \end{aligned} \tag{5.10}$$

Матрица для преобразования касательных

$$J = \begin{bmatrix} C\theta & -S\theta & -xS\theta f'(z) - yC\theta f'(z) \\ S\theta & C\theta & xC\theta f'(z) - yS\theta f'(z) \\ 0 & 0 & 1 \end{bmatrix}$$

Определитель матрицы (5.10) равен 1, поэтому преобразование скручивания сохраняет объем объекта.

Матрица преобразования нормалей

$$(J^{-1})^T = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ yf'(z) & -xf'(z) & 1 \end{bmatrix}$$

Обратное преобразование (скручивание в обратном направлении) задается соотношениями:

$$\begin{aligned} \theta &= f(z), \\ x &= XC\theta + yS\theta, \\ y &= -XS\theta + yC\theta, \\ z &= z. \end{aligned} \tag{5.11}$$

Общий изгиб вокруг оси x . Это преобразование также часто используется в практике. Угол изгиба задается следующим образом:

$$\begin{aligned} \theta &= k(\hat{y} - y_0), \\ C\theta &= \cos\theta, \\ S\theta &= \sin\theta, \end{aligned}$$

где K - кривизна изгиба, т.е. $1/K$ - радиус изгиба, y_0 - центр изгиба,

$$\hat{y} = \begin{cases} y_{\min}, & \text{если } y < y_{\min}, \\ y, & \text{если } y_{\min} \leq y \leq y_{\max}, \\ y_{\max}, & \text{если } y > y_{\max}. \end{cases}$$

Направление изгиба задается вектором $(S, y_0, \frac{1}{K})^T$, где S - параметр линии.

Преобразование изгиба задается следующими выражениями:

$$X = x$$

$$y = \begin{cases} -S\theta \left(z - \frac{1}{K}\right) + y_0, & \text{при } y_{\min} \leq y \leq y_{\max}, \\ -S\theta \left(z - \frac{1}{K}\right) + y_0 + C\theta (y - y_{\min}), & \text{при } y < y_{\min}, \\ -S\theta \left(z - \frac{1}{K}\right) + y_0 + C\theta (y - y_{\max}), & \text{при } y > y_{\max}. \end{cases}$$

$$z = \begin{cases} C\theta \left(z - \frac{1}{K}\right) + \frac{1}{K}, & \text{при } y_{\min} \leq y \leq y_{\max}, \\ C\theta \left(z - \frac{1}{K}\right) + \frac{1}{K} + S\theta (y - y_{\min}), & \text{при } y < y_{\min}, \\ C\theta \left(z - \frac{1}{K}\right) + \frac{1}{K} + S\theta (y - y_{\max}), & \text{при } y > y_{\max}. \end{cases}$$

Эти функции непрерывны в пределах каждой из трех областей значений y . Однако, при $y = y_{\min}$ и $y = y_{\max}$ производная имеет скачки. Нарушение непрерывности производной можно избежать, выбрав другую, гладкую функцию $\theta = f(y)$. В этом случае матрицы преобразований должны быть выведены заново.

Матрица преобразования касательных

$$J = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta(1 - \hat{K}z) & -S\theta \\ 0 & S\theta(1 - \hat{K}z) & C\theta \end{bmatrix},$$

где

$$\hat{K} = \begin{cases} K, & \text{если } \hat{y} = y; \\ 0, & \text{если } \hat{y} \neq y. \end{cases}$$

Матрица преобразования нормалей

$$(1 - \hat{K}z)(J^{-1})^T = \begin{bmatrix} 1 - \hat{K}z & 0 & 0 \\ 0 & C_\theta & -S_\theta(1 - \hat{K}z) \\ 0 & S_\theta & C_\theta(1 - \hat{K}z) \end{bmatrix}$$

Обратное преобразование задается выражениями

$$\theta_{min} = \kappa(y_{min} - y_0),$$

$$\theta_{max} = \kappa(y_{max} - y_0),$$

$$\hat{\theta} = -\text{tg}^{-1} \frac{y - y_0}{z - \frac{1}{\kappa}};$$

$$\theta = \begin{cases} \theta_{min}, & \text{при } \theta < \hat{\theta}_{min}, \\ \theta, & \text{при } \hat{\theta}_{min} \leq \hat{\theta} \leq \theta_{max}, \\ \theta_{max}, & \text{при } \hat{\theta} > \theta_{max}, \end{cases}$$

$$x = X,$$

$$\hat{y} = \frac{\theta}{\kappa} + y_0,$$

(5.13)

$$\hat{y} = \begin{cases} \hat{y}, & \text{при } y_{min} < \hat{y} < y_{max} \\ (y - y_0)C_\theta + (z - \frac{1}{\kappa})S_\theta + \hat{y}, & \text{при } \hat{y} = y_{min}, \\ & \text{или при } \hat{y} = y_{max}; \end{cases}$$

$$z = \begin{cases} \frac{1}{\kappa} + \sqrt{(y - y_0)^2 + (z - \frac{1}{\kappa})^2}, & \text{при } y_{min} < \hat{y} < y_{max}, \\ -(y - y_0)S_\theta + (z - \frac{1}{\kappa})C_\theta + \hat{y}, & \text{при } \hat{y} = y_{min} \text{ или } \hat{y} = y_{max}. \end{cases}$$

Деформация пространства кривых и поверхностей. Предполагается, что известен якобиан $J(x)$, но функция преобразования координат $\bar{X} = F(x)$ в явном виде не задана (отсутствует аналитическое выражение для преобразователя). Исходная кривая задана в параметрической форме

$$\bar{x} = \bar{x}(s), \quad s_0 \leq s \leq s_1.$$

Требуется определить кривую $\bar{X}(s)$ после деформации.

По определению

$$X(s) = \int_0^s \bar{X}'(s) ds + x_0 = \int_0^s J(\bar{x}(s)) \bar{x}'(s) ds + \bar{x}_0, \quad (5.14)$$

где $\bar{x}'(s)$ - производная (касательный вектор) исходной кривой $\bar{x}(s)$.

Преобразование поверхности или всего объекта выполняется аналогично. Во-первых, выбирается начало координат $\bar{0}$ на преобра-

зуюмом объекте. Для каждой точки \bar{x} поверхности объекта выбирается кусочно-гладкая пространственная кривая, соединяющая начало координат с точкой \bar{x} . Эта пространственная кривая подвергается деформации. Если $J(\bar{x})$ действительно является якобианом некоторой (неизвестной) функции деформации $\bar{X} = F(\bar{x})$, то преобразование \bar{x} в \bar{X} единственно, так как все гладкие пути, соединяющие $\bar{0}$ и \bar{x} , эквивалентны. Если уравнение поверхности задается в параметрической форме $\bar{x} = \bar{x}(\mu, \nu)$, то пространственная кривая на поверхности может быть получена в однопараметрической форме, например:

$$\mu = \mu(s), \quad \nu = \nu(s),$$

$$\hat{x}(s) = \bar{x}(\mu(s), \nu(s)).$$

Эта пространственная кривая должна быть кусочно-дифференцируемой, чтобы производные могли быть вычислены и проинтегрированы.

Уравнения для этой деформированной кривой

$$\begin{aligned} X(\mu(s), \nu(s)) &= \int_0^s J(x(\mu(\hat{s}), \nu(\hat{s}))) \bar{x}'(\mu(\hat{s}), \nu(\hat{s})) d\hat{s} + \\ + x_0 &= \int_0^s J(\bar{x}(\mu(\hat{s}), \nu(\hat{s}))) \left(\frac{\partial \bar{x}}{\partial \mu} \mu'(s) + \frac{\partial \bar{x}}{\partial \nu} \nu'(s) \right) d\hat{s} + \bar{x}_0. \end{aligned} \quad (5.17)$$

Столбцы якобиана $J(\bar{x})$ должны удовлетворять условию

$$J_{i,j} = J_{j,i}. \quad (5.18)$$

Только в этом случае применены преобразования для касательных и нормальных векторов. Значения якобиана могут быть непосредственно связаны со свойствами моделируемого материала, но можно также использовать уравнения пластичности и эластичности.

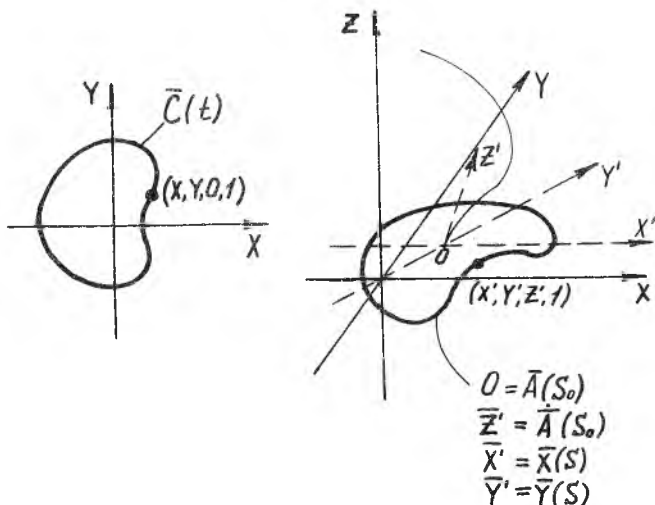
При вычислении деформаций поверхности удобно использовать прямоугольную сетку в пространстве параметров $\mu - \nu$. При этом размеры ячеек сетки могут рекурсивно делиться, если соседние нормы резко меняют направление. Тогда области объекта с большой кривизной поверхности рассматриваются более подробно.

6. ОБОБЩЕННЫЕ ЦИЛИНДРЫ

6.1. Определения и допущения

Идея обобщенного цилиндра, как и многие крупные идеи, проста. Обобщенные цилиндры представляют собой описания, которые состоят из оси и двумерной фигуры, перемещающейся вдоль этой оси. Ось не обязана быть прямой линией, а двумерная фигура кругом. Двумерная фигура может быть сколь угодно сложной и, кроме того, не обязана оставаться постоянной /28/.

Формальное описание обобщенного цилиндра можно представить следующим образом (рис. 6.1). Главная ось является пространственной линией $A(S)$, где S — параметр, изменяющийся от 0 до 1.



Р и с. 6.1. Определение обобщенного цилиндра

Для декартовой системы координат можно записать

$$\bar{A}(S) = [x_A(S), y_A(S), z_A(S)], \quad 0 \leq S \leq 1, \quad (6.1)$$

где координаты $x_A(S)$, $y_A(S)$, $z_A(S)$ являются функциями одного параметра S .

Кривая в сечении обобщенного цилиндра $C(t)$ определяется в плоскости X^4 аналогичным образом:

$$\bar{C}(t) = [x_c(t), y_c(t), z_c(t)], \quad 0 \leq t \leq 1.$$

Учитывая, что $z_c(t) = 0$, имеем

$$\bar{C}(t) = [x_c(t), y_c(t), 0], \quad 0 \leq t \leq 1. \quad (6.2)$$

Кривая $\bar{C}(t)$ замкнута, поэтому $\bar{C}(0) = \bar{C}(1)$. Точка на поверхности обобщенного цилиндра, следовательно, является функцией двух параметров S и t :

$$\begin{aligned} \bar{D}(s, t) &= \bar{A}(s) + x_c(t)\bar{X}(s) + y_c(t)\bar{Y}(s) + z_c(t)\bar{Z}(s) = \\ &= \bar{A}(s) + x_c(t)\bar{X}(s) + y_c(t)\bar{Y}(s). \end{aligned} \quad (6.3)$$

Здесь $\bar{X}(s), \bar{Y}(s), \bar{Z}(s)$ — ортогональные трехмерные единичные векторы, задающие локальную систему координат, относительно которой определяется кривая $\bar{C}(t)$ в уравнении (6.2). Вектор $\bar{Z}(s)$ является касательной к кривой $\bar{A}(s)$ в точке ее пересечения плоскости сечения обобщенного цилиндра. Уравнение (6.3) позволяет определить ориентацию обобщенного цилиндра через ориентацию плоскости сечения при ее движении от начальной точки ($S=0$) к конечной точке ($S=1$) оси. Вторым параметром для секущей плоскости является угол θ поворота кривой $\bar{C}(t)$ в этой плоскости относительно локальной системы координат, т.е. вокруг оси Z . Третий параметр характеризует изменение формы кривой в сечении. Таким образом, поверхность обобщенного цилиндра образуется интерполяцией через заданные в дискретных точках главной оси сечения в виде кривых произвольной формы.

Иногда вводится дополнительный параметр в определении обобщенного цилиндра — угол наклона α сечения цилиндра к главной оси (угол между плоскостью сечения и касательной к главной оси в точке пересечения осью плоскости). Этот угол может быть постоянным, хотя и отличаться от $\pi/2$, а может и изменяться. При $\alpha = \frac{\pi}{2}$ обобщенный цилиндр иногда называют прямым. Следует отметить, что использование данного параметра сильно увеличивает неоднозначность

ность описания цилиндра, так как даже простые формы, например круговой цилиндра, могут быть представлены бесконечным числом различных обобщенных цилиндров.

Для практических целей обобщенный цилиндр удобнее представлять в дискретной форме. Неформальное определение обобщенного цилиндра в таком случае включает следующие положения:

главная ось является произвольной непрерывной трехмерной кривой;

каждое сечение является замкнутой, произвольной непрерывной планарной кривой;

сечения перпендикулярны главной оси;

главная ось проходит через центры тяжести сечений (или их приближения);

сечения определены явным образом только в дискретных точках вдоль главной оси.

Таким образом, дискретный обобщенный цилиндр (ДОЦ) представляется списком сечений. Каждое сечение является замкнутой непрерывной кривой, представляемой списком двумерных точек вдоль его границы (эти точки иногда называются *управляющими*), которые служат для интерполяции кривой.

Точки для сечения задаются относительно собственной (местной) координатной системы, начало которой точка $[0, 0]$ является центром тяжести сечения. Каждая такая центральная точка связана с трехмерной координатной системой всего обобщенного цилиндра. Главная ось является трехмерной пространственной кривой, интерполирующей этот список трехмерных точек. Следовательно, главная ось определяет ориентацию плоскостей сечений или, более точно, ориентацию местных координатных систем для каждого сечения.

Формальное определение дискретного обобщенного цилиндра:

$\langle \text{обобщенный цилиндр} \rangle ::= \{ \langle \text{сечение I} \rangle, \dots, \langle \text{сечение N} \rangle \}$

$\langle \text{сечение} \rangle ::= (\langle \text{центр} \rangle, \langle \text{управляющие точки} \rangle)$

$\langle \text{центр} \rangle ::= [x, y, z]$

$\langle \text{управляющие точки} \rangle ::= \{ \langle \text{узел I} \rangle, \dots, \langle \text{узел M} \rangle \}$

$\langle \text{узел} \rangle ::= [x, y],$

где $\{ \dots \}$ обозначает список элементов,

(\quad) означает пару элементов,

$[\quad]$ обозначает двумерные или трехмерные координаты

Координаты центральных точек ("центров") определяются в системе координат объекта, а координаты "узлов" — в системах координат сечений.

Последние исследования показали, что для представления сечений главной оси наиболее удобным при практических применениях обобщенных цилиндров является использование кубических B -сплайнов. В этом случае мы имеем дело с функциями одного параметра, что обуславливает следующие преимущества:

э ф ф е к т и в н а я и н т е р п о л я ц и я: каждый сегмент кривой, расположенный между двумя узлами, вычисляется с помощью нескольких матричных произведений, большинство из которых могут быть выполнены один раз в начале работы;

л о к а л ь н о с т ь: форма каждого сегмента зависит только от ближайших узлов точек, поэтому изменение любой такой точки мало изменяет кривую в целом;

г л а д к о с т ь: поверхность обобщенного цилиндра имеет три степени непрерывности в узловых точках: по положению, наклону и кривизне;

в о з м о ж н о с т ь а ф ф и н н ы х п р е о б р а з о в а н и й: аффинное преобразование кривой сводится к аффинному преобразованию узловых точек, определяющих данную кривую.

В табл. 6.1 сравниваются характеристики различных методов представления кривых в обобщенных цилиндрах: помимо B -сплайнов рассмотрены полигональные кривые (ломанные), полиномы, кривые второго порядка (конические сечения).

Т а б л и ц а 6.1

Метод	Локальность	Число управляющих точек	Точность эффективности	Гладкость	Примечание
Ломанные	Да	Много	Высокая (высокая)	Нет	Кусочно-линейные
Полиномы	Нет	Среднее	Низкая (низкая)	Да	-
Конические сечения	Нет	Несколько	Приемлемая (низкая)	Да	Ограниченные, частные кривые
B -сплайны	Да	Среднее	Приемлемая (высокая)	Да	Произвольные кривые кусочно-полиномиальные

6.2. Операции с обобщенными цилиндрами

Этапы определения положения точки на поверхности обобщенного цилиндра. Чтобы определить положение точки на поверхности обобщенного цилиндра, нужно выполнить несколько этапов интерполяции.

1. По значению параметра S определить соответствующее сечение, т.е. местную координатную систему и выполнить преобразование от координатной системы объекта к местной координатной системе.

2. Если точка не является узловой точкой главной оси, найти два ближайших "дискретных" сечения и определить на них точки с одинаковым значением параметра t .

3. Используя преобразование, найденное на этапе 1, перейти от точки, полученной на этапе 2, к трехмерной координатной системе обобщенного цилиндра.

Эту схему можно улучшить, если на этапе 2 взять два сечения, соответствующих значениям S , непосредственно следующих за данным значением S , и выполнить квадратичную или кубическую интерполяцию.

Ориентация обобщенного цилиндра. Первый этап может быть назван проблемой ориентации, во время которой точки двумерной кривой, определяющей сечение ДОЦ в некоторой точке главной оси преобразуются в пространственные точки на поверхности ДОЦ.

Точки будем задавать однородными координатами, т.е. четырех-элементными векторами (для двумерных точек координата Z равна нулю). Тогда все геометрические преобразования выражаются с помощью матриц 4×4 . В нашем случае

$$\bar{C}_S(t) \bar{T}_S = [x_{C_S}(t) \ y_{C_S}(t) \ 0 \ 1] \bar{T}_S = [x_D(S, t) \ y_D(S, t) \ z_D(S, t) \ 1] = \bar{D}(S, t), \quad (6.4)$$

где $\bar{D}(S, t)$ является точкой на поверхности ДОЦ, а $\bar{C}_S(t)$ - кривая сечения, соответствующая точке S на главной оси.

Преобразователь \bar{T}_S может быть представлено как композиция сдвига L_S и поворота R_S вокруг некоторой оси, проходящей через центр сечения:

$$T_S = R_S L_S \quad (6.5)$$

В начальный момент преобразования T_0 размещает плоскость $X'Y'$ перпендикулярно главной оси в начальной точке $\bar{A}'(0)$. Введем следующие обозначения:

$$\begin{aligned} \bar{A}'(S_i) & - \text{нормализованный касательный вектор в точке } S_i \\ & \text{главной оси } \bar{A}(S_i); \\ L_i & - \text{преобразование сдвига начала координат в точку } \bar{A}(S_i); \\ \Delta L_i & - \text{сдвиг от } \bar{A}(S_i) \text{ к } \bar{A}(S_{i+1}), \\ \Delta L_i & = L_i^{-1} L_{i+1}, \end{aligned} \quad (6.6)$$

$$\begin{aligned} \Delta T_i & - \text{преобразование от } \bar{A}(S_i) \text{ к } \bar{A}(S_{i+1}), \text{ сохраняющее} \\ & \text{перпендикулярность сечения к главной оси. В случае} \\ & \text{ДОО данное преобразование представляется совмещени-} \\ & \text{ем локального поворота } \Delta R_i \text{ и локального сдвига} \\ & \Delta L_i : \\ \Delta T_i & = \Delta R_i \Delta L_i \end{aligned} \quad (6.7)$$

$$\begin{aligned} \Delta R_i & - \text{преобразование поворота сущей плоскости в } \bar{A}(S_i) \\ & \text{к плоскости, параллельной сечению в } \bar{A}(S_{i+1}), \\ \Delta R_i & = L_i^{-1} R_i L_i, \end{aligned} \quad (6.8)$$

R_i - поворот от касательного вектора $A(S_i)$ к касательному вектору $A'(S_{i+1})$ по кратчайшему пути, определяемому соотношением (6.9).

При $i = 0$ это преобразование от вектора $\bar{z} = [0011]$ к $\bar{A}'(S_1)$.

Поворот R_i выполняется вокруг вектора \bar{V}_i на угол α_i :

$$\bar{V}_i = \bar{A}'(S_i) \times \bar{A}'(S_{i+1}), \quad \alpha_i = \arccos [\bar{A}'(S_i) \bar{A}'(S_{i+1})].$$

Обозначим также

$$\begin{aligned} C_S & = \cos(\alpha_i) = \bar{A}'(S_i) \bar{A}'(S_{i+1}), \\ S_n & = \sin(\alpha_i) = |\bar{V}_i|, \quad \bar{V}_i' = \frac{\bar{V}_i}{|\bar{V}_i|} = [a \ b \ c \ 1]. \end{aligned} \quad (6.9)$$

Тогда матрица поворота R_i может быть записана следующим образом:

$$R_i = \begin{bmatrix} a^2 + C_S(1-a^2) & a\beta(1-C_S) - CS_n & a\alpha(1-C_S) + \beta S_n 0 \\ a\beta(1-C_S) + CS_n & \beta^2 + C_S(1-\beta^2) & \beta\alpha(1-C_S) - aS_n 0 \\ a\alpha(1-C_S) - \beta S_n & \beta\alpha(1-C_S) + aS_n & \alpha^2 + C_S(1-\alpha^2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Отметим, что $\Delta R_0 = R_0$ и $L_0 = I$ (единичная матрица), тогда из уравнения (6.6)

$$\Delta L_0 = L_1, \quad T_0 = L_0^{-1} R_0 L_1 = R_0 L_1.$$

Затем, подставляя уравнения (6.6) и (6.8) в (6.7), получим

$$\Delta T_i = L_i^{-1} R_i L_i L_i^{-1} L_{i+1} = L_i^{-1} R_i L_{i+1}$$

и

$$T_i = \prod_{j=0}^{i-1} \Delta T_j = (L_0^{-1} R_0 L_1)(L_1^{-1} R_1 L_2) \cdots (L_{i-1}^{-1} R_i L_i) =$$

$$= \left(\prod_{j=0}^{i-1} R_j \right) L_i = R_S L_S, \quad (6.10)$$

где $\prod_{j=0}^{i-1} R_j$ соответствует суммарному изменению ориентации сечения вдоль точек главной оси, а L_S — характеризует общий сдвиг.

Интерполяция между сечениями. Рассмотрим теперь вопрос определения тех точек на поверхности обобщенного цилиндра, которые находятся между двумя заданными сечениями. В этом случае находят два сечения S_i и S_{i+1} , ближайших к заданной точке $A(S)$ главной оси, таких что $S_i < S < S_{i+1}$, и затем выполняют линейную интерполяцию

$$C_S(t) = \frac{C_i(t)(S - S_i) + C_{i+1}(t)(S_{i+1} - S)}{S_{i+1} - S_i}, \quad (6.11)$$

где $C_i(t)$ — контур сечения в точке $A(S_i)$ главной оси.

Лучшее решение можно получить, используя интерполяционную схему более высокого порядка. Например, выбрав четыре сечения (вместо двух), можно осуществить интерполяцию B — сплайном:

$$C_S(t) = [(s')^3 (s')^2 s' \dots 1] [C] [V_{C_S}(t)]^T, \quad (6.12)$$

где

$$s' = \frac{s - s_i}{s_{i+1} - s_i}$$

и $[V_{C_S}(t)]$ является вектором управляющих точек для этого сплайна. Управляющие точки могут быть получены из уравнений (4.4)–(4.7) разд. 4.2., в которых точки интерполяции $[P_{C_S}(t)]$ получаются при заданном t из выражения

$$[P_{C_S}(t)] = [C_{i-1}(t) C_i(t) C_{i+1}(t) C_{i+1}(t)]. \quad (6.13)$$

Когда $i=1$ или $i=n-1$, т.е. для первого или последнего, соответственно, куска главной оси используются формулы, аналогичные уравнению (6.4).

Визуализация обобщенного цилиндра. Для отображения (визуализации) поверхности ДОЦ важно иметь помимо координат узловых точек также и их топологию. ДОЦ позволяет легко получить любую топологическую информацию. Например, если каждое поперечное сечение ДОЦ поделено на равное число кусков, имеющих одинаковую длину в параметрическом пространстве, то тем самым на поверхности ДОЦ создается прямоугольная сетка $n \times m$ узлов в двумерном параметрическом пространстве. Каждый такой узел является точкой (s_i, t_j) , $i=1, \dots, n, j=1, \dots, m$, в параметрическом пространстве с мировыми координатами $\bar{D}(s_i, t_j)$. Поверхность ДОЦ представляется билинейными или бикубическими кусками, в зависимости от вида используемой аппроксимации.

Очень легко выполняется триангуляция (разбиение на треугольники) поверхности ДОЦ. Для этого достаточно разделить диагональ прямоугольника, образованный узлами $(s_i, t_j), (s_{i+1}, t_j), (s_i, t_{j+1}), (s_{i+1}, t_{j+1})$. Такая триангуляция может быть полезна при расчетах методом конечных элементов, анализе видимости методом приоритетов Энкарнако и т.д.

Вычисление характеристик обобщенного цилиндра. Если главная ось ДОЦ проходит через центры масс поперечных сечений, то достаточно просто могут быть получены оценки объема и площади поверхности обобщенного цилиндра. Объем обобщенного цилиндра получается интегрированием площади поперечного сечения вдоль главной оси. Если

$A_{\text{сеч}}(s)$ - площадь сечения в точке S главной оси, $L(A(s))$ - длина главной оси, объем $V_{\text{об}}$ равен:

$$V_{\text{об}} = \int_0^1 \left\{ A_{\text{сеч}}(s) \frac{\partial}{\partial x} L[A(s)] \right\} ds. \quad (6.14)$$

Учитывая, что контур сечения и главная ось представлены B -сплайнами и используя соотношения (4.10)-(4.15) интеграл (6.14) можно аппроксимировать суммой

$$V_{\text{об}} \approx \sum_{i=1}^{n-1} A_{\text{сеч}}(s_i) L(A(s_i)), \quad (6.15)$$

где $L(A(s_i))$ - длина i -го куска главной оси, которая может быть получена из соотношений (4.17)-(4.19).

Аналогичное соотношение может быть получено и для площади поверхности обобщенного цилиндра. В этом случае вместо $A_{\text{сеч}}(s_i)$ должна вычисляться $L(C_{s_i})$ - длина контура сечения $C_{s_i}(t)$.

7. ВИЗУАЛИЗАЦИЯ РЕЗУЛЬТАТОВ ГЕОМЕТРИЧЕСКОГО МОДЕЛИРОВАНИЯ

7.1. Анализ видимости

Общие сведения о задаче анализа видимости. Задача удаления невидимых линий в трехмерных сценах является одной из наиболее интересных в машинной графике. Эта задача в самом общем виде может быть сформулирована как нахождение преобразования, отображающего множество объектов в множество их видимых частей в двумерном пространстве картинной плоскости (изображения). Видимость различных частей объектов сцены зависит от точки, с которой сцена рассматривается. Значение имеет также глубина, т.е. расстояние от различных граней до точки наблюдения.

Существует большое число алгоритмов анализа видимостей, различающихся сложностью обрабатываемых объектов, качеством синтезируемых изображений и т.д. Однако, практически все алгоритмы удаления невидимых элементов требуют значительных вычислительных затрат, поэтому при выборе метода анализа видимости следует ограничиваться наиболее простым, но создающим графический образ, достаточный для решения поставленной задачи. Следует учитывать, например, что переход от "проволочного" к полутонному изображению увеличит время анализа в тысячи раз. По этой же причине часто заменяют более ес-

тественное центральное проецирование его аппроксимацией — ортогональным проецированием, что значительно упрощает задачу анализа видимости.

Иногда полезно определить заведомо невидимые грани объекта, которые сразу можно исключить из дальнейшей обработки. Остальные грани будут считаться потенциально видимыми (потенциально видимые грани могут быть заслонены другими объектами).

Грань потенциально видима, если угол между лучом зрения и нормалью к грани меньше или равен $\pi/2$, и невидима, если угол больше $\pi/2$. Пусть N — вектор, перпендикулярный грани; L — луч зрения, т.е. прямая, проходящая через точку наблюдения и основание нормали; тогда достаточно вычислить скалярное произведение:

$$\vec{L} \cdot \vec{N} = |\vec{L}| |\vec{N}| \cos \psi_{LN}.$$

и проверить его знак ($\cos \psi_{LN}$ меняет знак при $\psi_{LN} = \frac{\pi}{2}$).

Такой метод позволяет выявить невидимые грани. К потенциально видимым граням необходимо применять дополнительные тесты, реализованные в виде различных алгоритмов анализа видимости. Процедура определения невидимости занимает сравнительно немного времени. Зато процедура определения видимости является медленной, так как для определения видимости элемента необходимо перебрать все грани, которые могут его заслонить. Поэтому предварительный отбор (отсев) явно невидимых граней может сильно сократить расход машинного времени при решении задачи удаления невидимых линий или поверхностей. Для единичных выпуклых объектов проверка по тесту видимости полностью решает проблему видимости.

Рассмотрим пять наиболее известных алгоритмов анализа видимости.

Метод приоритетов Энкарнако. Метод применим к многогранникам (выпуклым и вогнутым) и лучше всего подходит к теоретико-графовой геометрической модели объекта.

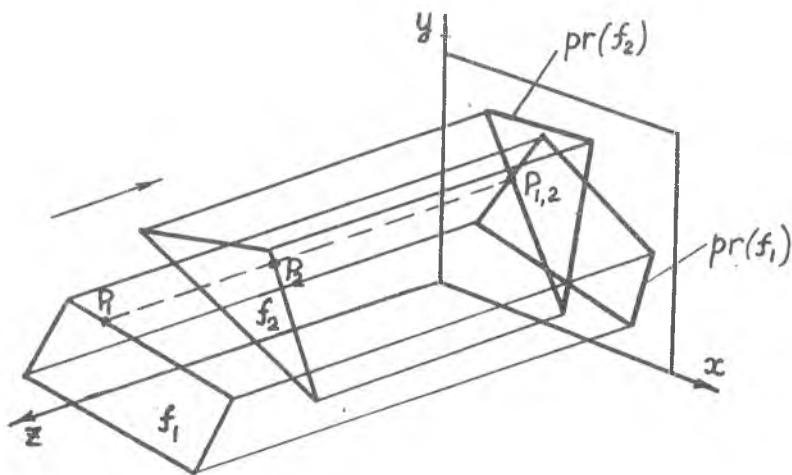
Алгоритм удаления невидимых поверхностей разбивается на четыре этапа:

- триангуляция;
- предварительное упорядочение;
- приписывание приоритетов;
- определение видимости.

На этапе триангуляции все грани разбиваются на треугольники путем проведения дополнительных (вспомогательных) ребер.

На этапе предварительного упорядочения выделяются все треугольники, перпендикулярные плоскости XU , т.е. картинной плоскости. Грань перпендикулярна картинной плоскости, если проекции ее вершин коллинеарны. Эти треугольники не заслонят другие грани, поэтому на следующих этапах они не рассматриваются.

Говорят, что грань f_i имеет приоритет над f_j , (рис. 7.1), если f_i ближе к точке наблюдения, чем f_j . Для каждого треугольника t_i ищется множество T_i всех треугольников, имеющих больший приоритет, чем t_i . Для этого вначале определяется, с какими треугольниками t_j пересекается данный треугольник t_i . Изолированные треугольники исключаются из рассмотрения. Для определения факта пересечения используется максимальное условие, т.е. считается, что треугольники t_1 и t_2 не перекрываются друг с другом, если справедливо хотя бы одно из следующих условий



$$z_1 > z_2 \Rightarrow f_1 \supseteq f_2$$

Р и с. 7.1. Определение приоритетов

$$\begin{aligned} \max(x_{1,j}) &< \min(x_{2,k}), \\ \max(x_{2,k}) &< \min(x_{1,j}), \\ \max(y_{1,j}) &< \min(y_{2,k}), \\ \max(y_{2,k}) &< \min(y_{1,j}), \\ j, k &= 1, 2, 3. \end{aligned}$$

Простое минимаксное условие требует выполнения лишь немногих логических операций, и такая предварительная проверка помогает уменьшить время выполнения программы.

Для остальных треугольников ищутся точки пересечения. При этом лучше использовать критерий (тест) пересечения сторон треугольника, а не тест принадлежности вершин треугольника внутренней части другого треугольника. Второй тест более сложный, так как включает вычисление точек пересечений, и, кроме того, должен быть выполнен для обоих треугольников. Если точек пересечения нет, то проверяется вложенность одного треугольника в другой.

Таким образом, находится множество треугольников T_i покрывающих, хотя бы частично, треугольник t_i .

Если $T_i = \emptyset$ (пустое множество), то ребра t_i полностью видимы (за исключением вспомогательных ребер, которые не отображаются).

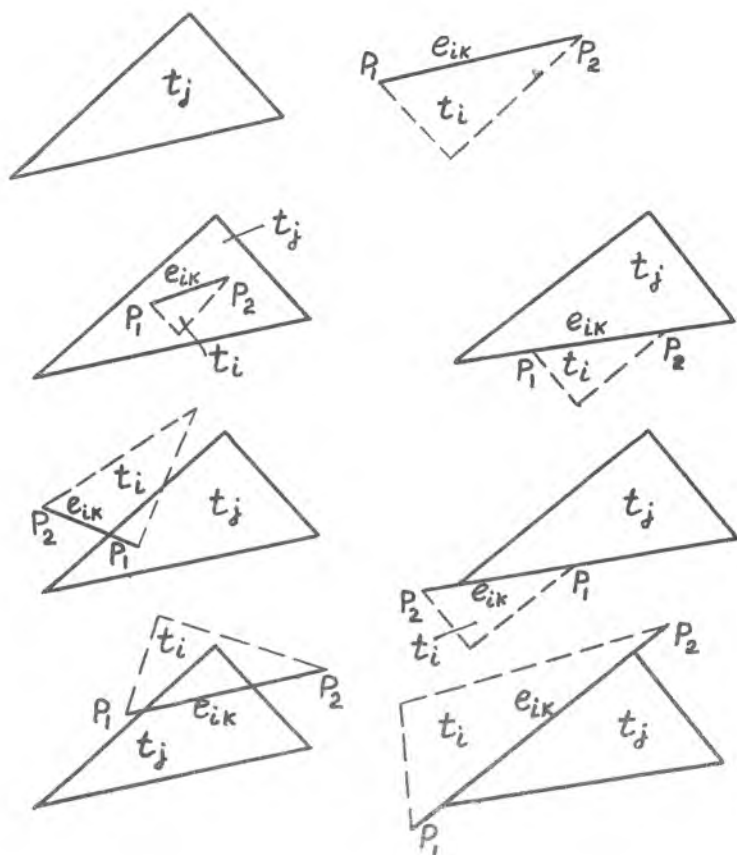
Для непустого T_i необходимо исследовать видимость каждого ребра треугольника t_i , т.е. ребер a_i, b_i, c_i по отношению к каждому треугольнику t_j из множества T_i .

Видимость одного из ребер треугольника t_i по отношению к треугольнику t_j ($t_i - t_j$) можно определить, найдя точки пересечения этих ребер с треугольником t_j и проверив, где находятся конечные точки ребра — внутри или вне t_j . Часть ребра, которая лежит внутри t_j , невидима. Часть ребра a_i, b_i или c_i , лежащая вне t_j , является потенциально видимой. Окончательное решение о видимости ребра можно принять только после исследования всех треугольников.

Различные случаи анализа видимости ребра показаны на рис. 7.2.

Алгоритм Варнока. Основной операцией в алгоритме Варнока является определение элементов в некоторой области экрана. Если элемент достаточно прост для непосредственного изображения, то выполняется

его построение. Если совокупность элементов в области слишком сложна, то данная область делится на меньшие части и правило применяется повторно.



Р и с. 7.2. Случаи анализа видимости в методе приоритетов Энкарнако

Количество делений окна, необходимых для построения изображения, зависит от выбора критерия сложности изображения. В одном из вариантов алгоритма информация в окне никогда не считается простой и окно делится до тех пор, пока не уменьшится до величины, равной разрешающей способности экрана.

Алгоритм Варнока состоит из пяти основных блоков: просмотра, решения, вывода элемента, вывода точки, управления.

Блок просмотра исследует конкретное окно и определяет, какие части объектов сцены (если они имеются) видны в данном окне.

Блок решения определяет, возможно ли непосредственное изображение элементов в данном окне. Если "да", то он вызывает блок вывода элемента. Если блок решения находит ситуацию слишком сложной, то заявление об отказе, вследствие чего блок управления производит деление окна на части и передает их на повторное исследование в блок просмотра.

Блок управления ведет также список неисследованных окон.

Если блок решения выдает отказ, а размер окна настолько мал, что оно покрывает только один элемент растра экрана, то тогда блок управления вызывает блок вывода точки для отображения на экране отдельной точки. Такой метод обработки универсален. Его можно применить для криволинейных поверхностей и для областей экрана любой желаемой формы.

Методика деления экрана для получения отображаемых элементов может быть снабжена тривиальными или очень сложными блоками просмотра и решения.

Рассмотрим подробнее работу блоков алгоритма Варнока.

Задачей блока просмотра является классификация многоугольников на охватывающие окно на экране (охват); не связанные с окном (несвязность); пересекающие данное окно (пересечение). Кроме того блок просмотра может вычислять некоторые характеристики пересечения (варианты) — пересечение ребром, одна вершина в окне и т.д. Если хотя бы одна сторона многоугольника пересекает окно, то многоугольник является пересекающим. Пересечение окна некоторой стороной многоугольника можно установить путем подстановки координат угловых точек окна в уравнение стороны. Если знаки четырех полученных значений будут одинаковы, то все четыре угла находятся по одну сторону от прямой, и прямая не пересекает окно. Если пересечение отсутствует, то имеется либо охват, либо несвязность.

Простая проверка по минимаксному критерию отсекает явно несвязные многоугольники. Проверка может быть выполнена очень быстро, если для каждого многоугольника хранятся минимальные и макси-

мальные значения координат всех вершин, которые дают границы описывающего прямоугольника. Сравнение этого прямоугольника с окном немедленно устанавливает необходимость дальнейших вычислений. Если эта проверка не устанавливает несвязанности, то проводится дополнительный анализ.

Функция блока решения – удаление невидимых линий. Если блок просмотра не обнаружил ни одного многоугольника, охватывающего или пересекающего окно, то окно, очевидно, пусто. Если есть охватывающий многоугольник, то определяется, закрывает ли он остальные многоугольники в окне. Для этого сравниваются глубины (Z – координаты) плоскостей многоугольников в четырех угловых точках окна. Если глубина охватывающего многоугольника во всех углах окна меньше соответствующих глубин всех других многоугольников, то этот охватывающий многоугольник действительно заслоняет все другие возможные элементы в данном окне, т.е. окно будет пусто.

Если такой охватывающий многоугольник не находится, то блок решения объявляет ситуацию слишком сложной для анализа, и блок управления осуществляет деление окна. Число делений окна можно сократить, если применить более сложный блок решения (но, с другой стороны, сложный блок решения может быть более медленным, чем несколько дополнительных делений в простом блоке решения).

Варианты более сложных блоков решения:

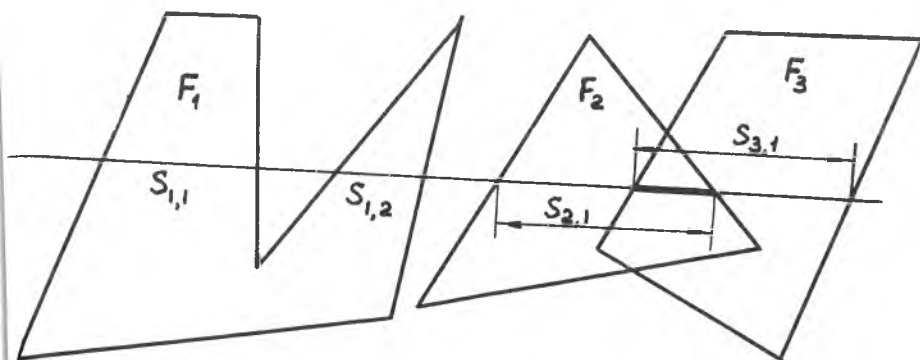
выделение случая, когда нет охвата и одно пересечение; тогда отображаются все части сторон многоугольника, попадающие внутрь окна (отсечение стороны пересекающего многоугольника, попадающей в пределы окна);

выделение случая, когда есть удаленный охватывающий многоугольник и единственный пересекающий многоугольник, лежащий полностью впереди охватывающего.

Алгоритм сканирующей прямой Уоткинса. Алгоритм сканирующей прямой удобен при использовании растровых дисплеев, подобных телевизионному монитору, в которых изображение формируется последовательно, строка за строкой. Алгоритм основан на использовании сканирующих прямых, накладываемых на изображение. Предполагается, что сканирующие прямые горизонтальны. Все сканирующие прямые обрабатываются одним способом, поэтому достаточно рассмотреть одну сканирующую прямую.

Алгоритм состоит из двух частей: нахождение "пробных интервалов", определение видимости.

Сначала находят отрезки сканирующей прямой, лежащие внутри многоугольников (рис. 7.3). Отрезки могут быть изолированными ($S_{1,1} \cup S_{1,2}$), или перекрывающимися ($S_{2,1} \cup S_{3,1}$). Изолированные отрезки видимы, так как их ничего не загораживает.



Р и с. 7.3. Сканирующая прямая

Для определения видимости перекрывающихся отрезков формируются пробные интервалы. Пробный интервал – это отрезок сканирующей прямой, на котором видимость не изменяется. Если допустить, что грани в объекте (мировом) пространстве не могут проникать друг в друга, то начальные точки пробных интервалов – это точки, в которых начинается или заканчивается перекрытие отрезков. Если же грани перекрываются, то необходимо рассмотреть также проекции на плоскость XZ .

Таким образом, границей пробного интервала может быть либо смежные отрезки, либо точка их пересечения.

Видимость отрезков нужно анализировать только внутри пробных интервалов, поскольку если в области $[x_1, x_2]$ имеется только один отрезок, то он всегда видимый. Если на пробном интервале число отрезков больше или равно 2, видимость определяют, используя глубину каждого отрезка (координату z). Тот отрезок, который находится ближе к точке наблюдения (z – координата меньше), закрывает остальные отрезки и является видимым внутри пробного интервала.

Процесс построения изображения можно значительно ускорить, используя предварительное упорядочение элементов по максимальному значению Y -координаты (при сканировании сверху вниз), а также сходство сканируемых строк (иногда сходство соседних строк называют когерентностью строк). Предварительное упорядочение элементов изображения по Y -координате позволяет быстро отбросить те элементы изображения, которые лежат в стороне от сканирующей прямой и ускорить процесс формирования отрезков.

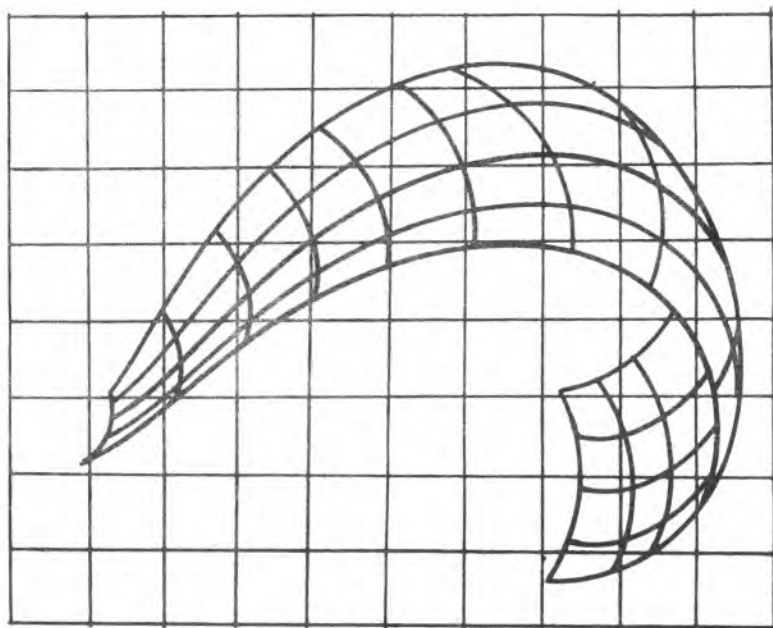
Сходство сканируемых строк означает, что изображения в соседних строках часто очень похожи друг на друга (но не всегда). Подобие строк можно использовать для предсказания положения пробных интервалов. Подобие строк можно использовать и при анализе видимости — видимость нужно определять только в тех случаях, когда при переходе к следующей сканирующей строке изменяется число отрезков на пробном интервале или же происходит пересечение двух отрезков (изменилась связь крайних точек отрезков с ребрами граней).

Метод сканирующей сетки Энкарнако. Этот алгоритм можно применять к произвольным криволинейным поверхностям, изображенным в виде сетки параметрических $U - 2^r$ линий. Название алгоритма связано с тем, что на проекцию поверхностей в картинной плоскости (получается обычно путем ортогонального проецирования) накладывающаяся прямоугольная (двумерная) сканирующая сетка (рис. 7.4).

Для каждой ячейки сетки составляется список кусков поверхности, проекции которых имеют общие точки с данной ячейкой, т.е. производится сортировка кусков поверхности по ячейкам сканирующей сетки. Здесь удобно использовать минимаксный тест: кусок поверхности заменяется огибающим прямоугольником и выявляются накрываемые им ячейки сканирующей сетки.

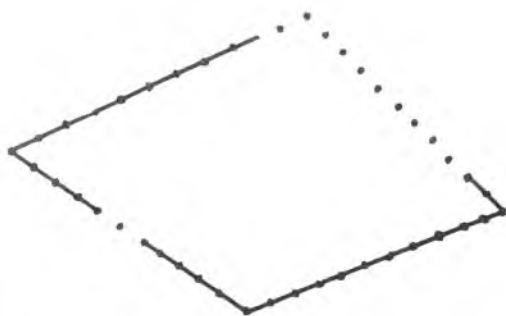
Следующий этап алгоритма — задание тестовых точек. Тестовые точки обычно равномерно распределяются по ребрам куска поверхности (как правило, ребра заменяются отрезками прямых). Каждая тестовая точка сопоставляется с кусками поверхности, лежащими в той же ячейке сетки. Если ни один кусок не накрывает точку, то она видима. Если некоторые точки невидимы, то рисуются только те части ребра, которые лежат между видимыми точками (рис. 7.5).

Каждый кусок криволинейной поверхности аппроксимируется четырьмя плоскостями. Для этого в куске поверхности проводятся диагонали,



Р и с. 7.4. Сканирующая сетка

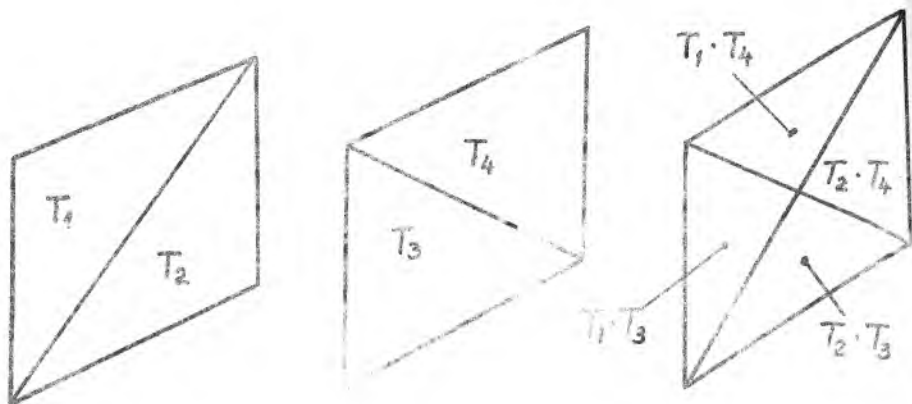
которые вместе с двумя ребрами определяют плоскость. Два варианта проведения диагоналей дают четыре различные плоскости (рис. 7.6). Нужны две проверки, чтобы определить, в каком из четырех треугольников лежит тестовая точка.



Р и с. 7.5. Распределение пробных точек на ребрах куска поверхности

Видимость тестовой точки определяется сравнением ее глубины с глубиной точки плоскости с теми же экранными координатами. Для надежности тестовая точка считается невидимой, если она закрывается двумя треугольниками ($T1 \cdot T3$, $T2 \cdot T3$ и т.д.). В противном случае она считается видимой

Важным вопросом является выбор величины

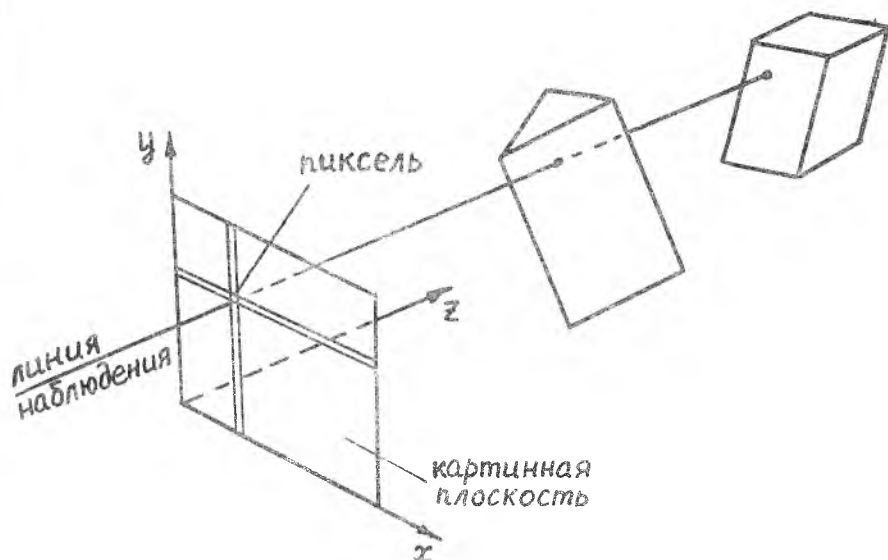


Р и с. 7.6. Аппроксимация куска поверхности четырьмя плоскостями

ячейки сканирующей сетки. Если сканирующая сетка имеет слишком крупные ячейки, то с каждой ячейкой будет связано много кусков, и выполнение тестов видимости займет много времени. Если используется частая сетка, то возрастает время начальной сортировки (формирования списка кусков для каждой ячейки) и объем необходимой памяти.

Для типичного изображения, имеющего несколько сотен кусков поверхности, пригодна сетка порядка сотни ячеек (10x10). В некоторых случаях число кусков поверхности неравномерно распределяется по ячейкам, тогда ячейка сетки, содержащая много кусков (более восьми), подразделяется на четыре подячейки. Эта процедура может быть повторена. Оптимальность деления ячеек сетки определяется, в основном, отношением времени выполнения минимальной процедуры и процедуры принадлежности.

Метод Z-буфера. Название метода связано с выделением в оперативной памяти специальной области — буфера для хранения дальностей от точки картинной плоскости до поверхности отображаемого объекта (рис. 7.7). В другом буфере, называемом буфером регенерации изображения, хранятся только значения яркостей для каждого элемента изображения (пикселя). Вначале в Z-буфер заносятся максимально возможные значения Z ("бесконечность"), а буфер регенерации заполняется значениями, описывающими фон.



Р и с. 7.7. Определение дальности до объектов от элементов картинной плоскости

Затем каждая проекция грани преобразуется в растровую форму (обычно это связано с алгоритмом закраски, см. п. 7.2), и полученные значения яркости или цвета для каждой точки области записываются в буфер регенерации. При этом грани одного или нескольких объектов обрабатываются поочередно.

Обработка заключается в следующем: сначала вычисляется дальность $z(x, y)$ от точки (x, y) картинной плоскости до рассматриваемой грани (она может быть криволинейной), а затем полученное значение $z(x, y)$ сравнивается со значением z -буфера, а яркость (или цвет) соответствующего элемента записывается в буфер регенерации. Таким образом, если точка обрабатываемой грани расположена ближе к наблюдателю, чем точка, значение яркости которой находилось до этого в z -буфере, то новые значения глубины и яркости запоминаются.

Если грань объекта плоская, то вычисление z -координат можно упростить. Для этого уравнение плоскости (носителя грани) решается относительно переменной z :

$$z = \frac{-D - Ax - By}{C}$$

где A, B, C, D — коэффициенты в уравнении плоскости. Тогда, учитывая, что обработка ведется по строкам растра, значения z в соседних элементах изображения с координатами (x, y) и $(x + \Delta x, y)$ равны:

$$z_0 = \frac{-D - Ax - By}{C}, \quad z_1 = \frac{-D - A(x + \Delta x) - By}{C}.$$

Выразив значение z_1 , через z_0 , получим

$$z_1 = \frac{-D - Ax - By - A\Delta x}{C} = z_0 - \frac{A}{C} \Delta x.$$

Отношение A/C для одной грани является постоянным, а $\Delta x = 1$. Поэтому, если задана (ранее вычислена) дальность в точке $(x+1, y)$ требуется выполнить лишь одно вычитание. Очередность появления объектов на экране определяется тем порядком, в котором обрабатываются грани объектов.

К достоинствам метода следует отнести очень простую реализацию, возможность обработки любых объектов, независимость времени анализа от сложности изображений сцены.

Недостатками алгоритма являются необходимость в большом объеме памяти под z -буфер (для растра 1024×1024 и 8 бит на каждый цвет, что сравнимо по качеству с телевизионным изображением, необходимо буфер объемом 3 мегабайт) и сравнительно большое время обработки (десятки минут на миниЭВМ типа СМ-4).

7.2. Формирование полутоновых изображений с тенями

Модели источников света и отражающих поверхностей. Полутоновое представление трехмерный объектов является одним из наиболее важных средств получения изображений в машинной графике. Для того, чтобы изображения выглядели реалистичными, необходимо учитывать различные свойства объектов такие, как отражение, поглощение, затенение и т.д. Реальные источники света не являются точечными, как они обычно рассматриваются в машинной графике, и число их может быть велико. Например, моделирование освещения в обычной комнате через окна представляет непростую задачу.

Наиболее простым для моделирования источником света является рассеянный естественный свет. Он обеспечивает постоянное освещение всех поверхностей независимо от их ориентации. Однако такой свет

дает весьма нечеткое изображение. Например, две грани куба при естественном освещении должны быть покрашены одинаково, так как имеют одинаковую яркость, и ребро между ними будет неразличимо.

Более сложным с точки зрения моделирования, но зато более реальным источником света является точечный источник, при использовании которого освещенность поверхности зависит от ее ориентации: чем меньше угол падения лучей, тем меньше освещенность. Наиболее ярко освещены поверхности, перпендикулярные лучам света.

Важной характеристикой объекта является отражательная способность поверхности. Большинство реальных объектов имеют матовые поверхности, обладающие свойством диффузного отражения, т.е. равномерного рассеивания света по всем направлениям. Поэтому кажется, что поверхности имеют одинаковую яркость независимо от угла обзора (при фиксированном положении источника света относительно объекта). Для таких поверхностей справедлив закон косинусов Ламберта

$$I = \rho \lambda \cos \theta = \rho \lambda (\bar{N} \bar{L}),$$

где \bar{N} - единичный нормальный вектор поверхности; \bar{L} - направление (ед.вектор) на точечный источник света; λ - интенсивность источника; ρ - коэффициент диффузного отражения (константа: $0 < \rho \leq 1$, зависит от материала):

В большинстве реальных сцен присутствует рассеянный свет, т.е. свет постоянной яркости, из-за отражений от различных поверхностей ("световой фон"). Даже если объект закрыт от прямых лучей точечного источника света, он все равно будет виден из-за наличия рассеянного света. В таких случаях обычно используют следующую модель источника света

$$I = \lambda_a \rho_a + \rho \lambda (\bar{N} \bar{L}),$$

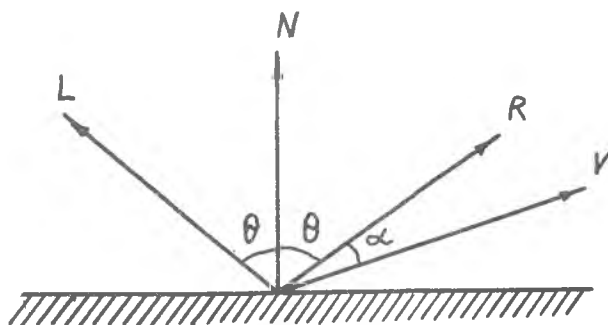
где λ_a - интенсивность рассеянного света; ρ_a - количество рассеянного света, которое отражается от поверхностей предмета.

Более реалистично выглядят объекты, если использовать модель источника света

$$I = \rho_a \lambda_a + \rho \lambda \frac{\bar{N} \bar{L}}{z + k},$$

где k - константа; z - расстояние от точки наблюдения до поверхности объекта.

Если поверхность объекта не является матовой, то при ее освещении от точечного источника на ней образуется световой блик из-за зеркального отражения. Если изменить положение головы, то можно заметить, что световой блик смещается. Это объясняется тем, что блестящие поверхности отражают свет неодинаково по всем направлениям. От тщательно отполированной поверхности ("зеркальной") свет отражается только в том направлении, для которого углы падения и отражения совпадают. Это означает, что наблюдатель может увидеть зеркально отраженный свет только в том случае, если угол α равен нулю (рис. 7.8).



Р и с. 7.8. Падающий и отраженный лучи, нормаль и линия наблюдения

Для неидеально отражающих поверхностей, (например яблоко) интенсивность отражения света резко падает с ростом α . В модели, предложенной Фонгом, убывание интенсивности описывается функцией $\cos^n \alpha$, где $n = 1 \dots 200$ в зависимости от вида поверхности, т.е.

$$I = \rho_a \lambda_a + \frac{\lambda}{2+k} [\rho \cos \theta + W(\theta) \cos^n \alpha] =$$

$$= \rho_a \lambda_a + \frac{\lambda}{2+k} [\rho (\bar{N} \cdot \bar{l}) + \rho_g (\bar{R} \cdot \bar{V})^n],$$

где $W(\theta)$ - доля зеркального отраженного света, причем обычно принимают $W(\theta) = \rho_s = \text{const}$; \bar{R} - направление (ед. вектора) отраженного света); \bar{V} - направление на наблюдателя (отсчитывается от основания нормали).

Общее описание алгоритма формирования полутонных изображений

Будем считать, что трехмерные объекты являются выпуклыми многогранниками. Каждая грань определяется цепочкой вершин, упорядоченных по часовой стрелке, если смотреть извне многогранника. Нормаль грани в этом случае направлена во внешнюю от многогранника область.

Введем функцию F , устанавливающую связь между гранью и точкой, и функцию H , устанавливающую связь между гранью и объектом (или другой гранью):

$$F_S(Q) = (P_3 - P_1) \times (P_2 - P_1) \cdot (Q - P_1) = a_S x + b_S y + c_S z + d_S,$$

$$H(S, V) = \begin{cases} 1, & \text{при } F_S(P_K) > 0, P_K \in V, \\ -1, & \text{при } F_S(P_K) < 0, P_K \in V, \\ 0 & \text{в остальных случаях} \end{cases}$$

где S - грань, включающая вершины P_i , $i = 1, 2, 3, \dots$;

$Q(x, y, z)$ - произвольная точка; (a_S, b_S, c_S) - нормаль грани S ;

V - многогранник, включающий вершины P_K , $K = 1, 2, \dots, n$.

Таким образом, точка Q лежит в положительном полупространстве грани S (т.е. той части пространства, куда направлена нормаль этой грани), если $F_S(Q) > 0$. Если $F_S(Q) = 0$, то точка лежит на плоскости, включающей грань S . Далее, при $H(S, V) = 1$ многогранник V целиком расположен в положительном полупространстве грани S . При $H(S, V) = 0$ многогранник пересекается плоскостью, включающей грань S .

Источник света может быть двух типов: плоский (распределенный) в виде выпуклого многоугольника с m вершинами и объемный (распределенный), являющийся выпуклым многогранником, все грани которого излучают свет. Плоский источник излучает свет только в положительном полупространстве (т.е. по направлению нормали).

Для того чтобы упростить дальнейшую обработку, определим три типа граней в зависимости от их положения относительно источника света:

$$\text{Тип А } H(S_f, S_e) = 1, H(S_e, S_f) \neq -1 \\ \text{или } H(S_f, V_e) = 1$$

$$\text{Тип Б } H(S_f, S_e) = 0, H(S_e, S_f) \neq -1 \\ \text{или } H(S_f, V_e) = 0.$$

Тип В $H(S_f, S_e) = -1$ или $H(S_e, S_f) = -1$

или

$$H(S_f, V_e) = -1,$$

где S_e - плоский источник, V_e - объемный источник, S_f - грань.

Таким образом, грани типа А освещаются всеми источниками полностью, грани типа Б освещаются частью источника, грани типа В не освещаются, поэтому при вычислениях освещенности их можно не учитывать.

Алгоритм формирования полутоновых изображений включает следующие этапы /17/:

1. Ввод исходных данных об объектах, точках наблюдения, направлениях наблюдения, световых источниках.
2. Проецирование вершин объектов на картинную плоскость и определение приоритетов при перекрытиях проекций (см. п. "Метод приоритетов Энкарнако").
3. Классификация граней согласно уравнениям (7.1), выделение частично освещенных или полностью затененных граней.
4. Определение объемов для каждого многогранника.
5. Выделение многогранников с тенями на видимых гранях, вычисление границ теней.
6. Сканирование плоскости изображения, анализ видимости поверхностей и определение освещенности, теней в пределах каждой сканирующей прямой.

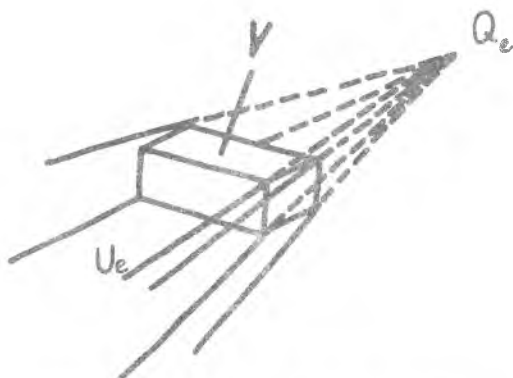
При изменении точек или направлений наблюдения, этапы 2, 3, 4, 5, 6 должны быть выполнены заново. При движении источников света пересчитываются шаги 3, 4, 5, 6. Рассмотрим их подробнее.

Теневые объемы для плоских и объемных источников света. Теневой объем V_e , образуемый одной из вершин Q_e источника света и выпуклым многогранником V , является незамкнутым выпуклым объемом, ограниченным плоскостями двух типов (рис. 7.9):

плоскости, проходящие через вершину Q_e и соседние вершины $P_{e,i}$ и $P_{e,i+1}$ граней, видимых из Q_e (т.е. граней, нормали которых составляют угол меньше $\pi/2$ с лучом из Q_e в точку основания нормали);

грани, видимые из Q_e .

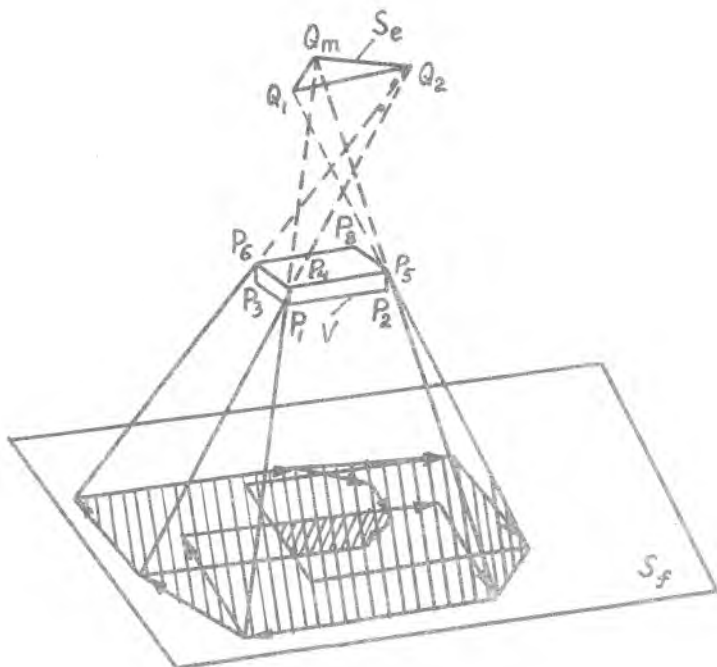
Для плоского источника S_e с m вершинами Q_1, Q_2, \dots, Q_m затененная часть пространства разбивается на область полной тени, теневой объем которой образуется как пересече-



Р и с. 7.9. Теневой объем, образуемый точечными источниками света и выпуклыми многогранниками

и и е всех теневых объемов U_1, U_2, \dots, U_m , соответствующих вершинам источника света, и область полутени, теневой объем которой образуется как объединение всех теневых объемов U_1, U_2, \dots, U_m . Пересечение этих теневых объемов с картинной плоскостью дает границы изображений (и сами изображения) тени и полутени от многогранника V (рис.

7.10). В качестве картинной плоскости может выступать также грань другого объекта.



Р и с. 7.10. Определение границ тени и полутени от многогранника для плоского распределения источника света

Для объемного источника света V_e области тени и полутени определяются аналогично. Когда многогранник V находится в отрицательном полупространстве источника S_e , т.е. $H(S_e, V) = -1$, то V не освещен, и объемы тени и полутени не определены. В противном случае можно предложить такую последовательность обработки.

Пусть \hat{S} является плоскостью, определяемой вершиной Q_e источника света и ребром V , \bar{S} — плоскостью, формируемой вершиной V и ребром источника света, и \dot{S} — одна из граней V . В табл. 7.1 перечислены условия для областей тени и полутени ("I" обозначает плоский S_e или объемный V_e источник света).

Т а б л и ц а 7.1

Типы плоскостей	Теневые объемы	
	Тень	Полутень
\hat{S}	$H(\hat{S}, V) = -1$ $H(\hat{S}, I) = -1$	$H(\hat{S}, V) = -1$ $H(\hat{S}, I) = 1$
\bar{S}		$H(\bar{S}, V) = -1$ $H(\bar{S}, I) = 1$
\dot{S}	$H(\dot{S}, I) \neq -1$	$H(\dot{S}, I) = 1$

Случай, когда $H(S_e, V) = 0$ соответствует пересечению плоскости содержащей S_e и V , поэтому только часть объекта V освещена. Области тени и полутени составляют только те части объектов из табл. 7.1, которые находятся в положительном пространстве S_e .

Для определения области (объема) полутени сначала находится контур многоугольника V для каждой вершины Q_e источника света S_e или V_e . Контур многогранника и вершина Q_e образуют некоторую пирамиду. Рассматривая поочередно боковые грани этой пирамиды, выделяются те из них, которые удовлетворяют условию $H(\hat{S}, S_e) = 1$ (или $H(\hat{S}, V_e) = 1$). Эти грани и определяют объем полутени. Для нахождения объема тени боковые грани пирамиды долж-

ны удовлетворяют условию $H(\hat{S}, S_e) = -1$ (или $H(\hat{S}, V_e) = -1$).

Границы теней на гранях объектов. Границы теней на гранях трехмерных объектов находятся пересечением граней с объемами тени или полутени. На первом этапе ищется пересечение с объемом полутени, так как область тени всегда находится внутри области полутени. Если пересечение обнаружено, то на грани имеется по меньшей мере одна полутень и проверяется наличие тени (путем пересечения грани и объема тени). Если \tilde{U} — объем полутени, образуемый от источника света S_e (или V_e) и выпуклого многогранника V , а \tilde{S}_p , $p = 1, 2, \dots$, — грани объема \tilde{U} , то условия существования полутени на грани S_f объекта могут быть записаны в виде

$$H(S_f, V) \neq -1. \quad (7.2)$$

$$H(\tilde{S}_p, S_f) \neq 1 \text{ для всех } \tilde{S}_p \in \tilde{U}. \quad (7.3)$$

Уравнение (7.2) означает, что тень не может быть создана многогранником, находящимся в отрицательном полупространстве грани S_f .

Очевидно, что для плоских источников при $H(S_e, V) = -1$ тени также не возникает, так как по определению такие источники излучают свет только в положительном полупространстве.

Если условия (7.2) и (7.3) выполняются, границы полутени легко могут быть вычислены пересечением граней S_f и \tilde{S}_p .

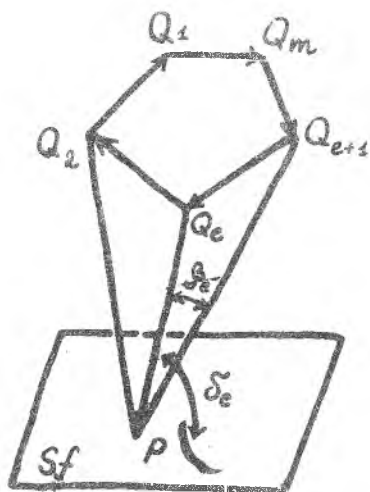
Вычисление освещенности. Освещенность вычисляется для точек, не затененных ни одним из объектов, а также для точек областей, попадающих в полутень одного или нескольких объектов. Для сокращения вычислительных затрат освещенность вычисляется на интервалах сканирующей линии, в пределах которых интенсивность каждого пикселя изменяется по линейному закону. Границы теней и граней определяют концы этих интервалов, где освещенность объектов и, следовательно, интенсивность элементов изображения обычно изменяется скачкообразно.

Поскольку предполагается, что световые источники образованы светящимися поверхностями с равными яркостями в каждой точке, то для вычисления освещенности можно использовать метод интегрирования по контуру. Плоский источник с m вершинами, интенсивность каждой из которых равна L , создает освещенность в точке P грани S_f , определяемой уравнением

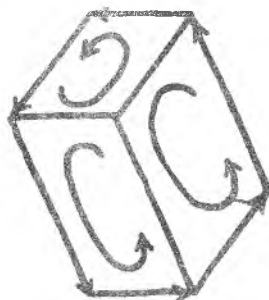
$$E = \frac{L}{2} \sum_{e=1}^m \beta_e \cos \delta_e, \quad (7.4)$$

где β_e - угол между PQ_e и PQ_{e+1} ; δ_e - угол между гранью S_f и треугольником P, Q_e, Q_{e+1} (рис. 7.11).

В случае, когда грань S_f принадлежит к типу Б, рассматривается только часть грани, находящейся с положительной стороны источника.



Р и с. 7.11. Вычисление освещенности в точке грани от плоского источника света



Р и с. 7.12. Вычисление освещенности от объемного источника света

Для объемных источников V_e суммируется излучение от каждой грани V_e , которая видна из точки P . Освещенность от каждой из этих граней V_e определяется по уравнению (7.4). На рис. 7.12 показан контур объемного источника, наблюдаемого из точки P . Как видно из рисунка, интегрирование вдоль ребра двух видимых граней выполняется в противоположных направлениях, поэтому освещенность может быть вычислена методом интегрирования вдоль контура (контур определен по часовой стрелке, однако интегрирование выполняют против часовой стрелки, чтобы получить положительное значение).

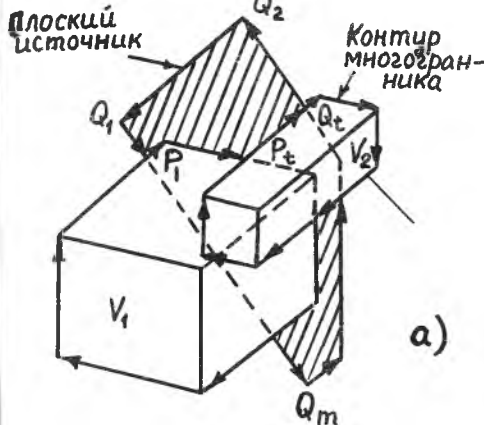
Рассмотрим теперь вычисление освещенности для затененных областей.

Области полной тени определяются довольно просто. Если рассматриваемая точка попадает в полную тень хотя бы одного объекта сцены, то освещенность P равна нулю.

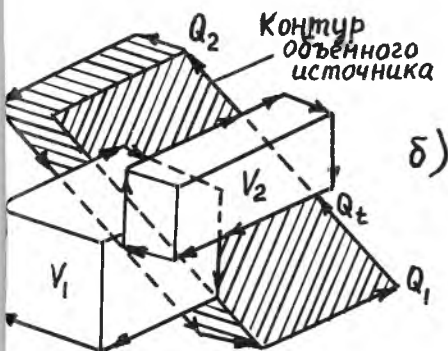
Вычисление полутеней более сложно. Поскольку область полутени образована лучами света, частично перекрываемыми некоторыми объектами сцены, то освещенность в точке P вычерчивается по уравнению (7.4) для части источника света, видимой из точки P . Для этого следует:

найти многогранник, отображающий полутень на точку P ;
найти видимый из точки P контур этого многогранника;

проинтегрировать освещенность вдоль видимых отрезков контура источника (например, $Q_1 Q_2$, $Q_2 Q_1$ и т.д. на рис. 7.13, а) в направлении против часовой стрелки;



а)



б)

Р и с. 7.13. Вычисление освещенности в области полутени

проинтегрировать освещенность вдоль контура другого многогранника, попадающего внутрь контура источника (например, $P_1 P_t$ на рис. 7.13, б) в направлении часовой стрелки.

Освещенность точки P получается суммированием значений, полученных на шаге 3 (вдоль видимых отрезков контура источника), и выполняется следующим образом. Находится пирамида (открытое выпуклое пространство), образованная точкой P и контуром выпуклого многогранника. Видимые отрезки контура источника получают пере-

сечением этой пирамиды с контуром источника. Такую работу следует выполнить для каждого многогранника, выявленного на шаге 1.

Для интегрирования на шаге 4 каждое ребро контура отсекается пирамидой, образованной точкой P и контуром источника. Внешние сегменты пирамиды являются отсекающими. Тогда отрезки для интегрирования находят так же, как на этапе 3.

Закраска областей. Область — это множество связанных элементов изображения (пикселей) в кадре. Область обладает некоторым цветом, который либо указывается пользователем непосредственно, либо вычисляется специальным алгоритмом. Задача изменения цвета некоторой области и называется закраской. Алгоритм закраски не использует никаких данных о том, что представляет собой данная область. Входными данными являются только серый цвет, желаемый новый цвет и одна точка, принадлежащая области. Алгоритм затем просматривает элементы области, изменяя их старый цвет на новый.

Цвет области формируется как комбинация некоторых базисных цветов. Алгоритм закраски состоит из четырех частей:

метод поиска границ области;

стартовая процедура, инициализирующая закраску;

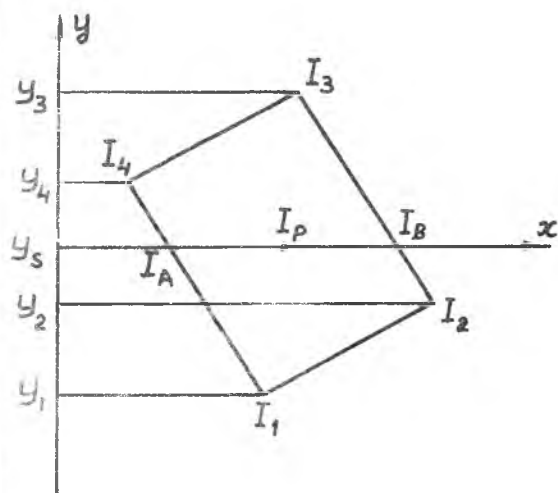
тест принадлежности, определяющий, находится ли некоторый пиксель внутри области;

процедура изменения цвета данного пикселя.

Существует несколько способов закраски областей: однотонная закраска; закраска, основанная на интерполяции векторов нормали.

При однотонной закраске вычисляется одно значение яркости (интенсивности), которое используется для закрашивания всего многоугольника. При этом предполагается, что источник света расположен в бесконечности ($\vec{NL} = const$ в пределах грани) и наблюдатель расположен в бесконечности ($\vec{NV} = const$ в пределах грани). Если эти предположения не выполняются, то можно воспользоваться усредненными значениями \vec{L} и \vec{V} , вычисленными, например, в центре многоугольника.

Закраска, основанная на интерполяции интенсивностей (метод Гуро), позволяет устранить дискретность изменения интенсивности и сделать изображение более реалистичным. Процесс закраски по методу Гуро осуществляется в 4 этапа. На первом этапе определяются нормали к поверхностям. На втором определяются



Р и с. 7.14. Интерполяция интенсивности вдоль сканирующей строки

сивностей в вершинах сначала вдоль каждого ребра, а затем между ребрами вдоль каждой сканирующей строки (рис. 7.14).

нормали в вершинах путем усреднения нормалей по всем граням, к которым принадлежит вершина.

На третьем этапе вычисляются значения интенсивностей в вершинах. При этом используется одна из моделей источников света (см. п. "Модели источников света..."). На четвертом этапе каждый многоугольник закрашивается путем линейной интерполяции значений интен-

О Г Л А В Л Е Н И Е

1.	Общие сведения о геометрическом моделировании.....	6
1.1.	Основные определения и допущения.....	6
1.2.	Условия правильного образования геометрических форм.....	10
2.	Описание объектов с помощью функции принадлежности....	13
2.1.	Рецепторная модель.....	13
2.2.	Геометрические модели с древовидными структурами данных.....	14
3.	Теоретико-графовое описание объектов.....	20
3.1.	Определения и допущения.....	20
3.2.	Структура данных для хранения описания объекта....	23
3.3.	Поисковые операции в структуре данных.....	25
3.4.	Операторы Эйлера.....	29
4.	Аналитические геометрические модели.....	30
4.1.	Определения и допущения.....	30
4.2.	B -сплайны.....	31
4.3.	Параметрические бикубические поверхности.....	37
5.	Описание объектов с помощью аналогов и операций с ними.....	42
5.1.	Параметрические модели.....	42
5.2.	Геометрические преобразования образцов.....	43
5.3.	Теоретико-множественные операции.....	44
5.4.	Преобразование формы объекта путем скругления его граней и вершин.....	46
6.	Обобщенные цилиндры.....	60
6.1.	Определения и допущения.....	60
6.2.	Операции с обобщенными цилиндрами.....	64
7.	Визуализация результатов геометрического моделирования.	68
7.1.	Анализ видимости.....	68
7.2.	Формирование полутоновых изображений с тенями....	80

Св.план 1988, поз. 147

О л е г П е т р о в и ч А л е к с а н д р о в

ГЕОМЕТРИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Редактор Е.Д.А н т о н о в а
 Техн.редактор Н.М.К а л е н ю к
 Корректор О.Ю.Н е н а ш е в а

ЕО 00301. Подписано в печать 14.10.88 г.
 Формат 60x84 1/16. Бумага оберточная белая.
 Оперативная печать. Усл.п.л. 4,5. Уч.-изд.л.4,0.
 Т. 1000 экз. Заказ № 5313. Цена 20 к.
 Куйбышевский ордена Трудового Красного Знамени
 авиационный институт имени академика С.П.Королева,
 г.Куйбышев, ул.Молодогвардейская, 151.
 Тип.им.В.П.Мяги Куйбышевского полиграфического
 объединения, 443099, г. Куйбышев, ул. Венцека, 60.