

лапок и гребешка. Хордовая аппроксимация устранила петли, заметные на рис. 5-11с на лапках, и дает более гладкое изображение гребешка. Несмотря на эти недостатки, нормализованное представление применяется для создания кубических ограничивающих кривых для кусков трехмерной поверхности (см. гл. 6).

Еще одно интересное замечание: если для нормализованного сплайна известна матрица $[B]^T = [B_1 \ B_2 \ B_3 \ B_4]$, то геометрическая матрица сегмента сплайна $[G]$ равна

$$[G] = [N]^{-1} [B]^T,$$

где $[N]^{-1}$ — обратная матрица $[N]$:

$$[N]^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}. \quad (5-29)$$

Точки на сегменте рассчитываются из уравнения (5-27).

5-5 ДРУГИЕ ГРАНИЧНЫЕ УСЛОВИЯ

В предыдущих рассуждениях предполагалось, что известны касательные векторы P'_1 и P'_n в концевых точках кусочного кубического сплайна. Такое граничное условие называется *закрепленным*. Неизвестные касательные векторы в промежуточных точках вычисляются инверсией трехдиагональной матрицы $[M]$ из уравнения (5-15).

Если задано мало точек или физические ограничения требуют определенной формы кривой у концов, возможны другие граничные условия. Например, можно задать кривизну в концевых точках. Нуловая кривизна называется *слабым граничным условием*. Аппроксимируя кривизну второй производной и вспоминая уравнение (5-9), получаем

$$P''(t) = \sum_{i=1}^4 (i-1)(i-2)B_i t^{i-3}, \quad 0 \leq t \leq t_{k+1}.$$

В начале первого сегмента сплайна $t = 0$. Отметим, что на результат влияет только член $i = 3$. Из уравнения (5-17)

$$P''(0) = 2B_3 = 2 \left(\frac{3(P_2 - P_1)}{t_2^2} - \frac{2P'_1}{t_2} - \frac{P'_2}{t_2} \right) = 0.$$

После перегруппировки получаем

$$P'_1 + \frac{P'_2}{2} = \frac{3}{2t_2}(P_2 - P_1). \quad (5-30)$$

Теперь первая строка в матрицах $[M]$ и $[R]$ (см. (5-15)) такова:

$$[1 \ 1/2 \ 0 \ \dots] [P'_1] = \left[\frac{3}{2t_2}(P_2 - P_1) \right].$$

В конце последнего сегмента $t = t_n$, $k = n - 1$. Здесь результат зависит от двух членов $i = 3$ и $i = 4$ в (5-9), а именно

$$P''(t_n) = 2B_3 + 6B_4 t_n = 0$$

или, используя уравнение (5-17),

$$\frac{6}{t_n^2}(P_{n-1} - P_n) + \frac{2P'_{n-1}}{t_n} + \frac{4P'_n}{t_n} = 0.$$

После перегруппировки

$$2P'_{n-1} + 4P'_n = \frac{6}{t_n}(P_n - P_{n-1}). \quad (5-31)$$

Последняя строка матриц $[M]$ и $[R]$ (см. (5-15)) выглядит теперь так:

$$[\dots \ 0 \ 2 \ 4] [P'_n] = \left[\frac{6}{t_n}(P_n - P_{n-1}) \right].$$

Заметим, что для сплайна слабые и закрепленные граничные условия могут меняться местами.

Пример 5-4 Кубический сплайн со слабым граничным условием

Пусть заданы три точки $P_1[0 \ 0]$, $P_2[1 \ 2]$ и $P_3[3 \ 2]$. Найти кубический сплайн, проходящий через них, со слабыми граничными условиями. Для t_k использовать хордовую аппроксимацию.

Коэффициенты t_k таковы:

$$t_2 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \sqrt{(1)^2 + (2)^2} = \sqrt{5}, \\ t_3 = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} = \sqrt{(3 - 1)^2 + (0)^2} = 2.$$

Модифицируя матрицы $[M]$ и $[R]$ с помощью уравнений (5-30) и (5-31), получим уравнение внутреннего касательного вектора в P'_2 :

$$\begin{bmatrix} 1 & 0.5 & 0 \\ 2 & 8.472 & 2.236 \\ 0 & 2 & 4 \end{bmatrix} \begin{bmatrix} P'_1 \\ P'_2 \\ P'_3 \end{bmatrix} = \begin{bmatrix} \frac{3}{2\sqrt{5}}([1 \ 2] - [0 \ 0]) \\ \frac{3}{2\sqrt{5}}\{5([3 \ 2] - [1 \ 2]) + 4([1 \ 2] - [0 \ 0])\} \\ 3([3 \ 2] - [1 \ 2]) \end{bmatrix} \\ = \begin{bmatrix} 0.671 & 1.342 \\ 0.391 & 5.367 \\ 6 & 0 \end{bmatrix}.$$

Инвертируя (3×3) -матрицу и умножая, найдем производные

$$\begin{bmatrix} P'_1 \\ P'_2 \\ P'_3 \end{bmatrix} = \begin{bmatrix} 1.157 & -0.079 & 0.044 \\ -0.315 & 0.157 & -0.088 \\ 0.157 & -0.079 & 0.294 \end{bmatrix} \begin{bmatrix} 0.671 & 1.342 \\ 9.391 & 5.367 \\ 6 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0.301 & 1.131 \\ 0.739 & 0.422 \\ 1.131 & -0.211 \end{bmatrix}.$$

Из уравнения (5-21) весовые функции при $\tau = 1/3$ и $2/3$ для первого сегмента имеют вид:

$$[F]_{\tau=1/3} = [0.741 \quad 0.259 \quad 0.331 \quad -0.166],$$

$$[F]_{\tau=2/3} = [0.259 \quad 0.741 \quad 0.166 \quad -0.331].$$

Затем уравнение (5-22) дает точки на первом сегменте, т. е.

$$P(1/3) = [0.741 \quad 0.259 \quad 0.331 \quad -0.166] \begin{bmatrix} 0 & 0 \\ 1 & 2 \\ 0.301 & 1.131 \\ 0.739 & 0.422 \end{bmatrix} =$$

$$= [0.237 \quad 0.823].$$

Аналогично

$$P(2/3) = [0.546 \quad 1.529].$$

Весовые функции при $\tau = 1/3, 2/3$ для второго сегмента таковы:

$$[F]_{\tau=1/3} = [0.741 \quad 0.259 \quad 0.296 \quad -0.148],$$

$$[F]_{\tau=2/3} = [0.259 \quad 0.741 \quad 0.148 \quad -0.296]$$

и уравнение (5-22) дает

$$P(1/3) = [1.570 \quad 2.156],$$

$$P(2/3) = [2.256 \quad 2.125].$$

Результаты показаны на рис. 5-12.

Интерес представляют еще два типа граничных условий: циклическое и ациклическое. Циклическое условие порождает замкнутую кривую или часть периодически повторяющейся кривой. Для этого необходимо, чтобы касательная вектора и кривизна на обоих концах были нулевыми:

$$P'_1(0) = P'_n(t_n), \quad (5-32)$$

$$P''_1(0) = P''_n(t_n). \quad (5-33)$$

Из уравнений (5-3) и (5-9), пользуясь уравнениями (5-17) и (5-32), получаем

$$P'_1 - P'_{n-1} = 2t_n \left[\frac{3(P_n - P_{n-1})}{t_n^2} - \frac{2P'_{n-1}}{t_n} - \frac{P'_n}{t_n} \right] +$$

$$+ 3t_n \left[\frac{2(P_{n-1} - P_n)}{t_n^3} + \frac{P'_{n-1}}{t_n^2} + \frac{P'_n}{t_n^2} \right]. \quad (5-34)$$

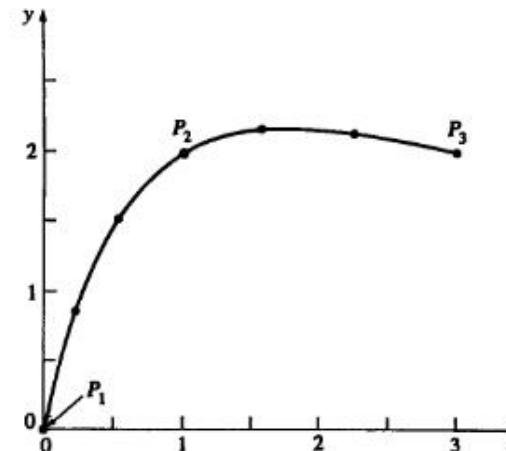


Рис. 5-12 Кубический сплайн для примера 5-4.

Аналогично, уравнение (5-33) приводится к виду

$$2 \left[\frac{3(P_2 - P_1)}{t_2^2} - \frac{2P'_1}{t_2} - \frac{P'_2}{t_2} \right] = 2 \left[\frac{3(P_n - P_{n-1})}{t_n^2} - \frac{2P'_{n-1}}{t_n} - \frac{P'_n}{t_n} \right] +$$

$$+ 6t_n \left[\frac{2(P_{n-1} - P_n)}{t_n^3} + \frac{P'_{n-1}}{t_n^2} + \frac{P'_n}{t_n^2} \right]. \quad (5-35)$$

Умножив (5-35) на t_n и вычитая его из (5-34), получим

$$P'_1 - P'_{n-1} - 2t_n \left[\frac{3(P_2 - P_1)}{t_2^2} - \frac{2P'_1}{t_2} - \frac{P'_2}{t_2} \right] =$$

$$= 3t_n^2 \left[\frac{2(P_{n-1} - P_n)}{t_n^3} + \frac{P'_{n-1}}{t_n^2} + \frac{P'_n}{t_n^2} \right] -$$

$$- 6t_n^2 \left[\frac{2(P_{n-1} - P_n)}{t_n^3} + \frac{P'_{n-1}}{t_n^2} + \frac{P'_n}{t_n^2} \right]$$

Вспомним, что $P'_1 = P'_n$, и перегруппируем члены:

$$2 \left(1 + \frac{t_n}{t_2} \right) P'_1 + \frac{t_n}{t_2} P'_2 + P'_{n-1} = 3 \frac{t_n}{t_2^2} (P_2 - P_1) - \frac{3}{t_n} (P_{n-1} - P_n). \quad (5-36)$$

Касательные векторы в точках внутреннего соединения опять получаются с помощью уравнения (5-15). Однако из-за того, что касательные векторы линейно зависимы ($P'_1 = P'_n$), матрица $[M]$ теперь имеет размер $(n-1) \times (n-1)$, где

первая строка состоит из коэффициентов уравнения (5-36):

$$\begin{bmatrix} 2\left(1 + \frac{t_n}{t_2}\right) & \frac{t_n}{t_2} & 0 & 0 & \cdots & 1 \\ t_3 & 2(t_2 + t_3) & t_2 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & t_n & 2(t_n + t_{n-1}) & t_{n-1} \end{bmatrix} \begin{bmatrix} P'_1 \\ P'_2 \\ P'_3 \\ \vdots \\ P'_{n-1} \end{bmatrix} = \begin{bmatrix} \frac{3}{t_2^2}(P_2 - P_1) - \frac{3}{t_n}(P_{n-1} - P_n) \\ \frac{3}{t_2^2}\{t_2^2(P_3 - P_2) + t_3^2(P_2 - P_1)\} \\ \frac{3}{t_3 t_4}\{t_3^2(P_4 - P_3) + t_4^2(P_3 - P_2)\} \\ \vdots \\ \frac{3}{t_{n-1} t_n}\{t_{n-1}^2(P_n - P_{n-1}) + t_n^2(P_{n-1} - P_{n-2})\} \end{bmatrix}. \quad (5-37)$$

Эта матрица уже не трехдиагональная. Ациклический сплайн похож на циклический за исключением того, что

$$P'_1(0) = -P'_n(t_n), \quad (5-38)$$

$$P''_1(0) = -P''_n(t_n). \quad (5-39)$$

Та же процедура, что и для циклических граничных условий, дает

$$2\left(1 + \frac{t_n}{t_2}\right)P'_1 + \frac{t_n}{t_2}P'_2 - P'_{n-1} = 3\frac{t_n}{t_2^2}(P_2 - P_1) + \frac{3}{t_n}(P_{n-1} - P_n). \quad (5-40)$$

Из уравнения (5-40) видно, что единственное отличие — это разные знаки у 1 в $M(1, n-1)$ в матрице $[M]$ и у второго члена $R(1, 1)$ в уравнении (5-37). Ациклические сплайны полезны при изображении кривых, у которых касательные векторы в концах имеют одинаковую величину и противоположные направления, например как у многослойной деревянной теннисной ракетки. В табл. 5-3 собраны граничные условия для кубических сплайнов.

У кубических сплайнов первые и вторые производные непрерывны для любых граничных условий, но с увеличением количества заданных точек время обращения матрицы касательных векторов может стать слишком большим.

На рис. 5-13 изображены два кубических сплайна, проходящих через пять точек: с закрепленными и слабыми граничными условиями. Касательные векторы в закрепленных концах — $[1 \ 1]$ и $[1 \ 1]$, соответственно. В этом случае различие кривых несущественно. Другие касательные векторы, например $[1 \ 1]$ и $[1 \ 1]$, могут существенно изменить форму сплайна с закрепленными концами.

На рис. 5-14 показано влияние изменения величины, а не направления касательных векторов в концах замкнутого кубического сплайна. Эти кривые симметричны, так как квадратная матрица в уравнении (5-15) трехдиагональна.

Таблица 5-3 Граничные условия для кубических сплайнов

Граничное условие	Ненулевые элементы в первой и последней строках $[M]$	Первая и последняя строки $[R]$
Закрепленное	$M(1, 1) = 1$ $M(n, n) = 1$	$R(1, 1) = P'_1$ $R(n, 1) = P'_n$
Слабое	$M(1, 1) = 1$ $M(1, 2) = 1/2$ $M(n, n-1) = 2$ $M(n, n) = 4$	$R(1, 1) = \frac{3}{2t_2}(P_2 - P_1)$ $R(n, 1) = \frac{6}{t_n}(P_n - P_{n-1})$
Циклическое	$M(1, 1) = 2\left(1 + \frac{t_n}{t_2}\right)$ $M(1, 2) = \frac{t_n}{t_2}$ $M(1, n-1) = 1$	$R(1, 1) = 3\left(\frac{t_n}{t_2^2}\right)(P_2 - P_1)$ $-\frac{3}{t_n}(P_{n-1} - P_n)$ $R(n, 1)$ не определено
Ациклическое	$M(1, 1) = 2\left(1 + \frac{t_n}{t_2}\right)$ $M(1, 2) = \frac{t_n}{t_2}$ $M(1, n-1) = -1$	$R(1, 1) = 3\left(\frac{t_n}{t_2^2}\right)(P_2 - P_1)$ $+\frac{3}{t_n}(P_{n-1} - P_n)$ $R(n, 1)$ не определено

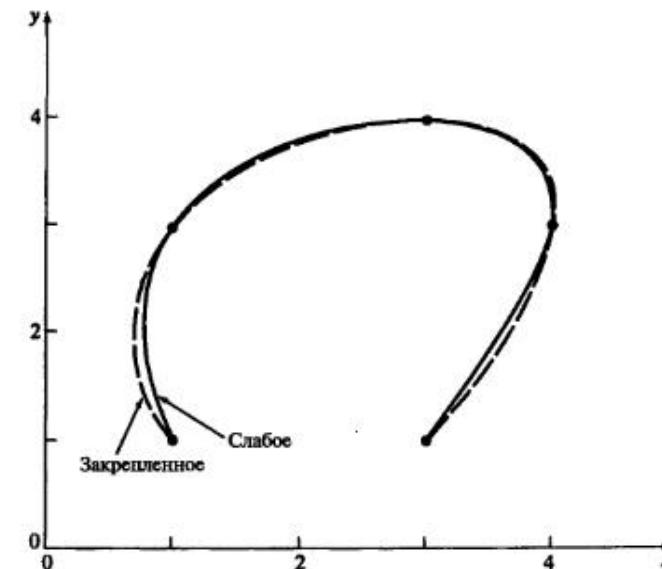


Рис. 5-13 Сравнение закрепленного и слабого граничных условий для нормализованного кусочно кубического сплайна.

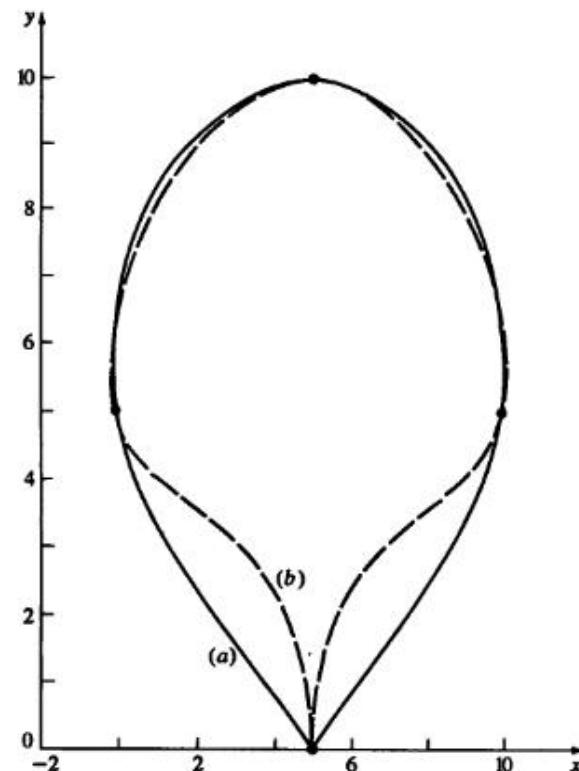


Рис. 5-14 Влияние величины касательного вектора на форму нормализованного кусочно-кубического сплайна. (а) $P_1'[0 \quad 1], P_2'[0 \quad -1]$; (б) $P_1'[0 \quad 10], P_2'[0 \quad -10]$.

На рис. 5-15 через те же пять точек проходит сплайн со слабыми и циклическими граничными условиями. Обратите внимание на то, что сплайн со слабыми условиями симметричен, а циклический — нет.

На рис. 5-16 сравниваются слабое и циклическое граничные условия. Циклическое условие можно использовать и для незамкнутых кривых: направление начального касательного вектора должно совпадать с направлением касательного вектора в конце.

Несмотря на то, что параметрические кусочные сплайны удобны и применяются в ряде отраслей (производстве автомобилей, судостроении и авиастроении), у них есть некоторые недостатки. Так, они неточно представляют конические сечения и асимптотические кривые и часто приводят к видимой осцилляции. Осцилляция возникает из-за того, что сплайн испытывает локальное влияние каждой точки, а третья производная только кусочно постоянна. Разрывы третьей производной порождают отклонения, и полученная кривая не является гладкой, несмотря на то, что ее вторая производная непрерывна.

Один из методов борьбы с осцилляциями — математический аналог приложения силы к концам сплайна. Пусть имеется физический сплайн — тонкая, гибкая

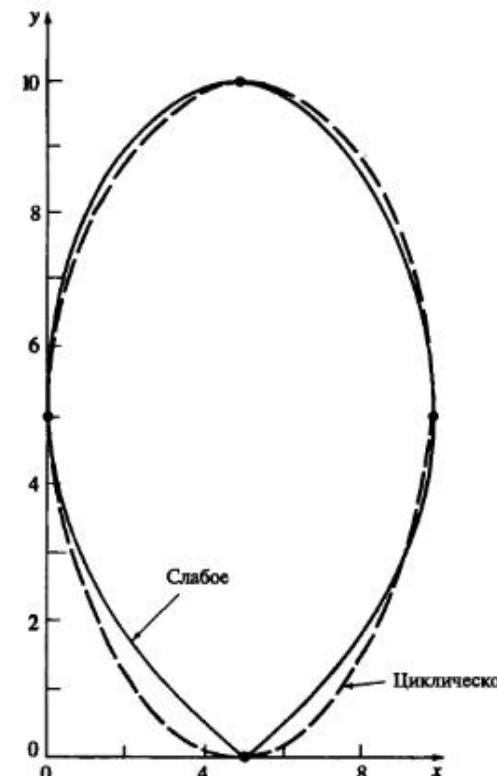


Рис. 5-15 Сравнение слабого и циклического граничных условий для нормализованного кусочно-кубического сплайна.

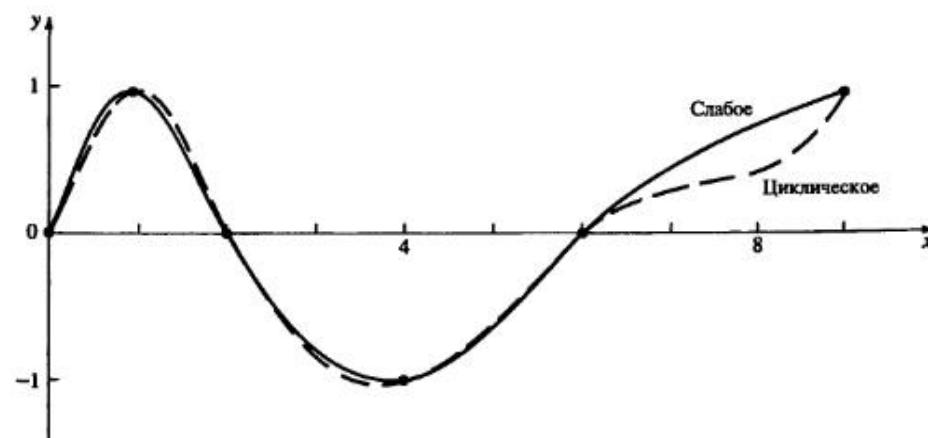


Рис. 5-16 Сравнение слабого и циклического граничных условий для открытых кривых.

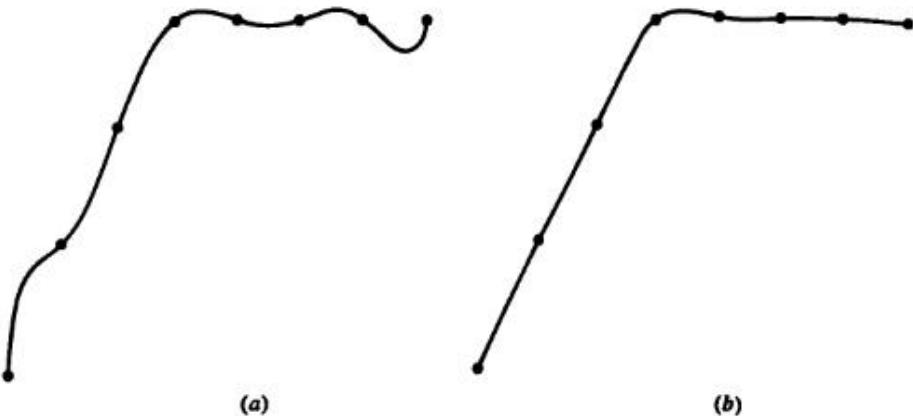


Рис. 5-17 Эффект напряжения кусочно кубического сплайна. (a) Без напряжения; (b) с напряжением.

рейка, нагруженная в некоторых точках. Чтобы устраниТЬ малые колебания, нужно приложить силу к концам рейки. Изучение нагруженных сплайнов выходит за рамки этой книги, однако мы сделаем несколько замечаний. В первых разработках [5-6] и [5-7] рассматривается экспоненциальный сплайн, требующий больших вычислений. В работе [5-8] рассматривается другой полиномиальный вид сплайна, ν -сплайн. Эффект напряжения кубического сплайна показан на рис. 5-17.

5-6 ПАРАБОЛИЧЕСКАЯ ИНТЕРПОЛЯЦИЯ

Кубические сплайны — это мощное и удобное средство, но и они небезупречны: необходимо учитывать влияние направления и величины касательных векторов, указывать все точки кривой до ее изображения, невозможна локальная коррекция кривой. Последнее особенно важно для интерактивной работы. Расчет кубического сплайна требует обращения большой матрицы, зависящей от всех элементов сплайна; т. е. изменение любого сегмента затрагивает все остальные сегменты. Воздействие уменьшается при удалении от точки возмущения, но полностью пре-небречь им нельзя. Параболическая интерполяция разрешает большинство этих проблем за счет того, что она только непрерывна, т. е. в точках соединения сегментов сохраняется непрерывность лишь первой производной. Для многих прикладных задач этого достаточно, причем параболическая интерполяция не требует больших расчетов.

Параболическая интерполяция была разработана Оверхаузером¹ [5-9]. Оверхаузер строил кривую интерполяции, исходя из геометрических соображений. Идея состоит в линейной интерполяции пересекающихся частей двух парабол. Параболы заданы четырьмя последовательными точками: первая — тремя первыми точками, вторая — тремя последними. Пересечение лежит между второй и третьей

¹ В первом издании данной книги был рассмотрен метод Оверхаузера.

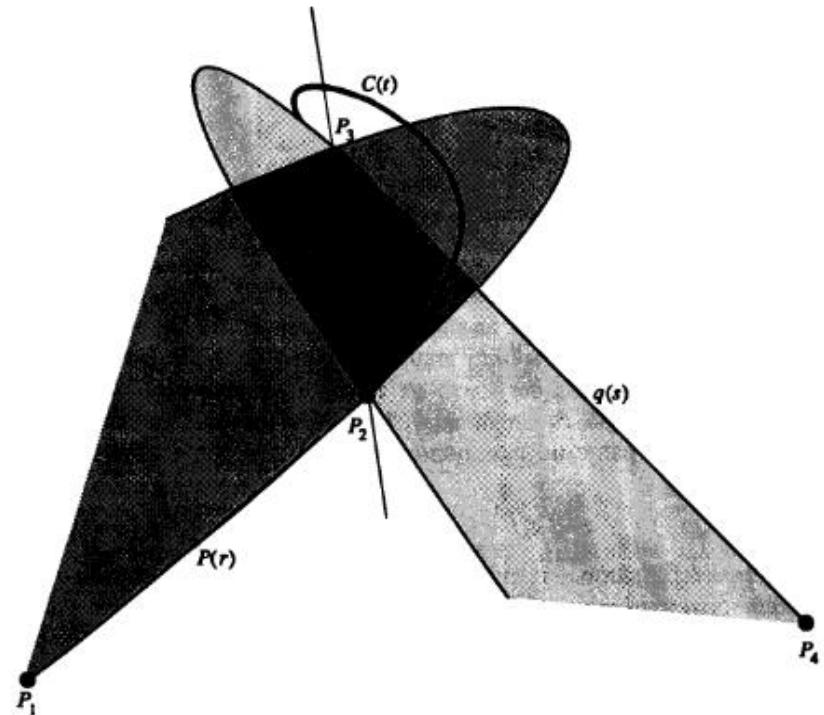


Рис. 5-18 Параболическая интерполяция.

точками. Несмотря на то, что параболы — плоские кривые, их линейная интерполяция это кубическая пространственная кривая, как показано на рис. 5-18.

В работе [5-10] рассматривается метод построения одного из семейств параболически интерполированных кривых с помощью матриц. Этот алгоритм требует меньше вычислений. Рассмотрим обобщенный вывод для всего семейства.

Параболически интерполированная кривая имеет вид

$$C(t) = (1 - t)p(r) + tq(s), \quad (5-41)$$

где \$r, s, t\$ — параметры, \$p(r), q(s)\$ — параметрические параболы, проходящие через \$P_1, P_2, P_3\$ и \$P_2, P_3, P_4\$, соответственно, как показано на рис. 5-19. Для простоты параболы на рис. 5-19 лежат в одной плоскости, но это не обязательно (рис. 5-18).

Параметрическое представление \$p(r)\$ и \$q(s)\$ следующее:

$$p(r) = [r^2 \ r \ 1][B], \quad (5-42)$$

$$q(s) = [s^2 \ s \ 1][D], \quad (5-43)$$

где \$[B]\$ и \$[D]\$ — матрицы, представляющие положение вектор-точек \$P_1, P_2, P_3\$ и \$P_2, P_3, P_4\$, соответственно. Результат интерполяции — кубическая кривая

$$C(t) = [t^3 \ t^2 \ t \ 1][A][G] = [T][A][G], \quad (5-44)$$

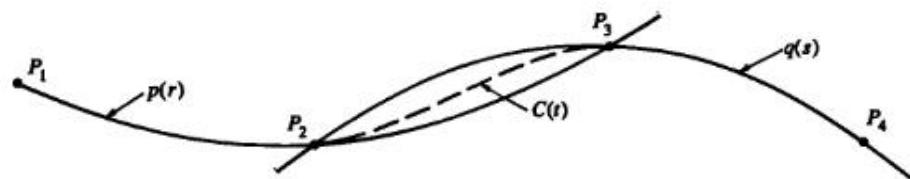


Рис. 5-19 Обозначение для параболической интерполяции.

где $[T][A]$ — матрица интерполяционных функций, а $[G]$ — геометрическая матрица вектор-точек P_1, P_2, P_3, P_4 . Чтобы определить $[B]$ и $[D]$, а затем $[A]$ и $[G]$, необходимо установить связь между параметрами r, s, t . Из рис. 5-19, замечая, что r меняется от 0 до 1 на сегменте от P_1 до P_3 вдоль $p(r)$, s меняется от 0 до 1 на сегменте от P_2 до P_4 вдоль $q(s)$, и t меняется от 0 до 1 на сегменте от P_2 до P_3 вдоль $C(t)$, разумно предположить, что r и t , а также s и t связаны линейно. Отсюда

$$r = k_1 t + k_2, \quad s = k_3 t + k_4, \quad (5-45)$$

где k_i — константы, заданные граничными условиями в вектор-точках P_1, P_2, P_3 и P_4 . Предположим, что данные распределены равномерно или почти равномерно, и диапазон параметров нормализован, т. е. $0 \leq r, s, t \leq 1$. Тогда можно условиться, что

$$p(0) = P_1, \quad p(1/2) = P_2, \quad p(1) = P_3, \quad (5-46a)$$

$$q(0) = P_2, \quad q(1/2) = P_3, \quad q(1) = P_4, \quad (5-46b)$$

$$C(0) = P_2, \quad C(1) = P_3. \quad (5-46c)$$

Здесь основные предположения таковы: $p(r) = P_2$ для $r = 1/2$ и $q(s) = P_3$ для $s = 1/2$. Эти предположения сделаны в работе [5-10]. В результате получаем единственный член семейства параболически интерполированных кривых, как будет показано ниже.

В предположениях уравнений (5-45)

$$@ P_2 : \quad r = 1/2, t = 0 \Rightarrow k_2 = 1/2$$

$$@ P_3 : \quad r = 1, t = 1 \Rightarrow k_1 + k_2 = 1 \Rightarrow k_1 = 1/2$$

$$@ P_2 : \quad s = 0, t = 0 \Rightarrow k_4 = 0$$

$$@ P_3 : \quad s = 1/2, t = 1 \Rightarrow k_3 = 1/2$$

Итак

$$r(t) = \frac{1}{2}(1+t), \quad s(t) = \frac{1}{2}t. \quad (5-47)$$

Вспомним уравнение (5-42) и используем уравнение (5-46a), чтобы выразить $[B]$ через P_1, P_2, P_3 ,

$$p(0) = P_1 = [0 \ 0 \ 1][B], \quad (5-48a)$$

$$p(1/2) = P_2 = [1/4 \ 1/2 \ 1][B], \quad (5-48b)$$

$$p(1) = P_3 = [1 \ 1 \ 1][B]. \quad (5-48c)$$

Запишем в виде одной матрицы

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1/4 & 1/2 & 1 \\ 1 & 1 & 1 \end{bmatrix} [B] = [M][B].$$

Отсюда

$$[B] = [M]^{-1} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} 2 & -4 & 2 \\ -3 & 4 & -1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}. \quad (5-49)$$

Аналогично через P_2, P_3, P_4 находится выражение $[D]$. Пользуясь уравнением (5-46b), получаем

$$q(0) = P_2 = [0 \ 0 \ 1][D], \quad (5-50a)$$

$$q(1/2) = P_3 = [1/4 \ 1/2 \ 1][D], \quad (5-50b)$$

$$q(1) = P_4 = [1 \ 1 \ 1][D]. \quad (5-50c)$$

Сравнение с уравнениями (5-48) сразу же дает

$$[D] = [M]^{-1} \begin{bmatrix} P_2 \\ P_3 \\ P_4 \end{bmatrix} = \begin{bmatrix} 2 & -4 & 2 \\ -3 & 4 & -1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_2 \\ P_3 \\ P_4 \end{bmatrix}. \quad (5-51)$$

Вспомним уравнение (5-41) и подставим уравнения (5-42) и (5-43):

$$C(t) = (1-t)[r^2 \ r \ 1][B] + t[s^2 \ s \ 1][D].$$

Используем уравнение (5-47), чтобы переписать это только в терминах параметра t ,

$$C(t) = \left[-\frac{1}{4}(t^2 + t^2 - t - 1) \quad -\frac{1}{2}(t^2 - 1) \quad 1 - t \right] [B] + \left[\frac{t^3}{4} \quad \frac{t^2}{2} \quad t \right] [D].$$

Подставив $[B]$ и $[D]$ из уравнений (5-49) и (5-51), получим

$$C(t) = \left[-\frac{t^3}{2} + t^2 - \frac{t}{2} \quad t^3 - t^2 - t + 1 \quad -\frac{t^3}{2} + \frac{t}{2} \right] \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} + \left[\frac{t^3}{2} - \frac{3}{2}t^2 + t \quad -t^3 + 2t^2 \quad \frac{t^3}{2} - \frac{t^2}{2} \right] \begin{bmatrix} P_2 \\ P_3 \\ P_4 \end{bmatrix}.$$

Перепишем уравнение так, чтобы включить все четыре точки P_1, P_2, P_3, P_4 :

$$C(t) = \begin{bmatrix} -\frac{t^3}{2} + t^2 - \frac{t}{2} & t^3 - t^2 - t + 1 & -\frac{t^3}{2} + \frac{t}{2} & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix} =$$

$$+ \begin{bmatrix} 0 & \frac{t^3}{2} - \frac{3}{2}t^2 + t & -t^3 + 2t^2 & \frac{t^3}{2} - \frac{t^2}{2} \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}.$$

Наконец, перепишем результат в форме уравнения (5.44)

$$C(t) = [t^3 \quad t^2 \quad t \quad 1][A][G] = [T][A][G], \quad (5.44)$$

где

$$[A] = \left(\frac{1}{2}\right) \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \quad (5.52)$$

и

$$[G]^T = [P_1 \quad P_2 \quad P_3 \quad P_4]. \quad (5.53)$$

Заметим, что снова (см. разд. 5-3 и уравнение (5.22)) результат имеет вид произведения матрицы интерполяционных функций и геометрической матрицы. Интерполяционные функции $[F] = [T][A]$ показаны на рис. 5-20. Реализация алгоритма очень проста. Рассмотрим подробности на примере.

Пример 5-5 Параболическая интерполяция

Рассмотрим снова четыре вектор-точки $P_1[0 \quad 0], P_2[1 \quad 1], P_3[2 \quad -1]$ и $P_4[3 \quad 0]$ из предыдущих примеров 5-2 и 5-3 (см. рис. 5-21). Найти параболическую интерполяцию между P_2 и P_3 . Вычислить промежуточные точки при $t = 1/3, 2/3$.

Из уравнений (5.44), (5.52) и (5.53) для $t = 1/3$ получаем

$$C\left(\frac{1}{3}\right) = \left(\frac{1}{2}\right) \begin{bmatrix} \frac{1}{27} & \frac{1}{9} & \frac{1}{3} & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 2 & -1 \\ 3 & 0 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{4}{3} & \frac{4}{9} \end{bmatrix}.$$

Аналогично при $t = 2/3$ имеем $C(2/3) = [5/3 \quad -4/9]$. Полученная кривая изображена на рис. 5-21.

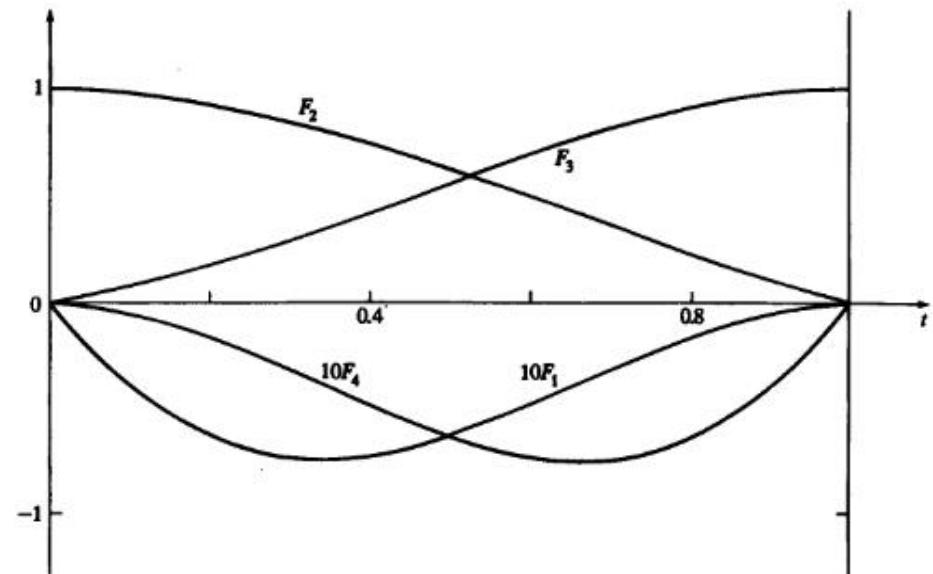


Рис. 5-20 Весовые функции для параболически интерполированных кривых в случае $\alpha = \beta = 0.5$.

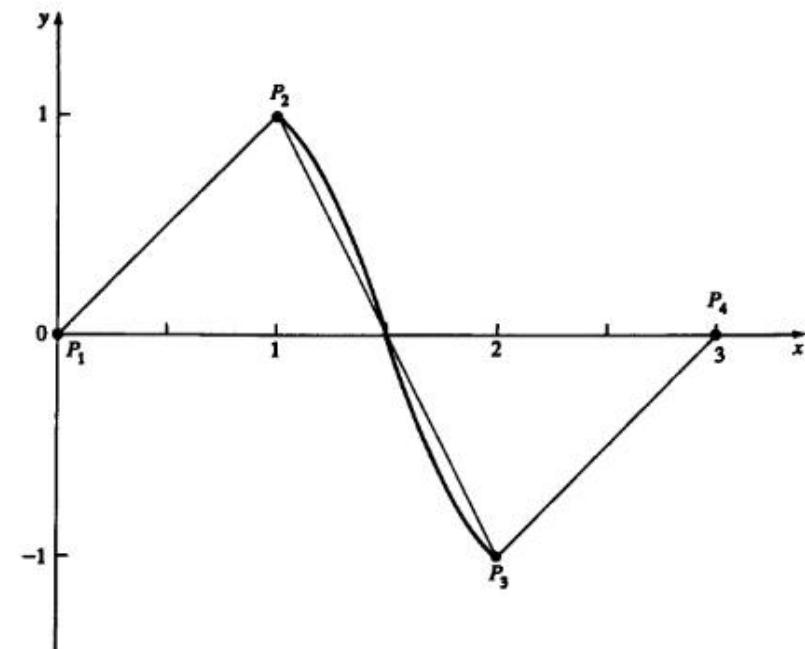


Рис. 5-21 Параболически интерполированная кривая для примера 5-5.

5-7 ОБОБЩЕННАЯ ПАРАБОЛИЧЕСКАЯ ИНТЕРПОЛЯЦИЯ

В примере параболической интерполяции в разд. 5-6 предполагается, что параметры r и s в точках P_2 и P_3 , соответственно, равны $1/2$. Если данные распределены неравномерно, кривая становится менее гладкой. Более общим предположением была бы нормализованная хордовая аппроксимация. Пусть

$$\alpha = \frac{|[P_2] - [P_1]|}{|[P_3] - [P_2]| + |[P_2] - [P_1]|}, \quad 0 < \alpha < 1, \quad (5-54)$$

и

$$\beta = \frac{|[P_3] - [P_2]|}{|[P_4] - [P_3]| + |[P_3] - [P_2]|}, \quad 0 < \beta < 1. \quad (5-55)$$

Тогда уравнения (5-46) принимает вид

$$p(0) = P_1, \quad p(\alpha) = P_2, \quad p(1) = P_3, \quad (5-56a)$$

$$q(0) = P_2, \quad q(\beta) = P_3, \quad q(1) = P_4, \quad (5-56b)$$

$$C(0) = P_2, \quad C(1) = P_3. \quad (5-56c)$$

В этих предположениях получаем линейные выражения $r(t)$ и $s(t)$ (см. (5-47))

$$r(t) = (1 - \alpha)t + \alpha, \quad s(t) = \beta t. \quad (5-57)$$

Из уравнений (5-42) и (5-56a)

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ \alpha^2 & \alpha & 1 \\ 1 & 1 & 1 \end{bmatrix} [B] = [M][B].$$

Отсюда

$$[B] = [M]^{-1} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{\alpha} & \frac{-1}{\alpha(1-\alpha)} & \frac{1}{1-\alpha} \\ \frac{-(1+\alpha)}{\alpha} & \frac{1}{\alpha(1-\alpha)} & \frac{-\alpha}{1-\alpha} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}. \quad (5-58)$$

Аналогично

$$[D] = \begin{bmatrix} \frac{1}{\beta} & \frac{-1}{\beta(1-\beta)} & \frac{1}{1-\beta} \\ \frac{-(1+\beta)}{\beta} & \frac{1}{\beta(1-\beta)} & \frac{-\beta}{1-\beta} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_2 \\ P_3 \\ P_4 \end{bmatrix}. \quad (5-59)$$

Вспомним уравнение (5-41) и подставим уравнения (5-58) и (5-59):

$$\begin{aligned} C(t) &= (1 - t)[\{(1 - \alpha)t + \alpha\}^2 \quad (1 - \alpha)t + \alpha \quad 1] \times \\ &\times \begin{bmatrix} \frac{1}{\alpha} & \frac{-1}{\alpha(1-\alpha)} & \frac{1}{1-\alpha} \\ \frac{-(1+\alpha)}{\alpha} & \frac{1}{\alpha(1-\alpha)} & \frac{-\alpha}{1-\alpha} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} + \\ &+ t[(\beta t)^2 \quad \beta t \quad 1] \begin{bmatrix} \frac{1}{\beta} & \frac{-1}{\beta(1-\beta)} & \frac{1}{1-\beta} \\ \frac{-(1+\beta)}{\beta} & \frac{1}{\beta(1-\beta)} & \frac{-\beta}{1-\beta} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_2 \\ P_3 \\ P_4 \end{bmatrix}. \end{aligned}$$

Так же, как в предыдущем разделе, это можно записать в матричном виде

$$C(t) = [t^3 \quad t^2 \quad t \quad 1][A][G] - [T][A][G], \quad (5-44)$$

где

$$[A] = \begin{bmatrix} \frac{-(1-\alpha)^2}{\alpha} & \frac{(1-\alpha)+\alpha\beta}{\alpha} & \frac{-(1-\alpha)-\alpha\beta}{1-\beta} & \frac{\beta^2}{1-\beta} \\ \frac{2(1-\alpha)^2}{\alpha} & \frac{-2(1-\alpha)-\alpha\beta}{\alpha} & \frac{2(1-\alpha)-\beta(1-2\alpha)}{(1-\beta)} & \frac{-\beta^2}{1-\beta} \\ \frac{\alpha}{-(1-\alpha)^2} & \frac{\alpha}{(1-2\alpha)} & \alpha & 0 \\ 0 & \frac{\alpha}{1} & 0 & 0 \end{bmatrix} \quad (5-60)$$

и опять

$$[G]^T = [P_1 \quad P_2 \quad P_3 \quad P_4], \quad (5-53)$$

$$F_1(t) = \frac{(1-\alpha)^2}{\alpha}(-t^3 + 2t_2 - t), \quad (5-61a)$$

$$F_2(t) = \frac{[(1-\alpha)+\alpha\beta]}{\alpha}t^3 + \frac{[-2(1-\alpha)-\alpha\beta]}{\alpha}t^2 + \frac{(1-2\alpha)}{\alpha}t + 1, \quad (5-61b)$$

$$F_3(t) = \frac{[-(1-\alpha)-\alpha\beta]}{1-\beta}t^3 + \frac{[2(1-\alpha)-\beta(1-2\alpha)]}{\alpha}t^2 + \alpha t, \quad (5-61c)$$

$$F_4(t) = \frac{\beta^2}{1-\beta}(t^3 - t^2). \quad (5-61d)$$

На рис. 5-22 изображены интерполяционные функции $F_i(t)$ для частного случая $\beta = 1 - \alpha$, т. е. когда длина хорды от P_1 до P_2 равна длине от P_3 до P_4 : $|[P_2 - P_1]| = |[P_4 - P_3]|$.

Рассмотрим это на примере.

Пример 5-6 Обобщенная параболическая интерполяция

Снова рассмотрим данные из предыдущего примера (рис. 5-22). Найти параболическую интерполяцию между P_2 и P_3 в обобщенной формулировке. Вычислить промежуточные точки при $t = 1/3, 2/3$.

Найдем α и β из уравнений (5-54) и (5-55):

$$c_{21} = |[P_2] - [P_1]| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \\ = \sqrt{(1 - 0)^2 + (1 - 0)^2} = \sqrt{2},$$

$$c_{32} = |[P_3] - [P_2]| = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} = \\ = \sqrt{(2 - 1)^2 + (-1 - 1)^2} = \sqrt{5},$$

$$c_{43} = |[P_4] - [P_3]| = \sqrt{(x_4 - x_3)^2 + (y_4 - y_3)^2} = \\ = \sqrt{(3 - 2)^2 + (0 + 1)^2} = \sqrt{2}.$$

Итак,

$$\alpha = \frac{c_{21}}{c_{32} + c_{21}} = \frac{\sqrt{2}}{\sqrt{5} + \sqrt{2}} = 0.387,$$

$$\beta = \frac{c_{32}}{c_{43} + c_{32}} = \frac{\sqrt{5}}{\sqrt{2} + \sqrt{5}} = 0.613.$$

Заметим, что $\beta = 1 - \alpha$.

Из уравнения (5-60)

$$[A] = \begin{bmatrix} -0.969 & 2.194 & -2.194 & 0.969 \\ 1.937 & -3.775 & 2.806 & -0.969 \\ -0.969 & 0.581 & 0.387 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

и из уравнения (5-44)

$$C\left(\frac{1}{3}\right) = \begin{bmatrix} \frac{1}{27} & \frac{1}{9} & \frac{1}{3} & 1 \end{bmatrix} [A] \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 2 & -1 \\ 3 & 0 \end{bmatrix} = [1.360 \quad 0.496].$$

Аналогично, при $t = 2/3$, $C(2/3) = [1.640 \quad -0.496]$. На рис. 5-23 изображена получившаяся кривая, а также кривые из примера 5-5. Замечен большой изгиб около заданных точек.

На рис. 5-24 показана локальная коррекция параболически интерполированных кривых. Здесь имеется 11 заданных точек, или радиус-векторов, и 9 параболических составных сегментов. Центральная точка располагается в трех положениях. Заметим, что влияние перемещения этой точки на форму кривой ограничивается ± 2 -я сегментами.

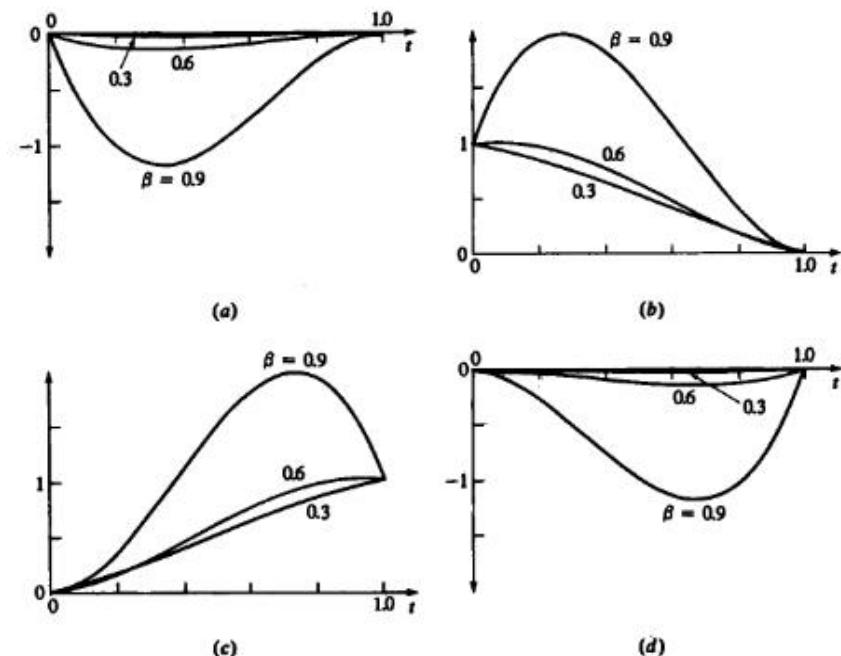


Рис. 5-22 Обобщенные параболические весовые функции, $\alpha = 1 - \beta$. (a) F_1 ; (b) F_2 ; (c) F_3 ; (d) F_4 .

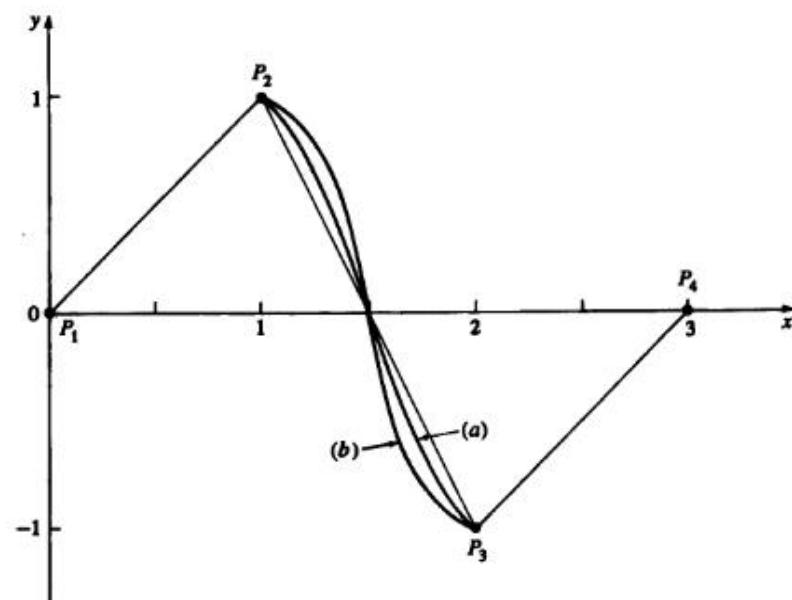


Рис. 5-23 Сравнение результатов параболической интерполяции. (a) Пример 5-5, $\alpha = \beta = 1/2$; (b) Пример 5-6, $\alpha = 1 - \beta$.

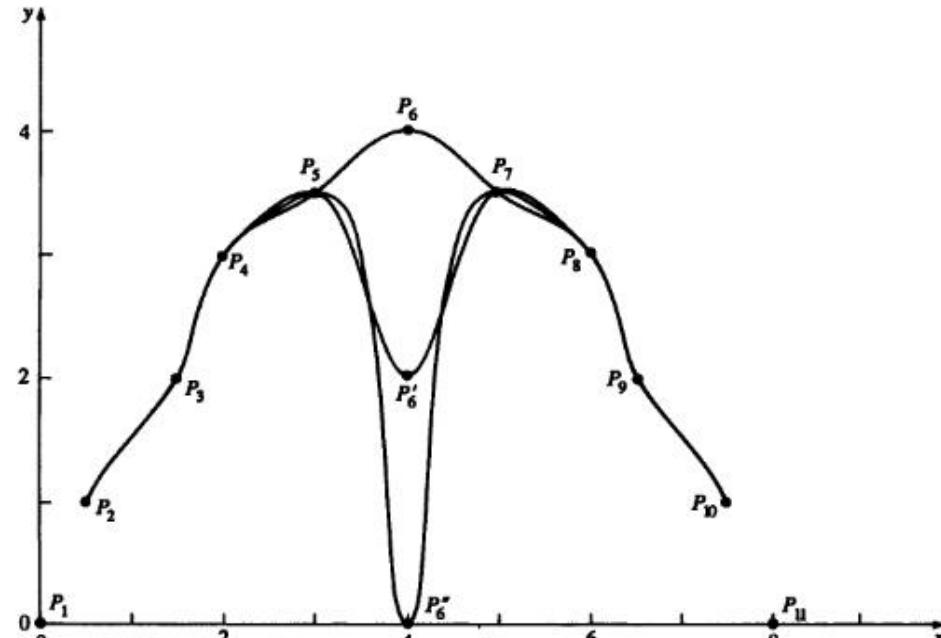


Рис. 5-24 Локальная коррекция параболически интерполированных кривых.

5-8 КРИВЫЕ БЕЗЬЕ

До сих пор мы обсуждали, как провести кривую через заданное множество точек. Рассмотренные методы во многих случаях дают прекрасные результаты и особенно удобны при описании формы, основа которой получена с помощью экспериментов или математических расчетов. Это, например, крыло самолета, составные части мотора, механические и структурные детали. Существует, однако, и другой класс задач, когда решение зависит как от функциональных, так и от эстетических требований, например дизайн поверхности машины, фюзеляжа самолета, формы корабля, мебели или посуды. Кроме количественных критериев здесь требуется учет практического опыта, и часто необходимо интерактивное вмешательство разработчика.

Рассмотренные выше методы, в частности кубические сплайны, неудобны для интерактивной работы. Направление и величина касательных не дают необходимого интуитивного представления о кривой, так как неочевидна связь между набором чисел и формой соответствующей кривой.

Пьер Безье предложил другой метод создания кривых и поверхностей любой формы. Безье вывел математическую основу своего метода из геометрических соображений [5-11] – [5-13], но в работах [5-14] и [5-15] показано, что его результат эквивалентен базису Бернштейна или функции полиномиальной аппроксимации.

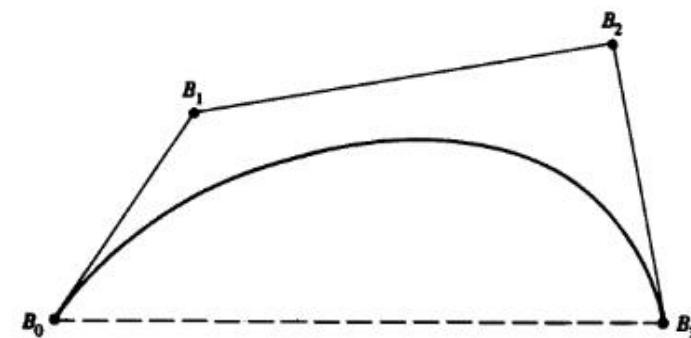


Рис. 5-25 Кривая Безье и определяющие ее точки.

Кривая Безье задается многоугольником, как показано на рис. 5-25. Так как базис Безье является бернштейновским, сразу же известны некоторые свойства кривых Безье. Например:

Функции базиса вещественны.

Степень многочлена, определяющего участок кривой, на единицу меньше количества точек соответствующего многоугольника.

Основа формы кривой повторяет очертания многоугольника.

Первая и последняя точки кривой совпадают с соответствующими точками определяющего многоугольника.

Векторы касательных в концах кривой по направлению совпадают с первой и последней сторонами многоугольника.

Кривая лежит внутри выпуклой оболочки многоугольника, т. е. внутри самого большого многоугольника, построенного по заданным точкам. На рис. 5-25 выпуклая оболочка обозначена штриховой и тонкой линиями.

Кривая обладает свойством уменьшения вариации. Это означает, что кривая пересекает любую прямую линию не чаще, чем определяющий многоугольник.

Кривая инвариантна относительно аффинных преобразований.

На рис. 5-26 показано несколько четырехточечных многоугольников Безье и соответствующих кривых. На основе перечисленных выше свойств можно легко научиться предсказывать форму кривой по виду многоугольника.

Математическое параметрическое представление кривой Безье имеет вид

$$P(t) = \sum_{i=0}^n B_i J_{n,i}(t), \quad 0 \leq t \leq 1, \quad (5-62)$$

где базис Безье или Бернштейна, или функция аппроксимации

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (5-63)$$

с

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} \quad (5-64)$$

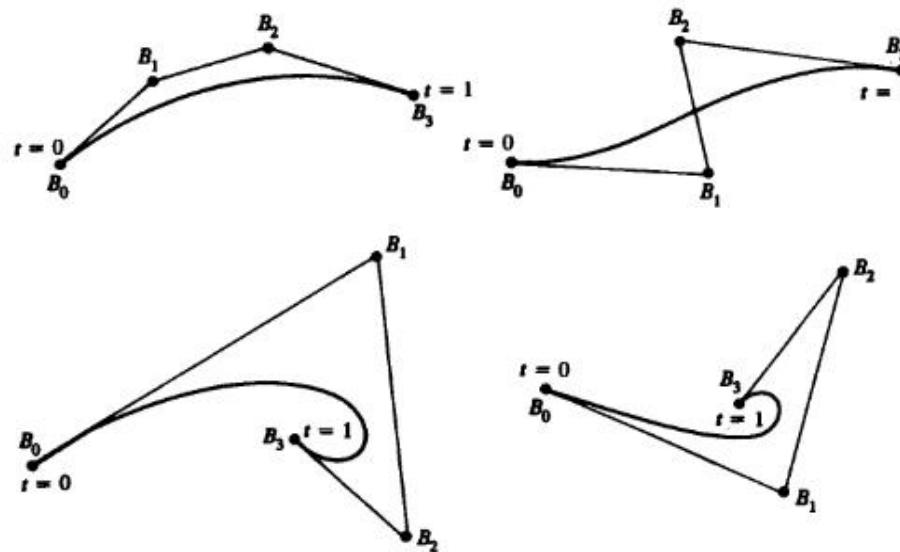


Рис. 5-26 Многоугольники Безье для кубических кривых.

$J_{n,i}(t)$ — это i -я функция базиса Бернштейна порядка n . Здесь n — порядок определяющей функции базиса Бернштейна — и, следовательно, сегмента полиномиальной кривой, на единицу меньше количества точек определяющего многоугольника. Как показано на рис. 5-25, вершины многоугольника Безье нумеруются от 0 до n^1 . Поэтому $(0)^0 \equiv 1$ и $0! \equiv 1$.

На рис. 5-27 изображены аппроксимирующие функции для разных значений n . Заметим, что функции симметричны. Каждая функция имеет порядок n , например все четыре функции на рис. 5-27b для $n = 3$ кубические. Максимум каждой функции достигается при $t = i/n$ и равен (5-14).

$$J_{n,i}\left(\frac{i}{n}\right) = \binom{n}{i} \frac{i^i(n-i)^{n-i}}{n^n}. \quad (5-65)$$

¹ Обозначения и отсчет выбраны в соответствии с традициями, принятыми в литературе по кривым Безье и базису Бернштейна. Для программирования удобнее другое представление

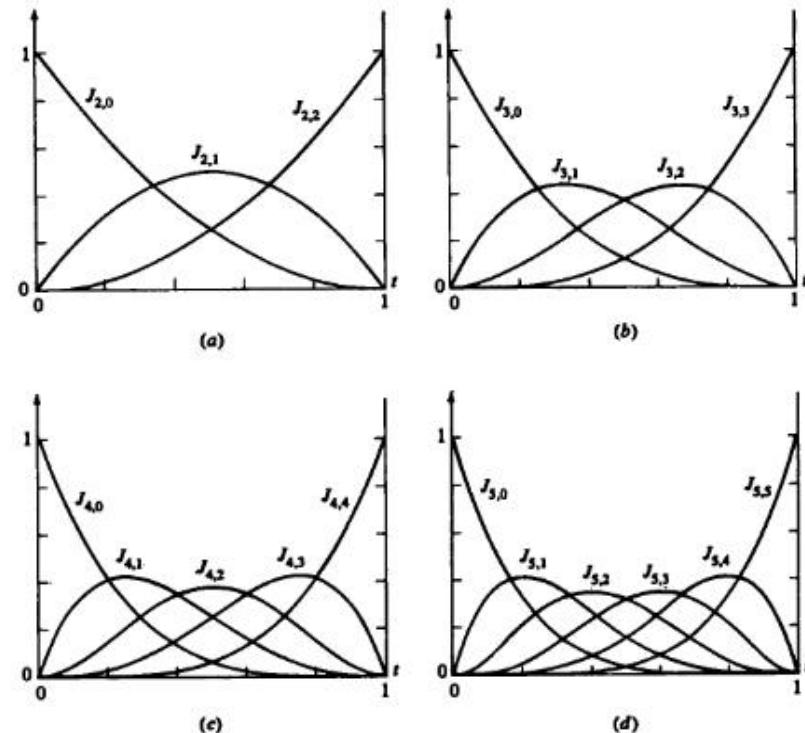
$$P(t) = \sum_{I=1}^N B_I J_{N,I}(t),$$

где

$$J_{N,I} = \binom{N-1}{I-1} t^{I-1} (1-t)^{N-I}$$

$$\binom{N-1}{I-1} = \frac{(N-1)!}{(I-1)!(N-I)!}.$$

Точки многоугольника нумеруются от 1 до N . Для перехода к этим обозначениям положим $n = N - 1$ и $i = I - 1$.

Рис. 5-27 Весовые функции Безье/Бернштейна. (a) Многоугольник из трех точек, $n = 2$; (b) из четырех точек, $n = 3$; (c) из пяти точек, $n = 4$; (d) из шести точек, $n = 5$.

Например, для кубической кривой $n = 3$. Максимум $J_{3,1}$ и $J_{3,2}$ достигается в $1/3$ и $2/3$, соответственно, и имеет значения

$$J_{3,1}\left(\frac{1}{3}\right) = \frac{4}{9} \quad \text{и} \quad J_{3,2}\left(\frac{2}{3}\right) = \frac{4}{9}.$$

Рисунок 5-27b иллюстрирует этот пример.

Рассмотрим уравнения (5-62) и (5-64) для первой точки на кривой, т. е. при $t = 0$

$$J_{n,0}(0) = \frac{n!(1)(1-0)^{n-0}}{n!} = 1, \quad i = 0,$$

и

$$J_{n,i}(0) = \frac{n!(0)^i(1-0)^{n-i}}{i!(n-i)!} = 0, \quad i \neq 0.$$

Итак,

$$P(0) = B_0 J_{n,0}(0) = B_0,$$

первая точка кривой совпадает с первой точкой многоугольника.

Аналогично, для последней точки кривой, т. е. при $t = 1$

$$J_{n,n}(1) = \frac{n!(1)^n(0)^{n-n}}{n!(1)} = 1, \quad i = n,$$

$$J_{n,i}(1) = \frac{n!}{i!(n-i)!} t^i (1-1)^{n-i} = 0, \quad i \neq n.$$

Итак,

$$P(1) = B_n J_{n,n}(1) = B_n$$

и последняя точка на кривой Безье совпадает с последней точкой определяющего многоугольника.

Далее, можно показать, что для любого значения параметра t сумма базисных функций равна 1, т. е.

$$\sum_{i=0}^n J_{n,i}(t) = 1. \quad (5-66)$$

Рассмотрим метод построения Безье на примере.

Пример 5-7 Кривая Безье

Пусть заданы вершины многоугольника Безье $B_0[1 \ 1]$, $B_1[2 \ 3]$, $B_2[4 \ 3]$ и $B_3[3 \ 1]$. Найти семь точек, лежащих на кривой Безье.

Рассмотрим уравнения (5-62) – (5-64):

$$P(t) = \sum_{i=0}^n B_i J_{n,i}(t),$$

где

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

и

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}.$$

В нашем случае $n = 3$, так как имеется четыре вершины. Отсюда

$$\binom{n}{i} = \binom{3}{i} = \frac{6}{i!(3-i)!}$$

и

$$J_{3,0}(t) = (1)t^0(1-t)^3 = (1-t)^3,$$

$$J_{3,1}(t) = 3t(1-t)^2,$$

$$J_{3,2}(t) = 3t^2(1-t),$$

$$J_{3,3}(t) = t^3.$$

Итак,

$$\begin{aligned} P(t) &= B_0 J_{3,0} + B_1 J_{3,1} + B_2 J_{3,2} - B_3 J_{3,3} \\ &= (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3. \end{aligned}$$

Таблица 5-4 Коэффициенты для кривой Безье

t	$J_{3,0}$	$J_{3,1}$	$J_{3,2}$	$J_{3,3}$
0	1	0	0	0
0.15	0.614	0.325	0.058	0.003
0.35	0.275	0.444	0.239	0.042
0.5	0.125	0.375	0.375	0.125
0.65	0.042	0.239	0.444	0.275
0.85	0.003	0.058	0.325	0.614
1	0	0	0	1

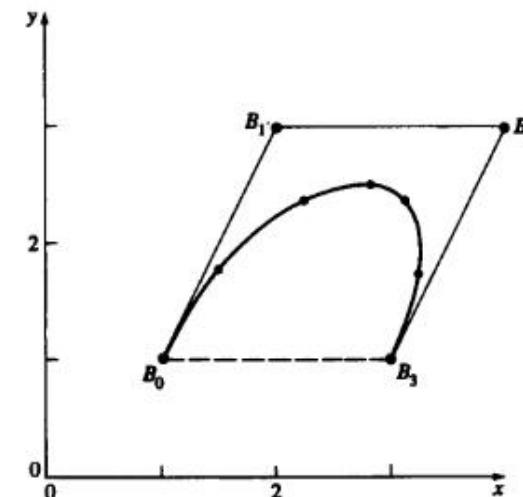


Рис. 5-28 Сегмент кривой Безье, пример 5-7.

Значения $J_{n,i}$ для различных значений t приведены в табл. 5-4.

Точки на кривой:

$$P(0) = B_0 = [1 \ 1],$$

$$P(0.15) = 0.614B_0 + 0.325B_1 + 0.058B_2 + 0.003B_3 = [1.5 \ 1.765],$$

$$P(0.35) = 0.275B_0 + 0.444B_1 + 0.239B_2 + 0.042B_3 = [2.248 \ 2.367],$$

$$P(0.5) = 0.125B_0 + 0.375B_1 + 0.375B_2 + 0.125B_3 = [2.75 \ 2.5],$$

$$P(0.65) = 0.042B_0 + 0.239B_1 + 0.444B_2 + 0.275B_3 = [3.122 \ 2.367],$$

$$P(0.85) = 0.003B_0 + 0.058B_1 + 0.325B_2 + 0.614B_3 = [3.248 \ 1.765],$$

$$P(1) = B_3 = [3 \ 1].$$

Эти точки показаны на определяющем многоугольнике на рис. 5-28.

Уравнение кривой Безье можно записать в матричном виде, так же как уравнения для кубических сплайнов и параболической интерполяции (см. уравнения

5-27 и 5-44):

$$P(t) = [T][N][G] = [F][G]. \quad (5-67)$$

Здесь $[F] = [J_{n,0} \ J_{n,1} \ \dots \ J_{n,n}]$ и $[G]^T = [B_0 \ B_1 \ \dots \ B_n]$.Особенный интерес представляют матричные формы для малых значений n .Для многоугольника из четырех точек ($n = 3$) кривая Безье имеет вид

$$P(t) = [(1-t)^3 \ 3t(1-t)^2 \ 3t^2(1-t) \ t^3] \begin{bmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{bmatrix}.$$

Группируя коэффициенты, получим

$$P(t) = [T][N][G] = [t^3 \ t^2 \ t \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{bmatrix}. \quad (5-68)$$

Аналогично, кривая Безье четвертого порядка ($n = 4$), заданная многоугольником из пяти точек:

$$P(t) = [t_4 \ t_3 \ t_2 \ t \ 1] \begin{bmatrix} 1 & -4 & 6 & -4 & 1 \\ -4 & 12 & -12 & 4 & 0 \\ 6 & -12 & 6 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix}. \quad (5-69)$$

В работе [5-19] приводится обобщенное представление:

$$P(t) = [T][N][G],$$

где

$$[T] = [t^n \ t^{n-1} \ \dots \ t \ 1],$$

$$[N] = \begin{bmatrix} \binom{n}{0} \binom{n}{n} (-1)^n & \binom{n}{1} \binom{n-1}{n-1} (-1)^{n-1} & \dots & \binom{n}{n} \binom{n-n}{n-n} (-1)^0 \\ \binom{n}{0} \binom{n}{n-1} (-1)^{n-1} & \binom{n}{1} \binom{n-1}{n-2} (-1)^{n-2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \binom{n}{0} \binom{n}{1} (-1)^1 & \binom{n}{1} \binom{n-1}{0} (-1)^0 & \dots & 0 \\ \binom{n}{0} \binom{n}{0} (-1)^0 & 0 & \dots & 0 \end{bmatrix}. \quad (5-70)$$

Матрица $[G]^T$ — это опять $[B_0 \ B_1 \ \dots \ B_n]$. Отдельные члены матрицы $[N]$ таковы:

$$(N_{i+1,j+1})_{i,j=0}^n = \begin{cases} \binom{n}{j} \binom{n-j}{n-i-j} (-1)^{n-i-j}, & 0 \leq i+j \leq n, \\ 0 & \text{иначе} \end{cases}$$

Уравнение (5-70) можно записать в более удобном виде

$$[N] = [C][D], \quad (5-71)$$

где

$$[C] = \begin{bmatrix} \binom{n}{n} (-1)^n & \binom{n}{1} \binom{n-1}{n-1} (-1)^{n-1} & \dots & \binom{n}{n} \binom{n-n}{n-n} (-1)^0 \\ \binom{n}{n-1} (-1)^{n-1} & \binom{n}{1} \binom{n-1}{n-2} (-1)^{n-2} & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \binom{n}{1} (-1)^1 & \binom{n}{1} \binom{n-1}{0} (-1)^0 & & 0 \\ \binom{n}{0} (-1)^0 & 0 & \dots & 0 \end{bmatrix},$$

$$[D] = \begin{bmatrix} \binom{n}{0} & \dots & 0 \\ \vdots & \binom{n}{0} & \vdots \\ \vdots & \ddots & \vdots \\ 0 & \dots & \binom{n}{n} \end{bmatrix}.$$

Уравнения (5-70) или (5-71) удобнее для расчета при больших значениях n . Заметим, что для всех n матрица $[N]$ симметрична относительно главной диагонали и правый нижний угол состоит из нулей.

Для каждой отдельной кривой Безье необходимо знать касательные векторы в ее концах, но если необходимо соблюдать непрерывность кривизны и на-клона в точках соединения кривых, рассчитывать нормали к поверхности для освещения, вычислять локальную кривизну, то требуется знать как первую, так и вторую производную кривой Безье.

Из уравнения (5-62) первая производная кривой Безье имеет вид:

$$P'(t) = \sum_{i=0}^n B_i J'_{n,i}(t). \quad (5-72)$$

Вторая производная такова:

$$P''(t) = \sum_{i=0}^n B_i J''_{n,i}(t). \quad (5-73)$$

Формально дифференцируя уравнение (5-63), получаем производные базисных функций

$$\begin{aligned} J'_{n,i}(t) &= \binom{n}{i} \{i t^{i-1} (1-t)^{n-i} - (n-i) t^i (1-t)^{n-i-1}\} \\ &= \binom{n}{i} t^i (1-t)^{n-i} \left\{ \frac{i}{t} - \frac{(n-i)}{(1-t)} \right\} \\ &= \frac{(i-n)t}{t(1-t)} J_{n,i}(t). \end{aligned} \quad (5-74)$$

Аналогично, вторые производные имеют вид:

$$J''_{n,i}(t) = \left\{ \frac{(i-nt)^2 - nt^2 - i(1-2t)}{t^2(1-t)^2} \right\} J_{n,i}(t). \quad (5-75)$$

В начале и конце кривой Безье, т. е. при $t = 0$ и $t = 1$, численный расчет уравнений (5-74) и (5-75) представляет затруднения¹.

Другой способ вычисления r -й производной при $t = 0$:

$$P^r(0) = \frac{n!}{(n-r)!} \sum_{i=0}^r (-1)^{r-i} \binom{r}{i} B_i \quad (5-76)$$

и при $t = 1$:

$$P^r(1) = \frac{n!}{(n-r)!} \sum_{i=0}^r (-1)^i \binom{r}{i} B_{n-i}. \quad (5-77)$$

Отсюда первые производные в концах будут

$$P'(0) = n(B_1 - B_0) \quad (5-78)$$

и

$$P'(1) = n(B_n - B_{n-1}). \quad (5-79)$$

Это показывает, что касательные к кривой Безье в первой и последней точках параллельны соответствующим сторонам многоугольника.

Аналогично, вторые производные в концах таковы:

$$P''(0) = n(n-1)(B_0 - 2B_1 + B_2) \quad (5-80a)$$

и

$$P''(1) = n(n-1)(B_n - 2B_{n-1} + B_{n-2}). \quad (5-80b)$$

Вторые производные в концах зависят от двух ближайших сторон, т. е. от трех ближайших вершин. В общем случае, r -я производная в начальной и конечной точках зависит от этих точек и r ближайших вершин многоугольника.

Подробнее рассмотрим это на примере.

Пример 5-8 Производные кривых Безье

Рассмотрим четырехточечный многоугольник Безье, например, как на рис. 5-26 и 5-28. Вспомним представление кривой

$$P(t) = B_0 J_{3,0}(t) + B_1 J_{3,1}(t) + B_2 J_{3,2}(t) + B_3 J_{3,3}(t).$$

Отсюда первая производная

$$P'(t) = B_0 J'_{3,0}(t) + B_1 J'_{3,1}(t) + B_2 J'_{3,2}(t) + B_3 J'_{3,3}(t).$$

¹ Правильный результат получается при алгебраическом вычислении и подстановке до численного расчета (см. пример 5-8).

Вспомним пример 5-7 и непосредственно продифференцируем базисные функции

$$\begin{aligned} J_{3,0}(t) &= (1-t)^3 & \Rightarrow J'_{3,0}(t) &= -3(1-t)^2, \\ J_{3,1}(t) &= 3t(1-t)^2 & \Rightarrow J'_{3,1}(t) &= 3(1-t)^2 - 6t(1-t), \\ J_{3,2}(t) &= 3t^2(1-t) & \Rightarrow J'_{3,2}(t) &= 6t(1-t) - 3t^2, \\ J_{3,3}(t) &= t^3 & \Rightarrow J'_{3,3}(t) &= 3t^2 \end{aligned}$$

Подставим $t = 0$:

$$J'_{3,3}(0) = -3, \quad J'_{3,1}(0) = 3, \quad J'_{3,2}(0) = 0, \quad J'_{3,3}(0) = 0.$$

Подстановка дает

$$P'(0) = -3P_0 + 3P_1 = 3(P_1 - P_0).$$

Поэтому направление касательной в начале кривой совпадает с первой стороной многоугольника (см. рис. 5-28).

В конце кривой $t = 1$ и

$$J'_{3,0}(1) = 0, \quad J'_{3,1}(1) = 0, \quad J'_{3,2}(1) = -3, \quad J'_{3,3}(1) = 3.$$

Аналогично, подстановка дает

$$P'(1) = -3P_2 + 3P_3 = 3(P_3 - P_2)$$

и направление касательного вектора в конце кривой совпадает с последней стороной многоугольника.

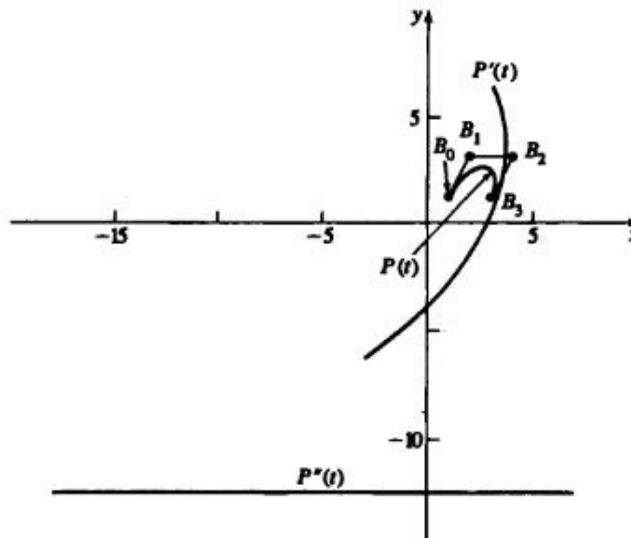
Чтобы вычислить производные вдоль кривой, воспользуемся функциями базиса и уравнениями (5-74) и (5-75):

$$\begin{aligned} J'_{3,0}(t) &= \frac{(0-3t)}{t(1-t)}(1-t)^3 = -3(1-t)^2, \\ J'_{3,1}(t) &= \frac{(1-3t)}{t(1-t)}(3t)(1-t)^2 = 3(1-3t)(1-t) = 3(1-4t+3t^2), \\ J'_{3,2}(t) &= \frac{(2-3t)}{t(1-t)}(3t^2)(1-t) = 3t(2-3t), \\ J'_{3,3}(t) &= \frac{3(1-t)}{t(1-t)}t^3 = 3t^2. \end{aligned}$$

Результаты легко вычисляются как для $t = 0$, так и для $t = 1$. Подставляя в уравнение (5-72), получаем первую производную в любой точке кривой. Например, при $t = 1/2$ имеем

$$\begin{aligned} P'\left(\frac{1}{2}\right) &= -3\left(1-\frac{1}{2}\right)^2 B_0 + 3\left(1-\frac{3}{2}\right)\left(1-\frac{1}{2}\right) B_1 + \left(\frac{3}{2}\right)\left(2-\frac{3}{2}\right) B_2 + \frac{3}{4} B_3 \\ &= -\frac{3}{4}B_0 - \frac{3}{4}B_1 + \frac{3}{4}B_2 + \frac{3}{4}B_3 = -\frac{3}{4}(B_0 + B_1 - B_2 - B_3). \end{aligned}$$

Результат для точек B_0, B_1, B_2, B_3 из примера 5-7 изображен на рис. 5-29.

Рис. 5-29 Кривая Безье и ее производные: $P(t)$; $P'(t)$; $P''(t)$.

Аналогично, вторые производные имеют вид:

$$J_{3,0}''(t) = \frac{\{(-3t)^2 - 3t^2\}}{t^2(1-t)^2}(1-t)^3 = 6(1-t),$$

$$J_{3,1}''(t) = \frac{\{(1-3t)^2 - 3t^2 - (1-2t)\}}{t^2(1-t)^2}(3t)(1-t)^2 = -6(2-3t),$$

$$J_{3,2}''(t) = \frac{\{(2-3t)^2 - 3t^2 - 2(1-2t)\}}{t^2(1-t)^2}(3t^2)(1-t) = 6(1-3t),$$

$$J_{3,3}''(t) = \frac{\{(3-3t)^2 - 3t^2 - 3(1-2t)\}}{t^2(1-t)^2}t^3 = 6t.$$

Уравнение (5-73) при $t = 1/2$ дает

$$\begin{aligned} P''\left(\frac{1}{2}\right) &= 6\left(1-\frac{1}{2}\right)B_0 - 6\left(2-\frac{3}{2}\right)B_1 + 6\left(1-\frac{3}{2}\right)B_2 + 3B_3 = \\ &= 3B_0 - 3B_1 - 3B_2 + 3B_3 = 3(B_0 - B_1 - B_2 + B_3). \end{aligned}$$

Иллюстрация также приведена на рис. 5-29. Заметим, что вектор из начала координат в любую точку на каждой из кривых представляет соответственно направление и величину радиуса-вектора и приблизительную кривизну в этой точке на кривой.

Условие непрерывности соседних кривых Безье формулируется очень просто. Пусть кривая Безье $P(t)$ степени n задана вершинами B_i , а соседняя кривая Безье $Q(s)$ степени m — вершинами C_i . Тогда непрерывность первой производной в точке соединения выражается соотношением

$$P'(1) = gQ'(0),$$

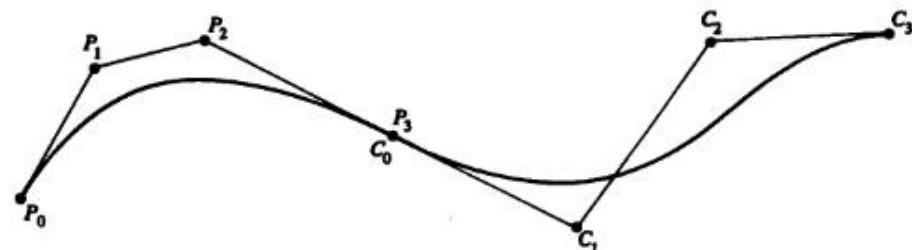


Рис. 5-30 Непрерывность первой производной для кубических кривых Безье.

где g — скаляр. Пользуясь уравнениями (5-78) и (5-79), получим

$$C_1 - C_0 = \frac{n}{m}(B_n - B_{n-1}).$$

Из непрерывности кривой следует, что $C_0 = B_n$ и

$$C_1 = \frac{n}{m}(B_n - B_{n-1}) + B_n.$$

Отсюда направления касательных на стыке совпадают, если три вершины $B_{n-1}, B_n = C_0, C_1$ коллинеарны, т. е. B_n должна лежать на линии между B_{n-1} и C_1 .

Если совпадают еще и величины касательных векторов, то $B_n = C_0$ является серединой отрезка от B_{n-1} до C_1 :

$$C_1 - C_0 = B_n - B_{n-1} = C_0 - B_{n-1}$$

или

$$C_1 + B_{n-1} = 2C_0 = 2B_n.$$

На рис. 5-30 приведена иллюстрация для $n = m = 3$, т. е. для двух кубических кривых Безье.

Условие непрерывности вторых производных между соседними кривыми:

$$m(m-1)(C_0 - 2C_1 + C_2) = n(n-1)(B_{n-2} - 2B_{n-1} + B_n).$$

Из условий C^0 и C^1 непрерывности в точке соединения получаем положение третьей определяющей вершины второго многоугольника, выраженное через три последние вершины первого многоугольника, а именно

$$C_2 = \frac{n(n-1)}{m(m-1)}B_{n-2} - 2\left\{\frac{n(n-1)}{m(m-1)} + \frac{n}{m}\right\}B_{n-1} + \left\{1 + 2\frac{n}{m} + \frac{n(n-1)}{m(m-1)}\right\}B_n$$

Это означает, что вершины $B_{n-2}, B_{n-1}, B_n = C_0, C_1, C_2$ должны либо составлять выпуклый многоугольник, либо лежать на одной прямой, чтобы сохранить C^2

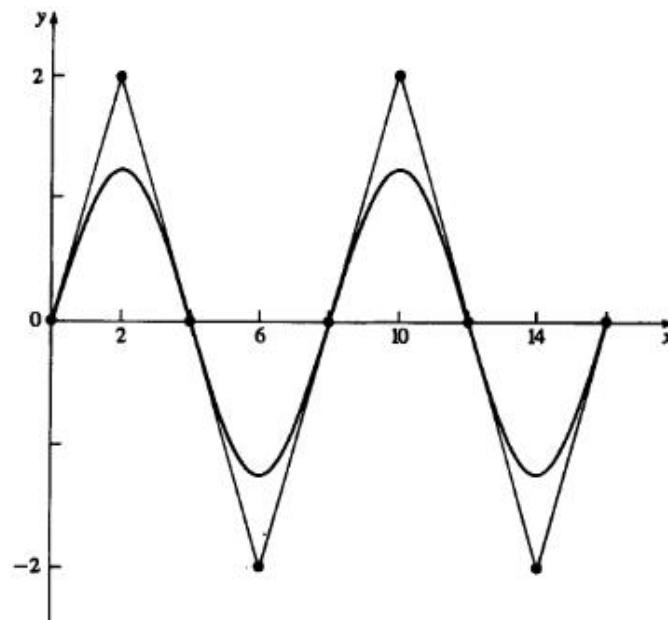


Рис. 5-31 Непрерывность второй производной для кривых Безье четвертой степени.

непрерывность на стыке. Для кубических кривых Безье ($n = m = 3$) это условие имеет вид

$$C_2 = B_{n-1} - 4(B_{n-1} - B_n).$$

Несколько карандашных набросков на бумаге покажут, что данное требование существенно ограничивает множество кривых; поэтому на практике для соблюдения непрерывности вторых производных используются полиномиальные кривые более высокого порядка. На рис. 5-31 приведен пример непрерывности вторых производных для двух пятиточечных кривых Безье.

Если необходима более гибкая кривая Безье, увеличивается количество определяющих точек и степень полинома. Для каждой точки на кривой Безье с n определяющими вершинами многоугольника B_0, \dots, B_n та же самая точка на новой кривой Безье с $n + 1$ определяющими вершинами B_0^*, \dots, B_{n+1}^* задается в виде

$$P(t) = \sum_{i=0}^n B_i J_{n,i}(t) = \sum_{i=0}^{n+1} B_i^* J_{n+1,i}(t), \quad (5-81)$$

где

$$B_0^* = B_0,$$

$$B_i^* = \alpha_i B_{i-1} + (1 - \alpha_i) B_i, \quad \alpha_i = \frac{i}{n+1}, \quad i = 1, \dots, n,$$

$$B_{n+1}^* = B_n.$$

Здесь можно с успехом применить методику из работы [5-14]. В предельном случае многоугольник сходится к кривой. Дополнительную гибкость кривой можно также получить разбиением кривой Безье на две новых таким образом, чтобы они вместе совпадали с исходной кривой. В работах Барского [5-16, 5-18] было показано, что любая кривая Безье может быть разбита с использованием произвольного параметра в диапазоне $0 \leq t \leq 1$. Простейший случай — это средняя точка, т. е. $t = 1/2$ (см. [5-18]). При разбиении средней точкой получаются два специальных вида кубических кривых Безье.

Кубическая кривая Безье (см. упражнение 5-7) задается в виде

$$P(t) = (1-t)^3 B_0 + 3t(1-t)^2 B_1 + 3t^2(1-t) B_2 + t^3 B_3, \quad 0 \leq t \leq 1.$$

с определяющими вершинами B_0, B_1, B_2, B_3 . Тогда многоугольник C_0, C_1, C_2, C_3 определяет кривую Безье $Q(u)$, $0 \leq u \leq 1$, соответствующую первой половине исходной кривой, а именно $P(t)$, $0 \leq t \leq 1/2$, как и требовалось. Подобным образом, многоугольник D_0, D_1, D_2, D_3 определяет кривую Безье $R(v)$, $0 \leq v \leq 1$, соответствующую второй половине исходной кривой, т. е. $P(t)$, $1/2 \leq t \leq 1$, как и требовалось. Новые определяющие вершины многоугольника C_i и D_i получаются путем приравнивания радиус-векторов и касательных векторов при $u = 0$, $t = 0$; $u = 1$, $t = 1/2$ и $v = 0$, $t = 1/2$; $v = 1$, $t = 1$. Из соотношений (5-62) и (5-72) получаем

$$C_0 = B_0,$$

$$3(C_1 - C_0) = \frac{3}{2}(B_1 - B_0),$$

$$3(C_3 - C_2) = \frac{3}{8}(B_3 + B_2 - B_1 - B_0),$$

$$C_3 = \frac{1}{8}(B_3 + 3B_2 + 3B_1 + B_0).$$

Решение этих уравнений дает

$$C_0 = B_0,$$

$$C_1 = \frac{1}{2}(B_1 + B_0),$$

$$C_2 = \frac{1}{4}(B_2 + 2B_1 + B_0),$$

$$C_3 = \frac{1}{8}(B_3 + 3B_2 + 3B_1 + B_0).$$

Аналогично,

$$D_0 = \frac{1}{8}(B_3 + 3B_2 + 3B_1 + B_0),$$

$$D_1 = \frac{1}{4}(B_3 + 2B_2 + B_1),$$

$$D_2 = \frac{1}{2}(B_3 + B_2),$$

$$D_3 = B_3.$$

Эти результаты обобщаются до

$$C_i = \sum_{j=0}^i \binom{i}{j} \frac{B_j}{2^i}, \quad i = 0, 1, \dots, n, \quad (5-82a)$$

$$D_i = \sum_{j=1}^n \binom{n-i}{n-j} \frac{B_j}{2^{n-i}}, \quad i = 0, 1, \dots, n. \quad (5-82b)$$

Дальнейшее применение определяющих многоугольников с помощью кривой Безье оставляем для самостоятельной работы.

5-9 В-СПЛАЙНЫ

С математической точки зрения кривая, заданная вершинами многоугольника, зависит от интерполяции или аппроксимации, устанавливающей связь кривой и многоугольника. Здесь основой является выбор базисных функций. Как было отмечено в разд. 5-8, базис Бернштейна порождает кривые Безье вида (5-62), но он обладает двумя свойствами, которые ограничивают гибкость кривых. Во-первых, количество вершин многоугольника жестко задает порядок многочлена. Например, кубическая кривая должна быть задана четырьмя вершинами и тремя отрезками. Многоугольник из шести точек всегда порождает кривую пятого порядка. Единственный способ понизить степень кривой — это сократить количество вершин, а повысить степень кривой — увеличить их число.

Второе ограничение следует из глобальной природы базиса Бернштейна. Это означает, что величина аппроксимирующих функций $J_{n,i}(t)$ из уравнения (5-63) ненулевая для всех значений параметра на кривой. Любая точка на кривой Безье зависит от всех определяющих вершин, поэтому изменение какой-либо одной вершины оказывает влияние на всю кривую. Локальные воздействия на кривую невозможны.

Например, так как наклон концов кривой Безье задан соответствующими сторонами многоугольника, можно передвинуть среднюю вершину пятиточечного многоугольника, не меняя направления на концах. Однако из-за глобальности базиса Бернштейна меняется форма всей кривой. Отсутствие локальной коррекции может оказаться решающим в некоторых прикладных задачах.

Существует неглобальный базис, называемый базисом В-сплайна, включающий базис Бернштейна как частный случай. В-сплайны неглобальны, так как с каждой вершиной B_i связана своя базисная функция. Поэтому влияние каждой вершины на кривую проявляется только при тех значениях параметра, где соответствующая базисная функция не равна нулю. Базис В-сплайна также позволяет менять порядок базисных функций и, следовательно, всей кривой без изменения количества вершин. Теория В-сплайнов была предложена в работе [5-20]. Рекурсивное определение для численного решения было выведено независимо Коксом [5-21] и де Буром [5-22]. Гордон и Ризенфельд [5-15], [5-23] определяли кривые через базис В-сплайна.

Пусть $P(t)$ определяет кривую как функцию от параметра t , тогда В-сплайн имеет вид

$$P(t) = \sum_{i=1}^{n+1} B_i N_{i,k}(t), \quad t_{\min} \leq t < t_{\max}, \quad 2 \leq k \leq n+1, \quad (5-83)$$

где B_i есть $n+1$ вершина многоугольника, а $N_{i,k}$ — нормализованные функции базиса В-сплайна.

Для i -й нормализованной функции базиса порядка k (степени $k-1$) функции базиса $N_{i,k}(t)$ определяются рекурсивными формулами Кокса—де Бура:

$$N_{i,1}(t) = \begin{cases} 1 & \text{если } x_i \leq t < x_{i+1} \\ 0 & \text{иначе} \end{cases} \quad (5-84a)$$

и

$$N_{i,k}(t) = \frac{(t - x_i)N_{i,k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t)N_{i+1,k-1}(t)}{x_{i+k} - x_{i+1}}. \quad (5-84b)$$

Величины x_i — это элементы узлового вектора, удовлетворяющие отношению $x_i \leq x_{i+1}$. Параметр t изменяется от t_{\min} до t_{\max} вдоль кривой $P(t)$ ¹. Считается, что $0/0 = 0$.

Формально В-сплайн определяется как полиномиальный сплайн порядка k (степени $k-1$), так как он удовлетворяет следующим условиям:

Функция $P(t)$ является полиномом степени $k-1$ на каждом интервале $x_i \leq t < x_{i+1}$.

$P(t)$ и ее производные порядка $1, 2, \dots, k-2$ непрерывны вдоль всей кривой.

Так, например, В-сплайн четвертого порядка — это кусочная кубическая кривая.

Из того что В-сплайн задается базисом В-сплайна, сразу следует еще несколько его свойств:

Сумма базисных функций В-сплайна для любого значения параметра t (см. работы [5-15] и [5-22])

$$\sum_{i=1}^{n+1} N_{i,k}(t) \equiv 1. \quad (5-85)$$

Каждая базисная функция положительна или равна нулю для всех значений параметра, т. е. $N_{i,k} \geq 0$.

Кроме $k=1$, все базисные функции имеют ровно один максимум.

Максимальный порядок кривой равен количеству вершин определяющего многоугольника.

Кривая обладает свойством уменьшения вариации. Кривая пересекает любую прямую не чаще, чем ее определяющий многоугольник.

¹ Отметим, что в отличие от кривых Безье вершины определяющего многоугольника нумеруются от 1 до $n+1$.

Общая форма кривой повторяет форму определяющего многоугольника. Чтобы применить к кривой любое аффинное преобразование, необходимо применить его к вершинам определяющего многоугольника.

Кривая лежит внутри выпуклой оболочки определяющего многоугольника.

Последнее свойство В-сплайна сильнее, чем у кривых Безье. У В-сплайна порядка k (степени $k - 1$) точки кривой лежат внутри выпуклой оболочки k соседних точек. Таким образом, все точки на В-сплайне должны лежать внутри объединения всех выпуклых оболочек k последовательных вершин. На рис. 5-32 приводится иллюстрация для различных значений k , причем выпуклые оболочки выделены серым цветом. В частности, при $k = 2$ выпуклая оболочка совпадает с многоугольником, т. е. В-сплайн — это сам многоугольник.

С помощью свойства выпуклой оболочки легко показать, что если все точки многоугольника коллинеарны, то соответствующий В-сплайн — прямая линия для всех k . Далее, если в неколлинеарном определяющем многоугольнике встречаются l коллинеарных вершин, то прямые участки кривой (если они есть) начинаются и кончаются по крайней мере за $k - 2$ отрезка от начала и конца серии коллинеарных вершин. Если последовательность коллинеарных вершин полностью лежит внутри неколлинеарного многоугольника, число коллинеарных участков кривой не меньше, чем $l - 2k + 3$. Если же эта последовательность находится на конце неколлинеарного многоугольника, то число коллинеарных участков кривой не меньше $l - k + 1$. Иллюстрация приведена на рис. 5-33.

Если имеется $k - 1$ совпадающих вершин, т. е. $B_i = B_{i+1} = \dots = B_{i+k-2}$, то выпуклая оболочка вершин от B_i до B_{i+k-2} — это сама вершина. Отсюда следует, что В-сплайн должен проходить через вершину B_i . На рис. 5-34 изображен пример такой точки для $k = 3$. Далее, так как В-сплайн везде C^{k-2} непрерывен, он также C^{k-2} непрерывен в B_i .

Наконец, заметим, что свойство непрерывности плавно переводит В-сплайн во вложенные отрезки прямой, как показано на рис. 5-35.

Уравнения (5-84) указывают, что выбор узлового вектора оказывает существенное влияние на базисные функции В-сплайна $N_{i,k}(t)$ и, следовательно, на сам В-сплайн. Единственное требование к узловому вектору: $x_i \leq x_{i+1}$, т. е. это монотонно возрастающая последовательность вещественных чисел. Обычно используются три типа узловых векторов: равномерные, открытые равномерные (или открытые) и неравномерные.

Отдельные узловые значения равномерного узлового вектора распределены на одинаковом расстоянии, например

$$[0 \ 1 \ 2 \ 3 \ 4]$$

$$[-0.2 \ -0.1 \ 0 \ 0.1 \ 0.2]$$

В частности, равномерные узловые векторы обычно начинаются в нуле и увеличиваются на 1 к некоторому максимальному значению или нормируются в диапазоне между 0 и 1 равными десятичными значениями, например,

$$[0 \ 0.25 \ 0.5 \ 0.75 \ 1.0].$$

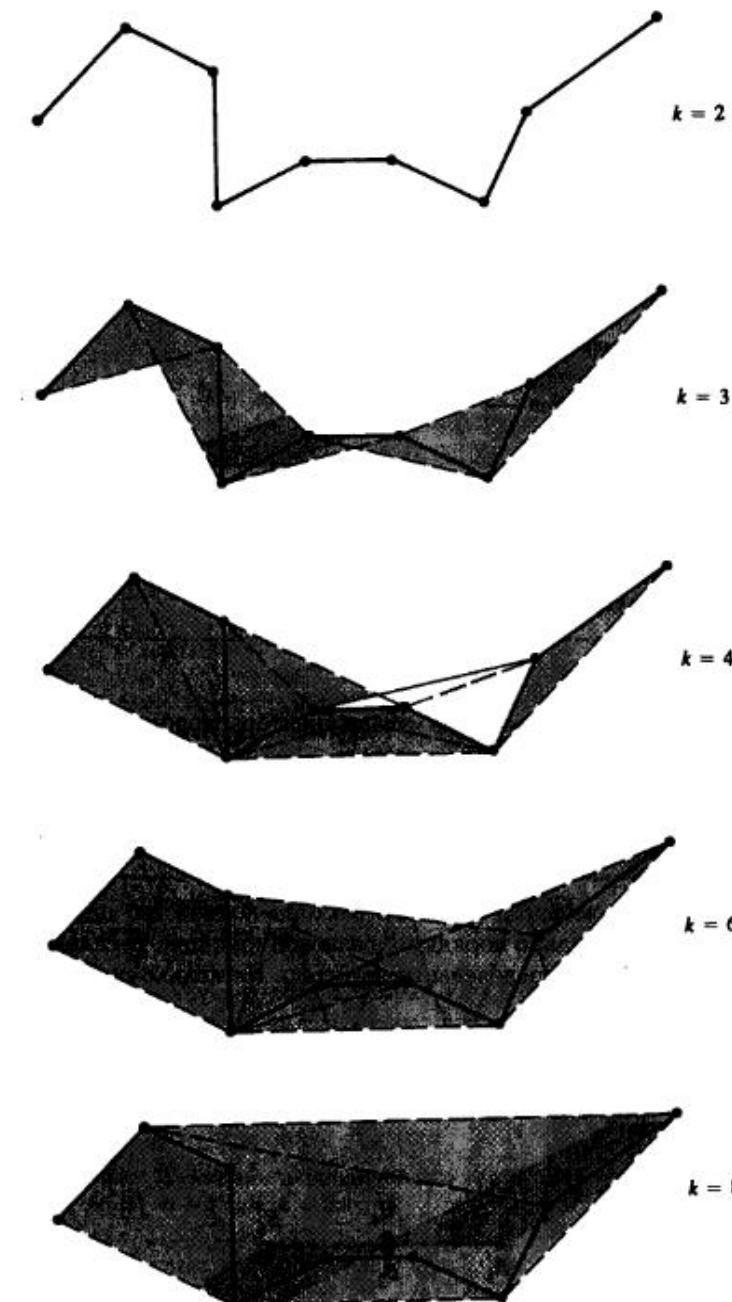
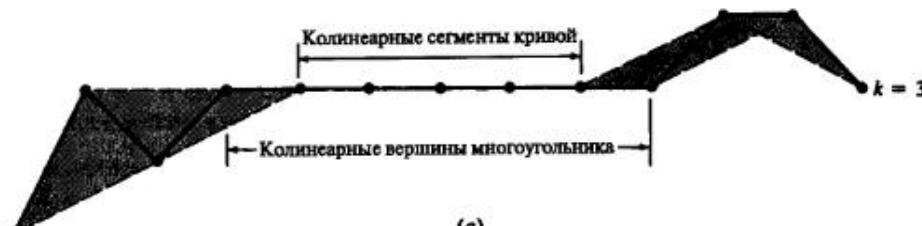


Рис. 5-32 Свойства выпуклой оболочки В-сплайнов.



(a)



(b)

Рис. 5-33 Свойства выпуклой оболочки В-сплайнов для коллинеарных сегментов кривой. (a) Внутренние вершины определяющего многоугольника; (b) вершины в конце определяющего многоугольника.

Для данного порядка k равномерные узловые векторы порождают периодические равномерные функции базиса, для которых

$$N_{i,k}(t) = N_{i-1,k}(t-1) = N_{i+1,k}(t+1).$$

То есть каждая функция базиса – это параллельный перенос другой функции, см. рис. 5-36.

У открытого равномерного узлового вектора количество одинаковых узловых значений в концах равно порядку k базисной функции В-сплайна. Внутренние узловые значения распределены равномерно. Несколько примеров с целыми

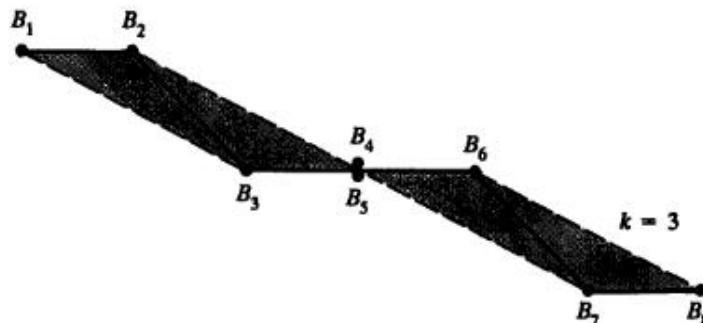


Рис. 5-34 Выпуклая оболочка для кратных вершин, $k = 3$.

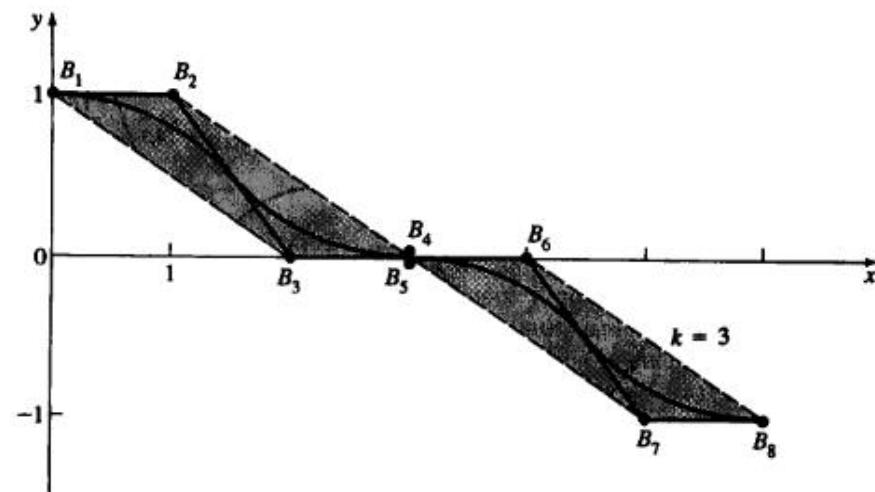


Рис. 5-35 Плавное (C^{k-2}) превращение в отрезки прямой.

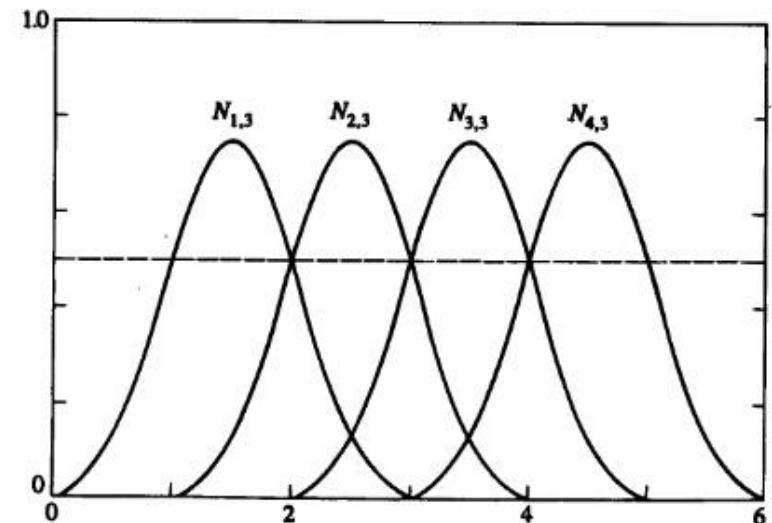


Рис. 5-36 Базисные функции периодического равномерного В-сплайна, $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6]$, $n + 1 = 4$, $k = 3$.

приращениями:

$$k = 2 \quad [0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 4],$$

$$k = 3 \quad [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3],$$

$$k = 4 \quad [0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2 \ 2]$$

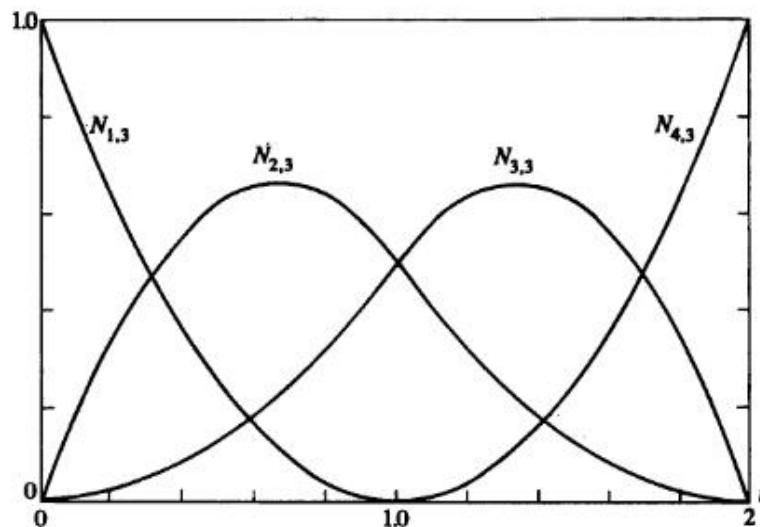


Рис. 5-37 Базисные функции открытого равномерного В-сплайна, $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$, $k = 3$, $n + 1 = 4$.

или для нормализованных приращений

$$\begin{aligned}k &= 2 \quad [0 \ 0 \ 1/4 \ 1/2 \ 3/4 \ 1 \ 1], \\k &= 3 \quad [0 \ 0 \ 0 \ 1/3 \ 2/3 \ 1 \ 1 \ 1], \\k &= 4 \quad [0 \ 0 \ 0 \ 0 \ 1/2 \ 1 \ 1 \ 1 \ 1].\end{aligned}$$

Формально открытый равномерный узловый вектор определяется как

$$\begin{aligned}x_i &= 0 \quad 1 \leq i \leq k \\x_i &= i - k \quad k + 1 \leq i \leq n + 1 \\x_i &= n - k + 2 \quad n + 2 \leq i \leq n + k + 1\end{aligned}$$

Получающиеся базисные функции ведут себя примерно так же, как и кривые Безье. Фактически, если количество вершин многоугольника равно порядку базиса В-сплайна и используется открытый равномерный узловый вектор, базис В-сплайна сводится к базису Бернштейна. Отсюда В-сплайн является кривой Безье. В этом случае узловой вектор — это просто k нулей, за которыми следует k единиц. Например, для четырех вершин открытый равномерный узловый вектор:

$$[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1].$$

В результате мы имеем кубическую кривую Безье — В-сплайн. Соответствующие базисные функции изображены на рис. 5-27б. На рис. 5-37 приведен еще один пример открытых базисных функций.

Неравномерные узловые векторы отличаются тем, что их внутренние узловые величины располагаются на разном расстоянии друг от друга и/или совмещаются.

Векторы могут быть периодическими или открытыми, например

$$[0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 2 \ 2],$$

$$[0 \ 1 \ 2 \ 2 \ 3 \ 4],$$

$$[0 \ 0.28 \ 0.5 \ 0.72 \ 1].$$

На рис. 5.38б-е показаны примеры неравномерных базисных функций В-сплайна порядка $k = 3$. У соответствующих узловых векторов на концах находится по k совмещенных одинаковых значений. Для сравнения на рис. 5-38а приведены базисные функции для открытого равномерного вектора. Отметим, что на рис. 5-38а и б функции симметричны, а также что у неравномерных базисов симметрия нарушается: 5-38с-е. Кроме того, при совмещенных узловых значениях у одной из функций появляется излом. На рис. 5-38д и е видно, что положение излома зависит от расположения совмещенного значения в узловом векторе.

Формула Кокса-де Бура (5-84) для расчета базисных функций В-сплайна recursive, поэтому функция порядка k зависит от базисных функций более низкого порядка вплоть до 1. Пусть дана базисная функция $N_{i,k}$. Тогда эту зависимость можно выразить в виде треугольника

$$\begin{array}{ccccccc} & & N_{i,k} & & & & \\ & N_{i,k-1} & & N_{i+1,k-1} & & & \\ & N_{i,k-1} & & N_{i+1,k-2} & & N_{i+2,k-2} & \\ & & & & & & \\ & & & & & & \\ & N_{i,1} & & N_{i+1,1} & & N_{i+2,1} & N_{i+3,1} \dots N_{i+k-1,1} \end{array}$$

Тогда обратная зависимость, т. е. влияние одной базисной функции первого порядка $N_{i,1}$ на функции более высоких порядков, такова:

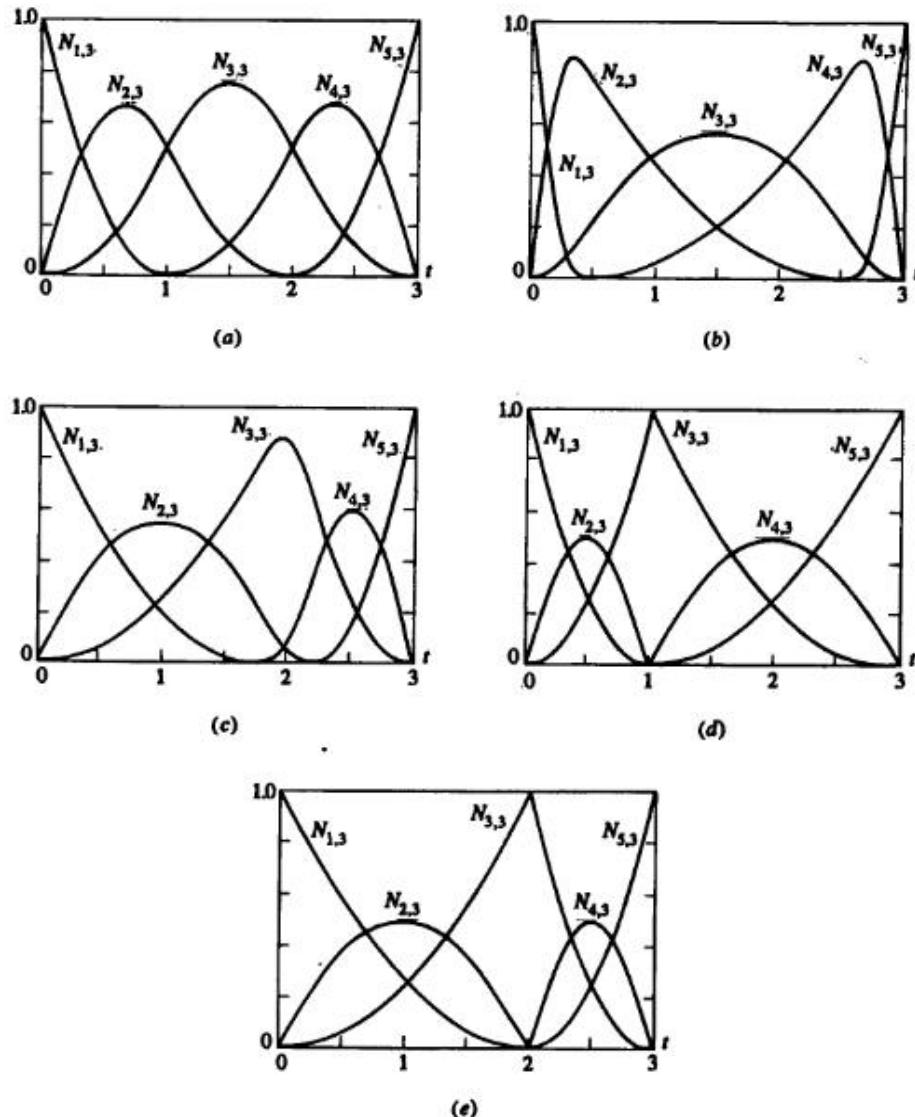
$$\begin{array}{ccccccc} & N_{i-k+1,k} & \dots & N_{i+k-1,k} & N_{i,k} & N_{i+1,k} & \dots & N_{i+k-1,k} \\ & & & & & & & \\ & N_{i-1,2} & & N_{i,2} & N_{i+1,2} & & & \\ & & & & N_{i,1} & & & \end{array}$$

Рассмотрим пример расчета базисных функций.

Пример 5-9 Расчет периодических базисных функций

Найти четыре базисные функции $N_{i,3}(t)$, $i = 1, 2, 3, 4$, третьего порядка ($k = 3$). Число $n + 1$ в данном случае равно 4. Зависимость базисных функций для $N_{i,3}$ изображается следующей диаграммой:

$$\begin{array}{cccccc} N_{1,3} & N_{2,3} & N_{3,3} & N_{4,3} & & \\ N_{1,2} & N_{2,2} & N_{3,2} & N_{4,2} & N_{5,2} & \\ N_{1,1} & N_{2,1} & N_{3,1} & N_{4,1} & N_{5,1} & N_{6,1} \end{array}$$

Рис. 5-38 Функции неравномерного базиса для $n + 1 = 5, k = 3$.

- (a) $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$;
- (b) $[X] = [0 \ 0 \ 0 \ .4 \ 2.6 \ 3 \ 3 \ 3]$;
- (c) $[X] = [0 \ 0 \ 0 \ 1.8 \ 2.2 \ 3 \ 3 \ 3]$;
- (d) $[X] = [0 \ 0 \ 0 \ 1 \ 1 \ 3 \ 3 \ 3]$;
- (e) $[X] = [0 \ 0 \ 0 \ 2 \ 2 \ 3 \ 3 \ 3]$.

Обратные зависимости для $i \geq 1$:

$$\begin{array}{ccccccc} N_{1,3} & N_{2,3} & N_{3,3} & N_{4,3} & N_{5,3} & N_{6,3} \\ N_{1,2} & N_{2,2} & N_{3,2} & N_{4,2} & N_{5,2} & \\ N_{1,1} & N_{2,1} & N_{3,1} & N_{4,1} & & \end{array}$$

Каков диапазон узлового вектора, необходимый для этого расчета? Из уравнения (5-84) следует, что для вычисления $N_{6,1}$ необходимы узловые величины x_6 и x_7 , а для $N_{1,1} - x_1$ и x_2 , т. е. необходимы значения от 0 до $n+k$. Отсюда количество узловых значений равно $n+k+1$. Узловой вектор для заданных периодических функций:

$$[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6],$$

где $x_1 = 0, \dots, x_7 = 6$. Диапазон параметра $0 \leq t \leq 6$. Используя уравнение (5-84) и приведенные выше диаграммы, получим базисные функции для различных значений параметра:

$$0 \leq t < 1$$

$$N_{1,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 1$$

$$N_{1,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 1$$

$$N_{1,3}(t) = \frac{t^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 1$$

$$1 \leq t < 2$$

$$N_{2,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 2$$

$$N_{1,2}(t) = (2-t); \quad N_{2,2}(t) = (t-1); \quad N_{i,2}(t) = 0, \quad i \neq 1, 2$$

$$N_{1,3}(t) = \frac{t}{2}(2-t) + \left(\frac{3-t}{2}\right)(t-1);$$

$$N_{2,3}(t) = \frac{(t-1)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 1, 2, 3$$

$$2 \leq t < 3$$

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 3$$

$$N_{2,2}(t) = (3-t); \quad N_{3,2}(t) = (t-2); \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$N_{1,3}(t) = \frac{(3-t)^2}{2};$$

$$N_{2,3}(t) = \frac{(t-1)(3-t)}{2} + \frac{(4-t)(t-2)}{2};$$

$$N_{3,3}(t) = \frac{(t-2)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 1, 2, 3$$

$$3 \leq t < 4$$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

$$N_{3,2}(t) = (4-t); \quad N_{4,2}(t) = (t-3); \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N_{2,3}(t) = \frac{(4-t)^2}{2}; \quad N_{3,3}(t) = \frac{(t-2)(4-t)}{2} + \frac{(5-t)(t-3)}{2};$$

$$N_{4,3}(t) = \frac{(t-3)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 2, 3, 4$$

$4 \leq t < 5$

$N_{5,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 5$

$N_{4,2}(t) = (5-t); \quad N_{5,2}(t) = (t-4); \quad N_{i,2}(t) = 0, \quad i \neq 4, 5$

$N_{3,3}(t) = \frac{(5-t)^2}{2};$

$N_{4,3}(t) = \frac{(t-3)(5-t)}{2} + \frac{(6-t)(t-4)}{2};$

$N_{i,3}(t) = 0, \quad i \neq 3, 4$

 $5 \leq t < 6$

$N_{6,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 6$

$N_{5,2}(t) = (6-t); \quad N_{i,2}(t) = 0, \quad i \neq 5$

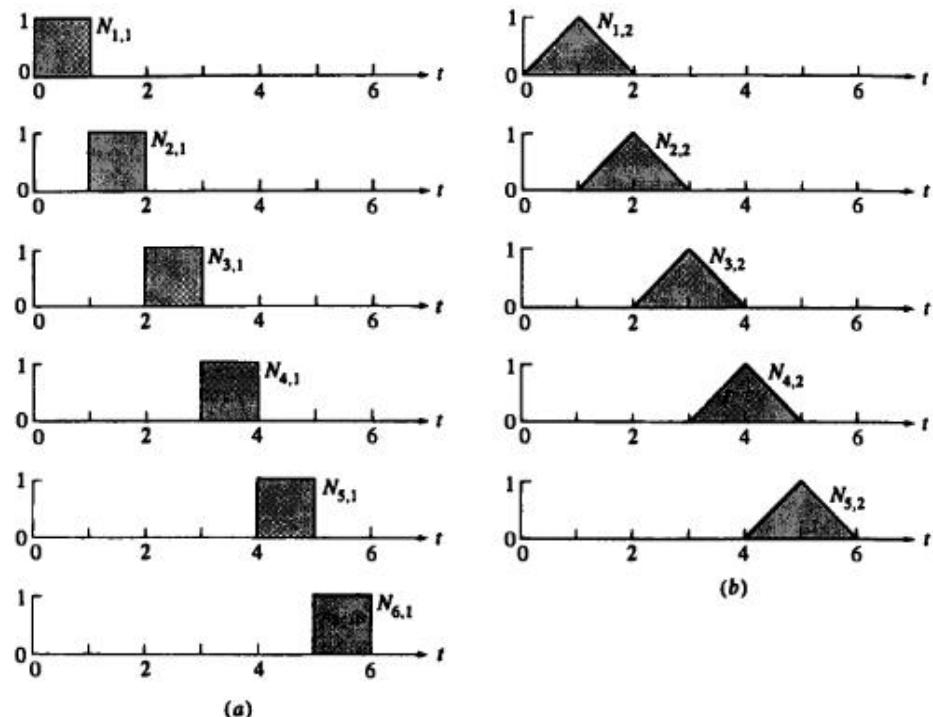
$N_{4,3}(t) = \frac{(6-t)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 4.$

Знак $<$ в определении $N_{i,1}$ приводит к тому, что при $t = 6$ все функции равны 0.

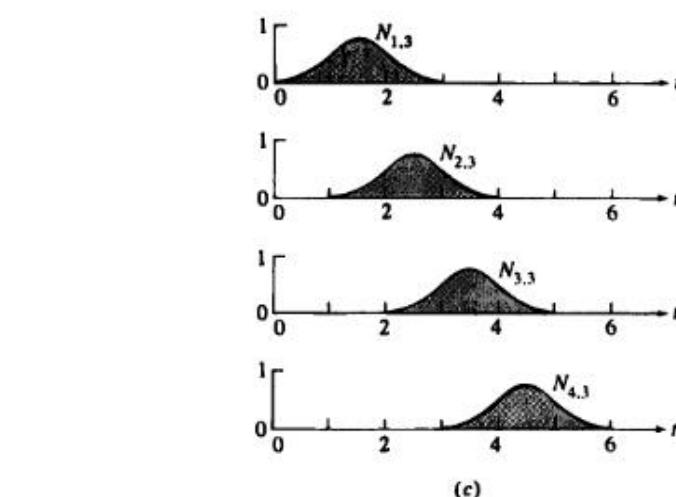
Результаты показаны на рис. 5-36 и 5-39c. Отметим, что каждая из базисных функций является кусочной параболической (квадратичной) кривой. Параболические сегменты на интервалах $x_i \rightarrow x_{i+1}$, $x_{i+1} \rightarrow x_{i+2}$, $x_{i+2} \rightarrow x_{i+3}$ объединяются и составляют базисные функции $N_{i,3}$. Каждая функция представляет собой параллельный перенос другой.

Пример 5-9 показывает, как построить базис N по функциям базиса более низкого порядка. На рис. 5-39a изображены функции первого порядка из примера 5-9, на рис. 5-39b — второго порядка, и на рис. 5-39c — третьего порядка. Обратим внимание на то, как растягивается диапазон ненулевых значений функций с увеличением их порядка. Говорят, что функция базиса обеспечивает поддержку на интервале от x_i до x_{i+k} .

Внимательно рассматривая рис. 5-36, можно заметить важное свойство функций равномерного базиса. Из уравнения (5-85) известно, что $\sum N_{i,k}(t) = 1$ для любого значения параметра t . Отсюда следует, что все множество периодических базисных функций для $k = 3$ определено только в диапазоне $2 \leq t \leq 4$. За его границами $\sum N_{i,k}(t) \neq 1$. Для равномерного начинающегося с 0 узлового вектора с целыми приращениями пригодный диапазон параметра: $k - 1 \leq t \leq (n + k) - (k - 1) = n + 1$. Для более общих или нормализованных векторов сокращение диапазона параметра соответствует потере $k - 1$ интервала узловых значений на каждом конце вектора.



(a)



(b)

Рис. 5-39 Построение периодических базисных функций $n+1 = 4$. (a) $k = 1$; (b) $k = 2$; (c) $k = 3$.

Пример 5-10 Расчет открытого равномерного базиса

Найти четыре ($n = 3$) базисные функции $N_{i,3}(t)$, $i = 1, 2, 3, 4$, третьего порядка ($k = 3$).

Напомним, что формально открытый узловой вектор с целыми интервалами между внутренними узловыми значениями определяется как

$$\begin{aligned}x_i &= 0, \quad 1 \leq i \leq k, \\x_i &= i - k, \quad k + 1 \leq i \leq n + 1, \\x_i &= n - k + 2, \quad n + 2 \leq i \leq n + k + 1.\end{aligned}$$

Диапазон изменения параметра $0 \leq t \leq n - k + 2$, т. е. от 0 до максимального узлового значения. Как и в примере 5-9, количество узловых значений равно $n + k + 1$. Если брать узловые значения, то вектор для данного примера примет вид

$$[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2],$$

где $x_1 = 0, \dots, x_7 = 2$. Параметр t изменяется от 0 до 2.

Пользуясь уравнениями (5-84) и диаграммами зависимости, получаем функции базиса для различных диапазонов параметра:

$$0 \leq t < 1$$

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 3$$

$$N_{2,2}(t) = 1 - t; \quad N_{3,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$N_{1,3}(t) = (1 - t)^2; \quad N_{2,3}(t) = t(1 - t) + \frac{(2 - t)}{2}t;$$

$$N_{3,3}(t) = \frac{t^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 1, 2, 3$$

$$1 \leq t < 2$$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

$$N_{3,2}(t) = (2 - t); \quad N_{4,2}(t) = (t - 1); \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N_{2,3}(t) = \frac{(2 - t)^2}{2}; \quad N_{3,3}(t) = \frac{t(2 - t)}{2} + (2 - t)(t - 1);$$

$$N_{4,3}(t) = (t - 1)^2; \quad N_{i,3}(t) = 0, \quad i \neq 2, 3, 4.$$

Эти результаты приведены на рис. 5-40.

Сравнивая результаты примера 5-10 (рис. 5-40) и 5-9 (рис. 5-39), мы видим, что они существенно различаются для периодического и открытого равномерного узловых векторов. В частности, отметим, что у открытых равномерных узловых векторов на всем диапазоне изменения параметра определен полный набор базисных функций; т. е. $\sum N_{i,k}(t) = 1$ для всех $0 \leq t \leq n - k + 2$. У периодического вектора диапазон параметра уменьшается.

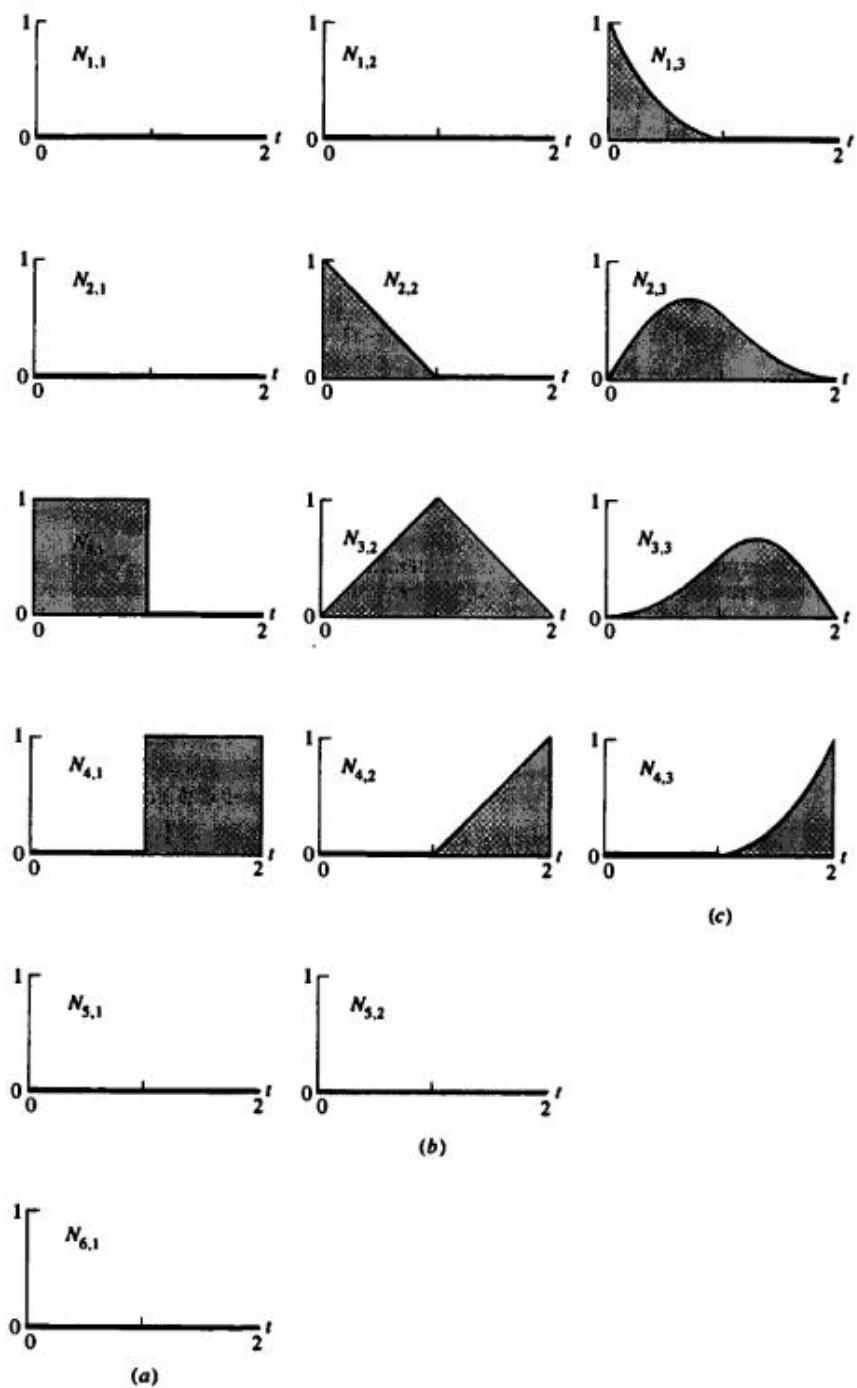


Рис. 5-40 Построение открытых базисных функций $n + 1 = 4$. (a) $k = 1$; (b) $k = 2$; (c) $k = 3$.

Пример 5-11 Расчет неравномерных базисных функций

Найти пять ($n+1 = 5$) базисных функций $N_{i,3}(t)$, $i = 1, 2, 3, 4, 5$, третьего порядка ($k = 3$) с узловым вектором $[X] = [0 \ 0 \ 0 \ 1 \ 1 \ 3 \ 3 \ 3]$, содержащим повторяющееся внутреннее узловое значение. Из уравнений (5-84) и диаграмм зависимости

$$0 \leq t < 1$$

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 2$$

$$N_{2,2}(t) = 1 - t; \quad N_{3,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$N_{1,3}(t) = (1-t)^2; \quad N_{2,3}(t) = t(1-t) + (1-t)t = 2t(1-t);$$

$$N_{3,3}(t) = t^2; \quad N_{i,3}(t) = 0, \quad i \neq 1, 2, 3$$

$$1 \leq t < 3$$

$$N_{i,1}(t) = 0, \quad \text{все } i$$

$$N_{i,2}(t) = 0, \quad \text{все } i$$

$$N_{i,3}(t) = 0, \quad \text{все } i.$$

В частности, как следствие повторения узлового значения, $N_{4,1}(t) = 0$ для всех t .

$$1 \leq t < 3$$

$$N_{5,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 5$$

$$N_{4,2}(t) = \frac{(3-t)}{2}; \quad N_{5,2}(t) = \frac{(t-1)}{2}; \quad N_{i,2}(t) = 0, \quad i \neq 4, 5$$

$$N_{3,3}(t) = \frac{(3-t)^2}{4};$$

$$N_{4,3}(t) = \frac{(t-1)(3-t)}{4} + \frac{(3-t)(t-1)}{4} = \frac{(3-t)(t-1)}{2};$$

$$N_{5,3}(t) = \frac{(t-1)^2}{4}; \quad N_{i,3}(t) = 0, \quad i \neq 3, 4, 5.$$

Результат приведен на рис. 5-38d.

Заметим, что для всех значений t имеем $\sum N_{i,k}(t) = 1.0$. Например, для $0 \leq t < 1$,

$$\sum_{i=1}^5 N_{i,3}(t) = (1-t)^2 + 2t(1-t) + t^2 = 1 - 2t + t^2 + 2t - 2t^2 + t^2 = 1.$$

Аналогично для $1 \leq t < 3$

$$\begin{aligned} \sum_{i=1}^5 N_{i,3}(t) &= \frac{1}{4}[(3-t)^2 + 2(3-t)(t-1) + (t-1)^2] \\ &= \frac{1}{4}[9 - 6t + t^2 - 6 + 8t - 2t^2 + 1 - 2t + t^2] \\ &= \frac{4}{4} = 1 \end{aligned}$$

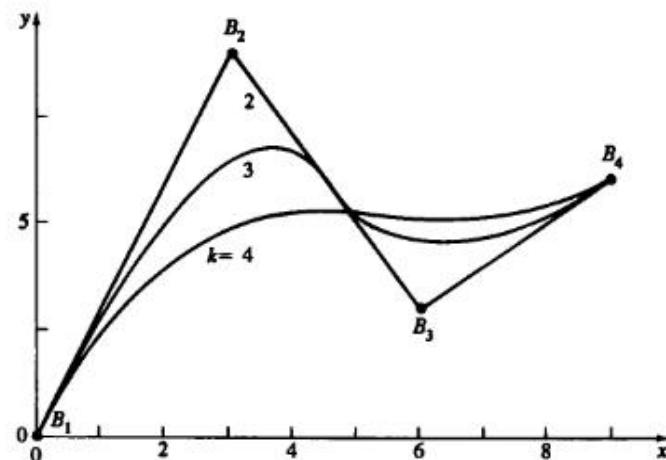


Рис. 5-41 Зависимость формы В-сплайна от его порядка.

Из вышесказанного видно, какое влияние оказывает выбор узлового вектора на вид базисных функций В-сплайна и, следовательно, на форму В-сплайна.

Гибкость базиса В-сплайна позволяет воздействовать на форму кривой различными способами:

Изменяя тип узлового вектора и базиса: периодический равномерный, открытый равномерный и неравномерный.

Меняя порядок k базисных функций.

Меняя количество и расположение вершин определяющего многоугольника.

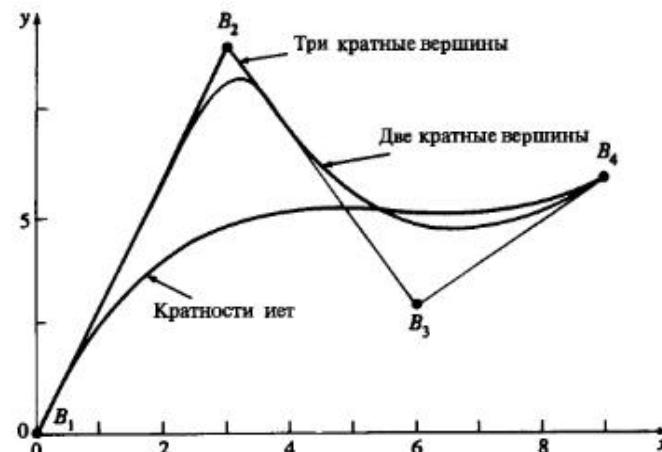
Используя повторяющиеся вершины.

Используя повторяющиеся узловые значения в узловых векторах.

Рассмотрим эти способы сначала для открытых В-сплайнов, затем для равномерных периодических и неравномерных В-сплайнов.

Открытый В-сплайн по своим свойствам во многом аналогичен кривым Безье. Как уже отмечалось, если порядок В-сплайна равен количеству вершин определяющего многоугольника, то базис В-сплайна сводится к базису Бернштейна, а сам В-сплайн становится кривой Безье. У открытого В-сплайна любого порядка ($k \geq 2$) первая и последняя точки кривой совпадают с соответствующими вершинами многоугольника. Далее, наклон кривой в первой и последней вершинах многоугольника равен наклону соответственных сторон многоугольника.

На рис. 5-41 изображены три открытых В-сплайна различного порядка, заданные одним набором из четырех вершин. Кривая четвертого порядка — это кривая Безье — один кубический полиномиальный сегмент. Кривая третьего порядка состоит из двух параболических сегментов, соединяющихся в центре второго отрезка с непрерывностью C^1 . Кривая второго порядка совпадает с определяющим многоугольником. Она состоит из трех линейных сегментов, соединяющихся во второй и третьей вершинах с непрерывностью C^0 . Угол наклона на концах, заданный наклоном сторон многоугольника, одинаков для всех трех кривых. Отметим

Рис. 5-42 Влияние кратности вершины на форму В-сплайна, $k = 4$.

также, что по мере возрастания порядка кривой, она все меньше напоминает исходный многоугольник и становится более гладкой.

На рис. 5-42 изображается влияние повторяющихся или совпадающих вершин. Все В-сплайны имеют порядок $k = 4$. Нижняя кривая определена четырьмя вершинами с узловым вектором $[0\ 0\ 0\ 1\ 1\ 1]$. У средней кривой пять определяющих вершин, причем две из них повторяются во второй вершине многоугольника [3 9]. Узловый вектор — $[0\ 0\ 0\ 1\ 2\ 2\ 2]$. Верхняя кривая определена шестью вершинами с тремя повторяющимися в точке [3 9]. Узловый вектор — $[0\ 0\ 0\ 1\ 2\ 3\ 3]$. Соответствующие многоугольники для трех кривых соответственно таковы: B_1, B_2, B_3, B_4 ; B_1, B_2, B_2, B_4 и B_1, B_2, B_2, B_3, B_4 .

Нижняя кривая состоит из единственного кубического сегмента. Средняя кривая состоит из двух сегментов, соединенных между B_2 и B_3 . Верхняя кривая состоит из трех сегментов: первый от B_1 до B_2 , второй от B_2 до середины между B_2 и B_3 , третий от этой точки до B_4 . Обратим внимание на то, что с увеличением кратности вершины B_2 кривая все ближе подходит к B_2 . Когда кратность достигается $k - 1$, возникает острый угол в соответствии со свойством выпуклой оболочки В-сплайнов. При внимательном изучении рис. 5-42 можно заметить, что на обеих сторонах кратной вершины имеется линейный участок.

Несмотря на наличие углов, сохраняется C^{k-2} дифференцируемость кривой. На первый взгляд это может показаться противоречием, однако излом определяется нулевым касательным вектором, что не исключает непрерывности его изменения. Возможность включения острых углов и изломов в непрерывно C^{k-2} дифференцируемые кривые это важное свойство В-сплайнов.

Наконец, заметим, что у всех кривых наклон в концах одинаков.

На рис. 5-43 показаны три В-сплайна четвертого порядка. Каждый определяющий многоугольник состоит из восьми вершин. Кривые отличаются тем, что точка B_5 передвигается в B'_5 и B''_5 . Перемещение точки B_5 воздействует на кривую только локально: изменяются лишь сегменты, отвечающие отрезкам B_3B_4 ,

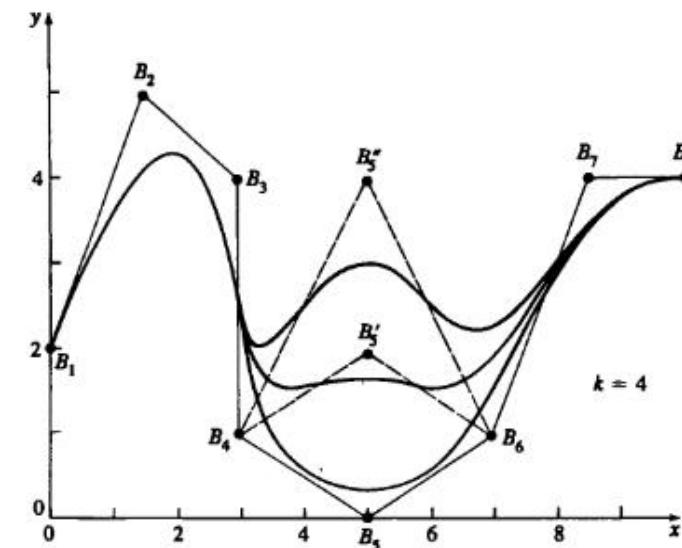


Рис. 5-43 Локальная коррекция В-сплайна.

B_4B_5 и B_5B_6 , B_6B_7 . В общем случае затрагиваются только $\pm k/2$ отрезков вокруг сдвинутой точки.

Для иллюстрации рассмотрим пример.

Пример 5-12 Расчет открытого В-сплайна

Рассмотрим многоугольник из примера 5-7: $B_1[1\ 1]$, $B_2[2\ 3]$, $B_3[4\ 3]$, $B_4[3\ 1]$. Найти В-сплайн второго и четвертого порядка. Для $k = 2$ открытый узловый вектор

$$[0\ 0\ 1\ 2\ 3\ 3],$$

где $x_1 = 0$, $x_2 = 0, \dots, x_6 = 3$. Параметр t изменяется от 0 до 3. Кривая состоит из трех линейных ($k - 1 = 1$) сегментов. Для $0 \leq t < 3$ функции базиса имеют вид:

$$0 \leq t < 1$$

$$N_{2,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 2$$

$$N_{1,2}(t) = 1 - t; \quad N_{2,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 1, 2$$

$$1 \leq t < 2$$

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 3$$

$$N_{2,2}(t) = 2 - t; \quad N_{3,2}(t) = (t - 1); \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$2 \leq t < 3$$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

$$N_{3,2}(t) = (3 - t); \quad N_{4,2}(t) = (t - 2); \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

Из уравнения (5-83) получаем параметрический В-сплайн

$$P(t) = B_1N_{1,2}(t) + B_2N_{2,2}(t) + B_3N_{3,2}(t) + B_4N_{4,2}(t).$$

Для каждого из этих интервалов

$$\begin{aligned} P(t) &= (1-t)B_1 + tB_2 = B_1 + (B_2 - B_1)t \quad 0 \leq t < 1, \\ P(t) &= (2-t)B_2 + (t-1)B_3 = B_2 + (B_3 - B_2)t \quad 1 \leq t < 2, \\ P(t) &= (3-t)B_3 + (t-2)B_4 = B_3 + (B_4 - B_3)t \quad 2 \leq t < 3. \end{aligned}$$

В каждом случае в результате получается уравнение прямой для ребра многоугольника, т. е. кривая совпадает с многоугольником.

Последняя точка кривой ($t = t_{\max} = 3$) требует особого внимания. Из-за того что интервал в уравнении (5-84а) открыт справа, все базисные функции N_{ik} при $t = 3$ равны нулю. Следовательно, последняя точка многоугольника формально не лежит на кривой. Фактически же это не так. Рассмотрим $t = 3 - \epsilon$, где ϵ — бесконечно малая величина. Если ϵ стремится к нулю, то в пределе последние точки кривой и многоугольника совпадают. На практике либо явно добавляют последнюю точку в описание кривой, либо определяют $N(t = t_{\max}) = 1.0$.

Для $k = 4$ порядок кривой совпадает с количеством вершин определяющего многоугольника, поэтому В-сплайн сводится к кривой Безье. Узловый вектор с $t_{\max} = n - k + 2 = 3 - 4 + 2 = 1 - [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$. Функции базиса таковы:

$$0 \leq t < 1$$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

$$N_{3,2}(t) = (1-t); \quad N_{4,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N_{2,3}(t) = (1-t)^2; \quad N_{3,3}(t) = 2t(1-t);$$

$$N_{4,3}(t) = t^2; \quad N_{i,3}(t) = 0, i \neq 2, 3, 4$$

$$N_{1,4}(t) = (1-t)^3; \quad N_{2,4}(t) = t(1-t)^2 + 2t(1-t)^2 = 3t(1-t)^2;$$

$$N_{3,4}(t) = 2t^2(1-t) + (1-t)t^2 = 3t^2(1-t); \quad N_{4,4}(t) = t^3$$

Из уравнения (5-83) получаем параметрический В-сплайн

$$P(t) = B_1 N_{1,4}(t) + B_2 N_{2,4}(t) + B_3 N_{3,4}(t) + B_4 N_{4,4}(t),$$

$$P(t) = (1-t)^3 B_1 + 3t(1-t)^2 B_2 + 3t^2(1-t) B_3 + t^3 B_4.$$

Итак, при $t = 0$

$$P(0) = B_1$$

и при $t = 1/2$

$$P\left(\frac{1}{2}\right) = \frac{1}{8}B_1 + \frac{3}{8}B_2 + \frac{3}{8}B_3 + \frac{1}{8}B_4$$

и

$$\begin{aligned} P\left(\frac{1}{2}\right) &= \frac{1}{8}[1 \ 1] + \frac{3}{8}[2 \ 3] + \frac{3}{8}[4 \ 3] + \frac{1}{8}[3 \ 1] \\ &= [11/4 \ 5/2]. \end{aligned}$$

Сравнивая с примером 5-7, видим, что результаты одинаковы. Получившаяся кривая изображена на рис. 5-28.

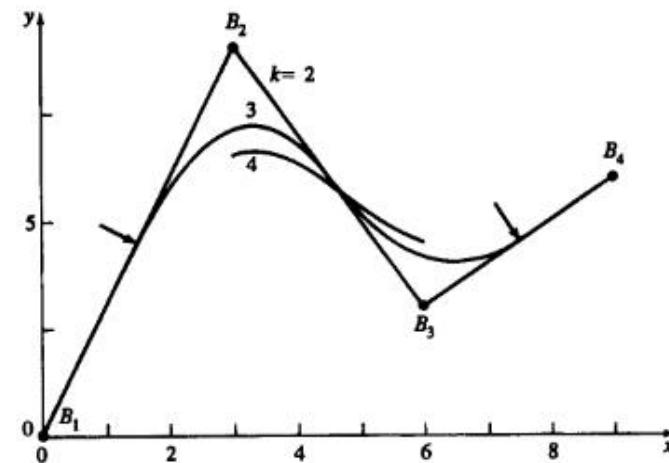


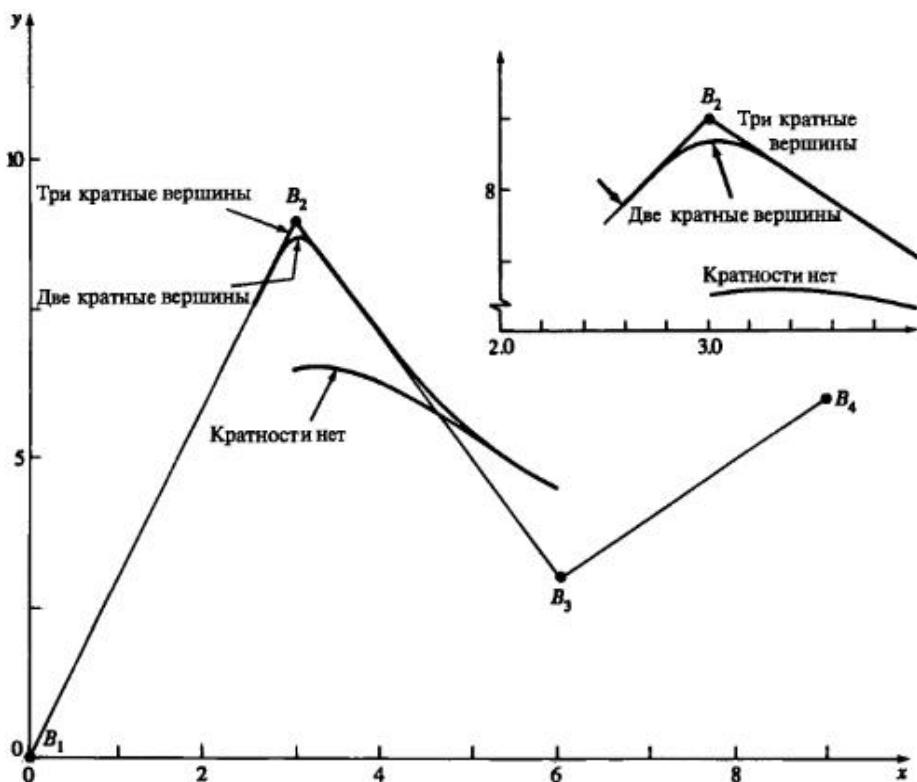
Рис. 5-44 Зависимость формы периодического В-сплайна от его порядка.

Теперь займемся периодическими В-сплайнами. На рис. 5-44 показаны три периодических В-сплайна разного порядка. Все кривые определены теми же вершинами, что и для открытых В-сплайнов на рис. 5-41. Для $k = 2$ В-сплайн опять совпадает с определяющим многоугольником. Отметим, однако, что у периодического сплайна при $k > 2$ первая и последняя точки на кривой не совпадают с первой и последней точками многоугольника. Наклон в первой и последней точках также может отличаться от наклона соответствующих сторон многоугольника. Для $k = 3$ В-сплайн начинается в середине первого ребра и оканчивается в середине последнего, как отмечено стрелками. Это происходит из-за сокращения диапазона параметра для базисных функций периодического В-сплайна. Для $k = 2$ периодический узловый вектор $-[0 \ 1 \ 2 \ 3 \ 4 \ 5]$ с диапазоном параметра $1 \leq t \leq 4$. Для $k = 3$ периодический узловый вектор $-[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6]$ с диапазоном параметра $2 \leq t \leq 4$. Для $k = 4$ периодический узловый вектор $-[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$ с диапазоном параметра $3 \leq t \leq 4$.

Сравнение полученных результатов и результатов для открытых узловых векторов на рис. 5-41 показывает, что кривую можно определить на полном диапазоне параметра, задавая кратные узловые значения на концах векторов. При этом кривая растягивается к концам многоугольника.

В данном случае кривая четвертого порядка опять состоит из единственного кубического сегмента; кривая третьего порядка — из двух параболических сегментов, соединенных в середине второго ребра с непрерывностью C^1 ; кривая второго порядка — из трех линейных сегментов, соединенных во второй и третьей вершинах с C^0 непрерывностью. Увеличение порядка опять сглаживает кривую, но в то же время и укорачивает ее.

Рис. 5-45 показывает, что воздействие кратных вершин одинаково для периодических и открытых В-сплайнов. Область вокруг точки изображена более подробно.

Рис. 5-45 Влияние кратности вершины на форму периодического В-сплайна, $k = 4$.**Пример 5-13 Расчет периодического В-сплайна**

Снова рассмотрим многоугольник на рис. 5-44. Вершины многоугольника — $B_1[0 \ 0]$, $B_2[3 \ 9]$, $B_3[6 \ 3]$, $B_4[9 \ 6]$. Найти периодический В-сплайн четвертого порядка ($k = 4$), заданный этим многоугольником.

Для $k = 4$ на диапазоне $3 \leq t < 4$ узловой вектор для периодических базисных функций — $[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$. Для этого диапазона базисные функции первого порядка (см. уравнение (5-84a), $(3 \leq t < 4)$:

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

Из уравнения (5-84b) получаем базисные функции высших порядков:

$$N_{3,2}(t) = (4-t); \quad N_{4,2}(t) = (t-3); \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N_{2,3}(t) = \frac{(4-t)^2}{2}; \quad N_{3,3}(t) = \frac{(t-2)(4-t)}{2} + \frac{(5-t)(t-3)}{2};$$

$$N_{4,3}(t) = \frac{(t-3)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 2, 3, 4$$

$$N_{1,4}(t) = \frac{(4-t)^3}{6};$$

$$N_{2,4}(t) = \frac{(t-1)(4-t)^2}{6} + \frac{(5-t)(4-t)(t-2)}{6} + \frac{(5-t)^2(t-3)}{6};$$

$$N_{3,4}(t) = \frac{(t-2)^2(4-t)}{6} + \frac{(t-2)(t-3)(5-t)}{6} + \frac{(6-t)(t-3)^2}{6};$$

$$N_{4,4}(t) = \frac{(t-3)^3}{6}$$

При $t = 3$

$$N_{1,4}(3) = \frac{(4-3)^2}{6} = \frac{1}{6}$$

$$N_{2,4}(3) = \frac{(3-1)(4-3)^2}{6} + \frac{(5-3)(4-3)(3-2)}{6} + \frac{(5-3)^2(3-3)}{6} = \frac{2}{3}$$

$$N_{3,4}(3) = \frac{(3-2)^2(4-3)}{6} + \frac{(3-2)(3-3)(5-3)}{6} + \frac{(6-3)(3-3)^2}{6} = \frac{1}{6}$$

$$N_{4,4}(3) = \frac{(3-3)^3}{6} = 0$$

Точка на В-сплайне при $t = 3$ имеет вид

$$\begin{aligned} P(3) &= \frac{1}{6}B_1 + \frac{2}{3}B_2 + 0B_4 = \\ &= \frac{1}{6}[0 \ 0] + \frac{2}{3}[3 \ 9] + \frac{1}{6}[6 \ 3] + 0[9 \ 6] = \\ &= [3 \ 6.5] \end{aligned}$$

Полная кривая приведена на рис. 5-44.

Периодические В-сплайны очень удобны для построения замкнутых кривых. На рис. 5-46а изображен периодический В-сплайн четвертого порядка ($k = 4$), построенный для замкнутого многоугольника $B_1B_2B_3B_4B_5B_6B_7B_8B_1$.

Первая вершина совпадает с последней. В-сплайн получается незамкнутым из-за ограниченного диапазона параметра. Здесь периодический равномерный узловый вектор: $[0 \ 1 \ 2 \dots \ 10 \ 11 \ 12]$ с диапазоном параметра $3 \leq t \leq 9$.

Повторяя всего $k-2$ вершин в начале и/или конце замкнутого многоугольника, получаем замкнутый периодический В-сплайн. (Далее в этом разделе обсуждается другой метод в матричной формулировке.) Результат показан на рис. 5-46б, где определяющий многоугольник — $B_8B_1B_2B_3B_4B_5B_6B_7B_8B_1B_2$ с периодическим равномерным узловым вектором $[0 \ 1 \ 2 \dots \ 12 \ 13 \ 14]$ и диапазоном параметра $3 \leq t \leq 11$. Альтернативные многоугольники $B_1B_2B_3B_4B_5B_6B_7B_8B_1B_2B_3$ или $B_7B_8B_1B_2B_3B_4B_5B_6B_7B_8B_1$ приводят к тому же самому результату.

На рис. 5-47 одна вершина многоугольника B_4 передвинута на другое место. Ее воздействие распространяется на сегменты кривой, соответствующие $\pm k/2$ ребрам многоугольника по обе стороны от сдвинутой точки. На рис. 5-48 показан эффект кратной вершины B_4 . Область вокруг B_4 изображена более подробно. Опять отметим, что ее влияние распространяется также только на $\pm k/2$ сегмента кривой по обе стороны кратной вершины.

Теперь рассмотрим неравномерные В-сплайны. На рис. 5-49 кривая изменяется под воздействием кратных внутренних узловых значений. Верхняя кри-

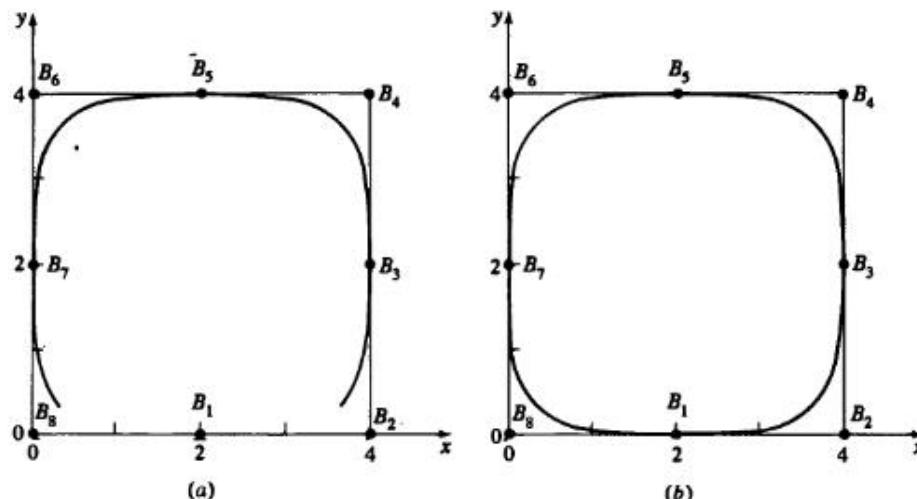


Рис. 5-46 Замкнутый периодический В-сплайн. (а) $B_1B_2B_3B_4B_5B_6B_7B_8B_1$ определяющий многоугольник; (б) $B_8B_1B_2B_3B_4B_5B_6B_7B_8B_1B_2$ – определяющий многоугольник.

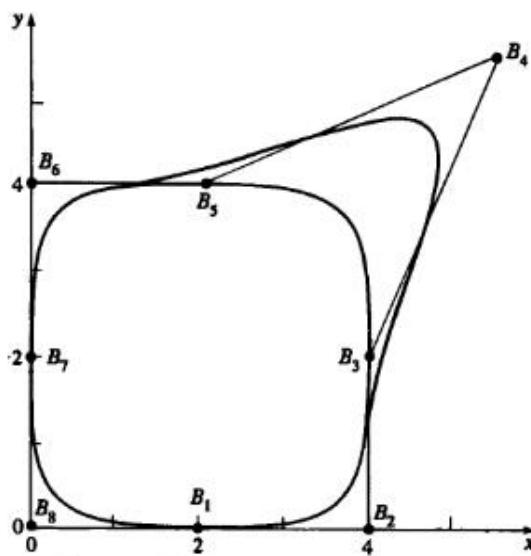


Рис. 5-47 Изменение формы В-сплайна после перемещения одной вершины многоугольника.

вия третьего порядка ($k = 3$) рассчитана для открытого узлового вектора $[0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$. Базисные функции для этой кривой изображены на рис. 5-38а. Нижняя кривая третьего порядка построена с неравномерным узловым вектором $[0 \ 0 \ 0 \ 1 \ 1 \ 3 \ 3 \ 3]$. Ее базисные функции – на

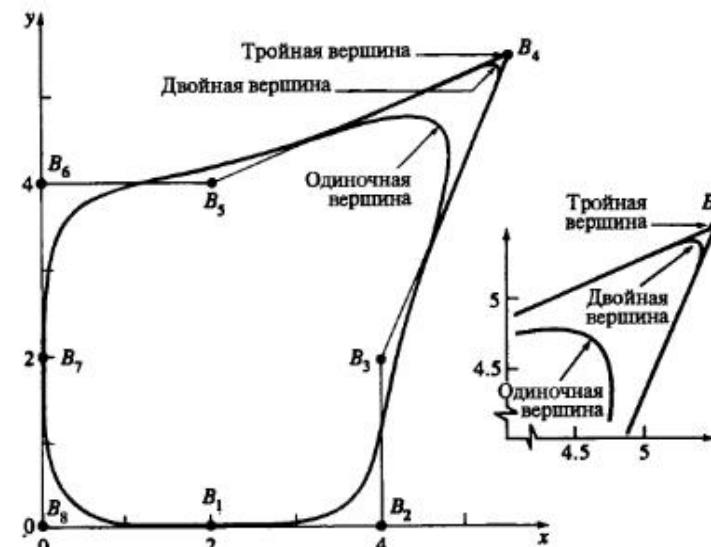


Рис. 5-48 Влияние кратных вершин на форму замкнутого периодического В-сплайна.

рис. 5-38д. Такая же кривая получается с неравномерным узловым вектором $[0 \ 0 \ 0 \ 2 \ 2 \ 3 \ 3 \ 3]$ и базисными функциями на рис. 5-38е.

Из рис. 5-49 видно, что кратные внутренние узловые значения порождают излом в вершине B_3 . Кратное значение порождает ребро кулевоей длины, поэтому уменьшается диапазон поддержки базисных функций. Далее, кратные внутренние узловые значения, в отличие от кратных вершин многоугольника, снижают дифференцируемость базисной функции в x_i до C^{k-m-1} , где $m \leq k - 1$ равно кратности внутреннего узлового значения. Локально неравномерная кривая на рис. 5-49 C^0 ($k - m - 1 = 3 - 2 - 1 = 0$) непрерывная в окрестности B_3 , что и приводит к появлению угла.

Открытые неравномерные В-сплайны третьего порядка ($k = 3$) на рис. 5-50 построены с помощью узлового вектора с внутренними значениями, пропорциональными длинам ребер между вершинами многоугольника. Узловой вектор имеет вид

$$x_i = 0 \quad 1 \leq i \leq k,$$

$$x_{i+k} = \left(\frac{\left(\frac{i}{n-k+2} \right) c_{i+1} + \sum_{j=1}^i c_j}{\sum_{i=1}^n c_i} \right) (n-k+2), \quad 1 \leq i \leq n-k+1,$$

$$x_i = n-k+2, \quad n+1 \leq i \leq n+k, \quad (5-86)$$

где $c_i = |B_{i+1} - B_i|$. Для вершин, находящихся на одинаковом расстоянии друг от друга, узловой вектор имеет целые равномерно распределенные значения, т.е. вектор является открытым и равномерным. В работе [5-25] предложен по-

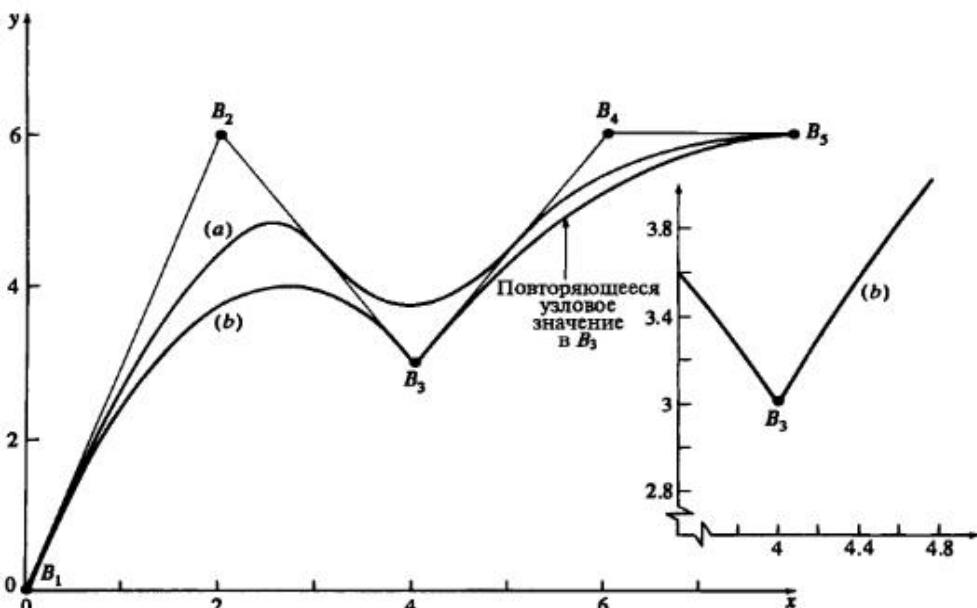


Рис. 5-49 Неравномерные В-сплайны, $k = 3$. (a) $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$; (b) $[X] = [0 \ 0 \ 0 \ 1 \ 1 \ 3 \ 3 \ 3]$.

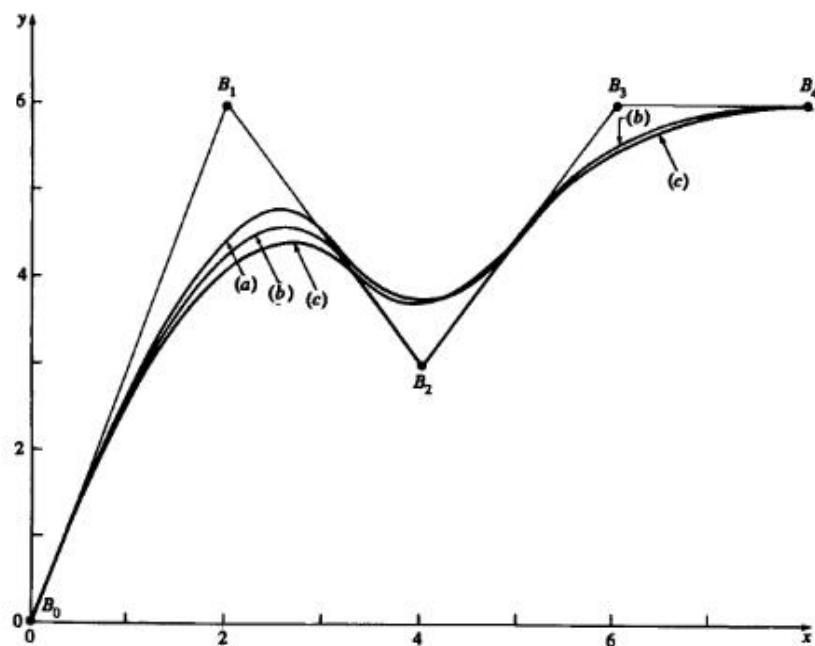


Рис. 5-50 Сравнение открытых неравномерных В-сплайнов: (a) равномерный узловый вектор; (b) неравномерный узловый вектор, пропорциональный длинам хорд; (c) неравномерный узловый вектор, пропорциональный длинам хорд, с двойной вершиной в B_2 .

хожий метод, позволяющий получить отдельные внутренние узловые значения. Для сравнения приведена кривая с открытым равномерным вектором, а также кривая с парой совпадающих вершин B_3 .

Отсюда следует, что неравномерные В-сплайны не сильно отличаются от равномерных при небольшом изменении относительного расстояния между вершинами. Рассмотрим пример.

Пример 5-14 Неравномерный В-сплайн

Найти открытый В-сплайн третьего порядка, заданный точками $B_1[0 \ 0]$, $B_2[2 \ 6]$, $B_3[4 \ 3]$, $B_4[6 \ 6]$, $B_5[8 \ 6]$ с помощью неравномерного вектора с узловыми значениями, пропорциональными длинам ребер многоугольника.

Сначала найдем длины хорд

$$c_1 = \sqrt{(2-0)^2 + (6-0)^2} = \sqrt{40} = 6.325$$

$$c_2 = \sqrt{(4-2)^2 + (3-6)^2} = \sqrt{13} = 3.606$$

$$c_3 = \sqrt{(6-4)^2 + (6-3)^2} = \sqrt{13} = 3.606$$

$$c_4 = \sqrt{(8-6)^2 + (6-6)^2} = \sqrt{4} = 2.0$$

Их суммарная длина

$$\sum_{i=1}^4 c_i = 15.537.$$

Из уравнения (5-86) получаем внутренние узловые значения

$$x_4 = \left[\frac{\frac{c_2}{3} + c_1}{15.537} \right] = 1.453,$$

$$x_5 = \left[\frac{\frac{2c_3}{3} + c_1 + c_2}{15.537} \right] (3) = 2.382.$$

Узловой вектор имеет вид

$$[x] = [0 \ 0 \ 0 \ 1.45 \ 2.382 \ 3 \ 3 \ 3]$$

где $x_1 = 0$, $x_2 = 0, \dots, x_8 = 3$. Диапазон параметра $0 \leq t \leq 3$. Кривая состоит из трех параболических ($k-1 = 2$) сегментов.

Функции базиса для $0 \leq t < 1.453$ таковы

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 3$$

$$N_{2,2}(t) = \frac{(1.453-t)}{1.453}; \quad N_{3,2}(t) = \frac{t}{1.453}; \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$N_{1,3}(t) = \frac{(1.453-t)^2}{(1.453)^2}; \quad N_{2,3}(t) = \frac{t}{(1.453)^2}(1.453-t) + \frac{(2.382-t)t}{(2.382)(1.453)}$$

$$N_{3,3}(t) = \frac{t^2}{(2.382)(1.453)}; \quad N_{i,3}(t) = 0, \quad i \neq 1, 2, 3$$

Функции базиса для $1.453 \leq t < 2.382$:

$$\begin{aligned} N_{4,1}(t) &= 1; \quad N_{i,1}(t) = 0, \quad i \neq 4 \\ N_{3,2}(t) &= \frac{(2.382 - t)}{(2.382 - 1.453)}; \\ N_{4,2}(t) &= \frac{(t - 1.453)}{(2.382)(1.453)}; \quad N_{i,2}(t) = 0, \quad i \neq 3, 4 \\ N_{2,3}(t) &= \frac{(2.382 - t)^2}{2.382(2.382 - 1.453)}; \\ N_{3,3}(t) &= \frac{t}{2.382} \frac{(2.382 - t)}{(2.382 - 1.453)} + \frac{(3-t)}{(3-1.453)} \frac{(t-1.453)}{(2.382-1.453)}; \\ N_{4,3}(t) &= \frac{(t-1.453)^2}{(3-1.453)(2.382-1.453)}; \quad N_{i,3}(t) = 0, \quad i \neq 2, 3, 4 \end{aligned}$$

Функции базиса для $2.382 \leq t < 3$:

$$\begin{aligned} N_{5,1}(t) &= 1; \quad N_{i,1}(t) = 0, \quad i \neq 5 \\ N_{4,2}(t) &= \frac{(3-t)}{(3-2.382)}; \\ N_{5,2}(t) &= \frac{(t-2.382)}{(3-2.382)}; \quad N_{i,2}(t) = 0, \quad i \neq 4, 5 \\ N_{3,3}(t) &= \frac{(3-t)^2}{(3-1.453)(3-2.382)}; \\ N_{4,3}(t) &= \frac{(t-1.453)(3-t)}{(3-1.453)(3-2.382)} + \frac{(3-t)(t-2.382)}{(3-2.382)^2}; \\ N_{5,3}(t) &= \frac{(t-2.382)^2}{(3-2.382)^2}; \quad N_{i,3}(t) = 0, \quad i \neq 3, 4, 5 \end{aligned}$$

Вспомним, что любая точка кривой задается уравнением:

$$P(t) = N_{1,3}(t)B_1 + N_{2,3}(t)B_2 + N_{3,3}(t)B_3 + N_{4,3}(t)B_4 + N_{5,3}(t)B_5$$

т.е. при $t = 1/2$

$$\begin{aligned} P\left(\frac{1}{2}\right) &= 0.430B_1 + 0.498B_2 + 0.072B_3 + (0)B_4 + (0)B_5 \\ &= 0.430[0 \ 0] + 0.498[2 \ 6] + 0.072[4 \ 3] \\ &= [1.284 \ 3.202] \end{aligned}$$

и при $t = 2$

$$\begin{aligned} P(2) &= (0)B_1 + 0.066B_2 + 0.726B_3 + 0.208B_4 + (0)B_5 \\ &= 0.066[2 \ 6] + 0.726[4 \ 3] + 0.208[6 \ 6] \\ &= [4.284 \ 3.822] \end{aligned}$$

Результаты приведены на рис. 5-50.

Уравнения В-сплайна можно записать в матричной форме, подобно кубическим сплайнам, параболической интерполяции и кривым Безье (см. уравнения (5-276), (5-44) и (5-67)). У периодических В-сплайнов матричная форма особенно проста.

Вспомним, что базисные функции периодического В-сплайна являются параллельным переносом друг друга (см. рис. 5-36) и что влияние каждой функции распространяется только на k интервале. Поэтому для целых узловых значений все базисные функции на единичном интервале $0 \leq t^* < 1$ имеют один и тот же вид $N_{i,k}^*(t^*)$, т. е. иногда удобно нормализовать параметризацию. Точка на параметрическом В-сплайте на единичном интервале имеет вид

$$P_j(t^*) = \sum_{i=0}^{k-1} N_{i+1,k}^*(t^*) B_{j+i}, \quad 1 \leq j \leq n-k+1, \quad 0 \leq t^* < 1, \quad (5-87)$$

где j — счетчик сегментов кривой, а n на единицу меньше количества вершин. Заметим, что уравнение (5-87) содержит только k членов:

$$P_j(t^*) = N_{1,k}^* B_j + N_{2,k}^* B_{j+1} + \cdots + N_{k,k}^* B_{j+k-1}.$$

Для $k = 3$ базисные функции на интервале параметра $0 \leq t^* < 1$

$$\begin{aligned} N_{1,3}^*(t^*) &= \frac{(1-t^*)^2}{2} \\ N_{2,3}^*(t^*) &= \frac{-2t^{*2} + 2t^* + 1}{2} \\ N_{3,3}^*(t^*) &= \frac{t^{*2}}{2} \end{aligned}$$

Тогда уравнение (5-87) принимает вид

$$\begin{aligned} 2P_j(t^*) &= (1-2t^*+t^{*2})B_j + (-2t^{*2}+2t^*+1)B_{j+1} + t^{*2}B_{j+2} = \\ &= t^{*2} \left(\begin{array}{ccc} B_j & -2B_{j+1} & B_{j+2} \end{array} \right) \\ &+ t^* \left(\begin{array}{ccc} -2B_j & 2B_{j+1} & 0 \cdot B_{j+2} \end{array} \right) \\ &+ \left(\begin{array}{ccc} B_j & B_{j+1} & 0 \cdot B_{j+2} \end{array} \right) \end{aligned}$$

Перепишем в матричной форме:

$$\begin{aligned} P_j^*(t^*) &= [T^*][N^*][G] = \\ &= \frac{1}{2}[t^{*2} \ t^* \ 1] \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} B_j \\ B_{j+1} \\ B_{j+2} \end{bmatrix} \end{aligned} \quad (5-88)$$

Аналогично, для $k = 4$ базисные функции на интервале параметра $0 \leq t^* < 1$ таковы:

$$N_{1,4}^*(t^*) = \frac{-t^{*3} + 3t^{*2} - 3t^* + 1}{6}$$

$$N_{2,4}^*(t^*) = \frac{3t^{*3} + 6t^{*2} + 4}{6}$$

$$N_{3,4}^*(t^*) = \frac{-t^{*3} + 3t^{*2} + 3t^* + 1}{6}$$

$$N_{4,4}^*(t^*) = \frac{t^{*3}}{6}$$

Матричная форма для $k = 4$ имеет вид

$$P_j(t^*) = [T^*][N^*][G] =$$

$$= \frac{1}{6} [t^{*3} \quad t^{*2} \quad t^* \quad 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} B_j \\ B_{j+1} \\ B_{j+2} \\ B_{j+3} \end{bmatrix} \quad (5-89)$$

Уравнения (5-87) и (5-89) содержат произведение весовой функции и геометрической матрицы (см. уравнения (5-27), (5-44), (5-67); т.е.

$$P_j(t^*) = [T^*][N^*][G] = [F][G] \quad 1 \leq j \leq n - k + 1, \quad 0 \leq t^* < 1 \quad (5-90)$$

где

$$[T^*] = [t^{*k-1} \quad t^{*k-2} \quad \dots \quad t^* \quad 1] \quad 0 \leq t^* < 1$$

$[G]^T = [B_j \quad \dots \quad B_{j+k-1}]$ содержит геометрию — вершины многоугольника, а $[F]$ состоит из соответствующих базисных функций.

В работе [5-19] показано, что для периодических В-сплайнов общий вид $[N]$:

$$[N^*] = [N_{i+1,j+1}^*], \quad (5-91)$$

где

$$\begin{aligned} N_{i+1,j+1}^* &= \frac{1}{(k-1)!} n_{i+1,j+1}^* \\ &= \frac{1}{(k-1)!} \binom{k-1}{i} \sum_{l=j}^{k-1} (k-(l+1))^i (-1)^{l-j} \binom{k}{l-j}, \quad 0 \leq i, \quad j \leq k-1. \end{aligned}$$

Вспомним, что у периодических замкнутых В-сплайнов необходимо повторить некоторые вершины, чтобы ликвидировать разрыв. В матричном виде это очень просто. Замкнутый периодический сплайн имеет вид

$$P_{j+1}(t^*) = \sum_{i=0}^{k-1} N_{i+1,k}^*(t^*) B_{((j+i) \bmod (n+1)) + 1}, \quad 0 \leq j \leq n. \quad (5-92)$$

Или в матричном виде

$$P_{j+1}(t^*) = [T^*][N^*] \begin{bmatrix} B_{j \bmod (n+1)+1} \\ B_{((j+1) \bmod (n+1))+1} \\ \vdots \\ B_{((j+n-k) \bmod (n+1))+1} \end{bmatrix}, \quad (5-93)$$

где $[T^*]$ и $[N^*]$ взяты из уравнения (5-90), а \bmod — остаток; например, $3 \bmod 2 = 1$.

Открытые В-сплайны также можно записать в матричном виде, однако из-за кратных узловых значений на концах это не так удобно, как для периодических. В общем случае матричный вид открытого В-сплайна с целыми узловыми значениями таков:

$$P(t) = [F][G] = [T][N][G], \quad (5-94)$$

где

$$[G]^T = [B_1 \quad \dots \quad B_{n+1}],$$

а элементы $[F]$ и $[N]$ получены по алгоритму Кокса–де Бура (уравнение (5-84)) для каждого ненулевого интервала вектора $[X]$. У В-сплайнов низкого порядка, заданных большим количеством точек, большинство членов $[F]$ равны нулю. Используя это свойство, можно значительно сэкономить вычисления.¹

В работе [5-19] приведена обобщенная формулировка для $[N^*]$ на нормализованном интервале $0 \leq t^* < 1$; однако из-за кратности концевых узловых значений первые и последние $k-1$ N^* матриц являются частными случаями.

Рассмотрим пример матричного метода.

Пример 5-15 Расчет замкнутого В-сплайна

Найти замкнутый В-сплайн четвертого порядка ($k = 4$), заданный многоугольником, изображенным на рис. 5-46. Записать результат в матричном виде. Вершины многоугольника: $B_1[2 \ 0]$, $B_2[4 \ 0]$, $B_3[4 \ 2]$, $B_4[4 \ 4]$, $B_5[2 \ 4]$, $B_6[0 \ 4]$, $B_7[0 \ 2]$, $B_8[0 \ 0]$, $B_9[2 \ 0]$, $n = 8$.

Для каждого единичного интервала $0 \leq t^* < 1$ из уравнений (5-89) и (5-93) получим

$$P_{j+1}(t^*) = \frac{1}{6} [t^{*3} \quad t^{*2} \quad t^* \quad 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} B_{(j \bmod 8)+1} \\ B_{((j+1) \bmod 8)+1} \\ B_{((j+2) \bmod 8)+1} \\ B_{((j+3) \bmod 8)+1} \end{bmatrix}.$$

¹ См. алгоритм из Приложения G.

При $t^* = 1/2$ на первом сегменте ($j = 0$) кривой

$$\begin{aligned} P_1\left(\frac{1}{2}\right) &= \frac{1}{6} \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix} = \\ &= \frac{1}{48} [1 \ 23 \ 23 \ 1] \begin{bmatrix} 2 & 0 \\ 4 & 0 \\ 4 & 2 \\ 4 & 4 \end{bmatrix} = \\ &= \frac{1}{48} [190 \ 50] = [3.958 \ 1.042]. \end{aligned}$$

При $t^* = 1/2$ на восьмом сегменте ($j = 7$) кривой

$$\begin{aligned} P_8\left(\frac{1}{2}\right) &= \frac{1}{48} [1 \ 23 \ 23 \ 1] \begin{bmatrix} B_8 \\ B_1 \\ B_2 \\ B_3 \end{bmatrix} = \\ &= \frac{1}{48} [1 \ 23 \ 23 \ 1] \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 4 & 0 \\ 4 & 2 \end{bmatrix} = \\ &= \frac{1}{48} [142 \ 2] = [2.958 \ 0.417]. \end{aligned}$$

Результат показан на рис. 5-46б.

Производная в любой точке В-сплайна вычисляется формальным дифференцированием. Из уравнения (5-83)

$$P(t) = \sum_{i=1}^{n+1} B_i N_{i,k}(t) \quad (5-83)$$

первая производная:

$$P'(t) = \sum_{i=1}^{n+1} B_i N'_{i,k}(t) \quad (5-95)$$

и вторая производная:

$$P''(t) = \sum_{i=1}^{n+1} B_i N''_{i,k}(t). \quad (5-96)$$

Штрихи обозначают дифференцирование по параметру t .

Производные базисных функций также получают формальным дифференцированием. После однократного дифференцирования уравнения (5-84) имеем

$$N'_{i,k}(t) = \frac{N_{i,k-1}(t) + (t - x_i) N'_{i,k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t) N'_{i+1,k-1}(t) - N_{i+1,k-1}(t)}{x_{i+k} - x_{i+1}}. \quad (5-97)$$

Из уравнения (5-84а) следует, что $N'_{i,1}(t) = 0$ для всех t . Следовательно, для $k = 2$ уравнение (5-97) принимает вид

$$N'_{i,2}(t) = \frac{N_{i,1}(t)}{x_{i+k-1} - x_i} - \frac{N_{i+1,1}(t)}{x_{i+k} - x_{i+1}}. \quad (5-98)$$

Вторые производные базисных функций получаем, дифференцируя уравнение (5-97),

$$\begin{aligned} N''_{i,k}(t) &= \frac{2N'_{i,k-1}(t) + (t - x_i) N''_{i,k-1}(t)}{x_{i+k-1} - x_i} + \\ &+ \frac{(x_{i+k} - t) N''_{i+1,k-1}(t) - 2N'_{i+1,k-1}(t)}{x_{i+k} - x_{i+1}}. \end{aligned} \quad (5-99)$$

Здесь как $N''_{i,1}(t) = 0$, так и $N''_{i,2}(t) = 0$ равны нулю для всех t . Следовательно, для $k = 3$ уравнение (5-97) имеет вид

$$N''_{i,3}(t) = 2 \left(\frac{N'_{i,2}(t)}{x_{i+k-1} - x_i} - \frac{N'_{i+1,2}(t)}{x_{i+k} - x_{i+1}} \right). \quad (5-100)$$

На рис. 5-51 изображены несколько базисных функций В-сплайна и их производные для $k = 4$. Отметим, что при $k = 4$ каждая базисная функция — кусочно кубическая, первая производная — кусочно параболическая, вторая производная — кусочно линейная. Третья производная разрывна и состоит из различных констант.

Рассмотрим пример.

Пример 5-16 Расчет производных В-сплайна

Рассмотрим многоугольник из примера 5-12: $B_1[1 \ 1]$, $B_2[2 \ 3]$, $B_3[4 \ 3]$, $B_4[3 \ 1]$ и соответствующий сплайн. Найти первую производную кривой второго порядка ($k = 2$).

Для $k = 2$ открытый узловый вектор имеет вид:

$$[0 \ 0 \ 1 \ 2 \ 3 \ 3],$$

где $x_1 = 0$, $x_2 = 0, \dots, x_6 = 3$. Диапазон параметра $0 \leq t \leq 3$. Из уравнения (5-95) получаем первую производную для $n = 3$, $k = 2$

$$P'(t) = B_1 N'_{1,2}(t) + B_2 N'_{2,2}(t) + B_3 N'_{3,2}(t) + B_4 N'_{4,2}(t).$$

Из уравнения (5-98) следует, что

$$N'_{i,2}(t) = \frac{N_{i,1}(t)}{x_{i+k-1} - x_i} - \frac{N_{i+1,1}(t)}{x_{i+k} - x_{i+1}}.$$

Пользуясь результатами уравнения (5-12), для $0 \leq t < 1$ получим

$$N_{2,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 2,$$

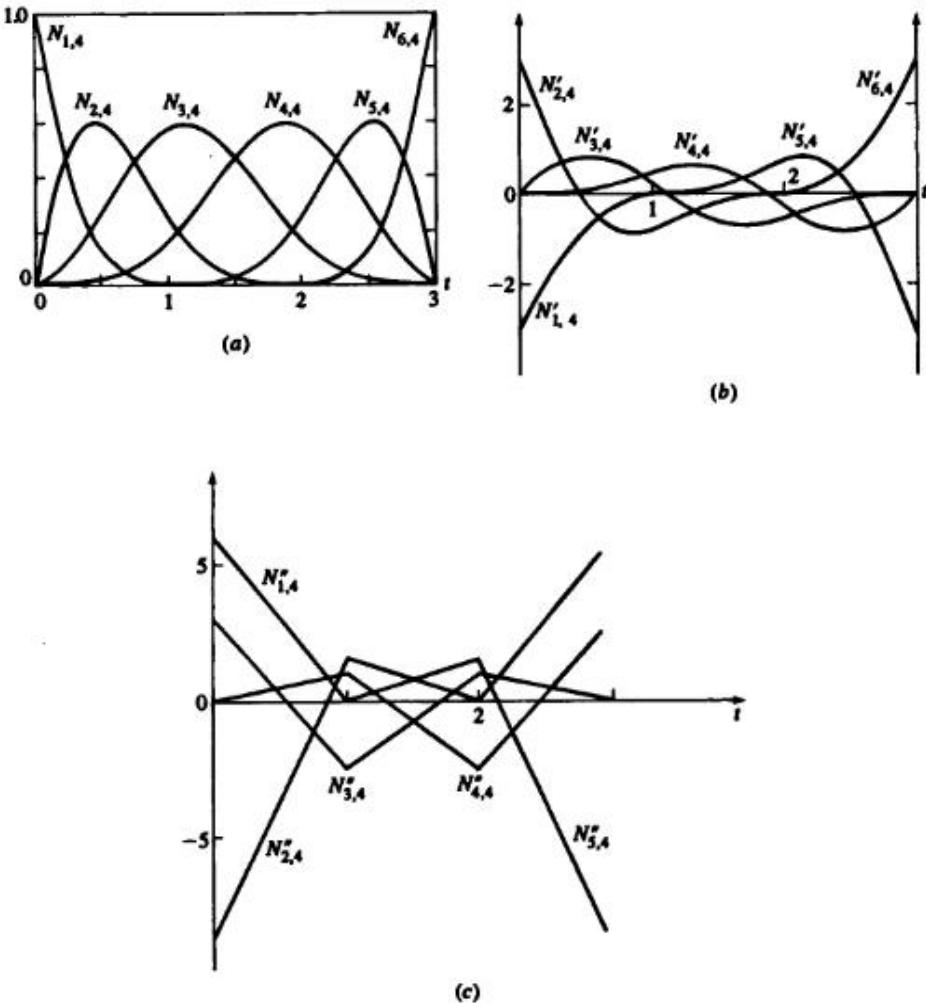


Рис. 5.51 Базисные функции В-сплайна и их первые и вторые производные, $k = 4$, $n = 6$. (a) Базисные функции; (b) первая производная; (c) вторая производная.

и

$$N'_{1,2}(t) = -1; \quad N'_{2,2}(t) = 1; \quad N'_{i,2}(t) = 0, \quad i \neq 2.$$

Итак

$$P'(t) = B_2 - B_1,$$

что является наклоном (касательным вектором) первого ребра многоугольника, как оно и должно быть. Для $1 \leq t < 2$

$$N_{3,1}(t) = 1; \quad N_{i,1}(t) = 0 \quad i \neq 3$$

и

$$N'_{2,2}(t) = -1; \quad N'_{3,2}(t) = 1; \quad N'_{i,2}(t) = 0 \quad i \neq 2, 3$$

т. е.

$$P'(t) = B_3 - B_2,$$

что равно наклону (касательному вектору) второго ребра.

Наконец, для $2 \leq t < 3$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0 \quad i \neq 4$$

и

$$N'_{3,2}(t) = -1; \quad N'_{4,2}(t) = 1; \quad N'_{i,2}(t) = 0 \quad i \neq 3, 4,$$

т. е.

$$P'(t) = B_4 - B_3$$

— наклон последнего ребра.

При $t = 3$, $N_{i,1}(3) = 0$ для всех i . Следовательно, формально $P'(3) = 0$, но предел $t = 3 - \varepsilon$, при $\varepsilon \rightarrow 0$ показывает, что на самом деле $P'(3) = (B_4 - B_3)$.

5-10 КОНЕЧНЫЕ УСЛОВИЯ ДЛЯ ПЕРИОДИЧЕСКИХ В-СПЛАЙНОВ

Как было показано ранее, конец и начало периодического В-сплайна не совпадают с первой и последней вершинами определяющего многоугольника. Отсюда возникают два вопроса. Во-первых, где же именно начинается и заканчивается В-сплайн, и каковы его граничные условия, т. е. производные в этих точках? Во-вторых, что влияет на конечные точки и производные в них? В работе [5-24] изучается этот вопрос для частного случая кубических ($k = 4$) В-сплайнов. Мы рассмотрим более общий случай.

Произвольный периодический В-сплайн начинается в точке $P_s = P(t = x_k)$ и оканчивается в точке $P_e = P(t = x_{n+1})$. Для целых узловых значений, начинающихся с нуля, $P_s = P(t = k)$ и $P_e = P(t = n)$. Из уравнений (5-87) и (5-91), а также из того, что любая точка В-сплайна зависит от k ближайших вершин, следует, что для преобразованного параметра $0 \leq t^* < 1$ начальная точка соответствует $t^* = 0$. Итак

$$P_s = \frac{1}{(k-1)!} (N_{k,1}^* B_1 + N_{k,2}^* B_2 + \cdots + N_{k,k}^* B_k).$$

Заметим, что $N_{k,k}^* = 0$ для всех k

$$P_s = \frac{1}{(k-1)!} (N_{k,1}^* B_1 + N_{k,2}^* B_2 + \cdots + N_{k,k-1}^* B_{k-1}), \quad n \geq k. \quad (5-101)$$

В конце интервала $t^* = 1$. Известно, что $\sum_{i=1}^k N_{i,1}^* = 0$, поэтому конечная точка

$$P_e = \frac{1}{(k-1)!} \left(\sum_{i=1}^k N_{i,2}^* B_{n-k+3} + \sum_{i=1}^k N_{i,3}^* B_{n-k+4} + \cdots + \sum_{i=1}^k N_{i,k}^* B_{n+1} \right) \quad n \geq k. \quad (5-102)$$

Для квадратичного ($k = 3$) периодического сплайна уравнения (5-101) и (5-102) принимают вид

$$\begin{aligned} P_s &= \frac{1}{2}(B_1 + B_2), \\ P_e &= \frac{1}{2}(B_n + B_{n+1}). \end{aligned}$$

Следовательно, квадратичный периодический В-сплайн начинается и кончается, соответственно, в середине первого и последнего ребер многоугольника.

Конечные точки кубического ($k = 4$) периодического сплайна таковы:

$$\begin{aligned} P_s &= \frac{1}{6}(B_1 + 4B_2 + B_3), \\ B_e &= \frac{1}{6}(B_{n-1} + 4B_n + B_{n+1}). \end{aligned}$$

Из уравнения (5-90) первая производная в начальной точке

$$\begin{aligned} P'_s &= [T^{*'}]_{t^*=0}[N^*][G] = \\ &= \frac{1}{(k-1)!}(N_{k-1,1}^* B_1 + N_{k-1,2}^* B_2 + \cdots + N_{k-1,k-1}^* B_{k-1}) \end{aligned} \quad (5-103)$$

так как $N_{k-1,k}^* = 0$ для всех k . Первая производная в конечной точке

$$\begin{aligned} P'_e &= [T^{*'}]_{t^*=1}[N^*][G] = \\ &= \frac{1}{(k-1)!} \left(\sum_{i=1}^{k-1} (k-i) N_{i,2}^* B_{n-k+3} + \right. \\ &\quad \left. + \sum_{i=1}^{k-1} (k-i) N_{i,3}^* B_{n-k+4} + \cdots + \sum_{i=1}^{k-1} (k-i) N_{i,k}^* B_{n+1} \right), \end{aligned} \quad (5-104)$$

где $[T^{*'}] = [(k-1)t^{*(k-2)} \quad (k-2)t^{*(k-3)} \quad \dots \quad 1 \quad 0]$ — производная вектора параметра.

В случае квадратичных ($k = 3$) периодических В-сплайнов эти результаты сводятся к

$$\begin{aligned} P'_s &= \frac{1}{2}(-2B_1 + 2B_2) = B_2 - B_1 \\ P'_e &= \frac{1}{2}(-2B_n + 2B_{n+1}) = B_{n+1} - B_n, \end{aligned}$$

т. е. векторам касательных (наклону) первого и последнего ребер. Для кубических ($k = 4$) кривых

$$\begin{aligned} P'_s &= \frac{1}{6}(-3B_1 + 3B_3) = \frac{1}{2}(B_3 - B_1) \\ P'_e &= \frac{1}{6}(-3B_{n-2} + 3B_{n+1}) = \frac{1}{2}(B_{n+1} - B_{n-1}). \end{aligned}$$

Здесь касательные векторы (наклоны) зависят от касательных к трем первым и к трем последним сегментам кривой.

Вторые производные в конечных точках:

$$\begin{aligned} P''_s &= [T^{*''}]_{t^*=0}[N^*][G] = \\ &= \frac{2}{(k-1)!}(N_{k-2,1}^* B_1 + N_{k-2,2}^* B_2 + \cdots + N_{k-2,k-1}^* B_{k-1}) \end{aligned} \quad (5-105)$$

$$\begin{aligned} P''_e &= [T^{*''}]_{t^*=1}[N^*][G] = \\ &= \frac{1}{(k-1)!} \left(\sum_{i=1}^{k-2} (k-i)(k-i-1) N_{i,2}^* B_{n-k+3} + \right. \\ &\quad \left. + \sum_{i=1}^{k-2} (k-i)(k-i-1) N_{i,3}^* B_{n-k+4} + \right. \\ &\quad \left. + \cdots + \sum_{i=1}^{k-2} (k-i)(k-i-1) N_{i,k}^* B_{n+1} \right) \end{aligned} \quad (5-106)$$

где $[T^{*''}] = [(k-1)t^{*(k-2)} \quad (k-2)t^{*(k-3)} \quad \dots \quad 1 \quad 0]$ — вторая производная вектора параметра.

Для кубических ($k = 4$) сплайнов уравнения (5-105) и (5-106) дают

$$\begin{aligned} P''_s &= \frac{1}{6}(6B_1 - 12B_2 + 6B_3) = B_1 - 2B_2 + B_3, \\ P''_e &= \frac{1}{6}(6B_{n-1} - 12B_n + 6B_{n+1}) = B_{n-1} - 2B_n + B_{n+1}. \end{aligned}$$

Существует два метода регулирования положения первой и последней точек, а также конечных условий: кратные вершины и псевдовершины.

Кратные вершины на концах периодического В-сплайна притягивают концы сплайна к соответствующим вершинам. Если определены $k-1$ совпадающая вершина, то конечные точки кривой совпадают с вершинами многоугольника и касательные (наклон) совпадают с направлением соседних ненулевых ребер.

Например, для $k = 3$ с двойными вершинами на концах, т. е. $B_1 = B_2$ и $B_n = B_{n+1}$ из уравнений (5-101) и (5-102) следует

$$\begin{aligned} P_s &= \frac{1}{2}(B_1 + B_2) = \frac{1}{2}(2B_1) = B_1, \\ P_e &= \frac{1}{2}(B_n + B_{n+1}) = \frac{1}{2}(2B_{n+1}) = B_{n+1}. \end{aligned}$$

Для $k = 4$ с двойными вершинами на концах эти же уравнения дают

$$P_s = \frac{1}{6}(5B_1 + B_3),$$

$$P_e = \frac{1}{6}(B_{n-1} + 5B_{n+1}).$$

Отсюда начальная точка находится на расстоянии от одной шестой ребра, соединяющего точки B_1 и B_3 , а конечная точка находится на расстоянии в пять шестых ребра от B_{n-1} до B_{n+1} .

Если в концах располагаются по три кратные вершины, т. е. $B_1 = B_2 = B_3$ и $B_{n-1} = B_n = B_{n+1}$ для $k = 4$, то

$$P_s = \frac{1}{6}(B_1 + 4B_1 + B_1) = B_1,$$

$$P_e = \frac{1}{6}(B_{n+1} + 4B_{n+1} + B_{n+1}) = B_{n+1}.$$

Кривая начинается и кончается в первой и последней вершинах многоугольника. На рис. 5-52 изображено влияние кратных вершин.

С тремя кратными вершинами на концах первый и последний сегменты В-сплайна для $k = 4$ (см. уравнение 5-89) имеют вид:

$$P_1(t^*) = B_1 + \frac{t^{*3}}{6}[B_4 - B_1] = B_3 + \frac{t^{*3}}{6}[B_4 - B_3] \quad 0 \leq t^* < 0$$

и

$$P_n(t^*) = B_{n+1} + \frac{(1-t^{*3})}{6}(B_n - B_{n+1}).$$

При этом первый и последний сегменты линейны. Первый сегмент на одну шестую длины ребра от $B_1 = B_2 = B_3$ до B_4 совпадает с этим ребром. Последний сегмент на одну шестую ребра от B_{n-2} до $B_{n-1} = B_n = B_{n+1}$ совпадает с соответствующим ребром.

Несмотря на то, что участки можно сделать как угодно малыми, в некоторых прикладных областях это может быть неудобным. В таких случаях лучше пользоваться открытыми В-сплайнами.

Псевдовершины в концах периодического В-сплайна, в отличие от кратных вершин, позволяют управлять как положением конечных точек, так и граничными условиями. В общем случае псевдовершины не обозначаются, и пользователь не может контролировать их. Как показано на рис. 5-53, B_0 и B_{n+2} — псевдовершины в начале и конце В-сплайна. В этих обозначениях уравнения (5-101) и (5-102) принимают вид

$$P_s = \frac{1}{(k-1)!}(N_{k,1}^* B_0 + N_{k,2}^* B_1 + \cdots + N_{k,k-1}^* B_{k-2}) \quad n \geq k \quad (5-107)$$

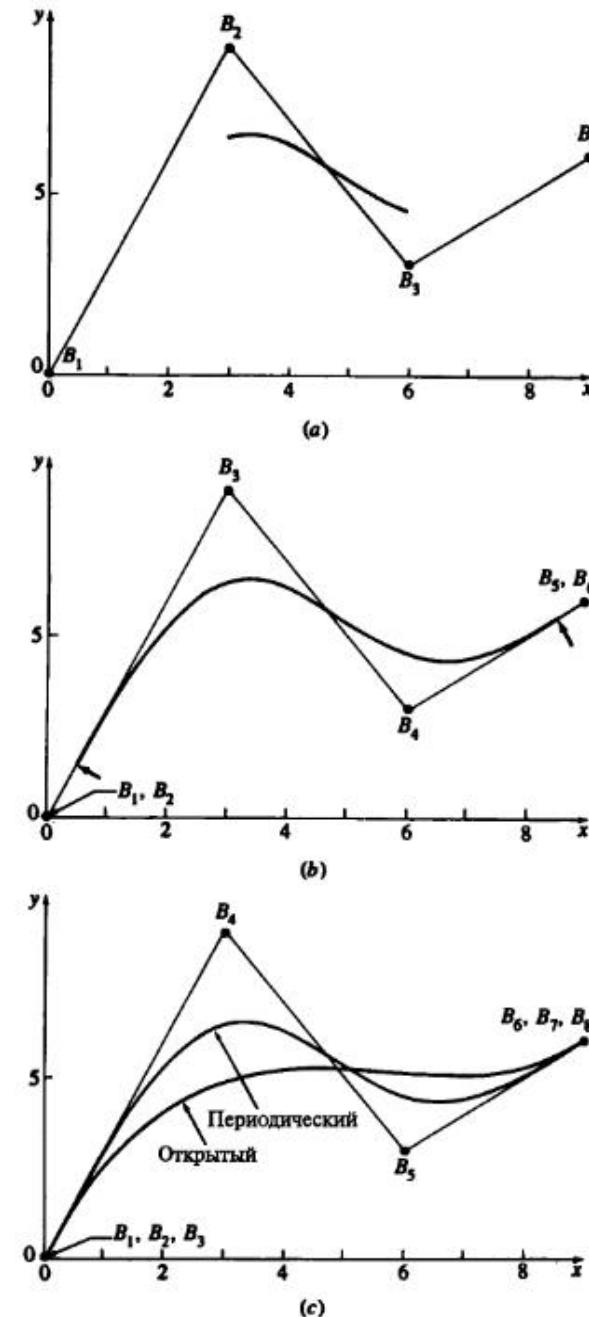


Рис. 5-52 Совпадающие кратные вершины на концах определяющего многоугольника ($k = 4$). (a) Кратных вершин нет; (b) две кратные вершины; (c) три кратные вершины и соответствующие открытые В-сплайны.

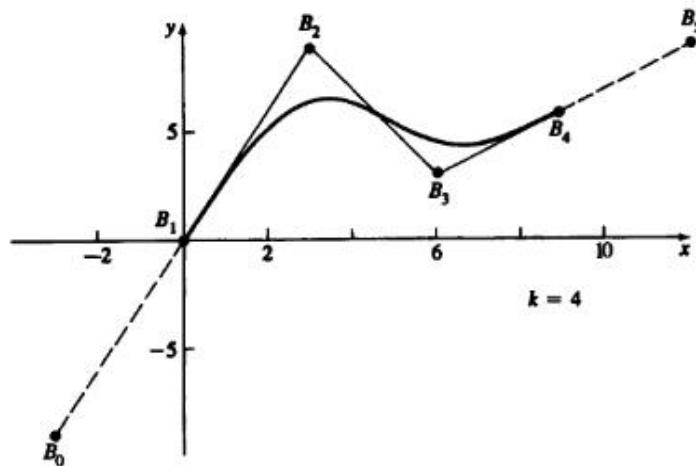


Рис. 5-53 Псевдовершины определяют начальную и конечную точки периодического B-сплайна.

и

$$P_e = \frac{1}{(k-1)!} \left(\sum_{i=1}^k N_{i,2}^* B_{n-k+4} + \sum_{i=1}^k N_{i,3}^* B_{n-k+5} + \cdots + \sum_{i=1}^k N_{i,k}^* B_{n+2} \right) \quad n \geq k \quad (5-108)$$

При $P_s = B_1$ и $P_e = B_{n+1}$ эти уравнения имеют вид

$$B_0 = ((k-1)! - N_{k,2}^*) B_1 = (N_{k,3}^* B_2 + \cdots + N_{k,k-1}^* B_{k-2}) \quad n \geq k \quad (5-109)$$

и

$$B_{n+2} = \left((k-1)! - \sum_{i=1}^k N_{i,k-1}^* \right) B_{n+1} - \left(\sum_{i=1}^k N_{i,2}^* B_{n-k+4} + \cdots + \sum_{i=1}^k N_{i,k-2}^* B_n \right) \quad n \geq k \quad (5-110)$$

где $N_{k,1}^* = 1$ и $\sum_{i=1}^k N_{i,k}^* = 1$.

Для $k = 3$ из уравнений (5-109) и (5-110) следует, что $B_0 = B_1$ и $B_{n+2} = B_{n+1}$, т.е. двукратные вершины на концах!

Для $k = 4$ аналогично

$$B_0 = (6 - 4)B_1 - B_2 = 2B_1 - B_2$$

и

$$B_{n+2} = (6 - 4)B_{n+1} - B_n = 2B_{n+1} - B_n.$$

Пример изображен на рис. 5-53.

Первая и вторая производные в концах выражены уравнениями (5-103) и (5-104) с соответствующими изменениями для учета псевдовершин. Например, используя приведенные результаты для B_0 , из уравнений (5-103) и (5-104) для $k = 4$ получаем

$$P'_s = \frac{1}{2}(B_2 - B_0) = \frac{1}{2}\{B_2 - (2B_1 - B_2)\} = B_2 - B_1$$

и

$$P'_e = \frac{1}{2}(B_{n+2} - B_n) = \frac{1}{2}\{2B_{n+1} - B_n - B_n\} = B_{n+1} - B_n.$$

Итак, кривая касательна к первому и последнему ребрам многоугольника.

Аналогично, для $k = 4$ и точек B_0 и B_{n+2} из уравнений (5-105) и (5-106) следует

$$P''_s = B_0 - 2B_1 + B_2 = 2B_1 - B_2 - 2B_1 + B_2 = 0,$$

$$P''_e = B_n + 2B_{n+1} + B_{n+2} = B_n - 2B_{n+1} + 2B_{n+1} - B_n = 0,$$

т.е. «кривизна» в концах нулевая.

Переписав уравнения (5-103) и (5-104), можно определить псевдовершины, порождающие заданные граничные условия. В частности, из уравнения (5-103)

$$B_0 = \frac{1}{N_{k-1,1}^*} \left\{ (k-1)! P'_s - (N_{k-1,2}^* B_1 + \cdots + N_{k-1,k-1}^* B_{k-2}) \right\} \quad n \geq k \quad (5-111)$$

а из уравнения (5-104)

$$B_{n+2} = \frac{1}{\sum_{i=1}^{k-1} (k-1) N_{i,k}^*} \left\{ (k-1)! P'_e - \left(\sum_{i=1}^{k-1} (k-1) N_{i,2}^* B_{n-k+4} + \cdots + \sum_{i=1}^{k-1} (k-1) N_{i,k-1}^* B_{n+1} \right) \right\} \quad n \geq k \quad (5-112)$$

Для $k = 4$ уравнения (5-111) и (5-112) сводятся к

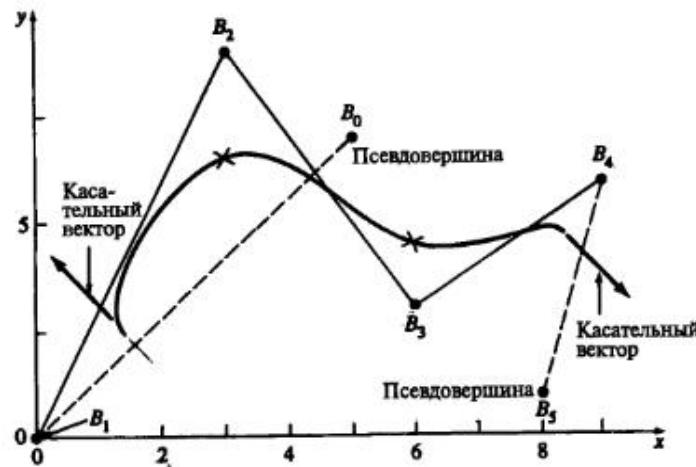
$$B_0 = B_2 - 2P'_s,$$

$$B_{n+2} = 2P'_e + B_n.$$

Типичный пример изображен на рис. 5-54. Конечные точки кривой получены подстановкой этих значений в уравнения (5-107) и (5-108). Из рисунка видно, что первая и последняя точки кривой с заданными касательными на концах для многоугольника $B_0 B_1 B_2 B_3 B_4 B_5$ не совпадают с соответствующими точками кривой для многоугольника $B_1 B_2 B_3 B_4$ (отмечено крестиками).

Для управления второй производной или приблизительной кривизной в концах кривой перепишем уравнения (5-105) и (5-106) в виде

$$B_0 = \frac{1}{N_{k-2,1}^*} \left\{ \frac{(k-1)!}{2} P''_s - (N_{k-2,2}^* B_1 + \cdots + N_{k-2,k-1}^* B_{k-2}) \right\}, \quad n \geq k, \quad (5-113)$$



квадратную матрицу (см. разд. 3-31). Вершины определяющего многоугольника, слаживающего кривую заданы условием

$$\begin{aligned}[D] &= [N][B], \\ [N]^T[D] &= [N]^T[N][B]\end{aligned}$$

и

$$[B] = [[N]^T \quad [N]]^{-1}[N]^T[D] \quad (5-117)$$

Оба метода подразумевают, что матрица $[N]$ известна. Базисные функции $N_{i,k}(t_j)$ и, следовательно, матрицу $[N]$ можно получить, если задан порядок В-сплайна k , количество точек многоугольника $n+1$, а также величина параметра вдоль кривой. При соблюдении условия $2 \leq k \leq n+1 \leq j$ порядок и количество вершин произвольны.

Величина параметра t_j для каждой заданной точки — это мера расстояния до точки вдоль В-сплайна. Удобна аппроксимация параметра с помощью длин хорд между заданными точками; в частности, для j точек величина параметра в точке l

$$\begin{aligned}t_1 &= 0 \\ \frac{t_l}{t_{\max}} &= \frac{\sum_{s=2}^l |D_s - D_{s-1}|}{\sum_{s=2}^j |D_s - D_{s-1}|} \quad l \geq 2.\end{aligned}$$

Максимальное значение параметра t_{\max} обычно выбирается в качестве максимальной величины узлового вектора. Подобные методы затрагиваются в работах [5-15] и [5-22].

Для открытого равномерного узлового вектора с k кратными вершинами на концах кривая Безье получается при $n = k$.

Рассмотрим пример.

Пример 5-17 Подгонка В-сплайна

Даны пять точек $D_1[0 \ 0]$, $D_2[1.5 \ 2]$, $D_3[3 \ 2.5]$, $D_4[4.5 \ 2]$, $D_5[6 \ 0]$. Найти определяющие многоугольники третьего порядка ($k = 3$) с пятью и четырьмя вершинами для В-сплайна, проходящего через данные точки. Использовать хордовую аппроксимацию параметра.

Сначала найдем длины хорд.

$$D_{21} = |D_2 - D_1| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \sqrt{(1.5)^2 + (2)^2} = \sqrt{6.25} = 2.5,$$

$$D_{32} = |D_3 - D_2| = \sqrt{(1.5)^2 + (.5)^2} = 1.58,$$

$$D_{43} = |D_4 - D_3| = \sqrt{(1.5)^2 + (-.5)^2} = 1.58,$$

$$D_{54} = |D_5 - D_4| = \sqrt{(1.5)^2 + (-2)^2} = 2.5$$

и

$$\sum_{s=2}^5 (D_s - D_{s-1}) = D_{51} = 8.16.$$

Итак

$$\begin{aligned}t_1 &= 0 \\ \frac{t_2}{t_{\max}} &= \frac{D_{21}}{D_{51}} = \frac{2.5}{8.16} = 0.31 \\ \frac{t_3}{t_{\max}} &= \frac{D_{31}}{D_{51}} = \frac{(2.5 + 1.58)}{8.16} = 0.5 \\ \frac{t_4}{t_{\max}} &= \frac{D_{41}}{D_{51}} = \frac{(2.5 + 1.58 + 1.58)}{8.16} = 0.69 \\ \frac{t_5}{t_{\max}} &= \frac{D_{51}}{D_{51}} = 1.\end{aligned}$$

Для пятиугольника максимальное значение узлового вектора у В-сплайна третьего порядка $n - k + 2 = 4 - 3 + 2 = 3$. Здесь n на единицу меньше количества вершин многоугольника. Узловой вектор с кратностью k на концах имеет вид

$$[0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3 \ 3].$$

Уравнение (5-115) с этими значениями выглядит следующим образом:

$$[D] = [N][B]$$

$$\begin{bmatrix} 0 & 0 \\ 1.5 & 2 \\ 3 & 2.5 \\ 4.5 & 2 \\ 6 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.007 & 0.571 & 0.422 & 0 & 0 \\ 0 & 0.125 & 0.75 & 0.125 & 0 \\ 0 & 0 & 0.422 & 0.571 & 0.007 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} [B].$$

Решим относительно $[B]$

$$[B] = [N]^{-1}[D]$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -0.013 & 2.037 & -307 & 0.286 & -0.002 \\ 0.003 & -0.387 & 1.769 & -0.287 & 0.003 \\ -0.002 & 0.286 & -1.307 & 2.037 & -0.013 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1.5 & 2 \\ 3 & 2.5 \\ 4.5 & 2 \\ 6 & 0 \end{bmatrix}$$

$$[B] = \begin{bmatrix} 0 & 0 \\ 0.409 & 1.378 \\ 3 & 2.874 \\ 5.591 & 1.377 \\ 6 & 0 \end{bmatrix}.$$

На рис. 5-56а изображены исходные точки, рассчитанные вершины многоугольника и кривая.

Для четырехугольника узловой вектор с кратностью k на концах:

$$[0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2].$$

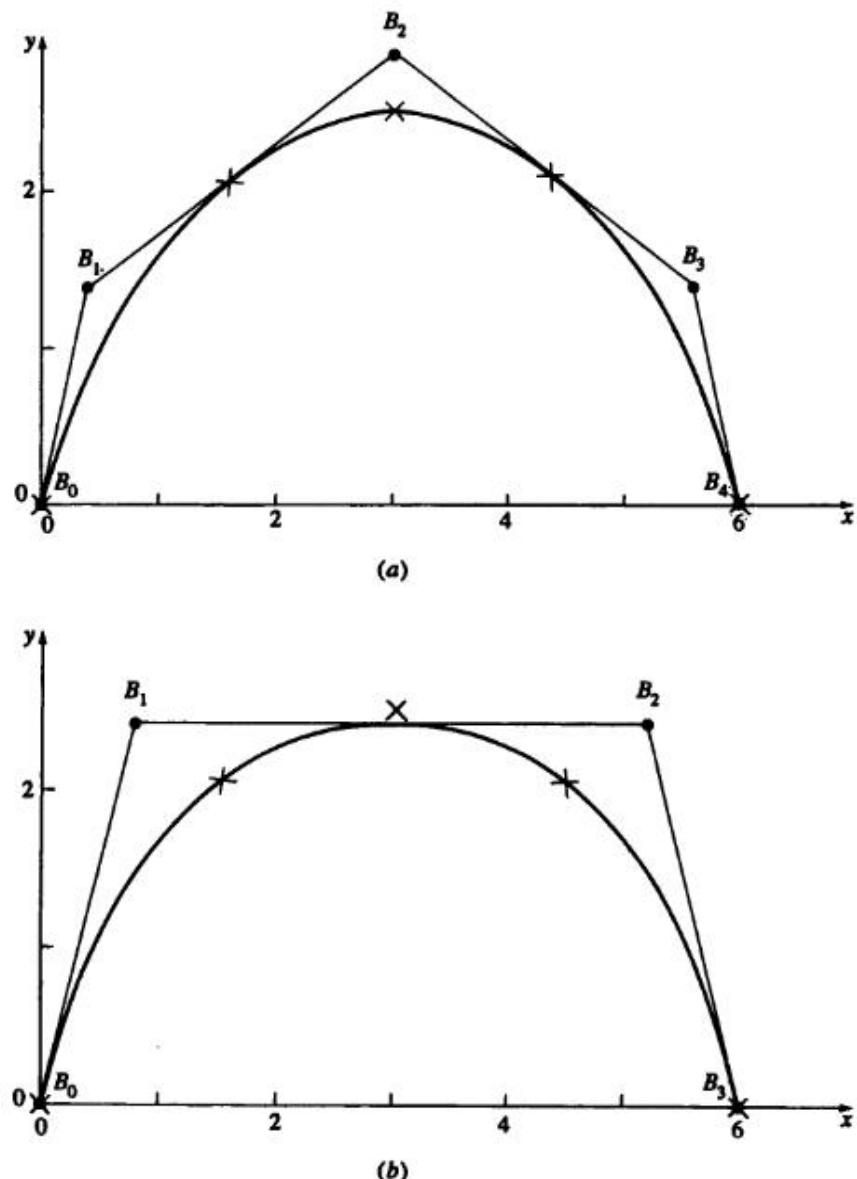


Рис. 5-56 Результаты для примера 5-17. (a) Пятиугольник; (b) четырехугольник.

Матрица $[N]$:

$$[N] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.15 & 0.662 & 0.188 & 0 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0.188 & 0.662 & 0.15 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Умножим на $[N]^T$ и обратим ее:

$$[[N]^T [N]]^{-1} = \begin{bmatrix} 0.995 & -0.21 & 0.106 & -0.005 \\ -0.21 & 2.684 & -1.855 & 0.106 \\ 0.106 & -1.855 & 2.684 & -0.21 \\ -0.005 & 0.106 & -0.21 & 0.995 \end{bmatrix}.$$

Тогда из уравнения (5-117) получим

$$[B] = [[N]^T [N]]^{-1} [N]^T [D] = \begin{bmatrix} 0 & 0 \\ 0.788 & 2.414 \\ 5.212 & 2.414 \\ 6 & 0 \end{bmatrix}.$$

На рис. 5-56 *b* изображены исходные точки, рассчитанные вершины многоугольника и кривая.

Этот метод допускает произвольное расположение всех точек определяющего многоугольника в трехмерном пространстве, но иногда бывает удобно ограничиться плоскостью, например при подгонке В-сплайна под заданный контур корабля. В работе [5-26] приведен соответствующий метод как для кривых, так и для поверхностей. Его сущность состоит в том, что итеративно вычисляется параметр для фиксированной координаты до тех пор, пока значение на кривой при величине параметра, вычисленной по подогнанному многоугольнику, не попадает в некоторую окрестность фиксированного значения, т. е. $|x_{\text{fixed}} - x_{\text{calc}}| \leq \text{errtol}$. Подгонка получается менее точной, но более удобной для модификаций.

5-12 РАЗБИЕНИЕ В-СПЛАЙНОВ

Для того чтобы увеличить гибкость кривой Безье, необходимо добавить вершину к определяющему многоугольнику, что повышает степень полиномиальной кривой (см. разд. 5-8). Гибкость В-сплайна также растет с увеличением порядка его базиса, т. е. полиномиальных сегментов. В работе [5-27] предлагаются теория и алгоритм увеличения порядка В-сплайна.

Повысить гибкость В-сплайна можно другим способом: вставить дополнительные узловые значения в определяющий узловой вектор. При этом полиномиальный сегмент для некоторого интервала узловых значений (параметрического интервала) локально разбивается на два. Есть два основных метода вставки узловых значений. Первый, так называемый алгоритм Осло, разработанный Коэном и др. ([5-28] и [5-29]), вставляет сразу несколько узловых значений. Второй метод Бема ([5-30] и [5-31]) последовательно вставляет по одному узловому значению. Мы рассмотрим только метод Осло.

Как повышение порядка, так и дополнительные узловые значения увеличивают гибкость базиса кривой (поверхности) и, следовательно, кривой без изменения ее формы. Это возможно потому, что один В-сплайн можно задать бесконечным множеством многоугольников с более чем минимальным количеством вершин. Форма кривой меняется путем передвижения вновь полученных вершин.

Рассмотрим исходную кривую $P(t) = \sum_{i=1}^{n+1} B_i N_{i,k}(t)$ с узловым вектором $[X] = [x_1 \ x_2 \ \dots \ x_{n+k-1}]$. Вставив узел, получаем новую кривую $R(s)$:

$$R(s) = \sum_{j=1}^{m+1} C_j M_{j,k}(s) \quad (5-118)$$

с новым узловым вектором

$$[Y] = [y_1 \ y_2 \ \dots \ y_{m+k+1}],$$

где $m > n$. Надо найти новые вершины определяющего многоугольника C_j , такие, что $P(t) = R(s)$. По алгоритму Осло [5-29]

$$C_j = \sum_{i=1}^{N+1} \alpha_{i,j}^k B_i, \quad 1 \leq i \leq n, \quad 1 \leq j \leq m, \quad (5-119)$$

где $\alpha_{i,j}^k$ заданы рекурсивным отношением

$$\alpha_{i,j}^1 = \begin{cases} 1 & x_i \leq y_j < x_{i+1}, \\ 0 & \text{в противном случае} \end{cases}, \quad (5-120a)$$

$$\alpha_{i,j}^k = \frac{y_{j+k-1} - x_i}{x_{i+k-1} - x_i} \alpha_{i,j}^{k-1} + \frac{x_{i+k} - y_{j+k-1}}{x_{i+k} - x_{i+1}} \alpha_{i+1,j}^{k-1}. \quad (5-120b)$$

Отметим, что $\sum_i^{n+1} \alpha_{i,j}^k = 1$.

На первый взгляд может показаться, что равномерный исходный вектор, не важно периодический или открытый, после вставки одного или более узловых значений станет неравномерным. Равномерность вектора можно сохранить, добавляя кратные узловые значения в середину каждого ненулевого интервала. Рассмотрим разбиение подробнее на примере.

Пример 5-18 Глобальное разбиение В-сплайна

Пусть дан открытый В-сплайн третьего порядка ($k = 3$), изначально заданный четырьмя ($n + 1 = 4$) вершинами $B_1[0 \ 0]$, $B_2[1 \ 1]$, $B_3[2 \ 1]$, $B_4[3 \ 0]$. Требуется провести разбиение с сохранением открытого равномерного узлового вектора.

Исходный открытый равномерный вектор имеет вид:

$$[X'] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2],$$

где два ненулевых интервала $0 \rightarrow 1$ и $1 \rightarrow 2$ соответствуют двум кусочно параболическим сегментам, составляющим В-сплайн. Предположим, что после разбиения мы должны иметь целочисленный равномерный вектор. Проведем преобразование параметра, умножая каждое узловое значение в $[X']$ на 2, и получим

$$[X] = [0 \ 0 \ 0 \ 2 \ 4 \ 4 \ 4]$$

с $x_1 = 0$, $x_2 = 0, \dots, x_7 = 4$. Кривая при этом не меняется.

Для разбиения кривой с учетом поставленных условий необходимо вставить узловые значения 1 и 3 в интервалы $0 \rightarrow 2$ и $2 \rightarrow 4$, соответственно. Запишем новый узловой вектор:

$$[Y] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 4 \ 4]$$

с $y_1 = 0$, $y_2 = 0, \dots, y_9 = 4$.

Теперь В-сплайн состоит из четырех параболических сегментов.

Из уравнения (5-119) получим шесть новых вершин C_j . Значения $\alpha_{i,j}^3$ вычисляются с помощью рекурсивных соотношений (5-120). В частности, из уравнения (5-120a) видно, что единственное ненулевое $\alpha_{i,j}^1$ первого порядка ($k = 1$) таково:

$$\alpha_{3,1}^1 = \alpha_{3,2}^1 = \alpha_{3,3}^1 = \alpha_{3,4}^1 = \alpha_{4,5}^1 = \alpha_{4,6}^1 = 1.$$

Из уравнения (5-120b) ненулевые $\alpha_{i,j}^2$ второго порядка ($k = 2$) таковы:

$j = 1$

$$\alpha_{3,1}^2 = \frac{y_2 - x_3}{x_4 - x_3} \alpha_{3,1}^1 = \frac{0 - 0}{2 - 0}(1) = 0,$$

$$\alpha_{2,1}^2 = \frac{x_4 - y_2}{x_4 - x_3} \alpha_{3,1}^1 = \frac{2 - 0}{2 - 0}(1) = 1,$$

$j = 2$

$$\alpha_{2,2}^2 = \frac{x_4 - y_3}{x_4 - x_3} \alpha_{3,1}^1 = \frac{2 - 0}{2 - 0}(1) = 1,$$

$$\alpha_{3,2}^2 = \frac{y_3 - x_3}{x_4 - x_3} \alpha_{3,2}^1 = \frac{0 - 0}{2 - 0}(1) = 0,$$

$j = 3$

$$\alpha_{3,3}^2 = \frac{y_4 - x_3}{x_4 - x_3} \alpha_{3,3}^1 = \frac{1 - 0}{2 - 0}(1) = \frac{1}{2},$$

$$\alpha_{2,3}^2 = \frac{x_4 - y_4}{x_4 - x_3} \alpha_{3,3}^1 = \frac{2 - 1}{2 - 0}(1) = \frac{1}{2},$$

$j = 4$

$$\alpha_{3,4}^2 = \frac{y_5 - x_3}{x_4 - x_3} \alpha_{3,4}^1 = \frac{2 - 0}{2 - 0}(1) = 1,$$

$$\alpha_{2,4}^2 = \frac{x_4 - y_5}{x_4 - x_3} \alpha_{3,4}^1 = \frac{2 - 2}{2 - 0}(1) = 0,$$

$j = 5$

$$\alpha_{4,5}^2 = \frac{y_6 - x_4}{x_5 - x_4} \alpha_{4,5}^1 = \frac{3 - 2}{4 - 2}(1) = \frac{1}{2},$$

$$\alpha_{3,5}^2 = \frac{x_5 - y_6}{x_5 - x_4} \alpha_{4,5}^1 = \frac{4 - 3}{4 - 2}(1) = \frac{1}{2},$$

$j = 6$

$$\alpha_{4,6}^2 = \frac{y_7 - x_4}{x_5 - x_4} \alpha_{4,6}^1 = \frac{4 - 2}{4 - 2}(1) = 1,$$

$$\alpha_{3,6}^2 = \frac{x_5 - y_7}{x_5 - x_4} \alpha_{4,6}^1 = \frac{4 - 4}{4 - 2}(1) = 0$$

третьего порядка ($k = 3$)

$j = 1$

$$\alpha_{2,1}^3 = \frac{y_3 - x_2}{x_4 - x_2} \alpha_{2,1}^2 = \frac{0 - 0}{2 - 0}(1) = 0,$$

$$\alpha_{1,1}^3 = \frac{x_4 - y_3}{x_4 - x_2} \alpha_{2,1}^2 = \frac{2 - 0}{2 - 0}(1) = 1,$$

$j = 2$

$$\alpha_{1,2}^3 = \frac{x_4 - y_4}{x_4 - x_2} \alpha_{2,2}^2 = \frac{2 - 1}{2 - 0}(1) = \frac{1}{2},$$

$$\alpha_{2,2}^3 = \frac{y_4 - x_2}{x_4 - x_2} \alpha_{2,2}^2 = \frac{1 - 0}{2 - 0}(1) = \frac{1}{2},$$

$j = 3$

$$\alpha_{3,3}^3 = \frac{y_5 - x_3}{x_5 - x_3} \alpha_{3,3}^2 = \frac{2 - 0}{4 - 0} \left(\frac{1}{2}\right) = \frac{1}{4},$$

$$\alpha_{2,3}^3 = \frac{y_5 - x_2}{x_4 - x_2} \alpha_{2,3}^2 + \frac{x_5 - y_5}{x_5 - x_3} \alpha_{3,3}^2 = \frac{2 - 0}{2 - 0} \left(\frac{1}{2}\right) + \frac{4 - 2}{4 - 0} \left(\frac{1}{2}\right) = \frac{1}{2} + \frac{1}{4} = \frac{3}{4},$$

$$\alpha_{1,3}^3 = \frac{x_4 - y_5}{x_4 - x_2} \alpha_{2,3}^2 = \frac{2 - 2}{2 - 0} \left(\frac{1}{2}\right) = 0,$$

$j = 4$

$$\alpha_{3,4}^3 = \frac{y_6 - x_3}{x_5 - x_3} \alpha_{3,4}^2 = \frac{3 - 0}{4 - 0}(1) = \frac{3}{4},$$

$$\alpha_{2,4}^3 = \frac{x_5 - y_6}{x_5 - x_3} \alpha_{3,4}^2 = \frac{4 - 3}{4 - 0}(1) = \frac{1}{4},$$

$j = 5$

$$\alpha_{2,5}^3 = \frac{x_5 - y_7}{x_5 - x_3} \alpha_{3,5}^2 = \frac{4 - 4}{4 - 0} \left(\frac{1}{2}\right) = 0,$$

$$\alpha_{3,5}^3 = \frac{y_7 - x_3}{x_5 - x_3} \alpha_{3,5}^2 + \frac{x_6 - y_7}{x_6 - x_4} \alpha_{4,5}^2 = \frac{4 - 0}{4 - 0} \left(\frac{1}{2}\right) + \frac{4 - 4}{4 - 2} \left(\frac{1}{2}\right) = \frac{1}{2},$$

$$\alpha_{4,5}^3 = \frac{y_7 - x_4}{x_6 - x_4} \alpha_{4,5}^2 = \frac{4 - 2}{4 - 2} \left(\frac{1}{2}\right) = \frac{1}{2},$$

$j = 6$

$$\alpha_{4,6}^3 = \frac{y_8 - x_4}{x_6 - x_4} \alpha_{4,6}^2 = \frac{4 - 2}{4 - 2}(1) = 1,$$

$$\alpha_{3,6}^3 = \frac{x_6 - y_8}{x_6 - x_4} \alpha_{4,6}^2 = \frac{4 - 4}{4 - 2}(1) = 0.$$

Новые вершины рассчитываются по уравнению (5-119):

$$C_1 = \sum_{i=1}^{n+1} \alpha_{i,1}^3 B_i = \alpha_{1,1}^3 B_1 + \alpha_{2,1}^3 B_2 + \alpha_{3,1}^3 B_3 + \alpha_{4,1}^3 B_4 = \\ = \alpha_{1,1}^3 B_1 = B_1 = [0 \quad 0].$$

Аналогично

$$C_2 = \frac{1}{2}(B_1 + B_2) = \frac{1}{2}([0 \quad 0] + [1 \quad 1]) = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix},$$

$$C_3 = \frac{3}{4}B_2 + \frac{1}{4}B_3 = \frac{3}{4}[1 \quad 1] + \frac{1}{4}[2 \quad 1] = \begin{bmatrix} \frac{5}{4} & 1 \end{bmatrix},$$

$$C_4 = \frac{1}{4}B_2 + \frac{3}{4}B_3 = \frac{1}{4}[1 \quad 1] + \frac{3}{4}[2 \quad 1] = \begin{bmatrix} \frac{7}{4} & 1 \end{bmatrix},$$

$$C_5 = \frac{1}{2}(B_3 + B_4) = \frac{1}{2}([2 \quad 1] + [3 \quad 0]) = \begin{bmatrix} \frac{5}{2} & \frac{1}{2} \end{bmatrix},$$

$$C_6 = B_4 = [3 \quad 0].$$

В результате увеличивается гибкость всей кривой.

В следующем примере в исходный открытый равномерный узловый вектор вносится один узел, после чего вектор становится неоднородным.

Пример 5-19 Локальное разбиение В-сплайна

Рассмотрим открытый В-сплайн третьего порядка ($k = 3$) из примера 5-18. Провести разбиение, добавив узловое значение 1 на интервале $0 \rightarrow 1$ вектора: $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$, где $x_1 = 0, \dots, x_7 = 2$. Новый узловый вектор имеет вид: $[Y] = [0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 2 \ 2]$, где $y_1 = 0, \dots, y_8 = 0$. Новый многоугольник имеет пять вершин $C_1 \dots C_5$.

Ненулевые $\alpha_{i,j}^k$, необходимые для расчета C_j :

$k = 1$

$$\alpha_{3,1}^1 = \alpha_{3,2}^1 = \alpha_{3,3}^1 = \alpha_{3,4}^1 = \alpha_{4,5}^1 = 1$$

$k = 2$

$$\alpha_{2,1}^2 = \alpha_{2,2}^2 = \alpha_{3,3}^2 = \alpha_{3,4}^2 = \alpha_{4,5}^2 = 1$$

$k = 3$

$$\alpha_{1,1}^3 = \alpha_{2,2}^3 = \alpha_{3,4}^3 = \alpha_{4,5}^3 = 1, \quad \alpha_{3,3}^3 = \alpha_{2,3}^3 = \frac{1}{2}.$$

Новые вершины:

$$C_1 = \alpha_{3,1}^3 B_1 = B_1 = [0 \quad 0],$$

$$C_2 = \alpha_{3,2}^3 B_2 = B_2 = [1 \quad 1],$$

$$C_3 = \alpha_{2,3}^3 B_2 + \alpha_{3,3}^3 B_3 = \frac{1}{2}(B_2 + B_3) = \frac{1}{2}([1 \quad 1] + [2 \quad 1]) = \begin{bmatrix} \frac{3}{2} & 1 \end{bmatrix},$$

$$C_4 = \alpha_{3,4}^3 B_3 = B_3 = [2 \quad 1],$$

$$C_5 = \alpha_{4,5}^3 B_4 = B_4 = [3 \quad 0].$$

Если C_3 сдвинуть до совпадения с C_2 , т. е. $C_2 = C_3 = [1 \quad 1]$, то получится как двойная вершина, так и двойной узел для $C_2 = C_3$. Для $k = 3$ В-сплайн имеет острый пик в $C_2 = C_3$ (см. задачу 5-30).

5-13 РАЦИОНАЛЬНЫЕ В-СПЛАЙНЫ

Впервые в машинной графике описание рациональных кривых и поверхностей было предложено в работе [5-32]. В литературе широко известны рациональные формы кубических сплайнов и кривых Безье ([5-33]—[5-37]), а также конических сечений [5-38]. Здесь мы рассмотрим только рациональные В-сплайны, так как они составляют общепринятую основу, а также из-за пространственных ограничений. Рациональные В-сплайны это единственное точное математическое представление, охватывающее все аналитические формы — прямые, плоскости, конические сечения, включающие окружности, кривые произвольной формы, квадрики и трехмерные поверхности, используемые в машинной графике и проектировании.

Первым рациональные В-сплайны изучил Весприлл [5-39]. Основа наших рассуждений приведена в работах [5-40] и [5-41]. Стоит отметить, что неравномерные рациональные В-сплайны (NURBS) с 1983 г. являются стандартом IGES (5-42). IGES — это стандарт обмена проектной информацией между системами машинного проектирования, а также между ними и системами автоматизированного производства. Рациональные В-сплайны применяются в ряде систем геометрического моделирования и реализованы аппаратно (СБИС или микрокод) в некоторых графических рабочих станциях.

Рациональный В-сплайн это проекция нерационального (полиномиального) В-сплайна, определенного в четырехмерном (4D) однородном координатном пространстве, на трехмерное (3D) физическое пространство. В частности

$$P(t) = \sum_{i=1}^{n+1} B_i^k N_{i,k}(t), \quad (5-121)$$

где B_i^k — вершины многоугольника для нерационального 4D В-сплайна в четырехмерном пространстве, $N_{i,k}(t)$ — функция базиса нерационального В-сплайна из уравнения (5-84).

Рациональный В-сплайн получается после проектирования, т. е. деления на однородную координату

$$P(t) = \frac{\sum_{i=1}^{n+1} B_i h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} = \sum_{i=1}^{n+1} B_i R_{i,k}(t), \quad (5-122)$$

где B_i — вершины трехмерного многоугольника для рационального В-сплайна, а

$$R_{i,k}(t) = \frac{h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} \quad (5-123)$$

— базисные функции рационального В-сплайна. Здесь $h_i \geq 0$ для всех i^1 .

¹ Базисные функции рационального В-сплайна могут быть определены и для $h_i < 0$ (см. [5-39]), но в данном случае это неудобно.

Как видно из уравнений (5-121) — (5-123), рациональные В-сплайны и их базисы это обобщение нерациональных В-сплайнов и базисов. Они наследуют почти все аналитические и геометрические свойства последних. В частности:

Каждая функция рационального базиса положительна или равна нулю для всех значений параметра, т. е. $R_{i,k} \geq 0$.

Для любого значения параметра t сумма базисных функций рационального В-сплайна равна единице, т. е.

$$\sum_{i=1}^{n+1} R_{i,k}(t) \equiv 1. \quad (5-124)$$

Кроме $k = 1$ каждая рациональная базисная функция имеет ровно один максимум.

Рациональный В-сплайн порядка k (степени $k - 1$) везде C^{k-2} непрерывен. Максимальный порядок рационального В-сплайна равен количеству вершин определяющего многоугольника.

Рациональный В-сплайн обладает свойством уменьшения вариации. Общая форма рационального В-сплайна повторяет очертания определяющего многоугольника.

Любое *проективное* преобразование рационального В-сплайна производится соответствующим преобразованием вершин определяющего многоугольника; т. е. кривая инвариантна относительно *проективного* преобразования. Это более сильное условие, чем для нерационального В-сплайна, который инвариантен только относительно *аффинного* преобразования.

Из уравнений (5-85) и (5-123) видно, что при всех $h_i = 1$, $R_{i,k}(t) = N_{i,k}(t)$, т. е. нерациональные В-сплайны и их базисы являются частным случаем рациональных. Легко показать, что открытый рациональный В-сплайн с порядком, равным количеству вершин определяющего многоугольника, представляет собой рациональную кривую Безье. В случае $h_i = 1$ рациональная кривая Безье сводится к нерациональной. Таким образом, В-сплайны включают как рациональные, так и нерациональные кривые Безье.

Рациональные В-сплайны — это четырехмерное обобщение нерациональных В-сплайнов; поэтому алгоритмы повышения степени (работа [5-27] и пример 6-18), разбиения (разд. 5-12 и работы [5-28] — [5-31]) и подгонки (разд. 5-11) для нерациональных В-сплайнов справедливы и для рациональных при распространении их на четырехмерное пространство.

Рациональные В-сплайны и их базисы можно получить с помощью открытых равномерных, периодических равномерных и неравномерных узловых векторов.

Однородные координаты h_i (также называемые весами) в уравнениях (5-122) и (5-123) предоставляют дополнительные возможности изгиба кривой; $h = 1$ называется аффинным пространством и соответствует физическому пространству. На рис. 5-57 показано влияние однородных координат h на базисные функции рационального В-сплайна, где $[0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3] \ (n+1 = 5, k = 3)$ — открытый равномерный узловой вектор, $h_i = 1, i+3$ — вектор однородных координат. Значе-

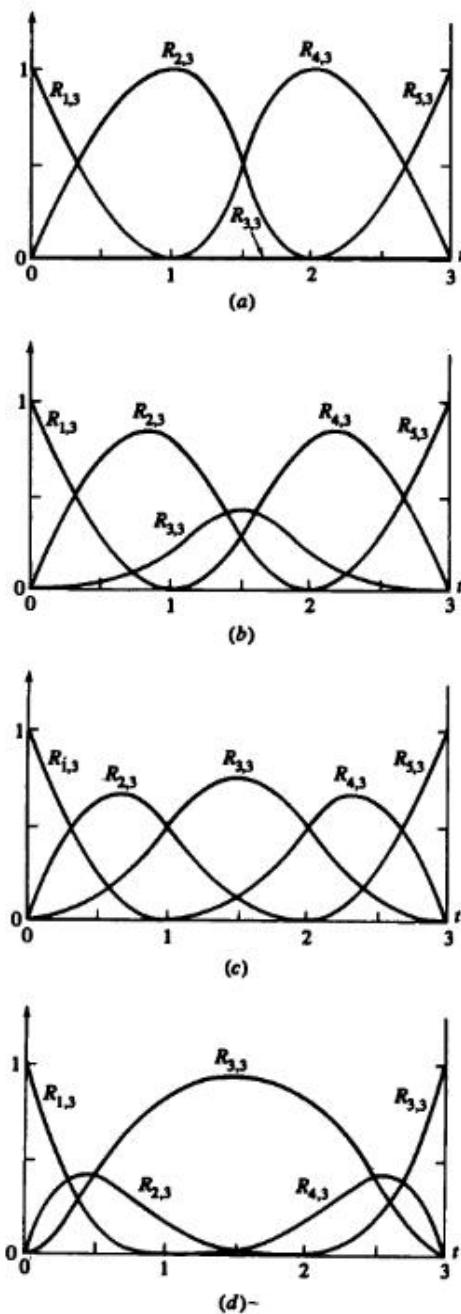


Рис. 5-57 Базисные функции рационального B-сплайна для $n+1 = 5$, $k = 3$ с открытым узловым вектором $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$. (а) $h_3 = 0$; (б) $h_3 = 0.25$; (в) $h_3 = 1.0$; (г) $h_3 = 5.0$.

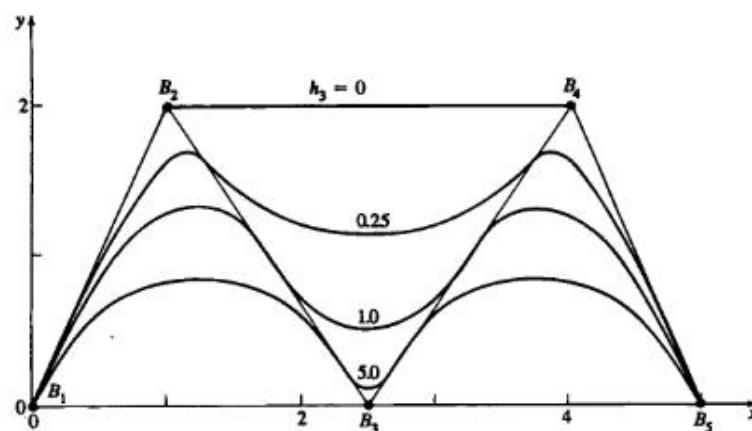


Рис. 5-58 Рациональные B-сплайны для $n+1 = 5$, $k = 3$ с открытым узловым вектором $[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.

ния h_3 лежат в диапазоне от 0 до 5. Базисные функции рационального B-сплайна на рис. 5.57с при $h = 1$ совпадают с соответствующими функциями нерационального B-сплайна. Рациональный B-сплайн для $h_3 = 1$ на рис. 5-58 также совпадает с соответствующим нерациональным. Отметим, что для $h_3 = 0$ (рис. 5-57а) $R_{3,3}$ везде равно нулю; т. е. соответственная вершина B_3 не оказывает никакого влияния на форму соответствующей кривой. Это показано на рис. 5-58, где вершины определяющего многоугольника B_2 и B_4 соединены прямой. Из рис. 5-57 также видно, что при увеличении h_3 также возрастает $R_{3,3}$ и, вследствие уравнения (5-124), $R_{2,3}$ и $R_{4,3}$ уменьшаются. На рис. 5-58 изображено воздействие на соответствующие рациональные B-сплайны. В частности отметим, что с увеличением h_3 кривая приближается к B_3 . Отсюда, как уже отмечалось, следует, что однородные координаты дают возможность увеличить гибкость кривой. Иллюстрация аналогичных свойств рациональных B-сплайнов и базисов четвертого порядка ($k = 4$) дана на рис. 5-59 и 5-60. Однако для кривых более высокого порядка на рис. 5-60 кривая при $h_3 = 0$ не вырождается в отрезок прямой между B_2 и B_4 .

На рис. 5-61 изображен периодический равномерный базис для $n+1 = 5$, $k = 3$ и узлового вектора $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$ и вектора однородных координат $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$ с $0 \leq h_3 \leq 5$. Здесь, как и в случае нерационального базиса, допустимый диапазон параметра $2 \leq t \leq 5$. На рис. 5-61 показан только этот диапазон. Для $h_3 = 1$ базисные функции рационального B-сплайна совпадают с соответствующими нерациональными, но для $h_3 \neq 1$ они становятся непериодическими и не являются параллельным переносом друг друга. На рис. 5-62 приведены соответствующие рациональные B-сплайны. Отметим, что конечные точки всех кривых совпадают.

На рис. 5-63 и 5-64 показаны соответствующие B-сплайны четвертого порядка ($k = 4$) и их базисы. Отметим, что начальные и конечные точки кривых лежат на прямых.

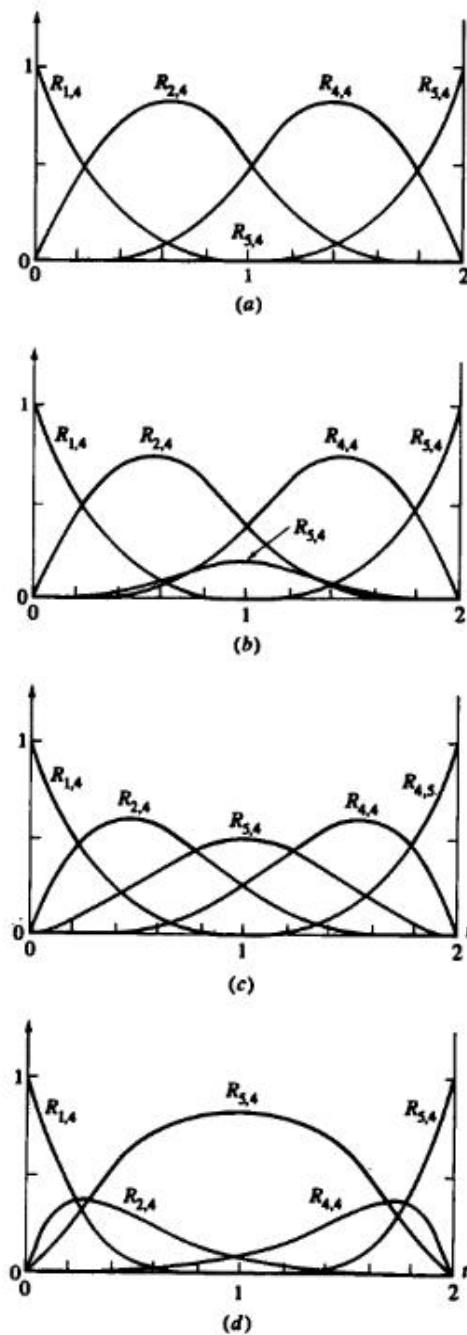


Рис. 5-59 Базисные функции рационального В-сплайна для $n+1 = 5$, $k = 4$ с открытым узловым вектором $[X] = [0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2 \ 2]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$. (а) $h_3 = 0$; (б) $h_3 = 0.25$; (в) $h_3 = 1.0$; (г) $h_3 = 5.0$.

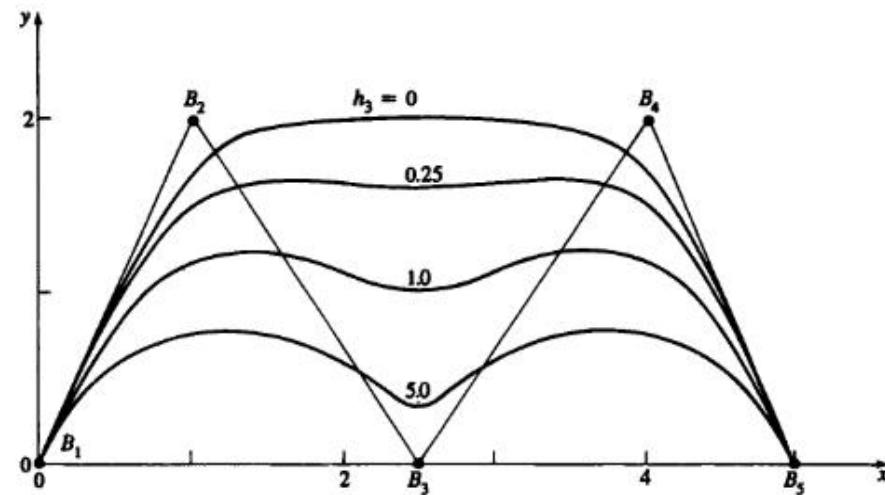


Рис. 5-60 Рациональные В-сплайны для $n+1 = 5$, $k = 4$ с открытым узловым вектором $[X] = [0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2 \ 2]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.

Вычисляя уравнения (5-122) и (5-123) для конечных точек кривой с аргументом $(t_{\max} - \epsilon)_{\epsilon \rightarrow 0}$ из примера 5-12, получаем, что первая и последняя точки открытого рационального В-сплайна совпадают с конечными точками определяющего многоугольника:

$$P(0) = B_1 \quad \text{и} \quad P(t_{\max}) = P(n - k + 2) = B_{n+1}.$$

Из рис. 5-65 видно, что перемещение одной вершины многоугольника влияет на кривую так же, как и в случае рациональных В-сплайнов. В данном случае $[H] = [1 \ 1 \ 0.25 \ 1 \ 1]$. Если $h_3 = 0$, то передвижение B_3 не оказывает никакого воздействия. С увеличением h_3 растет влияние движения точки B_3 .

На рис. 5-66 показано воздействие кратных вершин в B_3 на рациональный В-сплайн четвертого порядка. Как и для нерациональных кривых, $k - 1$ совпадающая вершина приводит к появлению острого угла или пика. Далее, кратная вершина порождает ребра нулевой длины, поэтому существование угла не зависит от значений $h_3 \geq 0$, соответствующих ей (см. задачу 5-33).

Рассмотрим пример расчета рационального В-сплайна.

Пример 5-20 Расчет открытых рациональных В-сплайнов

Пусть дан многоугольник с вершинами $B_1[0 \ 1]$, $B_2[1 \ 2]$, $B_3[2.5 \ 0]$, $B_4[4 \ 2]$, $B_5[5 \ 0]$. Найти точку при $t = 3/2$ на открытом рациональном В-сплайте третьего порядка ($k = 3$) с однородными векторами, заданными $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$, $h_3 = 0, 1/4, 1, 5$.

Узловой вектор $-[0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3]$. Диапазон параметра t : $0 \leq t \leq 3$. Кривые состоят из трех кусочно рациональных квадратичных кривых, по одной на каждом внутреннем интервале узлового вектора.

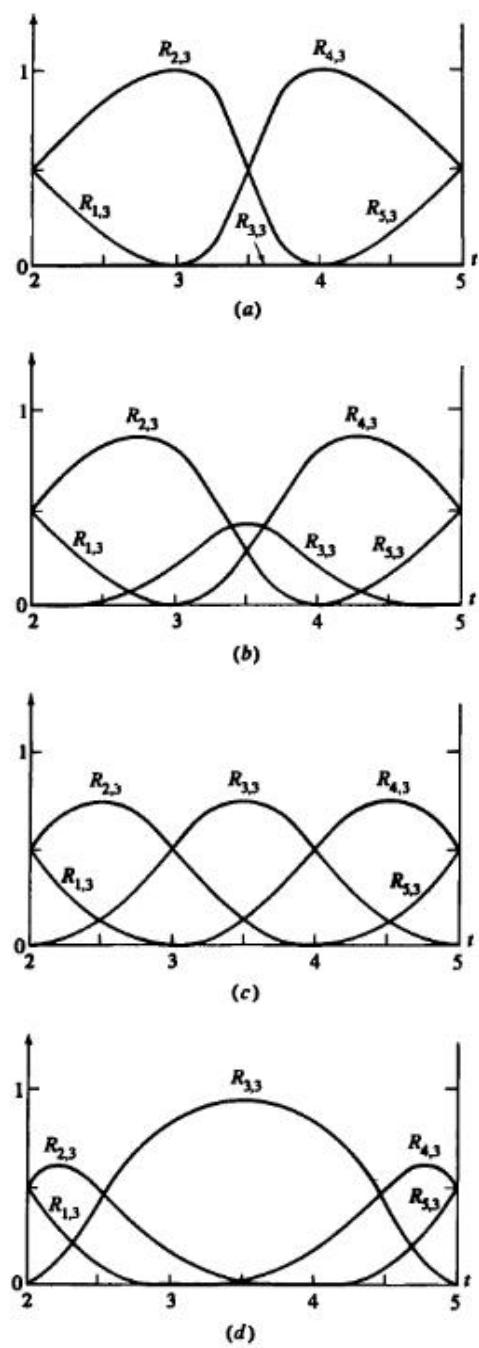


Рис. 5-61 Базисные функции рационального В-сплайна для $n + 1 = 5$, $k = 3$ с периодическим узловым вектором $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.
 (a) $h_3 = 0$; (b) $h_3 = 0.25$; (c) $h_3 = 1.0$; (d) $h_3 = 5.0$.

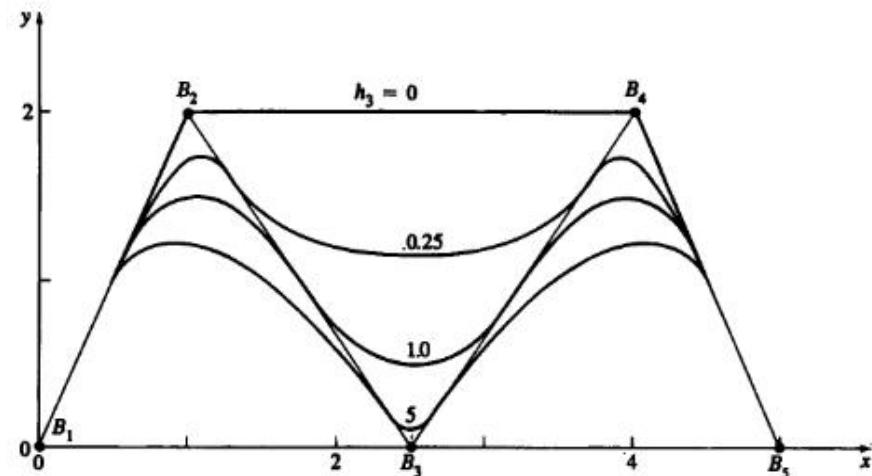


Рис. 5-62 Рациональные В-сплайны для $n + 1 = 5$, $k = 4$ с периодическим узловым вектором $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.

Из уравнения (5-84) на интервале $1 \leq t \leq 2$ получим базисные функции нерационального В-сплайна:

$$1 \leq t < 2$$

$$N_{4,1}(t) = 1; \quad N_{i,1}(t) = 0, \quad i \neq 4$$

$$N_{3,2}(t) = (2 - t); \quad N_{4,2}(t) = (t - 1); \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N_{2,3}(t) = \frac{(2 - t)^2}{2}; \quad N_{3,3}(t) = \frac{t(2 - t)}{2} + \frac{(3 - t)(t - 1)}{2};$$

$$N_{4,3}(t) = \frac{(t - 1)^2}{2}; \quad N_{i,3}(t) = 0, \quad i \neq 2, 3, 4$$

Отсюда и из уравнения (5-123) после определения знаменателя

$$\begin{aligned} S &= \sum_{i=1}^{n+1} h_i N_{i,k}(t) = h_1 N_{1,3}(t) + h_2 N_{2,3}(t) + h_4 N_{4,3}(t) + h_5 N_{5,3}(t) = \\ &= h_2 N_{2,3}(t) + h_4 N_{4,3}(t) = \\ &= \frac{(2 - t)^2}{2} + \frac{(t - 1)^2}{2} = \frac{2t^2 - 6t + 5}{2} \end{aligned}$$

получим базис рационального В-сплайна:

$$1 \leq t < 2$$

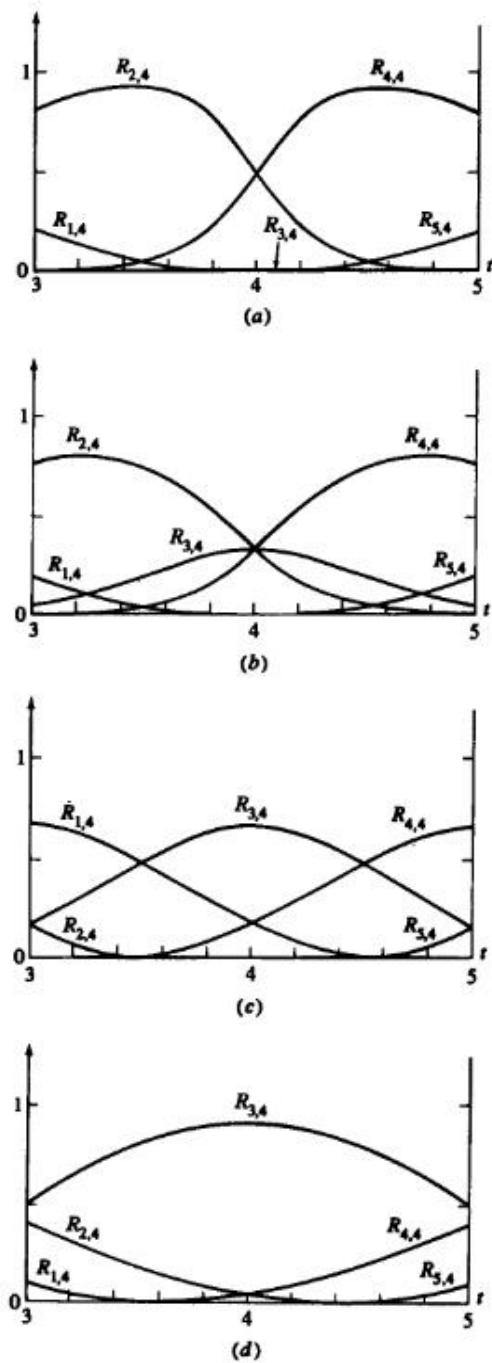


Рис. 5-63 Базисные функции рационального В-сплайна для $n+1 = 5$, $k = 4$ с периодическим узловым вектором $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.
 (a) $h_3 = 0$; (b) $h_3 = 0.25$; (c) $h_3 = 1.0$; (d) $h_3 = 5.0$.

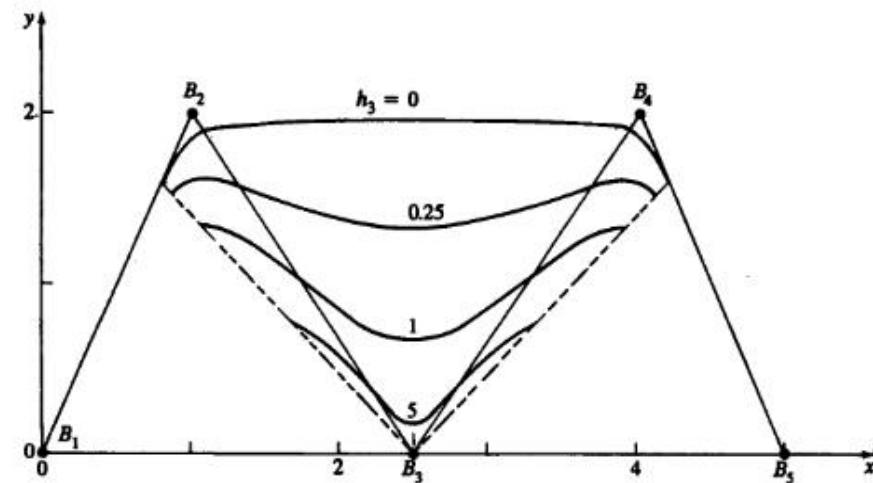


Рис. 5-64 Рациональные В-сплайны для $n+1 = 5$, $k = 4$ с периодическим узловым вектором $[X] = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, $[H] = [1 \ 1 \ h_3 \ 1 \ 1]$.

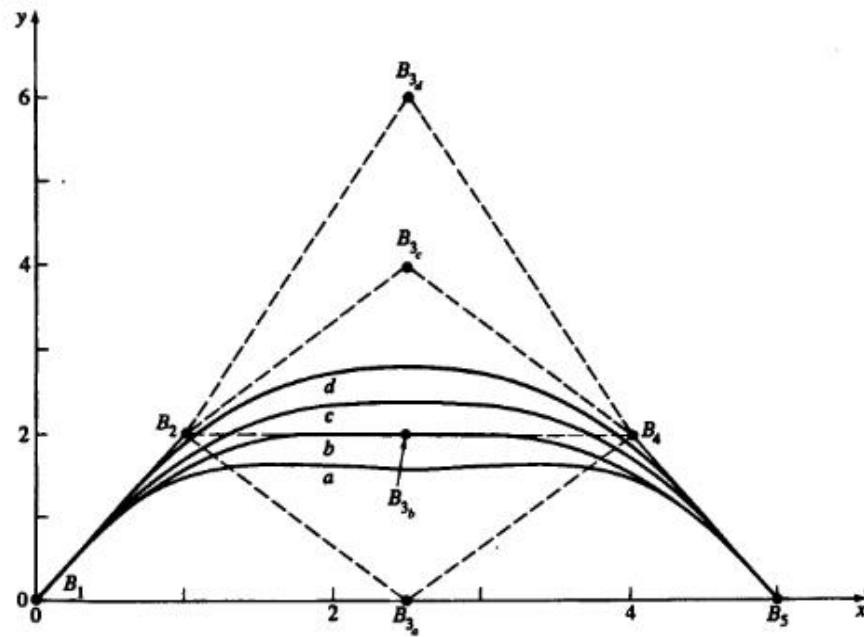


Рис. 5-65 Влияние перемещения одной вершины многоугольника на рациональный В-сплайн $n+1 = 5$, $k = 4$, $[H] = [1 \ 1 \ 0.25 \ 1 \ 1]$.

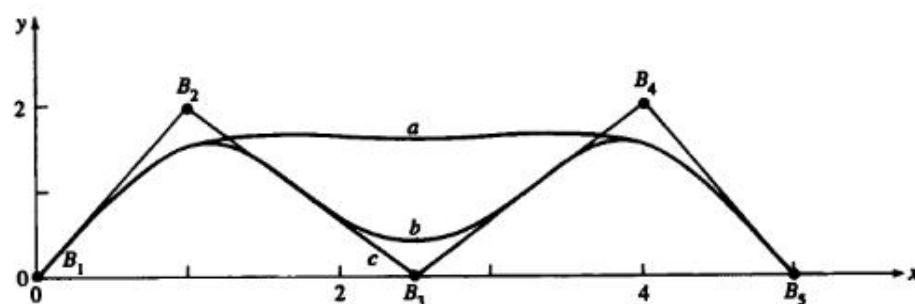


Рис. 5-66 Влияние кратной вершины многоугольника B_3 на рациональный B-сплайн $n+1 = 5, k = 4$. (a) Одна вершина $[H] = [1 \ 1 \ 0.25 \ 1 \ 1]$; (b) двойная вершина $[H] = [1 \ 1 \ 0.25 \ 0.25 \ 1 \ 1]$; (c) тройная вершина $[H] = [1 \ 1 \ 0.25 \ 0.25 \ 0.25 \ 1 \ 1]$.

$$h_3 = 0$$

$$R_{1,3}(t) = 0$$

$$R_{2,3}(t) = \frac{h_2 N_{2,3}(t)}{S} = \frac{(2-t)^2}{2t^2 - 6t + 5}$$

$$R_{3,3}(t) = 0$$

$$R_{4,3}(t) = \frac{h_4 N_{4,3}(t)}{S} = \frac{(t-1)^2}{2t^2 - 6t + 5}$$

$$R_{5,3}(t) = 0$$

$$h_3 = 1/4$$

$$\begin{aligned} S &= h_2 N_{2,3}(t) + h_3 N_{3,3}(t) + h_4 N_{4,3}(t) \\ &= \frac{(2-t)^2}{2} + \frac{t(2-t)}{8} + \frac{(3-t)(t-1)}{8} + \frac{(t-1)^2}{2} \\ &= \frac{6t^2 - 18t + 17}{8} \end{aligned}$$

$$R_{1,3}(t) = 0$$

$$R_{2,3}(t) = \frac{4(2-t)^2}{6t^2 - 18t + 17}$$

$$R_{3,3}(t) = \frac{t(2-t) + (3-t)(t-1)}{6t^2 - 18t + 17} = \frac{-2t^2 + 6t - 3}{6t^2 - 18t + 17}$$

$$R_{4,3}(t) = \frac{4(t-1)^2}{6t^2 - 18t + 17}$$

$$R_{5,3}(t) = 0$$

$$h_3 = 1$$

$$S = 1$$

$$R_{1,3}(t) = 0$$

$$R_{2,3}(t) = N_{2,3}(t) = \frac{(2-t)^2}{2}$$

$$R_{3,3}(t) = N_{3,3}(t) = \frac{t(2-t)}{2} + \frac{(3-t)(t-1)}{2}$$

$$R_{4,3}(t) = N_{4,3}(t) = \frac{(t-1)^2}{2}$$

$$R_{5,3}(t) = 0$$

$$h_3 = 5$$

$$S = \frac{(2-t)^2}{2} + \frac{5t(2-t)}{2} + \frac{5(3-t)(t-1)}{2} + \frac{(t-1)^2}{2} = -4t^2 + 12t - 5$$

$$R_{1,3}(t) = 0$$

$$R_{2,3}(t) = \frac{(2-t)^2}{2(-4t^2 + 12t - 5)}$$

$$R_{3,3}(t) = \frac{5t(2-t) + 5(3-t)(t-1)}{2(-4t^2 + 12t - 5)} = \frac{5(-2t^2 + 6t - 3)}{2(-4t^2 + 12t - 5)}$$

$$R_{4,3}(t) = \frac{(t-1)^2}{2(-4t^2 + 12t - 5)}$$

$$R_{5,3}(t) = 0$$

Полностью результаты приведены на рис. 5-57.

При $t = 3/2$

$$\begin{aligned} h_3 = 0 : \quad R_{1,3}(3/2) &= 0; \quad R_{2,3}(3/2) = \frac{1}{2}; \quad R_{3,3}(3/2) = 0; \\ R_{4,3}(3/2) &= \frac{1}{2}; \quad R_{5,3}(3/2) = 0 \end{aligned}$$

$$\begin{aligned} h_3 = \frac{1}{4} : \quad R_{1,3}(3/2) &= 0; \quad R_{2,3}(3/2) = \frac{2}{7}; \quad R_{3,3}(3/2) = \frac{3}{7}; \\ R_{4,3}(3/2) &= \frac{2}{7}; \quad R_{5,3}(3/2) = 0 \end{aligned}$$

$$\begin{aligned} h_3 = 1 : \quad R_{1,3}(3/2) &= 0; \quad R_{2,3}(3/2) = \frac{1}{8}; \quad R_{3,3}(3/2) = \frac{3}{4}; \\ R_{4,3}(3/2) &= \frac{1}{8}; \quad R_{5,3}(3/2) = 0 \end{aligned}$$

$$\begin{aligned} h_3 = 5 : \quad R_{1,3}(3/2) &= 0; \quad R_{2,3}(3/2) = \frac{1}{32}; \quad R_{3,3}(3/2) = \frac{15}{16}; \\ R_{4,3}(3/2) &= \frac{1}{32}; \quad R_{5,3}(3/2) = 0 \end{aligned}$$

Соответственные точки на рациональных B-сплайнах:

$$h_3 = 0 : \quad P(3/2) = \frac{1}{2}[1 \ 2] + \frac{1}{2}[4 \ 2] = \left[\begin{array}{cc} \frac{5}{2} & 2 \end{array} \right]$$

$$h_3 = \frac{1}{4} : \quad P(3/2) = \frac{2}{7}[1 \ 2] + \frac{3}{7}\left[\begin{array}{cc} \frac{5}{2} & 0 \end{array} \right] + \frac{2}{7}[4 \ 2] = \left[\begin{array}{cc} \frac{5}{2} & \frac{8}{7} \end{array} \right]$$

$$h_3 = 1 : \quad P(3/2) = \frac{1}{8}[1 \ 2] + \frac{3}{4}\left[\begin{array}{cc} \frac{5}{2} & 0 \end{array} \right] + \frac{1}{8}[4 \ 2] = \left[\begin{array}{cc} \frac{5}{2} & \frac{1}{2} \end{array} \right]$$

$$h_3 = 5 : \quad P(3/2) = \frac{1}{32}[1 \ 2] + \frac{15}{16}\left[\begin{array}{cc} \frac{5}{2} & 0 \end{array} \right] + \frac{1}{32}[4 \ 2] = \left[\begin{array}{cc} \frac{5}{2} & \frac{1}{8} \end{array} \right]$$

Полностью результаты приведены на рис. 5-58.

Формально дифференцируя уравнения (5-122) и (5-123), получаем производные рациональных B-сплайнов, а именно

$$P'(t) = \sum_{i=1}^{n+1} B_i R'_{i,k}(t) \quad (5-125)$$

где

$$R'_{i,k}(t) = \frac{h_i N'_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}} - \frac{h_i N_{i,k} \sum_{i=1}^{n+1} h_i N'_{i,k}}{\left(\sum_{i=1}^{n+1} h_i N_{i,k}\right)^2}. \quad (5-126)$$

Подставим $t = 0$ и $t = n - k + 2$:

$$P'(0) = (k-1) \frac{h_2}{h_1} (B_2 - B_1), \quad (5-127)$$

$$P'(n-k+2) = (k-1) \frac{h_n}{h_{n+1}} (B_{n+1} - B_n), \quad (5-128)$$

т.е. наклон кривой в концах совпадает с наклоном первого и последнего ребер многоугольника, соответственно.

Аналогично рассчитываются производные более высокого порядка (см. задачи 5-35 и 5-36).

Рассмотрим простой пример.

Пример 5-21 Производные открытых рациональных сплайновПусть дан определяющий многоугольник из примера 5-16 с вершинами $B_1[1 \ 1], B_2[2 \ 3], B_3[4 \ 3], B_4[3 \ 1]$. Найти первую производную рационального В-сплайна второго порядка ($k = 2$) с $[H] = [1 \ 1/2 \ 1 \ 1]$.Узловой вектор $[X] = [0 \ 0 \ 1 \ 2 \ 3 \ 3]$. Параметр изменяется в диапазоне $0 \leq t \leq 3$. Из уравнения (5-125) первая производная

$$P'(t) = B_1 R'_{1,2}(t) + B_2 R'_{2,2}(t) + B_3 R'_{3,2}(t) + B_4 R'_{4,2}(t)$$

Из уравнений (5-12) и (5-16) получаем иррациональные базисные функции и их производные:

$$0 \leq t < 1$$

$$N_{1,2}(t) = 1-t; \quad N_{2,2}(t) = t; \quad N_{i,2}(t) = 0, \quad i \neq 1, 2$$

$$N'_{1,2}(t) = -1; \quad N'_{2,2}(t) = 1; \quad N'_{i,2}(t) = 0, \quad i \neq 1, 2$$

Используя уравнение (5-126), найдем рациональный базис и его производные

$$\sum_{i=1}^{n+1} h_i N_{i,2} = \frac{2-t}{2}; \quad \sum_{i=1}^{n+1} h_i N'_{i,2} = -\frac{1}{2};$$

$$R'_{1,2}(t) = \frac{2}{2-t} \left\{ (-1) - (1-t) \frac{(-1)}{(2-t)} \right\} = \frac{-2}{(2-t)^2}$$

$$R'_{2,2}(t) = \left(\frac{1}{2}\right) \left(\frac{2}{2-t}\right) \left\{ 1-t \frac{(-1)}{(2-t)} \right\} = \frac{2}{(2-t)^2}$$

$$R'_{i,2}(t) = 0, \quad i \neq 1, 2$$

Итак

$$P'(t) = \frac{2}{(2-t)^2} (B_2 - B_1)$$

Отметим, что в отличие от иррационального В-сплайна из примера 5-16, при сохранении наклона равным наклону первого ребра значение производной теперь изменяется вдоль кривой. При $t = 0$, т. е. в начале кривой

$$P'(t) = \frac{1}{2} (B_2 - B_1)$$

величина производной в два раза меньше, чем у иррационального В-сплайна из примера 5-16.

На интервале

$$1 \leq t < 2$$

$$N'_{2,2}(t) = 2-t; \quad N_{3,2}(t) = t-1; \quad N_{i,2}(t) = 0, \quad i \neq 2, 3$$

$$N'_{2,2}(t) = -1; \quad N'_{3,2}(t) = 1; \quad N'_{i,2}(t) = 0, \quad i \neq 2, 3$$

Здесь

$$\sum_{i=1}^{n+1} h_i N_{i,2} = \frac{t}{2}; \quad \sum_{i=1}^{n+1} h_i N'_{i,2} = \frac{1}{2};$$

$$R'_{2,2}(t) = \left(\frac{1}{2}\right) \left(\frac{2}{t}\right) \left\{ (-1) - (2-t) \left(\frac{1}{t}\right) \right\} = -\frac{2}{t^2}$$

$$R'_{3,2}(t) = \frac{2}{t} \left\{ (1) - (t-1) \left(\frac{1}{t}\right) \right\} = \frac{2}{t^2}$$

$$R'_{i,2}(t) = 0, \quad i \neq 2, 3$$

Итак

$$P'(t) = \frac{2}{t^2} (B_3 - B_2)$$

На интервале

$$2 \leq t < 3$$

$$N_{3,2}(t) = 3-t; \quad N_{4,2}(t) = t-2; \quad N_{i,2}(t) = 0, \quad i \neq 3, 4$$

$$N'_{3,2}(t) = -1; \quad N'_{4,2}(t) = 1; \quad N'_{i,2}(t) = 0, \quad i \neq 3, 4$$

При этом

$$\sum_{i=1}^{n+1} h_i N_{i,2} = 1; \quad \sum_{i=1}^{n+1} h_i N'_{i,2} = 0;$$

$$R'_{3,2}(t) = N'_{3,2}(t) = -1; \quad R'_{4,2}(t) = N'_{4,2}(t) = 1;$$

$$R'_{i,2}(t) = 0, \quad i \neq 3, 4$$

То есть

$$P'(t) = B_4 - B_3$$

При $t = (3 - \varepsilon)$, $\varepsilon \rightarrow 0$, получаем тот же результат, что и при $t = 3$.

Как уже отмечалось, с помощью рациональных B-сплайнов можно представить все конические сечения. Кроме того, это единственное математическое описание, способное задать произвольную кривую, состоящую из конических сечений. Коническим сечениям соответствуют квадратичные уравнения, поэтому удобно сначала рассмотреть квадратичный рациональный B-сплайн ($k = 3$), заданный тремя вершинами ($n + 1 = 3$) с узловым вектором $[X] = [0 \ 0 \ 1 \ 1 \ 1]$:

$$P(t) = \frac{h_1 N_{1,3}(t)B_1 + h_2 N_{2,3}(t)B_2 + h_3 N_{3,3}(t)B_3}{h_1 N_{1,3}(t) + h_2 N_{2,3}(t) + h_3 N_{3,3}(t)}, \quad (5-129)$$

что фактически представляет собой рациональную кривую Безье третьего порядка (см. рис. 5-67). Предположим, что $h_1 = h_3 = 1$. Тогда уравнение (5-129) принимает вид

$$P(t) = \frac{N_{1,3}(t)B_1 + h_2 N_{2,3}(t)B_2 + N_{3,3}(t)B_3}{N_{1,3}(t) + h_2 N_{2,3}(t) + N_{3,3}(t)}. \quad (5-130)$$

При $h_2 = 0$ точки B_1 и B_3 соединены прямой линией. При $h_2 \rightarrow \infty$ повторяется контур определяющего многоугольника. Если $h_2 = 0$ и $t = 1/2$, достигается середина отрезка B_1B_2 , обозначенная на рис. 5-67 буквой M . Аналогично при $h_2 \rightarrow \infty$, $t = 1/2$, достигается точка B_2 . Для $0 < h_2 < \infty$ точка S , соответствующая точке кривой $P(t)$ при $t = 1/2$, движется по прямой линии между M и B_2 . S называется плечевой точкой. Значение h_2 определяет тип конического сечения. В работе [5-43] показано, что результат при

- $h_2 = 0$ — прямая линия,
- $0 < h_2 < 1$ — эллиптический сегмент,
- $h_2 = 1$ — параболический сегмент,
- $h_2 > 1$ — гиперболический сегмент.

Воспользуемся уравнением (5-84) и подставим $t = 1/2$ в уравнение (5-130):

$$P(t) = \frac{(1-t)^2 B_1 + 2h_2 t(1-t)B_2 + t^2 B_3}{(1-t)^2 + 2h_2 t(1-t) + t^2}.$$

При $t = 1/2$, $P(t) = S$, что дает

$$S = \frac{1}{1+h_2} \frac{B_1 + B_3}{2} + \frac{h_2}{1+h_2} B_2 = \frac{M}{1+h_2} + \frac{h_2}{1+h_2} B_2. \quad (5-131)$$

Запишем параметрическое уравнение прямой между M и B_2

$$S = (1-s)M + sB_2, \quad (5-132)$$

где s — параметр. Приравнивая коэффициенты уравнений (5-131) и (5-132), получаем

$$s = \frac{h_2}{1+h_2} \quad \text{и} \quad h_2 = \frac{s}{1-s} = \frac{M-S}{S-B_2}. \quad (5-133)$$

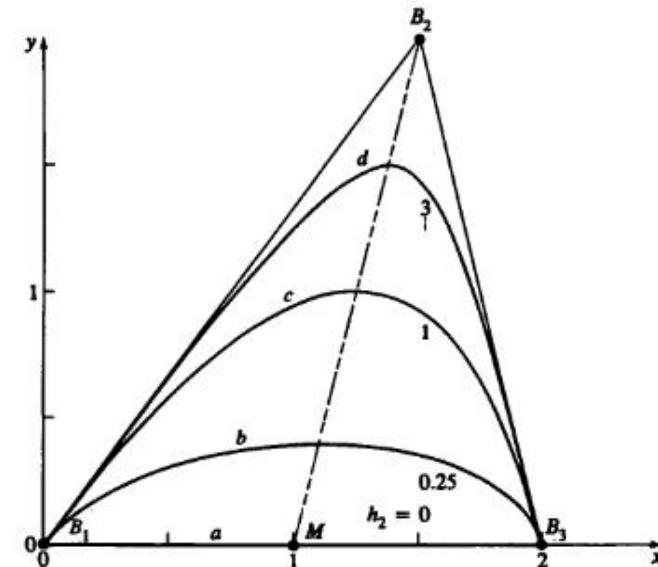


Рис. 5-67 Конические сечения, заданные рациональными B-сплайнами (кривыми Безье); (a) прямая линия, $h_2 = 0$; (b) эллипс, $h_2 = 0.25$; (c) парабола, $h_2 = 1$; (d) гипербола, $h_2 = 3$.

Параметр s определяет форму кривой и тип конического сечения, т. е. это удобное средство разработки.

Так как окружность является частным случаем эллипса, можно подобрать такое значение h_2 , что уравнение (5-130) будет уравнением дуги окружности. Из-за симметрии B_1 , B_2 и B_3 для дуги окружности формируют равнобедренный треугольник, как показано на рис. 5-68. Требуемое значение h_2 , определяется из рис. 5-68.

Треугольник $B_1B_2B_3$ равнобедренный, поэтому S — максимум кривой и касательная в точке S параллельна линии B_1B_3 . Треугольник B_1qS также равнобедренный с равными углами $\angle SB_1q$ и $\angle B_1Sq$ у основания, обозначенными $\theta/2$ на рис. 5-68. Из параллельности касательной S линии B_1B_3 следует, что углы $\angle qSB_1$ и $\angle SB_1M$ равны. Поэтому угол $\angle SB_1M = e/2$ равен половине угла у основания треугольника B_1 , B_2 и B_3 .

Отсюда и из уравнения (5-133) получаем

$$h_2 = \frac{M-S}{S-B_2} = \frac{e \operatorname{tg}(\frac{\theta}{2})}{f \sin \theta - e \operatorname{tg}(\frac{\theta}{2})}.$$

Подставляя тождество $(\theta/2) = \sin \theta / (1 + \cos \theta)$, получаем

$$h_2 = \frac{\frac{e \sin \theta}{1+\cos \theta}}{f \sin \theta - \frac{e \sin \theta}{1+\cos \theta}} = \frac{e}{f(1+\cos \theta) - e} = \frac{e}{f} = \cos \theta \quad (5-134)$$

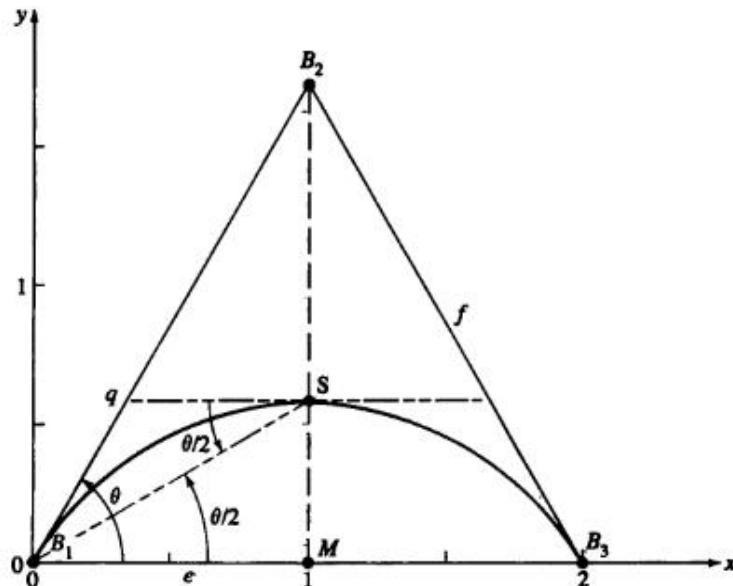


Рис. 5-68 Дуга окружности, построенная как рациональный В-сплайн.

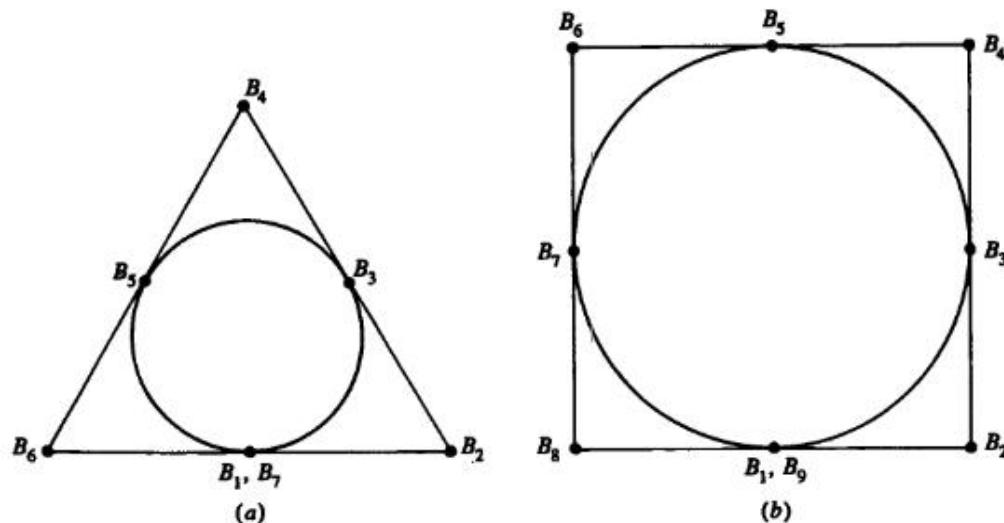
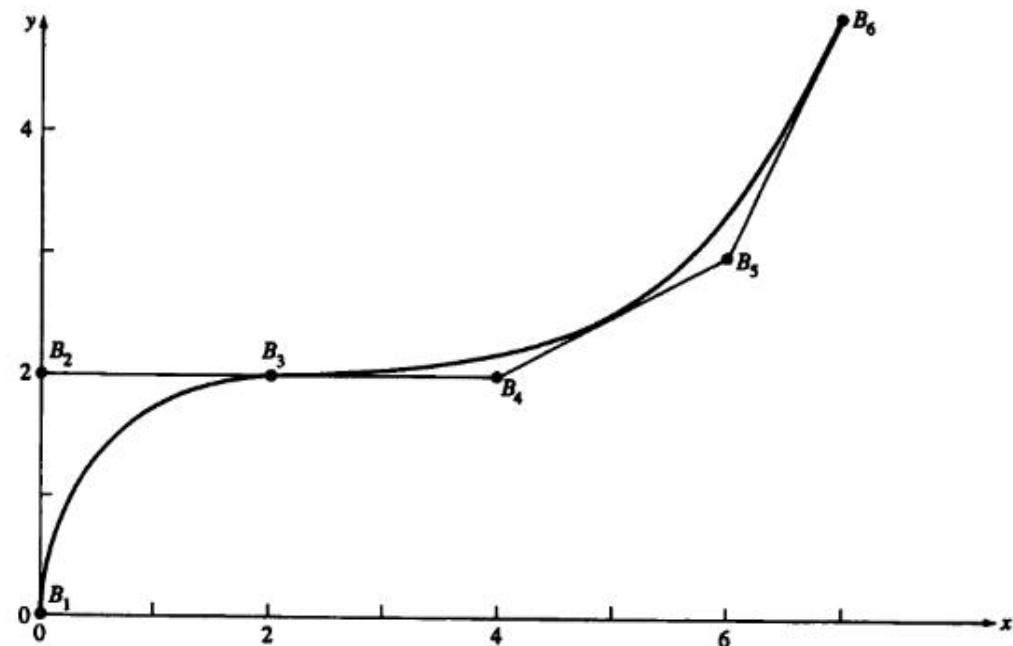
Рис. 5-69 Рациональные В-сплайны — окружности. (а) Три сегмента по 120° ; (б) четыре сегмента по 90° .

Рис. 5-70 Сопряжение рациональных В-сплайнов.

Дуга покрывает часть окружности для удвоенного угла θ . Для дуги 120° угол $\theta = 60^\circ$ и $h_2 = 1/2$. Для данного случая радиус окружности равен $2(S - M)$.

Полная окружность получается соединением нескольких сегментов. В частности, окружность можно составить из трех сегментов рационального квадратичного В-сплайна, каждый из которых покрывает 120° . Определяющие точки составляют равносторонний треугольник, как показано на рис. 5-69а. Неравномерный узловый вектор и вектор однородных координат:

$$\begin{aligned}[X] &= [0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 3] \\ [H] &= [1 \ 1/2 \ 1 \ 1/2 \ 1 \ 1/2 \ 1] \end{aligned}$$

Аналогично, полная окружность также задается четырьмя рациональными квадратичными сегментами В-сплайна, каждый по 90° , где определяющие вершины составляют квадрат, рис. 5-69б. Неравномерный узловый вектор и вектор однородных координат:

$$\begin{aligned}[X] &= [0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 4 \ 4 \ 4], \\ [H] &= [1 \ \sqrt{2}/2 \ 1 \ \sqrt{2}/2 \ 1 \ \sqrt{2}/2 \ 1 \ \sqrt{2}/2 \ 1]. \end{aligned}$$

Рассмотрим пример построения конических сечений с помощью В-сплайна и объединения их в произвольные кривые.

Пример 5-22 Конические сечения, использующие рациональные В-сплайны

Построить единственный рациональный В-сплайн третьего порядка, соединяющий 90° дугу окружности, заданную квадратичным рациональным В-сплайном с определяющими вершинами $B_1[0 \ 0]$, $B_2[0 \ 2]$, $B_3[2 \ 2]$ с квадратическим рациональным В-сплайном третьего порядка, заданным точками $B_3[2 \ 2]$, $B_4[4 \ 2]$, $B_5[6 \ 3]$, $B_6[7 \ 5]$ с $h_i = 1$, $4 \leq i \leq 7$.

90° дуга окружности имеет узловой вектор $[0 \ 0 \ 0 \ 1 \ 1 \ 1]$ и вектор однородных координат $[1 \ \sqrt{2}/2 \ 1]$. Соответствующие векторы рационального В-сплайна, заданного точками $B_3B_4B_5B_6$: $[0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$ и $[1 \ 1 \ 1 \ 1]$.

Неравномерный узловой вектор комбинированной прямой: $[X] = [0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 3 \ 3 \ 3]$; вектор однородных координат: $[1 \ \sqrt{2}/2 \ 1 \ 1 \ 1 \ 1]$. Результат изображен на рис. 5-70.

5-14 ЛИТЕРАТУРА

- 5-1 Böhm, W., Farin, G., and Kahmann, J. «A Survey of Curve and Surface Methods in CAGD», *Comp. Aid. Geom. Des.*, Vol. 1, pp. 1–60, 1984.
- 5-2 Higdon, A., Olsien, E., Stiles, W., and Weese, J. *Mechanics of Materials*, 2d ed., John Wiley & Sons, New York, 1967. 5-3 South, N.E., and Kelly, J.P., «Analytic Surface Methods», Ford Motor Company N/C Development Unit, Product Engineering Office, December 1965.
- 5-4 Nutbourne, A.W., «A Cubic Spline Package Part 2 – The Mathematics», *Comp. Aid. Des.*, Vol. 5, No. 1, January 1973.
- 5-5 Adams, J. A., «A Comparison of Methods for Cubic Spline Curve Fitting», *Comp. Aid. Des.*, Vol. 6, pp. 1–9, 1974.
- 5-6 Schweikert, D. C., «An Interpolation Curve Using a Spline in Tension», *J. Math. Phys.*, Vol. 45, pp. 312–317, 1966.
- 5-7 Cline, A. K., «Curve Fitting Using Splines Under Tension», *Atmos. Tech.*, No. 3, pp. 60–65, 1973.
- 5-8 Nielson, G., «Some Piecewise Polynomial Alternatives to Splines Under Tension», *Computer Aided Geometric Design*, Barnhill, R. E., and Riesenfeld, R. F. (eds.), pp. 209–235, Academic Press, New York, 1974.
- 5-9 Overhauser, A. W., «Analytic Definition of Curves and Surfaces by Parabolic Blending», Tech. Rep. No. SL68-40, Ford Motor Company Scientific Laboratory, May 8, 1968.
- 5-10 Brewer, J. A., and Anderson, D. C., «Visual Interaction with Overhauser Curves and Surfaces», *Comp. Graph.*, Vol. 11, pp. 132–137, 1977 (SIGGRAPH 77).
- 5-11 Bézier, P. E., «How Renault Uses Numerical Control for Car Body Design and Tooling», SAE paper 680010, Society of Automotive Engineers' Congress, Detroit, MI, 1968.
- 5-12 Bézier, P. E., «Example of an Existing System in the Motor Industry: The Unisurf System», *Proc. Roy. Soc. (London)*, Vol. A321, pp. 207–218, 1971.
- 5-13 Bézier, P. E., *Emploi des Machines à Commande Numérique*, Masson et Cie, Paris, 1970. Translated by Forrest, A. R., and Pankhurst, A. F., as Bézier, P. E., *Numerical Control Mathematics and Applications*, John Wiley & Sons, London, 1972.

- 5-14 Forrest, A. R., «Interactive Interpolation and Approximation by Bézier Polynomials», *Comp. J.*, Vol. 15, pp. 71–79, 1972.
- 5-15 Gordon, W. J., and Riesenfeld, R. F., «Bernstein-Bézier Methods for the Computer Aided Design of Free-from Curves and Surfaces», *J. ACM*, Vol. 21, pp. 293–310, 1974.
- 5-16 Bartels, R. H., Beatty, J. C., and Barsky, B. A., *Splines for Use in Computer Graphics & Geometric Modeling*, Morgan Kaufman, Los Altos, CA, 1987.
- 5-17 Barsky, B. A., «Arbitrary Subdivision of Bézier Curves», TR UCB/CSD 85/265, Comp. Asi. Div., Univ. of Cal., Berkeley, November 1985.
- 5-18 Clark, J. H., «A Fast Scanline Algorithm for Rendering Parametric Surfaces», *Comp. Graph.*, Vol. 14, pp. 7–12, 1979 [addendum to SIGGRAPH 79 Conf. Proc. — «papers to be published in the CACM»].
- 5-19 Cohen, E., and Riesenfeld, R. F., «General Matrix Representations for Bézier and B-Spline Curves», *Comp. in Indus.*, Vol. 3, pp. 9–15, 1982.
- 5-20 Schoenberg, I. J., «Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions», *Q. Appl. Math.*, Vol. 4, pp. 45–99; pp. 112–141, 1946.
- 5-21 Cox, M. G., «The Numerical Evaluation of B-Csplines», National Physical Laboratory DNAC 4, August 1971.
- 5-22 de Boor, C., «On Calculation with B-splines», *J. Approx. Theory*, Vol. 6, pp. 50–62, 1972.
- 5-23 Riesenfeld, R. F., «Application of B-Spline Approximation to Geometric Problems of Computer Aided Design», PhD dissertation, Syracuse Univ., Syracuse, NY, 1972. Also available as U. of Utah, UTEC-CSc-73-126, March 1973.
- 5-24 Barsky, B. A., «End Conditions and Boundary Conditions for Uniform B-Spline Curve and Surface Representations», *Comp. in Indus.*, Vol. 3, pp. 17–29, 1982.
- 5-25 Hartley, P. J., and Judd, C. J., «Parameterization of Bézier Type B-spline Curves and Surfaces», *Comp. Aid. Des.*, Vol. 10, pp. 130–134, 1978.
- 5-26 Rogers, D. F., and Fog, N. G., «Constrained B-spline Curve and Surface Fitting», *CADJ*, Vol. 21, pp. 641–648, 1989.
- 5-27 Cohen, E., Lyche, T., and Schumacher, L. L., «Zlgorithms for Degree-Raising of Splines», *ACM Trans. on Graph.*, Vol. 4, pp. 171–181, 1985.
- 5-28 Cohen, E., Lyche, T., and Riesenfeld, R. F., «Discrete B-Splines and Subdivision Techniques in Computer Aided Geometric Design and Computer Graphics», *Comp. Graph. Imag. Proc.*, Vol. 14, pp. 87–111, 1980.
- 5-29 Prautzsch, H., «A Short Proof of the Oslo Algorithm», *Comp. Aid. Geom. Des.*, Vol. 1, pp. 95–96, 1984.
- 5-30 Böhm, W., «Inserting New Knots into B-spline Curves», *Comp. Aid. Des.*, Vol. 12, pp. 199–201, 1980.
- 5-31 Böhm, W., and Prautzsch, H., «The Insertion Algorithm», *Comp. Aid. Des.*, Vol. 17, pp. 58–59, 1985.
- 5-32 Coons, S. A., «Surfaces for Computer Aided Design of Space Forms», MIT Project MAC-TR-41, 1967.
- 5-33 Farouki, R., and Hinds, J., «A Hierarchy of Geometric Forms», *IEEE Comp. Graph. & Appl.*, Vol. 5, No. 5, pp. 51–78, 1985.
- 5-34 Piegl, L., «A Geometric Investigation of the Rational Bézier Scheme of Computer Aided Design», *Comp. in Indust.*, Vol. 7, pp. 401–410, 1986.
- 5-35 Forrest, A. R., «The Twisted Cubic Curve: A Computer Aided Geometric Design Approach», *Comp. Aid. Des.*, Vol. 12, pp. 165–172, 1980.
- 5-36 Böhm, W., «On Cubics: A Survey», *Comp. Graph. Imag. Proc.*, Vol. 19, pp. 201–226, 1982.

- 5-37 Forrest, A. R., «Curves and Surfaces for Computer-Aided Design», PhD Dissertation, Cambridge Univ., 1968.
- 5-38 Ball, A. A., «Consurf I-III», *Comp. Aid. Des.*, Vol. 6, pp. 243–249, 1974; Vol. 7, pp. 237–242, 1975; Vol. 9, pp. 9–12, 1977.
- 5-39 Versprille, K. J., «Computer-Aided Design Applications of the Rational B-spline Approximation Form», PhD dissertation, Syracuse Univ., Syracuse, NY, February 1975.
- 5-40 Tiller, W., «Rational B-splines for Curve and Surface Representation», *IEEE Comp. Graph. & Appl.*, Vol. 3, No. 6, pp. 61–69, September 1983.
- 5-41 Piegl, L., and Tiller, W., «Curve and Surface Constructions Using Rational B-splines», *Comp. Aid. Des.*, Vol. 19, pp. 485–498, 1987.
- 5-42 IGES 'Initial Graphics Exchange Specifications, Version 3.0', Doc. No. NB-SIR 86-3359 Nat. Bur. of Stds., Gaithersburg, MD, USA, 1986.
- 5-43 Lee, E., «Rational Bézier Representation for Conics», in *Geometric Modeling*, Farin, G. (ed.), SIAM, pp. 3–27, 1986.

Глава 6

ПОВЕРХНОСТИ

6-1 ВВЕДЕНИЕ

Поверхности и их описание играют важную роль в конструировании и производстве. Очевидными примерами этого являются разработка и производство автомобильных кузовов, корабельных корпусов, авиационных фюзеляжей и крыльев; пропеллеров, турбин, компрессоров и лопастей вентиляторов; посуды, мебели и обуви. В этом случае сущность конструирования либо по функциональным, либо по эстетическим причинам составляет форма или геометрия поверхности. Описание поверхности также играет важную роль в представлении данных, полученных в медицине, геологии, физике и других естественных науках.

Традиционным способом представления поверхности является использование нескольких ортогональных проекций. По существу, поверхность задается сеткой ортогональных плоских кривых, лежащих на секущих плоскостях, и несколькими ортогональными проекциями определенных «характерных» пространственных линий (см. рис. 5-1). Эти кривые первоначально могут быть созданы на бумаге либо же взяты (оцифрованы) из трехмерной модели, например в автомобильной промышленности дизайнерами традиционно используется глиняная модель.

В машинной графике и автоматизированном проектировании выгодно разрабатывать «настоящую» трехмерную математическую модель поверхности. Такая модель позволяет на ранних стадиях и относительно легко провести анализ характеристик поверхности, например кривизны, или физических количественных характеристик, зависящих от поверхности, например объема, площади поверхности, момента инерции и т. д. Упрощается визуализация поверхности (см. [6-1]), применяемая для разработки или контроля за ходом разработки. Далее, по сравнению с традиционным методом, использующим сетку линий, также существенно упрощается генерация необходимой для изготовления поверхности информации, например управляющих программ для станка с числовым программным управлением. В ранних работах Безье [6-2], Сабина [6-3], Петерса [6-4] и других авторов продемонстрирована осуществимость этого подхода. Разработанные в последнее время методы описания поверхностей достигли такой стадии развития, что позволяют «почти» исключить традиционное описание поверхности с помощью сетки линий.

Существуют две основные идеи, лежащие в основе методов описания поверхностей. В первой, связанной в основном с именем Кунса, математическую поверхность пытаются создать по заранее известным данным. Во второй, связанной в основном с именем Безье, математическую поверхность пытаются создать *ab initio* (с самого начала). В первое время отрасли, связанные с числовыми параметрами, например конструирование, тяготели к первому подходу, тогда как отрасли, учитывающие визуальные, осознательные или эстетические факторы, например дизайнеры и художники, тяготели ко второму. Последние работы Роджерса ([6-5] – [6-7]) об интерактивных системах разработки корпусов кораблей и Коэна ([6-8]) о разработке поверхностей показали, что эти два подхода совместимы.

В следующих разделах представлены основы математического описания параметрических поверхностей, используемых в машинной графике и автоматизированном проектировании.

6-2 ПОВЕРХНОСТИ ВРАЩЕНИЯ

Возможно самым простым способом создания трехмерной поверхности является вращение двумерного объекта, например прямой или плоской кривой вокруг оси в пространстве. Такие поверхности называются поверхностями вращения. Сначала для простоты предположим, что ось вращения совпадает с осью x и положительно направлена. Предположим также, что объекты вращения — отрезок, прямая или плоская кривая — лежат на плоскости xy . Позднее мы рассмотрим метод, позволяющий избавиться от этих ограничений.

Самый простой объект, который можно вращать вокруг оси, — это точка. При условии что точка не лежит на оси, вращение на угол 2π (360°) породит окружность. Поворот на меньший угол даст дугу окружности.

Следующим по сложности является отрезок, параллельный, но не совпадающий с осью вращения. Вращение на угол 2π (360°) породит в этом случае круговой цилиндр. Радиусом этого цилиндра является длина перпендикуляра, опущенного с отрезка на ось вращения. Длина цилиндра равна длине отрезка. Пример изображен на рис. 6-1.

Если отрезок и ось вращения компланарны и отрезок не параллелен оси вращения, то в результате вращения вокруг оси на угол 2π (360°) мы получим усеченный круговой конус. Радиусы оснований усеченного конуса — длины перпендикуляров, опущенных с концов отрезка на ось вращения. Высота конуса — это длина спроектированного на ось вращения отрезка. Пример изображен на рис. 6-2.

И снова, если отрезок и ось вращения компланарны и отрезок перпендикурен оси вращения, то в результате вращения на угол 2π (360°) мы получим плоский диск. Если отрезок пересекает (или касается) ось вращения, то получится сплошной диск, в противном случае диск будет иметь круглое отверстие. Примеры изображены на рис. 6-3.

И наконец, если отрезок наклонен к оси вращения, т. е. некомпланарен, то вращение на угол 2π (360°) породит однополостный гиперболоид (см. разд. 6-4 и 6-7).

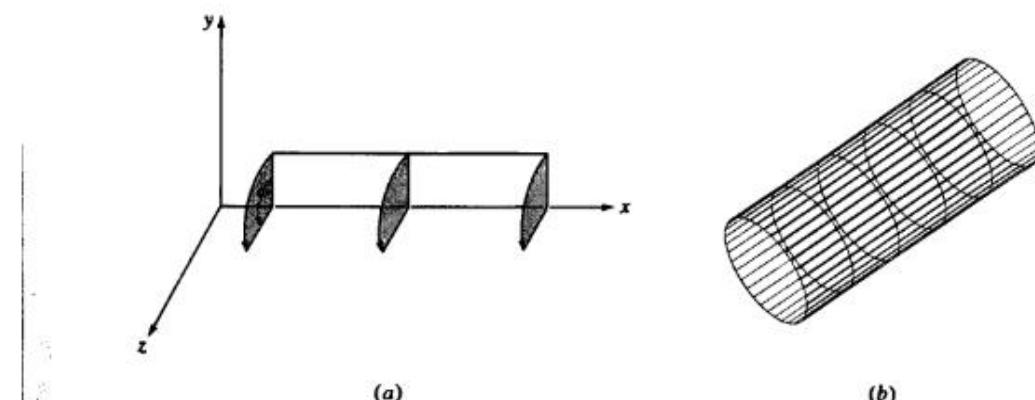


Рис. 6-1 Цилиндрическая поверхность вращения. (a) Схема построения; (b) результат.

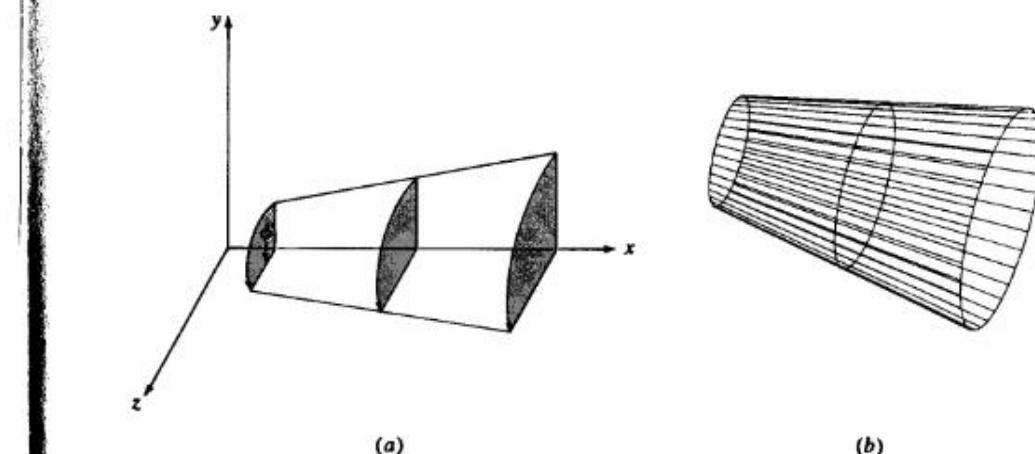


Рис. 6-2 Коническая поверхность вращения. (a) Схема построения; (b) результат.

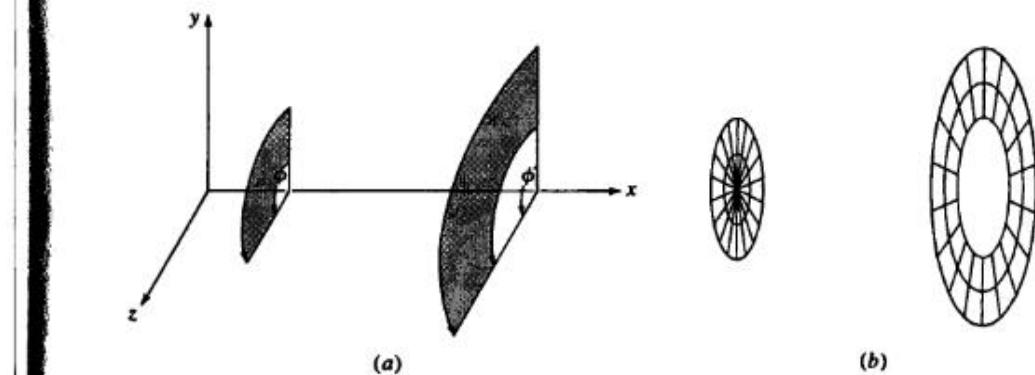


Рис. 6-3 Диск в качестве поверхности вращения. (a) Схема построения; (b) результат.

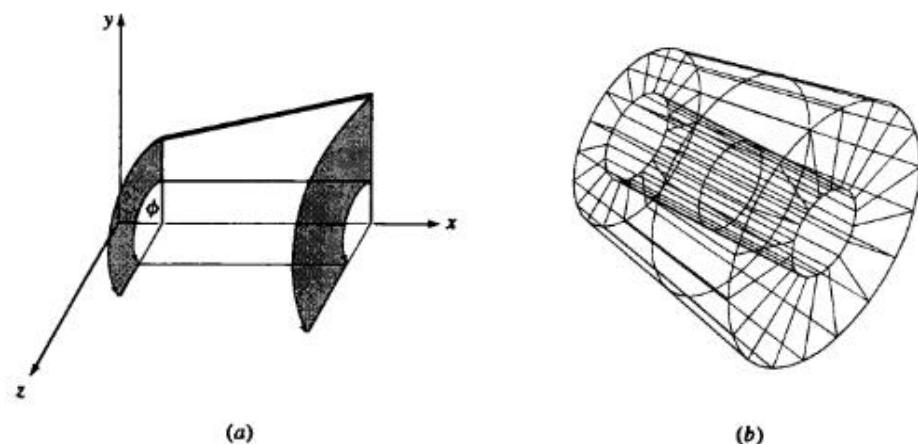


Рис. 6-4 Поверхность вращения из замкнутой ломаной. (a) Схема построения; (b) результат.

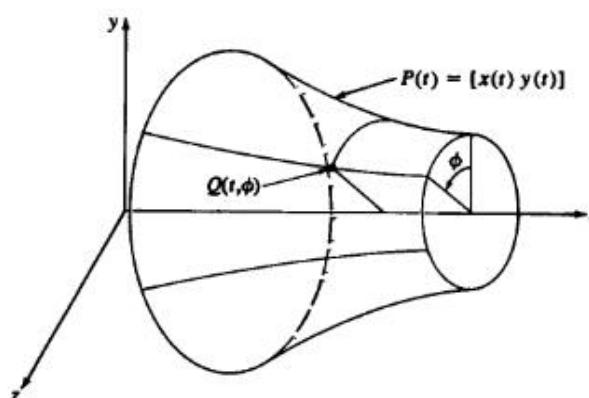


Рис. 6-5 Бипараметрическая поверхность вращения.

Для создания поверхностей вращения могут быть также использованы замкнутые и незамкнутые ломаные. На рис. 6-4 представлен конус с цилиндрическим отверстием.

Параметрическое уравнение точки на поверхности вращения можно получить, если вспомнить, что параметрическое уравнение вращаемого объекта, например

$$P(t) = [x(t) \quad y(t) \quad z(t)] \quad 0 \leq t \leq t_{\max},$$

есть функция одного параметра t . Вращение вокруг оси приводит к тому, что координаты зависят также от угла поворота. Таким образом, точка на поверхности вращения определяется двумя параметрами t и ϕ . Как показано на рис. 6-5, это бипараметрическая функция.

Для рассматриваемого частного случая, т. е. вращения вокруг оси x объекта, расположенного в плоскости xy , уравнение поверхности записывается¹

$$Q(t, \phi) = [x(t) \quad y(t) \cos \phi \quad y(t) \sin \phi] \quad (6-1)$$

Заметим, что здесь координата x не меняется. В качестве иллюстрации приведем пример.

Пример 6-1 Простая поверхность вращения

Рассмотрим отрезок с концами $P_1[1 \ 1 \ 0]$ и $P_2[6 \ 2 \ 0]$, лежащий в плоскости xy . Вращение отрезка вокруг оси x породит коническую поверхность. Определим на поверхности координаты точки с параметрами $t = 0.5$, $\phi = \pi/3$ (60°).

Параметрическое уравнение отрезка, соединяющего P_1 и P_2 , имеет вид

$$P(t) = [x(t) \quad y(t) \quad z(t)] = P_1 + (P_2 - P_1)t \quad 0 \leq t \leq 1$$

с декартовыми координатами

$$\begin{aligned} x(t) &= x_1 + (x_2 - x_1)t = 1 + 5t, \\ y(t) &= y_1 + (y_2 - y_1)t = 1 + t, \\ z(t) &= z_1 + (z_2 - z_1)t = 0. \end{aligned}$$

Используя уравнение (6-1), получим точку $Q(1/2, \pi/3)$ на поверхности вращения

$$\begin{aligned} Q(1/2, \pi/3) &= [1 + 5t \quad (1 + t) \cos \phi \quad (1 + t) \sin \phi] \\ &= \left[\frac{7}{2} \quad \frac{3}{2} \cos\left(\frac{\pi}{3}\right) \quad \frac{3}{2} \sin\left(\frac{\pi}{3}\right) \right] \\ &= \left[\frac{7}{2} \quad \frac{3}{4} \quad \frac{3\sqrt{3}}{4} \right] = [3.5 \quad 0.75 \quad 1.3]. \end{aligned}$$

Вращение плоских кривых также порождает поверхности вращения. Как показано на рис. 6-6а, сфера получается в результате вращения вокруг оси x расположенной в плоскости xy полуокружности, центрированной относительно начала координат. Вспомнив параметрическое уравнение окружности (см. разд. 4-5)

$$x = r \cos \theta \quad 0 \leq \theta \leq \pi \quad (4-4)$$

$$y = r \sin \theta$$

получим параметрическое уравнение сферы

$$\begin{aligned} Q(\theta, \phi) &= [x(\theta) \quad y(\theta) \cos \phi \quad y(\theta) \sin \phi] \\ &= [r \cos \theta \quad r \sin \theta \cos \phi \quad r \sin \theta \sin \phi] \quad 0 \leq \theta \leq \pi, \\ &\quad 0 \leq \phi \leq 2\pi \quad (6-2) \end{aligned}$$

¹ Заметим, что $Q(t, \phi)$ является вектор-функцией, т. е. в векторной форме

$$Q(t, \phi) = x(t)\mathbf{i} + y(t) \cos \phi \mathbf{j} + y(t) \sin \phi \mathbf{k},$$

где $\mathbf{i}, \mathbf{j}, \mathbf{k}$ – единичные векторы в x, y, z направлениях, соответственно.

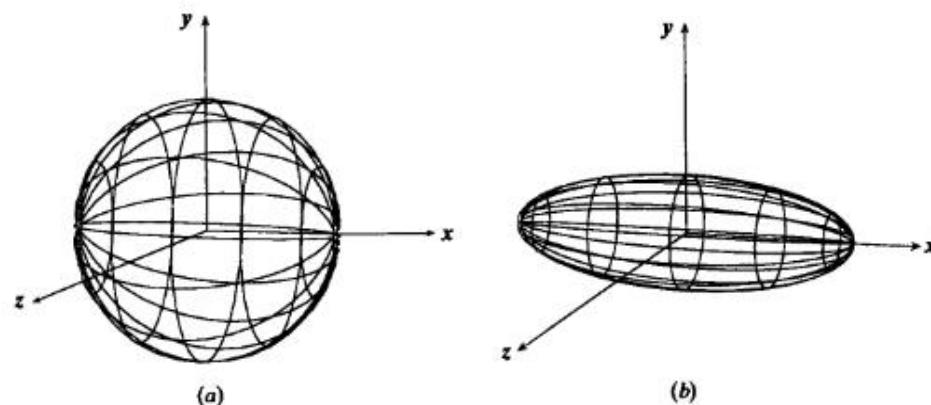


Рис. 6-6 Поверхности вращения. (a) Сфера; (b) эллипсоид.

Если вместо окружности подставить параметрическое уравнение центрированного полуэллипса, расположенного в плоскости xy , получится эллипсоид вращения. Напомнив параметрическое уравнение полуэллипса (см. разд. 4-6)

$$\begin{aligned} x &= a \cos \theta \quad 0 \leq \theta \leq \pi \\ y &= b \sin \theta \end{aligned} \quad (4-6)$$

получим для любой точки эллипсоида следующее параметрическое уравнение:

$$Q(\theta, \phi) = [a \cos \theta \quad b \sin \theta \cos \phi \quad b \sin \theta \sin \phi] \quad 0 \leq \theta \leq \pi, \\ 0 \leq \phi \leq 2\pi. \quad (6-3)$$

При $a = b = r$ уравнение (6-3) превращается в уравнение (6-2) для сферы. Эллипсоид вращения показан на рис. 6-6б.

Если ось вращения не проходит через центр окружности или эллипса, то в результате вращения получается тор с сечением в виде окружности или эллипса, соответственно. Параметрическое уравнение эллипса на плоскости xy с центром, не совпадающим с началом координат, выглядит так

$$\begin{aligned} x &= h + a \cos \theta \quad 0 \leq \theta \leq 2\pi, \\ y &= k + b \sin \theta, \end{aligned}$$

где (h, k) — это x, y — координаты центра эллипса, тогда параметрическое уравнение для любой точки тора имеет вид:

$$Q(\theta, \phi) = [h + a \cos \theta \quad (k + b \sin \theta) \cos \phi \quad (k + b \sin \theta) \sin \phi], \quad (6-4)$$

где $0 \leq \theta \leq 2\pi, 0 \leq \phi \leq 2\pi$. Если $a = b = r$, то уравнение (6-4) задает тор с сечением в виде окружности. Если $a \neq b$, то получится тор с сечением в виде эллипса. На рис. 6-7 представлены оба типа торов.

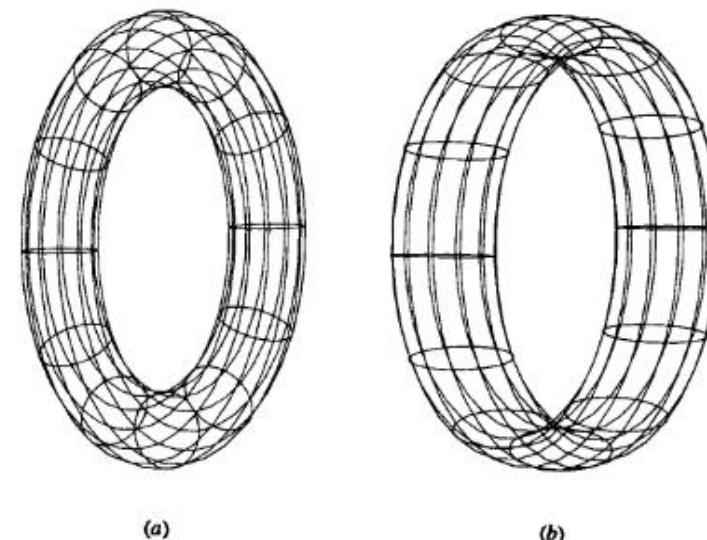


Рис. 6-7 Торы. (a) С сечением в виде окружности; (b) с сечением в виде эллипса.

Параболоид вращения получается при вращении параметрической параболы (см. разд. 4-7)

$$\begin{aligned} x &= a\theta^2 \quad 0 \leq \theta \leq \theta_{\max} \\ y &= 2a\theta \end{aligned} \quad (4-9)$$

вокруг оси x . Параметрическая поверхность задается уравнением

$$Q(\theta, \phi) = [a\theta^2 \quad 2a\theta \cos \phi \quad 2a\theta \sin \phi] \quad 0 \leq \theta \leq \theta_{\max}, \\ 0 \leq \phi \leq 2\pi. \quad (6-5)$$

Гиперболоид вращения получается при вращении параметрической гиперболы

$$\begin{aligned} x &= a \sec \theta \quad 0 \leq \theta \leq \theta_{\max} \\ y &= b \operatorname{tg} \theta \end{aligned} \quad (4-14)$$

вокруг оси x . Параметрическая поверхность задается уравнением

$$Q(\theta, \phi) = [a \sec \theta \quad b \operatorname{tg} \theta \cos \phi \quad b \operatorname{tg} \theta \sin \phi], \quad 0 \leq \theta \leq \theta_{\max}, \\ 0 \leq \phi \leq 2\pi. \quad (6-6)$$

Примеры показаны на рис. 6-8.

Для создания поверхности вращения можно использовать любую параметрическую кривую, например кубический сплайн, параболический сплайн, кривую Безье и В-сплайн. На рис. 6-9 изображена поверхность вращения, созданная из относительно простого параболического сплайна. На рис. 6-10 изображен бокал,

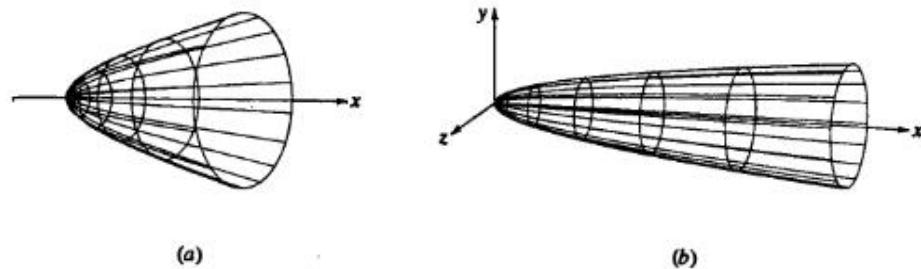


Рис. 6-8 Поверхности вращения. (a) Параболоид; (b) гиперболоид.

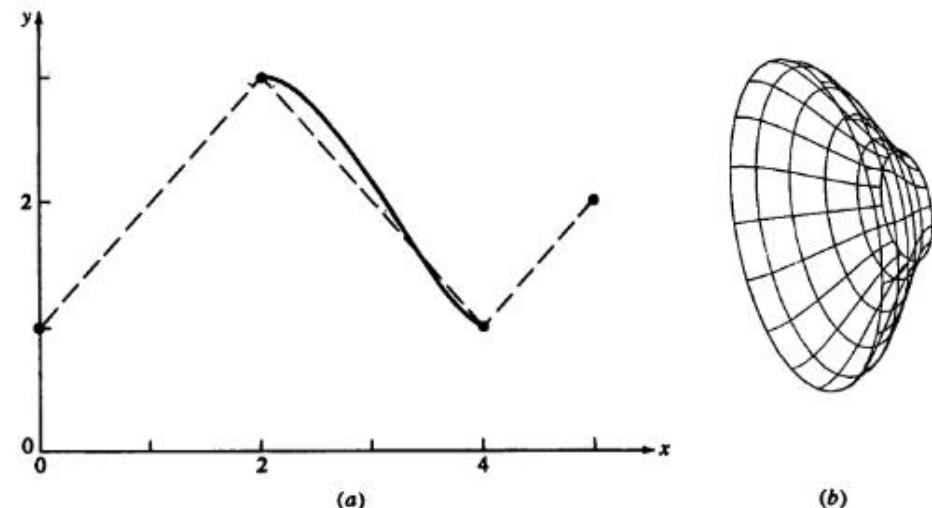


Рис. 6-9 Поверхность вращения из параболически интерполированной кривой. (a) Создание кривой; (b) поверхность.

созданный как поверхность вращения с помощью незамкнутого В-сплайна. Заметим, что бокал имеет как внутреннюю, так и внешнюю стороны. Вращение производится относительно оси y ¹.

¹ Метод, использованный при создании В-сплайна для бокала на рис. 6-10, может представлять определенный интерес. Сначала набросок бокала был сделан на миллиметровке. Затем тридцать четыре точки с этого рисунка были оцифрованы. С помощью описанного в разд. 5-11 итерационного процесса был получен исходный В-сплайн ($k = 4$, $n + 1 = 21$). После этого для окончательной разработки в интерактивную систему (см. [6-7]) были переданы определяющие точки кривой. Для изображенной поверхности вращения был использован В-сплайн, полученный из окончательных 25 вершин. Определяющие точки кривой: [0, 0.684], [0.302 0.684], [0.302 0.684], [1.598 -0.288], [1.088 0.405], [0.374 0.773], [0.848 0.993], [1.232 1.446], [1.451 1.875], [1.502 2.631], [1.448 3.308], [1.226 4.076], [1.397 4.449], [1.398 4.592], [1.268 4.572], [1.241 4.485], [1.109 4.170], [1.214 3.669], [1.313 2.889], [1.310 2.280], [1.172 1.575], [0.629 1.119], [0.281 0.882], [0.278 0.891], [0.897].

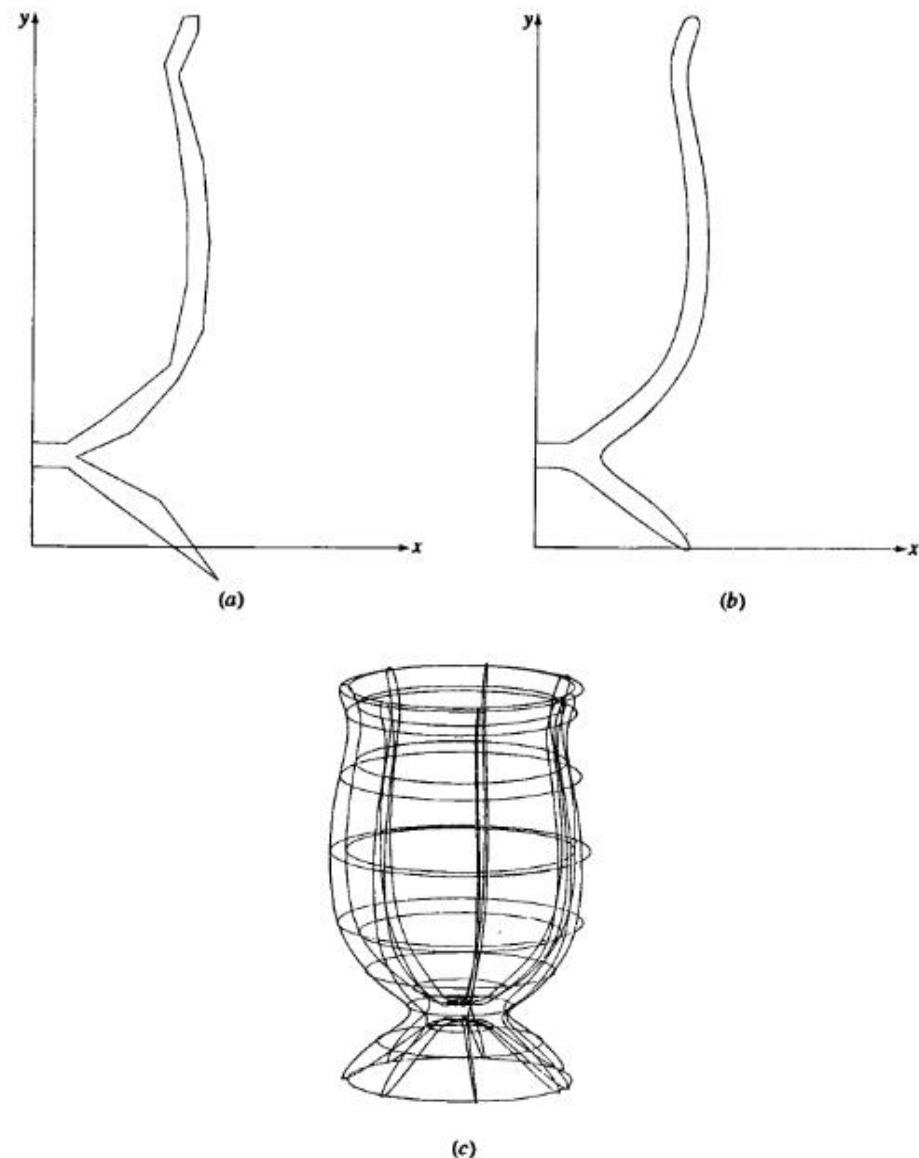


Рис. 6-10 В-сплайн поверхность вращения. (a) Вершины ломаной; (b) В-сплайн; (c) поверхность.

Напомним, что в матричной форме параметрическая пространственная кривая (см. уравнения (5-27), (5-44), (5-67) и (5-94)) задается следующим образом:

$$P(t) = [T][N][G],$$

где $[T]$, $[N]$ и $[G]$ – соответственно матрица параметров, матрица функций смещения и геометрическая матрица. Таким образом, в общей форме матричное уравнение поверхности вращения записывается в виде:

$$Q(t, \phi) = [T][N][G][S], \quad (6-7)$$

где $[S]$ представляет вклад вращения вокруг оси на угол ϕ . Для частного случая вращения вокруг оси x имеем:

$$[S] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6-8)$$

Эти методы иллюстрируются в следующем примере.

Пример 6-2 Поверхность вращения, созданная по параболической кривой

Рассмотрим параболическую кривую, заданную точками $P_1[0 \ 1 \ 0]$, $P_2[2 \ 3 \ 0]$, $P_3[4 \ 1 \ 0]$, $P_4[5 \ 2 \ 0]$. Будем вращать эту кривую вокруг оси x на угол 2π , чтобы получить поверхность вращения. Найдем на поверхности точку с параметрами $t = 0.5$, $\phi = \pi/3$ (60°).

Из уравнений (6-7) и (6-8) получим параметрическое уравнение поверхности вращения

$$Q(t, \phi) = [T][A][G][S],$$

где $[S]$, $[T]$, $[A]$ и G задаются уравнениями (5-44), (5-52) и (5-53) соответственно.

Конкретнее,

$$Q(t, \phi) = \left(\frac{1}{2}\right) [t^3 \ t^2 \ t \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 1 \\ 2 & 3 & 0 & 1 \\ 4 & 1 & 0 & 1 \\ 5 & 2 & 0 & 1 \end{bmatrix} \times$$

$$\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q(t, \phi) = \left(\frac{1}{2}\right) [t^3 \ t^2 \ t \ 1] \begin{bmatrix} -1 & 7 & 0 & 1 \\ 1 & -11 & 0 & 1 \\ 4 & 0 & 0 & 1 \\ 4 & 6 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Для $t = 0.5$ и $\phi = \pi/3$ (60°)

$$Q(0.5, \pi/3) = \frac{1}{2} \begin{bmatrix} \frac{1}{8} & \frac{1}{4} & \frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} -1 & 7 & 0 & 1 \\ 1 & -11 & 0 & 1 \\ 4 & 0 & 0 & 1 \\ 4 & 6 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{49}{16} & \frac{33}{32} & \frac{33\sqrt{3}}{32} & 1 \end{bmatrix} = [3.0625 \ 1.03125 \ 1.786181 \ 1].$$

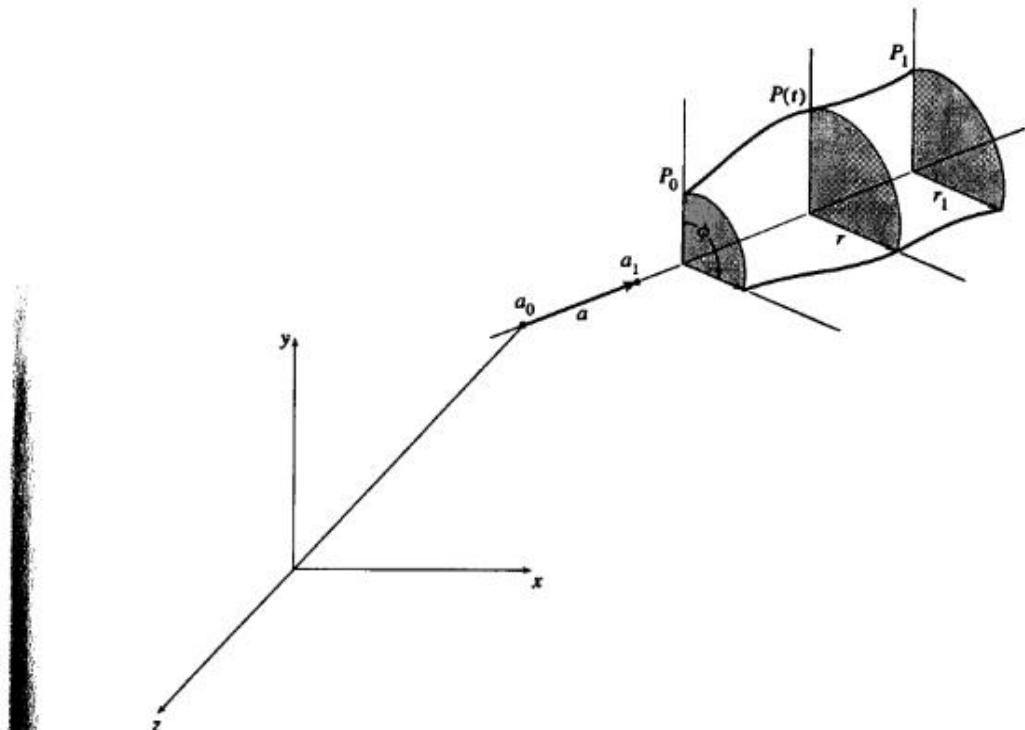


Рис. 6-11 Поверхность вращения вокруг произвольной оси.

Результаты изображены на рис. 6-9. Такая поверхность может быть результатом разработки кубка или даже газового канала двигателя или ракетного сопла.

Предыдущие результаты были получены путем вращения точки, отрезка, ломаной или кривой вокруг координатной оси, а именно вокруг оси x . К более общему случаю поворота вокруг произвольной оси в пространстве поверхность вращения, полученную в более удобной локальной системе координат, можно свести с помощью переносов и поворотов, приводящих поверхность в нужное положение.

На рис. 6-11 показана параметрическая кривая $P(t)$, повернутая вокруг произвольной оси в пространстве, проходящей через точки a_0 и a_1 и направленной от a_0 к a_1 . После того как поверхность создана в удобной системе координат для приведения поверхности вращения в нужное положение, нужно совершить следующие действия:

1. Перенести точку a_0 в начало координат.
2. Выполнить повороты, необходимые для совмещения осей a и $+z$ (см. разд. 5-9).
3. Повернуть вокруг оси y на угол $+90^\circ$ для совмещения осей a и x .¹

¹ Этот шаг необходим только для согласования с предыдущими рассуждениями. Поверхность вращения может быть легко построена для поворотов вокруг оси x .

Эти три шага необходимы только для того, чтобы найти обратное преобразование, размещающее поверхность вращения в нужном месте в трехмерном пространстве. Получив поверхность вращения вокруг оси x , приведем ее в нужное положение в пространстве:

1. Сдвинуть по оси x , чтобы переместить центр поверхности вращения в нужное положение на оси a .
2. Применить к поверхности вращения преобразование, обратное к суммарному преобразованию поворотов.
3. Применить к поверхности вращения обратный перенос точки a_0 .

Точка на поверхности вращения тогда задается уравнением:

$$Q(t, \phi) = [\bar{Q}][Tr][\bar{R}_y]^{-1}[R_y]^{-1}[R_x]^{-1}[Tr]^{-1}, \quad (6-9)$$

где $[Tr]$, $[R_x]$, $[R_y]$ задаются уравнениями (3-22) – (3-24). $[\bar{R}_y]^{-1}$ задается уравнением (3-8), и матрица $[\bar{Q}]$ задается в форме уравнения (6-7) с геометрической матрицей $[G]$, представленной в однородных координатах. $[S]$ теперь является матрицей 4×4 , заданной в виде

$$[S] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6-10)$$

Данный метод иллюстрируется на следующем примере.

Пример 6-3 Поверхность вращения вокруг произвольной оси

Найдем координаты точки с параметрами $t = \pi/2$, $\phi = \pi/6$ на поверхности вращения, образованной вращением эллипса с главной осью, наклоненной относительно оси вращения. Ось вращения проходит через центр эллипса и лежит в плоскости эллипса. Угол наклона $i = \pi/4$. Полусоси эллипса $a = 5$, $b = 1$. Ось проходит через точки $a_0[0 \ 10 \ 10]$ и $a_1[10 \ 10 \ 0]$. Центр эллипса находится в точке a_1 .

Для начала найдем направляющие косинусы оси вращения (см. уравнение 3-26)

$$[c_x \ c_y \ c_z] = [1/\sqrt{2} \ 0 \ -1/\sqrt{2}]$$

и (см. уравнение 3-18)

$$d = \sqrt{c_y^2 + c_z^2} = 1/\sqrt{2}.$$

Таким образом, используя уравнения (3-22) – (3-24)

$$[M_1] = [Tr][R_x][R_y] =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -10 & -10 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 & 1 & 0 & 0 \\ -1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 & -1 & 0 & 0 \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ -10/\sqrt{2} & 10 & 10/\sqrt{2} & 1 \end{bmatrix}$$

совместим ось вращения с осью z . Поворот вокруг оси y на угол 90° приводит к

$$[M_2] = [Tr][R_x][R_y][\bar{R}_y] =$$

$$= \begin{bmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 \sqrt{2} & -1 & 0 \sqrt{2} & 0 \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ -10/\sqrt{2} & 10 & 10/\sqrt{2} & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ 0 \sqrt{2} & -1 & 0 \sqrt{2} & 0 \\ -1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ 10/\sqrt{2} & 10 & 10/\sqrt{2} & 1 \end{bmatrix}.$$

Используя преобразование $[M_2]$ и однородные координаты, преобразуем центр эллипса, находящийся первоначально в a :

$$[10 \ 10 \ 10 \ 1] \begin{bmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} & 0 \\ 0 \sqrt{2} & -1 & 0 \sqrt{2} & 0 \\ 1/\sqrt{2} & 0 & -1/\sqrt{2} & 0 \\ -10/\sqrt{2} & 10 & 10/\sqrt{2} & 1 \end{bmatrix} = [20/\sqrt{2} \ 0 \ 0 \ 1]$$

т.е. в $h = 20/\sqrt{2}$ на оси x .

Вспомнив пример 4-4 из разд. 4-6, получим параметрические уравнения центрированного относительно начала координат эллипса, повернутого на угол i :

$$\begin{aligned} x &= a \cos t \cos i - b \sin t \sin i \quad 0 \leq t \leq 2\pi \\ y &= a \cos t \sin i + b \sin t \cos i \end{aligned}$$

которые могут быть записаны в форме уравнения (6-7)

$$[\cos t \ \sin t \ 0 \ 1] \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\cos i & \sin i & 0 & 0 \\ -\sin i & \cos i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Тогда поверхность вращения имеет вид:

$$\begin{aligned} [Q] &= [T][N][G][S] = \\ &= [\cos t \quad \sin t \quad 0 \quad 1] \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\cos i & \sin i & 0 & 0 \\ -\sin i & \cos i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \\ &\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Для $a = 5$, $b = 1$, $i = \pi/4$

$$[\bar{Q}] = [\cos t \quad \sin t \quad 0 \quad 1] \begin{bmatrix} 5\sqrt{2}/2 & 5\sqrt{2}/2 & 0 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Эта центрированная относительно начала координат поверхность вращения также показана на рис. 6-10.

Перенеся начало в (h, n) на оси x и заметив, что

$$\begin{aligned} [M_2]^{-1} &= [\bar{R}_y]^{-1}[R_y]^{-1}[R_x]^{-1}[Tr]^{-1} = \\ &= \begin{bmatrix} \sqrt{2}/2 & 0 & -\sqrt{2}/2 & 0 \\ 0 & -1 & 0 & 0 \\ -\sqrt{2}/2 & 0 & -\sqrt{2}/2 & 0 \\ 0 & 10 & 10 & 1 \end{bmatrix} \end{aligned}$$

получим

$$[Q] = [T][N][G][S][Tr_x][M_2]^{-1}$$

Точка на поверхности $Q(t, \phi)$ имеет координаты

$$\begin{aligned} Q(t, \phi) &= [\cos t \quad \sin t \quad 0 \quad 1] \begin{bmatrix} 5\sqrt{2}/2 & 5\sqrt{2}/2 & 0 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \\ &\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 20/\sqrt{2} & 0 & 0 & 1 \end{bmatrix} \times \\ &\times \begin{bmatrix} \sqrt{2}/2 & 0 & -\sqrt{2}/2 & 0 \\ 0 & -1 & 0 & 0 \\ -\sqrt{2}/2 & 0 & -\sqrt{2}/2 & 0 \\ 0 & 10 & 10 & 1 \end{bmatrix} \end{aligned}$$

или

$$\begin{aligned} Q(t, \phi) &= \left(\frac{1}{2}\right) [\cos t \quad \sin t \quad 0 \quad 1] \times \\ &\times \begin{bmatrix} 5(1 - \sin \phi) & -5\sqrt{2} \cos \phi & -5(1 + \sin \phi) & 0 \\ -(1 + \sin \phi) & -\sqrt{2} \cos \phi & (1 - \sin \phi) & 0 \\ 0 & 0 & 0 & 0 \\ 20 & 20 & 0 & 0 \end{bmatrix} \end{aligned}$$

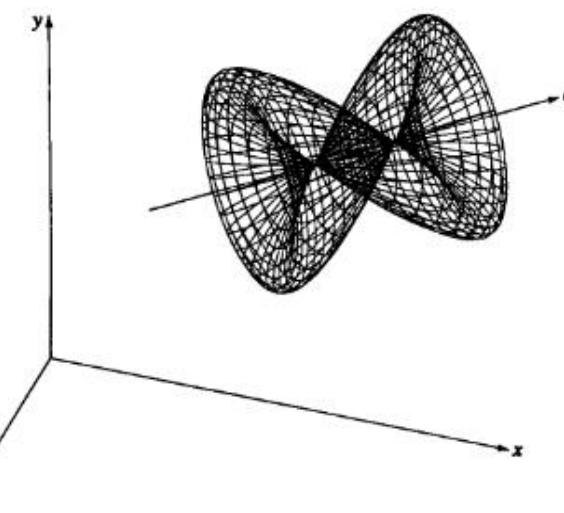


Рис. 6-12 Сложная эллиптическая поверхность вращения из примера 6-3.

Для $t = \pi/2$, $\phi = \pi/6$

$$\begin{aligned} Q(\pi/2, \pi/6) &= \left(\frac{1}{2}\right) [0 \quad 1 \quad 0 \quad 1] \begin{bmatrix} 5/2 & -5\sqrt{6}/2 & -15/2 & 0 \\ -3/2 & -\sqrt{6}/2 & 1/2 & 0 \\ 0 & 0 & 0 & 0 \\ 20 & 20 & 0 & 1 \end{bmatrix} = \\ &= [37/4 \quad 10 - \sqrt{6}/4 \quad 1/4 \quad 1] = \\ &= [9.25 \quad 9.388 \quad 0.25 \quad 1] \end{aligned}$$

Получившаяся в результате поверхность изображена на рис. 6-12. Заметим, что эта поверхность самопрересекается и имеет сложную форму.

Формальное дифференцирование уравнения (6-7) дает параметрические производные для поверхности вращения. А именно, производная в осевом направлении равна

$$Q_t(t, \phi) = [T'][N][G][S], \quad (6-11)$$

а производная в радиальном направлении

$$Q_\phi(t, \phi) = [T][N][G][S'], \quad (6-12)$$

где штрих обозначает соответствующее дифференцирование.

Нормаль к поверхности задается векторным произведением параметрических производных, т. е.

$$n = Q_t \times Q_\phi. \quad (6-13)$$

6-3 ЗАМЕТАЮЩИЕ ПОВЕРХНОСТИ

Пространственную поверхность также можно получить путем перемещения объекта, например отрезка, ломаной или кривой вдоль некоторой кривой, в пространстве. Получающиеся в результате поверхности называются заметающими. Генерация заметающих поверхностей часто используется в геометрическом моделировании. Самый простой заметающий объект — точка. Результатом заметания точки является, конечно, не поверхность, а пространственная кривая. Тем не менее на этом примере хорошо иллюстрируется основная идея.

Рассмотрим координатный вектор $P[x \ y \ z \ 1]$, перемещаемый вдоль кривой, заданной заметающим преобразованием $[T(s)]$. Координатный вектор $Q(s)$, представляющий получающуюся в результате кривую, задается выражением

$$Q(s) = P[T(s)] \quad s_1 \leq s \leq s_2 \quad (6-14)$$

Преобразование $[T(s)]$ определяет форму кривой. Например, если направляющая есть отрезок длины n , параллельный оси z , то (см. уравнение 3-14)

$$[T(s)] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & ns & 1 \end{bmatrix} \quad 0 \leq s \leq 1.$$

Если направляющая есть окружность с центром в начале координат, лежащая в плоскости $z = \text{const}$, то (см. уравнение 3-8)

$$[T(s)] = \begin{bmatrix} \left(\frac{r}{x}\right) \cos\{2\pi(s + s_i)\} & 0 & 0 & 0 \\ 0 & \left(\frac{r}{y}\right) \sin\{2\pi(s + s_i)\} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 0 \leq s \leq 1,$$

где $s_i = (1/2\pi) \arctg(y_i/x_i)$ и для $P[x \ y \ z \ 1]$, $r = \sqrt{x^2 + y^2}$. Здесь подстрочный индекс i используется для обозначения начальной, или стартовой точки.

Сложные направляющие можно сконструировать с помощью объединения более простых. Например, после объединения двух предыдущих преобразований получим один оборот спирали вдоль оси z , т. е.

$$[T(s)] = \begin{bmatrix} \left(\frac{r}{x}\right) \cos\{2\pi(s + s_i)\} & 0 & 0 & 0 \\ 0 & \left(\frac{r}{y}\right) \sin\{2\pi(s + s_i)\} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & ns & 1 \end{bmatrix} \quad 0 \leq s \leq 1.$$

Самая простая заметающая поверхность получается в результате перемещения отрезка вдоль направляющей. Напомним, что параметрическое уравнение отрезка

$$P(t) = P_1 + (P_2 - P_1)t \quad 0 \leq t \leq 1. \quad (6-15)$$

Соответствующая заметающая поверхность задается уравнением

$$Q(t, s) = P(t)[T(s)], \quad 0 \leq t \leq 1, \quad s_1 \leq s \leq s_2, \quad (6-16)$$

где $[T(s)]$ снова обозначает заметающее преобразование. Если оно состоит только из переносов и/или локальных или общих масштабирований, то в результате возникает плоская поверхность. Если заметающее преобразование включает повороты, то поверхность не будет плоской. На рис. 6-13 изображена спиральная заметающая поверхность, полученная из отрезка, в начальный момент параллельного оси y , с помощью одновременного переноса вдоль оси x и вращения вокруг нее. Более подробно этот метод объясняется в следующем примере.

Пример 6-4 Линейчатая заметающая поверхность

Рассмотрим отрезок, лежащий в плоскости xy , параллельный оси y и заданный концевыми точками $P_1[0 \ 0 \ 0]$ и $P_2[0 \ 3 \ 0]$. Найти точку с параметрами $t = 0.5$, $s = 0.5$ на заметающей поверхности, образованной одновременным переносом отрезка на 10 единиц вдоль оси x и его поворотом на 2π вокруг оси x .

В данном случае матрица заметающего преобразования — переноса и последующего поворота, задается матрицей¹

$$[T(s)] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(2\pi s) & \sin(2\pi s) & 0 \\ 0 & -\sin(2\pi s) & \cos(2\pi s) & 0 \\ ts & 0 & 0 & 1 \end{bmatrix}$$

Параметрическое уравнение отрезка

$$\begin{aligned} P(t) &= P_1 + (P_2 - P_1)t = [0 \ 0 \ 0 \ 1] + [0 - 0 \ 3 - 0 \ 0 - 0 \ 1 - 1]t \\ &= [0 \ 3t \ 0 \ 1] \end{aligned}$$

Из уравнения (6-15) заметающая поверхность определяется следующим образом:

$$Q(t, s) = [P(t)][T(s)]$$

$$= [0 \ 3t \ 0 \ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(2\pi s) & \sin(2\pi s) & 0 \\ 0 & -\sin(2\pi s) & \cos(2\pi s) & 0 \\ ts & 0 & 0 & 1 \end{bmatrix}$$

и

$$\begin{aligned} Q(0.5, 0.5) &= [0 \ 1.5 \ 0 \ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 5 & 0 & 0 & 1 \end{bmatrix} \\ &= [5 \ -1.5 \ 0 \ 1]. \end{aligned}$$

Результат изображен на рис. 6-13.

Для создания заметающих поверхностей также могут быть использованы параметрические кривые, например кубические сплайны, параболические сплайны,

¹ Та же самая матрица получается для поворота и последующего переноса.

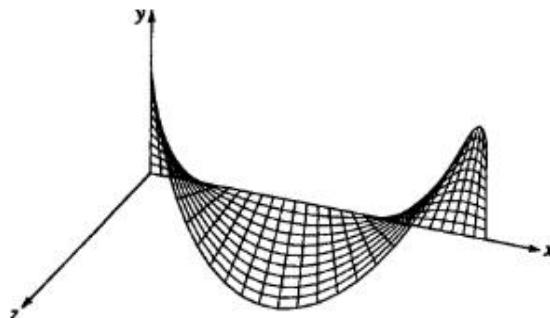


Рис. 6-13 Спиральная заматающая поверхность.

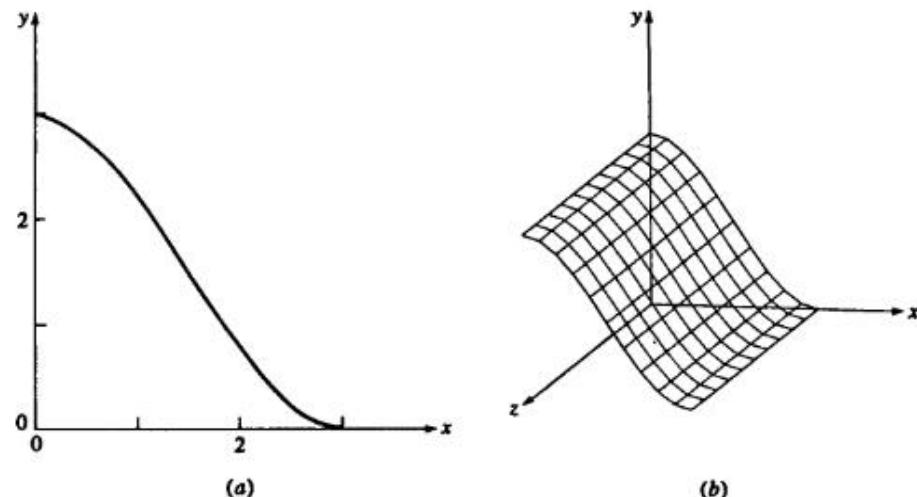


Рис. 6-14 Заматающая поверхность на основе кубического сплайна. (a) Кривая; (b) поверхность.

кривые Безье и В-сплайны. В этом случае уравнение поверхности идентично уравнению (6-15), где $P(t)$ теперь представляет параметрическую кривую. На рис. 6-14 показана заматающая поверхность, созданная из одного сегмента кубического сплайна, перенесенного вдоль оси z . Этот метод иллюстрируется примером.

Пример 6-5 Заматающая поверхность с кубическим сплайном в качестве образующей

Рассмотрим поверхность, образованную в результате переноса на 10 единиц вдоль оси z кубического сплайна, определяемого следующими данными: $P_1[0 \ 3 \ 0 \ 1]$, $P_2[3 \ 0 \ 0 \ 1]$, $P'_1[3 \ 0 \ 0 \ 0]$, $P'_2[3 \ 0 \ 0 \ 0]$.¹

¹ В однородных координатах касательный вектор имеет нулевой однородный координатный множитель.

Заматающая поверхность задается формулами:

$$Q(t, s) = [C(t)][T(s)] \quad 0 \leq t \leq 1, \quad 0 \leq s \leq 1.$$

Нормализованный сегмент кубического сплайна задается (см. уравнение 5-27)

$$[C(t)] = [t^3 \ t^2 \ t \ 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 3 & 0 & 1 \\ 3 & 0 & 0 & 1 \\ 3 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \end{bmatrix}$$

Сегмент кривой изображен на рис. 6-14а.

Заматающее преобразование (см. уравнение 3-14) имеет вид:

$$[T(s)] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & ns & 1 \end{bmatrix}$$

Следовательно,

$$Q(t, s) = [t^3 \ t^2 \ t \ 1] \begin{bmatrix} 0 & 6 & 0 & 0 \\ 6 & -9 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 10s & 1 \end{bmatrix}$$

Для $t = 0.5, s = 0.5$

$$\begin{aligned} Q(0.5, 0.5) &= [0.125 \ 0.25 \ 0.5 \ 1] \begin{bmatrix} 0 & 6 & 0 & 0 \\ 6 & -9 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 5 & 1 \end{bmatrix} \\ &= [3 \ 1.5 \ 5 \ 1]. \end{aligned}$$

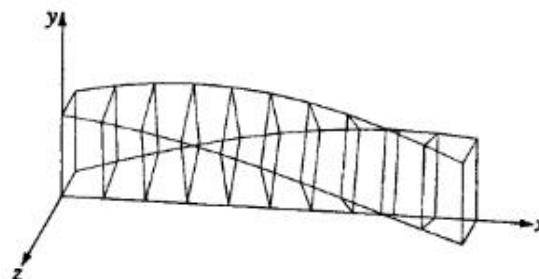
Результат изображен на рис. 6-14б.

Следует позаботиться о том, чтобы избежать вырождения поверхностей или их частей при генерации заматающих поверхностей из отрезков и кривых. Пример этого приведен на рис. 6-15. Здесь z -образную кривую, расположенную в плоскости xy , перемещают параллельно оси x . Заметим, что «хвосты» с левой и правой сторон явленияются вырожденными частями поверхности (т. е. отрезками) с нулевой площадью. Такие вырожденные части поверхности могут породить проблемы в геометрических моделирующих системах.

Кроме незамкнутых кривых для создания заматающих поверхностей используются замкнутые ломаные и кривые. При добавлении концевых поверхностей заматающая поверхность ограничивает конечный объем в пространстве. Подобным способом объемные примитивы создаются во многих геометрических моделирующих системах. Перемещаемый вдоль прямой направляющий квадрат или прямоугольник порождает прямоугольный параллелепипед. Окружность, перемещаемая вдоль прямой направляющей, порождает цилиндр. Окружность с уменьшающимся радиусом, перемещаемая вдоль прямой направляющей, порождает конус. Также возможно вращение вокруг направляющей оси. На рис. 6-16 показана



Рис. 6-15 Заметающая поверхность с вырожденными участками.

Рис. 6-16 Заметающая поверхность, образованная квадратом, перемещаемым вдоль оси x и одновременно вращаемым вокруг нее.

заметающая поверхность, образованная из плоского квадрата, перпендикулярного оси x и перемещаемого вдоль оси x с одновременным вращением на 90° вокруг этой же оси.

При перемещении плоского многоугольника или замкнутой кривой вдоль произвольной направляющей кривой следует отметить два важных момента. Во-первых, какая точка многоугольника постоянно лежит на направляющей? В общем случае это может быть любая точка многоугольника или замкнутой кривой. Для различных точек порожденные поверхности различаются.

Во-вторых, каково направление нормали многоугольника или замкнутой кривой во время перемещения вдоль направляющей? В этом случае обычно применяется два подхода. Нормаль может быть направлена как касательная к направляющей кривой. Либо может задаваться независимо от направляющей. Последняя альтернатива очень гибка. Два примера изображены на рис. 6-17, где показаны заметающие поверхности, образованные квадратом, центрированным на оси x и передвигаемым вдоль направляющей $x = ls$, $y = \cos(\pi s) - 1$. На рис. 6-17a нормаль направлена как ось x . На рис. 6-17b нормаль в каждой точке направлена так же, как и касательная к направляющей. Отметим разницу между этими двумя заметающими поверхностями. Детальный пример дополнительно иллюстрирует эту идею.

Пример 6-6 Сложная заметающая поверхность

Поверхность создается перемещением плоского квадрата, заданного вершинами $P_1[0 \quad -1 \quad 1]$, $P_2[0 \quad -1 \quad -1]$, $P_3[0 \quad 1 \quad -1]$, $P_4[0 \quad 1 \quad 1]$, вдоль направляющей кривой $x = 10s$, $y = \cos(\pi s) - 1$. При этом направление нормали к многоугольнику совпадает с касательной к образующей.

Направление касательной к образующей равно $[10 \quad -\pi \sin(\pi s) \quad 0]$. Таким образом, угол поворота вокруг оси z , необходимого для сополавленности нормали многоугольнику и касательной, равен

$$\psi = \arctg \left(\frac{-\pi \sin(\pi s)}{10} \right).$$

Заметающее преобразование равно, таким образом,

$$[T(s)] = \begin{bmatrix} \cos \psi & \sin \psi & 0 & 0 \\ -\sin \psi & \cos \psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 10s & \cos(\pi s) - 1 & 0 & 1 \end{bmatrix}.$$

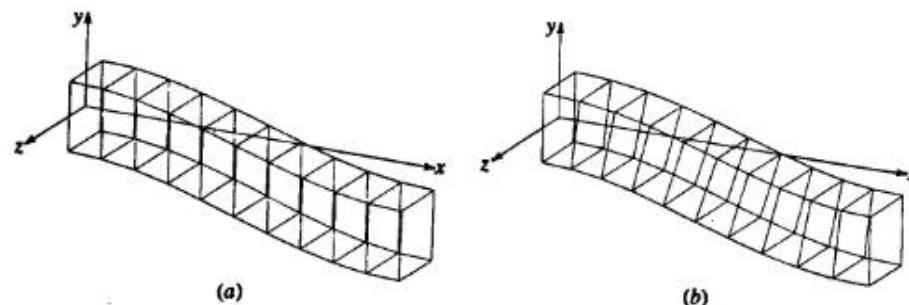
При $s = 0.5$ угол поворота равен

$$\psi = \arctg \left(\frac{-\pi \sin(\pi/2)}{10} \right) = \arctg(-\pi/10) = -17.44^\circ$$

Тогда при $s = 0.5$ квадрат задается матрицей

$$Q = \begin{bmatrix} 0 & -1 & 1 & 1 \\ 0 & -1 & -1 & 1 \\ 0 & 1 & -1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.954 & -0.3 & 0 & 0 \\ 0.3 & 0.954 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 5 & -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 4.7 & -0.954 & 1 & 1 \\ 4.7 & -0.954 & -1 & 1 \\ 5.3 & -0.046 & -1 & 1 \\ 5.3 & -0.046 & 1 & 1 \end{bmatrix}$$

Результат изображен на рис. 6-17b

Рис. 6-17 Прямоугольник перемещается вдоль направляющей кривой. (a) Нормаль направлена как ось x ; (b) нормаль направлена так же, как касательная к направляющей кривой.

Дополнительную информацию о заметающих поверхностях можно найти в [6-9] и [6-10], а также в приведенной в них библиографии.

6-4 КВАДРАТИЧНЫЕ ПОВЕРХНОСТИ

Обычные квадратичные поверхности, такие как сфера, конус, цилиндр, эллипсоид, параболоид вращения и гиперболоид вращения являются также и поверхностями вращения (см. разд. 6-2). Эти квадратичные поверхности, особенно сфера, конус и цилиндр, играют важную роль при изготовлении деталей механизмов, а также в описании обрабатываемых поверхностей. Например, баскетбольные и футбольные мячи имеют сферическую форму, воронки — коническую, банки для пива — цилиндрическую, а спутниковые антенны — параболическую. Квадратичные поверхности также важны и при описании более сложных поверхностей. Например, фюзеляж истребителя North American P-51 времен второй мировой войны был описан с помощью сегментов конических поверхностей [6-11]. Этот метод называется методом построения плавов с помощью конических сечений.

Существует два метода представления квадратичных поверхностей — алгебраический (включающий параметрические представления) и геометрический. Геометрические описания обычно применяются для естественных квадратичных поверхностей. Алгебраическая квадратичная поверхность задается в общем виде уравнением

$$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fxz + Gx + Hy + Jz + K = 0, \quad (6-17)$$

где $A, B, C, D, E, F, G, H, J, K$ — константы. Уравнение (6-17) является обобщением уравнения конических сечений (см. уравнение 4-31) на три измерения. В матричной форме обобщенное квадратичное уравнение можно записать как

$$[X][S][X]^T = 0, \quad (6-18)$$

где $[X] = [x \ y \ z \ 1]$, а

$$[S] = \begin{bmatrix} A & D/2 & F/2 & G/2 \\ D/2 & B & E/2 & H/2 \\ F/2 & E/2 & C & J/2 \\ G/2 & H/2 & J/2 & K \end{bmatrix} = \left(\frac{1}{2}\right) \begin{bmatrix} 2A & D & F & G \\ D & 2B & E & H \\ F & E & 2C & J \\ G & H & J & 2K \end{bmatrix}$$

Как и в случае конических сечений, квадратичные поверхности бывают либо центральными, либо нецентральными. Центральные квадрики — это эллипсоид и гиперболоид¹. Параболоиды — это нецентральные поверхности. Если квадрика центральная, то ее центр можно перенести в начало координат. Если это невозможно, то квадрика нецентральная, т. е. является параболоидом. Перенос центра в начало координат приводит к исчезновению линейных членов уравнения (6-17).

После переноса уравнение (6-18) превращается в

$$[X][Tr][S][Tr]^T[X]^T = [X][S'][X]^T = 0, \quad (6-19)$$

¹ Сфера есть частный случай эллипсоида. Цилиндр является предельным случаем для эллипса либо гиперболоида.

где матрица переноса имеет вид:

$$[Tr] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ l & m & n & 1 \end{bmatrix}.$$

Преобразованная матрица $[S']$ равна

$$[S'] = \left(\frac{1}{2}\right) \begin{bmatrix} 2A' & D' & F' & G' \\ D' & 2B' & E' & H' \\ F' & E' & 2C' & J' \\ G' & H' & J' & 2K' \end{bmatrix} \quad (6-20)$$

с $A' = A, B' = B, C' = C, D' = D, E' = E, F' = F$ и

$$\begin{aligned} G' &= 2Al + Dm + Fn + G, \\ H' &= Dl + 2Bm + En + H, \\ J' &= Fl + Em + 2Cn + J, \\ 2K' &= lg' + mh' + nj' + (gl + hm + jn + 2K). \end{aligned}$$

Исключение линейных членов, т. е. случай $C' = H' = J' = 0$, приводит к

$$\begin{aligned} 2Al + Dm + Fn + G &= 0, \\ Dl + 2Bm + En + H &= 0, \\ Fl + Em + 2Cn + J &= 0 \end{aligned}$$

или

$$[l \ m \ n] \begin{bmatrix} 2A & D & F \\ D & 2B & E \\ F & E & 2C \end{bmatrix} = \begin{bmatrix} -G \\ -H \\ -J \end{bmatrix}, \quad (6-21)$$

что можно записать в следующем виде:

$$[M][L] = [G].$$

Если матрица $[L]$ обратима, то решение существует и квадрика центральна, т. е. является эллипсоидом или гиперболоидом. Если же $[L]$ особенная (вырожденная) матрица, то решения не существует и квадрика является параболоидом. Приравнивание детерминанта $[L]$ к нулю дает условие для параболоида. Таким образом,

$$\det[L] = \begin{bmatrix} 2F & D & F \\ D & 2B & E \\ F & E & 2C \end{bmatrix} = 0 \quad (6-22)$$

$A(E^2 - BC) + B(F^2 - AC) + C(D^2 - AB) - (ABC + DEF) = 0$ является условием для параболоида. Далее, если $\det[L] > 0$, то мы получаем гиперболоид, и если $\det[L] < 0$ — эллипсоид.

Как и для конических сечений (см. разд. 4-10), независимо от того, обратима $[L]$ или нет, оси квадрики с помощью поворотов можно сделать параллельными координатным осям. Здесь требуется три поворота: два для того, чтобы сделать одну из осей квадрики параллельной координатной оси (см. разд. 3-9) и последний поворот вокруг этой же оси для того, чтобы сделать две другие оси квадрики параллельными другим координатным осям. Для квадрики в обобщенном виде решение получающихся уравнений для необходимых углов поворотов является нетривиальной задачей и здесь не приводится.

Аналогичным образом, если квадрика нецентральна, т. е. является параболоидом, то как было ранее упомянуто, нельзя исключить из уравнения все линейные члены. Тем не менее, два из трех линейных членов могут быть исключены. И снова, решение получающихся при этом уравнений является нетривиальной задачей и здесь не приводится.

Результатом вышеперечисленных операций должно было быть приведение квадрики к стандартному виду. Для центральной квадрики в стандартном виде центр находится в начале координат, а оси направлены в соответствии с координатными осями. Запишем получившееся стандартное уравнение в матричном виде

$$[X] \begin{bmatrix} \alpha & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \gamma & 0 \\ 0 & 0 & 0 & -\kappa \end{bmatrix} [X]^T = 0 \quad (6-23)$$

или

$$\alpha x^2 + \beta y^2 + \gamma z^2 = \kappa. \quad (6-24)$$

Если $\alpha \neq \beta \neq \gamma > 0$, $\kappa > 0$, то мы получим обобщенный эллипсоид (см. рис. 6-18a). Наибольшее из значений α , β , γ определяет главную ось. Если две из трех констант равны, например, $\alpha = \beta \neq \gamma > 0$, то мы получим эллипсоид вращения. Ось вращения связана с константой, отличной от двух остальных. Если $\alpha = \beta = \gamma > 0$, то мы получим сферу радиуса $\sqrt{\kappa/\alpha}$.

Переписав уравнение (6-24) в виде

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1$$

и устремив одну из констант к бесконечности, получим цилиндр как предельный случай эллипсоида. «Ось» этого эллипсоида, заданная отсутствующей координатой, «бесконечно» длина. Если две оставшиеся константы равны, например $\alpha = \beta$, то мы получим круговой цилиндр. Если они не равны, например $\alpha \neq \beta$, то получим эллиптический цилиндр. В этом случае уравнение (6-24) превращается в

$$\alpha x^2 + \beta y^2 = \kappa.$$

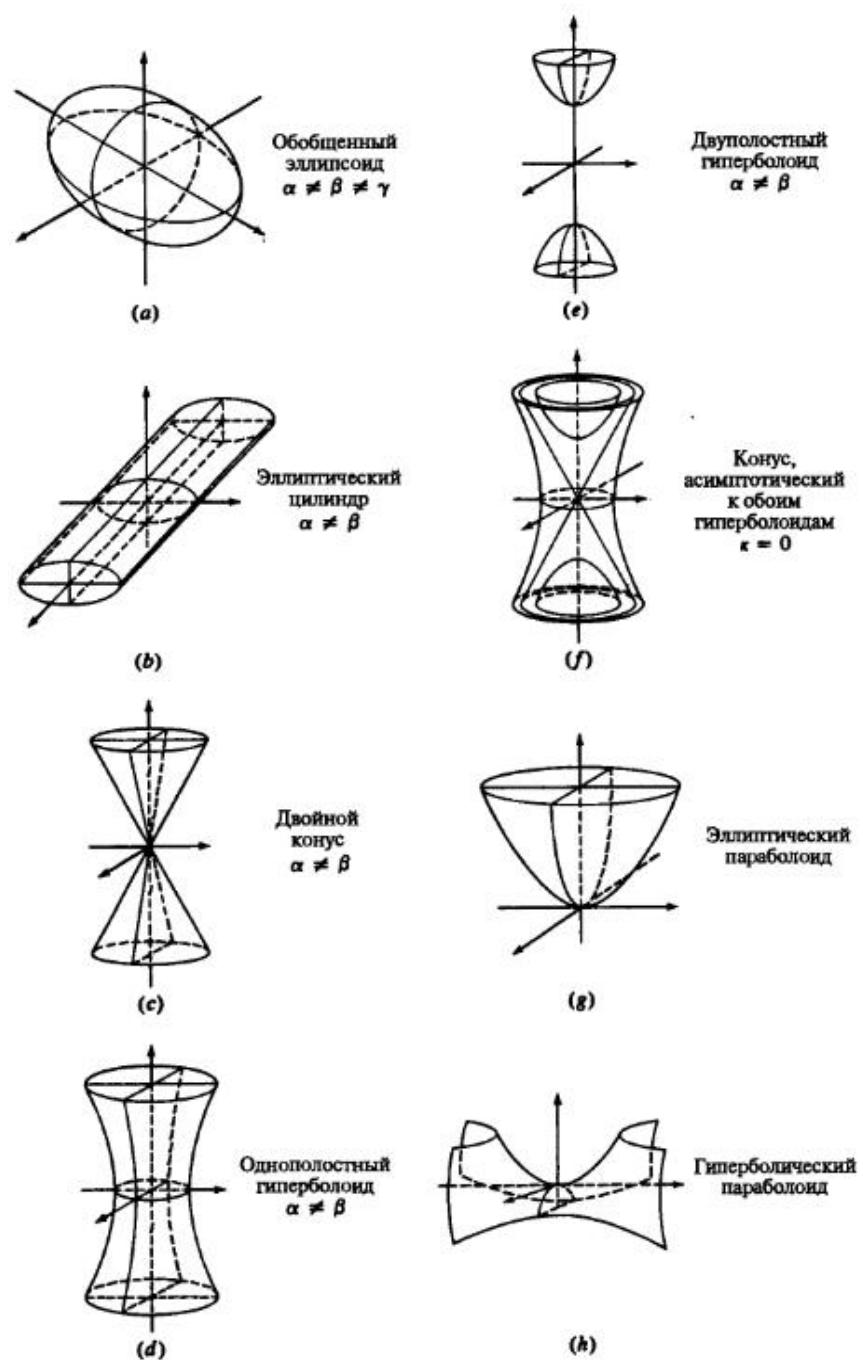


Рис. 6-18 Квадратичные поверхности. (a) Обобщенный эллипсоид $\alpha \neq \beta \neq \gamma$; (b) эллиптический цилиндр $\alpha \neq \beta$; (c) двойной конус $\alpha \neq \beta$; (d) однополостный гиперболоид $\alpha \neq \beta$; (e) двуполостный гиперболоид $\alpha \neq \beta$; (f) конус, асимптотический к обоим гиперболоидам $\kappa = 0$; (g) эллиптический параболоид; (h) гиперболический параболоид.

Посмотрим пример, приведенный на рис. 6-18b. Если $\kappa = 0$, то мы получим мнимый цилиндр, потому что сумма квадратов двух вещественных чисел не может быть равной нулю¹.

Если $\kappa = 0$, одна из констант отрицательна, а две другие положительны, например $\alpha, \beta > 0, \gamma < 0$, то мы получим двойной конус. Ось конуса связана с координатой отрицательной константы. Если $\alpha = \beta$, то получится правильный круговой конус. Если $\alpha \neq \beta$, получится эллиптический конус. В этом случае уравнение (6-24) превращается в

$$\alpha x^2 + \beta y^2 = \gamma z^2.$$

Пример изображен на рис. 6-18c.

Если $\kappa = 0, \alpha, \beta, \gamma \neq 0$, то конус мнимый, потому что сумма квадратов трех вещественных чисел не может быть равной нулю.

Если $\alpha, \beta, \gamma > 0$, а $\kappa < 0$, получится мнимый эллипсоид, потому что опять сумма квадратов трех вещественных чисел не может быть отрицательной.

Если одна из констант в уравнении (6-24) отрицательна, например $\gamma < 0$, а другие положительны, в результате квадратичная поверхность будет однополостным гиперболоидом. Ось гиперболоида связана с отрицательной константой, например, если $\gamma < 0$, то осью гиперболоида будет координатная ось z . На рис. 6-18d приведен пример. Если две положительные константы равны между собой, например $\alpha = \beta$, то получается гиперболоид вращения. В противном случае — эллиптический гиперболоид. Гиперболоид называется однополостным, потому что можно соединить любые две точки поверхности, не выходя за ее границы. В пределе, при приближении отрицательной константы к нулю, получится либо круговой, либо эллиптический цилиндр. Однополостный гиперболоид является двулинейчатой поверхностью (см. разд. 6-8). Двуполостный гиперболоид в некотором смысле противоположен однополостному. Здесь отрицательны две из трех констант α, β, γ , например $\alpha, \beta < 0$, а третья положительна. Ось гиперболоида связана с положительной константой. Пример приведен на рис. 6-18e. Соединить две любые точки двуполостного гиперболоида, не покидая поверхность, невозможно. Для $\alpha, \beta < 0, \gamma > 0$ вершины двух полостей гиперболоида находятся в диапазоне

$$\pm \left(\frac{\alpha \beta \kappa}{\gamma^2} \right)^{1/6}$$

на оси z^2 . Если отрицательные константы равны, то получается гиперболоид вращения. В противном случае получается эллиптический гиперболоид. Заметим, что число отрицательных констант равно числу полостей гиперболоида.

Если $\kappa = 0$, то оба гиперболоида вырождаются в конус, асимптотический к обоим из них, как это показано на рис. 6-18f.

¹ Кроме случая $x = y = 0$, т. е. прямой, совпадающей с осью z . — Прим. перев.

² $\pm \left(\frac{\alpha \beta \kappa}{\gamma^2} \right)^{1/6} = \pm c$, когда уравнение двуполостного гиперболоида записано в менее общем виде $\frac{x^2}{c^2} - \frac{y^2}{a^2} - \frac{z^2}{b^2} = 1$.

Теперь рассмотрим нецентральные квадратичные поверхности, а именно параболоиды. В стандартном виде, где z является осью параболоида, имеем уравнение

$$[X] \begin{bmatrix} \alpha & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & 0 \end{bmatrix} [X]^T = 0 \quad (6-25)$$

или

$$\alpha x^2 + \beta y^2 + \gamma z = 0. \quad (6-26)$$

Если $\alpha, \beta > 0$, то, как показано на рис. 6-18g, получается эллиптический параболоид. Если $\alpha = \beta > 0$, то получим параболоид вращения. Вершина параболоида находится в начале координат. При $\gamma < 0$ поверхность «открыта» в положительную сторону оси z . Аналогичным образом, при $\gamma > 0$ поверхность открыта в отрицательную сторону оси z . Если $\gamma = 0$, поверхность превращается в эллипс¹. Если либо α , либо $\beta < 0$, то, как показано на рис. 6-18h, получается гиперболический параболоид. Если $\alpha > 0, \beta < 0$, то фокальной осью является ось y . Аналогично, если $\alpha < 0, \beta > 0$, фокальная — ось x . Если $\gamma < 0$, то седло направлено вверх, как это показано на рис. 6-18h. Если $\gamma > 0$, то поверхность отражена относительно плоскости $z = 0$ и седло направлено вниз. Если $\gamma = 0$, то поверхность вырождается в гиперболу². Гиперболический параболоид иногда называют седловой поверхностью. Заметим, что даже при $\alpha = \beta$ гиперболический параболоид не является поверхностью вращения. Тем не менее это двулинейчатая поверхность (см. разд. 6-8).

И наконец, если либо α , либо $\beta = 0$, то получаем параболический цилиндр. Дальнейшую информацию о квадратических поверхностях и их свойствах можно найти в [6-12] — [6-14].

Хотя для нахождения свойств квадратичных поверхностей часто бывает полезна явная форма их задания, но, аналогично случаю конических сечений, параметрическое задание поверхностей позволяет получить более привлекательные и доходчивые графические изображения. Для квадратичных поверхностей со стандартной ориентацией параметрические представления задаются следующим образом:

Эллипсоид:

$$\begin{aligned} x &= a \cos \theta \sin \phi, & 0 \leq \theta \leq 2\pi, \\ y &= b \sin \theta \sin \phi, & 0 \leq \phi \leq 2\pi, \\ z &= c \cos \phi. \end{aligned} \quad (6-27)$$

¹ Так как знаки α и β одинаковые, то это прямая $x = y = 0$, т. е. совпадающая с осью z . — Прим. перев.

² Так как у α, β разные знаки, то получается пара скрещивающихся плоскостей. — Прим. перев.

Однополостный гиперболоид:

$$\begin{aligned}x &= a \cos \theta \operatorname{ch} \phi, & 0 \leq \theta \leq 2\pi, \\y &= b \sin \theta \operatorname{sh} \phi, & -\pi \leq \phi \leq \pi, \\z &= c \operatorname{sh} \phi.\end{aligned}\quad (6-28)$$

Двуполостный гиперболоид:

$$\begin{aligned}x &= \pm a \operatorname{ch} \phi, & 0 \leq \theta \leq 2\pi, \\y &= b \sin \theta \operatorname{sh} \phi, & -\pi \leq \phi \leq \pi, \\z &= c \cos \theta \operatorname{sh} \phi.\end{aligned}\quad (6-29)$$

Эллиптический параболоид:

$$\begin{aligned}x &= a \phi \cos \theta, & 0 \leq \theta \leq 2\pi, \\y &= b \phi \sin \theta, & 0 \leq \phi \leq \phi_{\max}, \\z &= \phi^2.\end{aligned}\quad (6-30)$$

Гиперболический параболоид:

$$\begin{aligned}x &= a \phi \operatorname{ch} \theta, & -\pi \leq \theta \leq \pi, \\y &= b \phi \operatorname{sh} \theta, & \phi_{\min} \leq \phi \leq \phi_{\max}, \\z &= \phi^2.\end{aligned}\quad (6-31)$$

Эллиптический конус:

$$\begin{aligned}x &= a \phi \cos \theta, & 0 \leq \theta \leq 2\pi, \\y &= b \phi \sin \theta, & \phi_{\min} \leq \phi \leq \phi_{\max}, \\z &= c \phi.\end{aligned}\quad (6-32)$$

Эллиптический цилиндр:

$$\begin{aligned}x &= a \cos \theta, & 0 \leq \theta \leq 2\pi, \\y &= b \sin \theta, & \phi_{\min} \leq \phi \leq \phi_{\max}, \\z &= \phi.\end{aligned}\quad (6-33)$$

Параболический цилиндр:

$$\begin{aligned}x &= a \theta^2, & 0 \leq \theta \leq \theta_{\max}, \\y &= 2a\theta, & \phi_{\min} \leq \phi \leq \phi_{\max}, \\z &= \phi.\end{aligned}\quad (6-34)$$

Параметрически заданные эллипсоид и часть гиперболического параболоида показаны на рис. 6-19.

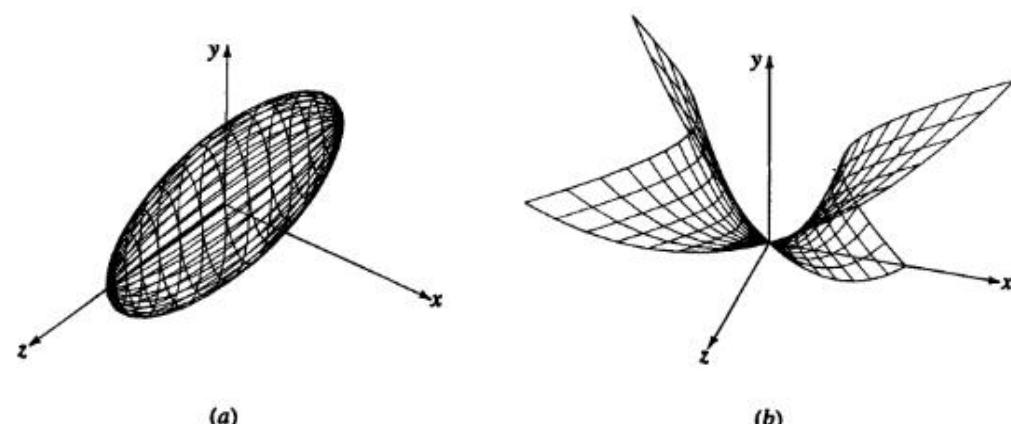


Рис. 6-19 Параметрические квадратичные поверхности. (a) Эллипсоид; (b) гиперболический параболоид.

Хотя представлять все квадратичные поверхности удобно с помощью обобщенного уравнения второй степени (см. уравнение (6-17)), а генерировать — с помощью параметрического задания, описанного выше, но при компьютерном моделировании квадратичные поверхности наиболее точно представляются геометрически (см. [6-15] и [6-17]). Геометрически любая квадратичная поверхность представляется точкой, двумя ортогональными (единичными) векторами и тремя скалярами. Точка — либо центр, либо вершина, фиксирует местоположение в пространстве. Вектора и их векторное произведение определяют оси поверхности или ориентацию. Скаляры определяют ее измерения или размер. Например, сфера определяется своим центром и радиусом, правильный цилиндр — вектором, задающим его ось, точкой на оси и радиусом, эллипсоид определяется центром, двумя векторами, представляющими две из его трех ортогональных осей и тремя скалярами, представляющими длины вдоль этих осей, и т. д. В табл. 6-1 перечислены геометрические описания квадратичных поверхностей.

При работе с геометрически заданными квадратичными поверхностями необходимо преобразовывать только определяющие точки и векторы. Скалярные величины остаются неизменными и, таким образом, не накапливают при вычислениях ошибок, возникающих из-за использования в компьютере арифметики с ограниченной разрядной сеткой.

Например, геометрически заданный радиус сферы всегда равен R . А радиус сферы, полученный путем преобразования алгебраического представления (см. уравнение (6-17)) равен $R \pm \epsilon$, где ϵ — небольшая числовая ошибка. Числовая стабильность определяющих скаляров очень важна, например при определении характера кривой пересечения двух квадратичных поверхностей или при определении идентичности двух поверхностей (см. [6-15]).

Таблица 6-1 Геометрические описания квадратичных поверхностей

Поверхность	Скаляры	Точка	Векторы
Плоскость	Нет	Любая точка на плоскости	Единичная нормаль
Сфера	Радиус	Центр	Нет
Правильный круговой конус	Половинный угол	Центр (вершина)	Единичный вектор, параллельный оси
Правильный круговой цилиндр	Радиус	Любая точка на оси	Единичный вектор, параллельный оси
Правильный эллиптический цилиндр	Длины осей эллипса (2)	Любая точка на оси	—
Правильный параболический цилиндр	Фокусное расстояние	Вершина	Единичный вектор оси
Эллиптический конус	Длины осей эллипса (2)	Центр	Единичный вектор, параллельный оси
Эллипсоид	Длины осей (3)	Центр	Два единичных вектора, параллельных двум из трех осей
Эллиптический параболоид	Длины осей эллипса	Вершина	Единичный вектор, параллельный оси
Гиперболический параболоид	Гиперболический и параболический фокусы (3)	Вершина	Единичный вектор, параллельный оси
Однополостный гиперболоид	Длины осей эллипса (2) Гиперболический фокус вдоль главной оси	Центр	Единичный вектор, параллельный оси
Двуполостный гиперболоид	Длины осей (2) Местоположение вершины на оси симметрии (1)	Центр	Единичный вектор, параллельный оси

6-5 КУСОЧНОЕ ПРЕДСТАВЛЕНИЕ ПОВЕРХНОСТЕЙ

В предыдущих разделах обсуждались создание и характеристики поверхностей, для которых существуют известные аналитические описания. Однако есть много поверхностей, для которых таких описаний сделать невозможно. Типичными примерами являются автомобильные кузова, фюзеляжи и крылья самолетов, корпуса кораблей, скульптуры, бутылки, башмаки и т. д. Эти поверхности представляются в кусочном виде, т. е. похожем на лоскутное одеяло. В рассуждениях используется векторное параметрическое представление, потому что оно не зависит от осей, устраняет бесконечные значения углов наклона относительно некоторой произвольной системы координат, устраняет неоднозначность представления многозначных поверхностей и пространственных функций, облегчает представле-

ние поверхностей в однородных координатах, и такое представление совместимо с использованием преобразований трехмерных однородных координат, обсуждавшихся в гл. 3.

Остаток этой главы будет посвящен обсуждению способов и методов математического описания кусков поверхностей.

Наша цель заключается в том, чтобы для создания полной поверхности объединить вместе отдельные куски вдоль их границ. Для кусочного представления кривых мы будем придерживаться принятого в гл. 5 изложения. Мы начнем наше обсуждение с достаточно интересного элемента или куска аналитической поверхности, а именно сферы.

На примере сферической поверхности мы продемонстрируем некоторые свойства кривых, полезные при описании поверхности. Определенные кривые на поверхности задаются плоскостями, пересекающими сферу. В качестве примера рассмотрим пересечение единичной сферы и плоскости, определяемое уравнением поверхности $z = \cos \phi_1 = a_1 = \text{constant}$ и показанное на рис. 6-20a. Получившаяся кривая является параллелью широты. Уравнение этой кривой получается путем решения системы двух уравнений поверхностей. Непараметрическое уравнение единичной сферы записывается в виде:

$$x^2 + y^2 + z^2 = 1.0.$$

Таким образом,

$$x^2 + y^2 = 1 - a_1^2$$

определяет пересечение плоскости и сферы.

На рис. 6-20b плоскость $\theta = \theta_0 = \text{constant}$ задается уравнением

$$x \sin \theta_0 - y \cos \theta_0 = 0$$

или

$$c_1 x - b_1 y = 0.$$

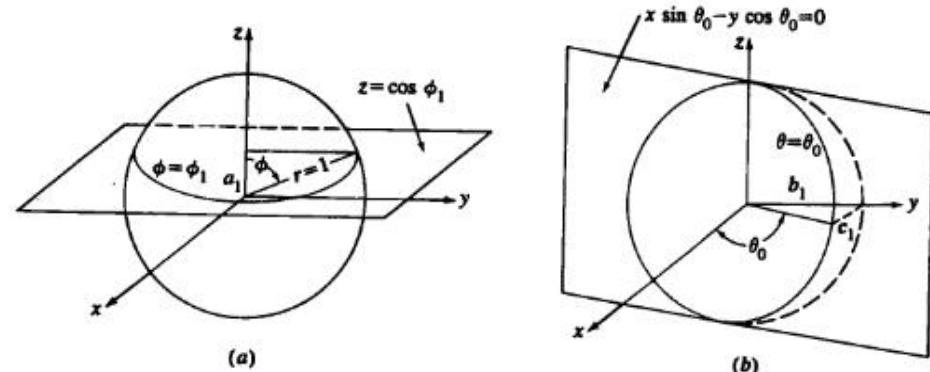


Рис. 6-20 Пересечение плоскости и сферы. (a) Параллель широты; (b) меридиан долготы.

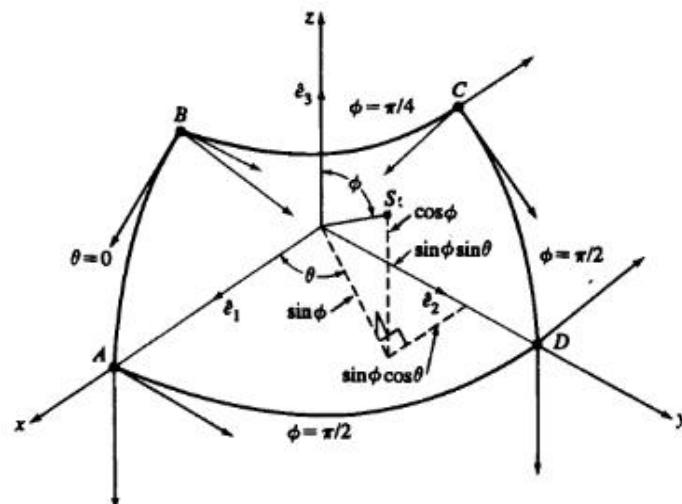


Рис. 6-21 Кусок сферической поверхности.

В результате пересечения этой плоскости и сферы получается меридиан долготы. Решение системы уравнений дает уравнение кривой пересечения; т. е.

$$y^2 \left[\left(\frac{b_1}{c_1} \right)^2 + 1 \right] + z^2 = 1.0.$$

Как показано на рис. 6-21, границы куска сферической поверхности на единичной сфере могут быть сформированы четырьмя плоскостями, двумя параллелями и двумя меридианами, пересекающими сферу. Векторное параметрическое уравнение для куска поверхности $Q(\theta, \phi)$

$$Q(\theta, \phi) = [\cos \theta \sin \phi, \quad \sin \theta \sin \phi, \quad \cos \phi] \quad \theta_1 \leq \theta \leq \theta_2, \quad \phi_1 \leq \phi \leq \phi_2. \quad (6-35)$$

Кусок поверхности — это все возможные положения точки в трехмерном пространстве, движущейся с двумя степенями свободы, управляемой двумя параметрическими переменными θ и ϕ , т. е. это бипараметрическая функция.

Параметрическое представление единичной сферической поверхности (см. уравнение (6-27) с $a = b = c = 1$) таково:

$$\begin{aligned} x &= \cos \theta \sin \phi, \\ y &= \sin \theta \sin \phi, \\ z &= \cos \phi. \end{aligned}$$

Показанный на рис. 6-21 кусок поверхности задан для диапазонов изменения параметров $0 \leq \theta \leq \pi/2$ и $\pi/4 \leq \phi \leq \pi/2$. Границы или ребра его задаются кривыми AB , BC , CD и DA . Для сферического куска на рис. 6-21 эти кривые являются дугами окружностей. Каждая кривая может быть задана двумя конечными точками

и касательными векторами в концах. Следовательно, четыре граничных кривых куска задаются четырьмя координатными векторами в углах и восемью касательными векторами, по два в каждом углу. Для сферического куска на рис. 6-21 касательные векторы задаются параметрическими производными $Q(\theta, \phi)$, т. е.

$$Q_\theta(\theta, \phi) = \frac{\partial Q}{\partial \theta}(\theta, \phi) = [-\sin \theta \sin \phi, \quad \cos \theta \sin \phi, \quad 0] \quad (6-36)$$

и

$$Q_\phi(\theta, \phi) = [\cos \theta \cos \phi, \quad \sin \theta \cos \phi, \quad -\sin \phi]. \quad (6-37)$$

Касательные векторы в каждом углу показаны на рис. 6-21.

Форма внутренней части поверхности около каждого угла управляет вектором кручения или смешанной производной в этом углу. Для куска сферической поверхности на рис. 6-21 смешанная производная или вектор кручения

$$Q_{\theta, \phi}(\theta, \phi) = \frac{\partial^2 Q}{\partial \theta \partial \phi} = \frac{\partial^2 Q}{\partial \phi \partial \theta} = [-\sin \theta \cos \phi, \quad \cos \theta \cos \phi, \quad 0]. \quad (6-38)$$

Подстановка значений параметров в углах куска поверхности дает векторы кручения в них. Внутренняя форма куска задается уравнением (6-35). Это уравнение можно считать сферической смешивающей функцией. Следовательно, четырехугольный кусок поверхности может быть полностью описан 4 координатными векторами в углах, 8 касательными векторами (по два в каждом углу), 4 векторами кручения в углах и смешивающей функцией, заданной в уравнении (6-35).

Нормаль в любой точке куска поверхности определяется векторным произведением производных по параметрам. Например, для сферической поверхности

$$\begin{aligned} Q_\theta \times Q_\phi &= \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ -\sin \theta \sin \phi & \cos \theta \sin \phi & 0 \\ \cos \theta \cos \phi & \sin \theta \cos \phi & -\sin \phi \end{vmatrix} \\ &= [-\cos \theta \sin^2 \phi, \quad \sin \theta \sin^2 \phi, \quad -\sin \phi \cos \phi]. \end{aligned} \quad (6-39)$$

На куске поверхности изопараметрические линии, т. е. линии с постоянным значением параметра, ортогональны. Следовательно, скалярное произведение производных по параметрам равно нулю. Например, для сферической поверхности

$$\begin{aligned} Q_\theta \cdot Q_\phi &= [-\sin \theta \sin \phi, \quad \cos \theta \sin \phi, \quad 0] \cdot [\cos \theta \cos \phi, \quad \sin \theta \cos \phi, \quad -\sin \phi] = 0 \\ &= [-\sin \theta \sin \phi, \quad \cos \theta \sin \phi, \quad 0] \cdot [\cos \theta \cos \phi, \quad \sin \theta \cos \phi, \quad -\sin \phi]^T = 0 \\ &= [-\cos \theta \sin \theta \cos \phi \sin \phi, \quad \cos \theta \sin \theta \cos \phi \sin \phi, \quad 0] = 0. \end{aligned} \quad (6-40)$$

6-6 ОТОБРАЖЕНИЕ ПАРАМЕТРИЧЕСКИХ ПОВЕРХНОСТЕЙ

Методы параметрического описания поверхностей наиболее удобно представляются в терминах отображения двупараметрической плоской поверхности из параметрического пространства uv в трехмерное объективное пространство xuz . Ограничимся здесь обсуждением отображения прямоугольной плоской поверхности

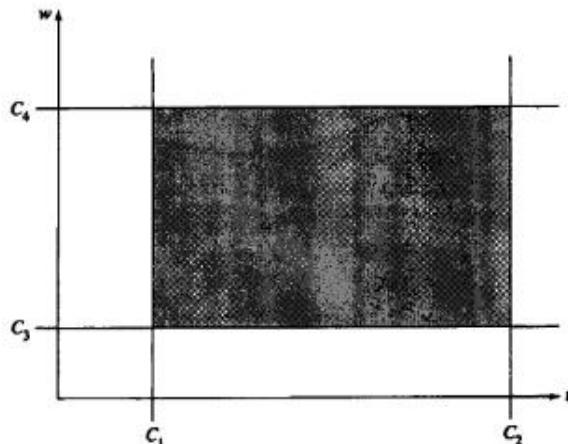


Рис. 6-22 Прямоугольная параметрическая плоская поверхность.

в параметрическом пространстве, показанной на рис. 6-22 и заданной уравнениями

$$\begin{aligned} u &= C_1 \quad C_3 \leq w \leq C_4, \\ u &= C_2 \quad C_3 \leq w \leq C_4, \\ w &= C_3 \quad C_1 \leq u \leq C_2, \\ w &= C_4 \quad C_1 \leq u \leq C_2. \end{aligned}$$

Поверхность в объектном пространстве представляется функциями, отображающими эту параметрическую поверхность в объектное пространство \$xyz\$, т. е.

$$\begin{aligned} x &= x(u, w), \\ y &= y(u, w), \\ z &= z(u, w). \end{aligned}$$

Простой двумерный пример послужит в качестве иллюстрации этого метода.

Пример 6-7 Двумерное отображение поверхности

Отобразить поверхность, описываемую в параметрическом пространстве уравнениями

$$\begin{aligned} x &= 3u + w \quad 0 \leq u \leq 1, \\ y &= 2u + 3w + uw \quad 0 \leq w \leq 1, \\ z &= 0 \end{aligned}$$

в объектное пространство. Для начала заметим, что так как \$z = \text{constant} = 0\$, то поверхность в объектном пространстве также является двумерной и лежит на плоскости \$z = 0\$.

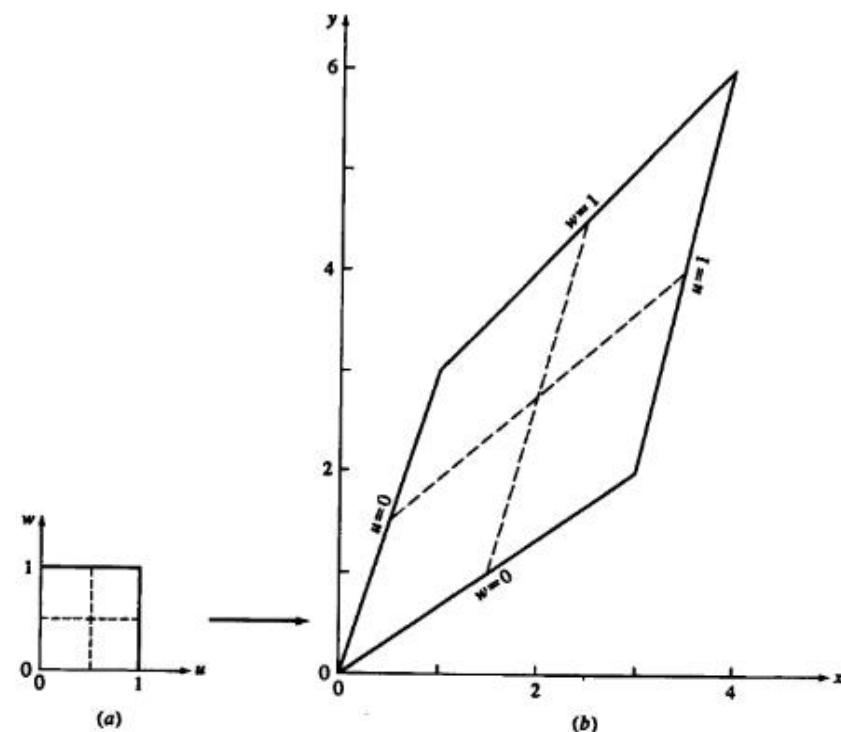


Рис. 6-23 Двумерное отображение поверхности. (a) Параметрическое пространство; (b) объектное пространство.

Границы поверхности в объектном пространстве определяются с помощью отображения в объектное пространство границ прямоугольника в параметрическом пространстве. Таким образом, для

$$\begin{aligned} u &= 0; & x &= w, & y &= 3w \quad \text{и} \quad y = 3x, \\ u &= 1; & x &= w + 3, & y &= 2(2w + 1) \quad \text{и} \quad y = 2(2x - 5), \\ w &= 0; & x &= 3u, & y &= 2u \quad \text{и} \quad y = (2/3)x, \\ w &= 1; & x &= 3u + 1, & y &= 3u + 3 \quad \text{и} \quad y = x + 2. \end{aligned}$$

Во всех случаях для получения уравнения вида \$y = y(x)\$ из соответствующего выражения исключался параметр (\$u\$ или \$w\$). Результаты представлены на рис. 6-23.

Как показано в примере, задание постоянного значения одному из параметров порождает кривую на поверхности в объектном пространстве. Такая кривая называется изопараметрической или параметрической линией. Если задать один из параметров как функцию другого в параметрическом пространстве, т. е. \$u = u(w)\$, то в результате также получится кривая на поверхности в объектном простран-

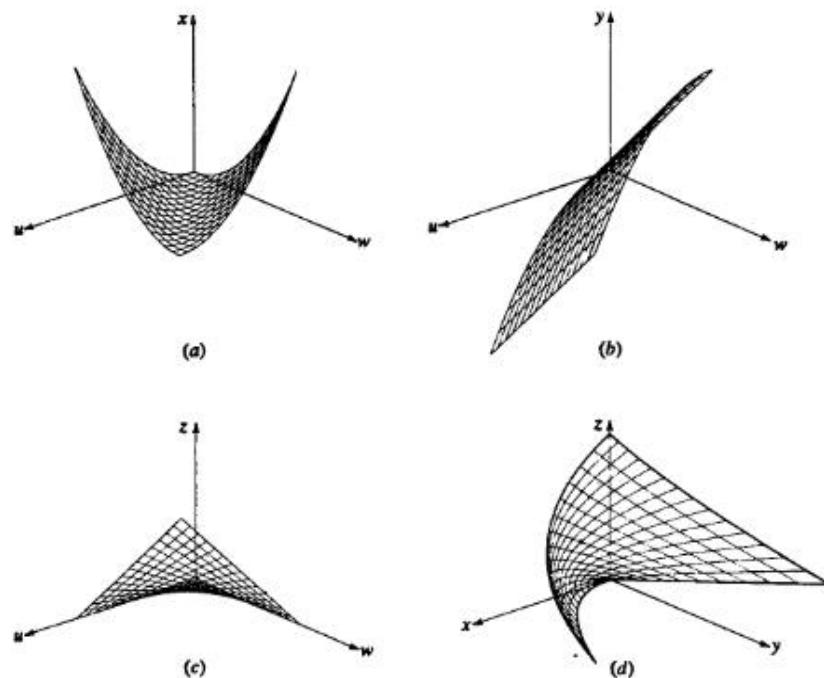


Рис. 6-24 Трехмерное отображение поверхности. (a) x компонента; (b) y компонента; (c) z компонента; (d) результат.

стве. Например, функции

$$u = w, \quad 0 \leq w \leq 1,$$

$$u = 1 - w,$$

представляют диагонали единичного квадрата в параметрическом пространстве.

Спецификация значений обоих параметров задает точку на поверхности в объектном пространстве. Другим способом задания точки (или точек) может служить пересечение двух кривых в параметрическом пространстве, например $f(u, w) = 0$ и $g(u, w) = 0$. Пересечение в параметрическом пространстве отображается или преобразуется в пересечение в объектном пространстве.

В более сложном трехмерном примере дополнительно иллюстрируется описываемая идея отображения.

Пример 6-8 Трехмерное отображение поверхности

Отобразить описанную в параметрическом пространстве поверхность

$$x(u, w) = (u - w)^2, \quad 0 \leq u \leq 1,$$

$$y(u, w) = u - w^2, \quad 0 \leq w \leq 1,$$

$$z(u, w) = uw$$

в объектное пространство. Вычислить координаты в объектном пространстве точки на поверхности с параметрами $u = w = 0.5$.

Сначала найдем граничные кривые

$$u = 0; \quad x = w^2, y = -w^2, z = 0 \quad \text{и} \quad x = -y, z = 0,$$

$$u = 1; \quad x = (1 - w)^2, y = 1 - w^2, z = w \quad \text{и} \quad x = (1 - z)^2, y = 1 - z^2,$$

$$w = 0; \quad x = u^2, y = u, z = 0 \quad \text{и} \quad x = y^2, z = 0,$$

$$w = 1; \quad x = (u - 1)^2, y = u - 1, z = u \quad \text{и} \quad x = y^2, z = 1 + y.$$

Границные кривые изображены на рис. 6-24d более толстыми линиями. Записав параметрическую поверхность в виде векторной функции

$$Q(u, w) = [x(u, w) \quad y(u, w) \quad z(u, w)] = [(u - w)^2 \quad u - w^2 \quad uw]$$

получим

$$Q(0.5, 0.5) = [0 \quad 0.25 \quad 0.25]$$

в качестве координат точки $u = w = 0.5$, отмеченной на рис. 6-24d жирной точкой. Заметим, что каждая из компонент поверхности в объектном пространстве является также функцией параметров u, w . Каждая из этих отдельных компонент показана на рис. 6-24a, b, c. Поверхность, изображенная на рис. 6-24d, является композицией всех преобразованных компонент.

И наконец, представляют интерес отображения вырожденных кусков, соответствующих точке и прямой. Для точки отображение записывается

$$x = \text{constant}, \quad y = \text{constant}, \quad z = \text{constant}.$$

Для прямой отображение записывается в виде $x = u, y = \text{constant}, z = \text{constant}$.

6-7 БИЛИНЕЙНАЯ ПОВЕРХНОСТЬ

Одной из самых простых является билинейная поверхность. Билинейная поверхность конструируется из четырех угловых точек единичного квадрата в параметрическом пространстве, т. е. из точек $P(0, 0), P(0, 1), P(1, 1)$ и $P(1, 0)$. Любая точка на поверхности определяется линейной интерполяцией между противоположными границами единичного квадрата, как это показано на рис. 6-25. Любая точка внутри параметрического квадрата задается уравнением

$$Q(u, w) = P(0, 0)(1 - u)(1 - w) + P(0, 1)(1 - u)w + P(1, 0)u(1 - w) + P(1, 1)uw. \quad (6-41)$$

В матричном виде

$$Q(u, w) = [1 - u \quad u] \begin{bmatrix} P(0, 0) & P(0, 1) \\ P(1, 0) & P(1, 1) \end{bmatrix} \begin{bmatrix} 1 - w \\ w \end{bmatrix}. \quad (6-42)$$

Необходимо, чтобы интерполируемая поверхность удовлетворяла исходным данным. В этом случае легко проверить, что угловые точки принадлежат этой поверхности, т. е. $Q(0, 0) = P(0, 0)$ и т. д.

Уравнение (6-42) задано в обобщенном матричном представлении интерполированной поверхности, а именно — матрица функций смешения по одной из бипа-

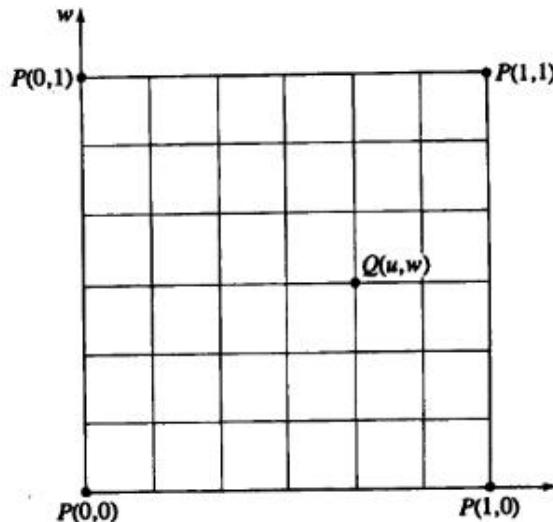


Рис. 6-25 Билинейная интерполяция в параметрическом пространстве.

параметрических переменных, геометрическая матрица, представляющая исходные данные, и матрица функций смешения по другой параметрической переменной. При изучении параметрических интерполированных поверхностей мы будем постоянно пользоваться этим представлением.

Если координатные векторы четырех точек, определяющих билинейную поверхность, заданы в трехмерном объектном пространстве, то будет трехмерна и билинейная поверхность, получаемая в результате отображения параметрического пространства в объектное. Если четыре определяющие точки не лежат в одной плоскости, то и билинейная поверхность также не лежит ни в какой плоскости. Действительно, в общем случае она сильно изогнута, пример этого показан на рис. 6-26. Определяющие точки являются концами противоположных диагоналей на противоположных гранях единичного куба. В результате получаем гиперболический параболоид. В качестве иллюстрации рассмотрим следующий пример.

Пример 6-9 Билинейная поверхность

Найти точку на билинейной поверхности, заданной точками $P(0,0) = [0 \ 0 \ 1]$, $P(0,1) = [1 \ 1 \ 1]$, $P(1,0) = [1 \ 0 \ 0]$, $P(1,1) = [0 \ 1 \ 0]$, т.е. концами противоположных диагоналей, лежащих на противоположных гранях единичного куба в объектном пространстве. Искомая точка имеет координаты $u = w = 0.5$ в параметрическом пространстве.

Напомним, что поверхность в объектном пространстве является векторной функцией:

$$Q(u, w) = [x(u, w) \ y(u, w) \ z(u, w)]$$

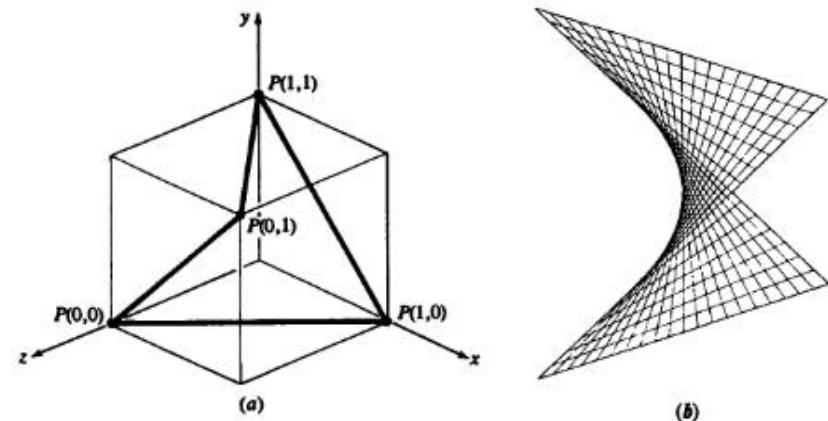


Рис. 6-26 Билинейная поверхность. (a) Определяющие угловые точки; (b) поверхность.

тогда из уравнения (6-41) имеем

$$\begin{aligned} Q(0.5, 0.5) &= [0 \ 0 \ 1](1 - 0.5)(1 - 0.5) + [1 \ 1 \ 1](1 - 0.5)(0.5) \\ &\quad + [1 \ 0 \ 0](0.5)(1 - 0.5) + [0 \ 1 \ 0](0.5)(0.5) \\ &= 0.25[0 \ 0 \ 1] + 0.25[1 \ 1 \ 1] \\ &\quad + 0.25[1 \ 0 \ 0] + 0.25[0 \ 1 \ 0] \\ &= [0.5 \ 0.5 \ 0.5] \end{aligned}$$

Вся поверхность изображена на рис. 6-26б.

Заметим, что каждая изопараметрическая линия на билинейной поверхности является прямой линией. В самом деле, эта поверхность является двулинейчатой (см. разд. 6-8).

6-8 ЛИНЕЙЧАТЫЕ И РАЗВЕРТЫВАЮЩИЕСЯ ПОВЕРХНОСТИ

Линейчатые поверхности часто используются в авиа- и кораблестроительной промышленности. Например, большинство авиационных крыльев являются цилиндрическими линейчатыми поверхностями. Линейчатая поверхность образуется при движении прямой линии вдоль направляющей с одной степенью свободы. Другой метод определения линейчатой поверхности состоит в следующем. Выберем произвольную точку на поверхности и будем вращать вокруг нормали плоскость, проходящую через нормаль к поверхности в этой точке (см. рис. 6-27). Если существует такая ориентация плоскости, при которой каждая точка на ребре плоскости контактирует с поверхностью, то поверхность линейчата в этом направлении. Если ребро вращающейся плоскости полностью соприкасается с поверхностью при более чем одной ориентации, то поверхность в этой точке многолинейчата.

Самой простой линейчатой поверхностью является плоскость. Для квадратичных поверхностей однолинейчаты конусы и цилиндры; однополостный гиперболоид и гиперболический параболоид двулинейчаты.

В терминах отображения параметрического пространства u, w в объектное пространство линейчатая поверхность получается с помощью линейного интерполяирования между двумя известными граничными кривыми, ассоциированными с противоположными сторонами единичного квадрата в параметрическом пространстве, скажем, между кривыми $P(u, 0)$ и $P(u, 1)$. Поверхность задается уравнениями:

$$Q(u, w) = P(u, 0)(1 - w) + P(u, 1)w \quad (6-43)$$

или

$$[Q] = [x(x, w) \quad y(u, w) \quad z(u, w)] = [1 - w \quad w] \begin{bmatrix} P(u, 0) \\ P(u, 1) \end{bmatrix}$$

Заметим, что снова $Q(0, 0) = P(0, 0)$ и т.д., то есть концы специфицированных кривых и углы поверхности совпадают. Кроме того, два края интерполированной поверхности совпадают с заданными кривыми, т.е. $Q(u, 0) = P(u, 0)$ и $Q(u, 1) = P(u, 1)$.

Теперь предположим, что известны кривые, соответствующие $Q(0, w)$ и $P(1, w)$. Тогда линейчатая поверхность задается следующим образом:

$$Q(u, w) = P(0, w)(1 - u) + P(1, w)u \quad (6-44)$$

или

$$[Q] = [x(x, w) \quad y(u, w) \quad z(u, w)] = [1 - u \quad u] \begin{bmatrix} P(0, w) \\ P(1, w) \end{bmatrix}$$

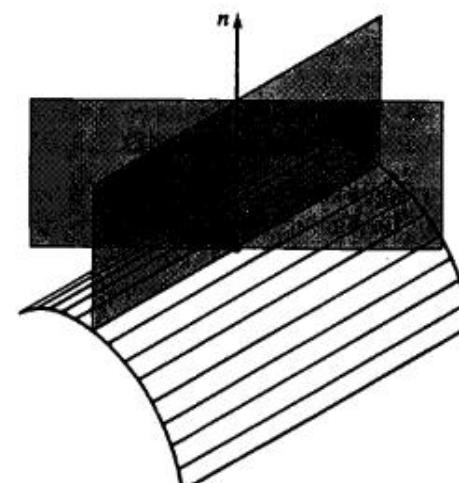


Рис. 6-27 Свойства линейчатой поверхности.

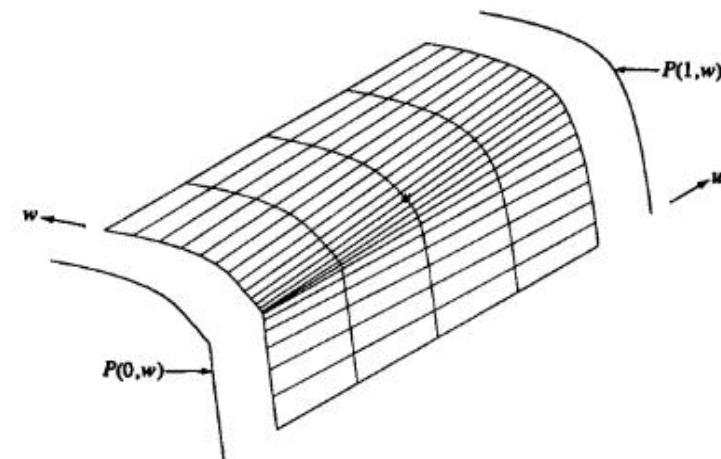


Рис. 6-28 Пример линейчатой поверхности.

В этом случае опять углы поверхности совпадают с концами заданных кривых, а соответствующие края интерполированной поверхности совпадают с заданными граничными кривыми. На рис. 6-28 приводится пример линейчатой поверхности. Изображенные на этом рисунке отстоящими на некотором расстоянии от поверхности граничные кривые являются В-сплайнами третьего порядка (см. разд. 5-9). Описанная методика иллюстрируется на примере.

Пример 6-10 Линейчатая поверхность

Рассмотрим линейчатую поверхность, сформированную линейным интерполяированием кривых $P(0, w)$ и $P(1, w)$. Найдем координаты точки $Q(u, w)$ на поверхности с параметрами $u = w = 0.5$.

$P(0, w)$ является незамкнутым В-сплайном третьего порядка ($k = 3$) с определяющими вершинами ломаной, заданными точками $B_1[0 \ 0 \ 0]$, $B_2[1 \ 1 \ 0]$, $B_3[1 \ 1 \ 0]$, $B_4[2 \ 1 \ 0]$ и $B_5[3 \ 0 \ 0]$ (см. разд. 5-9). Отметим, что вершины $B_2 = B_3$ совпадают, в результате чего на кривой получается излом. $P(1, w)$ также является незамкнутым В-сплайном третьей степени. Его вершины определяющей ломаной равны $\bar{B}_1[0 \ 0 \ 6]$, $\bar{B}_2[1 \ 1 \ 6]$, $\bar{B}_3[2 \ 1 \ 6]$, $\bar{B}_4[3 \ 0 \ 6]$.

Вспоминая обсуждение В-сплайнов, получим незамкнутые однородные узловые векторы для $P(0, w)$ и $P(1, w)$, соответственно,

$$[X] = [0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 3 \ 3],$$

$$[Y] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2].$$

Заметим, что ненормализованные диапазоны изменения параметров для этих двух кривых различны, $0 \leq t \leq 3$ для $P(0, w)$ и $0 \leq s \leq 2$ для $P(1, w)$. «Нормализованное» значение параметра линейчатой поверхности Q в точке $w = 0.5$ соответствует $t = 1.5$ для $P(0, w)$ и $s = 1.0$ для $P(1, w)$.

Используя уравнения (5-83) и (5-84), получим

$$\begin{aligned} P(0, w) &= P(t) \\ &= B_1 N_{1,3}(t) + B_2 N_{2,3}(t) + B_3 N_{3,3}(t) + B_4 N_{4,3}(t) + B_5 N_{5,3}(t). \end{aligned}$$

В точке $w = 0.5$ или $t = 1.5$

$$\begin{aligned} P(0, 0.5) &= P(1.5) = (0)B_1 + 0.125B_2 + 0.75B_3 + 0.125B_4 + (0)B_5 \\ &= 0.125[1 \ 1 \ 0] + 0.75[1 \ 1 \ 0] + 0.125[2 \ 1 \ 0] \\ &= [1.125 \ 1 \ 0]. \end{aligned}$$

Аналогичным образом

$$\begin{aligned} P(1, w) &= P(s) \\ &= \bar{B}_1 \bar{N}_{1,3}(s) + \bar{B}_2 \bar{N}_{2,3}(s) + \bar{B}_3 \bar{N}_{3,3}(s) + \bar{B}_4 \bar{N}_{4,3}(s). \end{aligned}$$

В точке $w = 0.5$ или $s = 1.0$

$$\begin{aligned} P(1, 0.5) &= P(1.0) = (0)\bar{B}_1 + 0.5\bar{B}_2 + 0.5\bar{B}_3 + (0)\bar{B}_4 \\ &= 0.5[1 \ 1 \ 6] + 0.5[2 \ 1 \ 6] \\ &= [1.5 \ 1 \ 6]. \end{aligned}$$

Использование уравнения (6-44) для получения точки на линейчатой поверхности даст

$$Q(u, w) = P(0, w)(1 - u) + P(1, w)u$$

и

$$\begin{aligned} Q(0.5, 0.5) &= P(0, 0.5)(1 - 0.5) + P(1, 0.5)(0.5) \\ &= 0.5[1.125 \ 1 \ 0] + 0.5[1.5 \ 1 \ 6] \\ &= [1.3125 \ 1 \ 3]. \end{aligned}$$

Результаты показаны на рис. 6-28. Жирной точкой отмечена точка поверхности, соответствующая $Q(0.5, 0.5)$. Обратите внимание на то, что кривая $P(0, w)$, содержащая излом, плавно переходит в гладкую кривую $P(1, w)$.

Особый практический интерес представляет вопрос, является ли линейчатая поверхность развертывающейся? Не все линейчатые поверхности развертывающиеся, однако все развертывающиеся поверхности являются линейчатыми.

Если поверхность развертывающаяся, то с помощью последовательности небольших поворотов вокруг образующей линии она может быть без растяжений и разрывов развернута или раскрыта в плоскость. Развертывающиеся поверхности особенно важны для листопрокатной промышленности и в меньшей степени для текстильной промышленности.

Ясно, что среди линейчатых квадратичных поверхностей развертывающимися являются конусы и цилиндры. Однако после небольшого размышления становится ясно, что ни однополостный гиперболоид (см. рис. 6-18d), ни гиперболический параболоид (см. рис. 6-26) не являются развертывающимися поверхностями, хотя они линейчатые.

Чтобы определить, будет ли развертывающейся поверхность или ее часть, необходимо рассмотреть кривизну параметрической поверхности. В произвольной

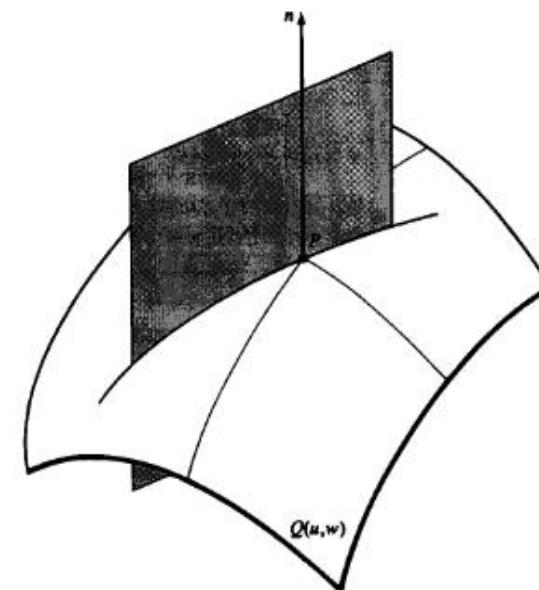


Рис. 6-29 Кривизна бипараметрической поверхности.

точке P на поверхности кривая пересечения поверхности и плоскости, содержащей нормаль к поверхности в точке P , имеет кривизну κ (см. рис. 6-29). При вращении этой плоскости вокруг нормали кривизна меняется. Великий швейцарский математик Эйлер показал, что существуют только два направления, для которых кривизна принимает минимальное и максимальное значения. Кривизны в этих направлениях называются главными кривизнами, κ_{\min} и κ_{\max} . Кроме того, направления главных кривизн ортогональны. Два сочетания главных кривизн представляют особый интерес — средняя и гауссова кривизны:

$$H = \frac{\kappa_{\min} + \kappa_{\max}}{2}, \quad (6-45)$$

$$K = \kappa_{\min} \kappa_{\max}. \quad (6-46)$$

Для развертывающейся поверхности гауссова кривизна K в любой точке равна нулю, т. е. $K = 0$. Ди [6-18] показал, что для бипараметрических поверхностей средняя и гауссова кривизны задаются выражением¹

$$H = \frac{A|Q_w|^2 - 2BQ_u \cdot Q_w + C|Q_u|^2}{2|Q_u \times Q_w|^3}, \quad (6-47)$$

$$K = \frac{AC - B^2}{|Q_u \times Q_w|^4}, \quad (6-48)$$

где

$$(ABC) = [Q_u \times Q_w] \cdot [Q_{uu} \ Q_{uw} \ Q_{ww}].$$

¹ Здесь подстрочные индексы используются для обозначения частных производных, т. е. $Q_u = \partial Q / \partial u$ и т. д.

Таблица 6-2 Типы поверхностей

$\kappa_{\min}, \kappa_{\max}$	K	Форма
Однаковые знаки	< 0	Эллиптическая (выпуклость или впадина)
Противоположные знаки	> 0	Гиперболическая (седловая точка)
Одна или обе кривизны равны нулю	0	Цилиндрическая/коническая (гребень, впадина, плоскость)

Как показано в табл. 6-2, знак гауссовой кривизны характеризует локальную форму поверхности: эллиптическую, гиперболическую, цилиндрическую или коническую. Так как гауссова кривизна развертывающейся поверхности должна быть нулевой, то поверхность должна быть скомпонована из цилиндрических, конических или плоских кусков. Приведенный ниже пример поможет проиллюстрировать эти рассуждения

Пример 6-11 Развертывающаяся поверхность

Показать, что эллиптический конус является развертывающейся поверхностью.

Перепишем уравнение (6-32) для параметрического эллиптического конуса в терминах u и w :

$$Q(u, w) = [a u \cos w \quad b u \sin w \quad c u].$$

Частные производные будут

$$Q_u = [a \cos w \quad b \sin w \quad c],$$

$$Q_w = [-a u \sin w \quad b u \cos w \quad 0],$$

$$Q_{uw} = [-a \sin w \quad b \cos w \quad 0],$$

$$Q_{uu} = [0 \quad 0 \quad 0],$$

$$Q_{ww} = [-a u \cos w \quad -b u \sin w \quad 0],$$

$$Q_u \times Q_w = [-bcu \cos w \quad -acu \sin w \quad abu],$$

$$|Q_u \times Q_w|^2 = (abu)^2 \left\{ \left(\frac{c}{a} \cos w \right)^2 + \left(\frac{c}{b} \sin w \right)^2 + 1 \right\} \neq 0 \quad u > 0$$

и

$$A = [-bcu \cos w \quad -acu \sin w \quad abu] \cdot [0 \quad 0 \quad 0] = 0,$$

$$B = [-bcu \cos w \quad -acu \sin w \quad abu] \cdot [-a \sin w \quad b \cos w \quad 0] \\ = abc u \sin w \cos w - abc u \sin w \cos w = 0,$$

$$C = [-bcu \cos w \quad -acu \sin w \quad abu] \cdot [-a u \cos w \quad -b u \sin w \quad 0] \\ = abc u^2 \cos^2 w + abc u^2 \sin^2 w = abc u^2.$$

Таким образом, используя уравнение (6-48), получим, что в любой точке поверхности

$$K = \frac{AC - B^2}{|Q_u \times Q_w|^4} = \frac{(0)(abcu^2) - (0)}{|Q_u \times Q_w|^4} = 0$$

и, следовательно, поверхность является развертывающейся. Заметим, что хотя $|Q_u \times Q_w|^2 = 0$ для $u = 0$, из правила Лопитала получаем, что $K = 0/0 = 0$ в $u = 0$.

6-9 ЛИНЕЙНАЯ ПОВЕРХНОСТЬ КУНСА

Если известны четыре граничные кривые $P(u, 0)$, $P(u, 1)$, $P(0, w)$ и $P(1, w)$ и для внутренней части куска поверхности используется билинейная смещающая функция, то в результате получаем линейную поверхность Кунса. На первый взгляд можно предположить, что желаемый результат дает простая сумма отдельных линейчатых поверхностей (уравнения (6-43) и (6-44)) в двух направлениях u , w :

$$Q(u, w) = P(u, 0)(1 - w) + P(u, 1)w + P(0, w)(1 - u) + P(1, w)u.$$

Однако, проверив этот результат в угловых точках куска поверхности, например

$$Q(0, 0) = P(0, 0) + P(0, 0) = 2P(0, 0)$$

и на границах, например

$$Q(0, w) = P(0, 0)(1 - w) + P(0, 1)w + P(0, w)$$

получим, что ни одно из этих значений не соответствует исходным данным. Это происходит из-за того, что угловые точки учитываются дважды, так как $P(0, 0)$ содержится в обеих граничных кривых $P(u, 0)$ и $P(0, w)$.

Правильный результат можно получить с помощью вычитания дополнительных членов, возникающих из-за удвоения угловых точек:

$$\begin{aligned} Q(u, w) = & P(u, 0)(1 - w) + P(u, 1)w + P(0, w)(1 - u) + P(1, w)u \\ & - P(0, 0)(1 - u)(1 - w) - P(0, 1)(1 - u)w. \\ & - P(1, 0)u(1 - w) - P(1, 1)uw. \end{aligned} \quad (6-49)$$

Теперь в угловых точках

$$Q(0, 0) = P(0, 0) \quad \text{и т. д.}$$

и вдоль границ

$$Q(0, w) = P(0, w)$$

$$Q(u, 1) = P(u, 1)$$

и т. д.

В матричной форме уравнение (6-49) имеет вид:

$$\begin{aligned} Q(u, w) = & [1 - u \quad w] \begin{bmatrix} P(0, w) \\ P(1, w) \end{bmatrix} + [P(u, 0) \quad P(u, 1)] \begin{bmatrix} 1 - w \\ w \end{bmatrix} \\ & - [1 - u \quad w] \begin{bmatrix} P(0, 0) & P(0, 1) \\ P(1, 0) & P(1, 1) \end{bmatrix} \begin{bmatrix} 1 - w \\ w \end{bmatrix} \end{aligned}$$

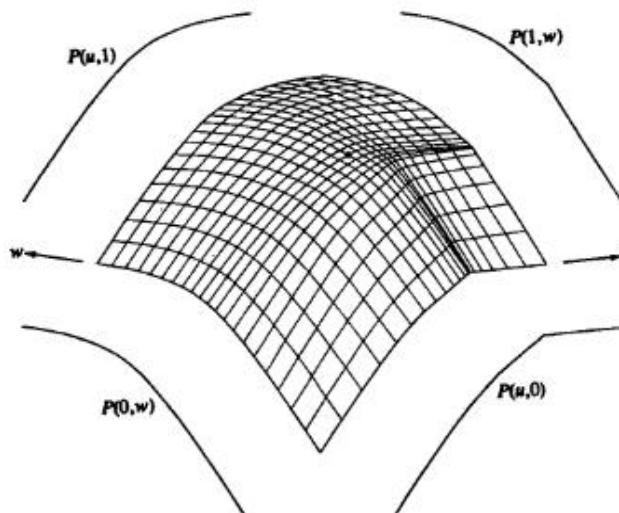


Рис. 6-30 Линейная поверхность Кунса.

или более компактно

$$Q(u, w) = [1 - u \quad u \quad 1] \begin{bmatrix} -P(0,0) & -P(0,1) & P(0,w) \\ -P(1,0) & -P(1,1) & P(1,w) \\ P(u,0) & P(u,1) & 0 \end{bmatrix} \begin{bmatrix} 1-w \\ w \\ 1 \end{bmatrix}. \quad (6-50)$$

Функции $(1-u)$, u , $(1-w)$ и w называются функциями смешения потому, что они смешивают граничные кривые для получения внутренней формы поверхности. Линейная поверхность показана на рис. 6-30. Линейная поверхность Кунса является самой простой из поверхностей Кунса. В разд. 6-10 обсуждается более общая поверхность Кунса. Следующий пример иллюстрирует использованный выше метод.

Пример 6-12 Линейная поверхность Кунса

Найти точку с координатами $u = w = 0.5$, расположенную на линейной поверхности Кунса, если четыре граничные кривые $P(u,0)$, $P(u,1)$, $P(0,w)$, $P(1,w)$ задаются незамкнутыми B-сплайнами третьего порядка ($k = 3$)

$$P(u,0) : B_1[0 \quad 0 \quad 0], B_2[1 \quad 1 \quad 0], B_3[2 \quad 1 \quad 0], B_4[3 \quad 0 \quad 0],$$

$$B_5[3 \quad 0 \quad 3]$$

$$P(u,1) : \bar{B}_1[0 \quad 0 \quad 3], \bar{B}_2[1 \quad 1 \quad 3], \bar{B}_3[2 \quad 1 \quad 3], \bar{B}_4[2 \quad 1 \quad 3]$$

$$P(0,w) : \bar{C}_1[0 \quad 0 \quad 3], \bar{C}_2[0 \quad 1 \quad 2], \bar{C}_3[0 \quad 1 \quad 1], \bar{C}_4[0 \quad 0 \quad 0]$$

$$P(1,w) : C_1[3 \quad 0 \quad 3], C_2[3 \quad 1 \quad 2], C_3[3 \quad 1 \quad 2], C_4[3 \quad 1 \quad 1],$$

$$C_5[3 \quad 0 \quad 0]$$

Вспоминая предыдущее обсуждение B-сплайнов (см. разд. 5-9), получим узловой вектор для $P(u,0)$ и $P(1,w)$

$$[X] = [0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 3 \quad 3]$$

Таким образом, ненормализованный диапазон параметра есть $0 \leq t \leq 3$. Для $(u, 1)$ и $P(0, w)$ узловый вектор с ненормализованным диапазоном параметра $0 \leq s \leq 2$ имеет вид

$$[Y] = [0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 2 \quad 2].$$

Соответствующими значениями для параметров в нормализованном диапазоне $0 \leq u, w \leq 1$ являются $t = 1,5$ и $s = 1,0$.

Воспользовавшись уравнениями (5-83) и (5-84), получим

$$P(u, 0) = P(t) = B_1 N_{1,3}(t) + B_2 N_{2,3}(t) + B_3 N_{3,3}(t) + B_4 N_{4,3}(t) + B_5 N_{5,3}(t)$$

$$\begin{aligned} P(0.5, 0) = P(1.5) &= (0)B_1 + 0.125B_2 + 0.75B_3 + 0.125B_4 + (0)B_5 \\ &= 0.125[1 \quad 1 \quad 3] + 0.75[2 \quad 1 \quad 3] + 0.125[2 \quad 1 \quad 3] \\ &= [1.875 \quad 1 \quad 3] \end{aligned}$$

$$P(u, 1) = P(s) = \bar{B}_1 \bar{N}_{1,3}(s) + \bar{B}_2 \bar{N}_{2,3}(s) + \bar{B}_3 \bar{N}_{3,3}(s) + \bar{B}_4 \bar{N}_{4,3}(s)$$

$$\begin{aligned} P(0.5, 1) = P(1.0) &= (0)\bar{B}_1 + 0.5\bar{B}_2 + 0.5\bar{B}_3 + (0)\bar{B}_4 \\ &= 0.5[1 \quad 1 \quad 0] + 0.5[2 \quad 1 \quad 0] \\ &= [1.5 \quad 1 \quad 0] \end{aligned}$$

$$P(0, w) = P(s) = \bar{C}_1 \bar{N}_{1,3}(s) + \bar{C}_2 \bar{N}_{2,3}(s) + \bar{C}_3 \bar{N}_{3,3}(s) + \bar{C}_4 \bar{N}_{4,3}(s)$$

$$\begin{aligned} P(0, 0.5) = P(1.0) &= (0)\bar{C}_1 + 0.5\bar{C}_2 + 0.5\bar{C}_3 + (0)\bar{C}_4 \\ &= 0.5[0 \quad 1 \quad 2] + 0.5[0 \quad 1 \quad 1] \\ &= [0 \quad 1 \quad 1.5] \end{aligned}$$

$$P(1, w) = P(t) = C_1 N_{1,3}(t) + C_2 N_{2,3}(t) + C_3 N_{3,3}(t) + C_4 N_{4,3}(t) + C_5 N_{5,3}(t)$$

$$\begin{aligned} P(1, 0.5) = P(1.5) &= (0)C_1 + 0.125C_2 + 0.75C_3 + 0.125C_4 + (0)C_5 \\ &= 0.125[3 \quad 1 \quad 2] + 0.75[3 \quad 1 \quad 2] + 0.125[3 \quad 1 \quad 1] \\ &= [3 \quad 1 \quad 1.875] \end{aligned}$$

Теперь, используя уравнение (6-50), имеем

$$Q(u, w) = [1 - u \quad u \quad 1] \begin{bmatrix} -P(0,0) & -P(0,1) & P(0,w) \\ -P(1,0) & -P(1,1) & P(1,w) \\ P(u,0) & P(u,1) & 0 \end{bmatrix} \begin{bmatrix} 1-w \\ w \\ 1 \end{bmatrix},$$

$$\begin{aligned} Q(0.5, 0.5) &= [0.5 \quad 0.5 \quad 1] \times \\ &\times \begin{bmatrix} -[0 \quad 0 \quad 3] & -[0 \quad 0 \quad 0] & [0 \quad 1 \quad 1.5] \\ -[3 \quad 0 \quad 3] & -[3 \quad 0 \quad 0] & [3 \quad 1 \quad 1.875] \\ [1.875 \quad 1 \quad 3] & [1.5 \quad 1 \quad 0] & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \\ 1 \end{bmatrix} = \\ &= [1.6375 \quad 2.25 \quad 1.6375]. \end{aligned}$$

Полученные результаты изображены на рис. 6-30. Отметим наличие плоского участка на поверхности, гауссова кривизна в этой области равна нулю. Следовательно, эта часть поверхности развертывающаяся. В остальной области гауссова кривизна положительна и поверхность не является развертывающейся.

6-10 БИКУБИЧЕСКАЯ ПОВЕРХНОСТЬ КУНСА

Хотя аналитические поверхности, например квадратичные, линейчатые и простые куски линейно интерполированных поверхностей, обсуждавшиеся в предыдущих разделах, и важны для конструирования и производства, но для многих приложений они не обладают достаточной гибкостью. Описание поверхностей, обсуждаемое в этом и дальнейших разделах главы, предоставляют необходимую гибкость с помощью использования полиномов более высоких степеней как для граничных кривых куска поверхности, так и для внутренних смешивающих функций. Поверхности, сгенерированные с помощью объединения таких кусков, называются скульптурными поверхностями. Для описания скульптурных поверхностей фундаментальное значение имеет кусок бикубической поверхности Кунса (см. [6-19]).

Для всех четырех граничных кривых куска бикубической поверхности Кунса используются нормализованные кубические сплайны (см. разд. 5-4). Для задания внутренней части куска используются кубические смешивающие функции. Таким образом, каждую граничную кривую можно представить в общем виде

$$P(t) = B_1 + B_2t + B_3t^2 + B_4t^3, \quad 0 \leq t \leq 1. \quad (5-2)$$

Для одного сегмента нормализованного кубического сплайна с известными касательными и координатными векторами на концах, каждая из четырех граничных кривых, $P(u, 0)$, $P(u, 1)$, $P(0, w)$ и $P(1, w)$ задается следующим образом (см.

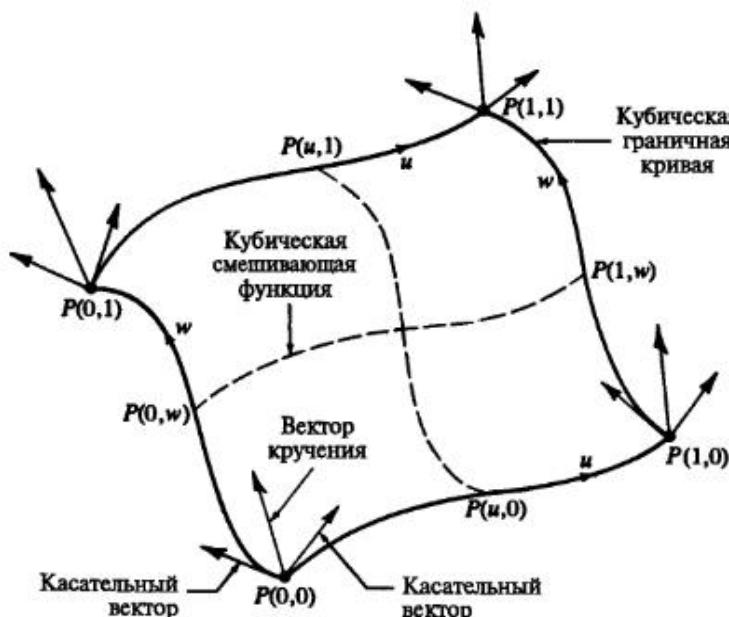


Рис. 6-31 Геометрия для куска бикубической поверхности Кунса.

уравнения (5-24), (5-26), (5-27):

$$P(t) = [T][N][G] = \\ = [t^3 \quad t^2 \quad t \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P'_1 \\ P'_2 \end{bmatrix} \quad 0 \leq t \leq 1, \quad (5-27)$$

где t становится соответственно u или w , а P_1, P_2, P'_1, P'_2 — координатные и касательные векторы на концах соответствующей граничной кривой (см. рис. 6-31). Кубическая смешивающая функция, использованная для обоих параметрических направлений, идентична функции, использовавшейся для интерполяции внутренней части нормализованного кубического сплайна, т. е.

$$[F] = [F_1(t) \quad F_2(t) \quad F_3(t) \quad F_4(t)] = [T][N] \quad (5-26) \\ = [t^3 \quad t^2 \quad t \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Расписывая смешивающие функции, имеем

$$F_1(t) = 2t^3 - 3t^2 + 1, \quad (5-25a)$$

$$F_2(t) = -2t^3 + 3t^2, \quad (5-25b)$$

$$F_3(t) = t^3 - 2t^2 + t, \quad (5-25c)$$

$$F_4(t) = t^3 - t^2, \quad (5-25d)$$

где t есть либо u либо w , соответственно. Эти функции изображены на рис. 5-8. Тогда определение куска бикубической поверхности Кунса записывается в виде

$$Q(u, w) = [F_1(u) \quad F_2(u) \quad F_3(u) \quad F_4(u)] \times \\ \begin{bmatrix} P(0,0) & P(0,1) & P_w(0,0) & P_w(0,1) \\ P(1,0) & P(1,1) & P_w(1,0) & P_w(1,1) \\ P_u(0,0) & P_u(0,1) & P_{uw}(0,0) & P_{uw}(0,1) \\ P_u(1,0) & P_u(1,1) & P_{uw}(1,0) & P_{uw}(1,1) \end{bmatrix} \begin{bmatrix} F_1(w) \\ F_2(w) \\ F_3(w) \\ F_4(w) \end{bmatrix} \quad (6-51)$$

для $0 \leq u \leq 1$ и $0 \leq w \leq 1$. С помощью уравнения (5-26) это можно записать более компактно:

$$Q(u, w) = [U][N][P][N]^T[W], \quad (6-52)$$

где $[U] = [u^3 \quad u^2 \quad u \quad 1]$ и $[W]^T = [w^3 \quad w^2 \quad w \quad 1]$.

Матрица $[P]$ в уравнении (6-51) содержит всю геометрическую информацию, необходимую для генерации одного бикубического куска поверхности Кунса. Подматрицы размера 2×2 в каждом углу $[P]$ содержат

$$[P] = \begin{bmatrix} \text{угловые коор-} & : & w\text{-каса-} \\ \text{динатные} & : & \text{тельные} \\ \text{векторы} & : & \text{векторы} \\ \dots & \dots & \dots \\ u\text{-каса-} & : & \text{векторы} \\ \text{тельные} & : & \text{кручения} \\ \text{векторы} & : & \end{bmatrix}$$

Таким образом, кусок бикубической поверхности определяется четырьмя координатными векторами в угловых точках, восемью касательными векторами (по два в каждом углу), четырьмя векторами кручения в углах и четырьмя кубическими смещающими функциями F_1, F_2, F_3 и F_4 .

Пример куска бикубической поверхности представлен на рис. 6-32. Заметим, что каждая из изображенных на рис. 6-32 изопараметрических линий является нормализованным кубическим сплайном. Несколько кусков могут быть соединены вдоль C^1 -гладких границ для образования более сложных поверхностей, являющихся C^1 -гладкими везде. Детально условия объединения кусков бикубических поверхностей обсуждаются в [6-19].

Параметрические производные в любой точке на бикубической поверхности получаются с помощью формального дифференцирования уравнения (6-52):

$$Q_u(u, w) = [U][N][P][N]^T[W], \quad (6-53)$$

$$Q_w(u, w) = [U][N][P][N]^T[W'], \quad (6-54)$$

$$Q_{uw}(u, w) = [U'][N][P][N]^T[W'], \quad (6-55)$$

$$Q_{uu}(u, w) = [U''][N][P][N]^T[W], \quad (6-56)$$

$$Q_{ww}(u, w) = [U][N][P][N]^T[W''], \quad (6-57)$$

где штрихи обозначают дифференцирование по соответствующей переменной и

$$[U'] = [3u^2 \quad 2u \quad 1 \quad 0],$$

$$[W']^T = [3w^2 \quad 2w \quad 1 \quad 0],$$

$$[U''] = [6u \quad 2 \quad 0 \quad 0],$$

$$[W'']^T = [6w \quad 2 \quad 0 \quad 0].$$

Нормаль к поверхности, используемая для определения невидимых поверхностей, вычисления освещенности (см. [1-11]) и создания управляющих программ для станков, задается выражением

$$\mathbf{n} = Q_u \times Q_w.$$

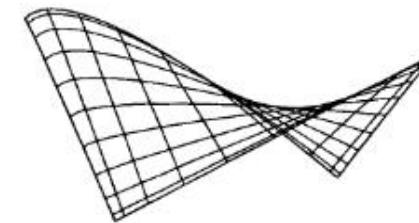


Рис. 6-32 Кусок бикубической поверхности Кунса.

Подробный пример иллюстрирует метод вычислений для бикубических кусков.

Пример 6-13 Кусок бикубической поверхности Кунса

Определить точку на куске бикубической поверхности Кунса, соответствующую параметрам $u = w = 0.5$. Вектор-точки для четырех углов равны $P(0,0) = [-100 \quad 0 \quad 100]$, $P(1,0) = [-100 \quad -100 \quad -100]$, $P(1,1) = [100 \quad -100 \quad -100]$, $P(0,1) = [100 \quad 100 \quad 0]$. Касательные векторы равны $P_u(0,0) = [100 \quad 100 \quad 0]$, $P_u(0,1) = [1 \quad 1 \quad 0]$, $P_u(1,0) = [1 \quad -1 \quad 0]$, $P_u(1,1) = [1 \quad -1 \quad 0]$, $P_w(0,0) = [0 \quad 10 \quad -10]$, $P_w(0,1) = [0 \quad -1 \quad -1]$, $P_w(1,0) = [0 \quad 1 \quad -1]$, $P_w(1,1) = [0 \quad -1 \quad -1]$. Векторы кручения равны $P_{uw}(0,0) = [0 \quad 0 \quad 0]$, $P_{uw}(0,1) = [0.1 \quad 0.1 \quad 0.1]$, $P_{uw}(1,0) = [0.1 \quad -0.1 \quad -0.1]$, $P_{uw}(1,1) = [0 \quad 0 \quad 0]$. Определить параметрические производные Q_u , Q_w и вектор нормали для $u = w = 0.5$.

Используя матричную запись из уравнения (6-51), получим

$$Q(u, w) = [U][N] \begin{bmatrix} [-100 \quad 0 \quad 100] & [-100 \quad -100 \quad -100] \\ [100 \quad -100 \quad 100] & [100 \quad 0 \quad -100] \\ [100 \quad 100 \quad 0] & [1 \quad 1 \quad 0] \\ [1 \quad -1 \quad 0] & [1 \quad -1 \quad 0] \\ [0 \quad 10 \quad -10] & [0 \quad -1 \quad -1] \\ [0 \quad 1 \quad -1] & [0 \quad -1 \quad -1] \\ [0 \quad 0 \quad 0] & [0.1 \quad 0.1 \quad 0.1] \\ [0.1 \quad -0.1 \quad -0.1] & [0 \quad 0 \quad 0] \end{bmatrix} [N]^T [W]^T$$

Для $u = 0.5$ смещающие функции по u имеют вид:

$$[U][N] = [u^3 \quad u^2 \quad u \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$= [0.125 \quad 0.25 \quad 0.5 \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = [0.5 \quad 0.5 \quad 0.125 \quad -0.125]$$

Для $w = 0.5$ смешивающие функции по w :

$$\begin{aligned}[N]^T[W] &= \begin{bmatrix} 2 & -3 & 0 & 1 \\ -2 & 3 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 2 & -3 & 0 & 1 \\ -2 & 3 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.125 \\ 0.25 \\ 0.5 \\ 1 \end{bmatrix} \\ &= [0.5 \ 0.5 \ 0.125 \ -0.125]^T.\end{aligned}$$

Таким образом, x компонента $Q(u, w)$ при $u = w = 0.5$ равна

$$\begin{aligned}Q_x(0.5, 0.5) &= [0.5 \ 0.5 \ 0.125 \ -0.125] \times \\&\quad \times \begin{bmatrix} -100 & -100 & 0 & 0 \\ 100 & 100 & 0 & 0 \\ 100 & 1 & 0 & 0.1 \\ 1 & 1 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= [12.375 \ 0 \ -0.0125 \ 0.0125] \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= 6.18\end{aligned}$$

Аналогично y и z компоненты таковы:

$$\begin{aligned}Q_y(0.5, 0.5) &= [0.5 \ 0.5 \ 0.125 \ -0.125] \times \\&\quad \times \begin{bmatrix} 0 & -100 & 10 & -1 \\ -100 & 0 & 1 & -1 \\ 100 & 1 & 0 & 0.1 \\ -1 & -1 & -0.1 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= -42.75\end{aligned}$$

$$\begin{aligned}Q_z(0.5, 0.5) &= [0.5 \ 0.5 \ 0.125 \ -0.125] \times \\&\quad \times \begin{bmatrix} 100 & -100 & -10 & -1 \\ 100 & -100 & -1 & -1 \\ 0 & 0 & 0 & 0.1 \\ 0 & 0 & -0.1 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= -0.56\end{aligned}$$

Следовательно, $Q(0.5, 0.5) = [6.18 \ -42.75 \ -0.56]$. Вся поверхность изображена на рис. 6-32.

Обратимся теперь к производным, вспомним уравнения (6-53)–(6-55) и заметим, что

$$\begin{aligned}[U'][N] &= [3u^2 \ 2u \ 1 \ 0][N] = \\&= [0.75 \ 1 \ 1 \ 0] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \\&= [-1.5 \ 1.5 \ -0.25 \ -0.25]\end{aligned}$$

$$\begin{aligned}[N]^T[W'] &= [N]^T[3w^2 \ 2w \ 1 \ 0]^T = \\&= \begin{bmatrix} 2 & -3 & 0 & 1 \\ -2 & 3 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.75 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \\&= [-1.5 \ 1.5 \ -0.25 \ -0.25]^T\end{aligned}$$

параметрические производные в $u = w = 0.5$ равны

$$\begin{aligned}Q_{uu}(0.5, 0.5) &= [-1.5 \ 1.5 \ -0.25 \ -0.25] \times \\&\quad \times \begin{bmatrix} -100 & -100 & 0 & 0 \\ 100 & 100 & 0 & 0 \\ 100 & 1 & 0 & 0.1 \\ 1 & 1 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= [274.75 \ 299.5 \ -0.025 \ -0.025] \begin{bmatrix} 0.5 \\ 0.5 \\ 0.125 \\ -0.125 \end{bmatrix} = \\&= 287.13\end{aligned}$$

Q_{uv} и Q_{uw} находятся аналогичным образом. В результате получим

$$Q_u(0.5, 0.5) = [287.13 \ -14.06 \ 1.69].$$

Теперь

$$\begin{aligned}Q_{ww}(0.5, 0.5) &= [0.5 \ 0.5 \ 0.125 \ -0.125] \times \\&\quad \times \begin{bmatrix} -100 & -100 & 0 & 0 \\ 100 & 100 & 0 & 0 \\ 100 & 1 & 0 & 0.1 \\ 1 & 1 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} -1.5 \\ 1.5 \\ -0.25 \\ -0.25 \end{bmatrix} = \\&= [12.38 \ 0 \ -0.0125 \ 0.0125] \begin{bmatrix} -1.5 \\ 1.5 \\ -0.25 \\ -0.25 \end{bmatrix} = \\&= -18.56\end{aligned}$$

Q_{uw} и Q_{ww} находятся аналогичным образом, откуда

$$Q_w(0.5, 0.5) = [-18.56 \ -19.69 \ -298.38]$$

И наконец, вектор нормали имеет вид:

$$\begin{aligned}\mathbf{n} &= Q_u \times Q_w = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ Q_{uu} & Q_{uy} & Q_{uz} \\ Q_{wu} & Q_{wy} & Q_{wz} \end{vmatrix} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 287.13 & -14.06 & 1.69 \\ -18.56 & -19.69 & -298.38 \end{vmatrix} \\ &= 4227.5\mathbf{i} + 85641.2\mathbf{j} - 5915.5\mathbf{k}.\end{aligned}$$

Единичная нормаль равна

$$\hat{\mathbf{n}} = [0.049 \ 0.996 \ -0.069]$$

Заметим, что векторы кручения в угловых точках слегка смешили нормаль от направления y .

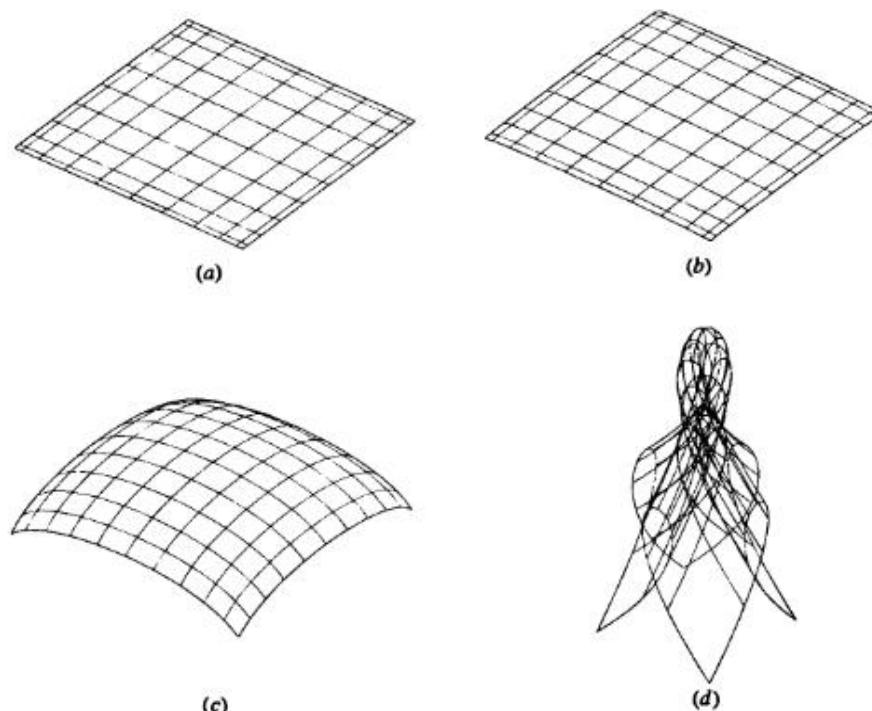


Рис. 6-33 Действие на кусок бикубической поверхности варьирования величины касательного вектора. Значения компонент касательного вектора равны (a) 1; (b) 10; (c) 100; (d) 1000.

Одна из проблем, возникающих при использовании бикубических кусков поверхности, заключается в несоответствии значений векторов положения, кручения и касательных векторов. Все они в общем случае имеют различные порядки величин. Например, для куска поверхности со стороной 10 единиц, касательные векторы могут иметь значения порядка 10 единиц векторы кручения — 100 единиц и векторы положения — 1000 единиц. Кроме того, не всегда очевиден результат изменения величины и направления касательных векторов и/или векторов кручения.

На рис. 6-33 показан результат варьирования величин касательных векторов в угловых точках куска бикубической поверхности. Векторы положения углов куска лежат в плоскости $y = 0$, отцентрированные относительно начала координат в $\pm 100x, z$. Касательные векторы лежат в плоскостях, расположенных перпендикулярно плоскости угловых точек и содержащих смежные углы куска поверхности. Направление касательных векторов выбрано таким образом, чтобы сделать кубическую граничную кривую выпуклой вверх («горбом вверх»), например $P_u(0, 0) = [1 \ 1 \ 0]$, $P_u(1, 0) = [1 \ -1 \ 0]$, и т. д. Величина векторов кручения равна нулю. Если значение компоненты у каждого касательного вектора равно нулю, то кусок поверхности плоский; если не равно нулю, то кусок изогнут.

На рис. 6-33 a изображен кусок бикубической поверхности с единичными значениями компонент касательного вектора. Хотя поверхность выглядит плоской, на самом деле она слегка округлена. Компонента y в центре поверхности, т. е. в $x = z = 0$ равна 0.5. На рисунках 6-33 $b-d$ показаны бикубические поверхности, для которых значения компонент касательного вектора увеличены соответственно на 1, 2 или 3 порядка величины, т. е. равны 10, 100 и 1000. Значения компоненты y в центральной точке теперь равны 5, 50 и 500. Заметим, что они также увеличиваются на порядок величины всякий раз, когда увеличивается значение касательного вектора.

Особый интерес представляет рис. 6-33 d . Изображенная на нем поверхность является самопересекающейся. Чтобы лучше понять этот результат, можно вспомнить обсуждение кубических сплайнов кривых в разд. 5-3 и в особенности рис. 5-10. Кроме того, заметим, что каждая изопараметрическая линия на поверхности является кубической сплайновой кривой. Каждая из изопараметрических линий образует петлю, показывая, что значение касательного вектора в угловой точке превысило критическое значение.

Результат изменения направления касательного вектора в угловой точке показан на рис. 6-34. Здесь изменены знаки компонент у касательных векторов $P(0, 1)$; их значения стали равны 100. Заметим, что на рис. 6-34 правый передний угол поверхности теперь вогнутый, а не выпуклый, как на рис. 6-33 c .

Кручение было ранее упомянуто в разд. 6-5 при обсуждении куска сферической поверхности. Там оно определялось как смешанная производная поверхности в угловых точках. В действительности вектор кручения в угловой точке позволяет модифицировать внутреннюю форму поверхности без модификации касательных векторов в этой точке. На рис. 6-35 показан результат модификации вектора кручения в одной угловой точке $P(0, 0)$ на поверхности. Для каждой из поверхностей на рис. 6-35 $P(0, 0)$ является нижней угловой точкой. Поверхность аналогична представленной на рис. 6-33 b , за исключением ненулевого вектора кручения в $P(0, 0)$. На рисунках 6-35 a и b со значениями компонент вектора кручения 10 и 100, соответственно, показан небольшой видимый эффект ненулевого вектора кручения. На рис. 6-35 c поверхность имеет выраженную выпуклую форму благодаря ненулевому вектору кручения. Рис. 6-35 d иллюстрирует действие больших значений этого вектора. Заметим, однако, что в отличие от рис. 6-33 d поверх-

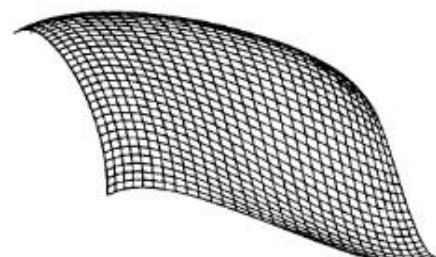


Рис. 6-34 Действие на кусок бикубической поверхности изменения направления касательного вектора.

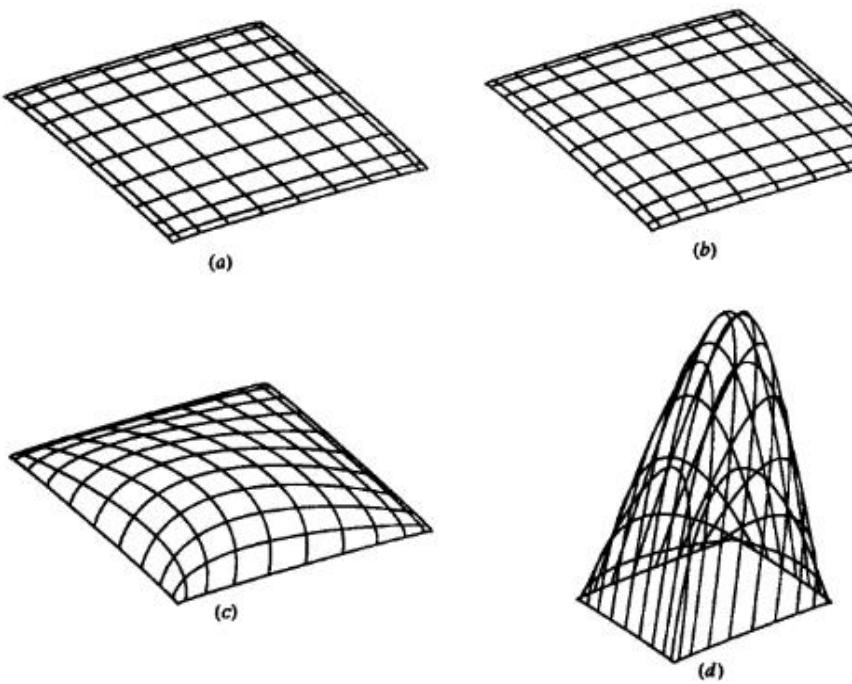


Рис. 6-35 Влияние вектора кручения на кусок бикубической поверхности. Значение компоненты вектора кручения в точке $P(0,0)$ равно (a) 10; (b) 100; (c) 1000; (d) 10000.

ность не самопересекается. Дополнительное тщательное исследование рис. 6-35d показывает, что поверхность слегка вогнута в трех угловых точках с нулевым кручением.

Поверхность с ненулевым кручением в угловой точке является в ней локально «плоской». Поверхность с нулевым кручением во всех четырех углах, как это показано на рис. 6-33, называется куском поверхности Фергюсона или F куском [6-20]. При совместном объединении для формирования более сложных поверхностей местоположение угловых точек кусков вполне очевидно. По этой причине такие куски поверхностей редко используются на практике.

Хотя приведенное здесь обсуждение ограничивается четырехсторонними кусками поверхностей, и другие формы кусков, как это показано на рис. 6-36, тоже имеют значение. Треугольные куски поверхностей рассмотрены в [6-21] и [6-22]. Пятиугольные куски поверхностей обсуждаются в [6-23].

6-11 ПОВЕРХНОСТИ БЕЗЬЕ

Бикубические поверхности Кунса предоставляют гибкое и мощное средство разработки поверхностей. Однако их практическое использование, как и для кубических сплайновых кривых, затрудняется необходимостью задания точной, интуитивно неочевидной математической информации, например координат точек,

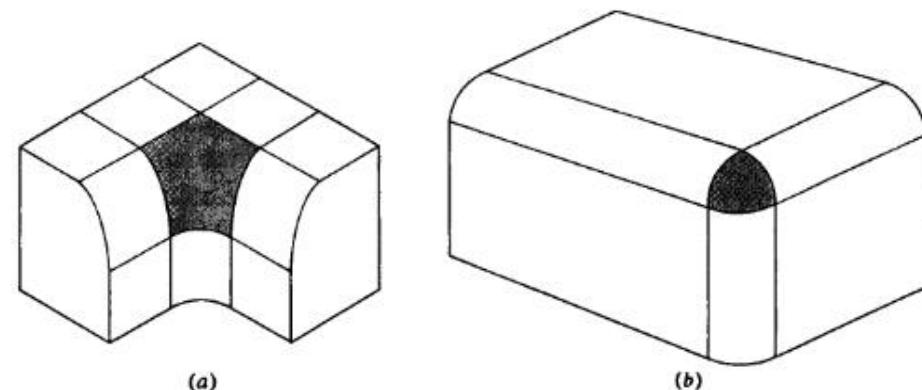


Рис. 6-36 Нечетырехугольные куски. (a) Пятиугольный; (b) треугольный.

касательных векторов и векторов кручения. Возникающие проблемы иллюстрируются на рисунках 6-33 — 6-35. Большинство из этих проблем можно преодолеть, распространив понятия кривых Безье на поверхности.

Декартово или тензорное произведение поверхности Безье задается в виде

$$Q(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J_{n,i}(u) K_{m,j}(w), \quad (6-58)$$

где $J_{n,i}(u)$ и $K_{m,j}(w)$ есть базисные функции Бернштейна в параметрических направлениях u и w (см. уравнения 5-63 и 5-64). Для удобства повторим здесь определение, приведенное ранее в разд. 5-8

$$J_{n,i}(u) = \binom{n}{i} u^i (1-u)^{n-i}, \quad (5-63)$$

$$K_{m,j}(w) = \binom{m}{j} w^j (1-w)^{m-j}$$

с

$$\binom{n}{j} = \frac{n!}{i!(n-i)!}, \quad (6-64)$$

$$\binom{m}{j} = \frac{m!}{j!(m-j)!}$$

Здесь элементы $B_{i,j}$ являются вершинами задающей полигональной сетки, как это показано на рис. 6-37. Индексы n и m на единицу меньше числа вершин многогранника в направлениях u и w , соответственно. Для четырехсторонних кусков поверхностей задающая полигональная сетка должна быть топологически прямоугольной, т. е. должна иметь одинаковое количество вершин в каждом «ряду».

Снова, как и для кривых Безье, из-за того что для смешивающих функций используется базис Бернштейна, многие свойства поверхности известны. Например:

Степень поверхности в каждом параметрическом направлении на единицу меньше числа вершин задающего многогранника в этом направлении.

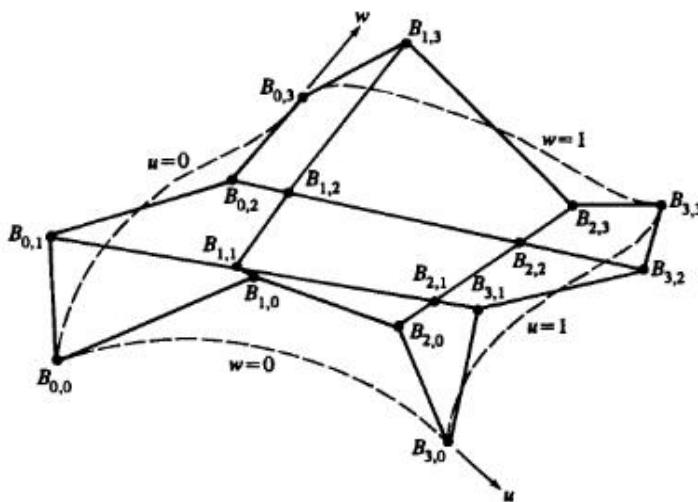
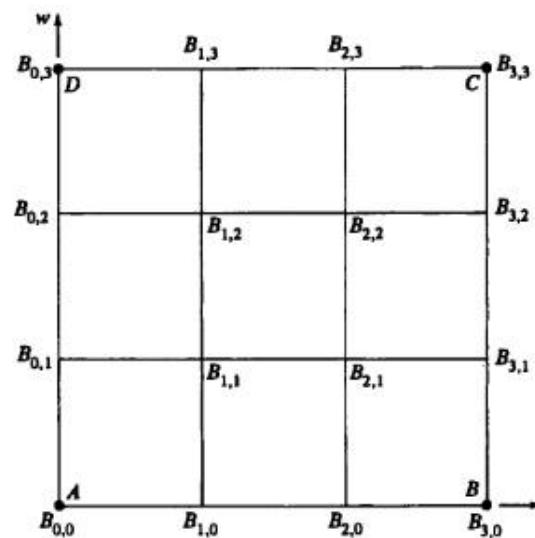


Рис. 6-37 Поверхность Безье и вершины характеристического многогранника.

Рис. 6-38 Схема задающей полигональной сетки 4×4 для поверхности Безье.

Гладкость поверхности в каждом параметрическом направлении на две единицы меньше числа вершин задающего многогранника в этом направлении. Поверхность отображает в общем виде форму задающей полигональной сетки. Совпадают только угловые точки задающей полигональной сетки и поверхности.

Поверхность содержится внутри выпуклой оболочки задающей полигональной сетки.

Поверхность не проявляет свойства затухания изменений. Это свойство не определено и неизвестно для поверхностей двух переменных.

Поверхность инвариантна относительно аффинного преобразования.

Каждая из граничных кривых поверхности Безье является кривой Безье. Запомним этот факт и рассмотрим задающую полигональную сетку для бикубической поверхности Безье размера 4×4 , изображенную схематично на рис. 6-38. Легко видеть, что направление и величина касательных векторов в угловых точках куска управляются положением соседних точек вдоль сторон сетки. А именно касательные векторы в направлениях u , w в точке A управляются вершинами полигональной сетки $B_{0,1}$ и $B_{1,0}$, соответственно. Аналогичным образом, вершины полигональной сетки $B_{2,0}$, $B_{3,1}$, $B_{3,2}$, $B_{2,3}$ и $B_{1,3}$, $B_{0,2}$ управляют касательными векторами в угловых точках B , C , D , соответственно. Четыре внутренние вершины полигональной сетки, $B_{1,1}$, $B_{2,1}$, $B_{2,2}$ и $B_{1,2}$ влияют на направление и величину векторов кручения в угловых точках A , B , C , D куска поверхности. Следовательно, пользователь может управлять формой куска поверхности, не зная конкретных значений касательных векторов и векторов кручения.

На рис. 6-39 показано несколько бикубических поверхностей Безье и их задающих полигональных сеток. Базовая полигональная сетка имеет размер 4×4 и отцентрирована относительно начала координат с угловыми точками, находящимися в ± 15 по x , z . Компонента y угловых вершин равна нулю. У всех других вершин эта компонента равна пяти. Базовая полигональная сетка и соответствующая ей поверхность Безье изображены на рис. 6-39a. На рис. 6-39 точка $B_{0,0}$ является левой угловой вершиной, а $B_{3,3}$ — правой угловой вершиной. Заметим, что центральные вершины базовой полигональной сетки образуют плоский крест (показанный затененным). Следовательно, центр получившейся поверхности minimally изогнут, хотя и не плоский.

На рис. 6-39b проиллюстрирован эффект увеличения в 2 раза величины касательного вектора в точке $B_{0,0}$ в обоих параметрических направлениях u и w с помощью перемещения точек $B_{1,0}$ и $B_{0,1}$. Вектор кручения не меняется. Отметим увеличение кривизны граничных кривых, отвечающих значениям параметров $u = 0$ и $w = 0$, и соответствующее изменение внутренности поверхности.

На рис. 6-39c показано действие изменения направления касательных векторов в точке $B_{0,3}$ в обоих параметрических направлениях u и w с помощью перемещения точек $B_{0,2}$ и $B_{1,3}$. Отметим изменение знака кривизны граничной кривой около точки $B_{0,3}$ и формы внутренней части поверхности по сравнению с базовой поверхностью.

На рис. 6-39d проиллюстрирован результат удвоения величины вектора кручения в точке $B_{0,0}$ без изменения его направления. В этом случае перемещается только точка $B_{1,1}$. Эффект этого изменения неуловим, но тем не менее важен для конструирования. Внимательное сравнение с базовой поверхностью на рис. 6-39a показывает, что параметрические линии вблизи точки $B_{0,0}$ имеют большую кривизну. Этот эффект распространяется приблизительно до центра поверхности.

В матричном виде декартово произведение поверхности Безье задается выражением

$$Q(u, w) = [U][N][B][M]^T[W], \quad (6-59)$$

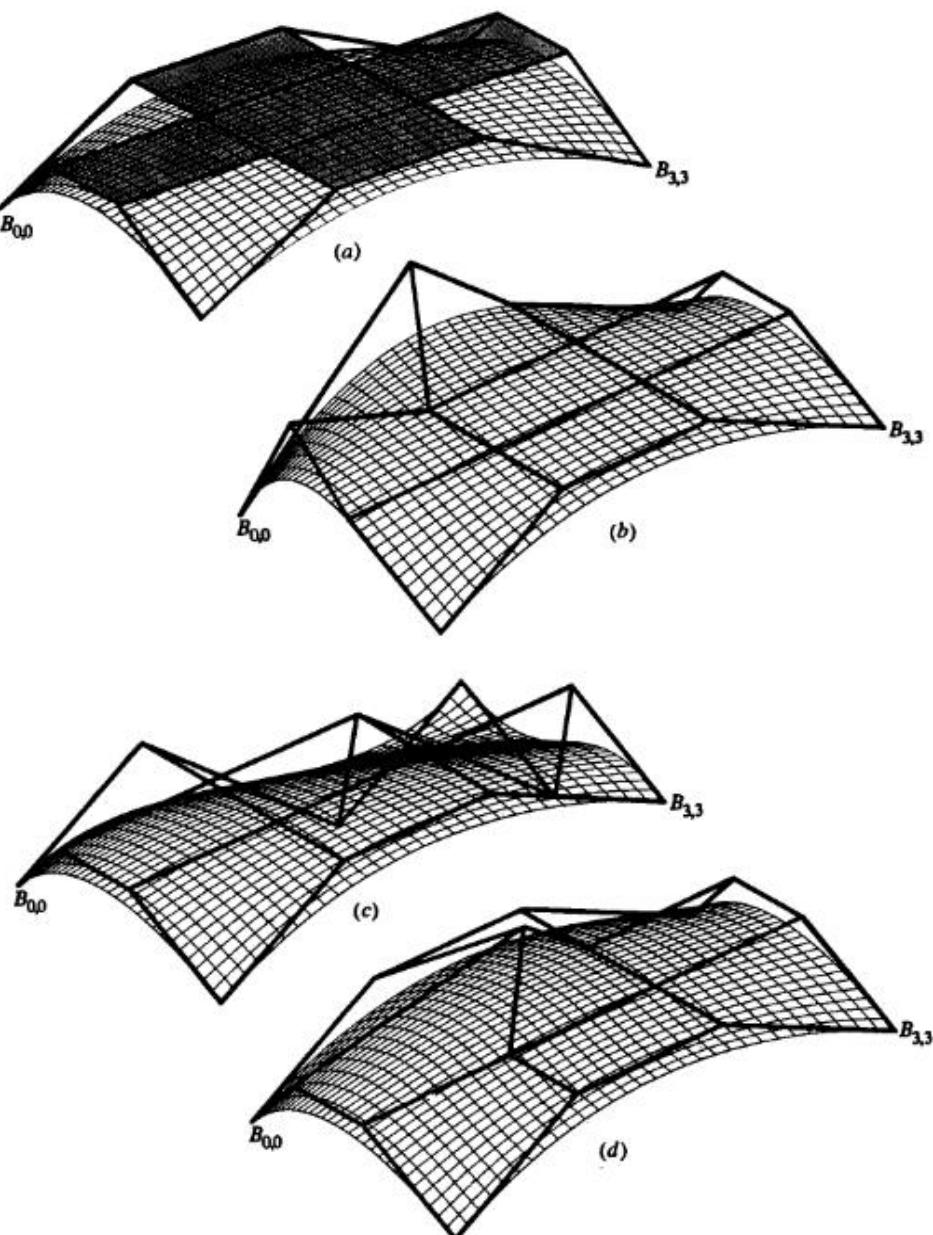


Рис. 6-39 Бикубические поверхности Безье. (a) Основная поверхность; (b) эффект изменения величины обоих касательных векторов в $B_{0,0}$; (c) эффект изменения направления касательного вектора в $B_{0,3}$; (d) эффект изменения величины вектора кручения в $B_{0,0}$.

где

$$\begin{aligned}[U] &= [u^n \ u^{n-1} \ \dots \ 1] \\ [W] &= [w^n \ w^{m-1} \ \dots \ 1]^T \\ [B] &= \begin{bmatrix} B_{0,0} & \dots & B_{0,m} \\ \vdots & \ddots & \vdots \\ B_{n,0} & \dots & B_{n,m} \end{bmatrix},\end{aligned}$$

а матрицы $[N]$ и $[M]$ задаются уравнением (5-70) или (5-71).

Для специального случая бикубической поверхности Безье размера 4×4 уравнение (6-59) сокращается до

$$Q(u, w) = [u^3 \ u^2 \ u \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\ B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,0} & B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,0} & B_{3,1} & B_{3,2} & B_{3,3} \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}. \quad (6-60)$$

Поверхность Безье не обязательно должна быть квадратной. Для сетки размера 5×3 уравнение (6-59) превращается в

$$Q(u, w) = [u^4 \ u^3 \ u^2 \ u \ 1] \begin{bmatrix} 1 & -4 & 6 & -4 & 1 \\ -4 & -12 & -12 & 4 & 0 \\ 6 & -12 & 6 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} B_{0,0} & B_{0,1} & B_{0,2} \\ B_{1,0} & B_{1,1} & B_{1,2} \\ B_{2,0} & B_{2,1} & B_{2,2} \\ B_{3,0} & B_{3,1} & B_{3,2} \\ B_{4,0} & B_{4,1} & B_{4,2} \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w^2 \\ w \\ 1 \end{bmatrix}. \quad (6-61)$$

Поверхность Безье размера 5×3 состоит из полиномиальных кривых четвертой степени в параметрическом направлении u и из квадратичных полиномиальных кривых в направлении w . Пример такой поверхности Безье показан на рис. 6-40. В данном случае, как это показано на рис. 6-40b, изменение центральной вершины стороны задающей сетки с пятью вершинами не влияет на касательные векторы в угловых точках.

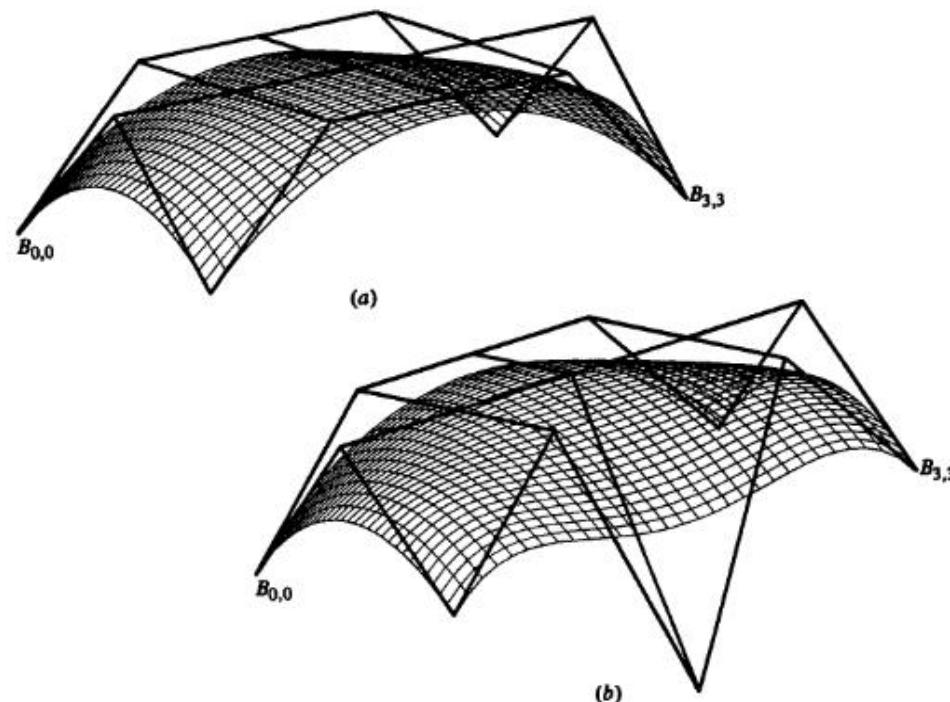


Рис. 6-40 Поверхность Безье размера 5×3 . (a) Основная поверхность; (b) эффект изменения центральной вершины граничной ломаной с пятью вершинами.

Производные поверхности Безье получаются с помощью формального дифференцирования уравнения (6-58) или (6-59). Если воспользоваться уравнением (6-58), то первые и вторые параметрические производные будут

$$Q_u(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J'_{n,i}(u) K'_{m,j}(w) \quad (6-62)$$

$$Q_w(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J_{n,i}(u) K'_{m,j}(w) \quad (6-63)$$

$$Q_{u,w}(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J'_{n,i}(u) K_m(j)(w) \quad (6-64)$$

$$Q_{u,u}(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J''_{n,i}(u) K'_{m,j}(w) \quad (6-65)$$

$$Q_{w,w}(u, w) = \sum_{i=0}^n \sum_{j=0}^m B_{i,j} J_{n,i}(u) K''_{m,j}(w) \quad (6-66)$$

где штрих обозначает дифференцирование относительно параметрической переменной. Производные функций базиса Бернштейна $J'_{n,i}$, $J''_{n,i}$, $K'_{m,j}$ и $K''_{m,j}$ приведены в уравнениях (5-74) и (5-75).

Легко найти соотношение между бикубическими поверхностями Безье и Кунса. Приравнивая уравнения (6-52) и (6-59), получим

$$\begin{aligned} Q_{\text{Кунс}}(u, w) &= Q_{\text{Безье}}(u, w), \\ [U][N_C][P][N_C]^T[W] &= [U][N_B][B][N_B]^T[W] \end{aligned}$$

где $[N_C]$ задан уравнением (5-76) и $[N_B]$ — уравнением (5-70). Следовательно геометрическая матрица $[P]$ бикубической поверхности Кунса задается в терминах полигональной сетки поверхности Безье следующим образом

$$[P] = [N_C]^{-1}[N_B][B][N_B]^T[N_C]^T]^{-1}$$

или

$$\begin{bmatrix} P(0, 0) & P(0, 1) & P_w(0, 0) & P_w(0, 1) \\ P(1, 0) & P(1, 1) & P_w(1, 0) & P_w(1, 1) \\ P_u(0, 0) & P_u(0, 1) & P_{uw}(0, 0) & P_{uw}(0, 1) \\ P_u(1, 0) & P_u(1, 1) & P_{uw}(1, 0) & P_{uw}(1, 1) \end{bmatrix} = \begin{bmatrix} B_0 & B_{0,3} \\ B_{3,0} & B_{3,3} \\ 3(B_{1,0} - B_{0,0}) & 3(B_{1,3} - B_{0,3}) \\ 3(B_{3,0} - B_{2,0}) & 3(B_{3,3} - B_{2,3}) \\ 3(B_{0,1} - B_{0,0}) & 3(B_{0,3} - B_{0,2}) \\ 3(B_{3,1} - B_{3,0}) & 3(B_{3,3} - B_{3,2}) \\ 9(B_{0,0} - B_{1,0} - B_{0,1} + B_{1,1}) & 9(B_{0,2} - B_{1,2} - B_{0,3} + B_{1,3}) \\ 9(B_{2,0} - B_{3,0} - B_{2,1} + B_{3,1}) & 9(B_{2,2} - B_{3,2} - B_{2,3} + B_{3,3}) \end{bmatrix} \quad (6-67)$$

Исследование правой нижней подматрицы размера 2×2 в уравнении (6-67) подтверждает, что четыре центральные вершины задающей полигональной сетки влияют на кручение в угловых точках куска бикубической поверхности Безье. Тем не менее кручение в угловых точках управляет не только центральными вершинами, но также и соседними касательными векторами. В самом деле, кручение в угловой точке управляется формой неплоского четырехугольника, сформированного угловой точкой, двумя соседними граничными точками и соседней центральной точкой.

Из уравнений (6-62)–(6-64) следует, что

$$\begin{bmatrix} P(0, 0) & P(0, 1) & P_w(0, 0) & P_w(0, 1) \\ P(1, 0) & P(1, 1) & P_w(1, 0) & P_w(1, 1) \\ P_u(0, 0) & P_u(0, 1) & P_{uw}(0, 0) & P_{uw}(0, 1) \\ P_u(1, 0) & P_u(1, 1) & P_{uw}(1, 0) & P_{uw}(1, 1) \end{bmatrix} = \begin{bmatrix} Q(0, 0) & Q(0, 1) & Q_w(0, 0) & Q_w(0, 1) \\ Q(1, 0) & Q(1, 1) & Q_w(1, 0) & Q_w(1, 1) \\ Q_u(0, 0) & Q_u(0, 1) & Q_{uw}(0, 0) & Q_{uw}(0, 1) \\ Q_u(1, 0) & Q_u(1, 1) & Q_{uw}(1, 0) & Q_{uw}(1, 1) \end{bmatrix}. \quad (6-68)$$

Аналогичным образом обратное соотношение между матрицами $[P]$ и $[B]$, выражающее вершины полигональной сетки Безье в терминах параметров бикубической поверхности Кунса, равно

$$\begin{bmatrix} B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\ B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,0} & B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,0} & B_{3,1} & B_{3,2} & B_{3,3} \end{bmatrix} = \left(\frac{1}{3}\right) \begin{bmatrix} 3P(0,0) & 3P(0,0) + P_w(0,0) \\ 3P(0,0) + P_u(0,0) & \frac{1}{3}\{P_{uw}(0,0) + 9P(0,0) - 3(P_w(0,0) + P_u(0,0))\} \\ 3P(1,0) - P_u(1,0) & \frac{1}{3}\{P_{uw}(1,0) + 9P(1,0) + 3(P_w(1,0) - P_u(1,0))\} \\ 3P(1,0) & 3P(1,0) + P_w(1,0) \\ 3P(0,1) - P_w(0,1) & 3P(0,1) \\ \frac{1}{3}\{P_{uw}(0,1) + 9P(0,1) + 3(P_u(0,1) - P_w(0,1))\} & 3P(0,1) + P_u(0,1) \\ \frac{1}{3}\{P_{uw}(1,1) + 9P(1,1) - 3(P_u(1,1) + P_w(1,1))\} & 3P(1,1) - P_u(1,1) \\ 3P(1,1) - P_w(1,1) & 3P(1,1) \end{bmatrix}. \quad (6-69)$$

Более полно концепция поверхности Безье иллюстрируется на следующем примере.

Пример 6-14 Поверхность Безье

Для изображенной на рис. 6-39а поверхности Безье для значений параметров $u = w = 0.5$ определить координаты точки на поверхности и первые производные в u и w направлениях. Найти также координаты точки и производные для модифицированной поверхности, показанной на рис. 6-39d. Сравнить полученные результаты. Вершины многогранника 4×4 поверхности Безье таковы:

$$[B] = \begin{bmatrix} [-15 \ 0 \ 15] & [-15 \ 5 \ 5] & [-15 \ 5 \ -5] & [-15 \ 0 \ -15] \\ [-5 \ 5 \ 15] & [-5 \ 5 \ 5] & [-5 \ 5 \ -5] & [-5 \ 5 \ -15] \\ [5 \ 5 \ 15] & [5 \ 5 \ 5] & [5 \ 5 \ -5] & [5 \ 5 \ -15] \\ [15 \ 0 \ 15] & [15 \ 5 \ 5] & [15 \ 5 \ -5] & [15 \ 0 \ -15] \end{bmatrix}$$

Для модифицированной поверхности на рис. 6-39d изменяется только вершина $B_{1,1}[0 \ 10 \ 0]$, т. е. изменение сказывается лишь на кручении в угловой точке $B_{0,0}$.

Напомним матричное представление поверхности, заданное в уравнениях (6-59) и (6-60), т. е.

$$Q(u, w) = [U][N][B][N]^T[W].$$

Здесь

$$\begin{aligned} [N][B][N]^T &= \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} [B] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} [0 \ 0 \ 0] & [0 \ 0 \ 0] & [0 \ 0 \ 0] & [0 \ 0 \ 0] \\ [0 \ 0 \ 0] & [0 \ -45 \ 0] & [0 \ 45 \ 0] & [0 \ -15 \ 0] \\ [0 \ 0 \ 0] & [0 \ 45 \ 0] & [0 \ -45 \ 0] & [30 \ 15 \ 0] \\ [0 \ 0 \ 0] & [0 \ -15 \ 0] & [0 \ 15 \ -30] & [-15 \ 0 \ 15] \end{bmatrix} \end{aligned}$$

Точка на поверхности, таким образом, будет

$$\begin{aligned} Q(0.5, 0.5) &= [0.125 \ 0.25 \ 0.5 \ 1][N][B][N]^T \begin{bmatrix} 0.125 \\ 0.25 \\ 0.5 \\ 1 \end{bmatrix} \\ &= [0 \ 4.6875 \ 0]. \end{aligned}$$

Производные по направлениям:

$$Q_u(0.5, 0.5) = [3u^2 \ 2u \ 1 \ 0][N][B][N]^T \begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

$$\begin{aligned} &= [0.75 \ 1 \ 1 \ 0][N][B][N]^T \begin{bmatrix} 0.125 \\ 0.25 \\ 0.5 \\ 1 \end{bmatrix} \\ &= [30 \ 0 \ 0] \end{aligned}$$

$$\begin{aligned} Q_w(0.5, 0.5) &= [u^3 \ u^2 \ u \ 1][N][B][N]^T \begin{bmatrix} 3w^2 \\ 2w \\ 1 \\ 0 \end{bmatrix} \\ &= [0.125 \ 0.25 \ 0.5 \ 1][N][B][N]^T \begin{bmatrix} 0.75 \\ 1 \\ 1 \\ 0 \end{bmatrix} \\ &= [0 \ 0 \ -30] \end{aligned}$$

Заметим, что векторы Q_u и Q_w ортогональны.

В модифицированной поверхности на рис. 6-39d изменено значение только точки $B_{1,1}$ стандартной поверхности. Новое значение координат равно $B_{1,1}[0 \ 10 \ 0]$. Новое значение произведения:

$$[N][B][N]^T =$$

$$\begin{bmatrix} [45 \ 45 \ -45] & [-90 \ -90 \ 90] & [45 \ 45 \ -45] & [0 \ 0 \ 0] \\ [-90 \ -90 \ 90] & [180 \ 135 \ -180] & [-90 \ -45 \ 90] & [0 \ -15 \ 0] \\ [45 \ 45 \ -45] & [-90 \ -45 \ 90] & [45 \ 0 \ -45] & [30 \ 15 \ 0] \\ [0 \ 0 \ 0] & [-0 \ -15 \ 0] & [0 \ -15 \ -30] & [-15 \ 0 \ 15] \end{bmatrix}$$

Новые координаты точки поверхности для $u = w = 0.5$:

$$Q(0.5, 0.5) = [0.703 \ 5.391 \ -0.703].$$

Новые параметрические производные для $u = w = 0.5$:

$$Q_u(0.5, 0.5) = [28.594 \ -1.406 \ 1.406]$$

$$Q_w(0.5, 0.5) = [-1.406 \ -1.406 \ -28.594]$$

Заметим в этом случае, что поскольку задающая полигональная сетка более не симметрична относительно оси y , то поверхность теперь не симметрична относительно этой оси. Далее, заметим, что хотя Q_u и Q_w все еще ортогональны, но как их величины, так и направления отличаются. Данные результаты показывают, что вектор кручения в одной угловой точке оказывает трудно уловимое, но существенное влияние на форму всей поверхности.

Приведенное выше обсуждение поверхностей Безье касалось определения и характеристик одного куска поверхности. Для того чтобы получить более сложные поверхности, надо объединить несколько кусков поверхности Безье. Подробное обсуждение этого вопроса лежит вне сферы данной книги. Интересующегося читателя мы отсылаем к [6-24] и [6-25]. Проблемы, возникающие при объединении кусков поверхности Безье с обеспечением гладкости вдоль соприкасающихся сторон, иллюстрируются рис. 6-41 на примере объединения двух кусков бикубической поверхности Безье вдоль одной стороны.

Для обеспечения непрерывности или C^0 гладкости вдоль границы необходимо, чтобы совпадали¹ две граничные кривые, а следовательно, и две граничные ломаные вдоль края поверхности. Для обеспечения непрерывности векторов наклона или касательных векторов или C^1 гладкости вдоль границы куска направление нормали к поверхности вдоль граничной кривой должно быть одинаковым для обоих кусков. Для этого можно использовать два условия. Первое требует, чтобы четыре отрезка полигональной сетки, встречающиеся у границы и пересекающие ее, были коллинеарными, как это показано выделенными линиями на рис. 6-41a. Второе, менее жесткое условие требует, чтобы только три ребра полигональной сетки, встречающиеся в концевых точках граничной кривой, были компланарными, как это показано выделенными линиями на рис. 6-41b.

6-12 В-СПЛАЙН ПОВЕРХНОСТИ

Естественным расширением понятия поверхности Безье является декартово произведение В-сплайн поверхности, определяемой выражением

$$Q(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N_{i,k}(u) M_{j,l}(w), \quad (6-70)$$

где $N_{i,k}(u)$ и $M_{j,l}(w)$ – базисные функции В-сплайна в бипараметрических направлениях u и w соответственно (см. уравнение 5-84). Для удобства повторим здесь определение базисных функций, данное ранее в разд. 5-9,

$$N_{i,1}(u) = \begin{cases} 1, & \text{если } x_i \leq u < x_{i+1}, \\ 0, & \text{в противном случае,} \end{cases} \quad (5-84a)$$

$$N_{i,k}(u) = \frac{(u - x_i)N_{i,k-1}(u)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - u)N_{i+1,k-1}(u)}{x_{i+k} - x_{i+1}} \quad (5-84b)$$

¹ Вспомним обсуждение в разд. 5-8 гладкости сегментов кривой Безье.

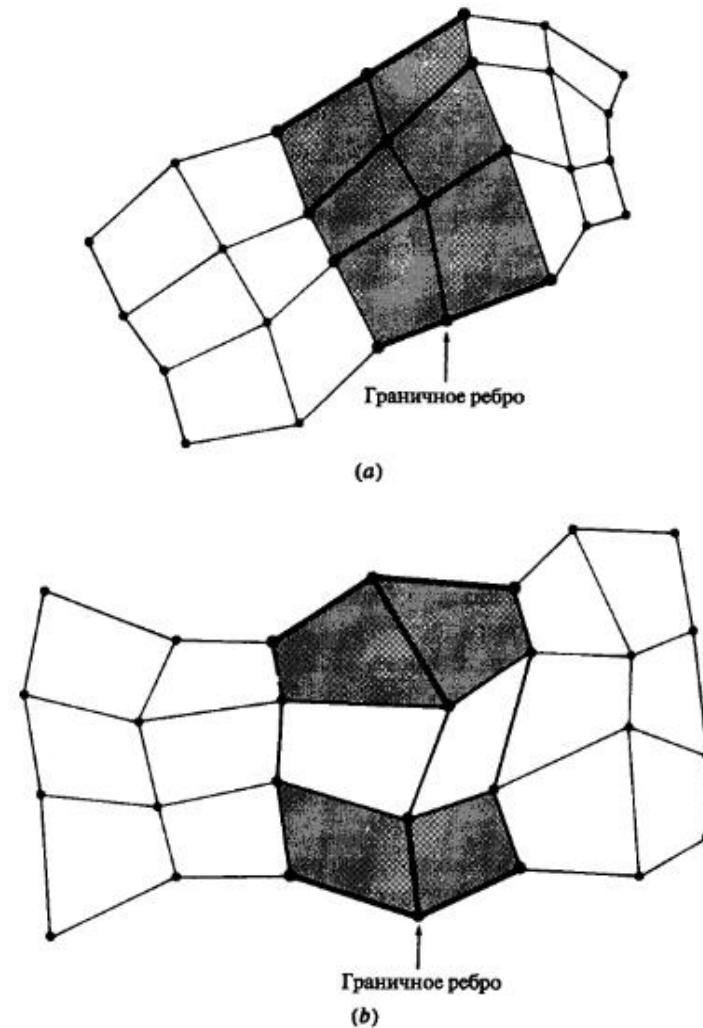


Рис. 6-41 Гладкость стыковки кусков поверхности Безье. (a) Линии сетки коллинеарны; (b) ребра ломаных компланарны.

и

$$M_{j,l}(w) = \begin{cases} 1, & \text{если } y_j \leq w < y_{j+1}, \\ 0, & \text{в противном случае,} \end{cases} \quad (5-84a)$$

$$M_{j,l}(w) = \frac{(w - y_j)M_{j,l-1}(w)}{y_{j+l-1} - y_j} + \frac{(y_{j+l} - w)M_{j+1,l-1}(w)}{y_{j+l} - y_{j+1}}, \quad (5-84b)$$

где x_i и y_j являются элементами узловых векторов, как это обсуждалось в разд. 5-9. Снова $B_{i,j}$ являются вершинами задающей полигональной сетки. Для четырехугольных «кусков» поверхности задающая полигональная сетка должна

быть топологически прямоугольной. Индексы n и m на единицу меньше числа вершин задающего многогранника соответственно в u и w параметрических направлениях.

Как и для В-сплайн кривых на форму и свойства В-сплайн поверхности существенно влияют узловые векторы $[X]$ и $[Y]$, причем используются незамкнутые, периодические и неоднородные узловые векторы. Хотя обычно для обоих параметрических направлений применяют узловые векторы одного и того же типа, но это не обязательно. Например, можно использовать незамкнутый узловый вектор и его соответствующие В-сплайн базисные функции для одного параметрического направления и периодический узловый вектор и его соответствующие В-сплайн базисные функции для другого. Практическим примером служит цилиндрическая поверхность с переменным сечением.

Так как для описания граничных кривых и для интерполяции внутренней части поверхности используется В-сплайн базис, то сразу же можно перечислить некоторые свойства В-сплайн поверхности:

Максимальный порядок поверхности в каждом параметрическом направлении равен числу вершин задающего многогранника в этом направлении.

Гладкость поверхности в каждом параметрическом направлении на две единицы меньше порядка поверхности в каждом направлении; т. е. C^{k-2} и C^{l-2} в u и w направлениях, соответственно.

Поверхность инвариантна относительно аффинного преобразования; т. е. поверхность преобразуется с помощью преобразования задающей полигональной сетки.

Свойство затухания изменений для В-сплайн поверхности в настоящее время неизвестно.

Влияние одной вершины полигональной сетки ограничивается $\pm k/2, \pm 1/2$ интервалами в каждом параметрическом направлении.

Если число вершин задающей полигональной сетки равно порядку в каждом параметрическом направлении и внутренних узловых величин нет, то В-сплайн поверхность превращается в поверхность Безье (см. рис. 6-39).

При триангуляции задающей полигональной сетки образует плоскую аппроксимацию поверхности.

Поверхность лежит внутри выпуклой оболочки задающей полигональной сетки, образуемой объединением всех выпуклых оболочек k, l соседних вершин полигональной сетки.

Из предыдущего обсуждения свойств выпуклой оболочки В-сплайн кривых (см. разд. 5-9) сразу же следует, что В-сплайн поверхность может содержать плоские области и линии резкого нарушения гладкости. Это свойство очень полезно во многих ситуациях, возникающих при конструировании. На рис. 6-42 $a - d$ изображена серия незамкнутых В-сплайн поверхностей и их характеристических многогранников третьего порядка в каждом характеристическом направлении. Отметим, что каждая из линий задающей полигональной сетки в направлении w является прямой линией с четырьмя вершинами. Получающаяся поверхность

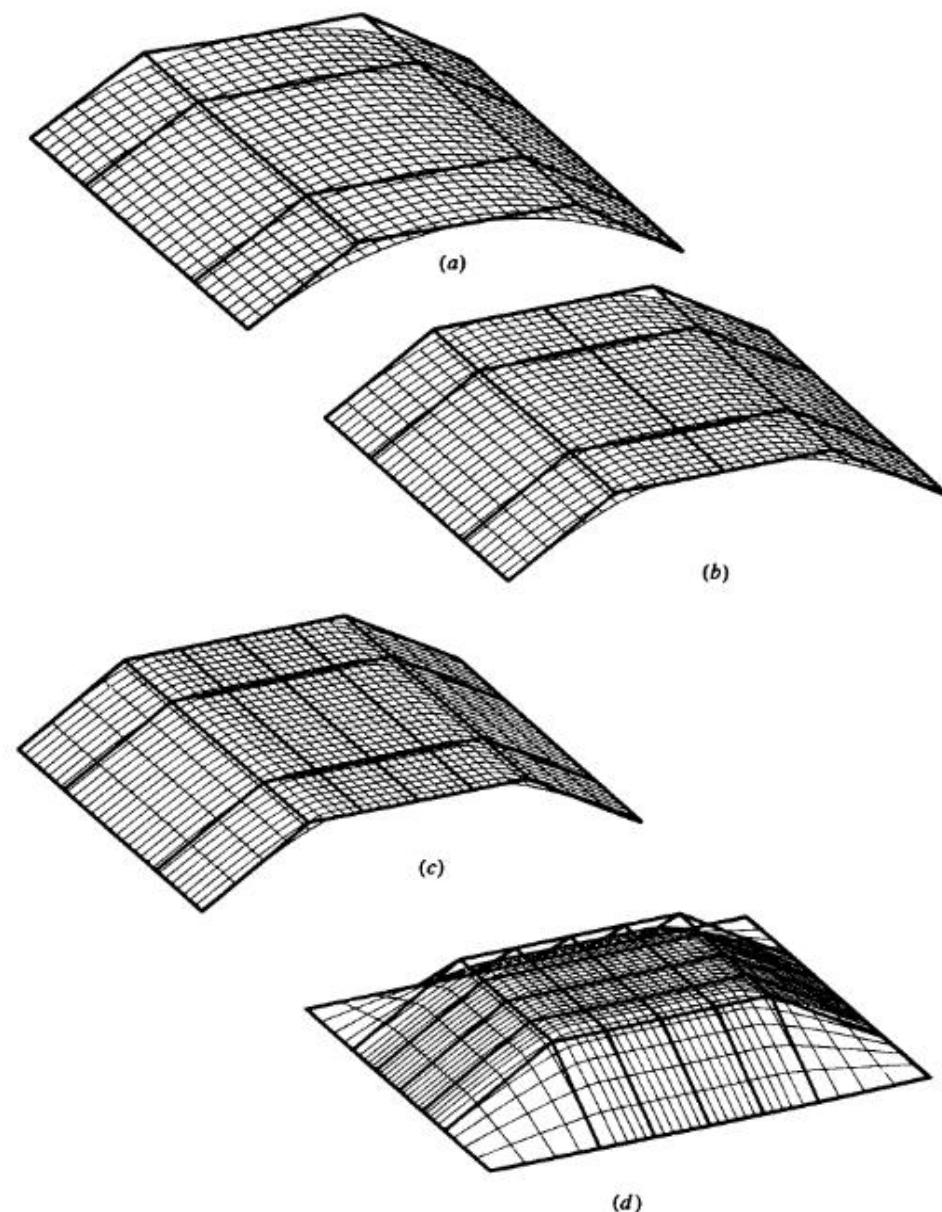


Рис. 6-42 В-сплайн поверхности третьего порядка. (a) Гладкая линейчатая поверхность; (b) небольшая внутренняя плоская область, вызванная коллинеарностью трех вершин сетки в u направлении; (c) обширная внутренняя плоская область, вызванная коллинеарностью пяти вершин сетки в u направлении; (d) плоская область внутри скульптурной поверхности.

линейчата в направлении w . Изображенная на рис. 6-42a В-сплайн поверхность, заданная четырьмя вершинами полигональной сетки в направлении u , плавно изогнута в этом направлении.

Изображенная на рис. 6-42b В-сплайн поверхность задана пятью вершинами полигональной сетки в направлении u . Три центральные вершины коллинеарны. Заметим, что центр получившейся поверхности имеет плоскую форму. Аналогичным образом коллинеарны пять из семи вершин задающей полигональной сетки в направлении u для поверхности, изображенной на рис. 6-42c. И снова в центральной области, которая имеет большие размеры, чем на рис. 6-42b, поверхность плоская.

На рис. 6-42d показано, что эти очень сильные свойства выпуклой оболочки распространяются на оба параметрических направления. Таким образом, плоская область может быть встроена во внутреннюю часть скульптурной поверхности. При увеличении порядка поверхности плоская область становится меньше.

На рис. 6-43 иллюстрируется эффект, возникающий при совпадении линий сетки. На рис. 6-43a три совпадающие линии сетки используются для образования линии складки в центре В-сплайн поверхности четвертого порядка. На рис. 6-43b показан результат смещения трех линий сетки в обоих параметрических направлениях. В этом случае В-сплайн поверхность четвертого порядка содержит два гребня, поднимающихся к точке в центре поверхности. Так же как и для В-сплайн кривых, линия складки возникает в том случае, когда совпадают $k - 1$ или $l - 1$ линий сетки. Кроме того, так как В-сплайн поверхность везде C^{k-2}/C^{l-2} гладка, то она гладка и на этой линии. Вдобавок данное свойство также гарантирует C^{k-2}/C^{l-2} гладкость перехода изогнутой поверхности в плоскую.

Превосходные свойства локального изменения В-сплайн кривых (см. разд. 5-9) переносятся на В-сплайн поверхности. Пример этого приведен на рис. 6-44, где незамкнутая бикубическая ($k = l = 4$) В-сплайн поверхность определена полигональной сеткой 9×9 ($m = n = 8$). Эта сетка, показанная как верхняя поверхность на рис. 6-44, плоская везде, за исключением центральной точки. Незамкнутый узловой вектор в обоих параметрических направлениях равен $[0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 6 \ 6 \ 6]$. Таким образом, у нас есть шесть промежутков параметров в каждом направлении, т. е. $0 - 1, 1 - 2, \dots, 5 - 6$. Каждый параметрический четырехугольник, например $0 \leq u \leq 1, 0 \leq w \leq 1$, образует подкусок¹ В-сплайн поверхности. Средняя поверхность, изображенная на рис. 6-44, составлена из параметрических линий на концах каждого параметрического интервала, т. е. в $u = w = 0, 1, 2, 3, 4, 5, 6$. Каждый четырехугольник представляет подкусок поверхности. Отметим, что влияние смещенной точки ограничивается $\pm k/2, l/2$ интервалами или подкусками.

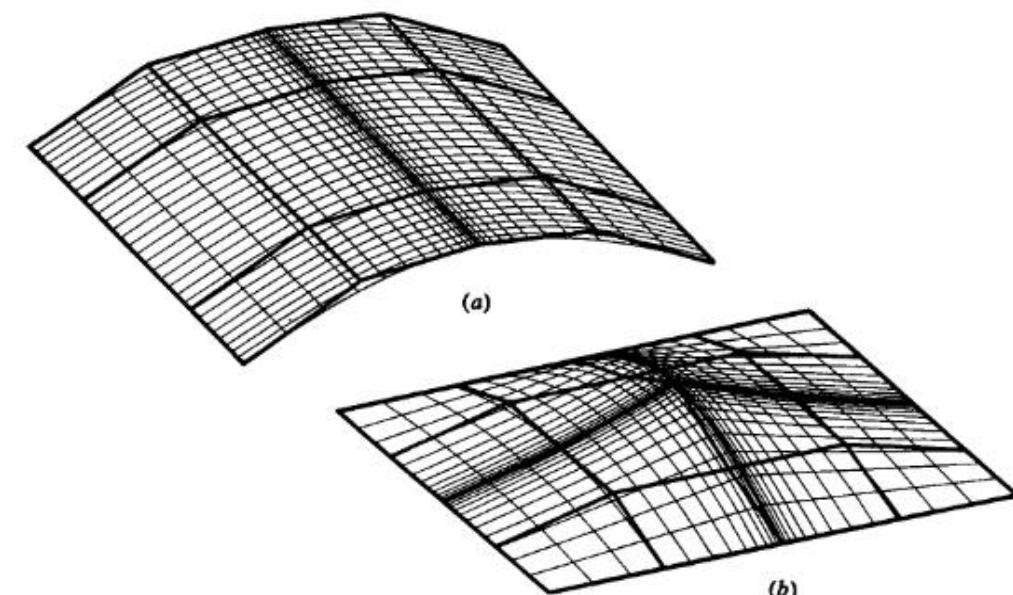


Рис. 6-43 В-сплайн поверхности четвертого порядка с несколькими совпадающими линиями сетки.

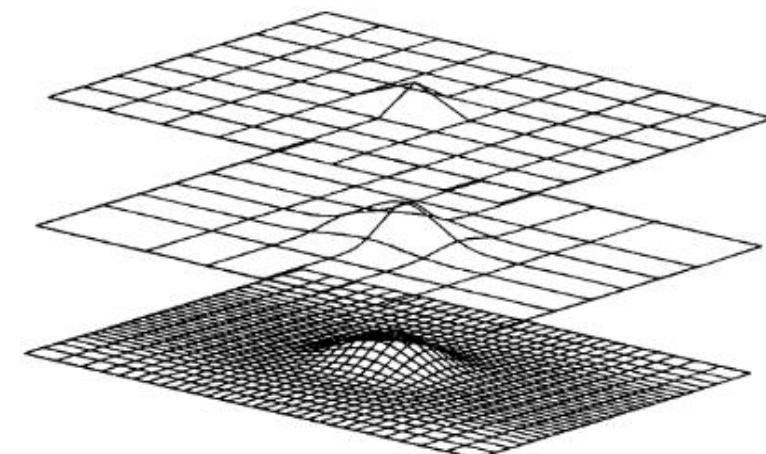


Рис. 6-44 Локальное изменение В-сплайн поверхностей.

¹ Некоторые авторы обозначают каждый подкусок как В-сплайн поверхность. В данном случае поверхность рассматривается как объект, заданный всей полигональной сеткой.

Параметрические производные В-сплайн поверхности получаются с помощью формального дифференцирования уравнения (6-70):

$$Q_u(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N'_{i,k}(u) M_{j,l}(w), \quad (6-71)$$

$$Q_w(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N_{i,k}(u) M'_j(w), \quad (6-72)$$

$$Q_{uw}(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N'_{i,k}(u) M'_{j,l}(w), \quad (6-73)$$

$$Q_{uu}(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N''_{i,k}(u) M_{j,l}(w), \quad (6-74)$$

$$Q_{ww}(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} N_{i,k}(u) M''_{j,l}(w), \quad (6-75)$$

где штрих обозначает дифференцирование относительно соответствующего параметра. Производные В-сплайн базисных функций задаются уравнениями (5-97)–(5-100).

Приведенный пример иллюстрирует метод вычисления В-сплайн поверхности.

Пример 6-15 Вычисление незамкнутой В-сплайн поверхности

Рассмотрим В-сплайн поверхность, заданную полигональной сеткой размера 4×4 :

$$\begin{array}{ll} B_{1,1}[-15 & 0 & 15] & B_{2,1}[-5 & 5 & -15] \\ B_{1,2}[-15 & 5 & 5] & B_{2,2}[-5 & 10 & 5] \\ B_{1,3}[-15 & 5 & -5] & B_{2,3}[-5 & 10 & -5] \\ B_{1,4}[-15 & 0 & -15] & B_{2,4}[-5 & 5 & -15] \end{array} \quad \begin{array}{ll} B_{3,1}[5 & 5 & 15] & B_{4,1}[15 & 0 & 15] \\ B_{3,2}[5 & 10 & 5] & B_{4,2}[15 & 5 & 5] \\ B_{3,3}[5 & 10 & -5] & B_{4,3}[15 & 5 & -5] \\ B_{3,4}[5 & 5 & -15] & B_{4,4}[15 & 0 & -15] \end{array}$$

Это поверхность четвертого порядка в направлении u ($k = 4$) и третьего порядка в направлении w ($l = 3$). Таким образом, поверхность составлена из двух подкусков: один для $0 \leq u \leq 1$, $0 \leq w \leq 1$ и другой для $0 \leq u \leq 1$, $1 \leq w \leq 2$. Надо найти точку в центре поверхности, т. е. при $u = 1/2$, $w = 1$.

Расписывая уравнение (6-70), получим

$$\begin{aligned} Q(u, w) &= \sum_{i=1}^4 \sum_{j=1}^4 B_{i,j} N_{i,4}(u) M_{j,3}(w) \\ &= N_{1,4}(B_{1,1}M_{1,3} + B_{1,2}M_{2,3} + B_{1,3}M_{3,3} + B_{1,4}M_{4,3}) \\ &\quad + N_{2,4}(B_{2,1}M_{1,3} + B_{2,2}M_{2,3} + B_{2,3}M_{3,3} + B_{2,4}M_{4,3}) \\ &\quad + N_{3,4}(B_{3,1}M_{1,3} + B_{3,2}M_{2,3} + B_{3,3}M_{3,3} + B_{3,4}M_{4,3}) \\ &\quad + N_{4,4}(B_{4,1}M_{1,3} + B_{4,2}M_{2,3} + B_{4,3}M_{3,3} + B_{4,4}M_{4,3}). \end{aligned}$$

Здесь узловые векторы в u направлении $[X] = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$. Вспомнив пример (5-12), получим выражения для базисных функций, т. е.

$$\begin{aligned} N_{1,4}\left(\frac{1}{2}\right) &= (1-u)^3 = \left(\frac{1}{2}\right)^3 = \frac{1}{8}, \\ N_{2,4}\left(\frac{1}{2}\right) &= 3u(1-u)^2 = (3)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right)^2 = \frac{3}{8}, \\ N_{3,4}\left(\frac{1}{2}\right) &= 3u^2(1-u) = (3)\left(\frac{1}{2}\right)^2\left(\frac{1}{2}\right) = \frac{3}{8}, \\ N_{4,4}\left(\frac{1}{2}\right) &= u^3 = \left(\frac{1}{2}\right)^3 = \frac{1}{8}. \end{aligned}$$

Аналогичным образом узловые векторы в направлении w имеют вид $[Y] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$. Вспомнив пример (5-10), получим выражения для базисных функций, т. е.

$$\begin{aligned} M_{1,3}(1) &= 0, \\ M_{2,3}(1) &= \frac{(2-w)^2}{2} = \frac{(2-1)^2}{2} = \frac{1}{2}, \\ M_{3,3}(1) &= \frac{(2-w)(3w-2)}{2} = \frac{(2-1)(3-2)}{2} = \frac{1}{2}, \\ M_{4,3}(1) &= (w-1)^2 = (1-1) = 0. \end{aligned}$$

Таким образом,

$$\begin{aligned} Q(1/2, 1) &= \frac{1}{8}\{(0)B_{1,1} + \frac{1}{2}B_{1,2} + \frac{1}{2}B_{1,3} + (0)B_{1,4}\} \\ &\quad + \frac{3}{8}\{(0)B_{2,1} + \frac{1}{2}B_{2,2} + \frac{1}{2}B_{2,3} + (0)B_{2,4}\} \\ &\quad + \frac{3}{8}\{(0)B_{3,1} + \frac{1}{2}B_{3,2} + \frac{1}{2}B_{3,3} + (0)B_{3,4}\} \\ &\quad + \frac{1}{8}\{(0)B_{4,1} + \frac{1}{2}B_{4,2} + \frac{1}{2}B_{4,3} + (0)B_{4,4}\} \\ Q(1/2, 1) &= \frac{1}{16}(B_{1,2} + B_{1,3}) + \frac{3}{16}(B_{2,2} + B_{2,3}) \\ &\quad + \frac{3}{16}(B_{3,2} + B_{3,3}) + \frac{1}{16}(B_{4,2} + B_{4,3}) \\ &= \frac{1}{16}\{[-15 \ 5 \ 5] + [-15 \ 5 \ -5]\} \\ &\quad + \frac{3}{16}\{[-5 \ 10 \ 5] + [-5 \ 10 \ -5]\} \\ &\quad + \frac{3}{16}\{[5 \ 10 \ 5] + [5 \ 10 \ -5]\} \\ &\quad + \frac{1}{16}[15 \ 5 \ 5] + [15 \ 5 \ -5]\} \\ &= [0 \ 35/4 \ 0] \end{aligned}$$

Периодические В-сплайн поверхности легко генерируются с помощью периодических базисных функций в уравнении (6-70), для получения которых используются периодические узловые векторы. На рис. 6-45 показано несколько приме-

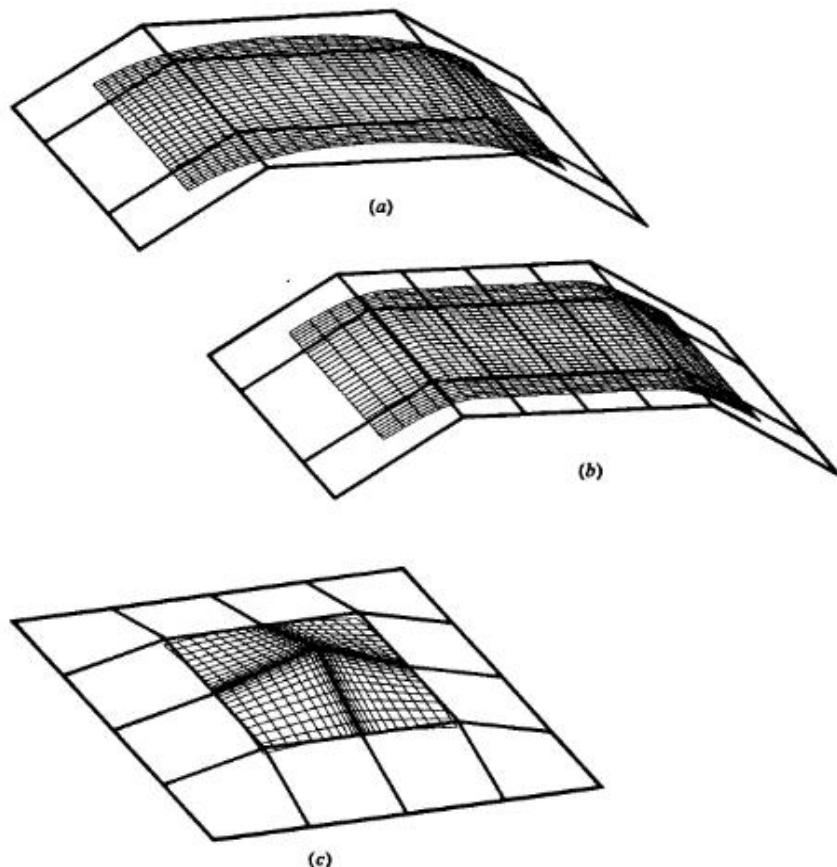


Рис. 6-45 Периодические В-сплайн поверхности для незамкнутых характеристических многогранников. (а) Гладкая линейчатая поверхность третьего порядка; (б) большая внутренняя плоская область третьего порядка, вызванная коллинеарностью пяти вершин сетки в *u* и направлениях; (с) острый выступ на поверхности четвертого порядка, вызванный пересечением нескольких совпадающих линий сетки.

ров периодических В-сплайн поверхностей, формируемых незамкнутыми задающими полигональными сетками. На рисунках 6-45*a* и *b* задающие полигональные сетки соответствуют сеткам рисунков 6-42*a* и *c*, а сетка рис. 6-45*c* соответствует сетке рис. 6-43*b*. Заметим, что во всех случаях, так же как и для периодических В-сплайн кривых, границы поверхности и многогранника не совпадают из-за уменьшения диапазона параметра, используемого для периодических В-сплайн базисных функций.

Замкнутые периодические В-сплайн поверхности демонстрируют свойства, аналогичные свойствам замкнутых периодических В-сплайн кривых. На рис. 6-46 показаны примеры трех поверхностей третьего порядка. Задающая полигональная сетка на рис. 6-46*a* образуется с помощью повторения через одинаковые промежутки вдоль оси *z* от точки *z* = -8 до *z* = 8 задающего многоугольника для

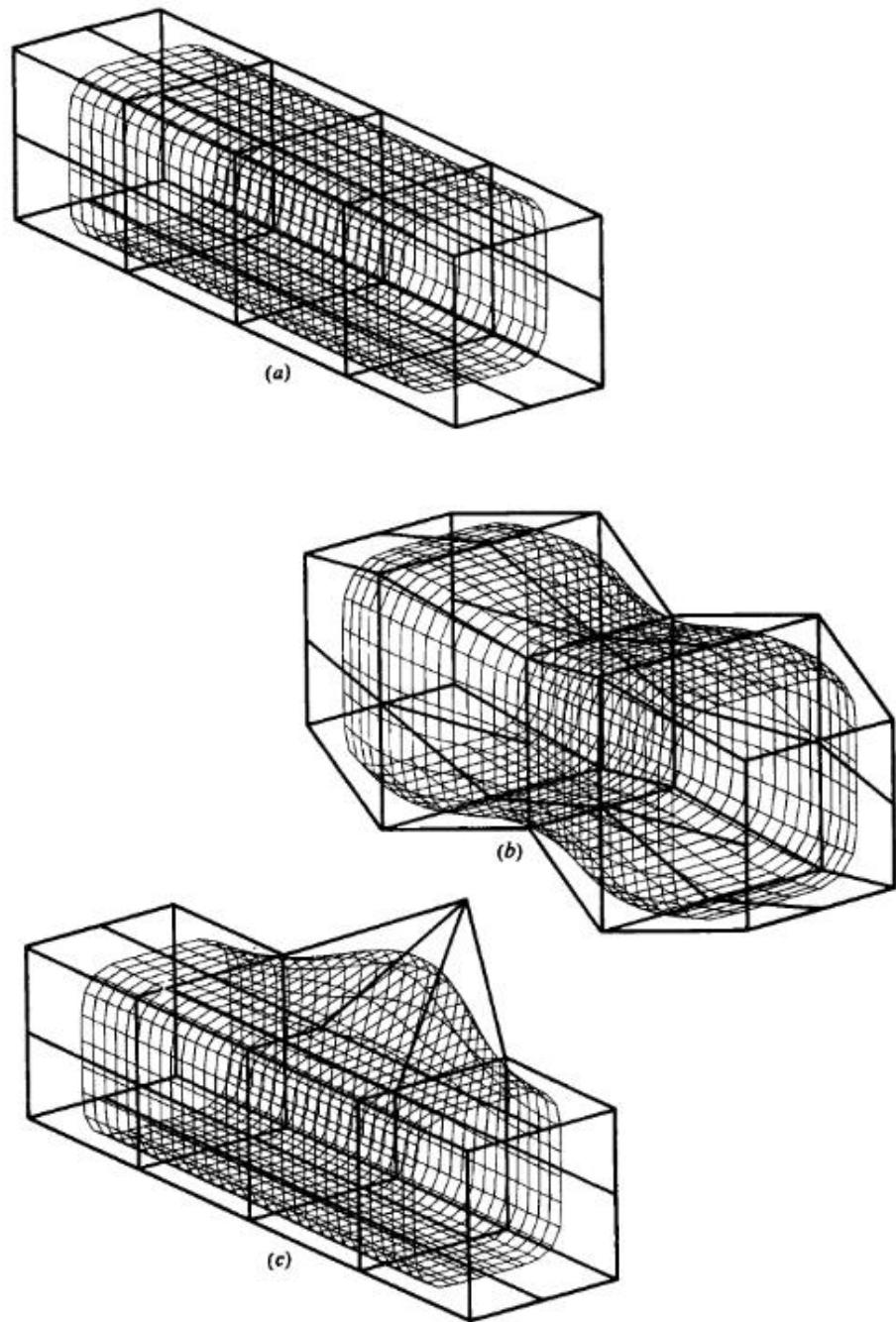


Рис. 6-46 Замкнутые периодические В-сплайн поверхности. (а) Прямая цилиндрическая поверхность; (б) возмущенная волнистая цилиндрическая поверхность; (с) эффект возмущения одной вершины сетки.

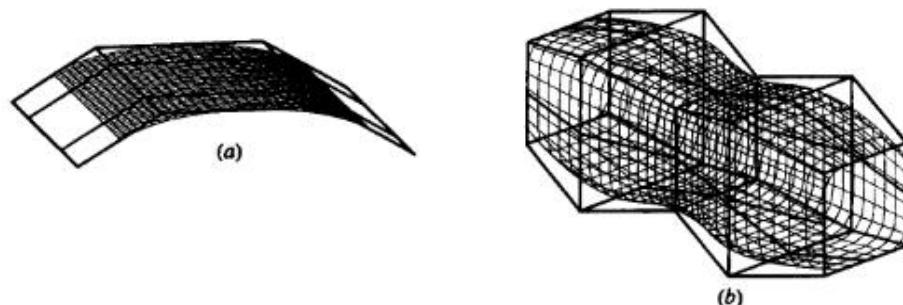


Рис. 6-47 Объединенные В-сплайн поверхности третьего порядка. (a) Незамкнутый характеристический многогранник; (b) замкнутый характеристический многогранник.

замкнутой В-сплайн кривой на рис. 6-46b. В результате получается цилиндрическая поверхность. Отметим, что поверхность не касается плоскостей первого и последнего задающих многоугольников. Характеристический многогранник для рис. 6-46b получен с помощью увеличения на единицу x - и y -размеров второго и четвертого задающих многоугольников на рис. 6-46a. В результате получается волнистый цилиндр. На рис. 6-46c показан локальный эффект возмущения одной вершины задающей сетки.

В уравнении (6-70) можно комбинировать незамкнутые и периодические В-сплайн базисные функции. Два примера этого представлены на рис. 6-47. Здесь в одном параметрическом направлении используются незамкнутый узловый вектор и базисная функция, а в другом направлении используются периодический узловый вектор и базисная функция. На рис. 6-47a показана комбинированная В-сплайн поверхность, заданная незамкнутой полигональной сеткой с рис. 6-42a. На рис. 6-47b показана комбинированная В-сплайн поверхность, заданная замкнутой полигональной сеткой с рис. 6-46b. Отметим, что поверхность совпадает с крайними линиями полигональной сетки в направлении u . Это свойство бывает полезным в некоторых случаях.

Матричное выражение для периодических В-сплайн поверхностей имеет вид

$$Q(s, t) = [U^*][N^*][B_{s,t}^*][M^*]^T[W^*]^T \quad (6-76)$$

где $[U^*]$ и $[W^*]$ являются репараметризованными параметрическими переменными в интервалах $0 \leq u^* \leq 1$ и $0 \leq w^* \leq 1$, заданными в уравнении (5-90). $[N^*]$ и $[M^*]$ задаются уравнением (5-91). Матрица $[B_{s,t}^*]$ представляет скользящую сетку $k \times l$ вершин характеристического многогранника, задающего подкусок на поверхности. Для периодических В-сплайн поверхностей, заданных незамкнутыми полигональными сетками

$$[B_{s,t}^*] = [B_{i,j}], \quad (6-77)$$

где

$$\begin{aligned} 1 \leq s &\leq n - k + 2, & s &\leq i \leq s + k - 1, \\ 1 \leq t &\leq m - l + 2, & t &\leq j \leq t + l - 1 \end{aligned} \quad (6-78)$$

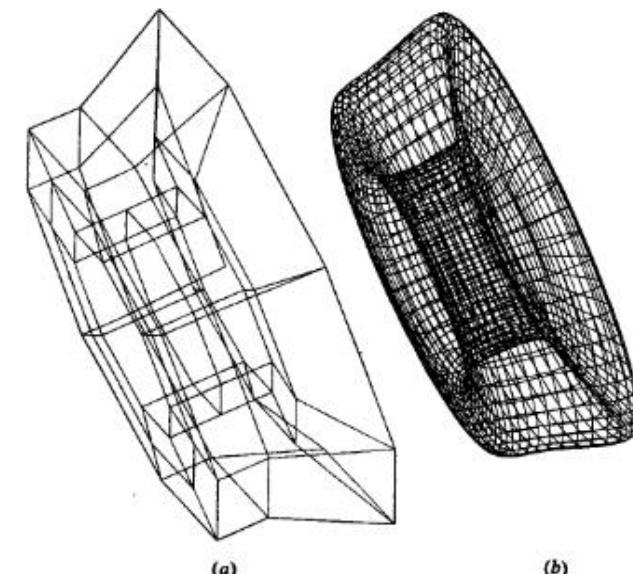


Рис. 6-48 Замкнутая тороидальная бикубическая ($k = l = 4$) В-сплайн поверхность. (a) Характеристический многогранник; (b) поверхность.

и $B_{i,j}$ представляет индивидуальные элементы задающей полигональной сетки.

Для полигональных сеток, замкнутых вдоль $u = 0$, т. е. с совпадающими первой и последней сеточными линиями в направлении u , скользящая сетка задается так¹

$$\begin{aligned} 1 &\leq s \leq n - k + 2, \\ 1 &\leq t \leq m + l, \\ s &\leq i \leq s + k - 1, \\ j &\in [(t - 1) \bmod (m + 1)] + 1 : (t + l - 2) \bmod (m + 1) + 1. \end{aligned} \quad (6-79)$$

Аналогичным образом, для полигональных сеток, замкнутых вдоль $w = 0$, скользящая сетка задается в виде

$$\begin{aligned} 1 &\leq s \leq n + 1, \\ 1 &\leq t \leq m - l + 2, \\ i &\in [(s - 1) \bmod (n + 1)] + 1 : (s + k - 2) \bmod (n + 1) + 1, \\ t &\leq j \leq t + l - 1. \end{aligned} \quad (6-80)$$

¹ $j \in [a : b]$ означает принадлежность множеству $a \dots b$. В данном случае списки вершин многогранника должны рассматриваться как закольцованые. Например, если $m + 1 = 4$, то $j \in [3 : 2]$ означает принадлежность множеству 3, 4, 1, 2 в указанном порядке.

И наконец, для полигональных сеток, замкнутых вдоль как $u = 0$, так и $w = 0$, скользящая сетка задается так:

$$\begin{aligned} 1 &\leq s \leq n - k + 2, \\ 1 &\leq t \leq m - l + 2, \\ i &\in [\{(s-1) \bmod (n+1)\} + 1 : (s+k-2) \bmod (n+1) + 1], \\ j &\in [\{(t-1) \bmod (m+1)\} + 1 : (t+l-2) \bmod (m+1) + 1]. \end{aligned} \quad (6-81)$$

В этом случае образуется полностью замкнутая поверхность. Пример изображен на рис. 6-48. Задающая полигональная сетка, изображенная на рис. 6-48a, формируется с помощью переноса вершин задающего многоугольника для периодической В-сплайн кривой на рис. 6-47 на -2 единицы по x и $+4$ единицы по y и затем вращения на 360° вокруг оси x с шагом 45° . Замкнутая периодическая бикубическая ($k = l = 4$) В-сплайн поверхность, изображенная на рис. 6-48b, имеет форму тора.

Матричное выражение для незамкнутых В-сплайн поверхностей имеет ту же форму, что и уравнение (6-76). Однако, так же как и для матричного выражения незамкнутых В-сплайн кривых, существование нескольких узловых значений на концах узлового вектора делает этот результат менее компактным и менее полезным, чем для периодических В-сплайн поверхностей. По этим причинам данный вопрос не рассматривается здесь более подробно.

6-13 В-СПЛАЙН ИНТЕРПОЛЯЦИЯ

В предыдущих разделах обсуждались свойства и создание В-сплайн поверхностей по известной задающей полигональной сетке. Также представляет интерес и обратная задача: т. е. задано известное множество данных на поверхности, надо найти задающую полигональную сетку для В-сплайн поверхности, наилучшим образом интерполирующей эти данные. Так как границы поверхности, представляемой этими данными, обычно известны, то здесь будут рассматриваться только незамкнутые В-сплайн поверхности. Разработка аналогичного метода для замкнутых поверхностей с помощью периодических В-сплайн поверхностей не представляет труда. Обсуждение ограничивается топологически прямоугольными сетками, т. е. данные умозрительно могут быть представлены как находящиеся в точках пересечения прямоугольной решетки. Эта задача схематично изображена на рис. 6-49, где показана задающая полигональная сетка размера 4×4 для матрицы данных 8×8 . Эта задача рассматривалась в [6-5] и в [6-26]. Для того чтобы повысить эффективность вычислений, в работе [6-26] были использованы известные свойства бикубических В-сплайн поверхностей. Применяемый нами метод более прямолинеен, но имеет меньшую вычислительную эффективность (см. разд. 5-11).

Вспомним уравнение (6-70) и заметим, что $Q(u, w)$ в этом уравнении являются известными точками на поверхности. Базисные функции $N_{i,k}(u)$ и $M_{j,l}(w)$ могут быть определены для известного порядка и известного числа вершин задающей полигональной сетки в каждом параметрическом направлении при условии, что

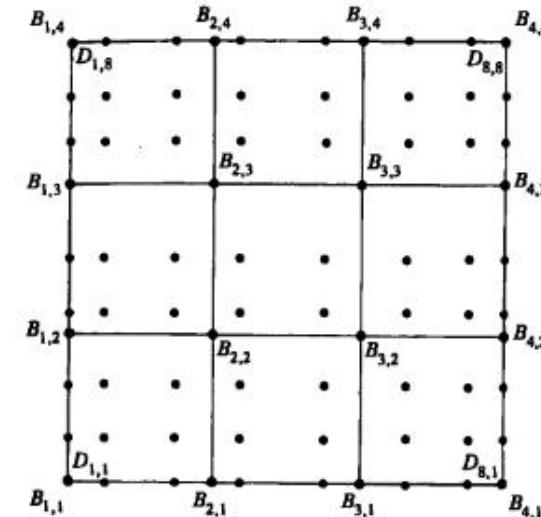


Рис. 6-49 Определение В-сплайн поверхности по известному набору данных.

известны значения параметров u, w в исходных точках. Следовательно, для каждой известной точки поверхности уравнение (6-70) дает линейное уравнение от неизвестных вершин B_{ij} задающей полигональной сетки. В результате расписывания уравнения (6-70) для исходной точки поверхности получим

$$D_{1,1}(u_1, w_1) =$$

$$N_{1,k}(u_1)[M_{1,l}(w_1)B_{1,1} + M_{2,l}(w_1)B_{1,2} + \dots + M_{m+1,l}(w_1)B_{1,m+1}] +$$

⋮

$$N_{n+1,k}(u_1)[M_{1,l}(w_1)B_{n+1,1} + M_{2,l}(w_1)B_{n+1,2} + \dots + M_{m+1,l}(w_1)B_{n+1,m+1}]$$

где для топологически прямоугольного множества $r \times s$ данных выполняются неравенства $2 \leq k \leq n+1 \leq r$ и $2 \leq l \leq m+1 \leq s$. Записав уравнения такого вида для всех исходных точек, мы получим систему уравнений. В матричном виде данный результат записывается так:

$$[D] = [C][B], \quad (6-82)$$

где $C_{i,j} = N_{i,k} M_{j,l}$. Для топологически прямоугольных исходных данных размера $r \times s$ матрица $[D]$ является матрицей $r \times s \times 3$, содержащей трехмерные координаты исходных точек, $[C]$ является матрицей $r \times s \times n \times m$ трехмерных координат искомых точек полигональной сетки.

Если матрица $[C]$ квадратная, то задающая полигональная сетка находится с помощью обращения матрицы, т. е.

$$[B] = [C]^{-1}[D]. \quad (6-83)$$

В этом случае получившаяся поверхность проходит через все исходные точки. Хотя эта поверхность будет везде C^{k-2}, C^{l-2} гладка, но она может быть непри-

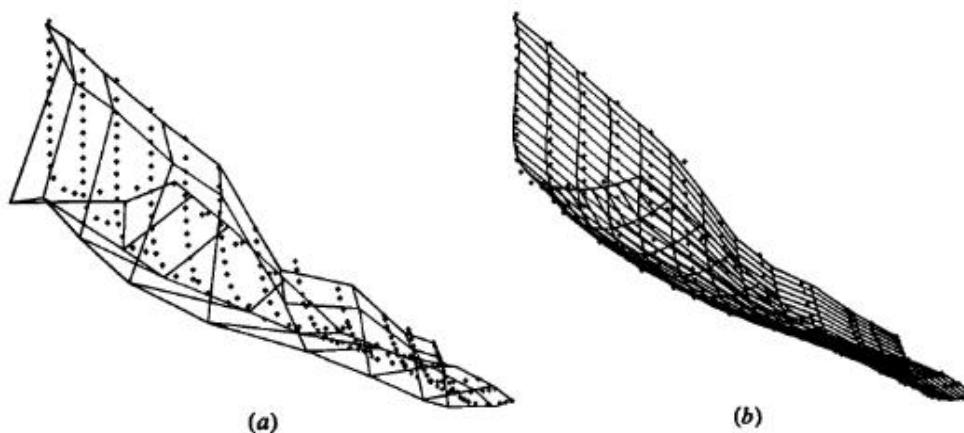


Рис. 6-50 Интерполяция В-сплайн поверхностью. (а) Исходные точки и интерполирующий характеристический многогранник; (б) исходные точки и сгенерированная поверхность.

емлема с точки зрения пользователя. Опыт показывает, что в общем случае чем меньше точек задающей полигональной сетки, тем лучше выглядит поверхность.

Если $[C]$ не квадратная, то мы имеем избыточное количество исходных данных и решение может быть получено в некотором усредненном смысле. В частности, решение может быть получено в виде

$$[B] = [[C]^T [C]]^{-1} [C]^T [D]. \quad (6-84)$$

Параметрические значения u и w для каждой исходной точки поверхности получаются с помощью аппроксимации длины хорды (см. разд. 5-11). Конкретнее, для r исходных точек значение параметра в l -й точке в параметрическом направлении u равно

$$u_1 = 0 \quad \frac{u_l}{u_{\max}} = \frac{\sum_{g=2}^l |D_{g,s} - D_{g-1,s}|}{\sum_{g=2}^r |D_{g,s} - D_{g-1,s}|}.$$

Аналогично для s исходных точек в параметрическом направлении w

$$w_1 = 0 \quad \frac{w_l}{w_{\max}} = \frac{\sum_{g=2}^l |D_{r,g} - D_{r,g-1}|}{\sum_{g=2}^s |D_{r,g} - D_{r,g-1}|},$$

где u_{\max} и w_{\max} являются максимальными значениями соответствующих узловых векторов. На рис. 6-50а показаны исходные данные и задающая полигональная

сетка, сгенерированная с помощью данного метода. На рис. 6-50б изображена В-сплайн поверхность, созданная по этой сетке.

Как было ранее отмечено для интерполяции с помощью В-сплайн кривой (см. разд. 5-11), ни данный, ни указанный в [6-5] методы не могут привести к появлению точек или линий складок (разрывов первой или второй производной) в получаемой поверхности.

Отметим из рис. 6-50, что вершины задающей полигональной сетки, полученные из уравнения (6-84), расположены произвольным образом в пространстве. Это неудобно, если в дальнейшем поверхность предполагается модифицировать. В работе [6-27] разработан итерационный метод, основанный на параметрических значениях u , w , в результате которого вершины сетки располагаются на плоскостях или вдоль кривых в трехмерном пространстве.

6-14 РАЗБИЕНИЕ В-СПЛайн ПОВЕРХНОСТЕЙ

В-сплайн поверхность разбивается с помощью отдельного разбиения каждой линии задающей полигональной сетки в одном или обоих параметрических направлениях. Может использоваться любой из методов разбиения В-сплайн кривой (см. разд. 5-12). Лучше всего продемонстрировать это на примере.

Пример 6-16 Разбиение незамкнутой В-сплайн поверхности

Рассмотрим незамкнутую В-сплайн поверхность, заданную полигональной сеткой размера 4×4 :

$$\begin{array}{llll} B_{1,1}[-15 & 0 & 15] & B_{2,1}[-5 & 5 & 15] & B_{3,1}[5 & 5 & 15] & B_{4,1}[15 & 0 & 15], \\ B_{1,2}[-15 & 5 & 5] & B_{2,2}[-5 & 10 & 5] & B_{3,2}[5 & 10 & 5] & B_{4,2}[15 & 5 & 5], \\ B_{1,3}[-15 & 5 & -5] & B_{2,3}[-5 & 10 & -5] & B_{3,3}[5 & 10 & -5] & B_{4,3}[15 & 5 & -5], \\ B_{1,4}[-15 & 0 & -15] & B_{2,4}[-5 & 5 & -15] & B_{3,4}[5 & 5 & -15] & B_{4,4}[15 & 0 & -15]. \end{array}$$

Это поверхность четвертого порядка в обоих параметрических направлениях ($k = l = 4$), состоящая из одного куска с диапазонами параметров $0 \leq u \leq 1$, $0 \leq w \leq 1$. Требуется разбить поверхность на 4 подкуска. Следует сохранять однородный незамкнутый узловой вектор.

Вспомним узловой вектор $[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$; для обоих параметрических направлений, найденных в разд. 5-12 и примере 5-18, репараметризуем этот вектор в $[0 \ 0 \ 0 \ 0 \ 2 \ 2 \ 2 \ 2]$, тогда поверхность разбивается с помощью вставки узлового значения 1 в интервале $0 \rightarrow 2$. Таким образом, новый узловой вектор задается координатами $[0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$. Применив уравнения (5-119) и (5-120) к каждой линии сетки в обоих направлениях, получим сетку для разбитой на 4 куска поверхности. Например, рассмотрим разбиение линии сетки в направлении w , заданной $B_{1,j}$, $1 \leq j \leq 4$. Здесь только

$$\alpha'_{4,1} = \alpha'_{4,2} = \alpha'_{4,3} = \alpha'_{4,4} = \alpha'_{4,5} = 1$$

не равны нулю.

Воспользовавшись уравнениями (5-120), получим

$$\begin{aligned} \alpha_{3,1}^2 = \alpha_{3,2}^2 = \alpha_{3,3}^2 = \alpha_{3,5}^2 = 1; \quad \alpha_{3,4}^2 = \alpha_{4,4}^2 = 1/2, \\ \alpha_{2,1}^3 = \alpha_{2,2}^3 = \alpha_{3,5}^3 = 1; \quad \alpha_{2,3}^3 = \alpha_{3,3}^3 = \alpha_{3,4}^3 = \alpha_{4,4}^3 = 1/2, \\ \alpha_{1,1}^4 = \alpha_{3,5}^4 = 1; \quad \alpha_{1,2}^4 = \alpha_{2,2}^4 = \alpha_{2,3}^4 = \alpha_{3,3}^4 = \alpha_{3,4}^4 = \alpha_{4,4}^4 = 1/2. \end{aligned}$$

Тогда из уравнения (5-119) получим вершины нового характеристического многогранника. В частности,

$$\begin{aligned} C_{1,1} &= \alpha_{1,1}^4 B_{1,1} + \alpha_{2,1}^4 B_{1,2} + \alpha_{3,1}^4 B_{1,3} + \alpha_{4,1}^4 B_{1,4} \\ &= B_{1,1} = [-15 \ 0 \ 15] \end{aligned}$$

$$\begin{aligned} C_{1,2} &= \alpha_{1,2}^4 B_{1,1} + \alpha_{2,2}^4 B_{1,2} + \alpha_{3,2}^4 B_{1,3} + \alpha_{4,2}^4 B_{1,4} \\ &= \frac{1}{2}(B_{1,1} + B_{1,2}) = [-15 \ 2.5 \ 10] \end{aligned}$$

$$\begin{aligned} C_{1,3} &= \alpha_{1,3}^4 B_{1,1} + \alpha_{2,3}^4 B_{1,2} + \alpha_{3,3}^4 B_{1,3} + \alpha_{4,3}^4 B_{1,4} \\ &= \frac{1}{2}(B_{1,2} + B_{1,3}) = [-15 \ 5 \ 0] \end{aligned}$$

$$\begin{aligned} C_{1,4} &= \alpha_{1,4}^4 B_{1,1} + \alpha_{2,4}^4 B_{1,2} + \alpha_{3,4}^4 B_{1,3} + \alpha_{4,4}^4 B_{1,4} \\ &= \frac{1}{2}(B_{1,3} + B_{1,4}) = [-15 \ 2.5 \ -10] \end{aligned}$$

$$\begin{aligned} C_{1,5} &= \alpha_{1,5}^4 B_{1,1} + \alpha_{2,5}^4 B_{1,2} + \alpha_{3,5}^4 B_{1,3} + \alpha_{4,5}^4 B_{1,4} \\ &= B_{1,4} = [-15 \ 0 \ -15] \end{aligned}$$

Выполнив ту же операцию для каждой линии сетки в направлении w , мы получим задающую полигональную сетку размера 4×5 для поверхности, состоящей из двух подкусков в направлении w и одного в направлении u .

$$\begin{array}{llll} C_{1,1}[-15 \ 0 \ 15] & C_{2,1}[-5 \ 5 \ 15] & C_{3,1}[5 \ 5 \ 15] & C_{4,1}[15 \ 0 \ 15] \\ C_{1,2}[-15 \ 2.5 \ 10] & C_{2,2}[-5 \ 7.5 \ 10] & C_{3,2}[5 \ 7.5 \ 10] & C_{4,2}[15 \ 2.5 \ 10] \\ C_{1,3}[-15 \ 5 \ 0] & C_{2,3}[-5 \ 10 \ 0] & C_{3,3}[5 \ 10 \ 0] & C_{4,3}[15 \ 5 \ 0] \\ C_{1,4}[-15 \ 2.5 \ -10] & C_{2,4}[-5 \ 7.5 \ -10] & C_{3,4}[5 \ 7.5 \ -10] & C_{4,4}[15 \ 2.5 \ -10] \\ C_{1,5}[-15 \ 0 \ -15] & C_{2,5}[-5 \ 5 \ -15] & C_{3,5}[5 \ 5 \ -15] & C_{4,5}[15 \ 0 \ -15] \end{array}$$

Элементы $\alpha_{i,j}^k$, приведенные выше, применяются также при разбиении поверхности в направлении u . В этом случае задающая полигональная сетка размера 5×4 такова:

$$\begin{array}{lll} C_{1,1}[-15 \ 0 \ 15] & C_{2,1}[-10 \ 2.5 \ 15] \\ C_{1,2}[-15 \ 5 \ 5] & C_{2,2}[-10 \ 7.5 \ 5] \\ C_{1,3}[-15 \ 5 \ -5] & C_{2,3}[-10 \ 7.5 \ -5] \\ C_{1,4}[-15 \ 0 \ -15] & C_{2,4}[-10 \ 2.5 \ -15] \end{array}$$

$$\begin{array}{lll} C_{3,1}[0 \ 5 \ 15] & C_{4,1}[10 \ 2.5 \ 15] & C_{5,1}[15 \ 0 \ 15] \\ C_{3,2}[0 \ 10 \ 5] & C_{4,2}[10 \ 7.5 \ 5] & C_{5,2}[15 \ 5 \ 5] \\ C_{3,3}[0 \ 10 \ -5] & C_{4,3}[10 \ 7.5 \ -5] & C_{5,3}[15 \ 5 \ -5] \\ C_{3,4}[0 \ 5 \ -15] & C_{4,4}[10 \ 2.5 \ -15] & C_{5,4}[15 \ 0 \ -15] \end{array}$$

Здесь поверхность состоит из двух подкусков в направлении u и одного в направлении w .

Разбиение поверхности как в u , так и в w направлениях приведет в результате к задающей полигональной сетке размера 5×5 , заданной в виде

$$\begin{array}{lll} C_{1,1}[-15 \ 0 \ 15] & C_{2,1}[-10 \ 2.5 \ 15] \\ C_{1,2}[-15 \ 2.5 \ 10] & C_{2,2}[-10 \ 5 \ 10] \\ C_{1,3}[-15 \ 5 \ 0] & C_{2,3}[-10 \ 7.5 \ 0] \\ C_{1,4}[-15 \ 2.5 \ -10] & C_{2,4}[-10 \ 5 \ -10] \\ C_{1,5}[-15 \ 0 \ -15] & C_{2,5}[-10 \ 2.5 \ -15] \end{array}$$

$$\begin{array}{lll} C_{3,1}[0 \ 5 \ 15] & C_{4,1}[10 \ 2.5 \ 15] & C_{5,1}[15 \ 0 \ 15] \\ C_{3,2}[0 \ 7.5 \ 10] & C_{4,2}[10 \ 5 \ 10] & C_{5,2}[15 \ 2.5 \ 10] \\ C_{3,3}[0 \ 10 \ 0] & C_{4,3}[10 \ 7.5 \ 0] & C_{5,3}[15 \ 5 \ 0] \\ C_{3,4}[0 \ 7.5 \ -10] & C_{4,4}[10 \ 5 \ -10] & C_{5,4}[15 \ 2.5 \ -10] \\ C_{3,5}[0 \ 5 \ -15] & C_{4,5}[10 \ 2.5 \ -15] & C_{5,5}[15 \ 0 \ -15] \end{array}$$

Заметим, что последняя сетка получается из первой или второй с помощью уравнений (5-119) и (5-120).

Исходная поверхность и все три полученные при ее разбиении сетки изображены на рис. 6-51. Каждая из поверхностей идентична исходной.

Ясно, что при дальнейшем разбиении поверхности задающая полигональная сетка приближается к поверхности.

6-15 ГАУССОВА КРИВИЗНА И КАЧЕСТВО ПОВЕРХНОСТИ

В автоматизированном проектировании большой интерес представляет разработка соответствующих методов для определения и/или визуализации качества или гладкости поверхностей. Хорошо известно, что используемые обычно бикубические поверхности (Кунса, Безье или В-сплайн), хотя и являются во всех точках C^2 гладкими, в некоторых местах могут быть плоскими или выпуклыми либо волнистыми. В настоящее время самые лучшие математические методы определения качества поверхности используют Эйлеровы (ортогональные) сетки минимальной и максимальной кривизны (см. [6-28] и [6-29]) и гауссовой кривизны (см. [6-28] – [6-32], и разд. 6-8).

Вспомним (разд. 6-8), что две комбинации главных кривизн, называемые средней и гауссовой (общей) кривизнами, характеризуют локальную форму поверхности. Средняя кривизна определяется как

$$\kappa_a = \frac{\kappa_{\min} + \kappa_{\max}}{2}. \quad (6-45)$$

Гауссова кривизна определяется как

$$\kappa_g = \kappa_{\min} \cdot \kappa_{\max}, \quad (6-46)$$

где κ_{\min} и κ_{\max} являются главными кривизнами. Гауссова кривизна в точке на поверхности показывает, является ли поверхность локально эллиптической, гиперболической или параболической (гауссова кривизна положительна, отрицательна или равна нулю).

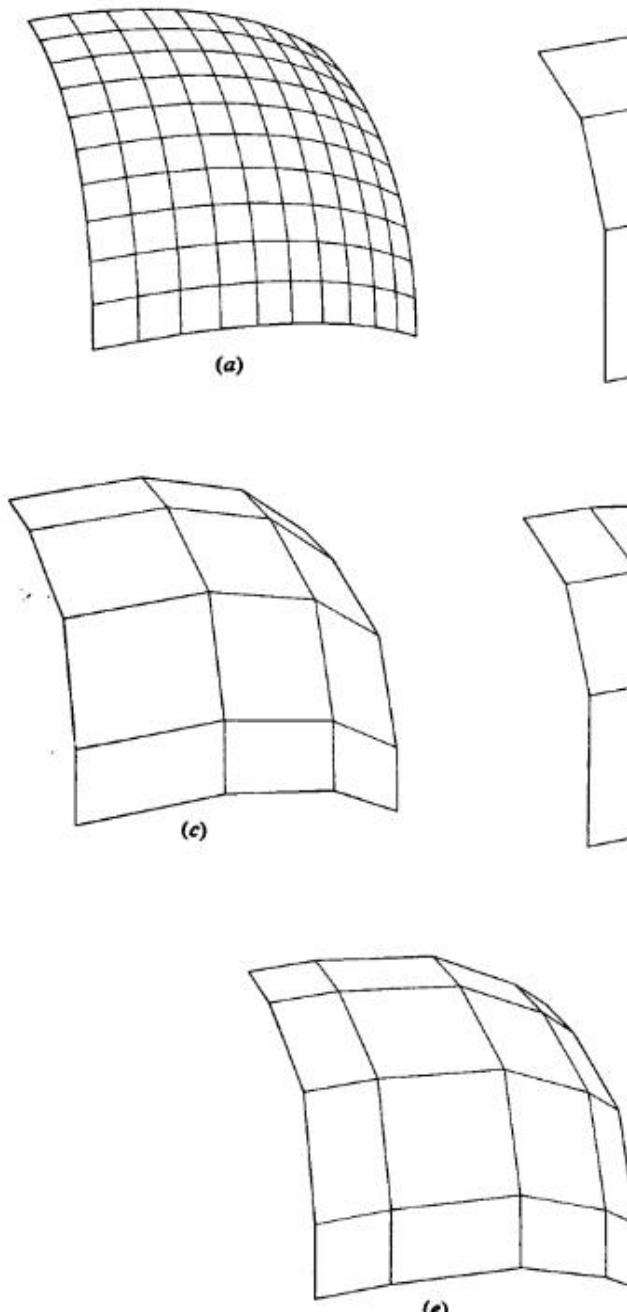


Рис. 6-51 Разбиение В-сплайн поверхности. (а) Поверхность; (б) исходная задающая полигональная сетка; (с) сетка, разбитая в *w* направлении; (д) сетка, разбитая в *u* направлении; (е) сетка, разбитая в обоих направлениях.

Интересно отметить здесь, что если гауссова кривизна равна нулю, то поверхность является развертывающейся, т. е. она может быть развернута в плоскость. Такая поверхность изогнута в одном направлении, например, конус или цилиндр. Это подразумевает, что одна из главных кривизн, κ_{\min} и κ_{\max} равна нулю. Следовательно, равна нулю и гауссова кривизна.

Существуют несколько методов визуализации средней и гауссовой кривизн поверхности. Если изобразительные возможности ограничены рисованием отрезков, то наиболее полезны контурные чертежи (см. [6-28] и [6-29]). В работах [6-30] и [6-32] показано, что эффективным методом этого является кодирование гауссовой кривизны на растровом изображении с помощью цветов или набора полутона серого цвета.

На рис. 6-52 показаны закодированные¹ с помощью полутона серого цвета изображения гауссовой кривизны для нескольких тестовых поверхностей вместе с соответствующим характеристическим многогранником (слева) и проволочным параметрическим представлением поверхностей (по середине). Все поверхности являются бикубическими ($k = l = 4$) В-сплайн поверхностями. На рис. 6-52 изображены три поверхности по мере увеличения степени нарушения их гладкости. На рис. 6-52а представлена совершенно гладкая поверхность, без изъянов. Два ярко выраженных гребня уменьшенной гладкости на рис. 6-52б вызваны совпадением на каждой стороне трех линий полигональной сетки. На рис. 6-52с удлиненная линия «складки» в середине поверхности получилась из-за совпадения трех линий полигональной сетки на участке, пересекающем несколько внутренних линий сетки, что показано на характеристическом многограннике.

Вообще, изображения закодированной гауссовой кривизны более наглядно показывают свойства поверхностей. Например, рисунки 6-52а и б демонстрируют большое отрицательное значение кривизны в угловых точках. Эта отрицательная кривизна является результатом ограничений на границы поверхности — они должны быть прямыми и плоскими, тогда как внутренняя область выпукла и положительно изогнута. Закодированное изображение гауссовой кривизны на рис. 6-52б подчеркивает уплощение области, расположенной между гребнями. Отметим, что так как гауссова кривизна равна нулю в этой области, то эта часть поверхности является развертывающейся. Отметим также, что задающая полигональная сетка в этой области является развертывающейся. И наконец, полоса поперек середины закодированного изображения на рис. 6-52с показывает, что поверхность в этой области представляет собой плоскость, согнутую посередине. Тот факт, что сгиб является прямой линией, объясняет исчезновение гауссовой кривизны вдоль этой линии.

¹ Эти изображения являются монохромными копиями цветных изображений из [6-33]. При кодировании изображений гауссовой кривизны были усреднены значения кривизн в четырех вершинах многоугольников, образуемых густой сеткой аппроксимирующих поверхность четырехугольников. Эта усредненная величина была присвоена каждому многоугольнику. Диапазон изменения кривизны был разделен на ряд одинаковых интервалов (за исключением концов), соответствующих доступному диапазону интенсивностей. Справа от изображения указан список полутона и соответствующих им значений кривизны. Ступенчатость (границы между различными интенсивностями) возникает из-за ограниченного числа доступных интенсивностей, а не из-за аппроксимации с помощью многоугольников.

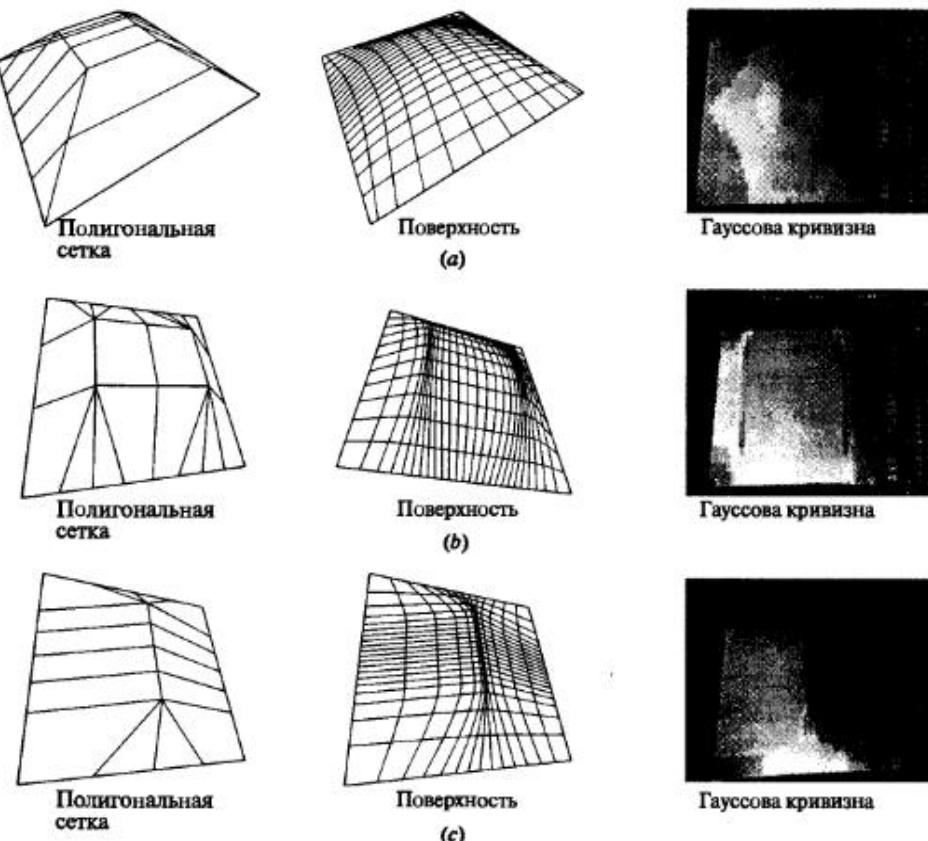


Рис. 6-52 Гауссова кривизна. (a) Гладкая поверхность; (b) короткая линия «складки»; (c) более длинная линия «складки». (С разрешения Дж. Дилла и Д. Роджерса.)

Метод вычисления гауссовой кривизны будет проиллюстрирован на примере.

Пример 6-17 Гауссова кривизна

Найти гауссову кривизну в точке с параметрами $u = 1/2, w = 1$ для незамкнутой В-сплайн поверхности, определенной ранее в примере 6-15.

Вспомним сначала базисные функции $N_{i,4}$ и $M_{j,1}$ из примера 6-15. По этим данным можно вычислить первую и вторую производные, необходимые для нахождения Q_u, Q_w, Q_{uw}, Q_{uu} и Q_{ww} , и последующего вычисления гауссовой кривизны. А именно

$$\begin{aligned} N_{1,4} &= (1-u)^3 & N'_{1,4} &= -3(1-u)^2 & N''_{1,4} &= 6(1-u) \\ N_{2,4} &= 3u(1-u)^2 & N'_{2,4} &= 3(1-u)(1-3u) & N''_{2,4} &= 6(3u-2) \\ N_{3,4} &= 3u^2(1-u) & N'_{3,4} &= 3u(2-3u) & N''_{3,4} &= 6(1-3u) \\ N_{4,4} &= u^3 & N'_{4,4} &= 3u^2 & N''_{4,4} &= 6u \end{aligned}$$

и

$$\begin{aligned} M_{1,3} &= 0 & M'_{1,3} &= 0 & M''_{1,3} &= 0 \\ M_{2,3} &= \frac{(2-w)^2}{2} & M'_{2,3} &= w-2 & M''_{2,3} &= 1 \\ M_{3,3} &= \frac{(2-w)(3w-2)}{2} & M'_{3,3} &= 4-3w & M''_{3,3} &= -3 \\ M_{4,3} &= (w-1)^2 & M'_{4,3} &= 2(w-1) & M''_{4,3} &= 2 \end{aligned}$$

Вычисление производных в точке $u = 1/2, w = 1$ и подстановка в уравнения (6-71) – (6-75) приводят к следующим результатам:

$$\begin{aligned} Q(1/2, 1) &= [0 \quad 35/4 \quad 0], \\ Q_u(1/2, 1) &= [30 \quad 0 \quad 0], \\ Q_w(1/2, 1) &= [0 \quad 0 \quad 10], \\ Q_{uw}(1/2, 1) &= [0 \quad 0 \quad 0], \\ Q_{uu}(1/2, 1) &= [0 \quad -30 \quad 0], \\ Q_{ww}(1/2, 1) &= [0 \quad -10 \quad 10]. \end{aligned}$$

Компоненты уравнения (6-48) для гауссовой кривизны таковы:

$$\begin{aligned} Q_u \times Q_w &= [30 \quad 0 \quad 0] \times [0 \quad 0 \quad 10] = [0 \quad -300 \quad 0] \\ |Q_u \times Q_w|^4 &= (300)^4 \\ A &= [Q_u \times Q_w] \cdot Q_{uu} = [0 \quad -300 \quad 0] \cdot [0 \quad -30 \quad 0] = 9000 \\ B &= [Q_u \times Q_w] \cdot Q_{uw} = [0 \quad -300 \quad 0] \cdot [0 \quad 0 \quad 0] = 0 \\ C &= [Q_u \times Q_w] \cdot Q_{ww} = [0 \quad -300 \quad 0] \cdot [0 \quad -10 \quad 10] = 3000 \end{aligned}$$

Используя уравнение (6-48), получим гауссову кривизну

$$\kappa_g = \frac{AC - B}{|Q_u \times Q_w|^4} = \frac{(9000)(3000) - (0)}{(300)^4} = 3.33 \times 10^{-3}$$

Так как $\kappa_g > 0$, то поверхность является локально эллиптической.

6-16 РАЦИОНАЛЬНЫЕ B-СПЛАН ПОВЕРХНОСТИ

Так же как и для рациональных кривых, возможно существование рациональных форм квадратичных поверхностей, бикубических поверхностей Кунса и поверхностей Безье. Однако здесь будут рассмотрены только рациональные B-сплайн поверхности, как из-за нехватки места, так и в связи с тем, что такие поверхности являются обобщением всех этих форм.

Декартово произведение рациональной B-сплайн поверхности в четырехмерном пространстве однородных координат задается формулой

$$Q(u, w) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j}^h N_{i,k}(u) M_{j,l}(w), \quad (6-85)$$

где $B_{i,j}^h$ являются 4D однородными вершинами характеристического многогранника, а $N_{i,k}(u)$ и $M_{j,l}(w)$ являются нерациональными B-сплайн базисными функциями, приведенными ранее в уравнении (5-84).

Проектирование обратно в трехмерное пространство с помощью деления на однородную координату дает рациональную В-сплайн поверхность

$$Q(u, w) = \frac{\sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N_{i,k}(u) M_{j,l}(w)}{\sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N_{i,k}(u) M_{j,l}(w)} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} B_{i,j} S_{i,j}(u, w), \quad (6-86)$$

где $B_{i,j}^h$ являются 3D-точками задающей полигональной сетки и $S_{i,j}(u, w)$ – базисными функциями от двух переменных рациональной В-сплайн поверхности

$$S_{i,j}(u, w) = \frac{h_{i,j} N_{i,k}(u) M_{j,l}(w)}{\sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N_{i,k}(u) M_{j,l}(w)}. \quad (6-87)$$

Удобно положить $h_{i,j} \geq 0$ для всех i, j .

Важно отметить здесь, что $S_{i,j}(u, w)$ не являются произведением $R_{i,k}(w)$ и $R_{j,l}(w)$ (см. уравнение 5-123). Тем не менее $S_{i,j}(u, w)$ имеют форму и аналитические свойства, похожие на функцию произведения $N_{i,k}(u) M_{j,l}(w)$. Следовательно, рациональные В-сплайн поверхности имеют аналитические и геометрические свойства, похожие на их нерациональные двойники. Конкретнее,

Сумма базисных функций рациональной поверхности для любых значений u, w равна

$$\sum_{i=1}^{n+1} \sum_{j=1}^{m+1} S_{i,j}(u, w) \equiv 1. \quad (6-88)$$

Каждая базисная функция рациональной поверхности положительна или равна нулю для всех значений параметров u, w , т. е. $S_{i,j} \geq 0$.

Кроме случая $k = l = 1$, каждая базисная функция рациональной поверхности имеет ровно один максимум.

Максимальный порядок рациональной В-сплайн поверхности в каждом параметрическом направлении равен числу вершин характеристического многогранника в этом направлении.

Рациональная В-сплайн поверхность порядка k, l (степени $k - 1, l - 1$) гладкая во всех точках C^{k-2}, C^{l-2} .

Рациональная В-сплайн поверхность инвариантна относительно проективного преобразования; т. е. любое проективное преобразование может быть применено к поверхности путем его применения к задающей полигональной сетке. Отметим, что это условие является более строгим, чем для нерациональной В-сплайн поверхности.

Поверхность лежит внутри выпуклой оболочки задающей полигональной сетки, образуемой объединением всех выпуклых оболочек k, l соседних вершин полигональной сетки.

Свойство затухания изменений для рациональных В-сплайн поверхностей неизвестно.

Влияние одной вершины полигональной сетки ограничивается $\pm k/2, \pm l/2$ интервалами в каждом параметрическом направлении.

При триангуляции задающая полигональная сетка образует плоскую аппроксимацию поверхности.

Если число вершин задающей полигональной сетки равно порядку в каждом параметрическом направлении и дублированных внутренних узловых величин нет, то рациональная В-сплайн поверхность является рациональной поверхностью Безье.

Из уравнений (6-86) и (6-87) ясно, что когда все $h_{i,j} = 1$, тогда $S_{i,j}(u, w) = N_{i,k}(u) M_{j,l}(w)$. Таким образом, базисные функции рациональной В-сплайн поверхности и сами поверхности превращаются в их нерациональные эквиваленты. Следовательно, рациональные В-сплайн поверхности представляют соответственно обобщение нерациональных В-сплайн поверхностей и рациональных и нерациональных поверхностей Безье.

Снова, как и в случае рациональных В-сплайн кривых, алгоритмы для увеличения степени, разбиения (см. разд. 6-14) и интерполяции (см. разд. 6-13) нерациональных В-сплайн поверхностей можно применять просто путем их использования для 4D-вершин задающей полигональной сетки.

Незамкнутый однородный, периодический однородный и неоднородный узловые векторы могут быть использованы для генерации рациональных В-сплайн базисных функций и рациональных В-сплайн поверхностей. Типы узловых векторов могут смешиваться. Например, в параметрическом u -направлении может использоваться незамкнутый однородный узловый вектор, а в w -направлении – неоднородный узловый вектор. Сначала мы остановимся на незамкнутых однородных узловых векторах.

На рис. 6-53 представлены бикубическая ($k = l = 4$) рациональная В-сплайн поверхность и ее задающая полигональная сетка для $h_{1,3} = h_{2,3} = 0, 1, 5$. Изображение на рис. 6-53c с $h_{1,3} = h_{2,3} = 1$ идентично нерациональной В-сплайн поверхности. Результаты варьирования значений однородной координаты можно заметить, сравнивая рис. 6-53c с рис. 6-53b и d. Эти результаты аналогичны полученным для рациональных В-сплайн кривых (см. разд. 5-13), но проявляются не в столь сильной степени. В данном случае уменьшение эффекта обусловлено тем, что $S_{i,j}(u, w)$ является функцией смешения от двух переменных.

На рис. 6-54a и b иллюстрируется результат, полученный при присвоении всем внутренним $h_{i,j}$ значений 0 и 500, соответственно; т. е. $h_{2,2} = h_{2,3} = h_{3,2} = h_{3,3} = h_{4,2} = h_{4,3} = 0$ или 500. Все остальные $h_{i,j} = 1$. Характеристический многогранник изображен на рис. 6-53a. Установка всех внутренних $h_{i,j} = 0$ приводит к игнорированию внутренних вершин характеристического многогранника. Интерполируются только вершины граничных кривых. Напротив, присвоение всем внутренним $h_{i,j} = 500$ сводит к минимуму влияние граничных вершин. Заметим, что изменение $h_{i,j}$ влияет на параметризацию поверхности. Это иллюстрируется скоплением параметрических линий около границ поверхности, когда внутренние $h_{i,j} = 0$ (см. рис. 6-54a), и во внутренней части поверхности, когда внутренние $h_{i,j} = 500$ (см. рис. 6-54b).

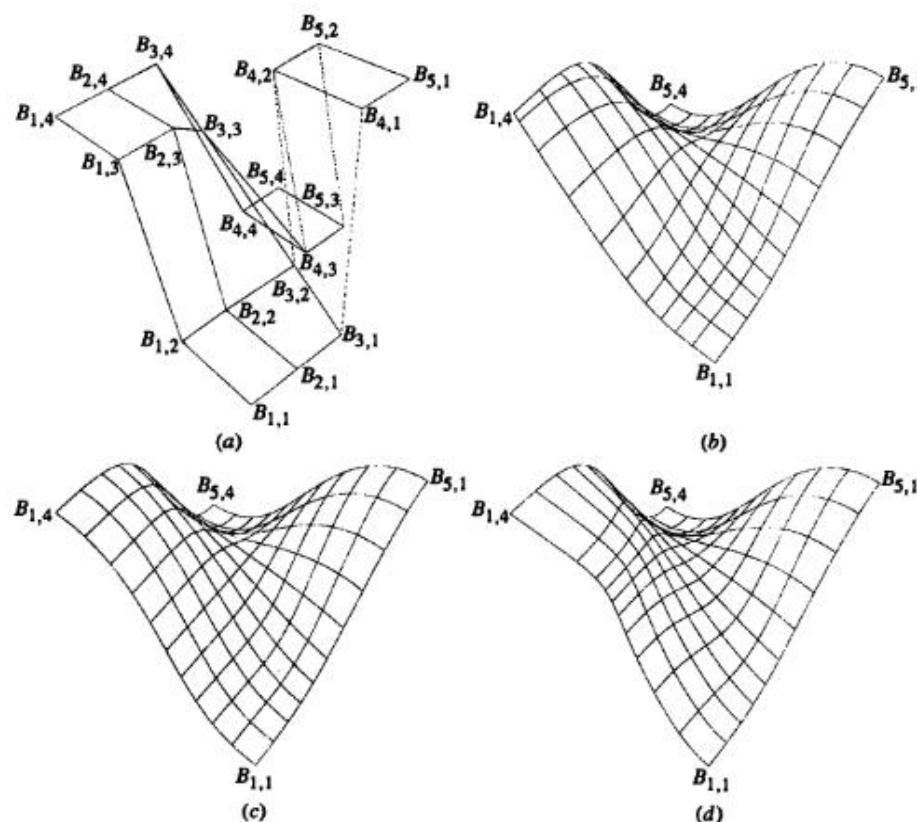


Рис. 6-53 Рациональные В-сплайн поверхности с $n+1 = 5$, $m+1 = 4$, $k = l = 4$.
(a) Характеристический многогранник; (b) $h_{1,3} = h_{2,3} = 0$; (c) $h_{1,3} = h_{2,3} = 1$; (d) $h_{1,3} = h_{2,3} = 5$.

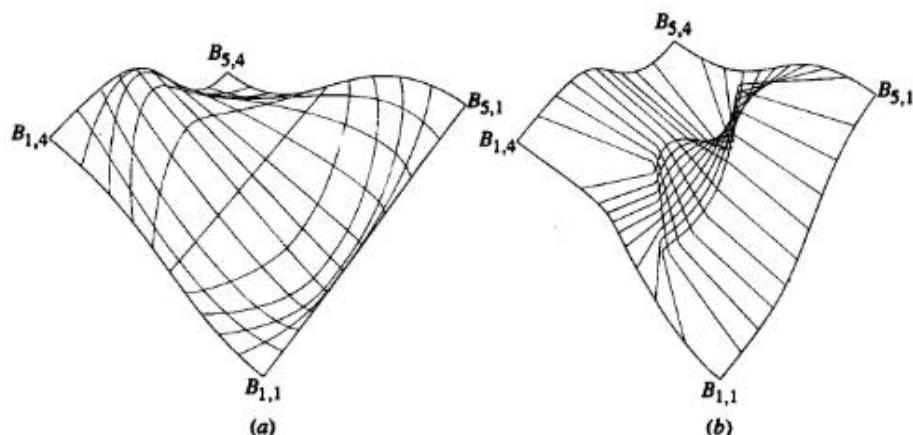


Рис. 6-54 Рациональные В-сплайн поверхности с $n+1 = 5$, $m+1 = 4$, $k+l = 4$. (a) Все внутренние $h_{i,j} - e = 0$; (b) все внутренние $h_{i,j} - e = 500$.

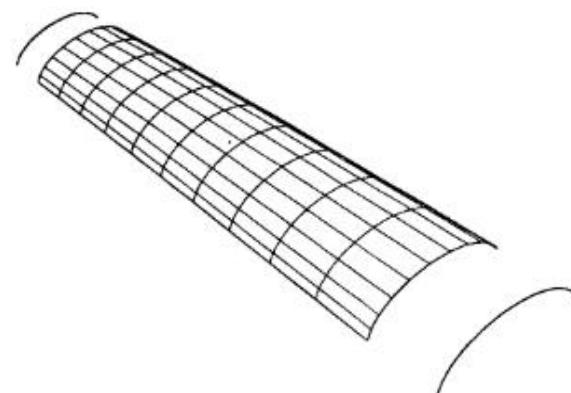


Рис. 6-55 Рациональный В-сплайн эллиптический цилиндр, сгенерированный заметанием рациональной эллиптической кривой на рис. 5-67б.

Результаты, возникающие при совпадении нескольких вершин или линий сетки аналогичны результатам, полученным для нерациональных В-сплайн поверхностей (см. разд. 6-12) и рациональных В-сплайн кривых (см. разд. 5-13). Также аналогичны результаты смещения одной вершины на поверхности.

Одним из самых привлекательных свойств рациональных В-сплайн поверхностей является их способность представлять квадратичные поверхности и плавно переходить в скульптурные поверхности высоких степеней. В качестве простого примера квадратичной поверхности рассмотрим обобщенный цилиндр, образуемый заметанием кривой. Ясно, что в направлении заметания такая поверхность должна быть второго порядка, т. е. прямой линией. Следовательно, для заметаемой в направлении u поверхности формула выглядит (см. [6-33]) следующим образом:

$$Q(u, w) = \sum_{i=1}^2 \sum_{j=1}^{m+1} B_{i,j} S_{i,j}(u, w), \quad (6-89)$$

где $S_{i,j}(u, w)$ имеет в параметрическом направлении w тот же порядок, что и кривая, и порядок 2 в направлении u . Кроме того, вершины характеристического многогранника в направлении u равны $B_{1,j} = B_j$ и $B_{2,j} = B_j + sD$, где D задает направление и расстояние заметания. Параметр s изменяется в диапазоне $0 \leq s \leq 1$. Элементы B_j являются вершинами задающего многоугольника для заметающей кривой. Однородные координаты остаются неизменными в направлении заметания; т. е. $h_{1,j} = h_{2,j} = h_j$, где h_j – это однородная координата для заметаемой кривой. На рис. 6-55 изображен эллиптический цилиндр, образованный с использованием эллиптической кривой, представленной на рис. 5-67б. С каждой стороны поверхности показана отодвинутая заметаемая кривая.

Рациональные В-сплайн поверхности используются также для создания линейчатых поверхностей. Эллиптический цилиндр, изображенный на рис. 6-55, является, конечно, линейчатой поверхностью. Для генерации с помощью рацио-

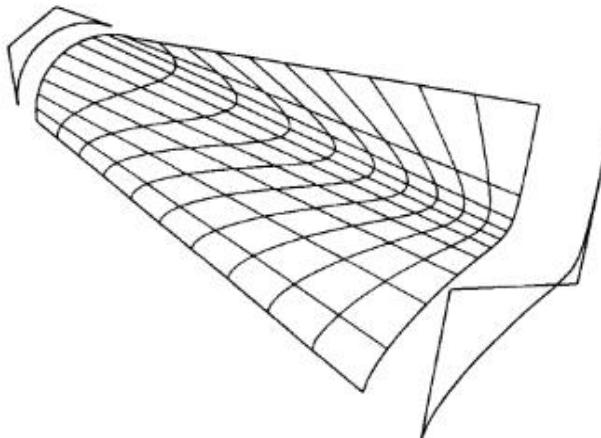


Рис. 6-56 Рациональная В-сплайн линейчатая поверхность.

нальных В-сплайнов линейчатой поверхности более общего вида требуется, чтобы обе кривые имели одинаковый порядок (степень), одинаковый узловый вектор и одинаковое число вершин задающего многоугольника. Если кривые имеют разный порядок (степень), то степень кривой меньшего порядка увеличивают (см. разд. 5-8 и пример 6-18). Требуемый узловый вектор является объединением узловых векторов двух кривых. Любые кратные узловые значения для любой кривой включаются в окончательный узловый вектор. Для обеспечения идентичности обоих узловых векторов используется вставка узлов (см. разд. 5-12). Увеличение степени и вставка узлов обеспечивают равенство числа вершин характеристического многоугольника для обеих кривых. Получающаяся рациональная В-сплайн линейчатая поверхность описывается уравнением (6-89) с

$$P_1(w) = Q(0, w) = \sum_{j=1}^{m+1} B_{1,j} R_{j,l}(w) \quad \text{и} \quad P_2(w) = \sum_{j=1}^{m+1} B_{2,j} R_{j,l}(w) = Q(1, w).$$

На рис. 6-56 показан пример линейчатой поверхности, переводящей четверть окружности в рациональную В-сплайн кривую четвертого порядка. Кривые и их характеристические многоугольники изображены отодвинутыми от края поверхности. Данный метод лучше проиллюстрировать на примере.

Пример 6-18 Рациональные В-сплайн линейчатые поверхности

Найти точку с параметрами $u = w = 0.5$ на линейчатой поверхности, образованной смещением 120° дуги окружности и рациональной кривой четвертого порядка. Дуга окружности представляется рациональной В-сплайн кривой, заданной $B_{1,1}[0 \ 0 \ 0]$, $B_{1,2}[1 \ \sqrt{3} \ 0]$, $B_{1,3}[2 \ 0 \ 0]$ и $[H] = [1 \ 1/2 \ 1]$. Вторая кривая определяется $B_{2,1}[0 \ 0 \ 10]$, $B_{2,2}[1 \ 1 \ 10]$, $B_{2,3}[2 \ 0 \ 10]$, $B_{2,4}[3 \ 1 \ 10]$ с $[H] = [1 \ 3/4 \ 5 \ 1]$.

Сначала необходимо увеличить степень дуги окружности. В действительности дуга является рациональной кривой Безье. Обсуждавшийся в разд. 5-8

метод увеличения степени в рациональном случае применяется к $4D$ однородным координатам. В результате получаем:

$$\begin{aligned} B_1^{h*} &= B_1^h, \\ B_i^{h*} &= \alpha_i B_{i-1}^h + (1 - \alpha_i) B_i^h \quad \alpha_i = \frac{i}{n+1} \quad i = 2, \dots, n, \\ B_{n+1}^{h*} &= B_n^h. \end{aligned}$$

После проецирования обратно в трехмерное пространство имеем:

$$\begin{aligned} B_1^* &= B_1, \\ B_i^* &= \frac{\alpha_i h_{i-1} B_{i-1} + (1 - \alpha_i) h_i B_i}{\alpha_i h_{i-1} + (1 - \alpha_i) h_i} \quad i = 2, \dots, n, \\ h_i^* &= \alpha_i h_{i-1} + (1 - \alpha_i) h_i. \end{aligned}$$

Использование данных результатов для увеличения степени 120° дуги приведет к

$$\begin{aligned} h_{1,1}^* &= h_{1,1} = 1, \\ B_{1,1}^* &= B_{1,1} = [0 \ 0 \ 0], \\ h_{1,2}^* &= \left(\frac{1}{3}\right)(1) + \left(\frac{2}{3}\right)\left(\frac{1}{2}\right) = \frac{2}{3}, \\ B_{1,2}^* &= \left\{ \frac{1}{3}(1)[0 \ 0 \ 0] + \frac{2}{3}\left(\frac{1}{2}\right)[1 \ \sqrt{3} \ 0] \right\} / \left(\frac{2}{3}\right) = \left[\frac{1}{2} \ \frac{\sqrt{3}}{2} \ 0\right], \\ h_{1,3}^* &= \left(\frac{2}{3}\right)\left(\frac{1}{2}\right) + \left(\frac{1}{3}\right)(1) = \frac{2}{3}, \\ B_{1,3}^* &= \left\{ \left(\frac{2}{3}\right)\left(\frac{1}{2}\right)[1 \ \sqrt{3} \ 0] + \frac{1}{3}(1)[2 \ 0 \ 0] \right\} / \left(\frac{2}{3}\right) = \left[\frac{3}{2} \ \frac{\sqrt{3}}{2} \ 0\right], \\ h_{1,4}^* &= h_{1,3} = 1, \\ B_{1,4}^* &= B_{1,3} = [2 \ 0 \ 0]. \end{aligned}$$

Теперь каждая кривая имеет четыре вершины характеристического многоугольника. Узловой вектор для каждой кривой равен $[X] = [Y] = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$. Следовательно, включение узлов применять не нужно.

Для $u = w = 0.5$ из уравнения (5-84) получим

$$\begin{aligned} N_{1,2} &= 0.5; & N_{2,2} &= 0.5 \\ M_{1,4} &= 0.125; & M_{2,4} &= 0.375; & M_{3,4} &= 0.375; & M_{4,4} &= 0.125 \end{aligned}$$

Тогда уравнение (6-87) приведет к следующим результатам:

$$\begin{aligned} S_{1,1} &= 0.0396 & S_{1,2} &= 0.0792 & S_{1,3} &= 0.0792 & S_{1,4} &= 0.0396 \\ S_{2,1} &= 0.0396 & S_{2,2} &= 0.0891 & S_{2,3} &= 0.594 & S_{2,4} &= 0.0396 \end{aligned}$$

Координаты точки на поверхности

$$Q(0.5, 0.5) = [1.634 \ 0.266 \ 7.624].$$

Вся поверхность изображена на рис. 6-56.

С помощью В-сплайнов могут быть также представлены поверхности вращения. Предположим, что

$$P(w) = \sum_{j=1}^{m+1} B_j B_{j,l}(w)$$

с узловым вектором $[Y]$ является рациональной В-сплайн кривой, и вспомним, что полная окружность получается при объединении четырех четвертей окружности, заданных девятью вершинами многоугольника (см. разд. 5-13). Это приводит нас к рациональной В-сплайн поверхности вращения, задаваемой (см. [6-33])

$$Q(u, w) = \sum_{i=1}^9 \sum_{j=1}^{m+1} B_{i,j} S_{i,j}(u, w), \quad (6-90)$$

где узловый вектор $[X] = [0 \ 0 \ 0 \ 1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 4 \ 4 \ 4]$. Предположим, что вращение происходит вокруг оси z , кривая $P(w)$ определена в плоскости xz , элементы $B_{i,j}$ задаются как $B_{1,j} = B_j$ для фиксированного j с $1 \leq j \leq 9$. Вершины характеристического многоугольника образуют угловые и средние точки квадрата, лежащего в плоскости, перпендикулярной оси z , и с размером стороны, равным удвоенному радиусу окружности вращения. Однородные весовые множители являются произведением множителей, необходимых для задания рациональной В-сплайн кривой и для задания окружности вращения. Конкретнее, для фиксированного j , $h_{1,j} = h_j$, $h_{2,j} = h_j\sqrt{2}/2$, $h_{3,j} = h_j$, $h_{4,j} = h_j\sqrt{2}/2, \dots, h_{9,j} = h_j$. На рис. 6-57 показаны характеристический многоугольник и сама кривая для окружности вращения и для вращаемой рациональной В-сплайн кривой. Также на рисунке приведены объединенный характеристический многогранник поверхности и сама поверхность.

На рис. 6-58 и 6-59 представлены обычные поверхности вращения — тор и сфера, вместе со своими характеристическими многогранниками. Тор генерируется путем вращения смещенной окружности вокруг одной из осей. Сфера генерируется путем вращения полуокружности, составленной из двух 90° дуг вокруг оси, проходящей через диаметр полуокружности.

Как было ранее упомянуто, одной из наиболее мощных характеристик рациональных в отличие от иррациональных В-сплайн поверхностей является их способность «упрятывать» (или включать) квадратичные элементы поверхности внутри обобщенной скульптурной поверхности. Например, как часть более общей поверхности, может быть включен цилиндрический элемент. На рис. 6-60 представлено три примера. Центральной частью каждой поверхности четвертого порядка является секция кругового цилиндра. Рис. 6-60a мог бы изображать переднюю кромку крыла или лопасти¹ турбины, рис. 6-60b — цилиндрический нос корабля. Обе поверхности генерируются с помощью задания дуги окружности третьего порядка (см. разд. 5-13), увеличения степени дуги (см. пример 6-18), создания из дуг линейчатой поверхности и ее включения между двумя крайними элементами поверхностей четвертого порядка. Случайно обе поверхности, показанные

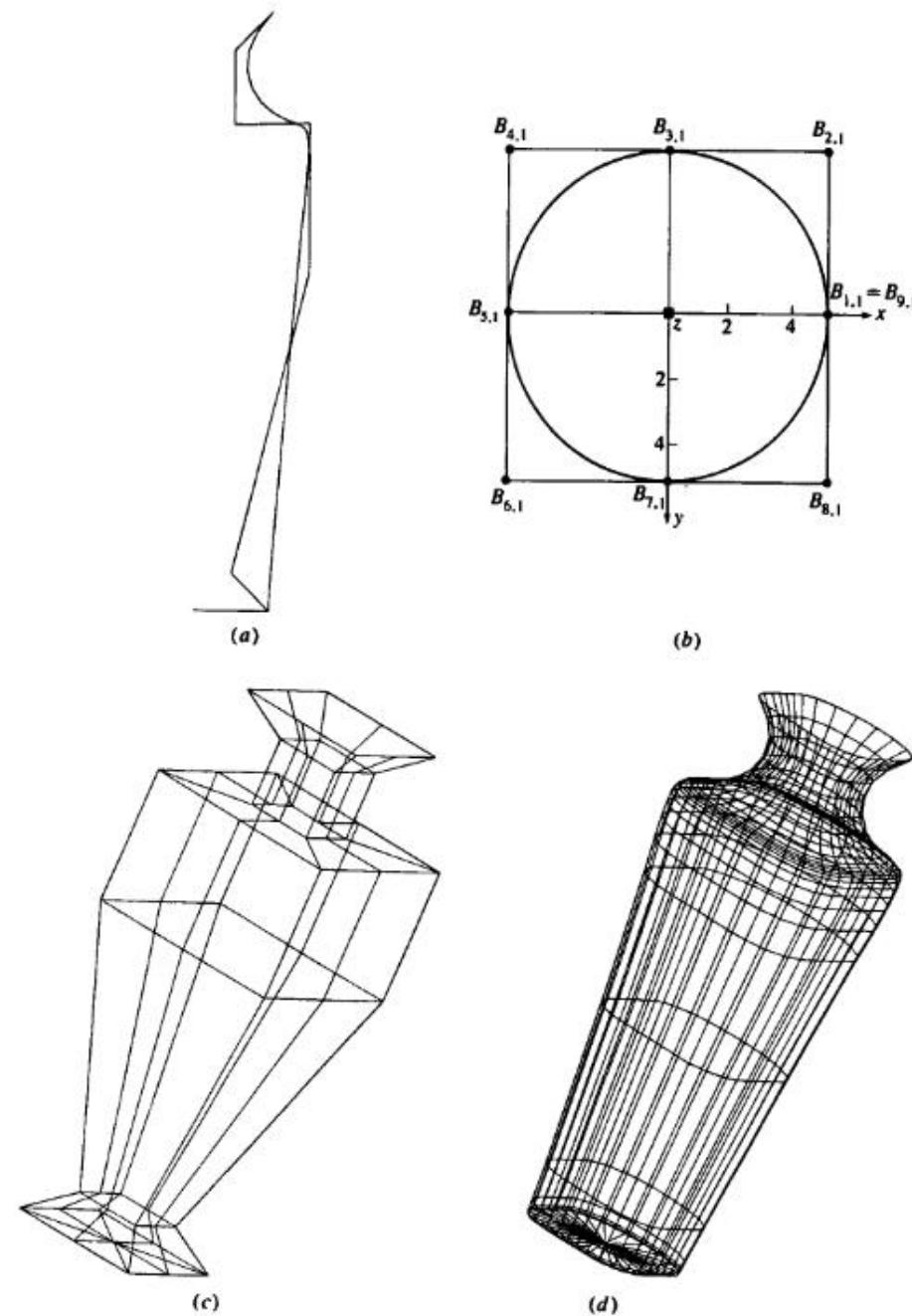


Рис. 6-57 Рациональная В-сплайн поверхность вращения. (a) Образующая кривая и задающая сетка; (b) окружность вращения; (c) характеристический многогранник; (d) поверхность вращения.

¹ В аэродинамических профилях NASA для задания передней кромки используется дуга окружности.

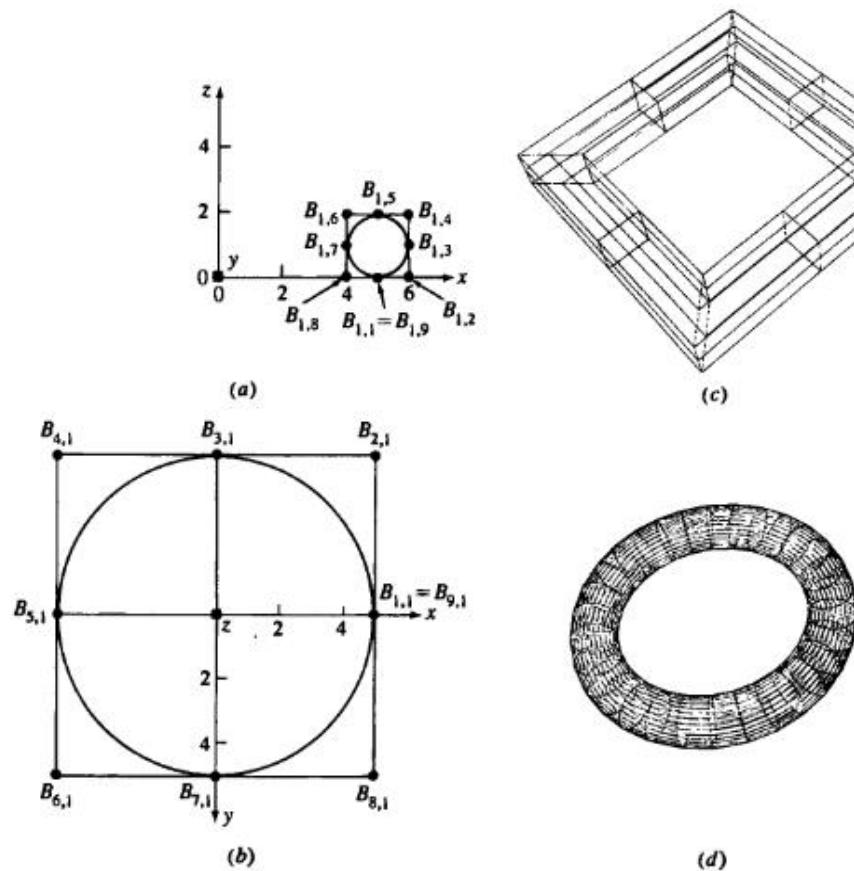


Рис. 6-58 Тор, сгенерированный как рациональная В-сплайн поверхность. (а) Смещенная окружность и задающий многоугольник; (б) окружность вращения и задающий многоугольник; (с) характеристический многогранник для тора; (д) тор.

на рис. 6-60 a и b , оказались линейчатыми развертывающимися поверхностями. На рис. 6-60 c изображен цилиндрический элемент, упрятанный в более общую поверхность.

Производные рациональной В-сплайн поверхности получаются с помощью формального дифференцирования уравнения (6-86):

$$Q_u = \frac{\bar{N}}{D} \left(\frac{\bar{N}_u}{\bar{N}} - \frac{\bar{D}_u}{\bar{D}} \right), \quad (6-91a)$$

$$Q_w = \frac{\bar{N}}{D} \left(\frac{\bar{N}_w}{\bar{N}} - \frac{\bar{D}_w}{\bar{D}} \right), \quad (6-91b)$$

$$Q_{uw} = \frac{\bar{N}}{D} \left(\frac{\bar{N}_{uw}}{\bar{N}} - \frac{\bar{N}_u \bar{D}_w}{\bar{N} \bar{D}} - \frac{\bar{N}_w \bar{D}_u}{\bar{N} \bar{D}} + 2 \frac{\bar{D}_u \bar{D}_w}{\bar{D} \bar{D}} - \frac{\bar{D}_{uw}}{\bar{D}} \right), \quad (6-91c)$$

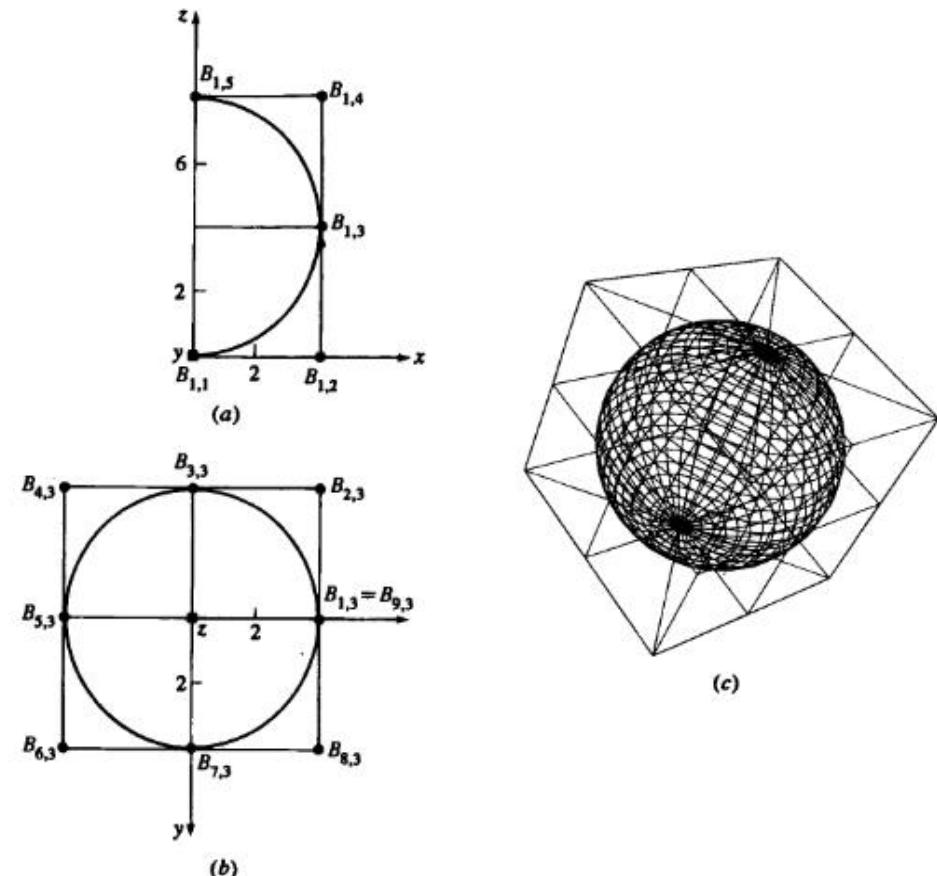


Рис. 6-59 Сфера, сгенерированная как рациональная В-сплайн поверхность. (а) Смещенная окружность и задающий многоугольник; (б) окружность вращения и задающий многоугольник; (с) характеристический многогранник и сфера.

$$Q_{uu} = \frac{\bar{N}}{\bar{D}} \left(\frac{\bar{N}_{uu}}{\bar{N}} - 2 \frac{\bar{N}_u}{\bar{N}} \frac{\bar{D}_u}{\bar{D}} + 2 \frac{\bar{D}_u^2}{\bar{D}^2} - \frac{\bar{D}_{uu}}{\bar{D}} \right), \quad (6-91d)$$

$$Q_{ww} = \frac{\bar{N}}{\bar{D}} \left(\frac{\bar{N}_{ww}}{\bar{N}} - 2 \frac{\bar{N}_w}{\bar{N}} \frac{\bar{D}_w}{\bar{D}} + 2 \frac{\bar{D}_w^2}{\bar{D}^2} - \frac{\bar{D}_{ww}}{\bar{D}} \right), \quad (6-91e)$$

где \bar{N} и \bar{D} являются числителем и знаменателем, соответственно, уравнения (6-86) с производными

$$\bar{N}_u = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N'_{i,k}(u) M_{j,l}(w),$$

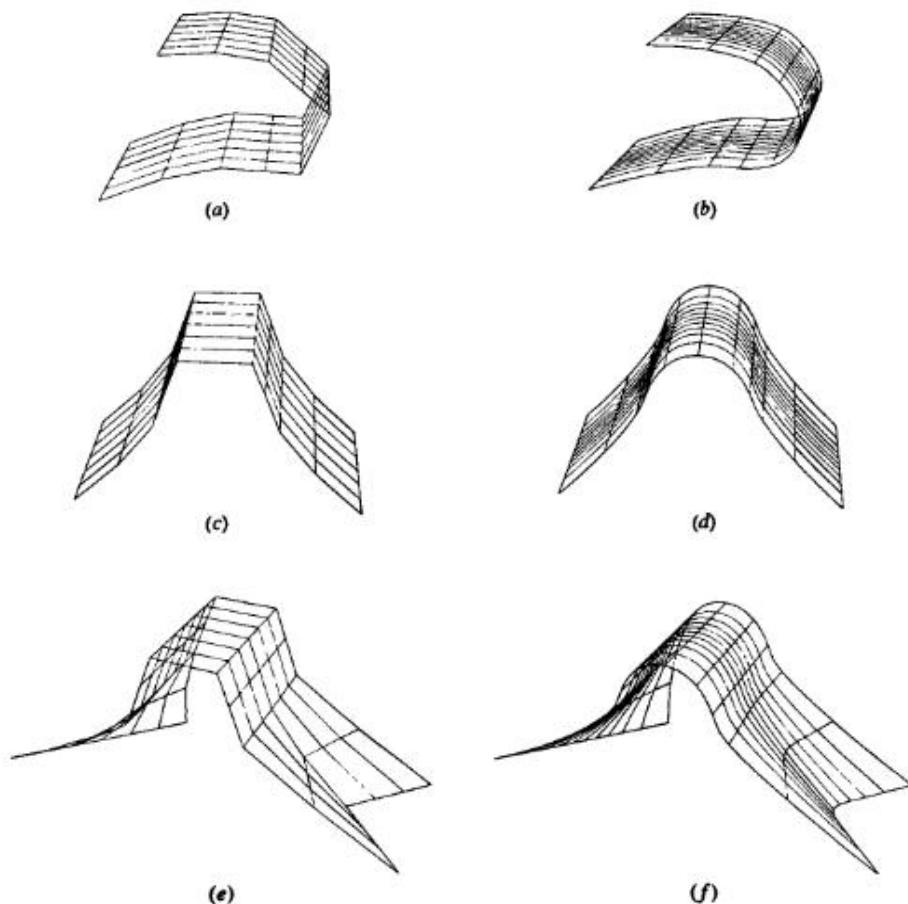


Рис. 6-60 Элемент квадратичной поверхности внутри более общей рациональной В-сплайн поверхности. Передняя кромка крыла. (a) Характеристический многогранник; (b) поверхность. Корма корабля. (c) Характеристический многогранник; (d) поверхность. Цилиндр как часть более общей поверхности. (e) Характеристический многогранник; (f) поверхность.

$$\bar{N}_w = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N_{i,k}(u) M'_{j,l}(w),$$

$$\bar{N}_{uw} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N'_{i,k}(u) M'_{j,l}(w),$$

$$\bar{N}_{uu} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N''_{i,k}(u) M_{j,l}(w),$$

$$\bar{N}_{ww} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} B_{i,j} N_{i,k}(u) M''_{j,l}(w),$$

$$\bar{D}_u = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N'_{i,k}(u) M_{j,l}(w),$$

$$\bar{D}_w = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N_{i,k}(u) M'_{j,l}(w),$$

$$\bar{D}_{uw} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N'_{i,k}(u) M'_{j,l}(w),$$

$$\bar{D}_{uu} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N''_{i,k}(u) M_{j,l}(w),$$

$$\bar{D}_{ww} = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} h_{i,j} N_{i,k}(u) M''_{j,l}(w).$$

Штрих обозначает производную относительно соответствующего параметра. Функции $N'_{i,k}(u) M'_{j,l}(w)$, $N''_{i,k}(u)$, $M''_{j,l}(w)$ задаются уравнениями (5-97)–(5-100).

Эти производные полезны при определении гауссовой кривизны поверхности (см. разд. 6-15), а также других свойств поверхности.

6-17 ЛИТЕРАТУРА

- 6-1 Rogers, D. F., *Procedural Elements for Computer Graphics*, McGraw-Hill, New York, 1985. (См. [1-1] на с. 74.)
- 6-2 Bézier P. E., *Emploi des Machines à Commande Numérique*, Masson et Cie, Paris, France, 1970; Bézier, P. E., «Example of an Existing System in the Motor Industry: The Unisurf System,» *Proc. Roy. Soc. (London)*, Vol. A321, pp. 207–218, 1971.
- 6-3 Sabin, M. A., «An Existing System in the Aircraft Industry. The British Aircraft Corporation Numerical Master Geometry System,» *Proc. Roy. Soc. (London)*, Vol. A321, pp. 197–205, 1971.
- 6-4 Peters, C. J., «Interactive Computer Graphics Application of the Bicubic Parametric Surface to Engineering Design Problems,» McDonnell Douglas Automation Company, St. Louis, Missouri, presented at SIAM 1973 National Meeting, Hampton, Va., 18–21 June 1973.
- 6-5 Rogers D. F., and Satterfield S. D., «Dynamic B-Spline Surfaces,» *Proc. of the Fourth International Conference on Computer Applications in the Automation of Shipyard Operation and Ship Design (ICCAS 82)*, 7–10 June 1982, Annapolis, Maryland, pp. 189–196, North Holland, 1982.
- 6-6 Rogers, D. F., and Satterfield S. G., «B-Spline Surfaces for Ship Hull Design,» *Comp. Graph.*, Vol. 14, pp. 211–217, 1980, (SIGGRAPH 80).
- 6-7 Rogers, D. F., Rodriguez, F., and Satterfield, S. G., «Computer Aided Ship Design and the Numerically Controlled Production of Towing Tank Models,» *Proc. of 16th Des. Auto. Conf.*, San Diego, California, 24–27 June 1979.
- 6-8 Cohen, E., «Some Mathematical Tools for a Modeler's Workbench,» *IEEE Comp. Graph. & Appl.*, Vol. 3, pp. 63–66, October 1983.
- 6-9 Tan, S. T., Yuen, M. F., and Hui, K. C., «Modeling Solids with Sweep Primitives,» *Comp. Mech. Engg.*, Vol. 6, pp. 60–72, 1987.
- 6-10 Coquillart, S., «A Control-Point-Based Seeping Technique,» *IEEE Comp. Graph. & Appl.*, Vol. 7, pp. 36–45, November 1987.

- 6-11 Patton, D., Dept. of Mech. and Prod. Engr., Royal Melbourne Inst. of Tech., Melbourne, Australia, private communication, 1988.
- 6-12 Dresden, A., *Solid Analytic Geometry and Determinants*, Dover Publications, New York, 1964.
- 6-13 Sommerville, D. M. Y., *Analytical Geometry of Three Dimensions*, Cambridge Univ. Press, 1959.
- 6-14 Snyder, V., and Sisam, C. H., *Analytic Geometry of Space*, Henry Holt and Co., New York, 1914.
- 6-15 Goldman, R. N., «Two Approaches to a Computer Model for Quadric Surfaces,» *IEEE Comp. Graph. & Appl.*, Vol. 3, pp. 21–24, September 1983.
- 6-16 Wilson, P. R., «Conic Representation for Shape Description,» *IEEE Comp. Graph. & Appl.*, Vol. 7, pp. 23–30, 1987.
- 6-17 Miller, J. R., «Geometric Approaches to Non-planar Quadric Surfaces Intersection Curves,» *ACM Trans. on Graph.*, Vol 6, pp. 274–307, 1987.
- 6-18 Dill, J. C., «An Application of Color Graphics to the Display of Surface Curvature,» *Comp. Graph.*, Vol. 15, pp. 153–161, 1981, (SIGGRAPH 81).
- 6-19 Coons, S., «Surfaces for Computer-Aided Design of Space Forms,» M. I. T. Proj. MAC, MAC-TR-41, June 1967. (Also as AD 663 504).
- 6-20 Ferguson, J. C., «Multivariable Curve Interpolation,» *J. ACM*, Vol. 2, pp. 221–228, April 1964.
- 6-21 Barnhill, R. E., «Smooth Interpolation over Triangles,» in *Computer Aided Geometric Design*, Barnhill, R. E., and Riesenfeld, R. F. (eds.), Academic Press, New York, pp. 45–70, 1974.
- 6-22 Barnhill, R. E., «Coons Patches,» *Comp. in Ind.*, Vol. 3, pp. 37–43, 1982.
- 6-23 Charrot, P., and Gregory, J., «A Pentagonal Surface Patch for Computer Aided Geometric Design,» *Comp. Aid. Geom. Des.*, Vol. 1, pp. 87–94, 1984.
- 6-24 Faux, I. D., and Pratt, M. J., *Computational Geometry for Design and Manufacture*, Ellis Horwood (John Wiley & Sons), 1979.
- 6-25 Bézier, P. E., *The Mathematical Basis of the Unisurf CAD System*, Butterworth, London, 1986.
- 6-26 Barsky, B. A., and Greenberg, D. P., «Determining a Set of B-spline Control Vertices to Generate an Interpolating Surface,» *Comp. Graph. Imag. Proc.*, Vol. 14, pp. 203–226, 1980.
- 6-27 Rogers, D. F., and Fog, N. G., «Constrained B-spline Curve and Surface Fitting,» *CADJ*, Vol. 21, pp. 641–648, 1989.
- 6-28 Munchmeyer, F. C., Schubert, C., and Nowacki, H., «Interactive Design of Fair Hull Surfaces Using B-splines,» *Proc. of the Third International Conference on Computer Applications in the Automation of Shipyard Operation and Ship Design (ICCAS 79)*, 18–21 June 1979, University of Strathclyde, Glasgow, Scotland, pp. 67–76, North Holland, 1979.
- 6-29 Munchmeyer, F. C., «The Gaussian Curvature of Coons Biquintic Patches,» *Proc. ASME Century 2 Inter. Comp. Tech. Conf.*, 12–15 August 1980.
- 6-30 Dill, J. C., «An Application of Color Graphics to the Display of Surface Curvature,» *Comp. Graph.*, Vol. 15, pp. 153–161, 1981, (SIGGRAPH 81).
- 6-31 Forrest, A. R., «On the Rendering of Surfaces,» *Comp. Graph.*, Vol. 13, pp. 253–259, 1979, (SIGGRAPH 79).
- 6-32 Dill, J. C., and Rogers, D. F., «Color Graphics and Ship Hull Surface Curvature,» *Proc. of the Fourth International Conference on Computer Applications in the Automation of Shipyard Operation and Ship Design (ICCAS 82)*, pp. 197–205, North Holland, 1982.
- 6-33 Piegl, L., and Tiller, W., «Curve and Surface Constructions Using Rational B-splines,» *Comp. Aid. Des.*, Vol. 19, pp. 485–498, 1987.

Приложение А

ГРАФИЧЕСКОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

А-1 ВВЕДЕНИЕ

Разрабатывая концепции графического программного обеспечения, удобно мыслить в терминах схемы, показанной стрелками на рис. 1-63. Конвейер начинается с прикладной программы и соответствующей ей базы данных, проходит через пакет машинной графики и соответствующую ему графическую базу данных и далее через соответствующий программный интерфейс внешнего устройства, для генерации изображения на физическом графическом устройстве, с которым пользователь может или не может взаимодействовать. На протяжении такого конвейера существуют два критических программных интерфейса: интерфейс прикладной программы и интерфейс внешнего устройства. В настоящее время предпринимаются попытки привести оба интерфейса в соответствие с международными стандартами.

Интерфейс прикладного программиста обычно оформляется как пакет подпрограмм или функций, доступных из языка высокого уровня, такого, как Фортран, Си, Бейсик или Паскаль. Интерфейс пользователя обычно пишется с помощью подпрограмм и функций интерфейса прикладного программиста, который в настоящее время представлен четырьмя основными стандартами Графической корневой системой (the Graphical Kernel System GKS) [A-1–A-3], GKS-3D [A-4] и Иерархической интерактивной графической системой программиста (the Programmer's Hierarchical Interactive Graphics System PHIGS) [A-5, A-6] и PHIGS+ [A-7]. Многие программы и функции, предусмотренные для интерфейса прикладного программиста, являются основным предметом данной книги и работы [A-8]. Примеры: поворот, перенос, перспективные проекции, отсечения, удаление невидимых линий и невидимых поверхностей.

Интерфейс внешнего устройства работает на гораздо более низком уровне, чем интерфейс прикладного программиста. Это связано с генерацией кодов (двоичных битовых карт), требуемых для управления конкретным физическим или виртуальным графическим устройством. Обычно системные программисты и разработчики интерфейса прикладной программы имеют дело с интерфейсом устройства. Примеры стандартов интерфейса устройства: Графический метафайл компьютера (the Computer Graphics Metafile CGM) [A-9]–[A-11], Графический интерфейс компьютера (the Computer Graphics Interface CGI) [A-10]–[A-12] и Интерфейс виртуального устройства (the Virtual Device Interface VDI) [A-13].

Детальное обсуждение различных стандартов выходит за рамки данной книги и проводится в цитируемой литературе. Здесь дается только краткое общее описание.

GKS – двумерный стандарт визуализации, не включает моделирования, предусмотрены только двумерные примитивы, такие, как ломаная, поворот, перенос, масштабирование. Трехмерные преобразования должны выполняться за пределами стандарта, т. е. трехмерный объект преобразуется в двумерную проекцию (см. гл. 3) и результат отображается с использованием программ двумерной графики GKS. GKS поддерживает широкий спектр устройств ввода/вывода (см. гл. 1), таких, как монохроматические и цветные ЭЛТ-дисплеи, принтеры, графопостроители, мыши, планшеты, устройства для ввода скалярных значений, функциональные переключатели и т. д. Доступ к программам GKS обеспечен из ряда языков высокого уровня: Фортран, Паскаль, Ада, Си, GKS – официальный стандарт как Международной организации по Стандартам (ISO), так и Американского национального института стандартов (ANSI) [A-1].

GKS-3D – расширение GKS, обеспечивающее возможности отображения и определения трехмерных объектов. В нем также нет функций моделирования. Включает невидимые линии и невидимые поверхности, а также параллельные и перспективное проецирование. В настоящее время стандарт GKS-3D продолжает дорабатываться [A-4].

В отличие от GKS и GKS-3D, стандарт PHIGS был изначально разработан как двойной стандарт для трехмерного манипулирования и визуализации и для трехмерного геометрического моделирования. Особо выделяется поддержка высокоинтерактивного программного обеспечения, такого, как САПР/АСТПП, молекулярное моделирование, моделирование сплошных тел, а также имитация и управление. PHIGS позволяет строить, преобразовывать, модифицировать и хранить трехмерные геометрические модели. Стандарт PHIGS продолжает дорабатываться [A-5].

PHIGS+ – расширение PHIGS, включает модели освещения, обработку полутонов, глубину, а также кривые и поверхности, такие, как рациональные В-сплайны (см. разд. 5-13 и 6-16). PHIGS специально разработан применительно к возможностям высокоинтерактивных графических рабочих станций. Стандарт, например включает представление таких эффектов, как раскраска по методу Гуро и Фонга, а также новые геометрические атрибуты, например нормали вершин.

CGI обеспечивает функциональное преобразование независимой от аппаратуры информации, генерируемой стандартом интерфейса прикладного программиста, в аппаратно зависимые данные, требуемые для управления конкретным графическим устройством. CGI определяет виртуальный интерфейс устройства, т. е. идеализированное стандартное графическое устройство, которое поддерживает ввод от пользователя, генерирует графическую информацию и манипулирует ею. Существует строго регламентированное взаимодействие между GKS и CGI.

CGM перехватывает и хранит информацию на уровне CGI. Результатом является форматированный файл, содержащий «картинку», который может храниться на диске или ленте и передаваться по телефонным линиям или компьютерным сетям.

Прежде чем обсуждать каждый из стандартов во всех деталях и сравнивать их, полезно получить представление об их функциональной и концептуальной структуре.

A-2 ГРАФИЧЕСКИЕ ПРИМИТИВЫ МАШИННОЙ ГРАФИКИ

При создании любой системы графического программного обеспечения физическое (виртуальное) графическое устройство управляется первичными аппаратно зависимыми командами нижнего уровня (примитивами). Графический примитив выполняет единичное действие. Графические примитивы обычно распадаются на три группы: управление курсором, управление состоянием устройства и графический ввод. Здесь понятия курсора и состояния используются в наиболее общем смысле. Из этих графических примитивов формируются различные графические команды (множественные действия), называемые элементарными графическими функциями, они рассматриваются в разд. A-3. Графические элементарные функции аппаратно независимы.

Существуют два графических примитива управления курсором:

переместить курсор – MOVE (x, y);
включить/выключить курсор ONOFF (i).

Функции этих примитивов не нуждаются в комментариях.

Графические примитивы управления устройством более многочисленны и разнообразны. Они получили общее имя SET (установить). Некоторые примеры:

установить алфавитно-цифровой режим;
установить графический режим;
установить режим абсолютных координат;
установить режим относительных координат;
установить цвет (красный, зеленый, синий);
очистить экран
и т. д.

В принципе существует только один графический примитив SET с различными аргументами. Перед тем как продолжить, кратко охарактеризуем функции каждого из этих различных состояний устройства.

Алфавитно-цифровой и графический режимы, как правило, относятся к устройствам, которые могут интерпретировать поток данных как в виде алфавитно-цифровых (ASCII) символов, так и в виде графических команд. Режимы абсолютных и относительных координат определяют, будут ли декартовы координаты, объявленные в графических примитивах, интерпретироваться как абсолютные или относительные. Команда установки цвета определяет цвет курсора для последующих команд, таких, как линия, точка и т. д. Команда очистки экрана перекрашивает экран дисплея в цвет фона.

Примитив графического ввода имеет общее имя GET (задать)
GET (значение).

Из сказанного ясно, что в принципе существуют только четыре графических примитива. В общем виде они могут быть представлены следующим образом:

ONOFF (управляющая переменная);
MOVE (коорд., коорд.);
SET (упр. переменная, (арг.1), (арг.2), ...);
GET (значение).

Здесь угловые скобки () используются для обозначения необязательных аргументов.

A-3 ЭЛЕМЕНТАРНЫЕ ГРАФИЧЕСКИЕ ФУНКЦИИ

Графические примитивы, которые обсуждались в разд. А-2, могут быть объединены в графические элементарные функции. Вообще говоря, это составные операции, т. е. они содержат более одного графического примитива. При аккуратной реализации они не зависят от конкретного физического устройства.

Так же как и графические примитивы, элементарные графические функции делятся на три группы: функции рисования, функции управления устройством или картинкой и функции управления интерактивным вводом. Функции рисования происходят от процесса рисования карандашом или перемещения карандаша по листу бумаги. Карандаш может быть перемещен над листом без рисования. Когда карандаш опускается на лист и перемещается, то он рисует линию. Карандаш может быть перемещен в некоторую позицию, в которой будет поставлена точка. Удобно иметь возможность задавать перемещения карандаша над бумагой и рисования им как в абсолютных так и в относительных координатах. Эти понятия проиллюстрированы шестью функциями рисования, или графическими элементарными функциями:

MOVE ABSOLUTE (x, y);
MOVE RELATIVE (Δ x, Δ y);
DRAW ABSOLUTE (x, y);
DRAW RELATIVE (Δ x, Δ y);
POINT ABSOLUTE (x, y);
POINT RELATIVE (Δ x, Δ y).

Также нужна возможность вывести символ или текст на поверхности рисования, сказав:

TEXT (строка).

Всегда удобно предполагать, что действие начинается от текущей позиции курсора. Функционально каждая из этих графических элементарных функций может быть описана в терминах графических примитивов. Например, алгоритм MOVE ABSOLUTE можно представить следующим образом:

установить режим абсолютных координат;
установить графический режим;

выключить курсор;
переместить курсор в указанную позицию.

Алгоритм POINT ABSOLUTE представляется в следующем виде:

Установить режим абсолютных координат.
Установить графический режим.
Выключить курсор.
Переместить курсор в указанную позицию.
Включить курсор.

Алгоритм DRAW ABSOLUTE:

Установить режим абсолютных координат.
Установить графический режим.
Включить курсор.
Переместить курсор в указанную позицию.

Относительные алгоритмы аналогичны приведенным, но с установкой режима относительных координат командой SET.

Элементарные функции управления устройством или картинкой инициируют устройство и начинают новый фрейм или картинку. Например, для этих целей могли бы служить функции:

Инициировать устройство.
Начать фрейм.

Функционально команда «Инициировать устройство» служит для установки предопределенного состояния устройства. Команда «Начать фрейм» служит для объявления регенерации графического устройства, при которой работа с текущей картинкой завершается, устройство заполняется цветом фона и начинается представление следующей картинки, очищается экран на запоминающей ЭЛТ, устанавливается новый чертеж для перьевых или струйных устройств и т. д.

Управление графическим вводом должно быть более сложным, чем простой графический примитив GET, обсуждавшийся в разд. А-2. Вот некоторые примеры элементарных функций графического ввода:

GET MOUSE (x, y).
GET VALUATOR (значение).
GET TABLET (x, y).
GET FUNCTION SWITCH (i).
и т. д.

Для выполнения этих функций графические примитивы можно объединить. Это иллюстрируют два примера алгоритмов. Первым является алгоритм GET VALUATER, который мог бы иметь следующий вид:

Установить режим абсолютных координат.
Установить графический режим.

Задать (значение).

Алгоритм команды GET MOUSE:

Установить режим абсолютных координат.

Установить графический режим.

Задать (значение x).

Задать (значение y).

Выше ничего не говорилось о типе графического устройства, пикселях и т.д. Читатель может удивиться, что не были включены команды SET PIXEL и GET PIXEL. Однако после минутного размышления он обнаружит, что в контексте графического устройства пиксель — это точка. Следовательно, команды для точек концептуально достаточны для команд пикселов.

Вообще говоря, элементарные графические функции — самый нижний уровень графических команд прикладного программиста. Во многих случаях прикладной программист ничего не знает о графических примитивах, и они могут быть ему фактически недоступны. Однако эти примитивы интересны для системного программиста. Если всю графическую систему представить в виде уровней или оболочек, то графические элементарные функции образуют первый уровень или первую оболочку, а графические примитивы — нулевой уровень или нулевую оболочку (нулевой, потому что они скрыты от прикладного программиста).

A-4 СИСТЕМЫ КООРДИНАТ

Для систем программного обеспечения машинной графики интерес представляют три координатные системы: мировые координаты, или координаты пользователя, псевдопространство с нормализованными координатами устройства и координаты устройства. Мировые координаты служат для описания прикладной области. Они обычно задаются в виде чисел с плавающей точкой. Координаты устройства используются для работы с конкретным графическим устройством: пиксели, адресуемые точки, дюймы, сантиметры и т. д. Они задаются целыми числами. Нормализованное пространство координат устройства называется псевдопространством, используемое для обеспечения независимости от устройства. Нормализованные координаты задаются числами с плавающей точкой из первого квадранта в интервалах $0 \leq ndcx \leq 1.0$, $0 \leq ndcy \leq 1.0$. Таким образом, начало координат находится в левом нижнем углу.

Как показано на рис. A-1 и обсуждалось в гл. 1, описание, манипулирование и отсечение базы данных происходит в мировых координатах. Сформированное таким образом в базе данных «окно» преобразуется в нормализованные координаты, а затем в координаты устройства. Размеры «окна» определяются в мировых координатах. Размеры экранного окна задаются в нормализованных координатах, чтобы обеспечить независимость от устройства. Оба преобразования из мировых в нормализованные координаты и из нормализованных координат в координаты устройства включают только простые перемещения и масштабирование¹.

¹ Предполагается, что преобразования перспективных проекций были выполнены перед отсечением.



Рис. A-1 Преобразования в графическом конвейере.

Пример служит для иллюстрации различных систем координат и преобразований.

Пример A-1 Преобразование координат

Отобразить квадрат с центром в начале координат, со стороной 36 единиц в верхний левый квадрант устройства с разрешением 1024 × 1024.

Чтобы задать некоторое рабочее пространство вокруг квадрата, определим окно со следующими левым, правым, верхним и нижним углами: $wleft = -20$, $wright = 20$, $wbottom = -20$, $wtop = 20$. Это действительно окно, а не изображение, которое определяет преобразование.

В нормализованных координатах верхний левый квадрант определяется так: $vleft = 0$, $vright = 0.5$, $vbottom = 0.5$, $vtop = 1.0$.

Преобразование из мировых координат в нормализованные требует масштабирования оконных координат и затем приведения результата к первому квадранту. Масштабные коэффициенты:

$$xwscale = \frac{1}{wright - wleft} = \frac{1}{20 - (-20)} = \frac{1}{40},$$

$$ywscale = \frac{1}{wtop - wbottom} = \frac{1}{20 - (-20)} = \frac{1}{40}.$$

Коэффициенты преобразования:

$$xwtran = \frac{-wleft}{wright - wleft} = \frac{-(-20)}{20 - (-20)} = \frac{1}{2}.$$

$$ywtran = \frac{-wbottom}{wtop - wbottom} = \frac{-(-20)}{20 - (-20)} = \frac{1}{2}.$$

Результатирующее преобразование плоского окна в нормализованные координаты:

$$\begin{aligned}[T_w] &= [S_w][Tr_w] = \begin{bmatrix} xwscale & 0 & 0 \\ 0 & ywscale & 0 \\ xwtran & ywtran & 1 \end{bmatrix} = \\ &= \begin{bmatrix} 1/40 & 0 & 0 \\ 0 & 1/40 & 0 \\ 1/2 & 1/2 & 1 \end{bmatrix}. \end{aligned}$$

Преобразование из нормализованных координат в координаты устройства вновь включает масштабирование и перенос. Когда координаты устройства описаны в целых числах, требуется особое внимание при определении масштабного и преобразующего множителей.

Хотя разрешение адресуемого в целых числах устройства составляет $xresolution \times yresolution$, адресуемый диапазон целых чисел расположен между 0 и $xresolution - 1$ и между 0 и $yresolution - 1$, т. е. если разрешающая способность равна 1024×1024 , то адресуемый интервал по обеим осям изменяется от 0 до 1023. Обозначив через $xmaxdev$ и $ymaxdev$ максимальные числа соответственно по осям x и y , получим масштабный множитель и величину переноса из нормализованных координат в координаты устройства:

$$\begin{aligned}xwscale &= \text{int}\{(vright - vleft)xmaxdev\} = \text{int}\{(0.5 - 0)(1023)\} = 511, \\ywscale &= \text{int}\{(vtop - vbottom)ymaxdev\} = \text{int}\{(1 - 0.5)(1023)\} = 511, \\xwtran &= \text{int}\{(vleft)(xmaxdev + 1)\} = \text{int}\{(0)(1024)\} = 0, \\ywtran &= \text{int}\{(vbottom)(ymaxdev + 1)\} = \text{int}\{(0.5)(1024)\} = 512.\end{aligned}$$

Результатирующие преобразование нормализованных координат к координатам устройства:

$$[T_v] = [S_v][Tr_v] = \begin{bmatrix} xwscale & 0 & 0 \\ 0 & ywscale & 0 \\ xwtran & ywtran & 1 \end{bmatrix} = \begin{bmatrix} 511 & 0 & 0 \\ 0 & 511 & 0 \\ 0 & 512 & 1 \end{bmatrix}.$$

Использование этих результатов для координат углов квадрата с центром в начале координат дает:

$$\begin{bmatrix} -18 & -18 & 1 \\ 18 & -18 & 1 \\ 18 & 18 & 1 \\ -18 & 18 & 1 \end{bmatrix} \begin{bmatrix} 1/40 & 0 & 0 \\ 0 & 1/40 & 0 \\ 1/2 & 1/2 & 1 \end{bmatrix} = \begin{bmatrix} 1/20 & 1/20 & 1 \\ 19/20 & 1/20 & 1 \\ 19/20 & 19/20 & 1 \\ 1/20 & 19/20 & 1 \end{bmatrix}$$

и

$$\begin{bmatrix} 1/20 & 1/20 & 1 \\ 19/20 & 1/20 & 1 \\ 19/20 & 19/20 & 1 \\ 1/20 & 19/20 & 1 \end{bmatrix} \begin{bmatrix} 511 & 0 & 0 \\ 0 & 511 & 0 \\ 0 & 512 & 1 \end{bmatrix} = \begin{bmatrix} 25.55 & 537.55 & 1 \\ 485.45 & 537.55 & 1 \\ 485.45 & 997.45 & 1 \\ 25.55 & 997.45 & 1 \end{bmatrix} = \\ = \begin{bmatrix} 25 & 537 \\ 485 & 537 \\ 485 & 997 \\ 25 & 997 \end{bmatrix}.$$

Целочисленная функция использована здесь для получения результатирующих целых адресуемых точек.

A-5 ПРОСТОЙ ГРАФИЧЕСКИЙ МЕТАФАЙЛ

К концепции хранения картинки в файле данных мы обращались при обсуждении графических стандартов. Применимым для этого графическим стандартом является Графический метафайл компьютера (CGM). Однако CGM более сложен и детализован, чем это требуется для простых разработок и студенческих работ. Простая альтернатива этому стандарту, называемая здесь стандартным дисплейным файлом, дана в таблице A-1¹.

Первая строка стандартного дисплейного файла служит для идентификации. Каждая последующая строка файла начинается с номера кода операции, за которым следует информация, требуемая для данной операции. Все значения координат определяются как числа с плавающей точкой в нормализованных координатах, т. е. в интервалах $0 \leq x \leq 1,0$, $0 \leq y \leq 1,0$. Все элементы данных разделены запятыми.

Простой стандартный дисплейный файл для квадрата из примера A-1 имеет вид:

```
0, 0.05, 0.05
2, 0.95, 0.05
2, 0.95, 0.05
2, 0.05, 0.95
2, 0.05, 0.05
```

Такой формат настолько прост, что позволяет создавать картинку с помощью текстового редактора. Обычно файл создается путем вызова графической процедуры, образованной из обращений к командам установки либо отмены, например:

```
Set sdf on/off filename.
```

Программы вывода сохраненной в метафайле картинки на некоторое графическое устройство достаточно элементарны. Такой формат допускает доступное расширение за счет включения локальной функциональности.

Возможности нормализованных координат и графического метафайла иллюстрирует рис. А-2. Изображения, включающие символы, были разработаны и отложены на устройстве с запоминающей трубкой. Результатирующая картинка заполнена в стандартном дисплейном файле, который был передан компьютером, используемым как контроллер реального времени для обрабатывающего центра с ЧПУ (рис. 1-1). Программа читает стандартный дисплейный файл и генерирует код для драйвера обрабатывающего центра. Полученный результат показан на рис. А-2.

A-6 ПРОСТОЕ ФОРМАТЫ ГРАФИЧЕСКОГО ВВОДА

В некоторых системах графического программного обеспечения необходимы стандарты форматов файла ввода. Для сложных или очень больших баз данных требуются формализованные структуры данных и систем управления. Однако для

¹ Подобная концепция используется более десятилетия в институте, где работают авторы.

Таблица А-1 Код стандартного дисплейного файла (.sdf)

Код	Операция	Параметры*
0	Абсолютное перемещение	x, y
1	Абсолютная точка	x, y
2	Абсолютное рисование	x, y
3	Относительное перемещение	$\Delta x, \Delta y$
4	Относительная точка	$\Delta x, \Delta y$
5	Относительное рисование	$\Delta x, \Delta y$
6	Текст	Строка текста
7	Установить цвет	красный, зеленый, синий
8	Установить таблицу просмотра	красные биты, зеленые биты, синие биты
9	Установить пиксель	x, y, красный, зеленый, синий
10	Новый фрейм	
11	Пауза	
12	Установить разрешение	xresolution, yresolution
99	Печать сообщения	Строка сообщения

* x, y — числа с плавающей точкой в NDC, т. е. в диапазоне $0 \leq x, y \leq 1.0$. Цвета красный, зеленый, синий являются числами с плавающей точкой в диапазоне $0 \leq$ красный, зеленый, синий ≤ 1.0 ; при этом равенство нулю означает отсутствие интенсивности, а равенство единице — полную интенсивность. Биты красного, зеленого и синего цветов суть степени 2 (см. разд. 1-9, где обсуждаются соответствующие таблицы).

относительно небольших баз данных вполне достаточны простейшие форматы, основанные на файле с последовательным доступом. Приведем два таких формата.

Первый формат файла ввода, называемый «стандартный графический формат» (.sgf) (табл. А.2), лучше всего подходит для простого рисования и для двумерного каркасного представления изображения. Второй, определяемый в табл. А-3, называется «стандартный полигональный формат» и особенно полезен для растровых дисплеев или для описания входной информации для алгоритмов невидимых линий и невидимых поверхностей. Заметим, что когда формат .spf используется для генерации каркасных изображений, каждая линия рису-



Рис. А-2 Гравюра, полученная из стандартного дисплейного файла.

Таблица А-2 Стандартный графический формат (.sgf)

Стандартный графический формат — это очень простой и общеупотребительный формат для описания любых дву- или трехмерных данных, которые могут быть представлены как список пар x, y или троек x, y, z. В файле допускаются пустые строки. Общая форма:

строка описания строка описания

x_1, y_1 x_1, y_1, z_1

x_2, y_2 x_2, y_2, z_2

x_n, y_n x_n, y_n, z_n

1e37, 1e37 1e37, 1e37, 1e37

Разделителем группы пар или троек является 1e37, 1e37 или 1e37, 1e37, 1e37*. Последняя строка файла не должна быть разделителем.

Обычно этот формат интерпретируется следующим образом: графическое устройство, начиная с первой точки группы, рисует все входящие в эту группу точки и соединяющие линии, пока не встречает разделитель.

* Значение разделителя сложилось исторически.

ется дважды. В обоих случаях такие атрибуты представления, как цвет линии или многоугольника, коэффициент отражения многоугольника или тела и т. д., хранятся в отдельной группе файлов.

Оба формата заключают в себе уже готовый алгоритм рисования или черчения.¹ Рисование в соответствии с форматом .sgf:

В абсолютных координатах переместиться к первой точке и затем рисовать каждую следующую точку, пока не будет обнаружен разделитель.

Повторять до конца файла.

Алгоритм рисования для .spf формата приводится как часть описания многоугольника:

Переместиться в абсолютных координатах к точке, определенной первым указателем.

Соединять точки, определенные каждым следующим указателем в списке.

Соединить точку, определенную последним указателем списка, с точкой, определенной первым указателем, чтобы замкнуть многоугольник.

Вообще удобно иметь согласованное направление (по часовой стрелке или против) для спецификации вершин многоугольника.

A-7 ПРОСТЫЕ АЛГОРИТМЫ СОЕДИНЕНИЯ

Оба формата (.sgf и .spf) определяют простой и ограниченный алгоритм соединения или рисования. Практический интерес представляют два альтернативных, но

¹ И стандартный графический формат (.sgf), и стандартный формат многоугольника (.spf) более десятилетия используются в институте, где работают авторы.

Таблица А-3 Простой полигональный формат файла (.spf)

Прежде всего мы приведем общее описание формата, а затем подробный пример. Файл естественным образом делится на две части. Первая часть образует описание всех координатных векторов в изображении, вторая — описание характеристик многоугольника. В файле допускаются пустые строки. Построчное описание файла имеет следующую форму:

описательная (идентифицирующая) строка
число координатных векторов, многоугольников
список векторов x, y, z

список многоугольников

Каждая строка содержит два типа данных: число вершин многоугольника, за которым следует список вершин в виде соответствующих номеров строк из списка координатных векторов.

Характерный пример:

многоугольник с отверстием

14,2

4,4

4,26

20,26

28,18

28,4

21,4

21,8

10,8

10,4

10,12

10,20

17,20

21,16

21,12

9,1,2,3,4,5,6,7,8,9

5,10,11,12,13,14

Это простой многоугольник с отверстием.

Даны 14 координатных векторов или вершин многоугольника и 2 многоугольника.

Первый многоугольник содержит 9 вершин.

Второй многоугольник содержит 5 вершин.

Предполагается, что многоугольники замкнуты, т.е. первая точка не должна быть переопределена в качестве последней точки списка, чтобы замкнуть многоугольник.

Значение $C(j)$ определяет номер строки в матрице координатных векторов.

Знак $C(j)$ используется для определения оператора рисования.

$C(j) < 0$ — перемещение в абсолютных координатах;

$C(j) > 0$ — рисование в абсолютных координатах.

Хотя такая схема достаточна, в некоторых отношениях она ограничена. Более гибкая и мощная схема использует двумерный управляющий массив $C(i, j)$. Алгоритм рисования определяется следующим образом:

значение $C(i, 1)$ служит для определения строки в матрице координатных векторов;

значение $C(i, 2)$ определяет оператор рисования (табл. А-1):

$C(i, 2) = 0$ — перемещение в абсолютных координатах;

$C(i, 2) = 1$ — перемещение в относительных координатах;

$C(i, 2) = 2$ — рисование в абсолютных координатах;

$C(i, 2) = 3$ — рисование в относительных координатах;

$C(i, 2) = 4$ — точка с абсолютными координатами;

$C(i, 2) = 5$ — точка с относительными координатами.

A-8 ПРОСТОЙ ГРАФИЧЕСКИЙ ПАКЕТ

В методике преподавания и изучения нового предмета, как правило, лучше начинать с простых понятий и тем. Приведение здесь минимального множества программ с использованием концепций, развивавшихся в этой книге и в работе [A-8], преследует цели обучения. Графический пакет растрового дисплея (RDGP) [A-14] представляет собой пример такого минимального множества графических программ. Программы разработаны для четырех языков: ANSI Бейсик, Фортран 77, Си, Паскаль. Описание каждой программы дано для всех четырех языков, сначала для ANSI Бейсика, потом для Фортрана 77, Си, Паскаля. Спецификации каждой программы отделены горизонтальными линиями. Типы переменных определены для каждого параметра каждой программы.

RDGP — ГРАФИЧЕСКИЙ ПАКЕТ РАСТРОВОГО ДИСПЛЕЯ¹

Программы естественным образом распадаются на три группы: управления состоянием дисплея, манипулирование пикселям и рисование линии. В таком порядке они и обсуждаются.

отдельно описанных алгоритма рисования. Допустим, что база данных трехмерных координатных векторов представлена в виде матрицы $n \times 4$. Также определен одномерный управляющий массив $C(j)$, задающий алгоритм рисования следующим образом:

¹ Copyright © 1985 by David F. Rogers and Stephen D. Rogers. Пакет программ RDGP на TRUE Бейсик, Фортран 77 и Си для IBM PC распространяется автором. За информацией обращаться по адресу: Prof. D. F. Rogers, 817 Holly Dr. E., Rt. 10, Annapolis, MD 21401, USA.

имя: rdinit (Инициализировать графический дисплей)
 назначение: Инициализировать графический пакет растрового дисплея и графическое устройство.
 описание: sub rdinit
 subroutine rdinit
 rdinit ()
 procedure rdinit;
 комментарии: rdinit вызывается только один раз в начале программы.

имя: erase (Очистить экран)
 назначение: Очистить графический экран указанным цветом.
 описание: sub erase (r, g, b)
 subroutine erase (r, g, b)
 real r, g, b
 erase (r, g, b)
 double r, g, b;
 procedure (r, g, b: real);
 комментарии: r, g, b — красная, зеленая и синяя составляющие цвета. Их значения задаются из интервала от 0 до 1. (r, g, b)=(0, 0, 0) — черный и (r, g, b)=(1, 1, 1.) — белый.

имя: setrsl (Установить разрешение)
 назначение: Установить псевдоразрешающую способность растрового дисплея.
 описание: sub setrst (xres, yres)
 subroutine, setrsl (xres, yres)
 integer xres, yres
 setrsl (xres, yres)
 int xres, yres;
 procedure setrsl (xres, yres: integer);
 комментарии: xres — псевдоразрешающая способность раstra по оси x или по горизонтали. Она может изменяться от 1 до максимального значения физической разрешающей способности раstra по горизонтали.

yres — псевдоразрешающая способность раstra по оси y или по вертикали. Может изменяться от 1 до максимального значения физической разрешающей способности раstra по вертикали. Псевдорастр располагается в центре физического пространства дисплея. Пиксели имеют квадратную форму, насколько это позволяют ограничения физического дисплея. Программа setrsl вызывается только один раз в начале программы.

имя: setrgb (Установить rgb)
 назначение: Установить текущий цвет для рисования.
 описание: sub, setrgb (r, g, b)
 subroutine setrgb (r, g, b)
 real r, g, b

495
 имя: setrgb (r, g, b)
 назначение: Установить цвет для команд рисования та, па, да, mr, pr, dr и text.
 описание: double r, g, b;
 procedure setrgb (r, g, b: real);
 комментарии: r, g, b — красная, зеленая и синяя составляющие цвета. Их значения задаются из интервала от 0 до 1. (r, g, b)=(0., 0., 0.) — черный и (r, g, b)=(1., 1., 1.) — белый. Эта команда обычно используется, чтобы установить цвет для команд рисования та, па, да, mr, pr, dr и text, данных ниже. Заметим, однако, что вызов команд setpxl и setscn может изменить цвет рисования.

имя: setlut (Установить таблицу просмотра)
 назначение: Установить таблицу цветов просмотра.
 описание: sub setlut (table, rbits, gbits, bbits)
 subroutine setlut (table, rbits, gbits, bbits)
 character table (*)
 integer rbites, gbits, bbits
 setlut (table, rbites, gbits, bbits)
 rdString = lstring (255);
 procedure setlut (table: rdString; rbites, gbits, bbitg: integer);
 комментарии: Таблица — это строка, которая определяет специфику установленной таблицы просмотра. Возможны следующие значения: по умолчанию (таблица просмотра отсутствует), серый и цветной.
 rbites, gbits и bbits — число бит, которые предполагается отвести под красную, зеленую и синюю составляющие буфера кадра. Число градаций каждого цвета 2^{rbites} . Каждый из параметров rbites, gbits, bbits должен быть меньше или равен ширине таблицы цветов просмотра. Их сумма должна быть меньше числа физических бит, отведенных под буфер кадра. Общее число отображаемых в некоторый момент времени цветов, равное $2^{(rbites+gbits+bbits)}$, выйдет за допустимую палитру, значение которой равно 2^{3*w} , если $w \geq n$, где n — ширина таблицы цветов просмотра. В случае использования таблицы серых тонов просмотра значение градации серого цвета равно rbites. Максимальное число градаций серого цвета — это w.

имя: setdm (Установить режим дисплея)
 назначение: Управление режимом дисплея.
 описание: sub setdm (str \$)
 subroutine setdm (str)
 character str (*)
 setdm (str)
 char * str;
 rdString = lstring (255);
 procedure setdm (str: rdString);
 комментарии: Предопределеными являются строки monitor (стандартное алфавитно-цифровое устройство) и graphics (стандартное графическое устройство).

имя:	setpxl (Установить пиксель)
назначение:	Установить единственный пиксель на линии развертки.
описание:	<pre>sub setpxl (x, y, r, g, b) subroutine setpxl (x, y, r, g, b) integer x, y real r, g, b setpxl (x, y, r, g, b) int x, y; double r, g, b; procedure setpxl (x, y: integer; r, g, b: real);</pre>
комментарии:	Координаты x и y указывают левый нижний угол псевдопикселя, который был определен программой setrl. Пиксель покрывает пространство правее и выше позиции, определенной x и y. r, g, b — для Си — указатели на них) — красная, зеленая и синяя составляющие цвета пикселя. Их значения находятся в интервале от 0 до 1. (r, g, b)=(0., 0., 0.) — черный и (r, g, b)=(1., 1., 1.) — белый.
имя:	setscl (Установить линию развертки)
назначение:	Установить группу пикселов на линии сканирования.
описание:	<pre>sub setscl (y, ra, ga, ba, xstart, xstop) subroutine setscl (y, ra, ga, ba, xstart, xstop) integer y, xstart, xstop real ra (*), ga (*), ba (*) setscl (y, ra, ga, ba, xstart, xstop) int y, xstart, xstop; double ra [], ga [], ba []; rgScanline = array [0... x max] of real; procedure setscl (y: integer; ra, ga, ba: rgScanline; xstart, xstop: integer);</pre>
комментарии:	y — устанавливаемая псевдолиния развертки. xstart — начальная позиция пикселов. xstop — позиция последнего пикселя. ra, ga, ba — массивы, содержащие красную, зеленую и синюю составляющие цветов пикселов. Значения элементов ra, ga, ba задаются из интервала от 0 до 1, где (0., 0., 0.) — черный и (1., 1., 1.) — белый. Первый элемент каждого массива используется для цвета пикселя, соответствующего xstart. Следующие пиксели линии развертки используют значения следующих элементов массивов.
имя:	getpxl (Задать пиксель)
назначение:	Дать характеристики единственного пикселя линии развертки.
описание:	<pre>sub getpxl (x, y, r, g, b) subroutine getpxl (x, y, r, g, b) integer x, y real r, g, b getpxl (x, y, r, g, b) int x, y; double *r, *g, *b; procedure getpxl (x, y: integer; var r, g, b: real);</pre>

имя:	commentaries (Комментарии)
назначение:	Координаты x и y указывают левый нижний угол псевдопикселя, который был определен программой setrl. Пиксель покрывает пространство правее и выше позиции, определенной x и y. r, g, b (для Си — указатели на них) — красная, зеленая и синяя составляющие цвета пикселя. Их значения находятся в интервале от 0 до 1. (r, g, b)=(0., 0., 0.) — черный и (r, g, b)=(1., 1., 1.) — белый.
имя:	getscn (Дать линию развертки)
назначение:	Дать характеристики группы пикселов линии развертки.
описание:	<pre>sub getscl (y, ra, ga, ba, xstart, xstop) subroutine getscl (y, ra, ga, ba, xstart, xstop) integer y, xstart, xstop real ra (*), ga (*), ba (*) getscl (y, ra, ga, ba, xstart, xstop) int y, xstart, xstop; double ra [], ga [], ba []; rdScanline = array [0...x max] of real; procedure getscl (y: integer; var ra, ga, ba: rdScanline; xstart, xstop: integer);</pre>
комментарии:	у — псевдолиния развертки, информацию о которой требуется прочитать. xstart — начальная позиция группы пикселов. xstop — позиция последнего пикселя. ra, ga, ba — массивы, которые получают значения красного, зеленого и синего компонентов цвета пикселов, о которых читается информация. Их значения находятся в интервале от 0 до 1. (ra, ga, ba)=(0., 0., 0.) — черный и (1., 1., 1.) — белый. Первый элемент каждого массива используется для цвета пикселя, соответствующего xstart. Следующие элементы массива для следующих пикселов линии развертки.
имя:	text (Текст)
назначение:	Отобразить текст.
описание:	<pre>sub text (str \$) subroutine text (str) character str (*) text (str) char *str; rdString = lstring (255); procedure txt (str: rdString)</pre>
комментарии:	Текст отображается, начиная с последней позиции курсора с использованием предопределенного размера символов. Позиция курсора указывает на левый нижний угол ячейки первого символа. Курсор остается в нижнем правом углу ячейки последнего символа строки. Замечание. В Фортране 77 текстовая/символьная строка должна заканчиваться символом \$, т.е. HELLO\$ служит для отображения на экране HELLO. Замечание. Так как text — это зарезервированное в Паскале слово, используется аббревиатура txt.

имя: ma (Абсолютное перемещение)
назначение: Переместить курсор в позицию с абсолютным значением x и y.
описание: sub ma (x, y)
 subroutine ma (x, y)
 real x, y
 ma (x, y)
 double x, y;
 procedure ma (x, y: real);
комментарии: x и y — действительные числа, x и y — абсолютные координаты для псевдорастра, установленного программой setrsl. 0, 0 — это левый нижний угол псевдорастра.

имя: mr (Относительное перемещение)
назначение: Переместить курсор на dx, dy.
описание: sub mr (dx, dy)
 subroutine mr (dx, dy)
 real dx, dy
 mr (dx, dy)
 double dx, dy;
 procedure mr (dx, dy: real);
комментарии: dx и dy — действительные числа. Перемещение происходит относительно последней позиции курсора в размерах псевдорастра, установленного программой setrsl.

имя: da (Абсолютное рисование)
назначение: Нарисовать тонкую линию в абсолютных координатах.
описание: sub da (x, y)
 subroutine da (x, y)
 real x, y
 da (x, y)
 double x, y;
 procedure da (x, y: real);
комментарии: x и y — действительные числа, заданные в размерах псевдорастра, установленного программой setrsl. Тонкая (шириной в один физический пиксел) линия рисуется от текущей позиции курсора до позиции x, y. Значения x, y — это абсолютные координаты, причем начальная пара 0, 0 соответствует левому нижнему углу псевдорастра, заданного программой setrsl. Для рисования линии используется текущий цвет, который устанавливается с помощью программы setrgb.

имя: dr (Относительное рисование)
назначение: Нарисовать тонкую линию, используя относительные координаты.
описание: sub dr (dx, dy)
 subroutine dr (dx, dy)
 real dx, dy

имя: dr (dx, dy)
назначение: double dx, dy;
описание: procedure dr (dx, dy: real);
комментарии: dx и dy — действительные числа, заданные в размерах псевдорастра, установленного программой setrsl. Тонкая (ширина в один физический пиксел) линия рисуется от последней позиции курсора x0, y0 до позиции x0+dx, y0+dy. Значения dx и dy — это относительные координаты, где x0, y0 — последняя позиция курсора. Для рисования линии используется текущий цвет. Текущий цвет устанавливается с помощью программы setrgb.

имя: pa (Абсолютная точка)
назначение: Отобразить точку в абсолютных координатах.
описание: sub pa (x, y)
 subroutine pa (x, y)
 real x, y
 pa (x, y)
 double x, y;
 procedure pa (x, y: real);
комментарии: Значения x и y — действительные числа, заданные в размерах псевдорастра, который установлен программой setrsl. Маленькая точка (размером в один физический пиксел) отображается в позиции x, y. Значения x, y представляют собой абсолютные координаты, причем пара 0, 0 соответствует левому нижнему углу псевдорастра, заданного программой setrsl. Для отображения точки используется текущий цвет. Текущий цвет устанавливается с помощью программы setrgb.

имя: pr (Относительная точка)
назначение: Отобразить точку, используя относительные координаты.
описание: sub pr (dx, dy)
 subroutine pr (dx, dy)
 real dx, dy
 pr (dx, dy)
 double dx, dy;
 procedure pr (dx, dy: real);
комментарии: Значения dx и dy — это действительные числа, заданные в размерах псевдорастра, установленного программой setrsl. Маленькая точка (размером в один физический пиксел) отображается в позиции x0 + dx, y0 + dy. dx, dy — относительные координаты, а x0, y0 — последняя позиция курсора. Для отображения точки используется текущий цвет. Текущий цвет устанавливается с помощью программы setrgb.

A-9 ЛИТЕРАТУРА¹

- A-1 «Graphical Kernel System (GKS)», ISO 7942, 15 August 1985; also, ANSI X3.124-1985, 24 June 1985.
- A-2 Bono, P., and Herman, I. (eds.), *GKS Theory and Practice*, Springer-Verlag, Heidelberg, 1987.
- A-3 Enderle, G., Kansy, K., and Pfaff, G., *Computer Graphics Programming, GKS – The Graphics Standard*, Springer/Verlag, Heidelberg, 2nd ed., 1987.
- A-4 «Graphical Kernel System for Three Dimensions (GKS-3D)», ISO DIS 8805, July 1987.
- A-5 «Programmer's Hierarchical Interactive Graphics System (PHIGS)», ISO DIS 9592, December 1987; also, dpANS X3.144.
- A-6 Brown, M., and Heck, M., *Understanding PHIGS*, Megatek Corp., San Diego, 1985.
- A-7 Van Dam, A., «PHIGS+, Functional Description Revision 3.0», *Comp. Graph.*, Vol. 22, pp. 125–218, 1988.
- A-8 Rogers, D.F., *Procedural Elements for Computer Graphics*, McGraw-Hill, New York, 1985. (См. [1-1] на с. 74.)
- A-9 «Computer Graphics Metafile for the Storage and Transfer of Picture Description Information (CGM)», ISO 8632-Parts 1 through 4: 1987; also, ANSI/X3.122-1986.
- A-10 Arnold, D.B., and Bono, P.R., *CGM and CGI*, Springer-Verlag, Heidelberg, 1988.
- A-11 McKay, L., *CGI/CGM PRIMER*, Nova Graphics International, Austin, TX 1987.
- A-12 «Computer Graphics Interfacing Techniques for Dialogues with Graphical Devices (CGI)», ISO DP 9636, 6 December 1986; also, dpANS X3.161.
- A-13 International Organization for Standardization: Information Processing Systems—Computer Graphics—Interface Techniques for Dialogues with Graphical Devices. Baseline Document. ISO/TC97 N1511 (1985).
- A-14 Rogers, D.F., and Rogers, S.D., «A Raster Display Graphics Package for Education», *IEEE Comp. Graph. & Appl.*, Vol. 6, pp. 51–58, April 1986.

Приложение В**МАТРИЧНЫЕ ОПЕРАЦИИ**

Это приложение содержит некоторые несложные правила из матричной алгебры. Матрица — это прямоугольный массив чисел, операции над которыми подчинены приведенным правилам.

B-1 ТЕРМИНОЛОГИЯ

Числа, которые составляют матрицу, называются ее элементами. Эти элементы образуют строки и столбцы внутри матрицы. Если число строк равно числу столбцов, матрица называется квадратной. Рассмотрим *квадратную* матрицу 4×4 :

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}.$$

Обозначим элементы матрицы через a_{ij} . Первый индекс — номер строки, второй — номер столбца. Таким образом, a_{34} — элемент из третьей строки, четвертого столбца. Матрица, состоящая из m строк и n столбцов, имеет размерность $m \times n$. Для квадратной матрицы выполняется равенство $m = n$. Элементы a_{11}, a_{22}, a_{33} и a_{44} при $i = j$ называются диагональными элементами. Сумма диагональных элементов квадратной матрицы называется ее следом.

Диагональной матрицей называют матрицу, в которой отличны от нуля только диагональные элементы, т. е. $a_{ij} = 0$, если $i \neq j$. Единичной матрицей называют матрицу, у которой $a_{ij} = 0$ для $i \neq j$ и $a_{ij} = 1$ для $i = j$. Пример единичной 3×3 матрицы:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Каждый элемент нулевой матрицы равен 0, т. е. $a_{ij} = 0$. Две матрицы равны, только когда каждый элемент первой матрицы равен соответствующему элементу второй матрицы, т. е. $a_{ij} = b_{ij}$.

¹ Эти документы имеются в различных источниках:

Опубликованные ANSI-стандарты (а именно GKS и CGM) и ISO-документы (CGI, CGM, GKS, GKS-3D, PHIGS) можно получить в ANSI (1430 Broadway, New York, NY 10018). Помимо США ISO-документы можно получить из национальных организаций по стандартам (например, BSI в Великобритании, AFNOR во Франции, DIN в ФРГ и JIS в Японии). Проекты ANSI-стандартов (CGI, GKS-3D и PHIGS) можно получить в секретариате X3 фирмы CBEMA (311 First Street, Suite 500, Washington, DC 20001). Документы IGES и PDES можно получить в фирме IGES (IGES Committee Chairman, Mr. Bradford Smith, National Bureau of Standards, Gaithersburg, MD 20760).

Симметричной называется квадратная матрица, у которой $a_{ij} = a_{ji}$. Например,

$$\begin{bmatrix} 2 & 3 & 6 \\ 3 & 17 & 5 \\ 6 & 5 & 12 \end{bmatrix}.$$

Строка матрицы является $1 \times n$ -матрицей. Строки матрицы иногда называют векторами строк или просто векторами. Столбец матрицы является $n \times 1$ -матрицей. Столбцы матрицы иногда называют векторами столбцов или просто векторами.

Скаляр — это 1×1 -матрица.

B-2 СЛОЖЕНИЕ И ВЫЧИТАНИЕ

Если две матрицы имеют одинаковую размерность, то для них определены операции сложения и вычитания. Чтобы сложить или вычесть две матрицы, надо сложить или вычесть соответствующие элементы, т. е. $c_{ij} = a_{ij} + b_{ij}$ или $c_{ij} = a_{ij} - b_{ij}$ соответственно. Например:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} = \begin{bmatrix} 2 & 6 & 10 \\ 6 & 10 & 14 \\ 10 & 14 & 18 \end{bmatrix}$$

и

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} - \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} = \begin{bmatrix} 0 & -2 & -4 \\ 2 & 0 & -2 \\ 4 & 2 & 0 \end{bmatrix}.$$

Матрица $[B]$ называется отрицанием матрицы A , если $b_{ij} = -a_{ij}$.

B-3 УМНОЖЕНИЕ

Умножение матриц — это очень полезная операция для машинной графики. Рассмотрим матрицу размера $n \times m$ и вторую матрицу размера $r \times s$. Для умножения этих матриц необходимо, чтобы m было равно r . Это значит, что число столбцов первой матрицы должно быть равно числу строк второй матрицы. Правила умножения матриц лучше всего описывает следующий пример. Если матрица $[A]$ имеет размерность 4×3 , а $[B]$ — квадратная 3×3 -матрица, то произведение этих матриц определяется следующим образом:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} =$$

$$= \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} & a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32} & a_{11}b_{13} + a_{12}b_{23} + a_{13}b_{33} \\ a_{21}b_{11} + a_{22}b_{21} + a_{23}b_{31} & a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32} & a_{21}b_{13} + a_{22}b_{23} + a_{23}b_{33} \\ a_{31}b_{11} + a_{32}b_{21} + a_{33}b_{31} & a_{31}b_{12} + a_{32}b_{22} + a_{33}b_{32} & a_{31}b_{13} + a_{32}b_{23} + a_{33}b_{33} \\ a_{41}b_{11} + a_{42}b_{21} + a_{43}b_{31} & a_{41}b_{12} + a_{42}b_{22} + a_{43}b_{32} & a_{41}b_{13} + a_{42}b_{23} + a_{43}b_{33} \end{bmatrix}.$$

B-4 ОПРЕДЕЛИТЕЛЬ КВАДРАТНОЙ МАТРИЦЫ

Иначе говоря, каждая строка в матрице $[A]$ поэлементно умножается на соответствующий столбец матрицы $[B]$, результаты складываются и записываются в качестве элемента результирующей матрицы $[C]$. Заметим, что умножение 4×3 -матрицы на 3×3 -матрицу, дает матрицу размерности 4×3 . Вообще говоря, в результате умножения $n \times m$ -матрицы на $r \times s$ -матрицу при $m = r$ мы получим $n \times s$ -матрицу. Рассмотрим численный пример:

$$\begin{bmatrix} 1 & 3 \\ 4 & 2 \\ 1 & 1 \\ 6 & 4 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1+9 & 2+12 \\ 4+6 & 8+8 \\ 1+3 & 2+4 \\ 6+12 & 12+16 \\ 3+6 & 6+8 \end{bmatrix} = \begin{bmatrix} 10 & 14 \\ 10 & 16 \\ 4 & 6 \\ 18 & 28 \\ 9 & 14 \end{bmatrix}.$$

Операция умножения матриц не является коммутативной, т. е. в общем случае произведение $[A][B]$ не равно $[B][A]$. Таким образом, важен порядок (последовательность) умножения. Для операции умножения матриц верны первый и второй дистрибутивные законы, т. е. $A(B+C) = AB+AC$ и $(A+B)C = AC+BC$. Также выполняется ассоциативный закон $A(BC) = (AB)C$.

B-4 ОПРЕДЕЛИТЕЛЬ КВАДРАТНОЙ МАТРИЦЫ

Определитель квадратной матрицы $[A]$ обозначается $|A|$. В теории матриц определитель имеет много полезных свойств. Для наших целей достаточно проиллюстрировать метод вычисления определителя на примере квадратной 3×3 -матрицы. Для матриц

$$M = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

определитель равен

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}$$

$$= a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31}) + a_{13}(a_{21}a_{32} - a_{22}a_{31}),$$

т. е. определитель имеет единственное численное значение. В качестве примера подсчитаем определитель

$$\begin{vmatrix} 3 & 9 & 4 \\ 6 & 1 & 8 \\ 2 & 5 & 3 \end{vmatrix}.$$

В результате получим

$$3(3 - 40) - 9(18 - 16) + 4(30 - 2) = 3(-37) - 9(2) + 4(28) = -17.$$

B-5 ТРАНСПОНИРОВАНИЕ МАТРИЦЫ

Транспонированная матрица — это исходная матрица, в которой строки заменены на столбцы, т. е. $b_{ij} = a_{ji}$. Например,

$$\begin{bmatrix} 1 & 6 & 3 & 7 \\ 2 & 8 & 11 & 4 \\ 6 & 3 & 2 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 2 & 6 \\ 6 & 8 & 3 \\ 3 & 11 & 2 \\ 7 & 4 & 1 \end{bmatrix},$$

где специальный символ T определяет операцию транспонирования. Заметим, что если исходная матрица имела размерность $m \times n$, то размерность транспонированной матрицы будет $n \times m$.

B-6 ОБРАЩЕНИЕ КВАДРАТНОЙ МАТРИЦЫ

В алгебре скаляров справедливо следующее утверждение, если $ax = y$, то $x = a^{-1}y$, где a^{-1} это просто обратная величина к a , т. е. $a^{-1} = 1/a$. В матричной алгебре операция деления не определена и обратной матрицы в смысле скалярной алгебры не существует, однако если $[A][X] = [Y]$, то $[X] = [A]^{-1}[Y]$, где $[A]^{-1}$ — матрица, обратная к квадратной матрице $[A]$.

Обратная матрица существует для квадратной и неединичной матрицы. Однако не каждая квадратная матрица имеет обратную. Но если определитель квадратной матрицы не равен нулю, то обратная матрица существует и единственна. Если обратная матрица существует, то $[A][A]^{-1} = [I]$, где $[I]$ — единичная матрица.

Рассмотрим произведение двух квадратных матриц, размерностью 3×3 .

$$\begin{bmatrix} 1 & 2 & 3 \\ 1 & 3 & 3 \\ 1 & 2 & 4 \end{bmatrix} \begin{bmatrix} 6 & -2 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Если произведение двух матриц $[A][B] = [I]$, то $[B] = [A]^{-1}$, т. е. $[B]$ — это матрица, обратная к $[A]$.

Существует много способов вычисления обратной матрицы. В большинстве языков программирования есть функции или подпрограммы, позволяющие вычислять значения элементов матрицы, обратной к данной. Описание таких алгоритмов выходит за рамки нашего краткого изложения.

Приложение С

ПСЕВДОКОД

Приведенный псевдокод призван помочь в понимании и реализации представленных в тексте алгоритмов. Он не задумывался как точный, синтаксически корректный и полный язык. Элементы псевдокода заимствованы из нескольких общепринятых языков программирования: Бейсика, ФорTRANа, Паскаля, Си и т. д. Псевдокод содержит структурные конструкции, например **if-then-else** и **while**. Для удобства включен оператор безусловного перехода **go to**. Оператор цикла **for-next** взят из Бейсика. Псевдокод включает также подпрограммные модули и матричные функции. Специальные функции и программы, такие, как **Min**, **Max**, **Push**, **Pop**, определены в алгоритмах в индивидуальном порядке.

Кратко охарактеризуем общие соглашения, используемые при описании алгоритмов. Все ключевые слова набираются жирным шрифтом строчными буквами. Все операторы внутри тела оператора **if-then-else** или цикла **for-next** выделяются отступом. Комментарии набираются курсивом и печатаются с таким же отступом, как и операторы, к которым они относятся. Многолитерные имена переменных, как правило, набираются строчными буквами. Однолитерные переменные могут представляться как прописной, так и строчной буквами. Функции выделяются жирным шрифтом и начинаются с прописной буквы. Далее следует детальное описание этих соглашений.

C-1 КОММЕНТАРИИ

Комментарии набираются курсивом. Они располагаются с тем же отступом, что и операторы, к которым они относятся. В начале алгоритма помещаются комментарии, кратко описывающие назначение этого алгоритма и определяющие используемые переменные.

C-2 КОНСТАНТЫ

Все константы являются десятичными числами, если в комментариях не оговорено иное. Например: 9, -3, 6.732, 1×10^{-9} , 3.

C-3 ПЕРЕМЕННЫЕ

Переменная — это имя, используемое для хранения значения. Это значение может изменяться. В длинных именах переменных используются, в основном, строчные

буквы; прописные используются в тексте книги лишь в тех случаях, когда это способствует лучшему пониманию программы. Однолитерные переменные могут набираться как строчной, так и прописной буквой. Для большей наглядности можно употреблять литеры с индексами. Примеры переменных: flag, P'_2 , x, y.

C-4 МАССИВЫ

Массив — имя для индексированной совокупности значений. Для имен массивов действуют те же соглашения, что и для переменных. Обращение ко всему массиву осуществляется по его имени, а к отдельным элементам — по имени массива, за которым в скобках следует список индексов. Примеры массивов: window, window(1, 3).

C-5 ОПЕРАТОР ПРИСВАИВАНИЯ

Знак «равно» используется для присвоения переменной, расположенной в левой части оператора, значения выражения из правой части этого оператора.

C-6 АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ

Обычные арифметические операции умножения, деления, сложения, вычитания и возведения в степень обозначаются знаками *, /, +, -, ^.

C-7 ЛОГИЧЕСКИЕ ОПЕРАЦИИ И ОПЕРАЦИИ ОТНОШЕНИЯ

Логические операции and и or выделяются жирным шрифтом, как это здесь показано. Операции отношения «равно», «не равно», «меньше», «больше», «меньше или равно», «больше или равно» обозначаются =, ≠, <, >, ≤, ≥ соответственно. Эти операции используются для проверки условий. Результатом проверки являются логические значения «истина» или «ложь».

C-8 ОПЕРАТОР finish

Оператор finish используется для обозначения конца работы алгоритма.

C-9 ОПЕРАТОРЫ while И end while

Операторы, расположенные внутри блока while-end while, выполняются циклически, до тех пор, пока истинно некоторое условие. Условие проверяется в начале блока. Когда условие перестает быть истинным, выполнение программы продолжается с оператора, следующего за end while. Оператор end while используется для обозначения конца блока. Все операторы внутри блока while-end while располагаются с отступом. Общая форма оператора

```
while (условие)
    [операторы, которые необходимо выполнить]
end while
```

Пример

```
i = 0
while (i < 5)
    x = x + 5
    i = i + 1
end while
finish
```

C-10 ОПЕРАТОР if-then

Оператор if-then используется для выбора альтернативы пути выполнения программы либо для присвоения переменной альтернативного значения в зависимости от истинности некоторого условия.

Если аргументом then служит номер оператора и условие истинно, то управление передается на оператор с этим номером. В противном случае выполняется следующий по порядку оператор. Номера операторов являются их метками.

Если аргументом then служит оператор присваивания и условие истинно, то присваивание выполняется. В противном случае присваивание не выполняется и управление передается сразу на следующий оператор. Общая форма оператора:

```
if (условие) then (номер оператора)
if (условие) then (оператор присваивания)
```

Примеры

```
if (i < 10) then 3
if (i < 10) then x = x + 1
```

C-11 ОПЕРАТОРЫ if-then-else И end if

Оператор if-then-else используется для выбора одного из двух альтернативных блоков операторов в зависимости от истинности некоторого условия. Для обозначения конца блока оператора if-then-else используется оператор end if. Оператор не подразумевает повторения. Выполняется только один из альтернативных блоков. После этого выполнение программы продолжается с оператора, следующего за end if. Все операторы внутри блока if-then-else располагаются с отступом. Общая форма оператора

```
if (условие) then
    [операторы, выполняемые, если условие истинно]
else
    [операторы, выполняемые, если условие ложно]
end if
```

Пример

```
if (i ≥ 0) then
    x = x + 1
else
    x = x - 1
end if
```

Если оператор **if-then-else** записывается в одной строке, то оператор **end if** опускается. Заметим, что если опускается **else** и вторая группа операторов, то в результате получается блок оператора **if-then**.

C-12 ОПЕРАТОР for-next

Управление циклом можно организовать с помощью оператора **for-next**, как и с помощью оператора **while**. Циклическое выполнение операторов, расположенных внутри тела цикла **for-next**, происходит до тех пор, пока значение переменной цикла находится в заданных пределах. Все операторы внутри блока **for-next** располагаются с отступом. Общая форма оператора:

```
for (пер.цикла) = (нач.знач) to (кон.знач) step(шаг.цикла)
    [операторы, которые надо выполнить]
next (пер.цикла)
```

Если шаг цикла отсутствует, то он полагается равным единице. Шаг цикла может быть отрицательным. Начальное, конечное значение переменной цикла и шаг цикла могут быть переменными.

Пример

```
for x = 1 to n step a
    y = y + x
next x
```

C-13 ОПЕРАТОР go to

Оператор **go to** вызывает безусловный переход на оператор, определяемый аргументом. Общая форма оператора:

```
go to (номер оператора)
```

Номера операторов являются метками. Они располагаются в крайней левой позиции строки оператора.

C-14 ПОДПРОГРАММНЫЕ МОДУЛИ

Подпрограмма — отдельный программный модуль, который вызывается с помощью оператора **call**. Начало подпрограммы определяет оператор **subroutine**. Выход из подпрограммы определяется оператором **return**. После выхода из подпрограммы управление возвращается оператору, следующему за оператором **call** в вызывающей программе. Оператор **subroutine** содержит список входных и список выходных переменных. Связь между вызывающей программой и подпрограммным модулем осуществляется только через эти переменные. Все остальные переменные подпрограммного модуля локализованы в этом модуле. Общая форма операторов **call**, **subroutine** и **return**:

```
call имя (входные переменные; выходные переменные)
subroutine имя (входные переменные; выходные переменные)
return
```

Списки входных и выходных переменных операторов **call** и **subroutine** разделяются точкой с запятой и должны соответствовать друг другу. Имена подпрограмм состоят из строчных букв, набранных жирным шрифтом. Пример подпрограммного модуля:

```
subroutine check (x, y; flag)
    if x < y then
        flag = 0
    else
        flag = 1
    end if
    return
```

C-15 ТРИГОНОМЕТРИЧЕСКИЕ ФУНКЦИИ

Тригонометрические функции, такие, как **Sin (θ)**, **Cos (θ)**, **Tan (θ)** и обратные к ним **Arccsin (x)**, **Arccos (s)**, **Arctan (x)** принимают и возвращают значения в градусах, если установлен переключатель **option angle degrees**. Общепринятые гиперболические функции и их обратные функции **Sinh (θ)**, **Cosh (θ)**, **tanh (θ)**, **Arccosh (x)**, **Arccosh (x)**, **Arctanh (x)** также принимают и возвращают значения в градусах, если установлен этот переключатель. В противном случае они принимают и возвращают значения в радианах. Имена этих функций начинаются с прописной буквы и выделяются жирным шрифтом.

C-16 МАТРИЧНЫЕ ФУНКЦИИ

Матричные функции служат для выполнения типичных операций матричной алгебры, например сложения, вычитания, умножения, обращения и транспонирования матриц. Матричные функции манипулируют целыми массивами. Размерность каждого массива, используемого в качестве аргумента матричной функции, должна соответствовать выполняемой операции. Размерность результирующей матрицы также соответствует выполненной операции. Например, в результате умножения 3 × 2-матрицы на 2 × 3-матрицу получается 3 × 3-матрица. Массив матрицы читается и записывается по строкам. Размерность матрицы может быть изменена во время выполнения операций. Массив матрицы может быть использован в обеих частях оператора присваивания. Имена матричных функций начинаются с прописной буквы и выделяются жирным шрифтом. Матричные функции:

сложение	Mat	A = B + C
вычитание	Mat	A = B - C
умножение	Mat	A = B * C
обращение	Mat	A = Inv(B)
транспонирование	Mat	A = Trn(B)
вычисление определителя	Значение	= Det(B)
единичная матрица	Mat	A = Idn
нулевая матрица	Mat	A = Zer(n1, n2)
постоянная матрица	Mat	A = Con(n1)
скалярное произведение	Значение	= Dot(A, B)

Функция **Zet** формирует массив матрицы размерностью $[n1 \times n2]$ и заполняет его нулями. Функция **Cop** заполняет массив заранее заданной размерности константы $n1$.

С-17 СПЕЦИАЛЬНЫЕ ФУНКЦИИ

В алгоритмах, представленных в данной книге, определены разнообразные специальные функции. Имена этих функций выделяются жирным шрифтом и начинаются с прописной буквы. В качестве примера приведем функцию

Max(x_1, x_2),

результатом которой является большее из значений x_1 и x_2 .

Приложение D

ФОРМАТ ФАЙЛА В-СПЛАЙН ПОВЕРХНОСТИ

В связи с возникшим значительным интересом к В-сплайн поверхностям, разработан ряд значительных форматов файлов для обмена их описания. Еще один формат предложен в табл. D-1. Четырехмерные однородные координаты используются для обозначения определяемых точек прямоугольной сетки. Это позволяет объединить трехмерные физические позиции точек прямоугольной сетки и однородный координатный (весовой) коэффициент в общем описании. Предлагаемый формат, вообще говоря, достаточен для управления рациональными однородными и неоднородными, периодическими и открытыми В-сплайн поверхностями. Нерациональные сплайны поверхностей определяются установкой всех однородных координатных коэффициентов (весов) равными единице. Рациональные поверхности Безье определяются посредством однородных открытых связующих векторов в виде $[k \text{ нулей}, k \text{ единиц}]$ с подходящими однородными координатными коэффициентами. Нерациональные поверхности Безье определяются с помощью соответствующих связующих векторов и установкой всех однородных координатных коэффициентов (весов) равными единице. Составные поверхности определяются простым повторением полного описания для каждой поверхности. Заметим, что такой подход позволяет описывать различные типы В-сплайнов и поверхностей Безье в одном файле. Запятые должны быть проставлены, как показано в таблице. В файле допустимы пробелы и пустые строки.

Таблица D-1 Формальное описание формата файла В-сплайн поверхности

Строка идентификации поверхности	например, номер корпуса (рис. 6-50)
нерациональная рациональная	рациональная, нерациональная в <i>u</i> , <i>w</i> соответственно
открытая, периодическая	открытая, периодическая, неоднородная в <i>u</i> , <i>w</i> соответственно
порядок базиса, порядок базиса	порядок в <i>u</i> , порядок в <i>w</i>
кол-во узлов сетки	кол-во узлов сетки в направлении <i>u</i>
кол-во узлов сетки	кол-во узлов сетки в направлении <i>w</i>
вектор связи в <i>u</i>	последовательность чисел 1, 2, 3, ...
вектор связи в <i>w</i>	последовательность чисел 1, 2, 3, ...
список 4D-точек сетки	массив $B(i, j)$ размерностью ((кол-во узлов сетки в направлении <i>u</i>) \times (кол-во узлов сетки в направлении <i>w</i>), 4). Элементы массива $B(i, j)$ располагаются с опережающим изменением индекса <i>j</i> , т.е. по строкам. Каждая строка содержит компоненты <i>x</i> , <i>y</i> , <i>z</i> , <i>h</i> , где <i>h</i> — однородный (весовой) коэффициент для каждой точки сетки.
	Замечание. Для нерациональных В-сплайнов все <i>h</i> = 1.

Пример файла

Пример поверхности (см. рис. 6-50)
нерац., нерац.

открыта, открыта

4, 4

8, 8

0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5

0, 0, 0, 0, 1, 2, 3, 4, 5, 5, 5, 5

Точки сетки $B(i, j)$, расположенные по строкам.

0	,	62.7896	,	-14.2345	,	1	37.8884	,	50.5464	,	108	,	1
0	,	58.3546	,	-11.7746	,	1	33.861	,	47.4467	,	108	,	1
0	,	50.7321	,	-7.7246	,	1	27.1525	,	36.9685	,	108	,	1
0	,	37.2752	,	-0.7445	,	1	23.2976	,	19.6017	,	108	,	1
0	,	24.6114	,	0.2266	,	1	19.832	,	11.186	,	108	,	1
0	,	12.2841	,	-0.7321	,	1	12.7107	,	2.5924	,	108	,	1
0	,	0.9638	,	-4.2096	,	1	5.5457	,	-0.4150	,	108	,	1
0	,	-0.02351	,	-0.06207	,	1	-0.01007	,	-0.04313	,	108	,	1
13.2994	,	57.2518	,	0	,	1	41.0213	,	48.1632	,	162	,	1
8.6079	,	50.7918	,	0	,	1	39.6319	,	39.2661	,	162	,	1
7.2922	,	52.4931	,	0	,	1	38.6344	,	30.4192	,	162	,	1
-0.6518	,	33.3121	,	0	,	1	33.8422	,	16.8768	,	162	,	1
1.5411	,	14.3331	,	0	,	1	28.2826	,	6.96425	,	162	,	1
3.7183	,	13.4063	,	0	,	1	18.2607	,	-1.0973	,	162	,	1
7.87419	,	1.8224	,	0	,	1	4.4469	,	0.2564	,	162	,	1
0.9025	,	-0.4589	,	0	,	1	0.1344	,	0.01421	,	162	,	1
17.0482	,	57.6166	,	13.5	,	1	40.4748	,	47.8315	,	216	,	1
14.0925	,	55.5365	,	13.5	,	1	41.1403	,	43.5986	,	216	,	1
11.067	,	48.282	,	13.5	,	1	40.9666	,	30.6183	,	216	,	1
3.0433	,	40.1302	,	13.5	,	1	41.4111	,	14.9213	,	216	,	1
0.2289	,	21.5281	,	13.5	,	1	34.093	,	0.2635	,	216	,	1
3.5539	,	7.0775	,	13.5	,	1	16.9322	,	0.9311	,	216	,	1
3.0443	,	0.7293	,	13.5	,	1	5.55319	,	0.167384	,	216	,	1
-0.2092	,	0.08977	,	13.5	,	1	0.06208	,	-0.07885	,	216	,	1
29.2903	,	53.3467	,	54	,	1	40.536	,	47.9986	,	270	,	1
30.7639	,	54.1187	,	54	,	1	40.8065	,	41.4721	,	270	,	1
21.1234	,	48.1403	,	54	,	1	39.8169	,	33.6058	,	270	,	1
12.0645	,	35.461	,	54	,	1	41.8009	,	13.1977	,	270	,	1
9.0546	,	17.3787	,	54	,	1	37.0837	,	0.3062	,	270	,	1
8.6407	,	6.64932	,	54	,	1	13.1546	,	1.2999	,	270	,	1
5.1127	,	-0.7182	,	54	,	1	7.0730	,	-0.2726	,	270	,	1
0.1572	,	-0.03143	,	54	,	1	0.07475	,	-0.04795	,	270	,	1

Приложение Е**ЗАДАЧИ**

Изучение компьютерной графики невозможно без ее практического использования, поэтому здесь приводится ряд задач и спецификаций программ. Задачи в основном направлены на то, чтобы подтвердить или развить концепции, представленные в тексте, или на то, чтобы накопить определенный опыт работы в соответствии с предлагаемой технологией. Как правило, задачи акцентируют внимание на каких-то отдельных конкретных аспектах материала. Задачи могут требовать, а могут и не требовать программирования.

Спецификации программ разработаны не только для иллюстрации концепций и технологий, представленных в книге, но также чтобы передать некоторый опыт создания достаточно сложных программ машинной графики. Хорошо реализованные, они могут составить инструментарий полезных графических программ. Результатом законченных программных проектов являются полезные программы общего назначения.

Это приложение содержит задачи. Спецификации программ приводятся в приложении F. Задачи и спецификации объединены в соответствии с главами книги.

ГЛАВА 2

2-1. Показать, что зеркальное отражение в плоскости относительно оси x , следующее за зеркальным отражением относительно прямой $y = -x$, эквивалентно полному повороту вокруг точки начала координат.

2-2. Для плоского прямоугольника, заданного координатными векторами своих вершин $P_1 [1 \ 1]$, $P_2 [3 \ 1]$, $P_3 [4 \ 2]$, $P_4 [2 \ 3]$, определить единственную матрицу преобразования, которая

- 1) зеркально отражает его относительно прямой $x = 0$;
- 2) осуществляет параллельный перенос на -1 в направлениях x и y ;
- 3) осуществляет поворот на 180° относительно точки начала координат.

Используя эти преобразования, определить новые координатные векторы вершин прямоугольника. Построить исходный и преобразованный прямоугольники на каком-либо графическом устройстве.

2-3. Показать, что 2×2 -матрица

$$[T] = \begin{bmatrix} 1-t^2 & 2t \\ \frac{1+t^2}{1-t^2} & \frac{1+t^2}{1-t^2} \\ -2t & \frac{1-t^2}{1+t^2} \\ \frac{1+t^2}{1-t^2} & \frac{1+t^2}{1-t^2} \end{bmatrix}$$

осуществляет полный поворот.

2-4. Единичный квадрат преобразуется 2×2 -матрицей. Результатирующие координатные векторы:

$$\begin{bmatrix} 0 & 0 \\ 2 & 3 \\ 8 & 4 \\ 6 & 1 \end{bmatrix}.$$

Найти матрицу преобразования.

2-5. Показать, что для $x = t^2$, $y = t$ результатом преобразования

$$\begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} 0 & -2 & 2 \\ -2 & 2 & -2 \\ 1 & 0 & 1 \end{bmatrix}$$

является множество точек, образующих единичную окружность.

ГЛАВА 3

3-1. Показать, что для достижения некоторого эффекта преобразования объекта путем конкатенации некоторых матриц слева требуется, чтобы они были применимы в обратном порядке, когда используется конкатенация справа, т. е.

$$[Tr][Rx][Ry][Data] = [Data][Ry][Rx][Tr],$$

где $[Data]$ — матрица данных для объекта.

3-2. Вывести результирующую матрицу преобразования для перевода m , n , l в x , y , z соответственно; поворота вокруг оси u на угол φ ; поворота вокруг оси x на угол θ , следующих за преобразованием центрального перспективного проецирования на плоскость $z = 0$, с центром проекции на оси z в точке $z = k$. Сравнить результат с (3-63).

3-3. Изменить порядок выполнения поворотов в задаче 3-2. Сравнить результаты.

3-4. Для куба из примера 3-23 определить координатные векторы после преобразования, использующего конкатенацию матриц из задачи 3-2 при $m = l = n = 0$, $\varphi = 60^\circ$, $\theta = 30^\circ$ и $k = 2.5$. Сравнить полученный результат с примером 3-23 после построения обоих результатов рядом на одном листе графической бумаги.

3-5. Изменить порядок выполнения поворотов в задаче 3-4. Сравнить полученный результат с результатом задачи 3-4 и с примером 3-23 путем построения этих результатов на одном листе графической бумаги.

3-6. Получить матрицу преобразования размера 4×4 для объекта, заданного координатными векторами

$$\begin{bmatrix} 0 & 0 & 2 & 1 \\ 2 & 0 & 2 & 1 \\ 2 & 1 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ 0 & 2 & 2 & 1 \\ 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 1 \\ 2 & 2 & 0 & 1 \\ 0 & 2 & 0 & 1 \\ 2 & 2 & 1 & 1 \end{bmatrix}$$

с алгоритмом соединения (разд. А-7)

$$C(j) = -1, 2, 3, 4, 5, 1, -6, 7, 8, 9, 6, -1, 6, -2, 7, -5, 9, -4, 10, 8, -10, 3$$

для вида из точки наблюдения (центр проекции), расположенной на 50 единиц правее, 60 единиц выше и выдвинутой на 150 единиц вперед от центра объекта. Помните, что мы смотрим в перспективу.

3-7. Повернуть куб с отсеченным углом (задача 3-6) на 30° относительно оси, проходящей через точки $P_1[-3 \quad -3 \quad 2]$ и $P_2[-3 \quad -2 \quad -2]$. Определить преобразованные координатные векторы. Нарисовать результат в диметрической проекции с коэффициентом искажения 0.625. Оба диметрических угла вращения считать положительными.

3-8. Показать, что любые комбинации полных поворотов и/или четное число зеркальных отражений могут быть представлены не более чем тремя поворотами вокруг координатных осей.

3-9. Продолжить анализ, приведенный в разд. 3-20 для случая, когда вектор зрения из центра проекции не перпендикулярен плоскости проекции.

ГЛАВА 4

4-1. Получить отношение между числом точек, радиусом и толерантностью (точностью приближения), в случае когда круг представляется вписанным в него многоугольником. Толерантность определяется как наибольшее расстояние между дугой и хордой, соединяющей две точки многоугольника на внешней окружности.

4-2. Кривая Ламе описывается формулой

$$\frac{x^n}{c_1} + \frac{y^n}{c_2} = 1.$$

Определить овал, проходящий через три точки $P_1[0 \quad 6]$, $P_2[5 \quad 5]$, $P_3[6 \quad 0]$. Определить точки пересечения овала и круга с радиусом 3 и центром в точке $x = 6$, $y = 0$.

4-3. Вычислить равное число точек на внешней дуге четверти круга в первом квадранте, используя непараметрический явно заданный (уравнение (4-1)), параметрический тригонометрический (уравнение (4-2)) и параметрический чередующийся (уравнение (4-3)) анализ. Построить и сравнить результаты.

4-4. Получить уравнение (4-13).

4-5. Разработать и реализовать алгоритм для определения вершин параметрической параболы, заданной уравнением 4-13.

4-6. Доработать метод, предложенный в разд. 4-9, для случая круга, касающегося заданной прямой.

4-7. Доработать метод, предложенный в разд. 4-9, для случая прямой, касающейся круга заданного радиуса с центром в заданной позиции.

4-8. Доработать метод, предложенный в разд. 4-9, для случая окружности заданного радиуса, касающейся двух окружностей заданного радиуса с центрами в заданных позициях. Небольшая подсказка: существует 16 возможных решений. Построение дерева возможных вариантов решения зависит от радиуса касательной окружности и расстояния между двумя заданными окружностями.

4-9. Доработать метод, предложенный в разд. 4-9, для случая окружности заданного радиуса, одновременно касающейся прямой и окружности заданного радиуса с центром в заданной позиции. Небольшая подсказка: существует 8 возможных решений. Прямая может пересекать или не пересекать окружность.

4-10. Доработать метод, предложенный в разд. 4-9, для случая эллипса с заданными большой и малой полуосами, касающегося двух пересекающихся прямых. Какая дополнительная информация могла бы потребоваться?

4-11. Решить задачи 4-5–4-8, заменив все окружности эллипсами с заданными большими и малыми полуосами. В каждом случае определить, если это необходимо, какая дополнительная информация требуется.

4-12. Какой тип конического сектора представлен каждым из следующих уравнений:

$$\begin{aligned}x^2 + y^2 + 4x - 2y &= 10, \\x^2 + y^2 + xy - x + y &= 3, \\x^2 - 4xy + 4y^2 + 3x &= 6, \\2x^2 - y^2 + 4xy + 2x - 3y &= 6, \\x^2 - 4xy + 3y^2 + 5y &= 7, \\x^2 - y^2 &= 1.\end{aligned}$$

4-13. Преобразовать каждое из уравнений задачи 4-12 к стандартной форме.

4-14. Показать, что уравнение (4-33) эквивалентно следующему:

$$A(x+m)^2 + B(y+n)(x+m) + C(y+n)^2 + F(x+m) + G(y+n) + H = 0.$$

4-15. Показать, что следующее квадратное уравнение определяет параболу, и нарисовать сегмент между $x = 1$ и $x = 3$ для y_{\min} :

$$16x^2 - 24xy + 9y^2 - 60x - 80y + 20 = 0.$$

4-16. Первый квадрант единичного круга параметрически представляется в виде

$$x = \frac{(1 - \sqrt{2})t^2 + (\sqrt{2} - 2)t + 1}{(2 - \sqrt{2})t^2 + (\sqrt{2} - 2)t + 1},$$

$$y = \frac{(1 - \sqrt{2})t^2 + \sqrt{2}t}{(2 - \sqrt{2})t^2 + (\sqrt{2} - 2)t + 1}.$$

Определить значение y для $x = 0,5$. Указание: использовать метод итераций и сначала определить t для $x = 0,5$.

ГЛАВА 5

5-1. Сгенерировать параметрическую кривую, показанную на рис. 5-3.

5-2. Заданы четыре координатных вектора $P_1[0 \ 0 \ 0]$, $P_2[1 \ 1 \ 1]$, $P_3[2 -1 \ -1]$, $P_4[3 \ 0 \ 0]$, определить кусочно непрерывный кубический сплайн кривой, проходящей через них, используя аппроксимацию хордами для t_k . Касательные векторы на концах: $P'_1[1 \ 1 \ 1]$ и $P'_4[1 \ 1 \ 1]$. Вычислить 7 точек каждого сегмента сплайна. Сравнить результат для $\tau = 1/3, 2/3$ с примером 5-2, двумерным аналогом данной задачи на плоскости.

5-3. Используя координатные векторы из примера 5-2, вычислить и сравнить формы кривой для кубического сплайна для касательных векторов в точке P_1 для $P'_1[\sqrt{2}\cos\theta, \sqrt{2}\sin\theta]$ при $\theta = 0^\circ, 22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ$, сохраняя $P_2[1 \ 1]$. Сравнить эти результаты с получаемыми в случае, когда оба касательных вектора — переменные $P'_1[\sqrt{2}\cos\theta \sqrt{2}\sin\theta]$ и $P'_2[\sqrt{2}\cos\theta \sqrt{2}\sin\theta]$. Заметим, что величина касательных векторов постоянна, изменяется только их направление.

5-4. Используя координатные векторы из примера 5-2, вычислить и сравнить формы кривой кубического сплайна для касательных векторов в точках P_1 и P_2 для $P'_1[a \ a]$ и $P'_2[a \ a]$ при $a = 1/4, 1/2, 1, 2, 4, 8$. Заметим, что направление касательных векторов постоянно, изменяется только их величина.

5-5. Используя координатные векторы из примера 5-2, вычислить и сравнить формы кривых кубического сплайна при условии незакрепленного конца. Сравнить результаты с теми, которые были получены в задаче 5-2 и в примере 5-2.

5-6. Заданы пять координатных векторов на плоскости

$$[P] = \begin{bmatrix} 0 & 0 \\ 0.5 & 0.6 \\ 1.0 & 1.0 \\ 1.5 & 0.8 \\ 1.0 & 0 \end{bmatrix}$$

Положив касательные векторы концов равными $P'_1[1 \ 1]$ и $P'_2[0 \ -1]$, определить $P(0.5)$ для каждого из четырех сегментов кубического сплайна, используя нормализованную аппроксимацию, т. е. приняв $0 \leq t \leq 1$.

5-7. Повторить задачу 5-2, используя нормализованную аппроксимацию для t_k , т. е. принял $t_k = 1.0$. Сравнить результат с результатами, которые были получены в задаче 5-2, и с примером 5-2, аналогом этой задачи для плоскости.

5-8. Определить кубический сплайн кривой через пять координатных векторов $P_1[1 \pi/4 0], P_2[1 3\pi/4 1], P_3[1 5\pi/4 2], P_4[1 7\pi/4 3], P_5[1 \pi/4 4]$, где компоненты координатных векторов заданы в цилиндрических координатах, т. е. $P_i[r \theta z]$. Использовать условия незакрепленного конца.

5-9. Вычислить параболически смешанную кривую, проходящую через точки, данные в задаче 5-2. Использовать технологию, описанную в разд. 5-6. Сравнить результаты с результатами задачи 5-2 и с примером 5-2.

5-10. Вывести матрицу $[A]$ (см. равенство (5-60)) для параболически смешанной кривой, используя длины хорд как максимальные значения параметров, т. е. полагая

$$\begin{aligned} p(r) &= [r^2 \ r \ 1][B], & 0 \leq r \leq d_{13}, \\ q(s) &= [s^2 \ s \ 1][D], & 0 \leq s \leq d_{24}, \\ C(t) &= [t^3 \ t^2 \ t \ 1][A][G], & 0 \leq t \leq d_{23}, \end{aligned}$$

где d_{12}, d_{23}, d_{34} – длины хорд для P_1P_2, P_2P_3, P_3P_4 и $d_{13} = d_{12} + d_{23}, d_{24} = d_{23} + d_{34}$. Заметим, что это не совсем точно соответствует результату Оверхаузера.

5-11. Вывести результат Оверхаузера, используя матричные методы из разд. 5-7.

5-12. Определить кривую Безье, используя координатные векторы, заданные в примере 5-2, как определение многоугольника. Вычислить результат для следующих значений параметров: 0, 1/3, 2/3, 1.

5-13. Вычислить координатный вектор P_3 , задающий касательный вектор, проходящий вдоль кривой, соединяющей два сегмента кривой Безье, иначе определяемых как $P_1[0 \ 0 \ 0], P_2[2 \ 2 \ -2], P_4[4 \ 0 \ 0]$ и $Q_1[4 \ 0 \ 0], Q_2[6 \ -2 \ 1], Q_3[8 \ -3 \ 2], Q_4[10 \ 0 \ 1]$.

5-14. Для поверхности Безье, заданной вершинами многоугольника из примера 5-7, найти определяющие вершины многоугольника, требуемые для генерации поверхности Безье шестого порядка, эквивалентной поверхности четвертого порядка из примера 5-7. Порядок наращивать последовательными шагами.

5-15. Найти определяющие вершины многоугольника, которые разбивают поверхность Безье, определенную в примере 5-7, на две кубические поверхности Безье.

5-16. Применить пошагово технологию разбиения к поверхности Безье и определяющим ее вершинам многоугольника из примера 5-7 и показать на этом примере, что определяющий многоугольник и поверхность сходятся.

5-17. Повторить пример 5-11 для узлового вектора $[0 \ 0 \ 0 \ 2 \ 2 \ 3 \ 3 \ 3]$. Результат показан на рис. 5-38e. Сравнить этот результат с примером 5-11 и с рис. 5-38d.

5-18. Определить алгебраические отношения для функций неоднородного базиса, полученного с помощью узловых векторов $[0 \ 0 \ 0 \ 0.4 \ 2.6 \ 3 \ 3 \ 3]$ и $[0 \ 0 \ 0 \ 1.8 \ 2.2 \ 3 \ 3 \ 3]$. Результаты показаны на рис. 5-38b и с. Сравнить эти результаты с рисунками 5-38d и 5-38e.

5-19. Проиллюстрировать зависимость базисных функций кривых В-сплайна четвертого и пятого порядков, определенных шестью вершинами многоугольника. Использовать технологию, подобную рассмотренной в разд. 5-9.

5-20. Определить В-сплайн кривой четвертого порядка, используя координатные векторы примера 5-2 как определяющий многоугольник. Сравнить результат с поверхностью Безье, полученной в задаче 5-12.

5-21. Определить В-сплайн кривой третьего порядка, используя координатные векторы примера 5-2 как определяющий многоугольник. Использовать параметрические значения 0, 1/2, 1, 3/2, 2.

5-22. Для определяющего многоугольника, заданного в примере 5-13, определить псевдовершины, необходимые для совпадения конечных точек периодического В-сплайна с B_1 и B_4 соответственно. Сравнить результат с рис. 5-54.

5-23. Для определяющего многоугольника из примера 5-13 определить псевдовершины, необходимые, чтобы сделать касательные векторы концов кривой равными $[-1 \ 1]$ и $[1 \ -1]$ соответственно. Вычислить также начальную и конечную точки кривой. Результаты сравнить с рис. 5-54.

5-24. Используя аппроксимацию хордами, обсуждавшуюся в разд. 5-9, найти открытый В-сплайн поверхности четвертого порядка для определяющего многоугольника, заданного в примере 5-14. Сравнить результат с примером 5-14.

5-25. Используя аппроксимацию хордами, обсуждавшуюся в разд. 5-9, найти периодический В-сплайн поверхности третьего порядка для определяющего многоугольника, заданного в примере 5-14. Сравнить результат с примером 5-14.

5-26. Используя аппроксимацию хордами, обсуждавшуюся в разд. 5-9, найти периодический В-сплайн поверхности четвертого порядка для определяющего многоугольника, заданного в примере 5-14. Сравнить результат с примером 5-14 и результатом задачи 5-24.

5-27. Показать, что две кривые В-сплайна, заданные в примере 5-18 вершинами многоугольника $B_1[0 \ 0], B_2[1 \ 1], B_3[2 \ 1], B_4[3 \ 0]$ с альтернативными узловыми векторами $[X'] = [0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$ и $[X] = [0 \ 0 \ 0 \ 2 \ 4 \ 4 \ 4]$, идентичны. Указание: вычислите число точек на первой кривой $0 \leq t' \leq 2$ и на второй кривой $0 \leq t = 2t' \leq 4$ и сравните результаты.

5-28. Показать, что кривые В-сплайна разбиения из примера 5-18 такие же, как и исходная кривая. Указание: вычислите несколько точек двух кривых и постройте их на такой же шкале.

5-29. Показать, что кривые В-сплайна разбиения из примера 5-19 такие же, как и исходная кривая. Указание: вычислите несколько точек двух кривых и постройте их на такой же шкале.

5-30. Показать, что если после разбиения исходной кривой из примера 5-19 новую вершину многоугольника C_3 переместить в позицию вершины C_2 , то результатирующая поверхность будет иметь угловую форму в точке $C_2 = C_3$.

5-31. Для лучшего понимания сущности функции оценки вектора, такой, как $P(t) = [x(t) \ y(t)]$, построить графики функций $x(t)$ от t , $y(t)$ от t и $y(t)$ от $x(t)$ для В-сплайна кривой четвертого порядка ($k = 4$), определенной $B_1[0 \ 0], B_2[2 \ 2], B_3[4 \ 2], B_4[6 \ 0]$ с открытым узловым вектором $[0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$. Решить эту же задачу для В-сплайна кривой третьего порядка ($k = 3$) с открытым узловым вектором $[0 \ 0 \ 0 \ 1 \ 2 \ 2 \ 2]$.

5-32. Вывести аналитическое выражение для второй производной вдоль кривой рационального В-сплайна. Оценить результат для $t = 0$ и $t = n + k - 2$.

5-33. Сгенерировать кривые рационального В-сплайна для определяющих вершин многоугольника $B_1[0 \ 0]$, $B_2[1 \ 2]$, $B_3[2.5 \ 0]$, $B_4[2.5 \ 0]$, $B_5[2.5 \ 0]$, $B_6[4 \ 2]$, $B_7[5 \ 0]$, при $[H] = [1 \ 1 \ h_3 \ h_4 \ h_5 \ 1 \ 1]$ для значений $h_3 = h_4 = h_5 = 0, 0.25, 1, 5$. Сравнить результаты с рис. 5-66.

5-34. Рассмотреть пример 5-20 при $k = 4$.

5-35. Найти первую и вторую производные при $t = 0$ и $t = 3$ для определяющего многоугольника из примера 5-20.

5-36. Найти первую и вторую производные при $t = 0$ и $t = 2$ для $k = 4$ для определяющего многоугольника из примера 5-20.

5-37. Сгенерировать тор как поверхность рационального В-сплайна, полученную вращением окружности радиуса 1 с центром в точке $x = 5$, $z = 1$ относительно оси z (рис. 6-58).

5-38. Сгенерировать сферу радиуса 1 как поверхность рационального В-сплайна, полученную вращением полуокружности радиуса 1 с центром в точке $x = 0$, $z = 1$ относительно оси z (рис. 6-59).

ГЛАВА 6

6-1. Сгенерировать поверхности, показанные на рис. 6-6.

6-2. Сгенерировать поверхности вращения для кривых, вычисленных в задачах 5-6 и 5-9.

6-3. Сгенерировать «кружку», показанную на рис. 6-10.

6-4. Добавить точку к «кружке», показанной на рис. 6-10, используя изогнутую поверхность.

6-5. Сгенерировать изогнутую поверхность, показанную на рис. 6-16.

6-6. Определить тип квадратичной поверхности, заданной каждым из этих уравнений:

$$2x^2 - 2yz + 2zx - 2xy - x - 2y + 3z - 2 = 0,$$

$$x^2 + z^2 + 2xy + 2xz - 2yz - 2x + 4y - 4 = 0,$$

$$x^2 + 4y^2 + z^2 - 4yz - 12zx + 24xy + 4x + 16y - 26z - 3 = 0,$$

$$4x^2 + y^2 + z^2 - 4xy + 6z + 8 = 0,$$

$$4y^2 + 4z^2 + 4xy - 2x - 14y - 22z + 33 = 0.$$

6-7. Используя параметрическое представление, сгенерировать гиперболический параболоид.

6-8. Нарисовать поверхности, составляющие отдельные компоненты, т.е. $Q(u, v, x)$, $Q(u, v, y)$, $Q(u, v, z)$ (рис. 6-24) и полную поверхность $Q(x, y, z)$ для точки, прямой и плоскости.

6-9. Для линейчатой поверхности с ребрами в виде кривых, заданных как

$$z = 0, \quad y = 0.25 \cos(2\pi x), \quad 0 \leq x \leq 1,$$

$$z = 1, \quad y = 0.25 \sin(2\pi x), \quad 0 \leq x \leq 1,$$

определить поверхность в точке $Q(0.5, 0.5)$.

6-10. Построить билинейную поверхность, соответствующую четырем точкам $P(0, 0) = [0.25 \ 0]$, $P(1, 0) = [0.75 \ 0]$, $P(0, 1) = [0.75 \ 0.9]$, $P(1, 1) = [0.25 \ 0.8]$.

6-11. Построить билинейную поверхность, которая соответствует четырем точкам, определяемым противоположными концами диаметров двух единичных окружностей, лежащих на параллельных плоскостях, отстоящих друг от друга на единицу и поворачиваемых на 180° с интервалом 15° .

6-12. Сгенерировать полную линейчатую поверхность, показанную на рис. 6-28 и обсуждавшуюся в примере 6-10. Является ли такая поверхность развернутой?

6-13. Сгенерировать полную линейчатую поверхность, показанную на рис. 6-30 и обсуждавшуюся в примере 6-12. Показать, что угловая часть этой поверхности развернута.

6-14. Показать на примере, что плоская бикубическая поверхность Кунса получается в том случае, когда координатные, касательные векторы и векторы вращения лежат в одной плоскости.

6-15. Найти матрицу $[P]$, которая задает кусок цилиндрической бикубической поверхности Кунса.

6-16. Найти матрицу $[P]$, которая задает кусок общей бикубической поверхности Кунса.

6-17. Определить точку на поверхности с координатами $u = w = 0.5$ для бикубической поверхности Кунса со следующими координатными векторами: $P(0, 0) = [-100 \ 0 \ 100]$, $P(0, 1) = [100 \ -100 \ 100]$, $P(1, 1) = [-100 \ 0 \ -100]$, $P(1, 0) = [-100 \ -100 \ -100]$, u -касательные векторы $P^{1,0}(0, 0) = [10 \ 10 \ 0]$, $P^{1,0}(0, 1) = [-1 \ -1 \ 0]$, $P^{1,0}(1, 1) = [-1 \ 1 \ 0]$, $P^{1,0}(1, 0) = [1 \ -1 \ 0]$, w -касательные векторы $P^{0,1}(0, 0) = [0 \ -10 \ -10]$, $P^{0,1}(0, 1) = [0 \ 1 \ -1]$, $P^{0,1}(1, 1) = [0 \ 1 \ 1]$, $P^{0,1}(1, 0) = [0 \ 1 \ 1]$, и векторы вращения $P^{1,1}(0, 0) = [0 \ 0 \ 0]$, $P^{1,1}(0, 1) = [0.1 \ 0.1 \ 0.1]$, $P^{1,1}(1, 1) = [0 \ 0 \ 0]$, $P^{1,1}(1, 0) = [-0.1 \ -0.1 \ -0.1]$. Систематически изменить векторы вращения и касательные векторы и сравнить результаты.

6-18. Продолжить анализ параболически смешанной кривой на случай параболической смешанной поверхности [5-10]. Указание: использовать параболически смешанные кривые для четырех граничных кривых и стиль задания линейной поверхности Кунса со смешанными функциями, заданными как $F_1(t) = 1 - 3t^2 + 2t^2$, $F_2(t) = 3t^2 - 2t^3$. Заметим, что результирующая поверхность контролируется четырьмя угловыми точками и двумя присоединенными векторами в концах граничных кривых. Что является продолжением отдельных кусков? Какой порядок продолжения может поддерживаться через границы кусков? Каково продолжение влияния перемещения единственной точки поверхности, построенной из множества кусков?

6-19. Показать для бикубической поверхности Безье 4×4 , что для $u = w = 0$

$$Q_{uw}(0, 0) = 9[(B_{1,1} - B_{0,1}) - (B_{1,0} - B_{0,0})].$$

Сравнить результат с уравнением (6-60).

6-20. Найти (4×4) определяющие вершины прямоугольной сетки Безье, которые задают кусок бикубической поверхности Кунса из примера 6-13.

6-21. Сгенерировать поверхность Безье из примера 6-14, не используя матричных методов.

6-22. Показать, что ни гиперболический параболоид, ни параболический лист не являются развертывающимися поверхностями.

6-23. Для поверхности из примера 6-12 показать, что кривизна Гаусса равна нулю для плоской области и положительна в остальной области.

6-24. Определить кривизну Гаусса при $u = w = 0.5$ для поверхности из примера 6-14, показанной на рис. 6-39а–д. Сравнить результаты.

6-25. Сгенерировать закрытые периодические поверхности В-сплайна, показанные на рис. 6-46.

6-26. Сгенерировать комбинированную открытую и периодическую поверхность В-сплайна, показанную на рис. 6-47.

6-27. Сгенерировать закрытую периодическую тороидальную поверхность В-сплайна, показанную на рис. 6-48.

6-28. Рассчитать поверхность В-сплайна четвертого порядка для определяющей прямоугольной сетки 4×5 , заданной так:

$$\begin{array}{lll} B_{1,1}[0 \ 0 \ 100] & B_{2,1}[25 \ 0 \ 150] & B_{3,1}[50 \ 0 \ 100] \\ & B_{4,1}[75 \ 0 \ 50] & B_{5,1}[100 \ 0 \ 100] \\ B_{1,2}[0 \ 33 \ 150] & B_{2,2}[25 \ 33 \ 200] & B_{3,2}[50 \ 33 \ 100] \\ & B_{4,2}[75 \ 33 \ 50] & B_{5,2}[100 \ 33 \ 50] \\ B_{1,3}[0 \ 66 \ 50] & B_{2,3}[25 \ 66 \ 25] & B_{3,3}[50 \ 66 \ 100] \\ & B_{4,3}[75 \ 66 \ 150] & B_{5,3}[100 \ 66 \ 150] \\ B_{1,4}[0 \ 100 \ 100] & B_{2,4}[25 \ 100 \ 50] & B_{3,4}[50 \ 100 \ 100] \\ & B_{4,4}[75 \ 100 \ 150] & B_{5,4}[100 \ 100 \ 100] \end{array}$$

Определить вычислением вручную точку в центре параметрической сетки поверхности 11×15 .

Написать программу для вычисления двух диагональных параметрических прямых $u = 2w$ и $u = 2(1 - w)$. Составить список точек от $(0, 0)$ до $(2, 1)$ для $u = 2w$ и от $(0, 1)$ до $(2, 0)$ для $u = 2(1 - w)$. Составить список значений $15u$ и $11w$.

Вычислить и изобразить параметрическую поверхность 11×15 . Использовать подходящее зрительное преобразование. Указание: вывести результаты в файл в стандартном графическом формате (.sgf) или в стандартном полигональном формате (.spf) и использовать трехмерные преобразования, разрабатываемые в спецификации 3-1 для изображения поверхности.

Приложение F

СПЕЦИФИКАЦИИ ПРОГРАММ

Спецификации программ разработаны не только для пояснения концепций и методов, представленных в книге, но также для передачи некоторого опыта создания достаточно сложных программ машинной графики. Хорошо выполненные, они могут составить инструментарий полезных графических программ. Законченные программные проекты дают в результате полезные программы общего назначения. Спецификации сгруппированы по главам книги.

Особенно рекомендуются две из этих спецификаций. Первая – это научная программа черчения (спецификация 2-4). Написание хорошей научной программы черчения – нетривиальная задача, однако она вполне доступна студентам старших курсов и дипломникам. Ее можно рекомендовать в качестве одного из двух курсовых проектов для студентов выпускных курсов (другим проектом может быть трехмерный манипулятор (спецификация 3-1)). Студенты старших курсов могут выполнить подобную разработку примерно в течение трех недель.

Вторая спецификация – это трехмерный манипулятор (спецификация 3-1). Трехмерный манипулятор позволяет изображать и совершать различные преобразования (вращение, перемещение, масштабирование, проецирование и т.д.) над трехмерными объектами. Трехмерный манипулятор значительно облегчает задачу визуализации объектов. Кроме того, он улучшает возможности визуализации в трех измерениях. В довершение всего, он полезен для лучшего усвоения материала по трехмерным кривым и поверхностям (гл. 5 и 6). Для студентов старших курсов он может служить вторым из двух курсовых проектов. Дипломник должен справиться с этой работой в течение примерно трех недель.

При проектировании и реализации этих программ особое внимание должно уделяться концепциям структурного программирования и особенно модуляризации программ. Каждая из предложенных спецификаций достаточно ясно распадается на умеренное количество графических инструментальных модулей. Например, большинство графических инструментов для трехмерного манипулятора даны в виде псевдокода в приложении G. Язык реализации несуществен. К настоящему времени все спецификации реализованы студентами на различных вычислительных машинах и различных языках программирования.

Необходима хорошая документация. Пример хорошего документирующего заголовка показан в табл. F-1. Необходимо также подробное документирование внутри программы.

Таблица F-1 Заголовок документации программы

имя:
программист:
дата:
история:
назначение:
метод:
система:
язык:
вызываемые подпрограммы:
файлы ввода:
команда компилятора:
команда компоновщика:
аргументы вызова:
список переменных:

Так как ряд спецификаций задуман для работы в интерактивном режиме, уместно сказать несколько слов об интерфейсе пользователя. В графической среде, не рассчитанной на работу в реальном времени, в качестве интерактивного устройства наиболее часто используется обычная алфавитно-цифровая клавиатура. Для такой среды простым в изучении, быстрым и эффективным является хорошо разработанный интерфейс клавиатуры.

Если не требуется специфический числовый ввод, интерфейс клавиатуры принимает для ввода буквенные слова. Цифровые списки ответов неприемлемы вследствие того, что пользователь мыслит в терминах слов и фраз, а не цифровых элементов или списков. Например, если желаемое действие — это поворот объекта, то пользователь думает о повороте, а не об элементе 3 из списка опций.

Далеко не все пользователи хорошо печатают на машинке. Поэтому число символов, необходимое для ввода команды, должно быть сведено к минимуму. Однако те, которые хорошо печатают, не должны быть поставлены в неудобное положение, раздражены использованием странных последовательностей символов. Одно из возможных решений состоит в том, чтобы допустить полное слово в качестве ввода и затем усечь его до минимального количества символов, необходимого для уникального определения команды. Например, команда rotate (поворот) усекается до первых двух символов, т. е. го. Допустимое усечение, которое дает уникальную последовательность символов, выделяется скобками, например (го)rate. Численный ввод должен допускаться в свободном формате. Например, в числах с плавающей точкой не обязательно присутствие десятичной точки, т. е. числа 30 и 30. эквивалентны.

Следовало бы обеспечить командное меню. Важный вопрос — как и когда оно поддерживается. Его следовало бы высвечивать один раз при иницировании выполнения программы и затем по запросу. Нецелесообразно высвечивать меню после приема каждой очередной команды, так как это раздражает и к тому же поглощает много времени. Полезно, чтобы командное меню полностью помещалось на экране. На каждом уровне программы, обрабатывающей команды, должно быть обеспечено соответствующее командное меню. Требуется, чтобы для

высвечивания командного меню использовалась единственная командная строка. Такая командная строка состоит из единственного слова или символа. Например «?» или «Option?» или «Command?».

Существует столько же способов запроса командного меню или меню (экрана) помощи, сколько существует программистов. Например, можно набрать h или help или ? или ?? или все, что угодно. Подбор подходящей команды часто оказывается довольно интересным занятием. Простое решение — принимать любой из этих запросов или возвращать простой carriage (символ возврата каретки). Инструментарий прост. Если при разборе командного ввода не обнаруживается допустимой команды, то высвечивается подходящее для этого уровня командное меню (экран подсказки).

При использовании концепции, называемой «командный стек», эффективность интерфейса значительно возрастает. Лучше всего ее смысл проясняется из примера. В спецификации трехмерного манипулятора (спецификация 3-1) команда поворота запрашивает описание оси поворота (ось x, y, z или произвольная) и угла поворота в градусах. Процессор команд верхнего уровня принимает команду rotate (поворот) и вызывает процессор команд нижнего уровня, который запрашивает и вводит оси и угол. Командный стек позволяет опытному пользователю набрать команду в виде

го x 30.

Здесь параметры выделены пробелами. Командный процессор верхнего уровня разбирает строку ввода, определяет, является ли первая команда го допустимой, и, если это так, передает оставшиеся команды командному процессору нижнего уровня. Командный стек позволяет опытному пользователю выполнять серии манипуляций, минимизируя ввод и время ожидания. Начинающий пользователь имеет доступ к таким же возможностям. Например, с помощью командного стека

```
sc o 2
го x 30
го у -45
dr
```

объект масштабируется по всем координатам с коэффициентом 2, поворачивается вокруг оси x на 30°, поворачивается вокруг оси у на 45°, а результат отображается в плоскости z = 0 с использованием ортографической проекции.

В завершение отметим, что интерфейс пользователя должен быть согласован на всех уровнях. Следует придерживаться «принципа наименьшего удивления». Иначе говоря, при наличии множества ответов на команду выбор должен основываться на том, «какой ответ вызовет наименьшее удивление пользователя».

ГЛАВА 2

Первые две спецификации были разработаны для того, чтобы ознакомить студентов с локальным графическим пакетом и заложить основу базисного инструментария графических модулей. Третья спецификация служит для приобретения опыта работы с оконной техникой на графическом устройстве и является основой

для разного рода управляющих преобразований. Четвертая относится к развитию программы визуализации научных результатов.

2-1. Разработать и реализовать программу рисования замкнутого многоугольника с вершинами $P_1[0 \ 0]$, $P_2[2 \ -2]$, $P_3[2 \ 1]$, $P_4[0 \ 3]$, $P_5[-2 \ 1]$, $P_6[-2 \ -2]$, выбрав направление обхода против часовой стрелки. Использовать команды окна и/или области вывода графического устройства, чтобы многоугольник заполнял все пространство экрана. Будьте внимательны при построении геометрических отношений, одна единица по оси x должна быть равна одной единице по оси y . Включить оси, проходящие через начало координат. Разбить оси на единичные интервалы, расставить маркировку на осях.

Разработать и включить в программу инструментальные модули для:

Рисования осей через точку $x0, y0$, от $x\text{-min}$ до $x\text{-max}$ и от $y\text{-min}$ до $y\text{-max}$. Размещения левых, правых или центральных осевых маркеров на осях x и y , определенной длины, расположенных через определенный интервал, начиная от определенной точки.

Разметки осевых маркеров подходящими значениями.

Рисования произвольного многоугольника. (Приложение A-7.)

2-2. Разработать и усовершенствовать программу генерации и отображения кривых, описанных параметрическими уравнениями

$$\text{a. } x = \{2 + 7 \cos(\sin(\theta) + \sin(121\theta))\} \cos(\theta) \quad \pi/2500 \leq \theta \leq 2\pi$$

$$y = \{2 + 7 \cos(\sin(\theta) + \sin(121\theta))\} \sin(\theta)$$

$$\text{b. } x = \{11/10 + \cos(8\theta)^{19}\} \cos(\theta) \quad \pi/2000 \leq \theta \leq 2\pi$$

$$y = \{1.1 + \cos(8\theta)^{19}\} \sin(\theta)$$

$$\text{c. } x = \{1 + 2 \sin(17\theta/4)\} \cos(\theta) \quad \pi/500 \leq \theta \leq 8\pi$$

$$y = \{1 + 2 \sin(17\theta/4)\} \sin(\theta)$$

$$\text{d. } x = \{-\theta \sin(\pi\theta)/8\} \cos(\theta) \quad 9\pi/1000 \leq \theta \leq 36\pi$$

$$y = \{-\theta \sin(\pi\theta)/8\} \sin(\theta)$$

$$\text{e. } x = \{\sin(13\theta/4)\} \cos(\theta) \quad \pi/100 \leq \theta \leq 100\pi$$

$$y = \{\sin(13\theta/4)\} \sin(\theta)$$

Используя команды работы с окнами и/или управления областями вывода, увеличить эти фигуры до размеров экрана. Будьте внимательны при построении геометрических отношений, одна единица по оси x должна быть равна одной единице по оси y . Включить оси, проходящие через начало координат. Разбить оси на единичные интервалы, расставить маркировку на осях.

Разработать и включить в программу инструментальные модули для:

Рисования осей через точку $x0, y0$, от $x\text{-min}$ до $x\text{-max}$ и от $y\text{-min}$ до $y\text{-max}$.

Размещения левых, правых или центральных осевых маркеров на осях x и y , определенной длины, расположенных через определенный интервал, начиная от определенной точки.

Разметки осевых маркеров подходящими значениями.

Рисования фигуры. (Приложение A-7.)

После преобразования

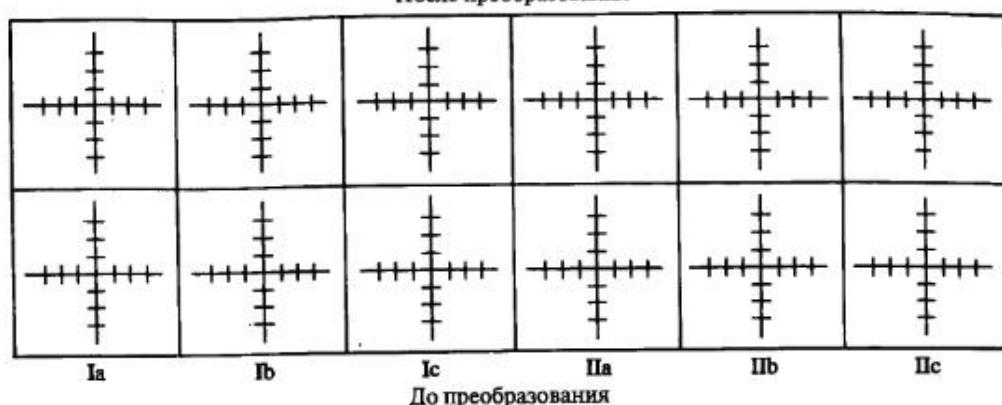


Рис. F-1

2-3. Выполнить следующие преобразования для двух прямых P_1P_2 и P_3P_4 , заданных $P_1[-2 \ -1], P_2[1 \ 2], P_3[-1 \ -1], P_4[-1 \ 2]$:

- a) поворот на -60° ;
- b) зеркальное отражение относительно прямой $y = x$;
- c) масштабирование с коэффициентом 2.

Выполнить следующие преобразования для второй пары прямых P_5P_6 и P_7P_8 , определенных через $P_5[-1 \ -1], P_6[2 \ 2], P_7[-3 \ -1], P_8[0 \ 2]$:

- a) поворот на 60° , а затем зеркальное отражение относительно оси x ;
- b) зеркальное отражение, относительно оси x , после поворота на 60° ;
- c) преобразования $\begin{bmatrix} -1 & 3 \\ 2 & 4 \end{bmatrix}$.

Получить указанные результаты с помощью вычислений вручную.

Разработать и реализовать программу отображения результатов на один экран в формате, показанном на рис. F-1. Три левые клетки в нижнем ряду содержат первую пару прямых до преобразования. Три правых клетки содержат вторую пару прямых до преобразования. В верхнем ряду находятся соответствующие преобразованные прямые. Каждая клетка содержит оси с разметкой. Так как пространство ограничено, может оказаться желательным не наносить метки значений маркеров. Однако необходимо указать значение каждого маркера в каждой клетке. Соответствующее изображение должно быть помечено как индикатор.

При разработке и реализации этих программ добросовестный пользователь обратится к концепции окон и областей вывода графического устройства (разд. 1-4 и приложение A-4). Если абстрагироваться от размера окна, размещения клеток на экране и конкретных операций преобразования, то действия, требуемые для генерации каждого бокса, представляются идентичными. Таким образом, главная программа выглядит так:

```

for i = 1 to 6
    применить преобразование, если надо
    определить размер окна
call window
for j = 1 to 2
    call viewport(j)
    call box
    call axes
    call tick
    call label
    call drawlines(j)
next j
next i

```

Разработать и включить в программу инструментальные модули для

- определения требуемого размера окна;
- рисования клетки вокруг окна;
- рисования осей через точку x_0, y_0 , от x_{\min} до x_{\max} и от y_{\min} до y_{\max} ;
- размещения левых, правых или центральных осевых маркеров на осях x и y определенной длины, расположенных через определенный интервал, начиная от определенной точки;
- разметки осевых маркеров подходящими значениями;
- рисования прямых.

Позаботьтесь также о включении общепринятых модулей двумерного поворота, отражения и масштабирования на плоскости (приложение G). Используйте эти модули в программах для генерации матриц необходимых преобразований. Когда требуются комбинированные преобразования (задания а и б для второй пары линий), нужно перемножить матрицы перед применением их к исходной паре прямых.

2-4. Разработать и реализовать программу визуализации научных данных, которая принимает входной поток данных в форме двумерного файла .sgf (табл. А-2 из приложения А-6). При решении данной задачи воспользуйтесь комментариями относительно интерфейса пользователя, данными в начале этого приложения. Программа должна удовлетворять следующим характеристикам и условиям:

ввод
черчение

Ввод данных из двумерного файла .sgf.
Отображение графиков в соответствии с пределами, метками, аннотациями и другими спецификациями. По умолчанию применяется автоматическая установка пределов, данные присоединяются в виде единой линии, нет меток, клетка без сетки и аннотация отсутствуют. (См. ниже.)

Помечаются график, ось x и ось y .
Обе оси линейные, одна или обе оси логарифмические. Требуются точные значения пределов по осям. Допустимые пределы по осям начинаются или заканчиваются в $0, \pm 1, \pm 2, \pm 5$. Таким образом, оси могут иметь пределы $0 \rightarrow 1, 0 \rightarrow 2, 0 \rightarrow 5, -1 \rightarrow 1, -1 \rightarrow 5, -2 \rightarrow 5$ и т.д. Случайные пределы, например $0 \rightarrow 1.736$, не допускаются, за исключением специально установленных командой range (пределы). Оси помечаются в соответствии с порядком значения 1. Величина метки ука-

зывается как множитель, расположенный следом за последней меткой оси, который задается в форме $\times 10^2$. Логарифмические оси размечаются только в интервалах, соответствующих 1, 5 и 10.

Изменение пределов графика. Запрашиваются минимальное и максимальное значения x и y .

Автоматически определяются приближенные значения пределов по осям x и y . (См. оси.)

Выбирается тип линий для определяемой кривой. Представлено меню допустимых типов. Минимально необходимое множество — это штриховая линия, непрерывная линия, точка в каждой позиции графика, звездочка в каждой позиции графика, плюс в каждой позиции графика или \times в каждой позиции графика. Допустимы комбинации позиций графика и типов линий.

Выбор типа маркера пометки. Представлено меню возможных типов. Минимально необходимое множество — это пометки на левой или правой стороне осей, если смотреть из начала координат в положительном направлении осей, пометки на обеих сторонах осей, длинные пометки в более важных точках деления и короткие в менее важных.

Способ высвечивания метки оси u : метка расположена по вертикали так, что читается сбоку, или так, что читается сверху вниз.

Включение/выключение сетки.

Построение графического квадрата, сохраняющего геометрические отношения, т.е. единица по оси x равна единице по оси y .

Переключение способа отображения графика на экране: график отображается в клетке с разметкой по всем четырем сторонам или график отображается только с двумя координатными осями.

Изменение размеров графика путем указания масштабных множителей по осям x и y .

Задает цвет отдельно каждой кривой графика из смешанной палитры красного, зеленого и синего цвета. Значения цвета задаются из интервала $0 \leq r, g, b \leq 1$, где 0 — отсутствие данного цветового компонента, 1 — максимальный вклад.

Размещает соответствующую текстовую аннотацию в произвольной позиции графика. Возможно как автоматическое, так и интерактивное размещение аннотации. Опция автоматического размещения служит для ввода строк и отображения их в левом верхнем углу графика. Опция интерактивного размещения позволяет позиционировать каждую введенную строку аннотации запросом координат x, y для левого нижнего угла строки. Представлено меню возможных шрифтов для вывода аннотации. (См. шрифты)

Позволяет отменить некоторые предыдущие команды, например метка, аннотация, пределы и т.д., в частности ввод последней команды.

Перезадает всем опциям значения по умолчанию. (См. черчение.)

Выбор одного из типов шрифта. Представлено меню возможных шрифтов.

Делает твердую копию изображения графика.

Представлено меню доступных опций, включающее вывод в файл в формате .sdf (табл. А-1 из приложения А-5).

Сохраняет текущий график в файле.

Загружает из файла ранее сохраненный график. После этого возможны любые модификации этого графика.

Генерирует кривую, проходящую через заданные точки кривой. Представлено меню возможных методов построения таких кривых, которое должно включать следующие методы: полиномов, степенной и экспо-

диапазон

автодиапазон

тип линии

пометки

у-метка

сетка
квадрат

клетка

масштаб

цвет

аннотация

отмена

переустановить
шрифт

твердая
копия

сохранить
восстановить

округлить

выход

ненциональный наименьших квадратов, кубический сплайн, сплайн Бе́зье и В-сплайн.

Прекращает работу программы и возвращает управление операционной системе.

ГЛАВА 3

Здесь приведена единственная спецификация – разработка и развитие трехмерного манипулятора. Предполагается, что объекты отображаются с использованием каркасного представления. Разрешен ввод множественных (составных) объектов. Для успешного выполнения задания требуется разработка необходимых модулей для трехмерного поворота, переноса, масштабирования и т. д., а также для получения ортографической, косоугольной, перспективной проекций и т. д.

3-1. Разработать и реализовать программу трехмерного манипулятора, который вводит данные из файла как в формате .sgf, так и в формате .spf, описанных в табл. А-2 и А-3 из приложения А-6. Комментарии по поводу интерфейса пользователя, приведенные в начале этого приложения, справедливы и при разработке этой программы. Программа должна удовлетворять следующим характеристикам и условиям:

ввод
разделитель
поворот

Ввод данных из файла как в формате .sgf, так и в формате .spf. Спецификации объектов разделяются с помощью знака #. Поворот объектов относительно координатных осей x , y , z или относительно произвольной оси вращения. Произвольная ось вращения определяется как с помощью точки и трех направляющих косинусов, так и двумя точками на осях координат. Произвольный поворот получается из поворотов относительно координатных осей. Поворот может задаваться как относительно мировых координатных осей, так и относительно локальных осей тела.

Параллельный перенос объектов в направлении x , y или z . Зеркальное отражение объектов относительно плоскостей $x = 0$, $y = 0$ или $z = 0$.

Масштабирование объектов локально, т. е. отдельно в направлении x , y или z , или глобально, т. е. одинаково по всем направлениям. Определены два типа трехмерных осей: оси тела и фиксированные оси. В любой момент времени могут отображаться как одни из них, так и те, и другие сразу. Размеры переменные. Маркеры разметки могут быть включены или выключены. Оси тела фиксированы в центре объекта или группы объектов. Они перемещаются, т. е. врачаются, переносятся, отражаются вместе с объектом. Фиксированные оси располагаются в точке начала мировой системы координат. Они являются субъектом преобразования кадра, т. е. изометрической, диметрической, косоугольной или перспективной проекции.

Проекция объектов на плоскость $z = 0$ – изометрическая, диметрическая, косоугольная, кабине, кавалье или центральная перспективная проекция. Изометрические и диметрические преобразования поддерживают выбор положительного или отрицательного поворота относительно осей x , y . Диметрическая и косоугольная проекции допускают переменный множитель искажения. Косоугольная, а также проекции кабине и кавалье допускают переменный угол наклона относительно горизонтали. Центральная перспективная проекция на плоскость $z = 0$

перенос
отражение

масштаб

оси

проекция

трансфокация	с переменным центром выполняется только для центров, расположенных на оси z .
исходный	Трансфокация (наезд) камеры относительно кадра изображения. Замечание: это не масштабирование объектов.
рисовать	Восстанавливает исходные данные объекта.
вывод	Рисует преобразованные объекты, используя ортографическую проекцию на плоскость $z = 0$.
сохрматр	Сохраняет текущие преобразованные данные точек в файле формата .sgf или .spf.
тень	Сохранить текущую матрицу преобразования в файле.
невидимые	Создать изображение с тенями. ¹ Требуется, чтобы данные вводились в формат .spf.
твердая копия	Создать изображение с невидимыми линиями/поверхностями. ² Требуется, чтобы данные вводились в формате .spf.
центр	Делает твердую копию изображения. Представлено меню доступных устройств вывода. Требуется возможность вывода в файл формата .sdf. (Табл. А-1 из приложения А-5.)
объект #	Изображение объектов центрируется относительно геометрического центра данных.
4 вида	Включает/выключает отдельный объект или группу объектов, задаваемую как #.
окружение	Создает четыре вида объектов: ортографические фронтальные проекции, виды сбоку и сверху в нижнем левом, нижнем правом и верхнем левом квадрантах соответственно, плюс текущую проекцию в верхнем правом квадранте.
выход	Удаляет с экрана любые элементы окружения, например меню. Прекращает работу программы и возвращает управление операционной системе.

Программа структурирована так, чтобы все матрицы преобразований применялись до выполнения команды рисования или какой-либо команды проецирования. Перед выполнением любой из этих команд комбинированная матрица преобразования применяется к базе данных, а результирующие преобразованные двумерные координатные векторы рисуются на устройстве вывода. Исходная трехмерная база данных не модифицируется командами рисования или проекции.

Главная программа состоит из условного цикла в виде:

```
Выход = произвольная строка
while Выход <> q then
    print Опцию ? на экран
    input Ответ от клавиатуры
    truncate Ответ до первых двух символов
    convert Ответ в регистр строчных букв
    if Ответ = ro then call подменю поворота
    if Ответ = re then call подменю отражения
    if Ответ = pr then call подменю проекции
    :
    print главный экран помощи
end while
```

¹ См. Д. Роджерс, Алгоритмические основы машинной графики. — М.: Мир, 1989.

² См. предыдущую ссылку.

Типичная программа подменю могла бы выглядеть так:

```
Выход = произвольная строка
while Выход <> q then
    print Вращение относительно какой оси?
    input Ответ от клавиатуры
    truncate Ответ до первых двух символов
    convert Ответ в регистр строчных букв
    if Ответ = x then call программу x-поворота
    if Ответ = y then call программу y-поворота
    if Ответ = z then call программу z-поворота
    if Ответ = g then call программу произвольного поворота
    print локальный экран помощи
end while
```

Опираясь на эту базовую концепцию, легко понять, что программа имеет по крайней мере три уровня действий. Первый — выбор основной операции, например поворот, второй — выбор ее параметров, например ось поворота, третий — выполнение операции.

При необходимости возможности, описанные в спецификации, можно сократить путем удаления требований к программе.

В качестве примера рассмотрим Т-блок, определенный ниже

0 0 4	A
4 0 4	B
4 2 4	C
3 2 4	C
3 4 4	E
1 4 4	F
1 2 4	G
0 2 4	H
0 0 0	I
4 0 0	J
4 2 0	K
3 2 0	L
3 4 0	M
1 4 0	N
1 2 0	O
0 2 0	P

Разработать единую матрицу преобразования, которая:

- сокращает размер на 1/2;
- поворачивает относительно оси x на 30° ;
- поворачивает относительно оси y на -30° ;
- переносит на -2 единицы в направлении y ;
- выполняет центральную перспективную проекцию на плоскость $z = 0$ с центром проекции $z = 10$.

Сначала разработать, потом применить комбинированную матрицу преобразования для этих операций к базе данных путем вычисления вручную. Построить

результат на графической бумаге. Вычислить и построить некоторые невидимые точки. Использовать результаты этих ручных вычислений для проверки операций трехмерного манипулятора.

ГЛАВА 4

Единственная приведенная здесь спецификация основывается на расширении материала разд. 4-9. Предлагаемое расширение дано в задачах с 4-6 по 4-10. Подобные конструкции требуются в системах компьютерной поддержки черчения и во многих управляющих приложениях.

4-1. Разработать и реализовать программу интерактивного задания геометрии, определенной в разд. 4-9, и/или ее расширения, определенного в задачах с 4-6 по 4-10. Будьте внимательны при построении корректных геометрических отношений для отображения результатов; например, окружность должна выглядеть как окружность, а не как эллипс. Комментарии по поводу интерфейса пользователя, приведенные в начале этого приложения, справедливы и при разработке этой программы. Программа должна удовлетворять следующим характеристикам и условиям:

line	Определяет отрезок по его двум конечным точкам.
circle	Определяет окружность по ее центру и радиусу (h, k) .
add	Добавляет окружность или отрезок к геометрии.
delete	Удаляет окружность или отрезок из геометрии.
move	Перемещает ранее определенные окружность или отрезок.
oxes	Включает/выключает оси.
ticks	Включает/выключает маркеры разметки осей.
tables	Включает/выключает метки маркеров разметки осей.
solution	Выбирает подходящее из возможных решений, используя такие концепции, как левый, правый, близкий, дальний, внутри, вне, см. разд. 4-9.

В качестве частей программы разработать и включить инструментальные модули для

- рисования осей, проходящих через точку $x0, y0$, от $x\text{-min}$ до $x\text{-max}$ и от $y\text{-min}$ до $y\text{-max}$;
- левого, правого или центрального расположения маркеров разметки осей x и y , определенной длины, начиная с заданной позиции через заданный интервал;
 - нанесения на маркеры разметки подходящих значений;
 - генерации и рисования окружности и дуги окружности.

Обязательно также разработайте модули необходимых двумерных преобразований, например поворота, переноса, зеркального отражения и т.д., необходимых для размещений в стандартной ориентации. Когда требуются множественные преобразования, перемножайте матрицы перед их применением к геометрической базе данных. Разработайте необходимые модули для поддержки движения по дереву решений, для выбора необходимого решения, для определения, является ли выбранное решение физически возможным, и выполнения необходимых вычислений для поиска некоторых неизвестных геометрических величин.

ГЛАВА 5

Здесь приведена спецификация программы генерации и манипуляции с кривыми. Каждая из этих кривых обсуждалась в гл. 5. В частности, речь шла о кубическом сплайне, параболически смешанном, сплайне Безье и В-сплайне как рациональных, так и нерациональных. Они используются в качестве базиса программы. Включение возможностей генерации составной кривой нескольких типов делает особенно удобным сравнение.

5-1. Разработать и реализовать программу генерации и манипуляции трехмерными кривыми, которая принимает входные данные в формате трехмерного файла .sgf (табл. А-2 из приложения А-6). Сгенерированная трехмерная кривая строится с использованием ортографической проекции на плоскость $z = 0$. Комментарии по поводу интерфейса пользователя, приведенные в начале этого приложения, справедливы и при разработке этой программы. Программа должна удовлетворять следующим характеристикам и условиям:

ввод	Ввод данных из файла в трехмерном формате .sgf.
тип	Выбор типа кривой. Представлено меню возможных типов кривых, например кубический сплайн, параболически смешанный, сплайн Безье, В-сплайн. Для кубического сплайна реализованы как нормализованная, так и хордовая аппроксимации и, по крайней мере, условия жесткого и свободного конца. Для параболически смешанной кривой реализована аппроксимация для α и β с переменным значением α , $\alpha = 0.5$ по умолчанию. Для кривых Безье и В-сплайна реализованы рациональный и нерациональный варианты, а также возможности повышения степени и подразбиения.
многоугольник	Рисует прямые линии между введенными точками данных. Для кривых Безье и В-сплайна это задание определяющего многоугольника.
метка	Отображает метки следом за отдельными точками входных данных, например P_1 , P_2 , P_3 и т. д.
добавить	Добавляет точку данных. Обязательна возможность добавить точку перед первой точкой данных.
удалить	Удаляет точку данных.
переместить	Перемещает точку данных.
очистить	Определяет, будет или нет экран очищаться перед рисованием первой кривой.
оси	Включает/выключает графические оси. Оси включают подходящие маркеры разметки и метки.
диапазон	Изменяет пределы графика. Запрашиваются минимальное и максимальное значения для x и y .
автодиапазон	Автоматически определяет подходящие пределы для x и y .
типа линии	Выбор типа линии для заданной кривой. Представлено меню возможных типов. Как минимум, оно должно включать штриховую и сплошную линии.
кривая #	Включает/выключает отдельную кривую или группу кривых, определенных как #.
восстановить #	Восстанавливает кривые, определяемые как #, из базы данных программы.
исходные	Восстанавливает исходные данные кривой.
печать	Печатает на экране конструктивные данные кривой, например определяющий многоугольник для кривой Безье или В-сплайна.

твердая	Делает твердую копию текущего изображения.
копия	Представлено меню доступных устройств вывода. Необходима возможность вывода в файл в формате .sdf. (Табл. А-1 из приложения А-5.) Сохраняет трехмерные точки данных генерированной кривой в файле формата .sgf для последующего отображения на экран и преобразований с помощью трехмерного манипулятора. (Спецификация 3-1.)
вывод	Прекращает работу программы и возвращает управление операционной системе.

При необходимости возможности, описанные в спецификации, можно сократить путем удаления требований к программе.

Используйте вычисления из примера 5-2 и задач 5-5, 5-7, 5-9, 5-10, 5-12, 5-20 и 5-21 для проверки работы программы генерации и преобразований трехмерных кривых.

5-2. Добавить к спецификации 5-1 возможности для кривых Безье и В-сплайна, как нерациональных, так и рациональных. Данные вводятся из файла в формате .sgf.

ГЛАВА 6

Здесь приведена спецификация программы генерации и манипуляции для поверхности. Она очень похожа на программу генерации и преобразований для кривой из гл. 5 и обладает многими аналогичными характеристиками. Она также может быть построена как расширение программы трехмерного манипулятора (спецификация 3-1). В этом случае генерация и манипуляция поверхностью является просто дополнительной функцией.

Поверхности, обсуждавшиеся в гл. 6, например билинейные, правильные, линейные или бикубические поверхности Кунса, поверхности Безье или В-сплайны, как нерациональные, так и рациональные, используются в качестве базиса программы. Включение возможностей генерации и отображения составных поверхностей делает особенно удобным исследование эффектов воздействия различных вариантов входных данных.

6-1. Разработать и реализовать программу генерации и манипуляции трехмерными поверхностями, которая получает входные данные в формате трехмерного файла .sgf, описанного в табл. А-2 приложения А-6 или точки многоугольной сетки в форме файла .prp для поверхности Безье или В-сплайна (табл. D-1 приложения D). Сгенерированная поверхность строится с использованием подходящей проекции на плоскости $z = 0$, например изометрической, диметрической, кабине или кавалье. Результаты выводятся в формате .sgf или в формате .spf и могут быть использованы впоследствии. Комментарии по поводу интерфейса пользователя, приведенные в начале этого приложения, справедливы и при разработке данной программы. Программа должна удовлетворять следующим характеристикам и условиям:

ввод	Ввод данных из файла в трехмерном формате .sgf.
поверхность	Генерирует поверхность. Для поверхностей Безье и В-сплайна реализованы как нерациональный, так и рациональный варианты, а также возможности увеличения порядка и степени разбиения. (Задачи 6-17 и 6-28 предлагаются в качестве тестовых поверхностей.)

границы диагонали параметризация	Вычисляет и отображает четыре граничные кривые поверхности. Вычисляет и отображает две диагональные кривые поверхности. Вычисляет и отображает ортогональную сетку изопараметрических линий на поверхности. Запрашивает число линий в каждом параметрическом направлении.
рисовать многоугольник	Отображает поверхность с использованием подходящей проекции, например изометрической, диметрической, кабине или кавалье. Рисует прямые линии между точками входных данных. Применяется для поверхностей Безье и В-сплайна, чтобы показать определяющий многоугольник.
оси	Определены два типа трехмерных осей: оси тела и фиксированные оси. В любой момент времени могут отображаться как одни из них, так и те и другие сразу. Размеры перемещенные. Маркеры разметки могут быть включены или выключены. Оси тела фиксированы в центре объекта или группы объектов. Они перемещаются, т. е. врачаются, переносятся, отражаются вместе с объектом. Фиксированные оси располагаются в точке начала мировой системы координат. Они являются субъектом преобразования кадра, т. е. изометрической, диметрической, косоугольной или перспективной проекции.
метка	Отображает метки вслед за отдельными точками входных данных, например P_1 , P_2 , P_3 и т.д.
модифицировать	Модифицирует характеристики поверхности. Представлено меню возможных модификаций, одной из которых является перемещение координатных векторов. Для бикубической поверхности Кунса также включены модификации касательных векторов и векторов вращения. Трансфокация камеры относительно кадра изображения. Замечание: это не масштабирование поверхности.
очистить	Определяет, будет ли очищаться экран перед рисованием текущей поверхности.
исходные печать	Восстанавливает исходные данные поверхности. Печатает на экране конструктивные данные поверхности, например определяющий многоугольник для поверхности Безье и В-сплайна.
твердая копия	Делает твердую копию текущего изображения. Представлено меню доступных устройств вывода. Необходима возможность вывода в файл в формате .sgf. (Табл. A-1 из приложения A-5.)
вывод	Сохраняет трехмерные точки данных сгенерированной поверхности в файле в трехмерном формате .sgf для последующего отображения на экран и манипулирования с помощью трехмерного манипулятора.
выход	Прекращает работу программы и возвращает управление операционной системе.

При необходимости возможности, описанные в спецификации, можно сократить путем удаления требований к программе.

Использовать вычисления из примеров 6-9, 6-10, с 6-12 по 6-15 и 6-18 и задач с 6-9 по 6-11 для проверки операций программы генерации и манипуляций с поверхностями.

6-2. Добавить к спецификации 5-1 возможности для трехмерных поверхностей. Данные вводятся из файла в формате .sgf. (Табл. A-2 из приложения A-6.)

Приложение G

АЛГОРИТМЫ

В этом приложении приведено несколько полезных алгоритмов. Все они представлены с помощью псевдокода, описанного в приложении С. Выбрать язык для представления алгоритмов было непросто. Перспективными кандидатами выглядели Паскаль, Фортран, Си и ANSI Бейсик. С точки зрения машинной графики каждый из этих языков имеет сильные и слабые стороны. Окончательное решение выбрано на основе педагогических соображений. Представление с помощью псевдокода требует, чтобы алгоритмы были переведены на «реальный» язык программирования. Есть надежда, что этот процесс улучшит понимание алгоритмов, а следовательно, и лежащей в их основе математики.

Версии алгоритмов на псевдокоде алгоритмически построены на основе работающих программ. Мы надеемся, что такой способ привел к получению более точных алгоритмов. Однако в процессе печати могли быть внесены отдельные ошибки. Поэтому при реализации следует еще раз внимательно сверить алгоритмы с математическим представлением в тексте.

Алгоритмы разрабатывались для обучения, т. е. они задуманы в качестве иллюстраций математических методов, обсуждавшихся в книге, но не как эффективно работающие программы для промышленной эксплуатации. Во многих случаях признавалась приемлемой вычислительная неэффективность, если она окупается тем, что алгоритм более полно соответствует основному тексту книги. Вычислительная неэффективность допускалась и для сокращения требуемой памяти. Таким образом, программы, разработанные в соответствии с предложенными алгоритмами, применимы для микрокомпьютерных операционных систем в условиях ограниченной доступной оперативной памяти. Примером служит алгоритм **bsplsurf** построения поверхности В-сплайна. В предложенном ниже алгоритме на псевдокоде все базовые функции для каждого значения параметра вычисляются вне главного цикла. Это сокращает требования к размерам памяти. Однако с точки зрения минимизации времени выполнения было бы более эффективно вычислять все базовые функции в главном цикле, получать результаты и запоминать их в большом массиве. Элементы этого массива используются затем в основном цикле для вычисления координатных векторов на поверхности. Через несколько минут работы становится ясно, что для достаточно сложных поверхностей размеры массива получаются довольно большими. Однако алгоритм выполняется в несколько раз быстрее, чем представленный алгоритм на псевдокоде. При реализации алгоритмов предполагается, что первоначально они будут реализованы

в том виде, в котором они даны в книге. Сначала надо удостовериться в правильности программы, а уже затем следует заниматься исследованием алгоритма и его математического обоснования с целью обнаружения методов повышения эффективности.

Алгоритмы сгруппированы по главам книги и расположены в алфавитном порядке для каждой главы.

ГЛАВА 2

Здесь приведены алгоритмы, выполняющие базовые двумерные манипуляции. Вот они в алфавитном порядке: **reflt2d**, **rot2d**, **scale2d**, **tran2d**.

reflt2d

Подпрограмма формирования и конкатенации матрицы отражения в плоскости к текущей преобразующей матрице (уравнения 2-34–2-36).

```
reflt(.) = матрица отражения
refltcode = код отражения
    1 = относительно оси x
    2 = относительно оси y
t(.) = текущая преобразующая матрица, с которой будет
      конкатенирована матрица отражения
```

subroutine reflt2d (rficode; t(.))

```
dimension reflt(3,3)
Mat reflt = Idn

if rficode = 1 then reflt(2,2) = -1 установить соответствующий элемент равным -1
if rficode = 2 then reflt(1,1) = -1
Mat t = t*reflt                                         конкатенировать матрицы
return
```

rot2d

Подпрограмма формирует матрицу поворота в плоскости и конкатенирует ее с текущей матрицей преобразования (уравнение (2-29)).

```
rot(.) = матрица поворота
theta = угол поворота в градусах
t(.) = матрица, с которой будет конкатенирована матрица поворота
```

subroutine rot2d (theta; t(.))

```
option angle degrees
dimension rot(3,3)
Mat rot = Idn
rot(1,1) = Cos(theta)                                    установка значений матрицы поворота
rot(1,2) = Sin(theta)
rot(2,2) = rot(1,1)
rot(2,1) = -rot(1,2)
```

```
Mat t = t*rot
return
```

конкатенация матриц

scale2d

Подпрограмма формирует матрицу масштабирования в плоскости и конкатенирует ее с текущей матрицей преобразования (уравнения (2-37)–(2-58)).

<i>a, d</i>	= масштабирующие множители в направлениях <i>x</i> и <i>y</i> ;
	<i>a, d > 1</i> – растяжение; <i>a, d < 1</i> – сжатие
<i>s</i>	= общий масштабирующий множитель; <i>s > 1</i> – увеличение;
	<i>s < 1</i> – сокращение
<i>Scale(.,.)</i>	= матрица масштабирования
<i>t(.,)</i>	= текущая преобразующая матрица, с которой будет конкатенирована матрица масштабирования

subroutine scale2d (a, d, s; t(.))

```
dimension scale(3,3)
Mat scale = Idn
scale(1,1) = a                                         установка значений матрицы масштабирования
scale(2,2) = d
scale(3,3) = 1/s
Mat t = t*scale                                         конкатенация матриц
return
```

tran2d

Подпрограмма формирует матрицу параллельного переноса в плоскости и конкатенирует ее с текущей матрицей преобразования (уравнение (2-50)).

<i>m, n</i>	= множитель переноса в направлениях <i>x</i> и <i>y</i> , соответственно
<i>t(.,)</i>	= матрица, с которой будет конкатенирована матрица переноса
<i>tran(.,)</i>	= матрица переноса

subroutine tran2d (m, n; t(.))

```
dimension tran(3,3)
Mat tran = Idn
tran(3,1) = m                                         установка значений матрицы переноса
tran(3,2) = n
Mat t = t*tran                                         конкатенация матриц
return
```

ГЛАВА 3

Здесь приведены алгоритмы, выполняющие базовые трехмерные манипуляции. Также предлагается ряд алгоритмов для проецирования на плоскость. В алфавитном порядке это: **cabinet**, **cavalier**, **dimetric**, **genrot**, **isometric**, **oblique**, **perspective**, **project**, **reflt3d**, **scale3d**, **tran3d**, **xrot**, **yrot**, **zrot**.

cabinet

Подпрограмма формирует матрицу проекции кабине и конкатенирует ее с текущей матрицей преобразования (с. 187 и пример 3-16).

*alpha = угол между горизонтальной и осью проекции z
t(.) = текущая преобразующая матрица, с которой будет конкатенирована матрица проекции кабине*

subroutine cabinet (alpha; t(.))

option angle degrees
выполнить проекцию кабине, вызвав oblique при $f = 0.5$
call oblique(alpha,0.5; t(.))
return

cavalier

Подпрограмма формирует матрицу проекции кавалье и конкатенирует ее с текущей матрицей преобразования (с. 187 и пример 3-16).

*alpha = угол между горизонтальной и осью проекции z
t(.) = текущая преобразующая матрица, с которой будет конкатенирована матрица проекции кавалье*

subroutine cavalier (alpha; t(.))

option angle degrees
выполнить проекцию кавалье, вызвав oblique при $f = 1$
call oblique(alpha,1; t(.))
return

dimetric

Подпрограмма формирует матрицу диметрической проекции и конкатенирует ее с текущей матрицей преобразования (уравнения (3-32), (3-39), (3-40) и пример 3-14).

*f = коэффициент искажения в направлении z
t(.) = текущая преобразующая матрица, с которой будет конкатенирована матрица диметрической проекции*

*xcode = код x поворота
0 = отрицательный угол поворота
1 = положительный угол поворота*

*ycode = код y поворота
0 = отрицательный угол поворота
1 = положительный угол поворота*

subroutine dimetric (ycode,xcode,f; t(.))

option angle degrees

найти углы поворота

*phi = Arcsin(f/(Sqrt(2 - f*f)))*

theta = Arcsin(f/Sqrt(2))

установить значения направлений поворота

*if ycode = 0 then phi = -phi
if xcode = 0 then theta = -theta
выполнить диметрические преобразования, вызвав yrot, xrot и project
call yrot (phi; t)
call xrot (theta; t)
call project (3; t)
return*

genrot

Подпрограмма формирует матрицу трехмерного поворота относительно произвольной оси в пространстве и конкатенирует ее с текущей матрицей преобразования методом совмещения оси поворота с осью z (уравнения (3-21) и (3-23)–(3-26)).

*a(.) = массив, содержащий две точки оси поворота
delta = угол поворота относительно произвольной оси в градусах
t(.) = текущая преобразующая матрица, с которой будет конкатенирована матрица поворота*

subroutine genrot (a(.),delta; t(.))

option angle degrees
dimension rx(4,4),ry(4,4),rdelta(4,4),rxinv(4,4),ryinv(4,4)
dimension tran(4,4),traninv(4,4)
dimension m(4,4),minv(4,4)
dimension templ(4,4),temp2(4,4)

Mat rx = Idn

Mat ry = Idn

Mat rdelta = Idn

Mat tran = Idn

Mat rxinv = Idn

Mat ryinv = Idn

Mat traninv = Idn

вычислить направляющие косинусы

cx = a(2,1) – a(1,1)

cy = a(2,2) – a(1,2)

cz = a(2,3) – a(1,3)

*nfactor = Sqrt(cx*cx + cy*cy + cz*cz)*

cx = cx/nfactor

cy = cy/nfactor

cz = cz/nfactor

*d = Sqrt(cy*cy + cz*cz)*

установить значения переменных матриц

заметим, что, так как gx и gy – абсолютные повороты, обращение является транспонированием

tran(4,1) = -a(1,1)

tran(4,2) = -a(1,2)

tran(4,3) = -a(1,3)

traninv(4,1) = a(1,1)

traninv(4,2) = a(1,2)

traninv(4,3) = a(1,3)

```

rx(2,2) = cz/d
rx(2,3) = cy/d
rx(3,2) = -rx(2,3)
rx(3,3) = rx(2,2)
Mat rxinv = Trn(rx)
ry(1,1)=d
ry(1,3)=cx
ry(3,1)=-cx
ry(3,3)=d
Mat ryinv = Trn(ry)
rdelta(1,1)=Cos(delta)
rdelta(2,2)=rdelta(1,1)
rdelta(1,2)=Sin(delta)
rdelta(2,1)=-rdelta(1,2)
конкатенировать переменные матрицы
Mat templ = rx*ry
Mat m = tran*templ
Mat temp2 = rxinv*traninv
Mat minv = ryinv*temp2
Mat templ = rdelta*minv
Mat temp2 = m*templ
Mat t = t*temp2
return

```

imsometric

Подпрограмма формирует матрицу изометрической проекции и конкатенирует ее с текущей матрицей преобразования (уравнения (3-32), (3-41) и пример 3-15).

```

t(.) = текущая преобразующая матрица, с которой будет
      конкатенирована матрица изометрической проекции
xcode = код x поворота
    0 = отрицательный угол поворота
    1 = положительный угол поворота
ycode = код y поворота
    0 = отрицательный угол поворота
    1 = положительный угол поворота

```

subroutine isometric (ycode,xcode; t(.))

option angle degrees

установить значения направлений поворота

```

if ycode = 0 then phi = -45
if ycode = 1 then phi = 45
if xcode = 0 then pheta = -35.26439
if xcode = 1 then pheta = 35.26439

```

выполнить изометрические преобразования, вызвав yrot, xrot и project

```

call yrot(phi; t)
call xrot(pheta; t)
call project(3; t)
return

```

oblique

Подпрограмма формирует матрицу косоугольной проекции и конкатенирует ее с текущей матрицей преобразования (уравнение (3-44)).

```

alpha = угол между горизонталью и осью проекции z
f = коэффициент искажения в направлении z
t(.) = текущая преобразующая матрица, с которой будет
      конкатенирована матрица косоугольной проекции

```

subroutine oblique (alpha,f; t(.))

option angle degrees

dimension oblique(4,4)

Mat oblique = Idn

```

oblique(3,3) = 0
oblique(3,1) = -f*Cos(alpha)
oblique(3,2) = -f*Sin(alpha)

```

установить значения преобразующей матрицы

Mat t = t*oblique

конкатенировать матрицы

return

perspective

Подпрограмма формирует матрицу центральной перспективной проекции на плоскость $z = 0$ и конкатенирует ее с текущей матрицей преобразования (уравнение (3-47)).

```

cp = центр проекции на оси z
t(.) = текущая преобразующая матрица, с которой будет
      конкатенирована матрица перспективной проекции

```

subroutine perspective (cp; t(.))

option angle degrees

dimension perspective(4,4)

Mat perspective = Idn

```

perspective(3,3) = 0

```

установить значения матрицы центральной перспективной проекции

```

perspective(3,4) = -1/cp

```

Mat t = t*perspective

конкатенировать матрицы

return

project

Подпрограмма формирует матрицу простой ортографической проекции на одну из координатных плоскостей и конкатенирует ее с текущей матрицей преобразования (уравнения (3-27)–(3-29)).

proj(.) = матрица проекции

projcode = код проекции

1 = на плоскость $x = 0$

2 = на плоскость $y = 0$

3 = на плоскость $z = 0$

```

t(.) = текущая преобразующая матрица, с которой будет
      конкатенирована матрица проекции

```

```

subroutine project (projcode; t())
dimension proj(4,4)
Mat proj = Idn
if projcode = 1 then proj(1,1) = 0           установить соответствующий элемент
                                                матрицы равным 0
if projcode = 2 then proj(2,2) = 0
if projcode = 3 then proj(3,3) = 0
Mat t = t*proj                                конкатенировать матрицы
return

```

reflt3d

Подпрограмма формирует матрицу зеркального отражения в пространстве и конкатенирует ее с текущей матрицей преобразования (уравнения (3-11)–(3-13)).

```

refltcode = код отражения
  1 = относительно плоскости x = 0
  2 = относительно плоскости y = 0
  3 = относительно плоскости z = 0
reflt(.) = матрица отражения
t(.)     = текущая преобразующая матрица, с которой будет
            конкатенирована матрица отражения

```

```

subroutine reflt3d (rltcode; t())
dimension reflt(4,4)
Mat reflt = Idn
if rltcode = 1 then reflt(1,1) = -1          установить соответствующий терм
                                                в матрице отражения равным -1
if rltcode = 2 then reflt(2,2) = -1
if rltcode = 3 then reflt(3,3) = -1
Mat t = t*reflt                                конкатенировать матрицы
return

```

scale3d

Подпрограмма формирует матрицу масштабирования в пространстве и конкатенирует ее с текущей матрицей преобразования (уравнения (3-3) и (3-4)).

```

a = масштабирующий множитель в направлении x a > 1 –
      растяжение; a < 1 – сжатие
e = масштабирующий множитель в направлении y e > 1 –
      растяжение; e < 1 – сжатие
j = масштабирующий множитель в направлении z j > 1 –
      растяжение; j < 1 – сжатие
s = общий масштабирующий множитель s > 1 – увеличение;
      s < 1 – сокращение
scale = масштабирующая матрица
t(.) = текущая преобразующая матрица, с которой будет
            конкатенирована масштабирующая матрица

```

```
subroutine scale3d (a,e,j,s; t())
```

```

option angle degrees
dimension scale(4,4)
Mat scale = Zer
scale(1,1) = a
scale(2,2) = e
scale(3,3) = j
scale(4,4) = 1/s
Mat t = t*scale
return

```

tran3d

Подпрограмма формирует матрицу параллельного переноса в пространстве и конкатенирует ее с текущей матрицей преобразования (уравнение (3-14)).

```

l = множитель переноса в направлении x
m = множитель переноса в направлении y
n = множитель переноса в направлении z
tran(.) = матрица переноса
t(.) = текущая преобразующая матрица, с которой будет
            конкатенирована матрица переноса

```

```

subroutine tran3d(l,m,n; t())
dimension tran(4,4)
Mat tran = Idn
tran(4,1) = l
tran(4,2) = m
tran(4,3) = n
Mat t = t*tran
return

```

xrot

Подпрограмма формирует матрицу поворота в пространстве относительно оси *x* и конкатенирует ее с текущей матрицей преобразования (уравнение (3-6)).

```

rot(.) = матрица поворота
theta = угол поворота в градусах
t(.) = текущая преобразующая матрица, с которой будет
            конкатенирована матрица поворота

```

```

subroutine xrot (theta; t())
option angle degrees
dimension rot(4,4)
Mat rot = Idn
rot(2,2) = Cos(theta)
rot(2,3) = Sin(theta)
rot(3,2) = -rot(2,3)
Mat t = t*rot
return

```

установка матрицы поворота
конкатенировать матрицы

yrot

Подпрограмма формирует матрицу поворота в пространстве относительно оси y и конкатенирует ее с прикладной матрицей преобразования (уравнение (3-8)).

phi = угол поворота в градусах
rot(,) = матрица поворота
t(,) = текущая преобразующая матрица, с которой будет конкатенирована матрица поворота

```
subroutine yrot (phi; t(,))
option angle degrees
dimension rot(4,4)
Mat rot = Idn
rot(1,1) = Cos(phi)
rot(3,1) = Sin(phi)
rot(3,3) = rot(1,1)
rot(1,3) = -rot(3,1)
Mat t = t*rot
return
```

установка матрицы поворота

конкатенировать матрицы

zrot

Подпрограмма формирует матрицу поворота в пространстве относительно оси z и конкатенирует ее с текущей матрицей преобразования (уравнение (3-7)).

rot(,) = матрица поворота
theta = угол поворота в градусах
t(,) = текущая преобразующая матрица, с которой будет конкатенирована матрица поворота

```
subroutine zrot (theta; t(,))
option angle degrees
dimension rot(4,4)
Mat rot = Idn
rot(1,1) = Cos(theta)
rot(1,2) = Sin(theta)
rot(2,2) = rot(1,1)
rot(2,1) = -rot(1,2)
Mat t = t*rot
return
```

установка матрицы поворота

конкатенировать матрицы

ГЛАВА 4

Здесь даны алгоритмы, которые генерируют точки на периметре конического сектора. В алфавитном порядке это **circle**, **ellipse**, **hyper 1**, **hyper 2**, **parabola**. Каждый из этих алгоритмов легко модифицируется для генерации точек на отдельной части конического сектора.

circle

Подпрограмма генерирует точки на внешней окружности круга (уравнение (4-5)).

<i>dtheta</i>	= $\cos(d\theta)$
<i>circ(,)</i>	= массив, содержащий точки плоскости
<i>circ(1,)</i>	= координата x
<i>circ(2,)</i>	= координата y
<i>dtheta</i>	= приращение θ
<i>h</i>	= координата x центра окружности
<i>k</i>	= координата y центра окружности
<i>n</i>	= число точек на окружности
<i>r</i>	= радиус
<i>sdtheta</i>	= $\sin(d\theta)$
<i>tcirc(,)</i>	= массив, содержащий точки плоскости
	<i>tcirc(1,)</i> = координата x
	<i>tcirc(2,)</i> = координата y
	<i>tcirc(3,)</i> = 1, однородная координата
<i>t(,)</i>	= матрица переноса в плоскости 3×3

subroutine circle (h,k,r,n; circ(,))

```
option angle degrees
dimension t(3,3),tcirc(256,3)
изменение размера, заполнение преобразования и массива tcirc
Mat rot = Idn
Mat tcirc = Zer(n,3)
Mat tcirc = Con(n,3)
определение констант
dtheta = 360/(n - 1)
cdtheta = Cos(dtheta)
sdtheta = Sin(dtheta)
определение начальной точки
tcirc(1,1) = r
tcirc(1,2) = 0
вычисление точек на исходной центрированной окружности
for i = 2 to n
    tcirc(i,1) = cdtheta*tcirc(i - 1,1) - sdtheta*tcirc(i - 1,2)
    tcirc(i,2) = sdtheta*tcirc(i - 1,1) + cdtheta*tcirc(i - 1,2)
next i
перенос результатов в h, k
call tran2d(h,k; t)
Mat tcirc = tcirc*t
приписывание перенесенных значений в массив 2D
for i = 1 to n
    circ(i,1) = tcirc(i,1)
    circ(i,2) = tcirc(i,2)
next i
return
```

ellipse

Подпрограмма генерирует точки на периметре эллипса (уравнение (4-8)).

```

a      = длина большой полуоси
b      = длина малой полуоси
c1     = константа —  $(a/b)*cdtheta$ 
c2     = константа —  $(b/a)*sdtheta$ 
cdtheta = константа —  $\cos(dtheta)$ 
dtheta  = приращение параметра
ellip(,) = массив, содержащий точки эллипса
          ellip(,1) = компонента x
          ellip(,2) = компонента y
h      = x координата центра эллипса
inclin = угол наклона большой оси в градусах
k      = y координата центра эллипса
n      = число точек на эллипсе
sdtheta = константа —  $\sin(dtheta)$ 
tellip(,) = 1, массив, содержащий однородные точки пространства
            tellip(,1) = x координата
            tellip(,2) = y координата
            tellip(,3) = 1, однородная координата
t(,)   = матрица преобразования в плоскости  $3 \times 3$ 

```

subroutine ellipse (h,k,a,b,inclin,n; ellip(,))

option angle degrees

dimension t(3,3),tellip(256,3)

изменение размера, заполнение преобразования и массива tellip

Mat t = Idn

Mat tellip = Zer(n,3)

Mat tellip = Con(n,3)

определение констант

dtheta = $360/(n - 1)$

cdtheta = Cos(dtheta)

sdtheta = Sin(dtheta)

c1 = $(a/b)*sdtheta$

c2 = $(b/a)*sdtheta$

определение начальной точки

tellip(1,1) = a

tellip(1,2) = 0

вычисление точек на исходном центрированном эллипсе

for i = 2 to n

 tellip(i,1) = tellip(i - 1,1)*cdtheta - c1*tellip(i - 1,2)

 tellip(i,2) = c2*tellip(i - 1,1) + tellip(i - 1,2)*cdtheta

next i

поворот на i и перенос результатов в h, k

call rot2d(inclin; t)

call tran2d(h,k; t)

Mat tellip = tellip*t

приписывание перенесенных значений в массив 2D

for i = 1 to n

 ellip(i,1) = tellip(i,1)

```

ellip(i,2) = tellip(i,2)
next i
return

```

hyper 1

Подпрограмма генерирует точки на гиперболе (уравнение (4-15)). Генерирует только гипербола в первом квадранте, открытая вправо, с осью x в качестве оси симметрии. Полная гипербола может быть получена зеркальным отражением сначала относительно оси x, затем относительно оси y. Произвольные ориентации и центральные позиции могут быть получены соответствующими поворотами и переносами.

```

a      = расстояние от центра гиперболы до вертикали
b      = определяет наклон асимптот =  $\pm b/a$ 
c1     = константа —  $(b)*tdtheta$ 
c2     = константа —  $(a)*cdtheta$ 
cdtheta = константа —  $\cos(dtheta)$ 
dtheta  = приращение параметра
hyper(,) = массив, содержащий точки параболы
          hyper(,1) = x компонента
          hyper(,2) = y компонента
n      = число точек на гиперболе
tdtheta = константа —  $\tan(dtheta)$ 
sdtheta(,) = 1, константа —  $\sin(dtheta)$ 

```

subroutine hyper1 (a,b,n; hyper(,))

изменение размера и заполнение массива hyper нулями

Mat hyper = Zer(n,2)

определение констант

dtheta = $(Pi/2)(n - 1)$

cdtheta = Cos(dtheta)

sdtheta = Sin(dtheta)

tdtheta = Tan(dtheta)

c1 = $(b)*tdtheta$

c2 = $(b)*cdtheta$

определение начальной точки

hyper(1,1) = a

hyper(1,2) = 0

вычисление точек на исходной центрированной гиперболе

for i = 2 to n

 hyper(i,1) = b*hyper(i - 1,1)/(c2 - sdtheta*hyper(i - 1,2))

 hyper(i,2) = b*(hyper(i - 1,2) + c1)/(b - hyper(i - 1,2)*tdtheta)

next i

return

hyper 2

Подпрограмма генерирует точки на гиперболе с использованием гиперболических функций (уравнение (4-17)). Генерируется только гипербола в первом квадранте, открытая вправо, с осью x в качестве оси симметрии. Полная гипербола

может быть получена зеркальным отражением сначала относительно оси x , потом относительно оси y . Произвольные ориентации и центральные позиции могут быть получены соответствующими поворотами и переносами.

```

a      = расстояние от центра гиперболы до вертикали
b      = определяет наклон асимптот =  $\pm b/a$ 
c1     = константа  $-(a/b)*sdtheta$ 
c2     = константа  $-(b/a)*sdtheta$ 
cdtheta = константа  $cosh(dtheta)$ 
dtheta = приращение параметра
hyper(.) = массив, содержащий точки параболы
          hyper(.1) = x компонента
          hyper(.2) = y компонента
n      = число точек на гиперболе
sdtheta = константа  $-sinh(dtheta)$ 
thetamax(,) = максимальное значение параметра

```

subroutine hyper2 (a,b,thetamax,n; hyper,())
изменение размера и заполнение массива *hyper* нулями

Mat hyper = Zer(n,2)

определение констант

dtheta = thetamax(n - 1)

cdtheta = Cosh(dtheta)

sdtheta = Sinh(dtheta)

c1 = (a/b)*sdtheta

c2 = (b/a)*sdtheta

определение начальной точки

hyper(1,1) = a

hyper(1,2) = 0

вычисление точек на исходной центрированной гиперболе

for i = 2 to n

hyper(i,1) = hyper(i - 1,1)*cdtheta + c1*hyper(i - 1,2)
 hyper(i,2) = c2*hyper(i - 1,1) + hyper(i - 1,2)*cdtheta

next i

return

parabola

Подпрограмма генерирует в первом квадранте точки на параболе, симметричной относительной оси x , открытой вправо (уравнение (4-12)). Для генерации полной параболы в пространстве, расположенной в произвольной позиции и произвольной ориентации, применяют необходимые преобразования зеркального отражения, поворота и параллельного переноса.

```

a      = расстояние от фокуса до вертикали параболы
a1     = константа  $-a*dtheta*dtheta$ 
b1     = константа  $2 * a * dtheta$ 
dtheta = приращение параметра
n      = число точек на параболе
para(.) = массив, содержащий точки параболы
          para(.1) = координата x
          para(.2) = координата y
thetamax = максимальное значение параметра

```

```

subroutine parabola (a,thetamax,n; para,())  
изменение размера и заполнение массива para нулями
Mat para = Zer(n,2)
определение констант
dtheta = thetamax/(n - 1)
a1 = a*dtheta*dtheta
b1 = 2*a*dtheta
определение начальной точки
para(1,1) = 0
para(1,2) = 0
вычисление точек на исходной параболе
for i = 2 to n
    para(i,1) = para(i - 1,1) + dtheta*para(i - 1,2) + a1
    para(i,2) = para(i - 1,2) + b1
next i
return

```

ГЛАВА 5

Здесь приведены алгоритмы реализации методов построения оболочек. Алгоритмы даны для кубического сплайна, параболически смешанных кривых и кривых В-сплайна. Алгоритмы каркасного представления кривых даны для кривых Безье, для рациональных и иррациональных кривых В-сплайна, однородных и неоднородных узловых векторов и для базисных функций В-сплайна: **basis**, **bezier**, **bsplfit**, **bspline**, **bsplineu**, **cblend**, **cspline**, **dbasis**, **dbasisu**, **dbezier**, **dbspline**, **dbsplineu**, **knot**, **knotc**, **knotu**, **matpbspl**, **nmatrix**, **parabld**, **param**, **rbasis**, **rbspline**, **rbsplinu**.

basis

Подпрограмма генерирует базисные функции В-сплайна для открытых неоднородных узловых векторов (уравнение (5-84)).

```

c      = порядок базисной функции В-сплайна
d      = первый терм рекурсивного отношения базисной функции
e      = второй терм рекурсивного отношения базисной функции
npts   = число вершин определяющего многоугольника
n(.)  = массив, содержащий базисные функции
        n(1,1) содержит базисную функцию, соответствующую  $B_1$  и т. д.
nplusc = константа  $-npts + c$  — максимальное
          значение номера узла
t      = значение параметра
temp() = временный массив
x()   = узловой вектор

```

subroutine basis (c,t,npts,x(); n,())

dimension temp(20) допускает 20 вершин многоугольника
nplusc = npts + c
вычисление основных функций $N_{i,1}$ (см. уравнения (5-84a))
for i = 1 to nplusc - 1

```

if t >= x(i) and t < x(i + 1) then
    temp(i) = 1
else
    temp(i) = 0
end if
next i
вычисление основных функций высшего порядка (см. уравнения (5-84б))
for k = 2 to c
    for i = 1 to nplusc - k
        if temp(i) ≠ 0 then  если основная функция равна нулю, пропустите вычисление
            d=((t-x(i))*temp(i))/(x(i+k-1)-x(i))
        else
            d=0
        end if
        if temp(i+1)≠0 then  если основная функция равна нулю, пропустите вычисление
            e = ((x(i + k) - t)*temp(i + 1))/(x(i + k) - x(i + 1))
        else
            e = 0
        end if
        temp(i)=d+e
    next i
next k
if t = x(nplusc) then temp(npts) = 1
ставить в массив
for i = 1 to npts
    n(1,i) = temp(i)
next i
if t = x(nplusc) then n(1,npts) = 1
return

```

bezier

Подпрограмма вычисляет кривую Безье (уравнение (5-62)).

b(,) = массив, содержащий вершины определяющего многоугольника
b(,1) содержит компоненту *x* вершины
b(,2) содержит компоненту *y* вершины
b(,3) содержит компоненту *z* вершины

Basis = функция для вычисления базисного значения Бернштейна (уравнение (5-63))

cpts = число точек, которые будут вычислены на кривой

Factrl = функции для вычисления факториала числа

j(,) = базисная функция Бернштейна

Ni = функции факториала для базисной функции Бернштейна

npts = число вершин определяющего многоугольника

p(,) = массив, содержащий точки кривой

p(,1) содержит компоненту *x* точки

p(,2) содержит компоненту *y* точки

p(,3) содержит компоненту *z* точки

t = значение параметра $0 \leq t \leq 1$

subroutine bezier (npts,b(,),cpts; p(,))

defNi(n,i)=Factrl(n)/Factrl(i)*Factrl(n-i)

начните с последней точки

начните с последней точки

```

def Basis(n,i,t) = Ni(n,i)*(t^i)*((1 - t)^(n - i))      базисная функция Бернштейна
dimension j(1,20)                                         допускает 20 вершин многоугольника
dimension temp(1,3)
icount = 0
Mat j = Zer(1,npts)
for t = 0 to 1 step 1/(cpts - 1)
    icount = icount + 1
    определение основной функции Бернштейна (см. уравнение (5-63))
    for i = 1 to npts
        j(1,i) = Basis(npts - 1,i - 1,t)
    next i
    определение точки на кривой
    Mat temp = j*b
    вставить в массив
    for i = 1 to 3
        p(icount,i) = temp(1,i)
    next i
mext t
return

```

bsplfit

Подпрограмма выравнивает кривую В-сплайна, используя открытый однородный узловый вектор (уравнение (5-117)).

b(,) = массив, содержащий вершины определяющего многоугольника
b(,1) содержит компоненту *x* вершины
b(,2) содержит компоненту *y* вершины
b(,3) содержит компоненту *z* вершины

dpts = число точек данных

d(,) = массив, содержащий точки данных
d(,1) содержит *x* компоненту точки данных
d(,2) содержит *y* компоненту точки данных
d(,3) содержит *z* компоненту точки данных

k = порядок базисной функции В-сплайна

n = матрица базисной функции

nbasis = массив, содержащий базисные функции для единственного значения *t*

ninv = обращение *trn(n)×n*

nplusc = число узловых значений

npts = число вершин определяющего многоугольника

ntemp = временная матрица для хранения *trn(n)×n*

ntmp = временная матрица для хранения обращения *trn(n)×n×d*

ntrn = транспонирование матрицы *n*

t = параметрическое значение $0 \leq t \leq 1$

tpar() = массив, содержащий аппроксимацию гордами для параметрических значений

x() = массив, содержащий узловой вектор

subroutine bsplfit (dpts,d(,),npts,k; b(,))

dimension nbasis(1,20),x(30),n(20,20),temp(1,3) допускает данные о 20 точках
dimension tpar(20),ninv(20,20),ntrn(20,20),ntemp(20,20),ntmp(20,3)

изменение размера и заполнение матриц нулями

функция факториал

```

Mat nbasis = Zer(1,npts)
Mat x = Zer(npts + k)
Mat n = Zer(dpts,npts)
Mat ntrn = Zer(npts,dpts)
Mat ntemp = Zer(npts,npts)
Mat ntmp = Zer(npts,2)
Mat ninv = Zer(npts,npts)
Mat tpar = Zer(dpts)
call knot(npts,k; x())
call param(dpts,d; tpar),
nplusc = npts + k
генерация матрицы основных функций
for i = 1 to dpts
    t = tpar(i)*x(nplusc)           вычисление значения параметра для одной строки
    call basis(k,t,npts,x; nbasis)   вычисление основной функции для одной строки
    for j = 1 to npts
        n(i,j) = nbasis(1,j)         построение матрицы строка за строкой
    next j
next i
генерация вершин определяющего многоугольника методом наименьших квадратов
Mat ntrn = trn(n)               нахождение транспонированной матрицы основной функции
Mat ntemp = ntrn*n              trn(n)*n
Mat ninv = Inv(ntemp)           обращение матрицы trn(n)*n
Mat ntmp = ntrn*d               обращение матрицы trn(n)*n x d
Mat b = ninv*ntmp               вычисление вершин определяющего многоугольника
return

```

bspline

Подпрограмма генерирует кривую В-сплайна, используя открытый однородный узловой вектор (уравнение (5-83)).

b(,) = массив, содержащий вершины определяющего многоугольника
b(,1) содержит компоненту *x* вершины
b(,2) содержит компоненту *y* вершины
b(,3) содержит компоненту *z* вершины
k = порядок базисной функции В-сплайна
nbasis = массив, содержащий базисные функции для единственного значения
nplusc = число узловых значений
npts = число вершин определяющего многоугольника
p(,) = массив, содержащий точки кривой
p(,1) содержит компоненту *x* точки
p(,2) содержит компоненту *y* точки
p(,3) содержит компоненту *z* точки
p1 = число вычисляемых точек кривой
t = параметрическое значение $0 \leq t \leq 1$
x() = массив, содержащий вектор

subroutine bspline (npts,k,p1,b(,); p(,))

dimension nbasis(1,20),x(30),temp(1,3) допускает 20 вершин многоугольников
 с базисной функцией порядка 5

nplusc = *npts* + *k*

обнулить и изменить размерность узлового вектора и базисного массива

```

Mat nbasis = Zer(1,npts)
Mat x = Zer(nplusc)
генерация однородного открытого узлового вектора
call knot(npts,k; x)
icount = 0
вычисление точек на кривой В-сплайна
for t = 0 to x(npts + k) step x(npts + k)/(p1 - 1)
    icount = icount + 1
    call basis(k,t,npts,x; nbasis)      генерация базисной функции для этого значения t
    Mat temp = nbasis*b                 генерация точки кривой
    p(icount,1) = temp(1,1)             приписывание текущего значения точки кривой
    p(icount,2) = temp(1,2)
    p(icount,3) = temp(1,3)
next t
return

```

bsplineu

Подпрограмма генерирует кривую В-сплайна, используя периодический однородный узловой вектор (уравнение (5-83)).

b(,) = массив, содержащий вершины определяющего многоугольника
b(,1) содержит компоненту *x* вершины
b(,2) содержит компоненту *y* вершины
b(,3) содержит компоненту *z* вершины
k = порядок базисной функции В-сплайна
nbasis = массив, содержащий базисные функции для единственного значения *t*
nplusc = число узловых значений
npts = число вершин определяющего многоугольника
p(,) = массив, содержащий точки кривой
p(,1) содержит компоненту *x* точки
p(,2) содержит компоненту *y* точки
p(,3) содержит компоненту *z* точки
p1 = число вычисляемых точек кривой
t = параметрическое значение $0 \leq t \leq 1$
x() = массив, содержащий узловой вектор

subroutine bspline (npts,k,p1,b(,); p(,))

dimension nbasis(1,20),x(30),temp(1,3) допускает 20 вершин многоугольника
 с базисной функцией порядка 5

обнулить и изменить размерность узлового вектора и базисного массива

```

Mat nbasis = Zer(1,npts)
Mat x = Zer(npts + k)
nplusc = npts + k
генерация однородного периодического узлового вектора
call knotu(npts,k; x)
icount = 0
вычисление точек на кривой В-сплайна
for t = (k - 1) to (npts - 1 + 1) step (npts - 1 + 1) - (k - 1)/(p1 - 1)

```