

Компьютерная графика

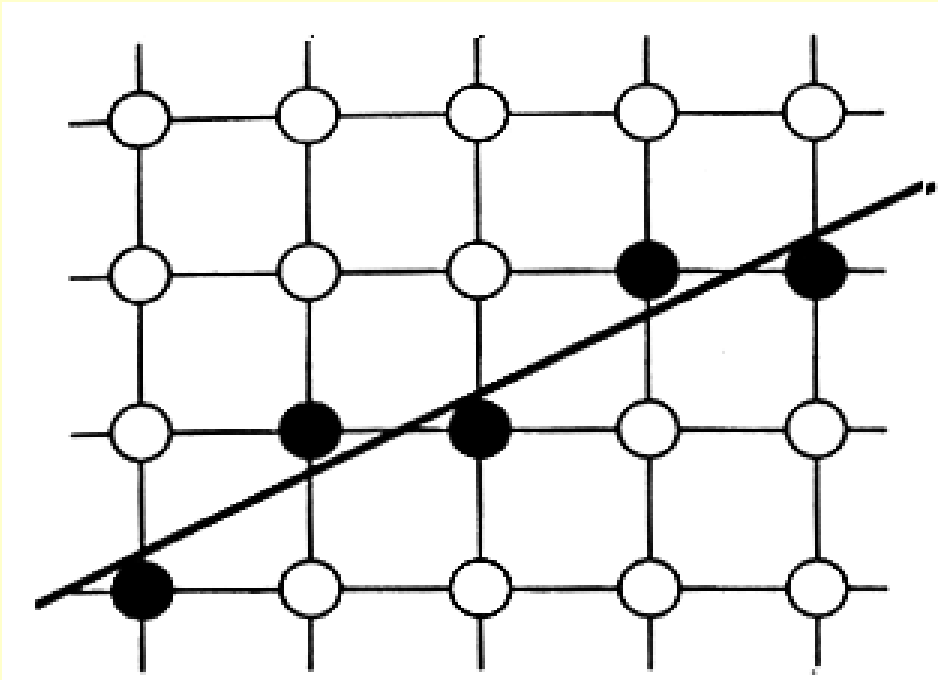
Лекция 12

Прямые Кривые Поверхности

28 апреля 2006 года

Алгоритм Брезенхема (1)

(прямая)



Отрезок, соединяющий
 $P(x_1, y_1)$ и $Q(x_2, y_2)$

$$0 \leq y_2 - y_1 \leq x_2 - x_1$$

$$y = y_1 + \frac{y_2 - y_1}{x_2 - x_1} (x - x_1), x \in [x_1, x_2]$$

Алгоритм Брезенхема (2)

(прямая)

$$F(x, y) = (x - x_1)dx - (y - y_1)dy$$

$$dx = x_2 - x_1$$

$$dy = y_2 - y_1$$

$F(x, y) = 0$ -- точка на отрезке

$F(x, y) < 0$ -- точка выше

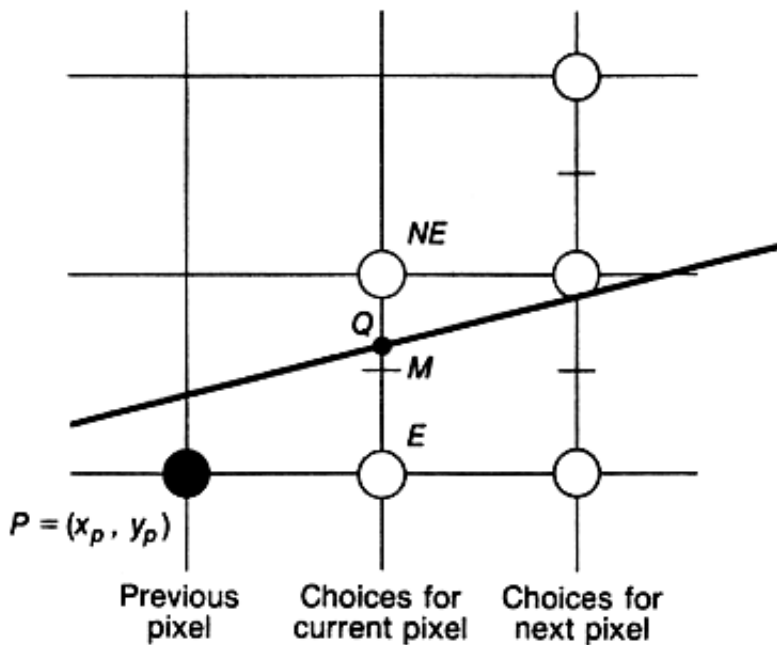
$F(x, y) > 0$ -- точка ниже

Точка Р определена, тогда координаты срединной точки

$$(x_p + 1, y_p + 1/2)$$

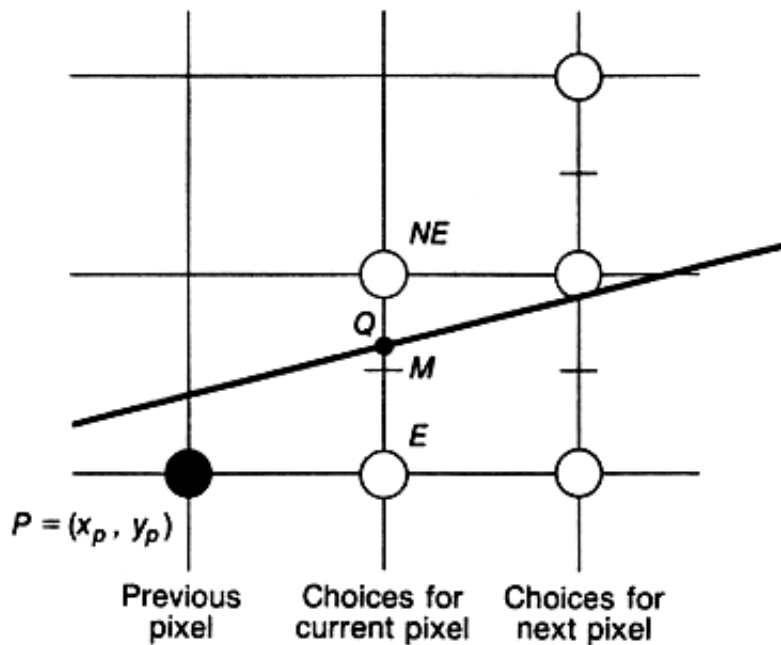
и значение функции в этой точке

$$d = F(x_p + 1, y_p + 1/2)$$



Алгоритм Брезенхема (3)

(прямая)



Если $d < 0$, то выбирается E и

$$d_{new} = F(x_p + 2, y_p + \frac{1}{2})$$

$$d_{new} - d_{old} = F(x_p + 2, y_p + \frac{1}{2}) - F(x_p + 1, y_p + \frac{1}{2})$$

$$\Delta_E = d_{new} - d_{old} = dy = y_2 - y_1$$

Если $d \geq 0$, то выбирается NE

$$d_{new} = F(x_p + 2, y_p + \frac{3}{2})$$

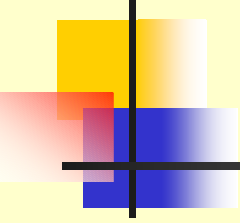
$$\Delta_{NE} = dy - dx = (y_2 - y_1) - (x_2 - x_1)$$

В начальной точке

$$d_{start} = F(x_1 + 1, y_1 + 1/2) =$$

$$(x_1 + 1 - x_1)dy - (y_1 + 1/2 - y_1)dx =$$

$$= dy - dx / 2$$



Алгоритм Брезенхема (4)

(прямая)

Одна неприятность -- деление на 2

Чтобы избежать вещественной арифметики,
сделаем преобразование

$$F'(x, y) = 2F(x, y)$$

$$d' = 2d$$

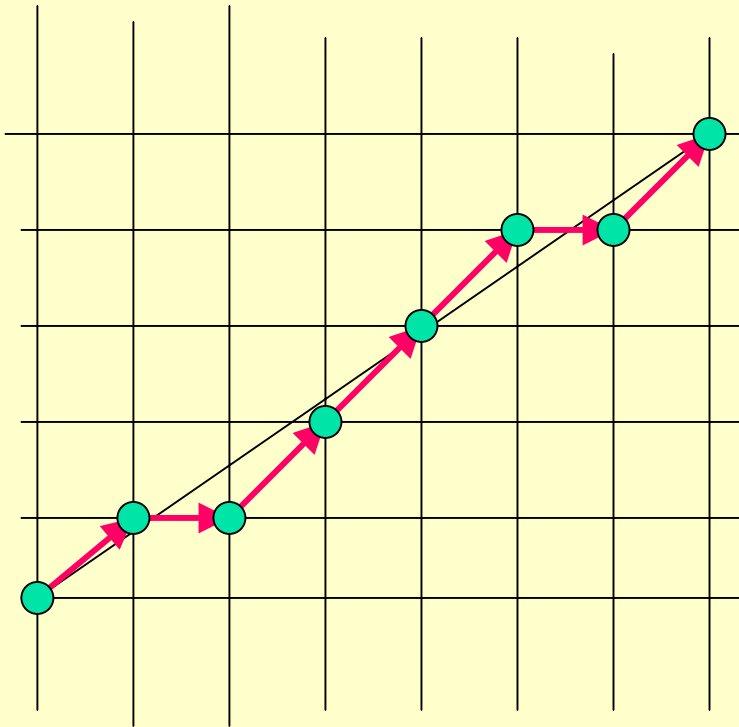
$$d_{start} = 2dy - dx$$

$$\Delta d' = 2\Delta d$$

Алгоритм Брезенхема (5)

(пример: $dx = 7, dy = 5$)

$$d_{start} = 2dy - dx = 3; d_{NE} = -4; d_E = 10$$



$$d_0 = 10 - 7 = 3 > 0 \quad (\text{NE})$$

$$d_1 = 3 - 4 = -1 < 0 \quad (\text{E})$$

$$d_2 = -1 + 10 = 9 \quad (\text{NE})$$

$$d_3 = 9 - 4 = 5 \quad (\text{NE})$$

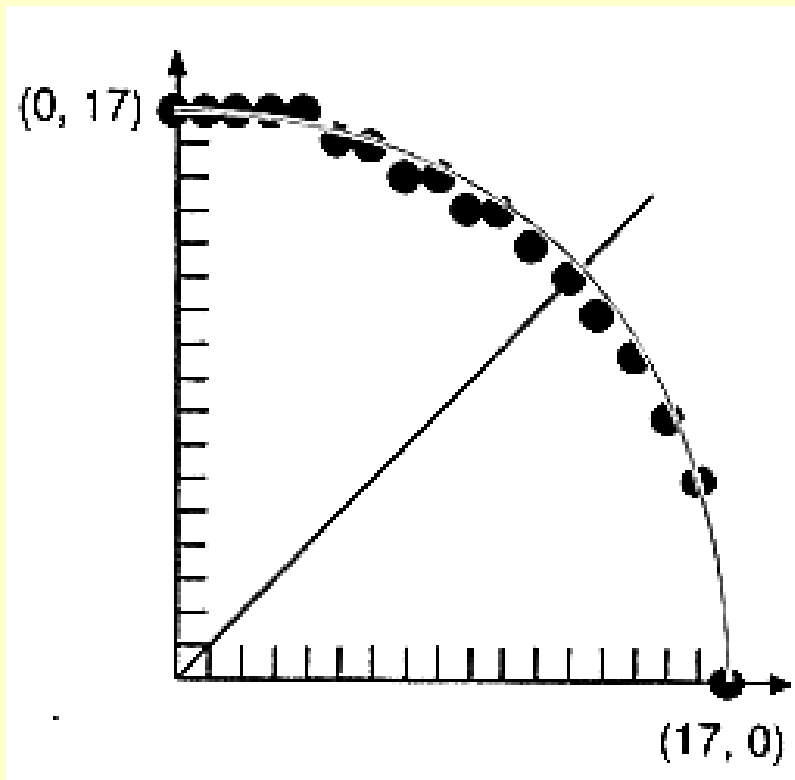
$$d_4 = 5 - 4 = 1 \quad (\text{NE})$$

$$d_5 = 1 - 4 = -3 \quad (\text{E})$$

$$d_6 = -3 + 10 = 7 \quad (\text{NE})$$

Алгоритм Брезенхема (6)

(окружность)



Неявное и явное представление

$$x^2 + y^2 = R^2$$

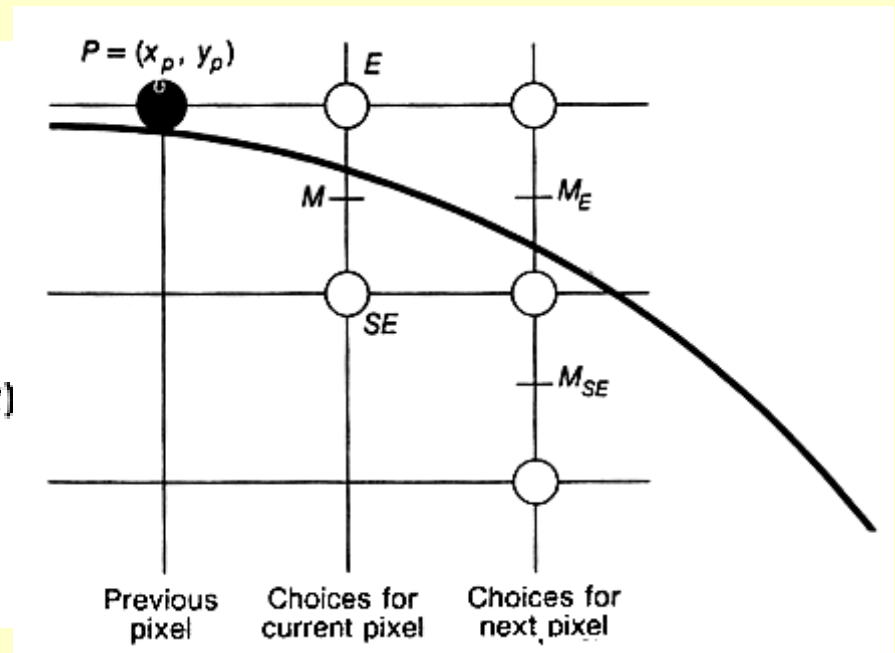
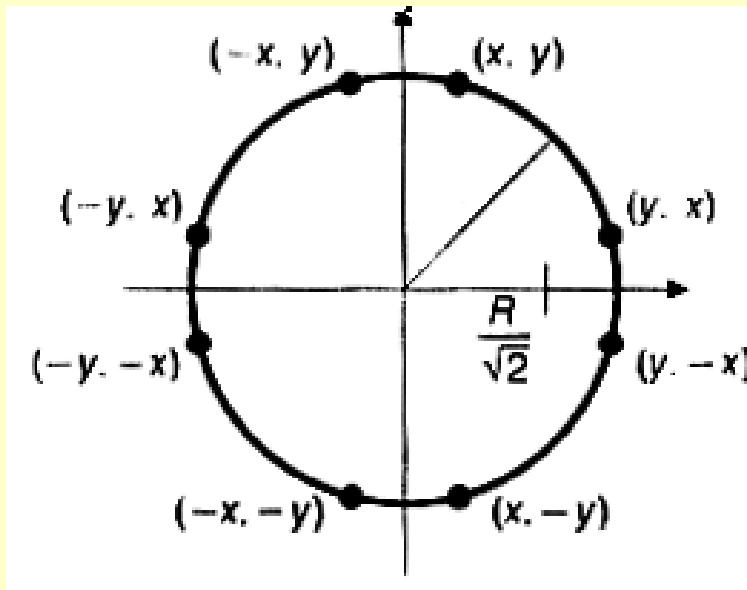
$$y = \pm \sqrt{R^2 - x^2}$$

Параметрическое представление

$$x = R \cos \theta$$

$$y = R \sin \theta$$

Алгоритм Брезенхема (7) (окружность)



Алгоритм Брезенхема (8) (окружность)

$$F(x, y) = x^2 + y^2 - R^2$$

Для точки P с коорд. (x_p, y_p)

$$d_{old} = F(x_p + 1, y_p - \frac{1}{2})$$

Для пиксела E:

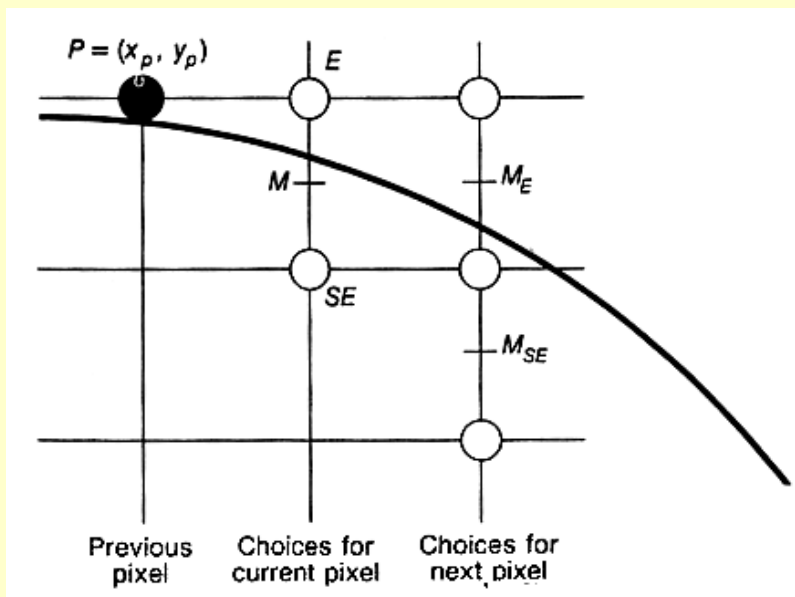
$$d_{new} = f(x_p + 2, y_p - \frac{1}{2}) = d_{old} + (2x_p + 3)$$

$$\Delta d_E = (2x_p + 3)$$

Для пиксела SE:

$$d_{new} = d_{old} + (2x_p - 2y_p + 5)$$

$$\Delta d_{SE} = (2x_p - 2y_p + 5)$$



Алгоритм Брезенхема (9) (окружность)

В начальной точке $(0, R)$

$$F(1, R - \frac{1}{2}) = 1 + (R^2 - R + \frac{1}{4}) - R^2 = \frac{5}{4} - R$$

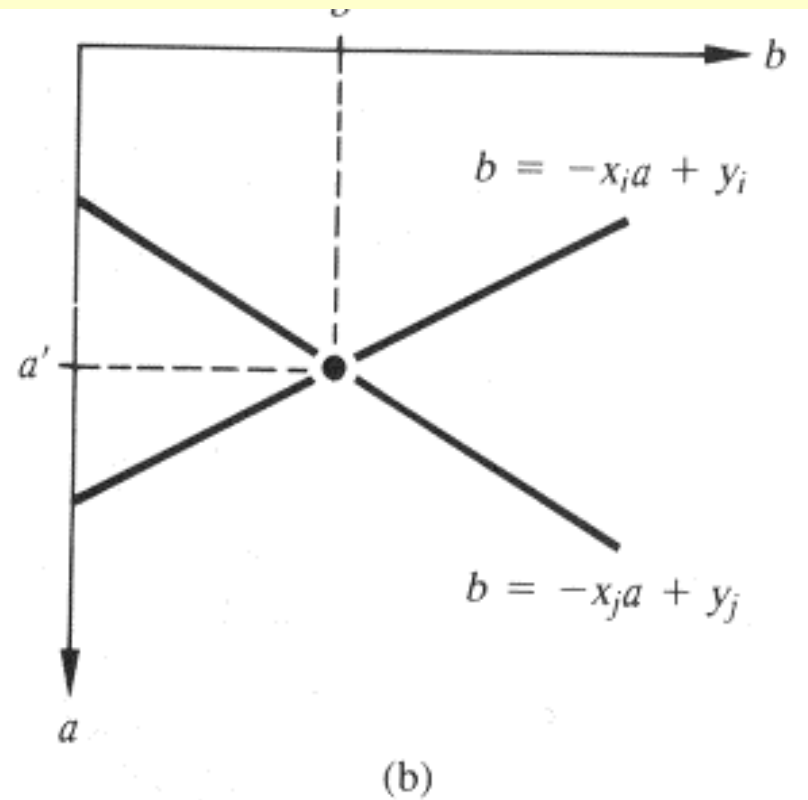
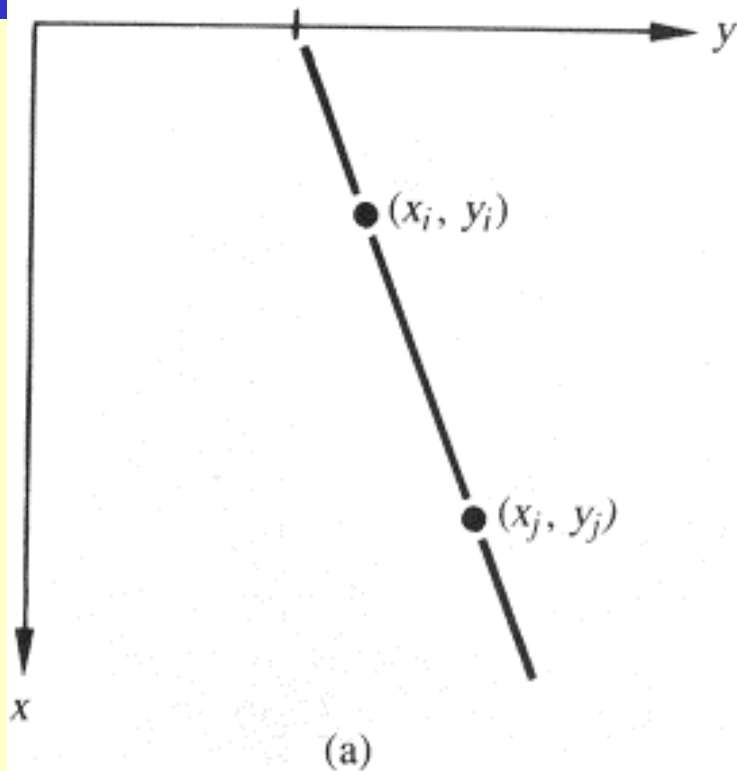
$$d_0 = \frac{5}{4} - R$$

И опять нужно исключить вещественные операции.

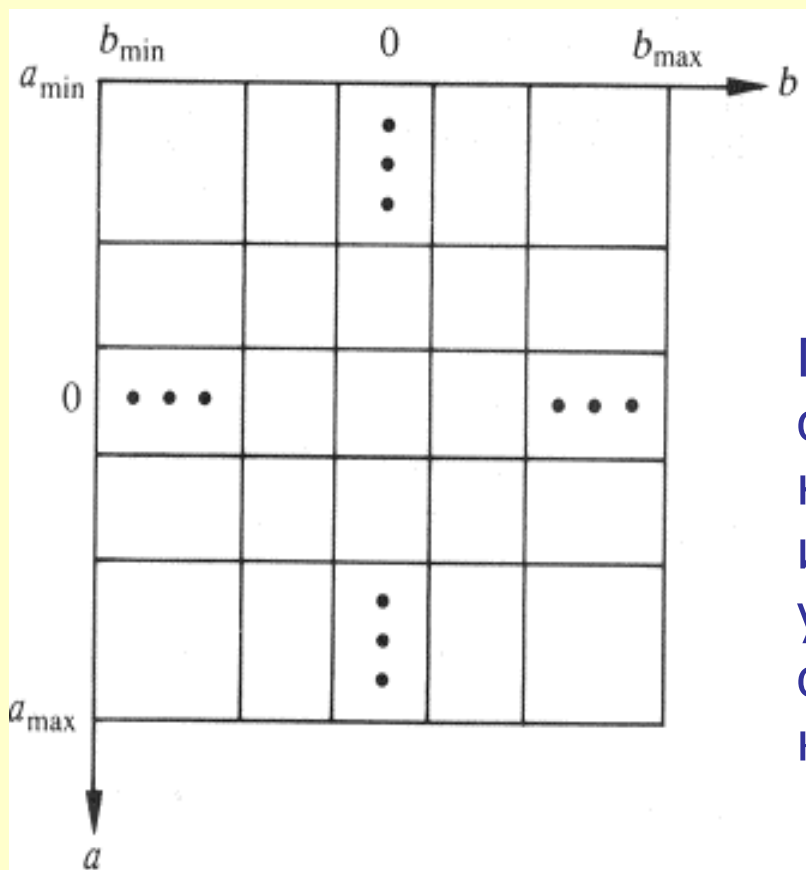
Сделав замену $h = d - 1/4$, получим $h = 1 - R$.

Тогда необходимо сравнивать h с $-1/4$, но так как приращения d – целые числа, то сравнивать можно с нулем.

Обнаружение прямых и дуг окружностей



Преобразование Хафа (1) (The Hough Transform)

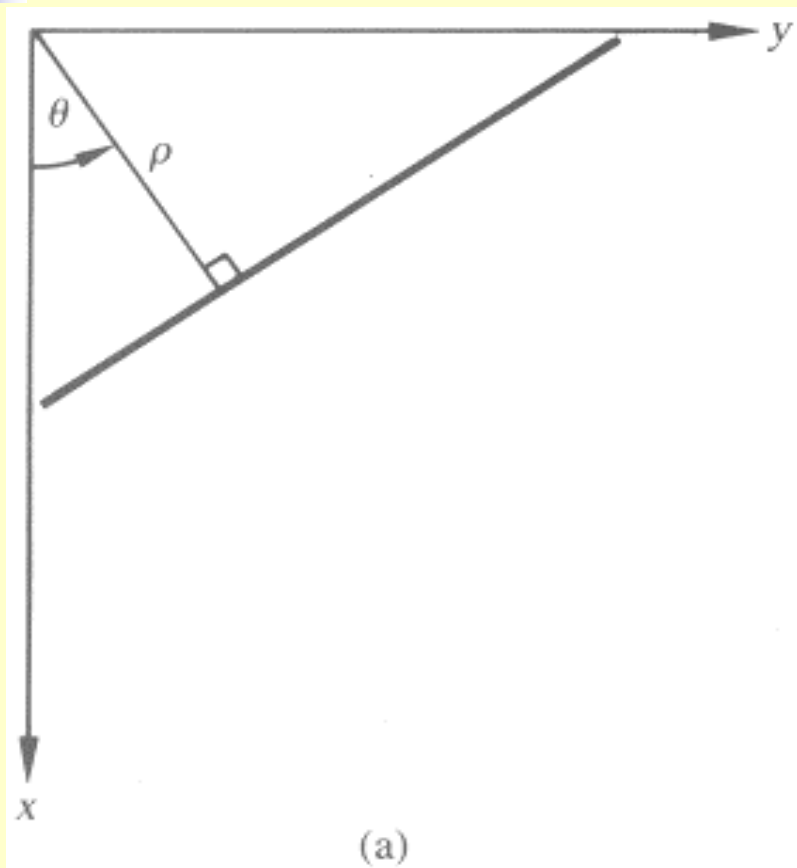


$$y_i = ax_i + b$$

$$b = -x_i a + y_i$$

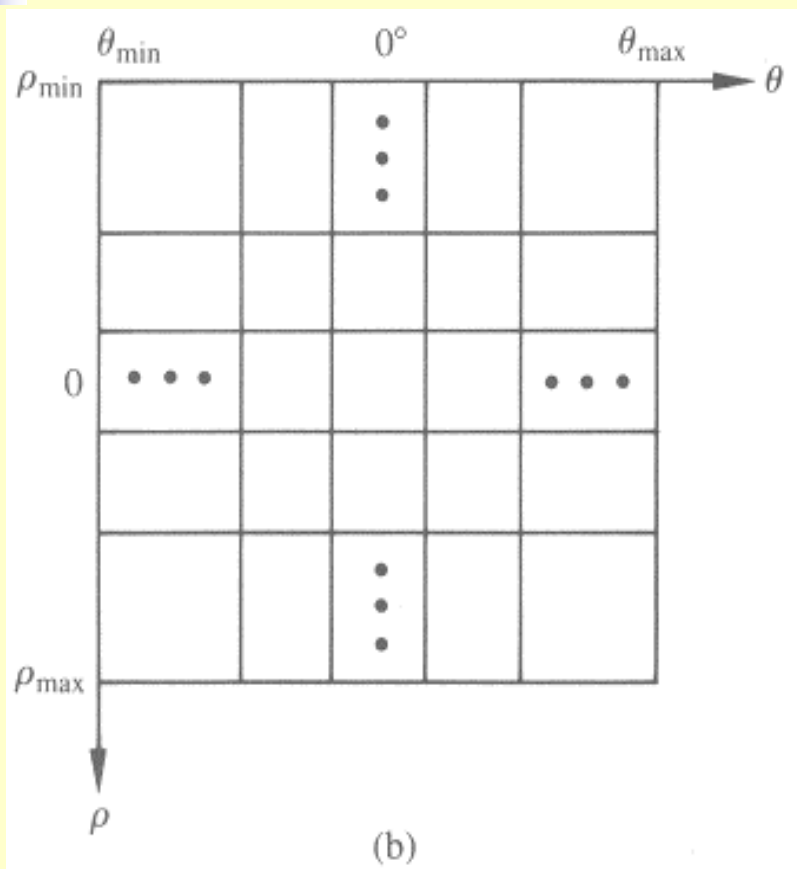
Преобразование Хафа – это метод обнаружения прямых и кривых линий на полутоновых и цветных изображениях. Метод позволяет указать параметры семейства кривых и обеспечивает поиск на изображениях кривых заданного класса.

Преобразование Хафа (2) (The Hough Transform)



$$x \cos \theta + y \sin \theta = \rho$$

Преобразование Хафа (3) (The Hough Transform)



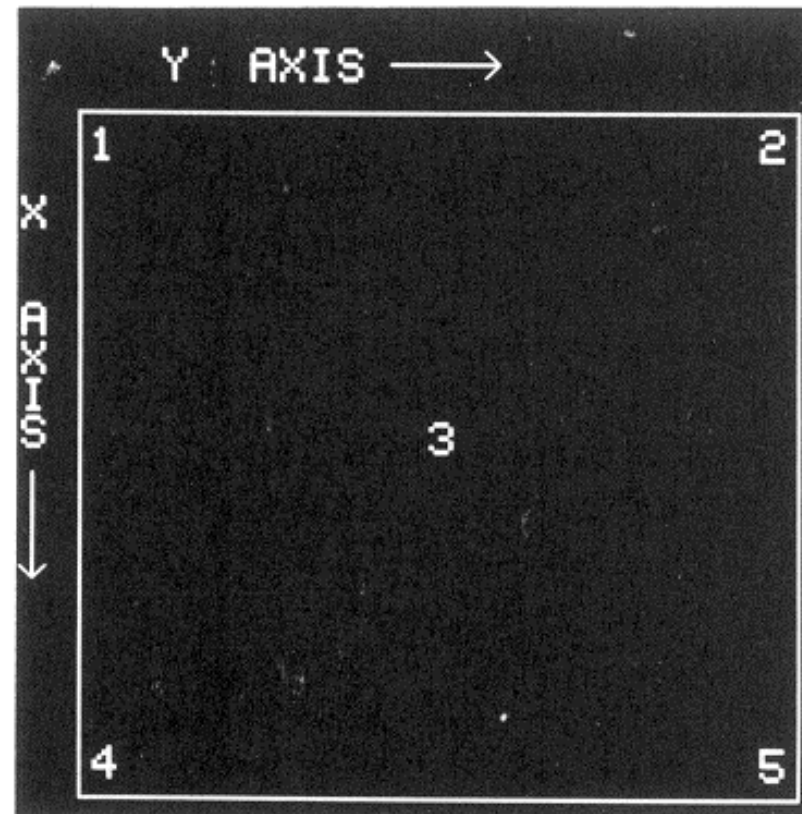
Прямая

$$x \cos \theta_j + y \sin \theta_j = \rho_i$$

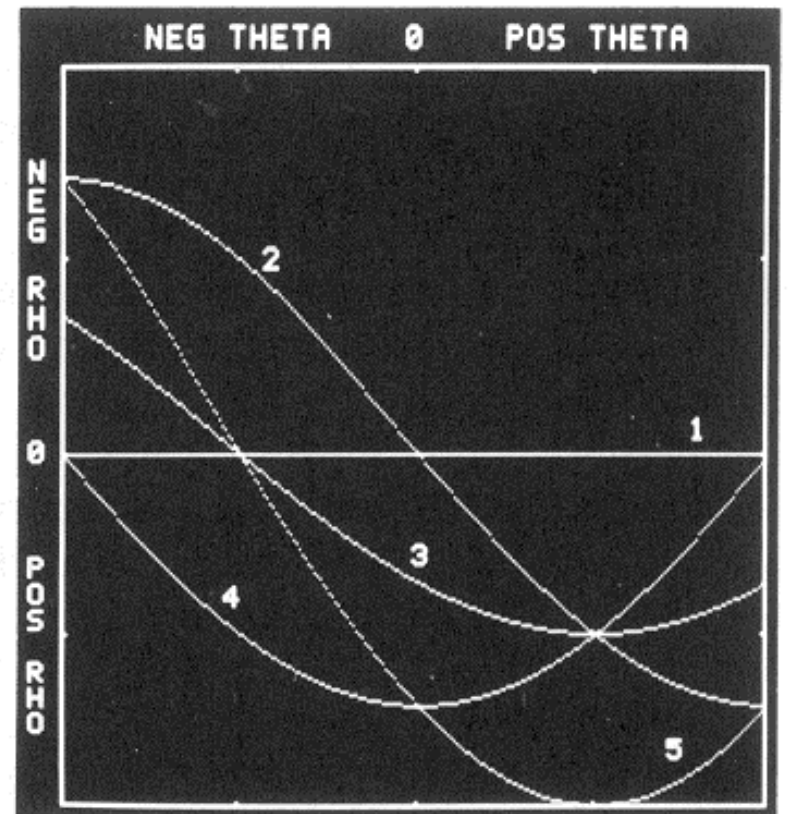
Окружность

$$(x - c_1)^2 + (y - c_2)^2 = c_3^2$$

Преобразование Хафа (4) (The Hough Transform)



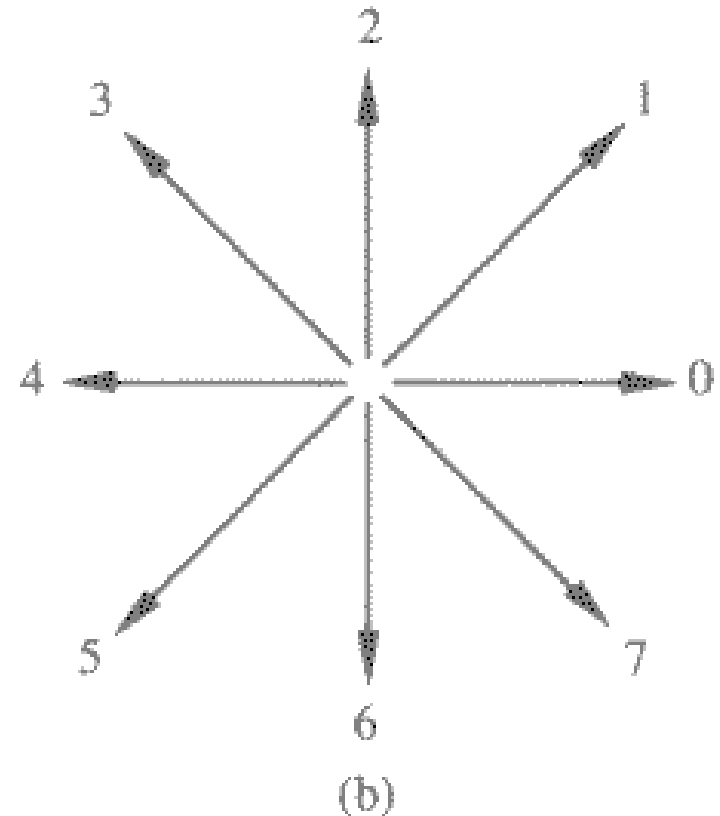
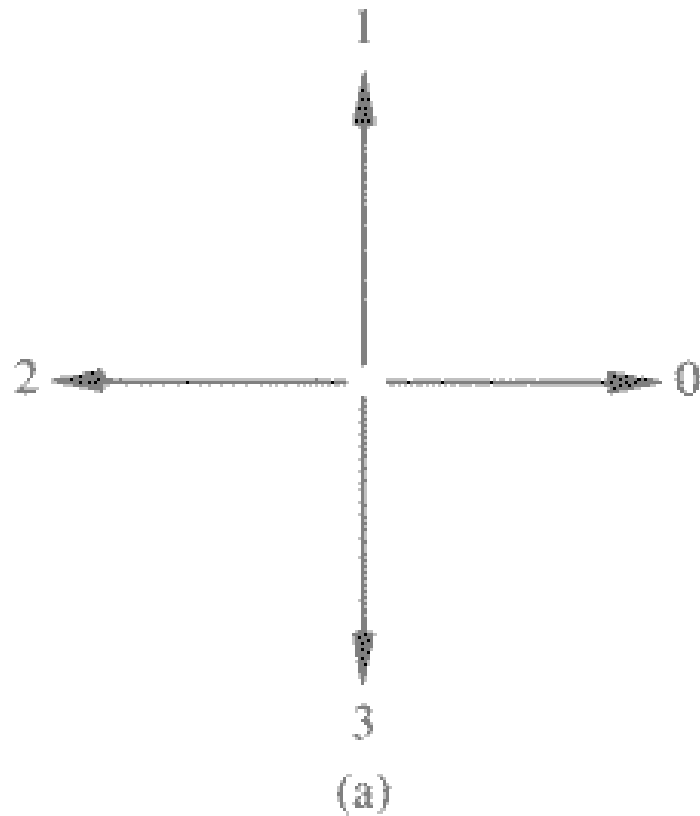
(a)



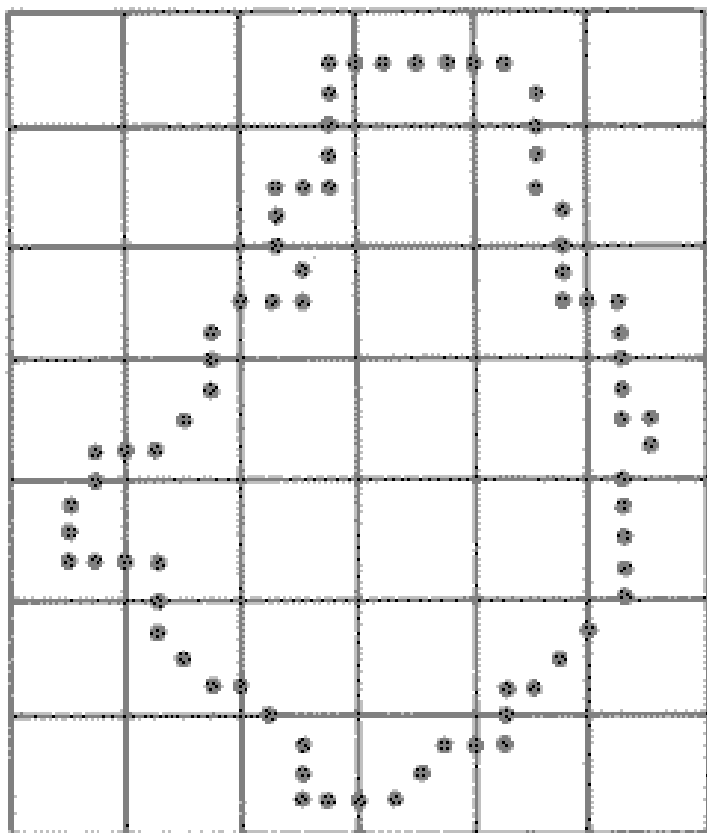
(b)

http://cgm.graphicon.ru/metodyi/hough_transform.html

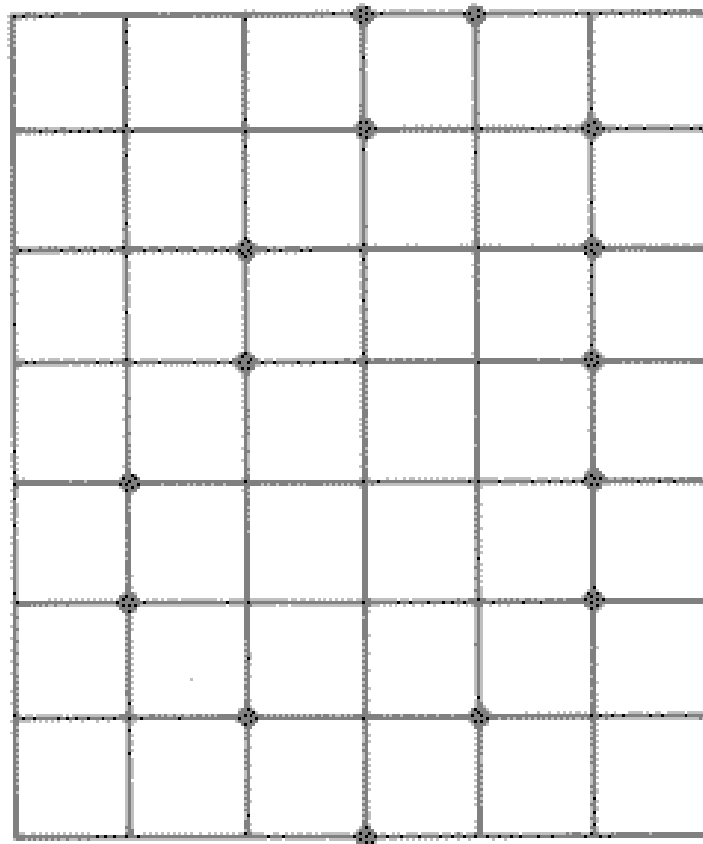
Цепное кодирование (Freeman's chain code)



Resampling (Изменение дискретизации)

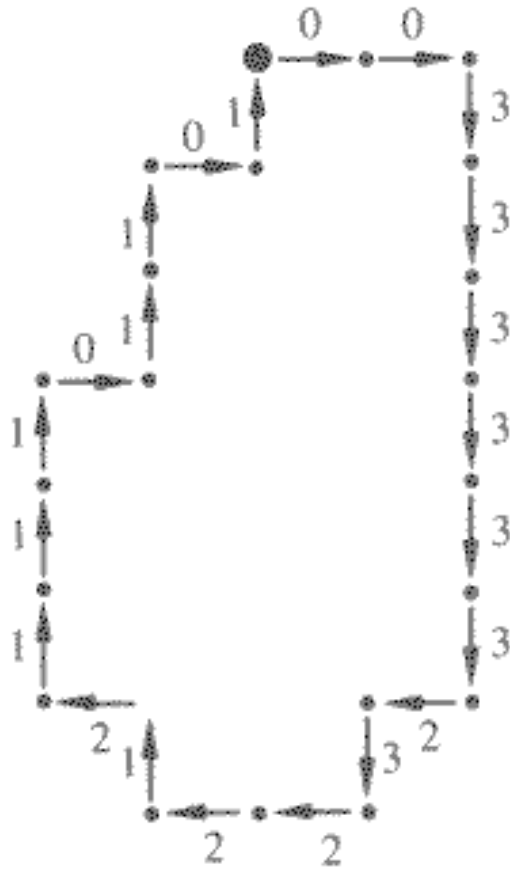


(a)

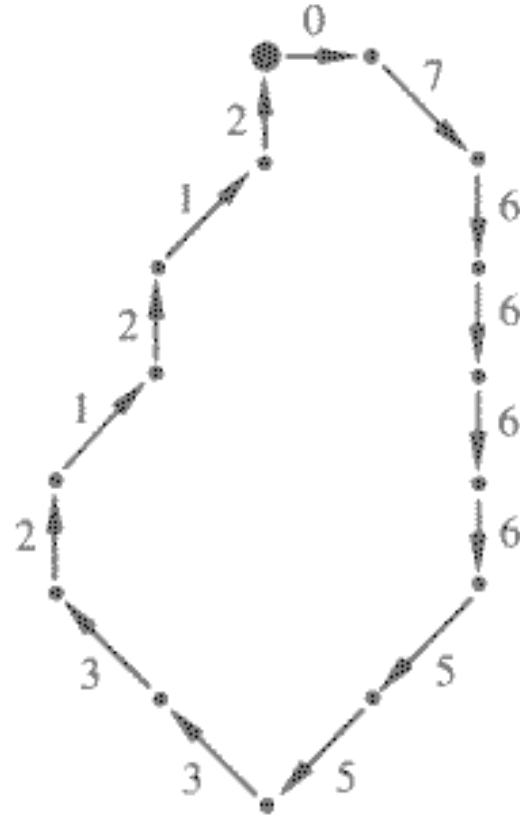


(b)

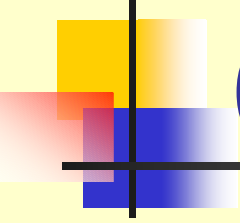
4-х и 8-ми связное кодирование



(c)



(d)



Сплайновые кривые (лекальные кривые)

В компьютерной графике сплайновой кривой называют любую составную кривую, сформированную полиномиальными участками (звеньями), которые удовлетворяют заданным условиям непрерывности (гладкости) на границах участков.

Интерполяция и аппроксимация



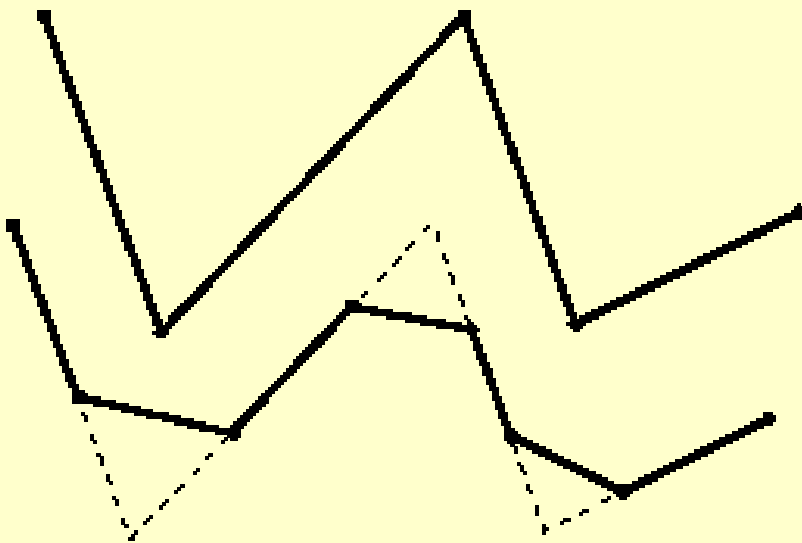
A set of six control points interpolated with piecewise continuous polynomial sections.

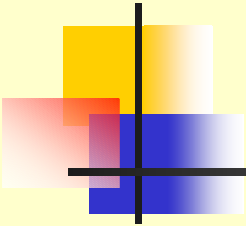


A set of six control points approximated with piecewise continuous polynomial sections.



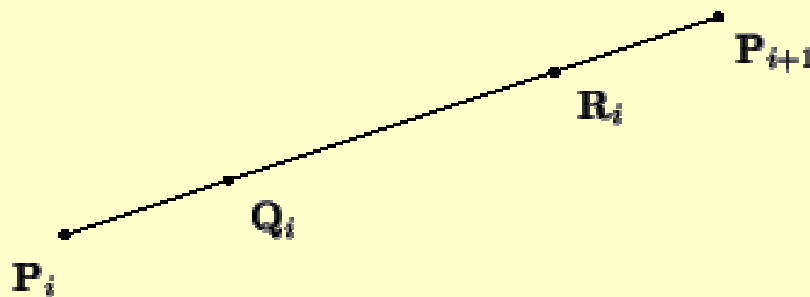
Срезаем углы





$$\{P_0, P_1, \dots, P_n\}$$

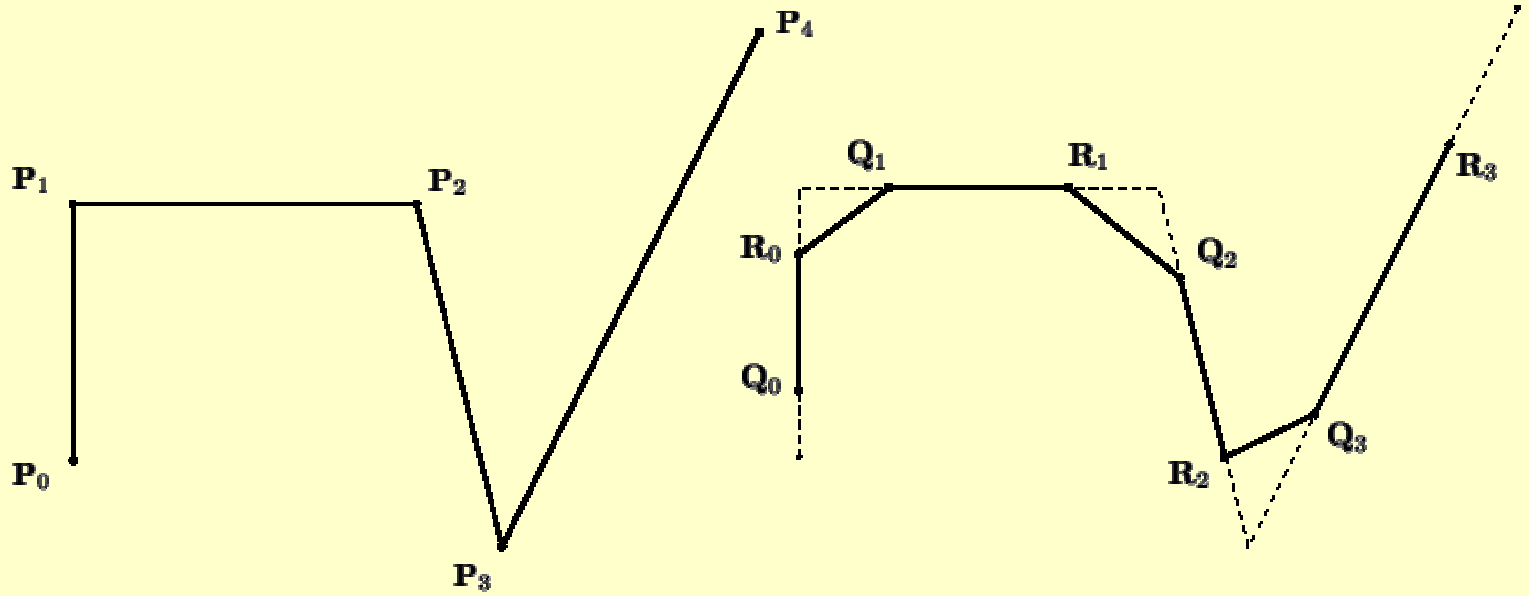
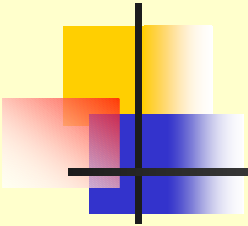
$$\{Q_0, R_0, Q_1, R_1, \dots, Q_{n-1}, R_{n-1}\}$$



$$Q_i = \frac{3}{4}P_i + \frac{1}{4}P_{i+1}$$

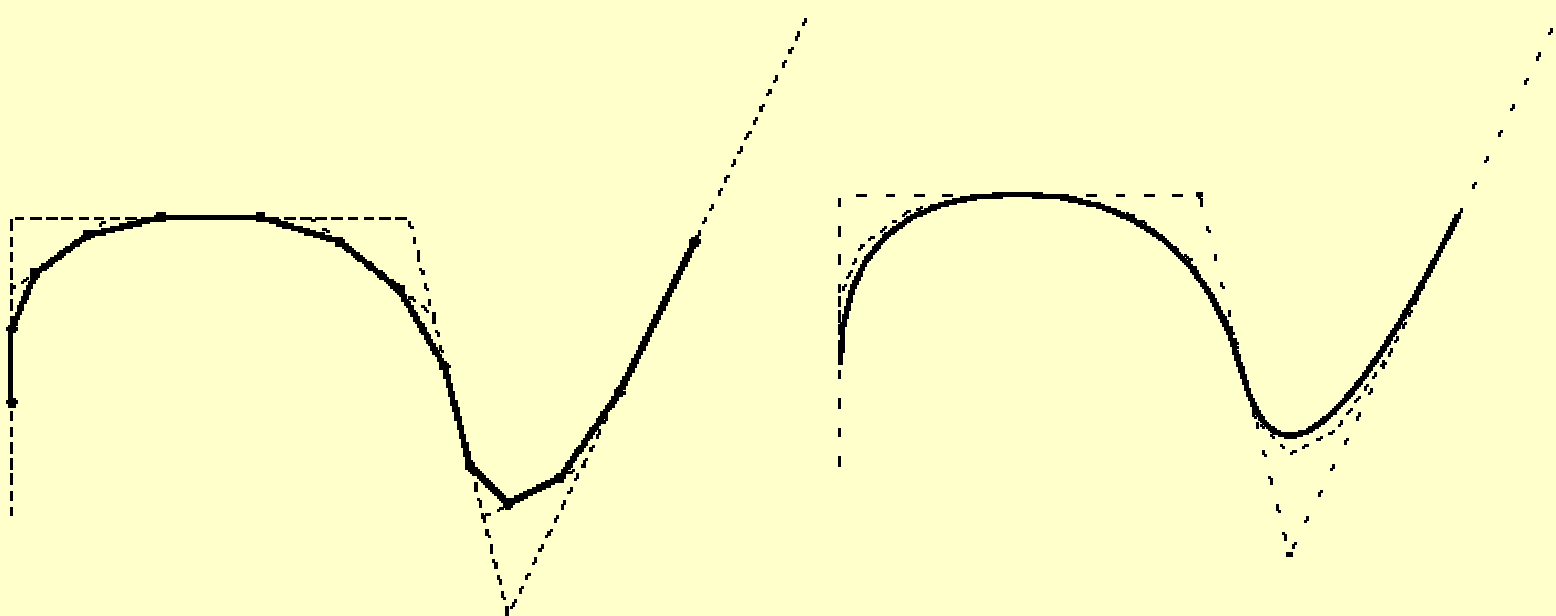
$$P_i = \frac{1}{4}P_i + \frac{3}{4}P_{i+1}$$

2n новых точек
рассматриваются как новая
управляющая ломаная



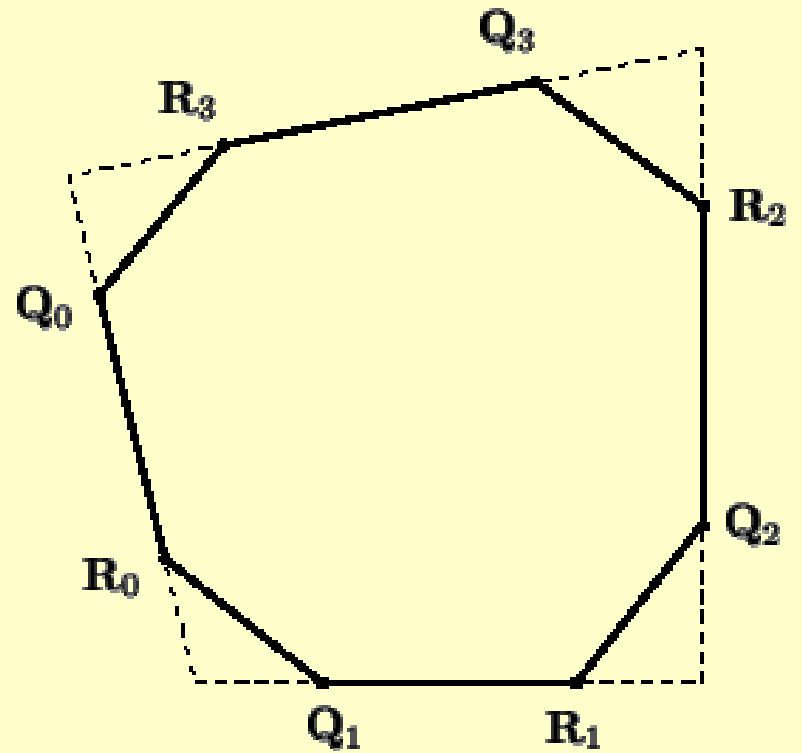
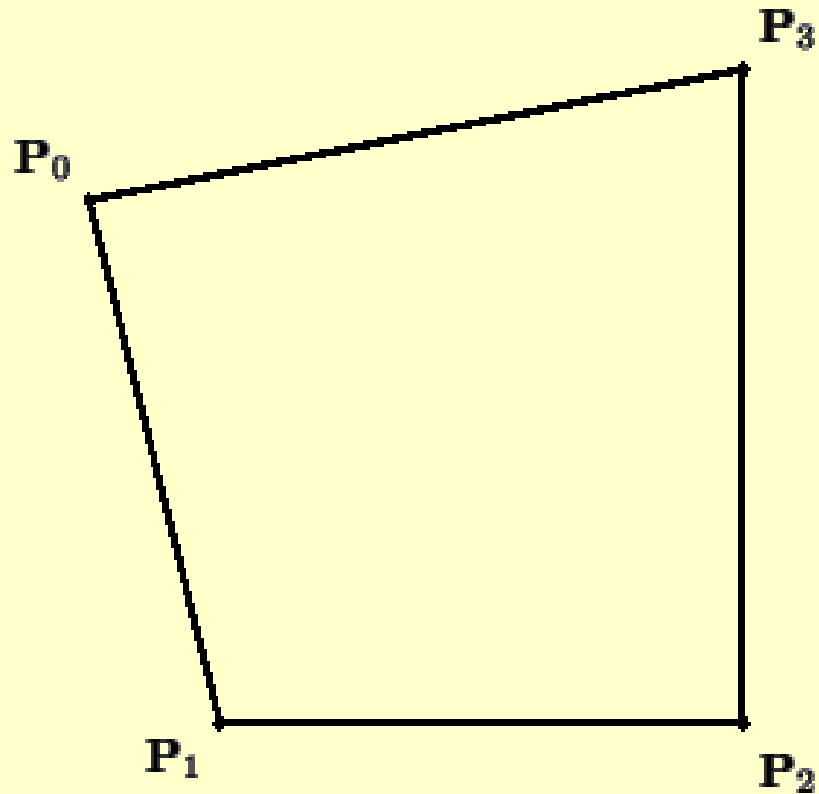


Итерация 2 и итерация N



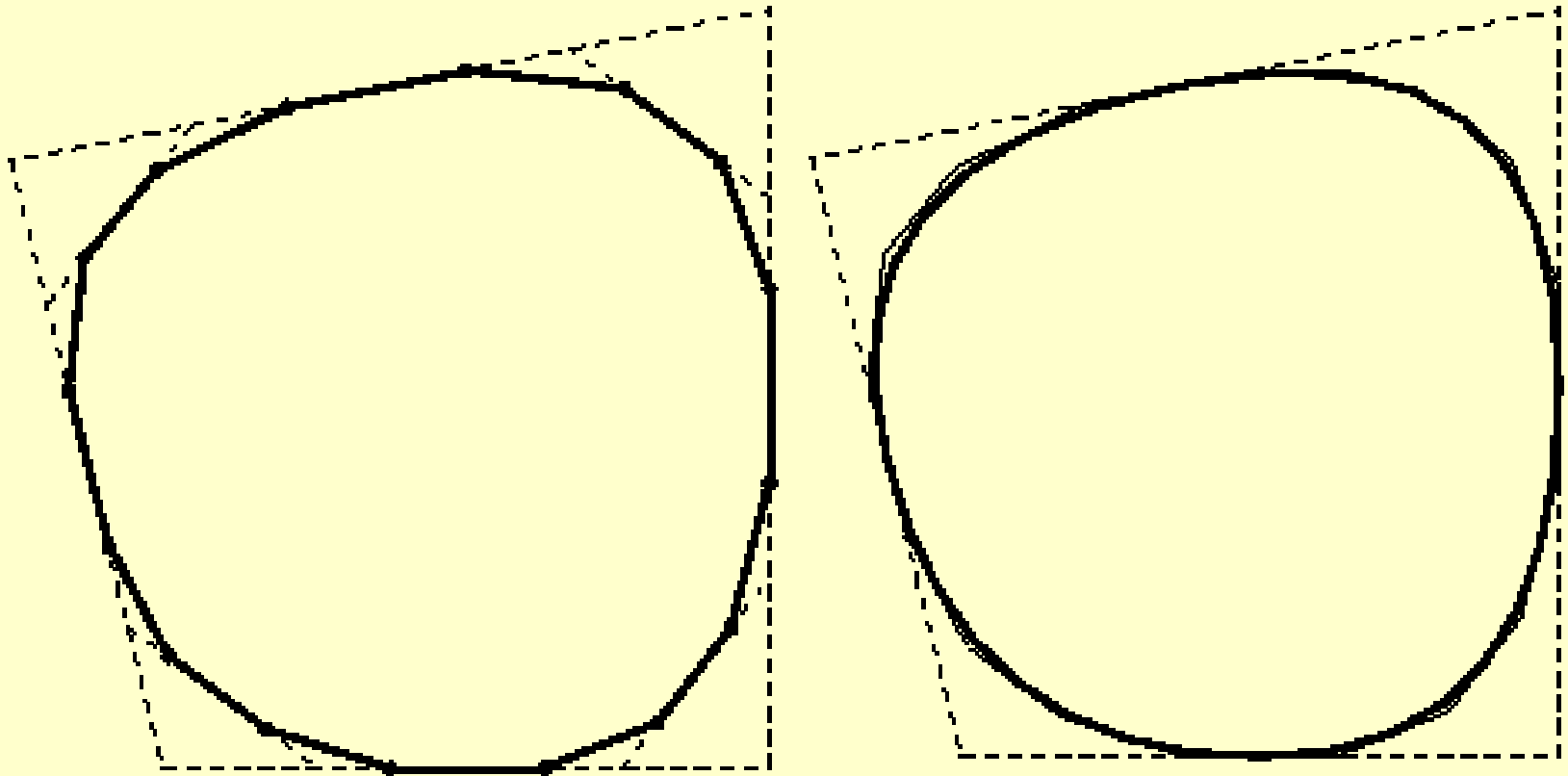


Замкнутые ломаные





Итерации



Алгоритм Чайкина

Chaiken's Algorithm

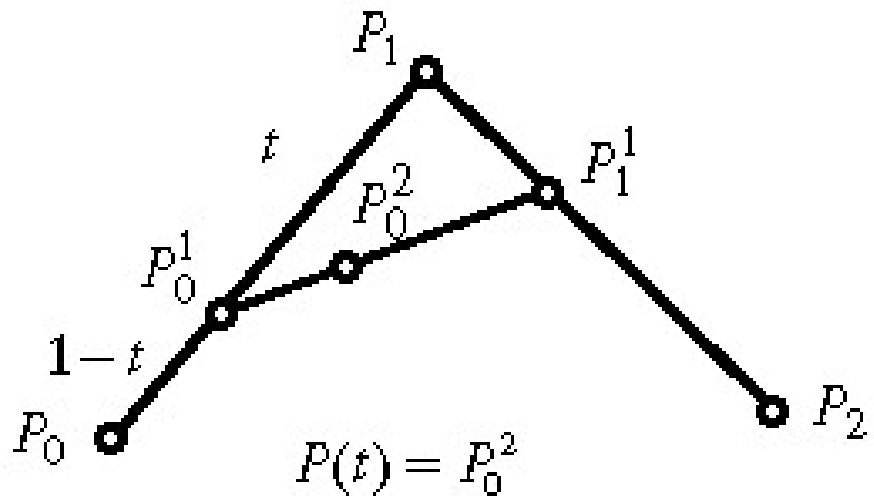
Step 1

$$P_0^1 = (1-t)P_0 + tP_1$$

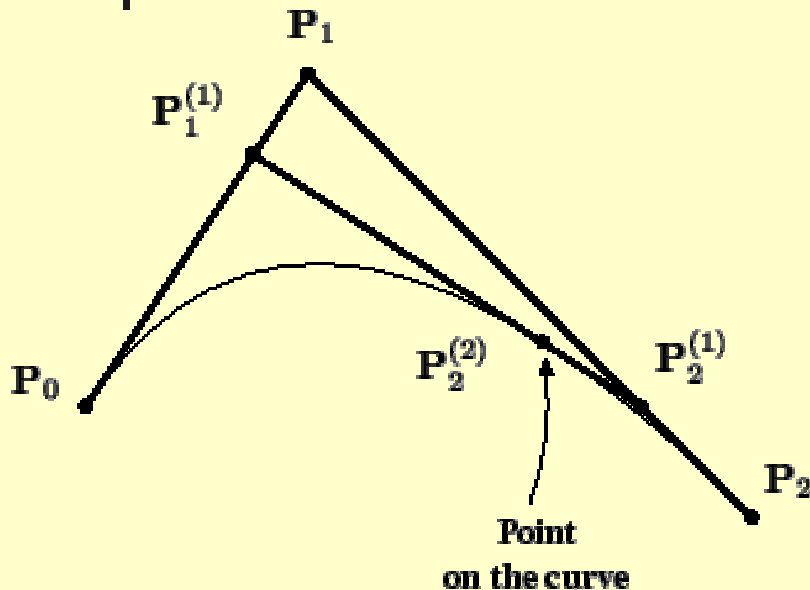
$$P_1^1 = (1-t)P_1 + tP_2$$

Step 2

$$P_0^2 = (1-t)P_0^1 + tP_1^1$$



Три контрольные точки



$$\mathbf{P}_1^{(1)} = (1-t)\mathbf{P}_0 + t\mathbf{P}_1 (= \mathbf{P}_0 + t(\mathbf{P}_1 - \mathbf{P}_0))$$

$$\mathbf{P}_2^{(1)} = (1-t)\mathbf{P}_1 + t\mathbf{P}_2$$

$$\mathbf{P}_2^{(2)} = (1-t)\mathbf{P}_1^{(1)} + t\mathbf{P}_2^{(1)}$$

$$\begin{aligned}\mathbf{P}(t) &= \mathbf{P}_2^{(2)}(t) \\ &= (1-t)\mathbf{P}_1^{(1)}(t) + t\mathbf{P}_2^{(1)}(t)\end{aligned}$$

$$\mathbf{P}_1^{(1)}(t) = (1-t)\mathbf{P}_0 + t\mathbf{P}_1, \text{ and}$$

$$\mathbf{P}_2^{(1)}(t) = (1-t)\mathbf{P}_1 + t\mathbf{P}_2$$



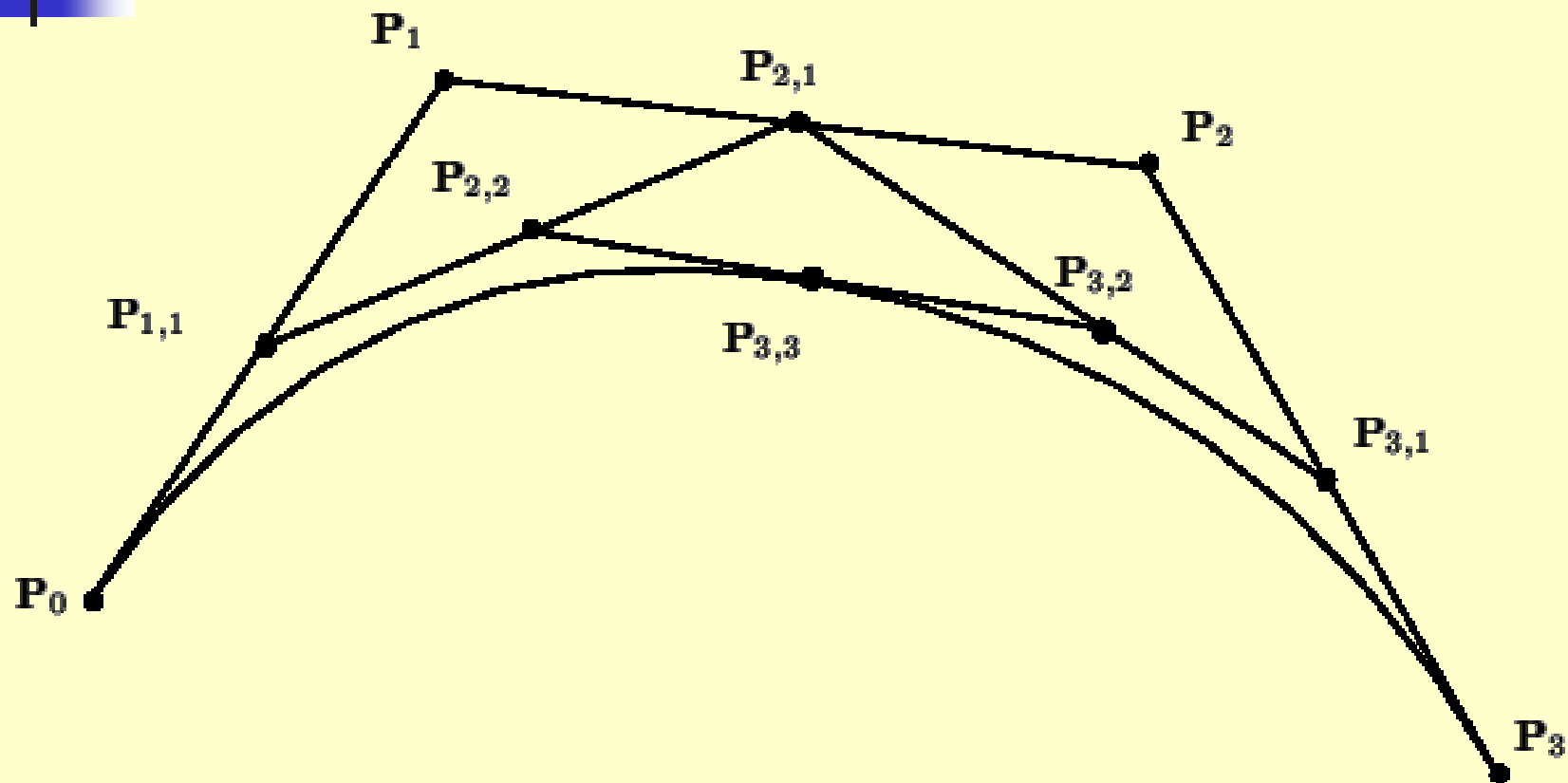
Три контрольные точки

$$\begin{aligned}\mathbf{P}(t) &= (1-t)\mathbf{P}_1^{(1)}(t) + t\mathbf{P}_2^{(1)}(t) \\ &= (1-t)[(1-t)\mathbf{P}_0 + t\mathbf{P}_1] + t[(1-t)\mathbf{P}_1 + t\mathbf{P}_2] \\ &= (1-t)^2\mathbf{P}_0 + (1-t)t\mathbf{P}_1 + t(1-t)\mathbf{P}_1 + t^2\mathbf{P}_2 \\ &= (1-t)^2\mathbf{P}_0 + 2t(1-t)\mathbf{P}_1 + t^2\mathbf{P}_2\end{aligned}$$

$$\begin{aligned}\frac{d}{dt}\mathbf{P}(t) &= -2(1-t)\mathbf{P}_0 + [-2t + 2(1-t)]\mathbf{P}_1 + 2t\mathbf{P}_2 \\ &= 2[(1-t)(\mathbf{P}_1 - \mathbf{P}_0) + t(\mathbf{P}_2 - \mathbf{P}_1)]\end{aligned}$$

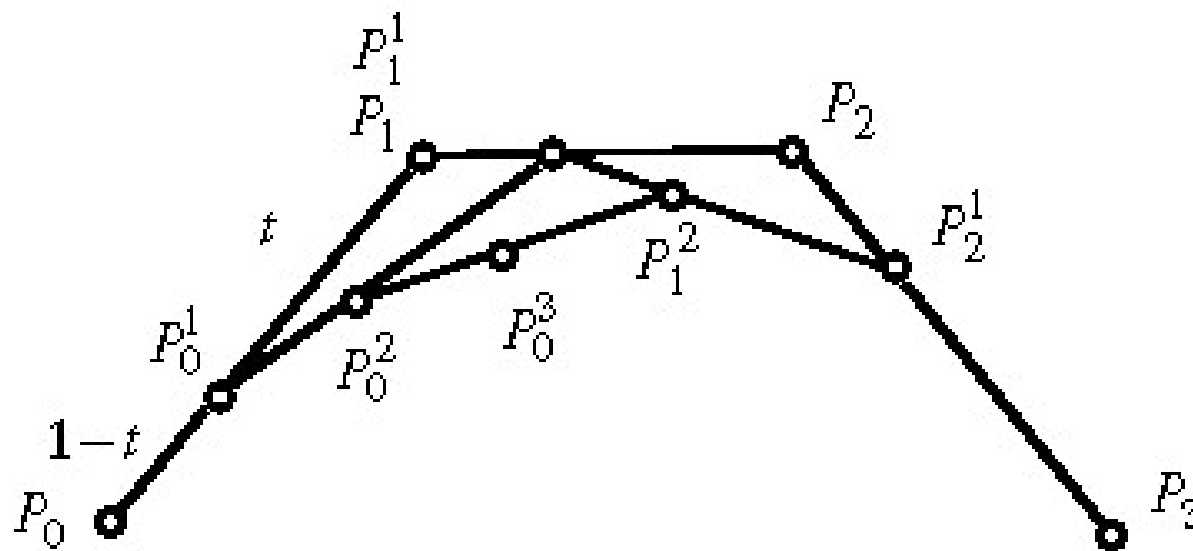
$$\frac{d}{dt}\mathbf{P}(0) = 2(\mathbf{P}_1 - \mathbf{P}_0)$$

Кривые Безье (по 4 точкам)



Четыре контрольные точки

Bezier Curves





Четыре контрольные точки

$$\begin{aligned}\mathbf{P}(t) &= \mathbf{P}_3^{(3)}(t) \\ &= t\mathbf{P}_3^{(2)}(t) + (1-t)\mathbf{P}_2^{(2)}(t) \\ &= t \left[t\mathbf{P}_3^{(1)}(t) + (1-t)\mathbf{P}_2^{(1)}(t) \right] \\ &\quad + (1-t) \left[t\mathbf{P}_2^{(1)}(t) + (1-t)\mathbf{P}_1^{(1)}(t) \right] \\ &= t^2\mathbf{P}_3^{(1)}(t) + 2t(1-t)\mathbf{P}_2^{(1)}(t) + (1-t)^2\mathbf{P}_1^{(1)}(t) \\ &= t^2 [t\mathbf{P}_3 + (1-t)\mathbf{P}_2] + 2t(1-t) [t\mathbf{P}_2 + (1-t)\mathbf{P}_1] \\ &\quad + (1-t)^2 [t\mathbf{P}_1 + (1-t)\mathbf{P}_0] \\ &= t^3\mathbf{P}_3 + 3t^2(1-t)\mathbf{P}_2 + 3t(1-t)^2\mathbf{P}_1 + (1-t)^3\mathbf{P}_0\end{aligned}$$

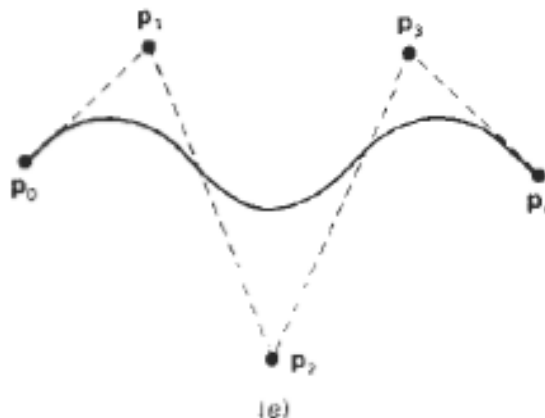
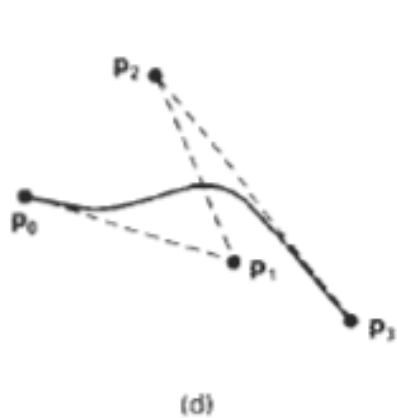
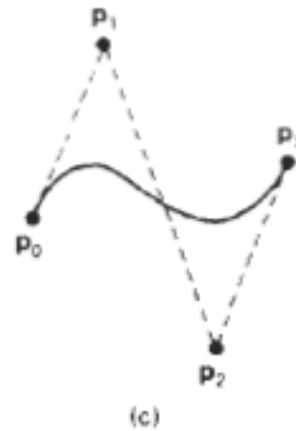
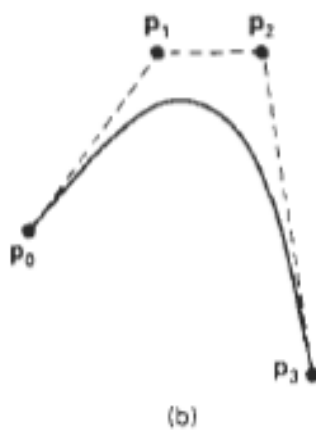
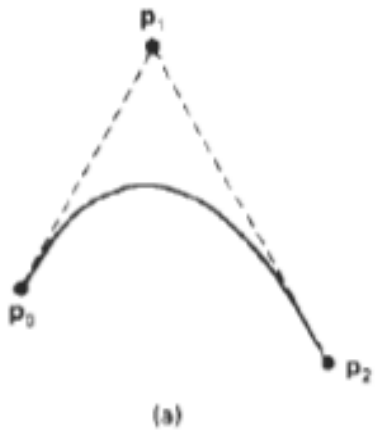


Кривые Безье

This spline approximation method was developed by the French engineer Pierre Bezier for use in the design of Renault automobile bodies. **Bezier** splines have a number of properties that make them highly useful and convenient for curve and surface design. They are also easy to implement. For these reasons, Bezier splines are widely available in various CAD systems, in general graphics packages (such as OpenGL), and in assorted drawing and painting packages (such as Aldus Superpaint and Cricket Draw).

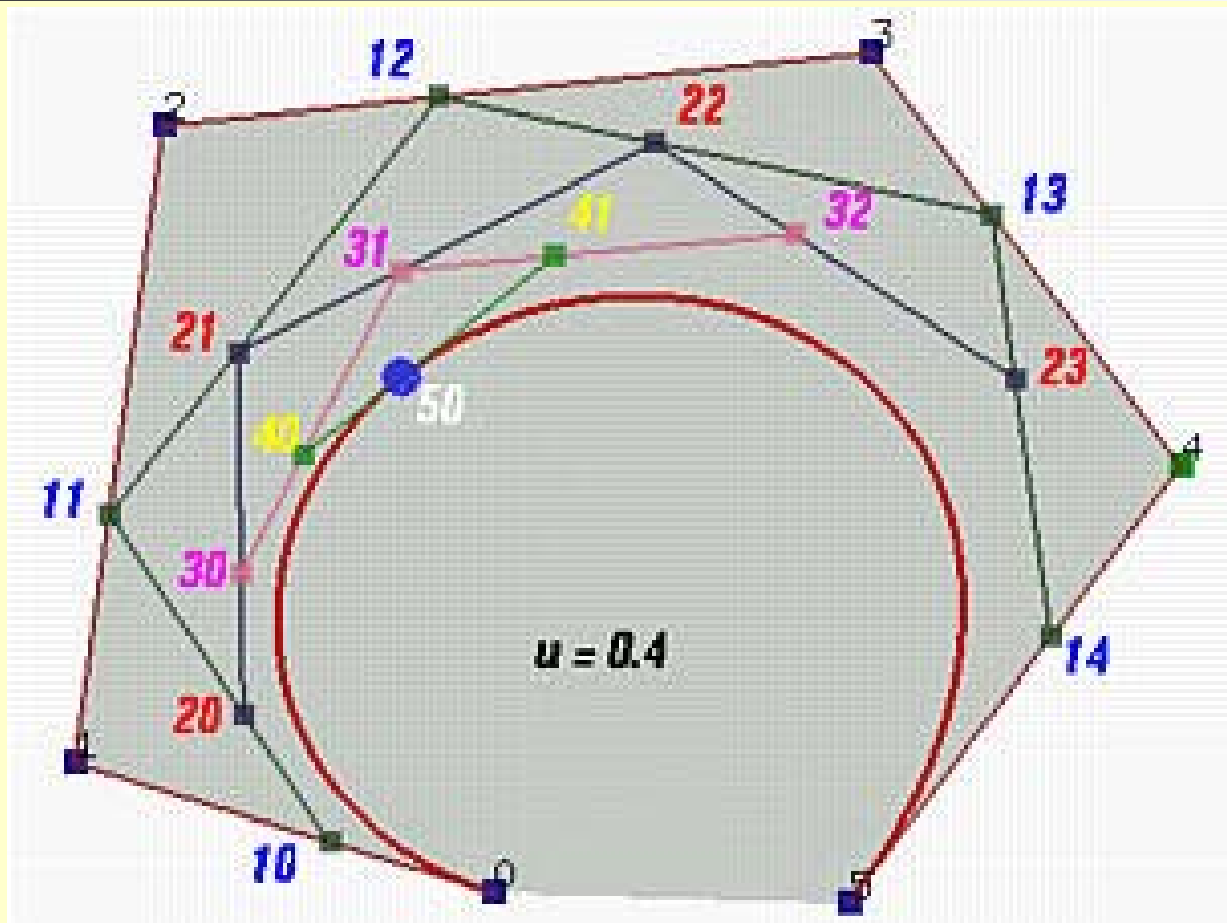
In general, a Bezier curve section can be fitted to any number of control points. The number of control points to be approximated and their relative position determine the degree of the Bezier polynomial. As with the interpolation splines, a Bezier curve can be specified with boundary conditions, with a characterizing matrix, or with blending functions. For general Bezier curves, the blending-function specification is the most convenient.

Кривые Безье



Examples of two-dimensional Bezier curves generated from three, four, and five control points. Dashed lines connect the control-point positions.

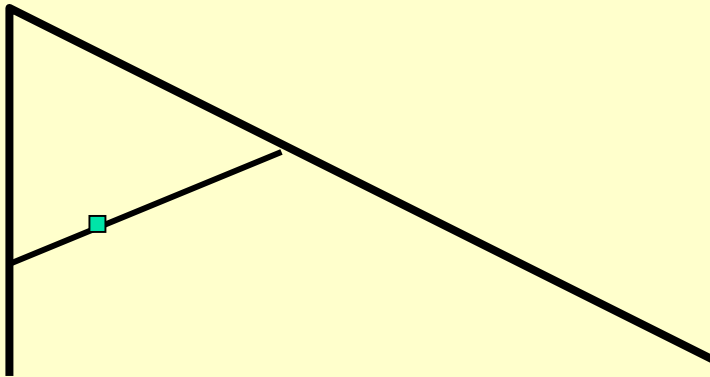
Шесть контрольных точек



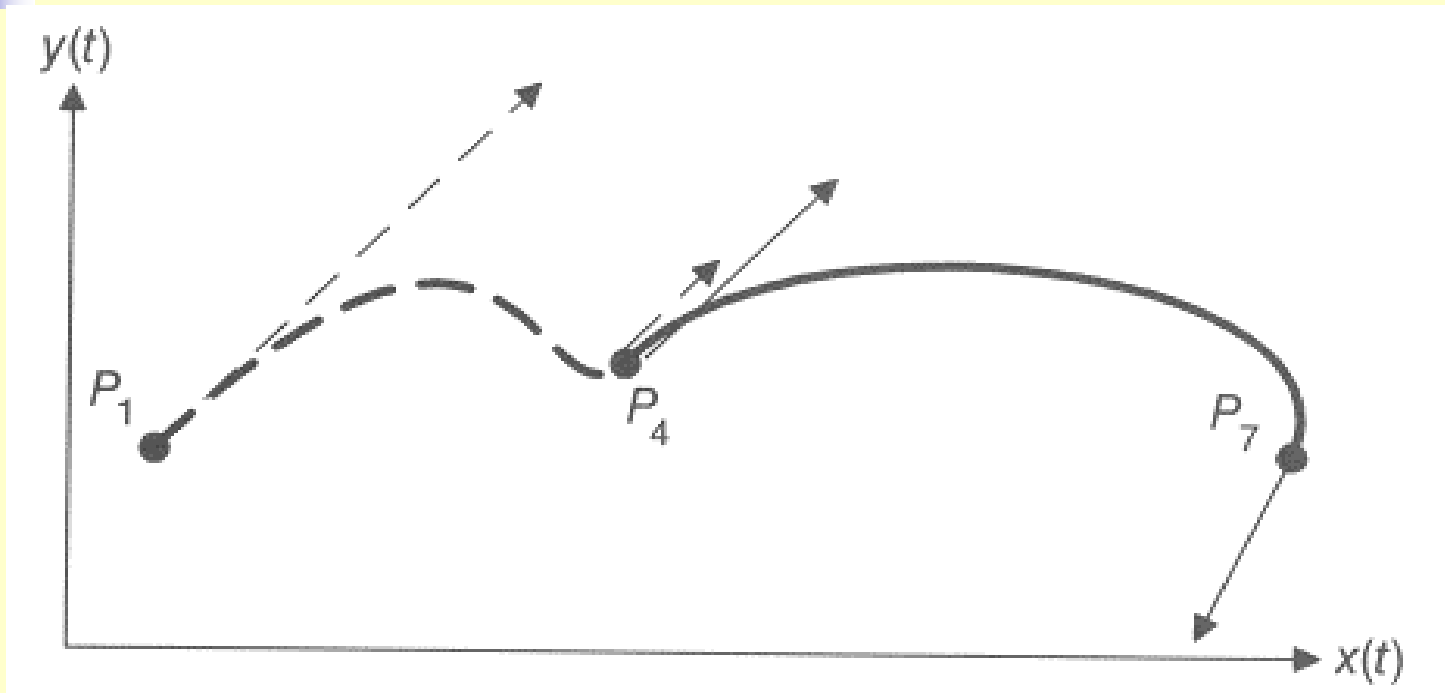


Задача (1)

Ломаная Безье задана тремя точками $(0, 0)$, $(0, 9)$, $(18, 0)$. Определите координаты точки на кривой Безье при $t = 1/3$.



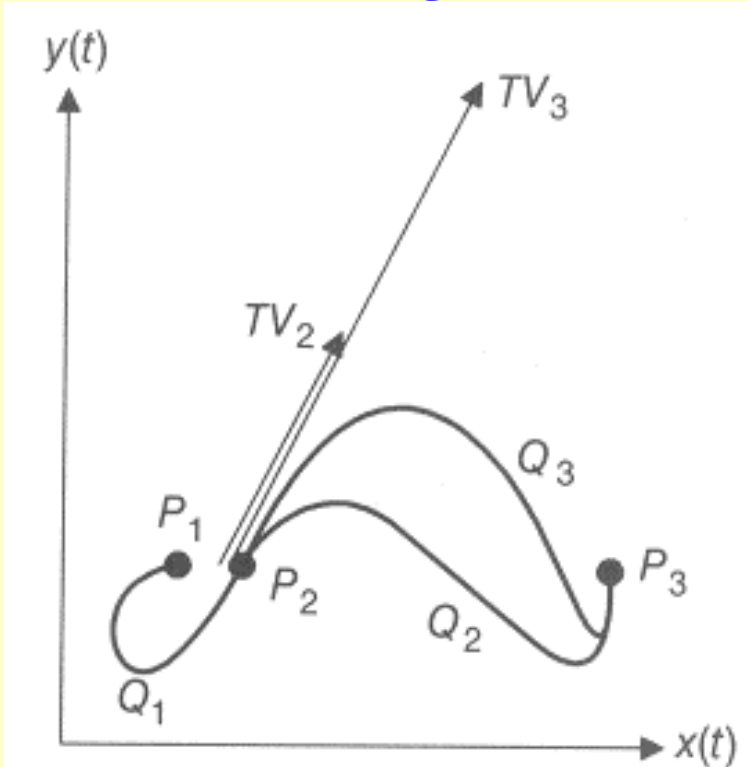
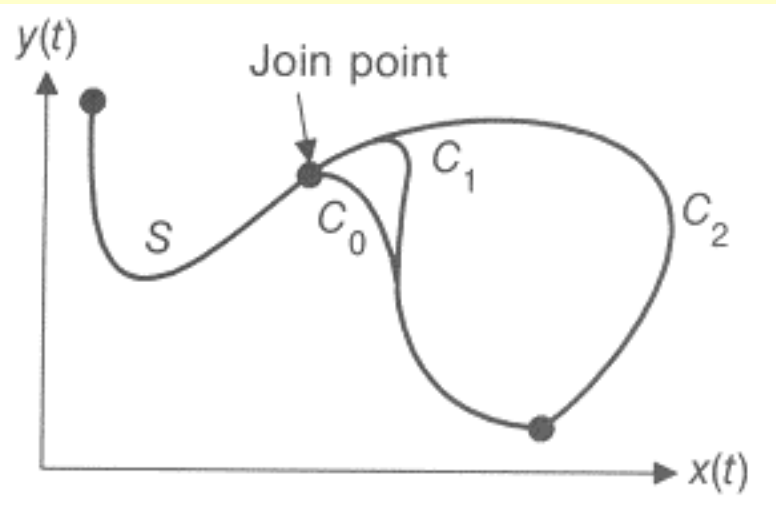
G^0, G^1 - геометрическая непрерывность



Two curve segments are said G^k *geometric continuous* at the joining point if and only if all i -th derivatives, i less than or equal to k , computed with arc length parameters agree at the joining point.

C^0 C^1 C^2 G^1

Two C^0 curve segments are said G^1 *geometric continuous* at the joining point if and only if vectors $f'(u)$ and $g'(v)$ are in the *same* direction at the joining point. Note that $f'(u)$ and $g'(v)$ are evaluated at the joining point.



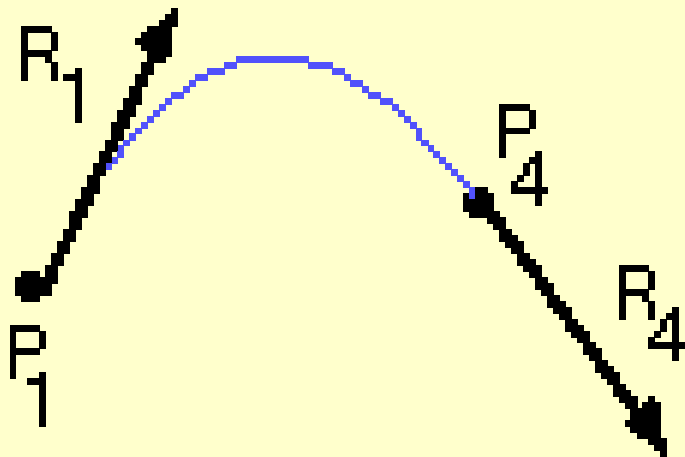


Задача (2)

Заданы два звена сплайна $\alpha(t) = (t, t^2 + 1)$ и $\beta(t) = (2t + 1, t^3 + 4t + 2)$, $0 \leq t \leq 1$.

Обеспечивается ли C^0 , C^1 , G^1 непрерывность в точке соединения $\alpha(1)$, $\beta(0)$?

Кривые Эрмита



$$G_H = [P_1, P_4, R_1, R_4]^T$$

$$x(t) = T A_H = T M_H G_H$$

$$T = [t^3 \ t^2 \ t \ 1]$$

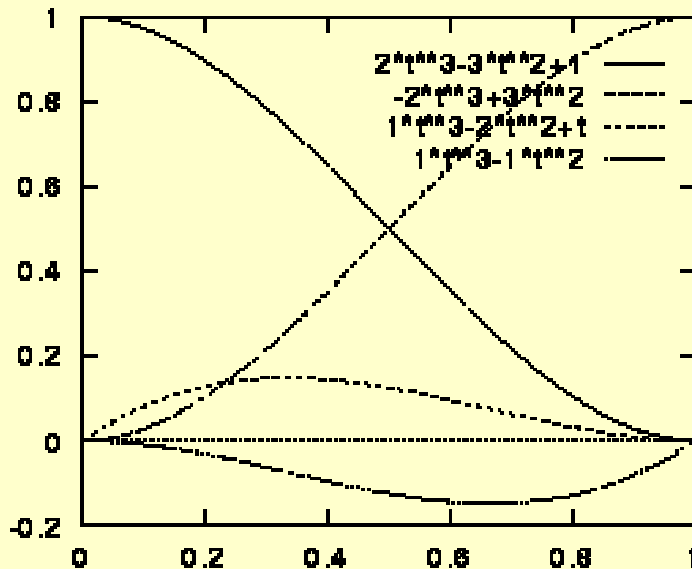
$$x(0) = P_{1x} = [0 \ 0 \ 0 \ 1] M_H G_{Hx}$$

$$x(1) = P_{4x} = [1 \ 1 \ 1 \ 1] M_H G_{Hx}$$

$$x'(0) = R_{1x} = [0 \ 0 \ 1 \ 0] M_H G_{Hx}$$

$$x'(1) = R_{4x} = [3 \ 2 \ 1 \ 0] M_H G_{Hx}$$

Смешивающие функции Эрмита (Hermite blending functions)



$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$f_1(t) = 2t^3 - 3t^2 + 1$$

$$f_2(t) = -2t^3 + 3t^2$$

$$f_3(t) = t^3 - 2t^2 + t$$

$$f_4(t) = t^3 - t^2$$

Кубические кривые Эрмита

Cubic Hermite Interpolation

Given: $P(0), P(1), P'(0), P'(1)$

Compute: $P(t) = at^3 + bt^2 + ct + d$ (Cubic Polynomial)

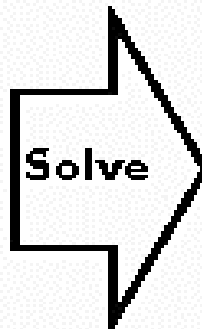
Derivative: $P'(t) = 3at^2 + 2bt + c$

$$P(0) = h_0 = d$$

$$P(1) = h_1 = a + b + c + d$$

$$P'(0) = h_2 = c$$

$$P'(1) = h_3 = 3a + 2b + c$$



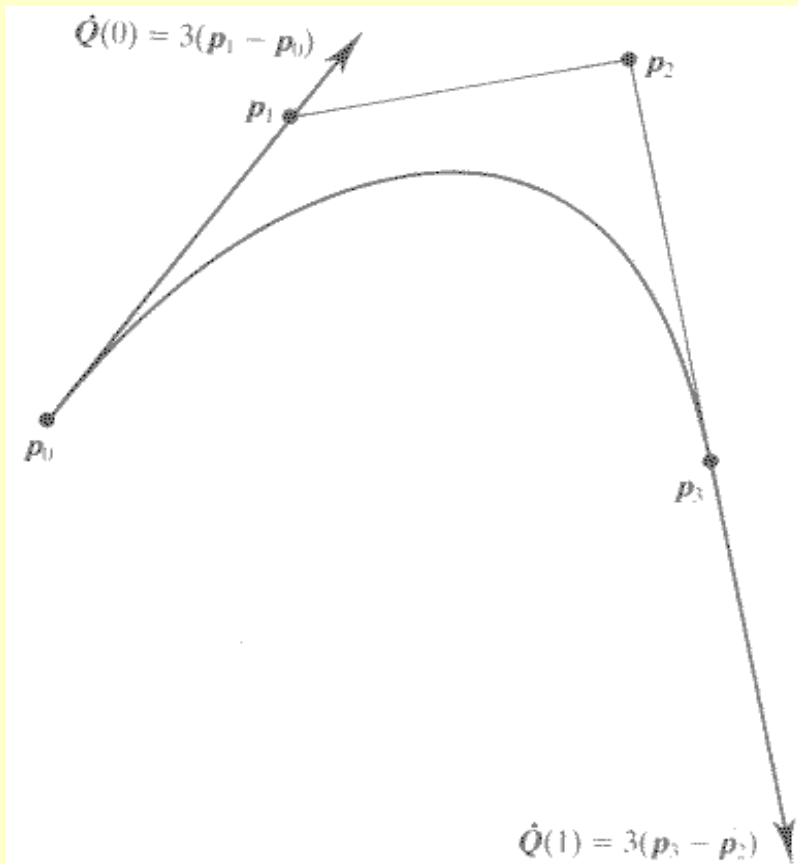
$$a = 2h_0 - 2h_1 + h_2 + h_3$$

$$b = -3h_0 + 3h_1 - 2h_2 - h_3$$

$$c = h_2$$

$$d = h_0$$

Связь кривых Эрмита и Безье



$$\frac{dQ(0)}{du} = n(p_1 - p_0)$$

$$\frac{dQ(1)}{du} = n(p_n - p_{n-1})$$

For example, for a cubic we have:

$$\dot{Q}(0) = 3(p_1 - p_0)$$

$$\dot{Q}(1) = 3(p_3 - p_2)$$

$$Q(u) = UM_B P_c = [u^3 \ u^2 \ u \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$



Полиномы Бернштейна

$$P(t) = \sum_{i=0}^3 P_i B_{i,3}(t)$$

Bezier Curves = Bernstein Polynomials

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

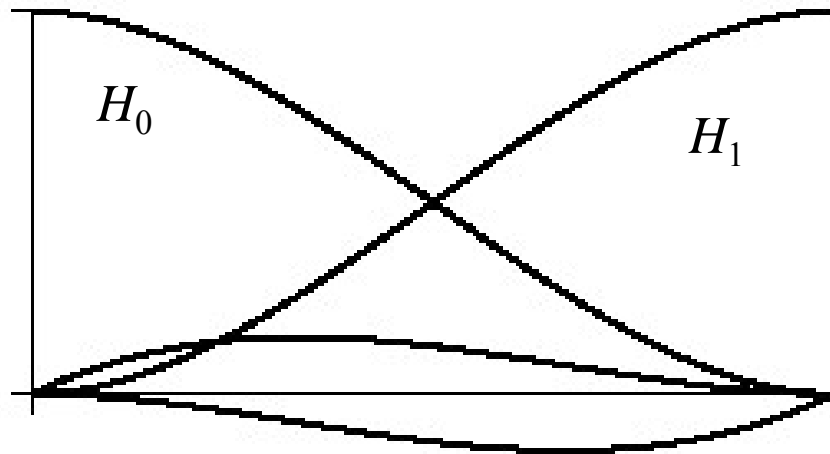
$$B_0^3(t) = \binom{3}{0} t^0 (1-t)^3 = -t^3 + 3t^2 - 3t + 1$$

$$B_1^3(t) = \binom{3}{1} t^1 (1-t)^2 = \frac{3!}{2!} (t(1-2t+t^2)) = 3t^3 - 6t^2 + 3t$$

$$B_2^3(t) = \binom{3}{2} t^2 (1-t)^1 = \frac{3!}{2!} (t^2(1-t)) = -3t^3 + 3t^2$$

$$B_3^3(t) = \binom{3}{3} t^3 (1-t)^0 = t^3$$

Hermite Basis Functions

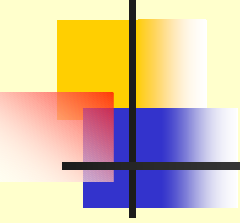


$$H_0(t) = 2t^3 + 3t^2 + 1$$

$$H_1(t) = -2t^3 + 3t^2$$

$$H_2(t) = t^3 - 2t^2 + t$$

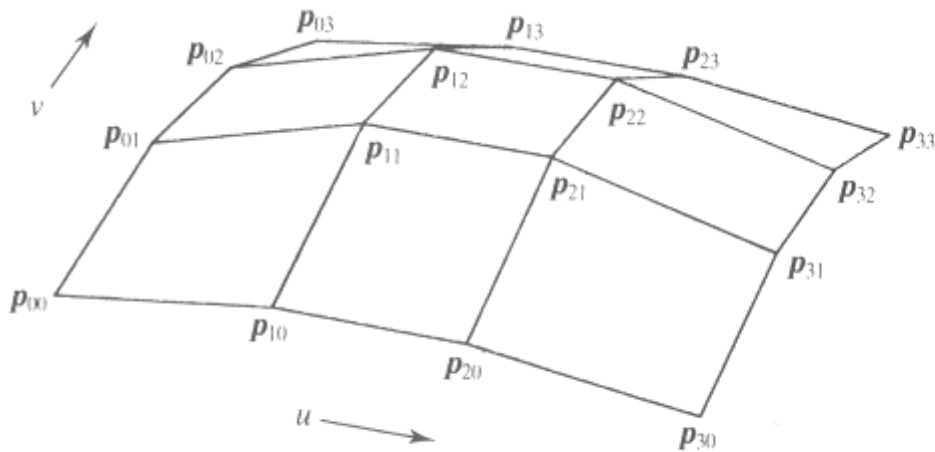
$$H_3(t) = t^3 - t^2$$



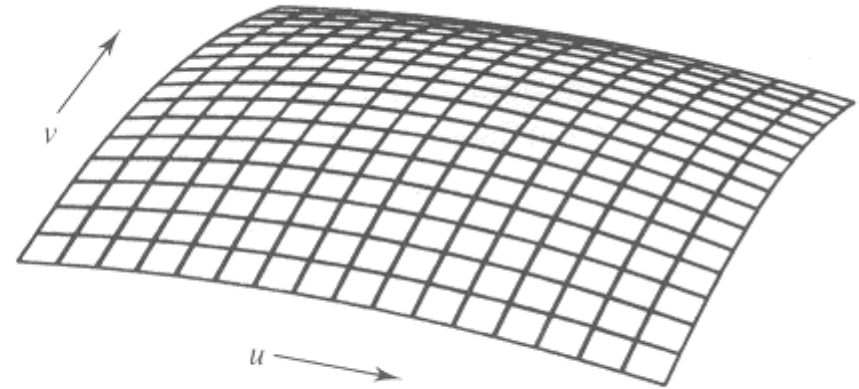
$$P(t) = T \cdot M_H \cdot G_H = B_H \cdot G_H =$$

$$= (2t^3 - 3t^2 + 1)P_1 + (-2t^3 + 3t^2)P_4 + (t^3 - 2t^2 + t)R_1 + (t^3 - t^2)R_4$$

Поверхности Безье



(a)

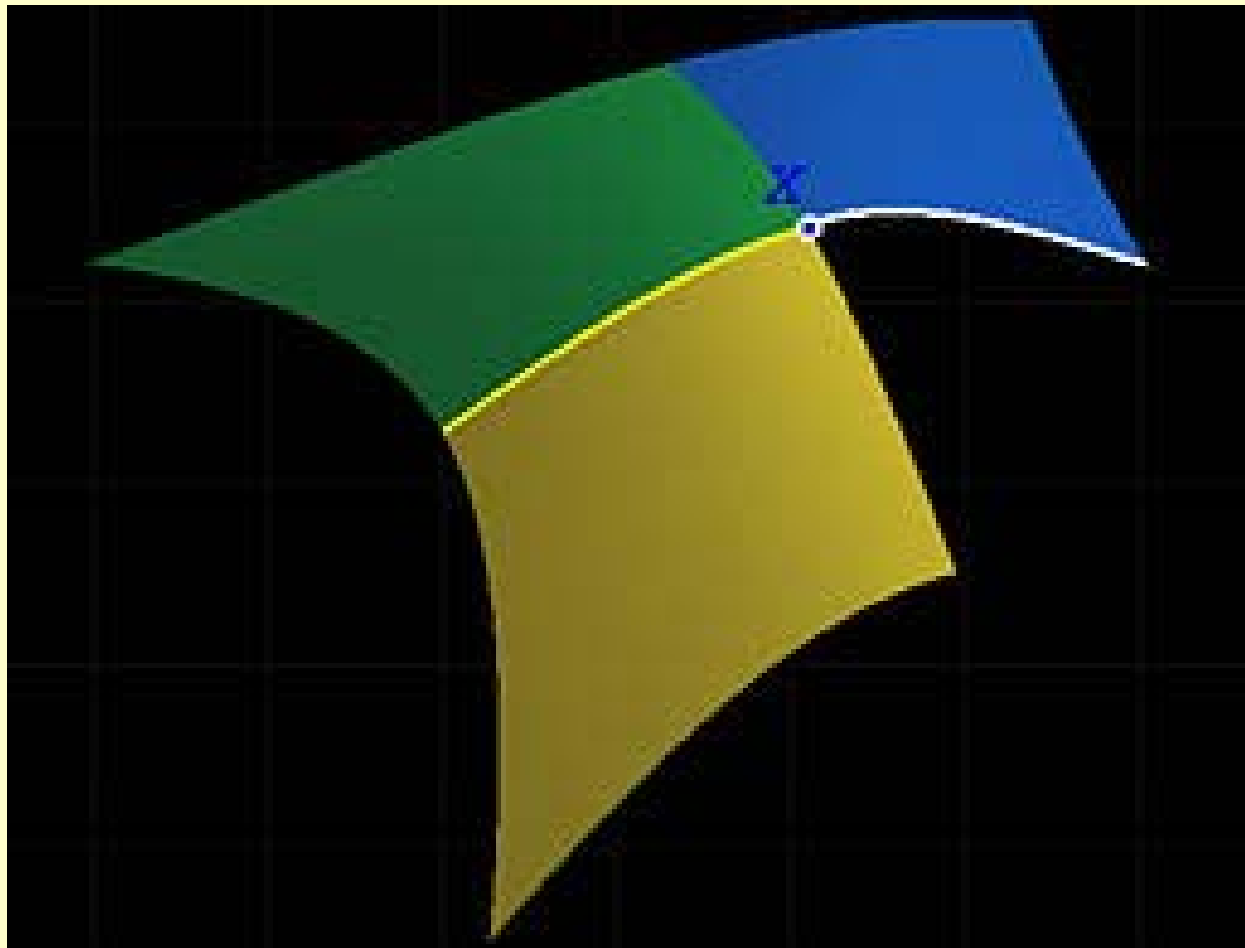


(b)

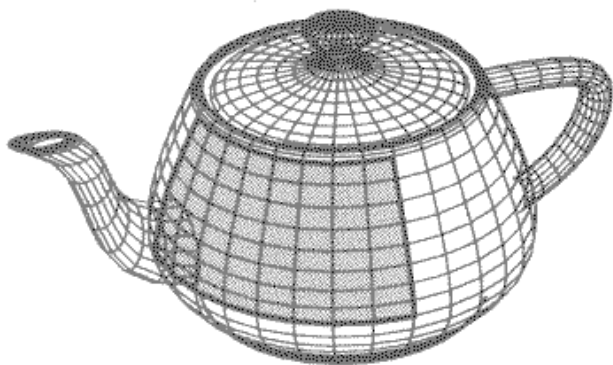
$$Q(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 p_{ij} b_i(u) b_j(v)$$



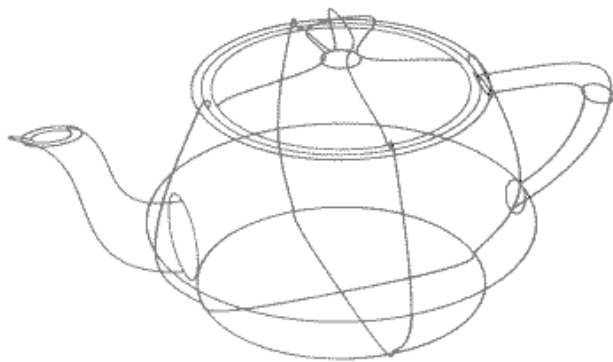
Непрерывность



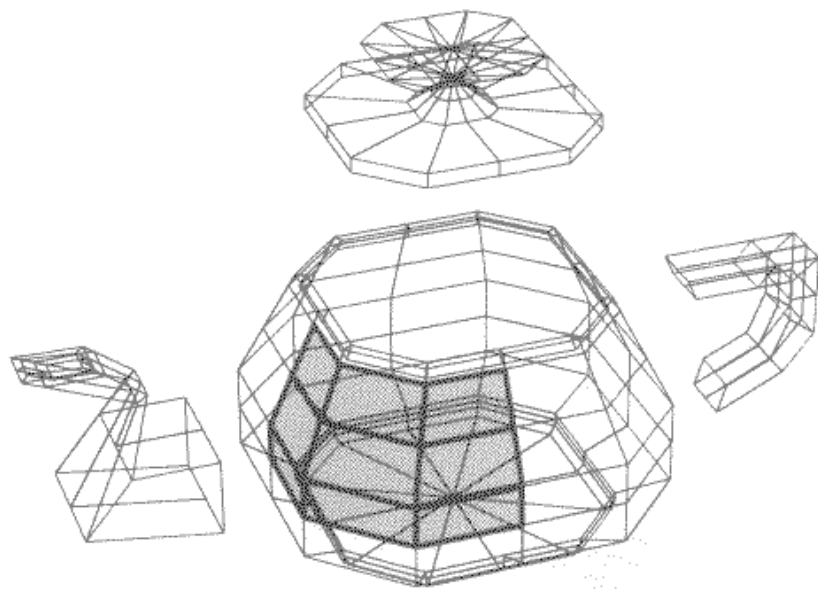
Чайник Юта (1)



(a)

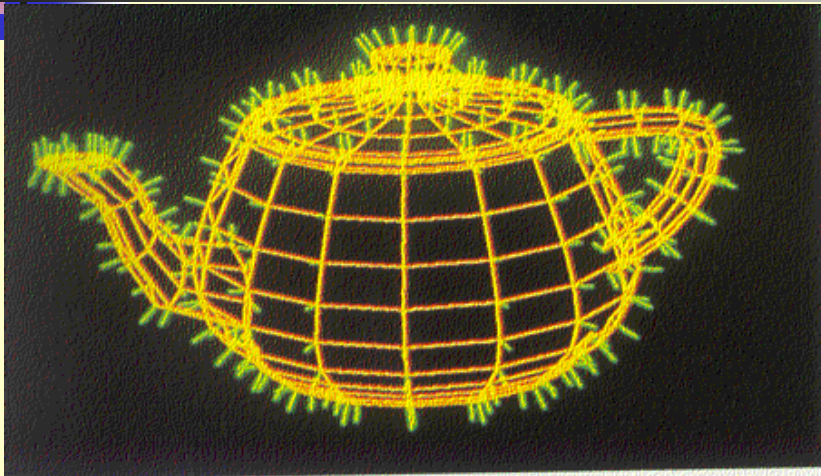


(b)



(c)

Чайник Юта (2)





Литература

- Дональд Херн, М. Паулин Бейкер
«Компьютерная графика и стандарт OpenGL» М.: Вильямс. 2005
- D.Hearn, M.P.Baker *«Computer Graphics. C Version»*
- Д. Роджерс, *«Математические основы машинной графики»*, Москва, Мир, 2001
- Л. Шапиро, Дж. Стокман,
«Компьютерное зрение», М.: БИНОМ.
Лаборатория знаний, 2006