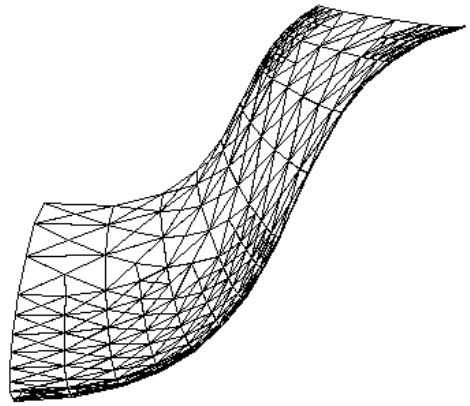
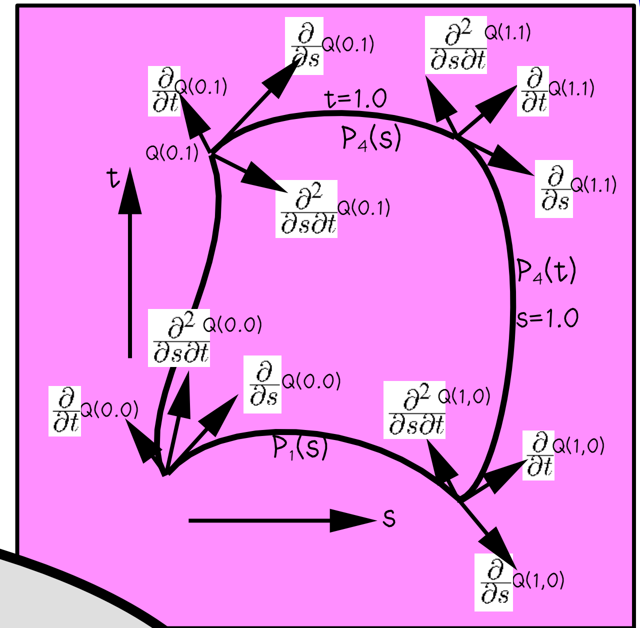
Computer Graphics



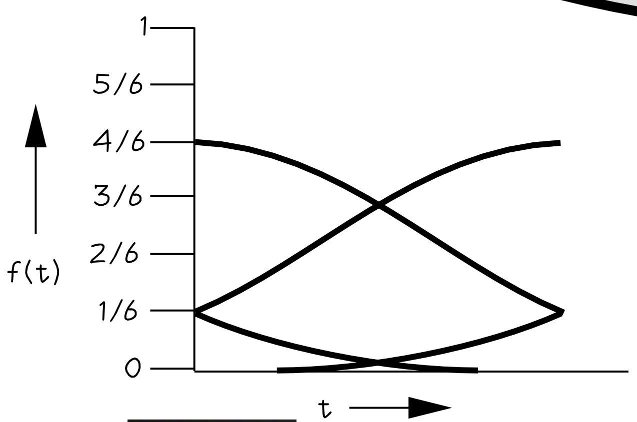


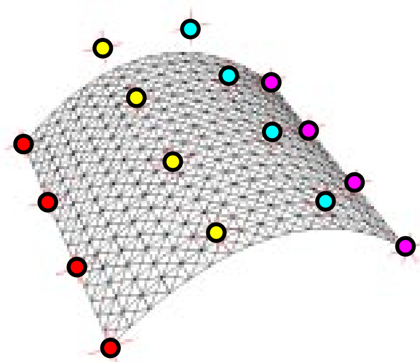

# More Curves & Surfaces

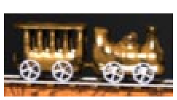

by

Brian Wyvill

University of Victoria



# Uniform Nonrational B-Splines

Shirley p334

continued

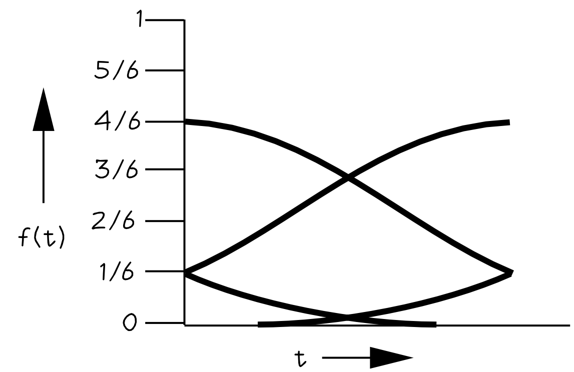
$$QG_{Bi} = \begin{vmatrix} P_{i-3} \\ P_{i-2} \\ P_{i-1} \\ P_i \end{vmatrix}$$

First segment $Q_3 = \begin{vmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{vmatrix}$ Parameter Range t3<=t<t4

(0<=t<1)

$$M_{BS} = 1/6 \begin{vmatrix} -1 & 3 & -3 & 0 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{vmatrix}$$

B-Spline Matrix

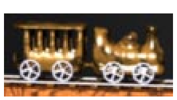



B-Spline Basis functions

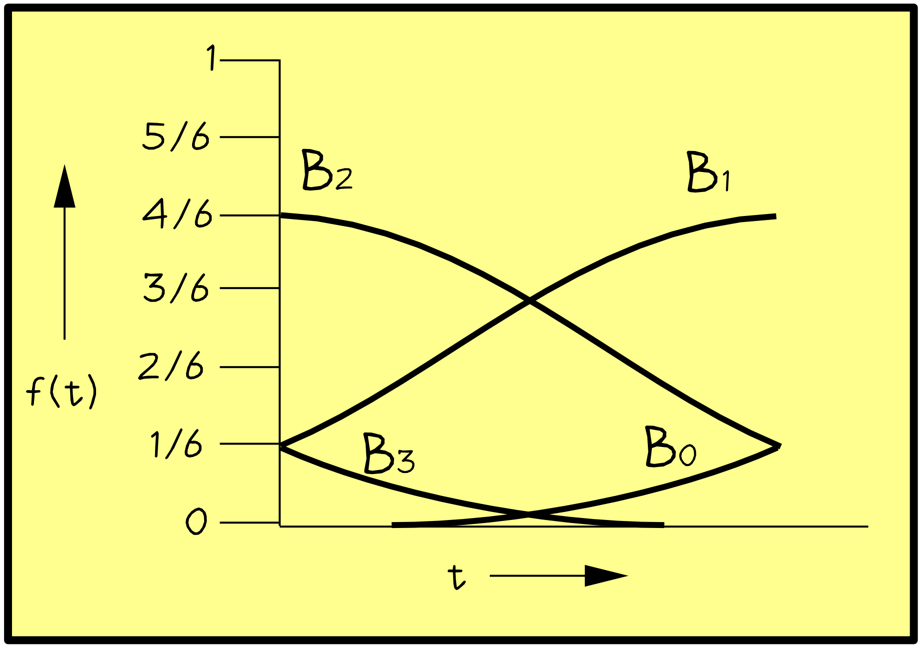
$$Q(t) = T\, M_{BS} G_{BS}$$

In the range (0<=t<1)

$$Q(t) = \frac{(1-t)^3}{6} P_{i-3} + \frac{(3t^3-6t^2+4)}{6} P_{i-2} + \frac{(-3t^3+3t^2+3t+1)}{6} P_{i-1} + \frac{t^3}{6} P_i$$



# B-Spline Basis functions




Four basis functions sum to 1 and are non-negative, convex hull holds.

$G^0$ $G^1$ and $G^2$ continuous.

To show continuity work with x components:

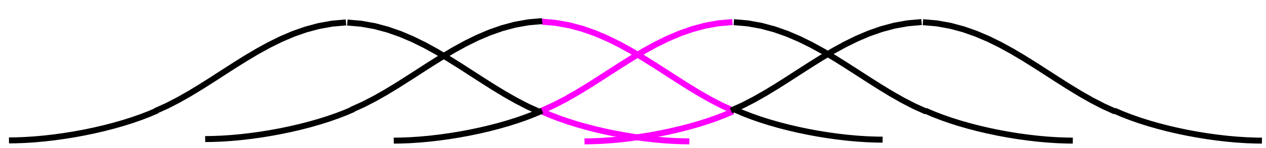
$$x_i \big|_{t=1} = x_{i+1} \big|_{t=0} = 1/6(P_{i-2} + 4P_{i-1} + P_i)$$

$$x'_i \big|_{t=1} = x'_{i+1} \big|_{t=0} = 1/2(-P_{i-2} + P_i)$$

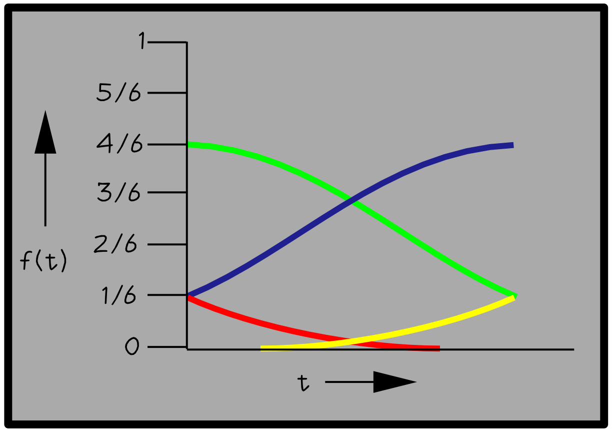
$$x''_i \big|_{t=1} = x''_{i+1} \big|_{t=0} = P_{i-2} - 2P_{i-1} + P_i$$

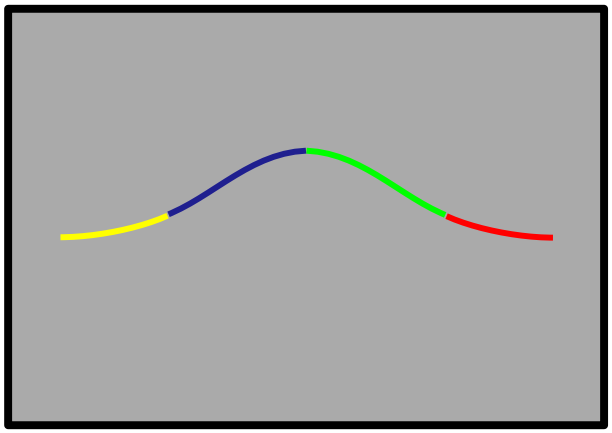
In the range (0<=t<1)

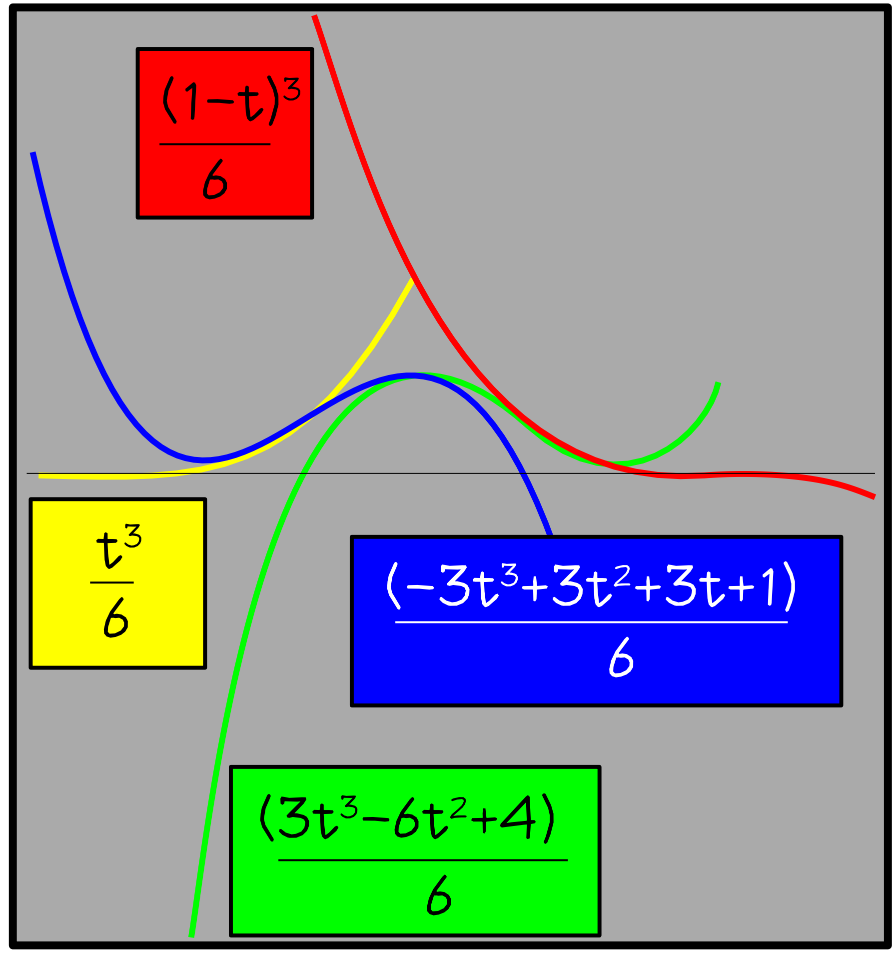
$$Q(t) = \frac{(1-t)^3}{6}P_{i-3} + \frac{(3t^3-6t^2+4)}{6}P_{i-2} + \frac{(-3t^3+3t^2+3t+1)}{6}P_{i-1} + \frac{t^3}{6}P_i$$



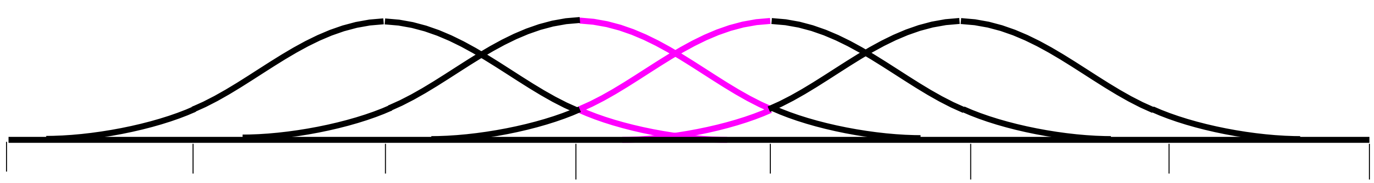
# B-Spline Basis functions



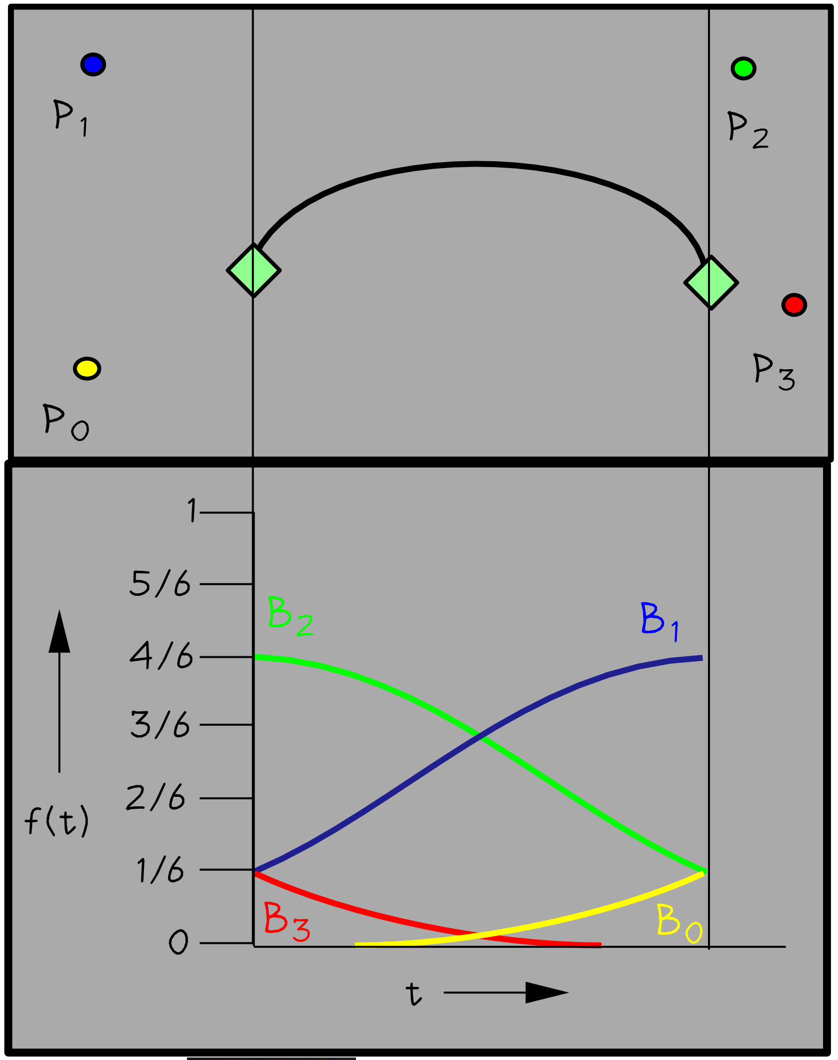



B-Spline basis functions in the range -2 to +2



In the range (0<=t<1)

$$Q(t) = \frac{(1-t)^3}{6}P_{i-3} + \frac{(3t^3-6t^2+4)}{6}P_{i-2} + \frac{(-3t^3+3t^2+3t+1)}{6}P_{i-1} + \frac{t^3}{6}P_i$$




$$Q(t) = B_0P_{i-3} + B_1P_{i-2} + B_2P_{i-1} + B_3P_i$$

$B_0$

$$\frac{t^3}{6}$$

$B_1$

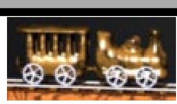
$$\frac{(-3t^3+3t^2+3t+1)}{6}$$

$$\frac{(3t^3-6t^2+4)}{6}$$

$B_2$

$$\frac{(1-t)^3}{6}$$

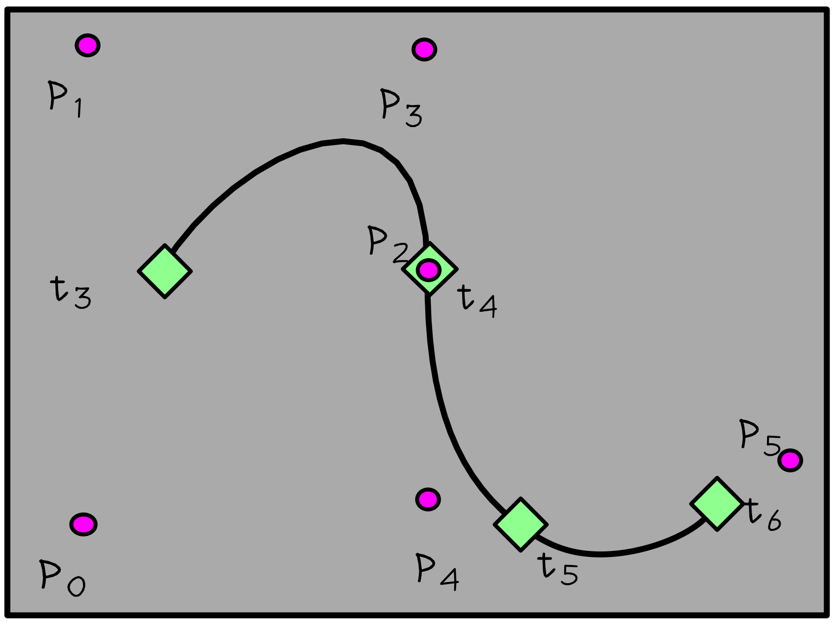
$B_3$



# Properties of these curves

These cubic curves are linear combinations of the four elements of the geometry vector. The curves can be transformed by transforming the geometry vector. The curves are invariant under affine transformations (scaling, rotation and translation).

B-Splines are $G^2$ continuous at the cost of loss of control. Control points can be replicated to force curve to pass through points:




Vertically aligned vertices constrain the curve to pass through one of them.

Coincident control vertices can also be used:

e.g. $P_{i-2}=P_{i-1}=P_i$ **resulting in:**

$$Q(i)=B_3P_{i-3} + (B_2+ B_1 +B_0)P_i$$

A straight line.

# NUBS and NURBS

## NUBS – Non-uniform non-rational B-Splines.

The parameter interval between successive knot values need not be uniform. The blending functions are no longer the same for each knot interval. Continuity can be reduced from $C^2$ to $C^1$ to $C^0$ to none. The curve can be made to interpolate a control point without introducing linear segments. Successive knot values can be equal, these coincident knots cause the curve segement to reduce to a point.

## NURBS – Non-uniform Rational B-Splines.

Rational cubic curve segments are ratios of polynomials:

$$x(t) = \frac{X(t)}{W(t)} \qquad y(t) = \frac{Y(t)}{W(t)} \qquad z(t) = \frac{Z(t)}{W(t)}$$

$$Q(t) = [X(t) \quad Y(t) \quad Z(t) \quad W(t)]$$

Each of X(), Y(), Z(), W() are cubic polynomial curves defined in homogeneous coordinates. This is useful as they are invariant under perspective transformations as well as the affine transformations. Transformations need only be applied to control points only.

# Parametric Bicubic Surfaces

As with curves only two parameters.
Geometry vector now becomes functions of t:

$$Q(s,t)=S \cdot M \cdot G(t)=S \cdot M \cdot \begin{vmatrix} G_1(t) \\ G_2(t) \\ G_3(t) \\ G_4(t) \end{vmatrix}$$

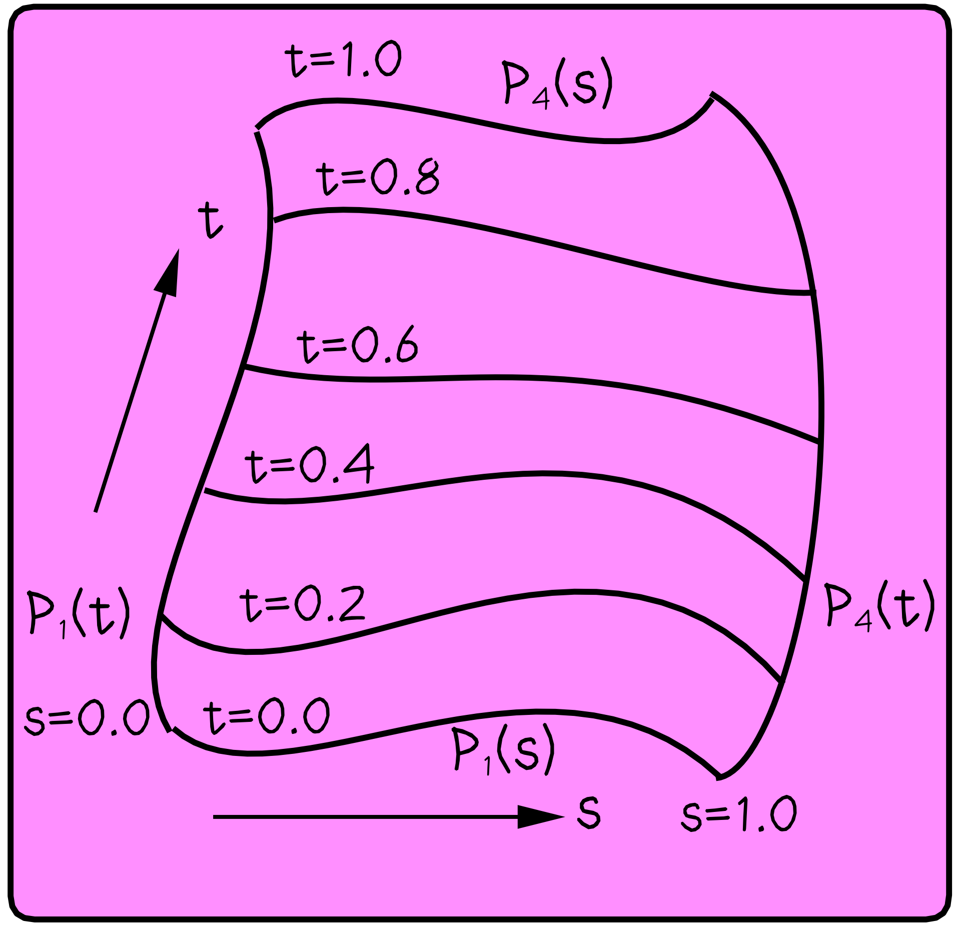
The $G_i(t)$ are themselves cubics.
Each can be represented as

$$G_i(t)=T \cdot M \cdot g_i \text{ where } g_i=\begin{vmatrix} g_{i1} \\ g_{i2} \\ g_{i3} \\ g_{i4} \end{vmatrix}$$

We want the $g_i$ as a row vector so we can substitute into the equation for Q(s,t).

Since $(A.B.C)^T=C^T.B^T.A^T$

$$G_i(t) = g_i^T \cdot M^T \cdot T^T = [g_{i1}\ g_{i2}\ g_{i3}\ g_{i4}] \cdot M^T \cdot T^T$$



# Hermite Surfaces

$G_i(t) = g_i^T \cdot M^T \cdot T^T = [g_{i1}\ g_{i2}\ g_{i3}\ g_{i4}] \cdot M^T \cdot T^T$

therefore $Q(s,t) = S \cdot M \begin{vmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \\ g_{41} & g_{42} & g_{43} & g_{44} \end{vmatrix} \cdot M^T \cdot T^T$

or $Q(s,t) = S \cdot M \cdot G \cdot M^T \cdot T^T \quad (0 <= t <= 1) \quad (0 <= s <= 1)$

thus an x,y,z point can be found for a given value of s and t.

If M is $M_H$ then we have a Hermite surface:

e.g. $x(s,t) = S \cdot M_H \begin{vmatrix} P_1(t) \\ P_4(t) \\ R_1(t) \\ R_4(t) \end{vmatrix}$

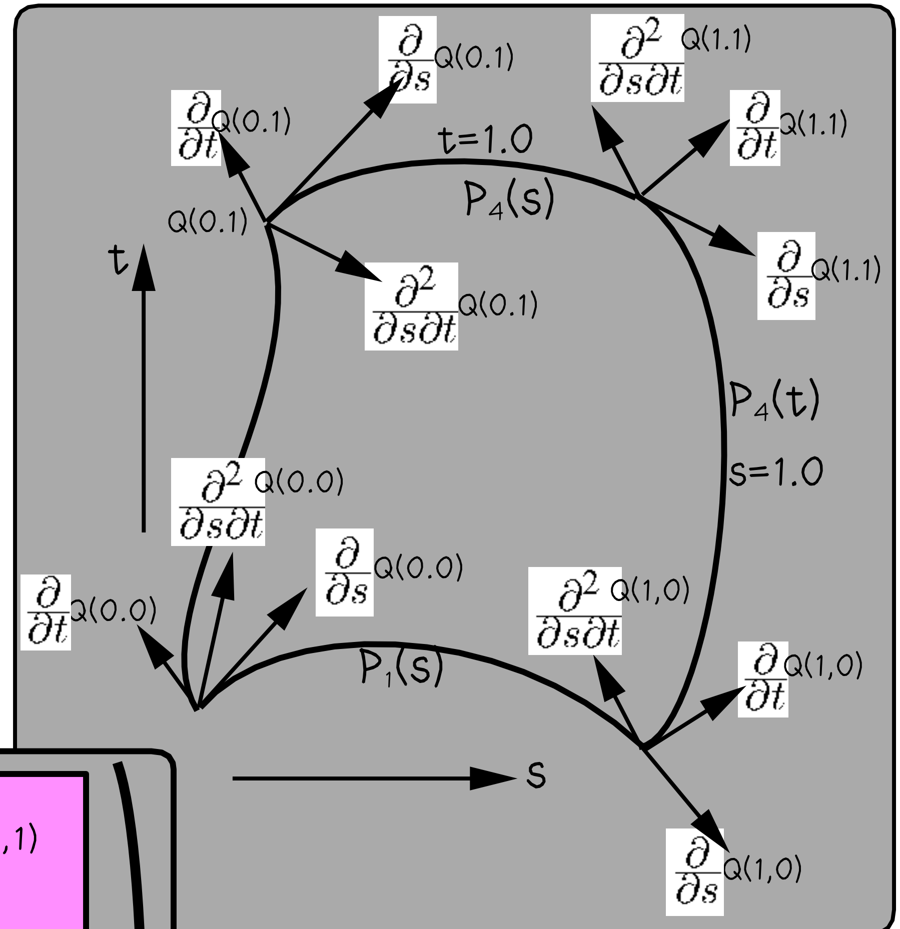
where:

$P_1(t) = T \cdot M_H \begin{vmatrix} g_{11} \\ g_{12} \\ g_{13} \\ g_{14} \end{vmatrix}$ $P_4(t) = T \cdot M_H \begin{vmatrix} g_{12} \\ g_{22} \\ g_{23} \\ g_{24} \end{vmatrix}$ $R_1(t) = T \cdot M_H \begin{vmatrix} g_{31} \\ g_{32} \\ g_{33} \\ g_{34} \end{vmatrix}$ $R_4(t) = T \cdot M_H \begin{vmatrix} g_{41} \\ g_{42} \\ g_{43} \\ g_{44} \end{vmatrix}$

# Hermite Surfaces

continued

$$Q(s,t) = S \cdot M \begin{vmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \\ g_{41} & g_{42} & g_{43} & g_{44} \end{vmatrix} \cdot M^T \cdot T^T$$




The 'g' matrix for the hermite surface is :

$$G_{H_x} = \begin{pmatrix} x(0,0) & x(0,1) & \frac{\partial}{\partial t}x(0,0) & \frac{\partial}{\partial t}x(0,1) \\ x(1,0) & x(1,\ 1) & \frac{\partial}{\partial t}x(1,0) & \frac{\partial}{\partial t}x(1,1) \\ \frac{\partial}{\partial s}x(0,0) & \frac{\partial}{\partial s}x(0,1) & \frac{\partial^2}{\partial s \partial t}x(0,0) & \frac{\partial^2}{\partial s \partial t}x(0,1) \\ \frac{\partial}{\partial s}x(1,0) & \frac{\partial}{\partial s}x(1,\ 1) & \frac{\partial^2}{\partial s \partial t}x(1,0) & \frac{\partial^2}{\partial s \partial t}x(1,1) \end{pmatrix}$$

# Continuity between surface patches

Patch 1

$$\begin{vmatrix} - & - & - & - \\ g_{21} & g_{22} & g_{23} & g_{24} \\ - & - & - & - \\ g_{41} & g_{42} & g_{43} & g_{44} \end{vmatrix}$$

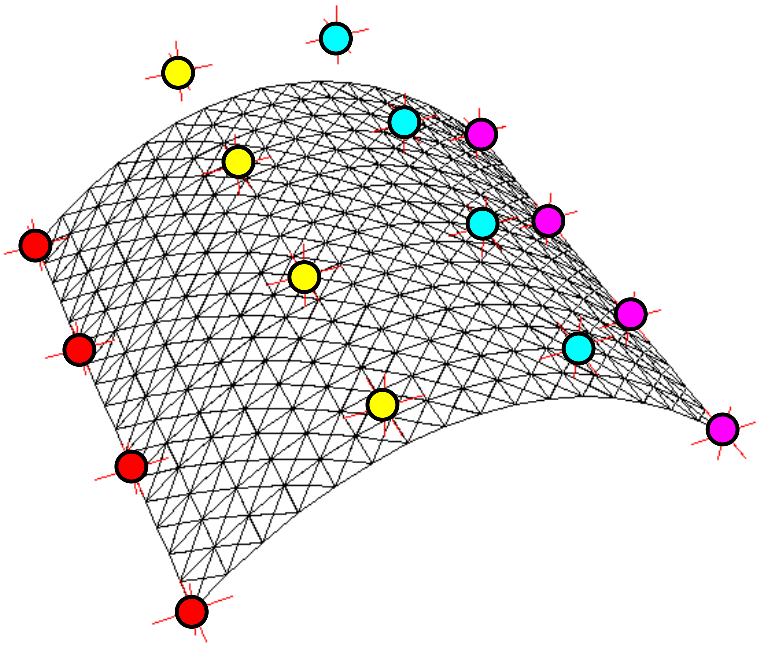
Patch 2

$$\begin{vmatrix} g_{21} & g_{22} & g_{23} & g_{24} \\ - & - & - & - \\ kg_{41} & kg_{42} & kg_{43} & kg_{44} \\ - & - & - & - \end{vmatrix} \quad k>0$$

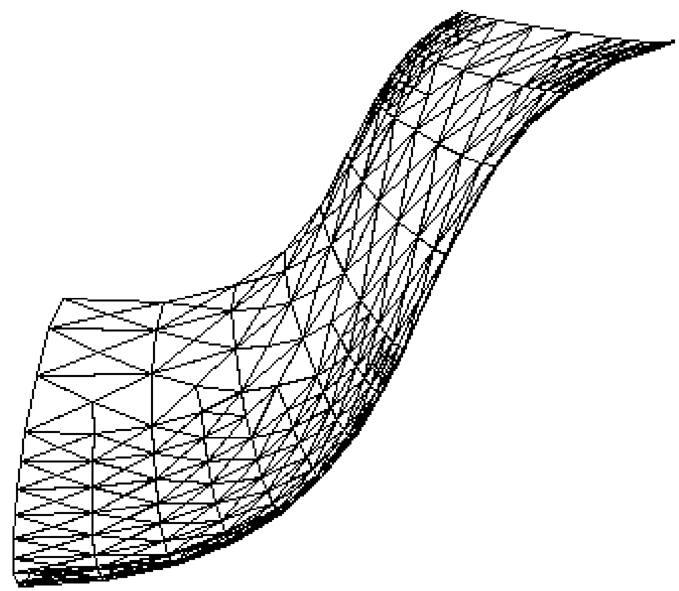
where:

$$P_1(t)=T \cdot M_H \begin{vmatrix} g_{11} \\ g_{12} \\ g_{13} \\ g_{14} \end{vmatrix} \qquad P_4(t)=T \cdot M_H \begin{vmatrix} g_{12} \\ g_{22} \\ g_{23} \\ g_{24} \end{vmatrix} \qquad R_1(t)=T \cdot M_H \begin{vmatrix} g_{31} \\ g_{32} \\ g_{33} \\ g_{34} \end{vmatrix} \qquad R_4(t)=T \cdot M_H \begin{vmatrix} g_{41} \\ g_{42} \\ g_{43} \\ g_{44} \end{vmatrix}$$

# Bezier patches



Uniformly polygonised patch
and Bezier control points



Polygonal Spout
Adaptive Subdivision algorithm at work on the spout

# Drawing Cubics

Straightforward implementation
by Horner's Rule
e.g.

```
void x(double t)
{
     return t*(t*(t*ax+bx)+cx)+dx;
}
```

5 multiplies 3 additions

**Repeated Evaluation of Cubic by Forward Differences**

Definition : $\Delta f(t) = f(t+\delta) - f(t) \quad \delta > 0$

rewriting: $f(t+\delta) = \Delta f(t) + f(t)$

Rewriting iteratively: $f_{n+1} = f_n + \Delta f_n$

$f(t) = at^3 + bt^2 + ct + d$

$\Delta f(t) = a(t+\delta)^3 + b(t+\delta)^2 + c(t+\delta) + d - (at^3 + bt^2 + ct + d)$

$\Delta f(t) = 3at^2\delta + t(3a\delta^2 + 2b\delta) + a\delta^3 + b\delta^2 + c\delta$ --(1)

So $\Delta f(t)$ is second degree. Applying forward differences again to reduce this further:

$\Delta^2 f(t) = \Delta(\Delta f(t)) = \Delta f(t+\delta) - \Delta f(t)$

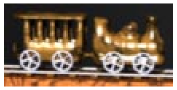
applying by writing $(t+\delta)$ for t in (1)
$\Delta f(t)$ or $(3at^2\delta + t(3a\delta^2 + 2b\delta) + a\delta^3 + b\delta^2 + c\delta)$

Yields $\Delta^2 f(t) = 6a\delta^2 t + 6a\delta^3 + 2b\delta^2$ --(2)

$\Delta^3 f(t) = \Delta(\Delta^2 f(t)) = \Delta^2 f(t+\delta) - \Delta^2 f(t)$

substituting $(t+\delta)$ for t in (2)

$\Delta^3 f(t) = 6a\delta^3$ --(3)



# Forward Differences

By Definition : $\Delta f(t)=f(t+\delta)-f(t)$

rewriting: $f(t+\delta)=f(t)+\Delta f(t)$

$$f_{n+1} = f_n + \Delta f_n$$

In other words $t_n = n\delta$ and $f_n=f(t_n)$

f is evaluated at equal intervals of size $\delta$.

At t=0

$$f_0 = d$$
$$\Delta f_0 = a\delta^3+b\delta^2+c\delta$$
$$\Delta^2 f_0 = 6a\delta^3 +2b\delta^2$$
$$\Delta^3 f_0 = 6a\delta^3$$

by definition: $\Delta^2 f(t) = \Delta f(t+\delta)-\Delta f(t)$

or $\Delta^2 f_n = \Delta f_{n+1} - \Delta f_n$

$$\Delta f_{n+1} = \Delta f_n + \Delta^2 f_n$$

similarly

$$\Delta^3 f(t) = \Delta^2 f(t+\delta)-\Delta^2 f(t)$$

$$\Delta^2 f_{n+1} = \Delta^2 f_n + \Delta^3 f_n$$

and $\Delta^3 f_n$ is constant

Repeat following steps m times with n initially 0:-

$$f_{n+1} = f_n + \Delta f_n$$
$$\Delta f_{n+1} = \Delta f_n + \Delta^2 f_n$$
$$\Delta^2 f_{n+1} = \Delta^2 f_n + \Delta^3 f_0$$

n=0

$$f_1 = f_0 + \Delta f_0$$
$$\Delta f_1 = \Delta f_0 + \Delta^2 f_0$$
$$\Delta^2 f_1 = \Delta^2 f_0 + \Delta^3 f_0$$

n=1

$$f_2 = f_1 + \Delta f_1$$
$$\Delta f_2 = \Delta f_1 + \Delta^2 f_1$$
$$\Delta^2 f_2 = \Delta^2 f_1 + \Delta^3 f_0$$

Suppose we want 64 steps

n=64 or $\delta=1/64$

$t_0 = 0\delta = 0$

$t_1 = 1\delta = 1/64$

$t_2 = 2\delta = 2/64$ etc.

# Forward Differences

continued

$$f_0 = d$$
$$\Delta f_0 = a\delta^3 + b\delta^2 + c\delta$$
$$\Delta^2 f_0 = 6a\delta^3 + 2b\delta^2$$
$$\Delta^3 f_0 = 6a\delta^3$$

$$\begin{bmatrix} f_0 \\ \Delta f_0 \\ \Delta^2 f_0 \\ \Delta^3 f_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

$$D_x = E(\delta)C_x$$
$$D_y = E(\delta)C_y$$
$$D_z = E(\delta)C_z$$

$f_0$
$\Delta f_0$
$\Delta^2 f_0$
$\Delta^3 f_0$

$f_x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$
similarly for y and z

so we have:

$$\begin{bmatrix} f_0 \\ \Delta f_0 \\ \Delta^2 f_0 \\ \Delta^3 f_0 \end{bmatrix}$$

x and y and z

and can calculate

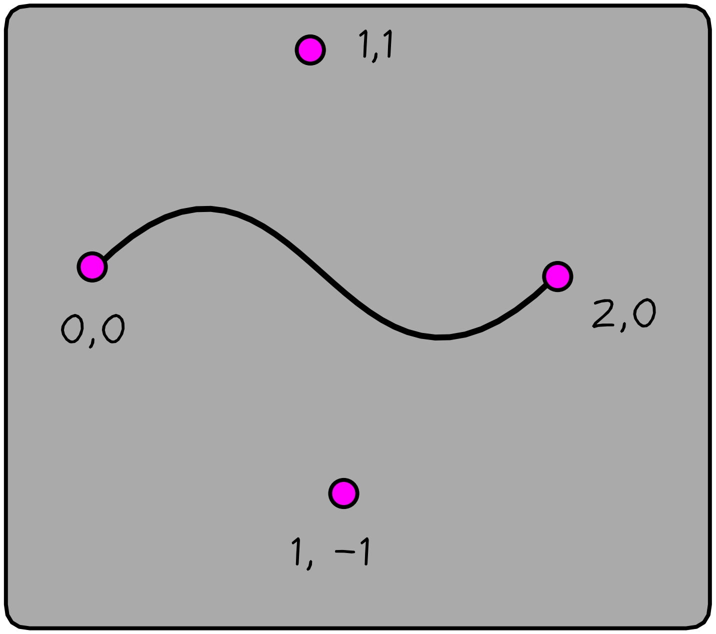
$$\begin{bmatrix} f_n \\ \Delta f_n \\ \Delta^2 f_n \\ \Delta^3 f_n \end{bmatrix}$$

x and y and z

algorithm writing $\Delta x$ for $\Delta f_{nx}$ etc.

```
for (i=0; i<n; i++) {
    x+=Δx;     Δx+=Δ²x;     Δ²x+=Δ³x;
    y+=Δy;     Δy+=Δ²y;     Δ²y+=Δ³y;
    z+=Δz;     Δz+=Δ²z;     Δ²z+=Δ³z;
    lineAbs(x,y,z);
}
```

# Example




Bezier Curve Segment

For $Qx(t) = T\ Mb\ Gbx$

$Cx = Mb\ Gbx$

$$Gbx = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix} \qquad Mb = \begin{bmatrix} 1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$Qx(t) = axt^3+bxt^2+cxt+dx$

$ax = 2$

$bx = -3$

$cx = 3$

$dx = 0$

we have:

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

taking 100 steps $\delta=0.01$

At $t=0$

$$f_0 = d = 0$$

$$\Delta f_0 = a\delta^3+b\delta^2+c\delta = 0.029702$$

$$\Delta^2 f_0 = 6a\delta^3+2b\delta^2 = -0.000588$$

$$\Delta^3 f_0 = 6a\delta^3 = 0.000012$$

# Example

continued

$f_0 = 0$

$\Delta f_0 = 0.029702$

$\Delta^2 f_0 = -0.000588$

$\Delta^3 f_0 = 0.000012$

```
for (i=0; i<n; i++) {
    x+=Δx;   Δx+=Δ²x;   Δ²x+=Δ³x;
    y+=Δy;   Δy+=Δ²y;   Δ²y+=Δ³y;
    z+=Δz;   Δz+=Δ²z;   Δ²z+=Δ³z;
    lineAbs(x,y,z);
}
```

| $x$ | $\Delta x$ | $\Delta^2 x$ | $\Delta^3 x$ |
|---|---|---|---|
| 0 | 0.029702 | 0.029114 | 0.000012 |
| 0.029702 | 0.029114 | −0.000576 | 0.000012 |
| 0.058816 | 0.028538 | −0.000564 | 0.000012 |
| 0.087354 | 0.027974 | −0.000552 | 0.000012 |
| 0.115328 | 0.027422 | −0.000540 | 0.000012 |
| .etc. until | ........................................ | | |
| 1.912645 | 0.028538 | 0.000576 | 0.000012 |
| 1.941183 | 0.029114 | 0.000588 | 0.000012 |
| 1.970297 | 0.029702 | 0.000600 | 0.000012 |
| 1.999999 | 0.030302 | 0.000612 | 0.000012 |