# Maths for Computer Graphics

## Interpolation

Interpolation is not a branch of mathematics but rather a collection of techniques.

An interpolant is a way of changing one number into another.

E.g. to change 2 into 4 we add 2.

The real function of an interpolant is to change one number into another in a number of equal or unequal steps.

E.g. If we started with 2 and repeatedly added 0.2, it would generate the sequence 2.2, 2.4, 2.6, 2.8, 3.0, 3.2, 3.4, 3.6, 3.8, and 4.

These numbers could then be used to translate, scale, rotate an object, move a virtual camera, or change the position, color or brightness of a virtual light source.

# Linear interpolant

A *linear interpolant* generates equal spacing between the interpolated values for equal changes in the interpolating parameter.

Given two numbers $n_1$ and $n_2$, which represent the start and final values of the interpolant, we require an interpolated value controlled by a parameter $t$ that varies between 0 and 1.

When $t = 0$, the result is $n_1$, and when $t = 1$, the result is $n_2$.

A solution to this problem is given by

$$n = n_1 + t(n_2 - n_1)$$

for when $n_1 = 2$, $n_2 = 4$ and $t = 0.5$

$$n = 2 + \tfrac{1}{2}(4 - 2) = 3$$

which is a half-way point.

# Maths for Computer Graphics

## Linear interpolation

It can be expressed differently:

$$n = n_1 + t(n_2 - n_1)$$
$$n = n_1 + tn_2 - tn_1$$
$$n = n_1(1 - t) + n_2 t$$

This is also called a *lerp*

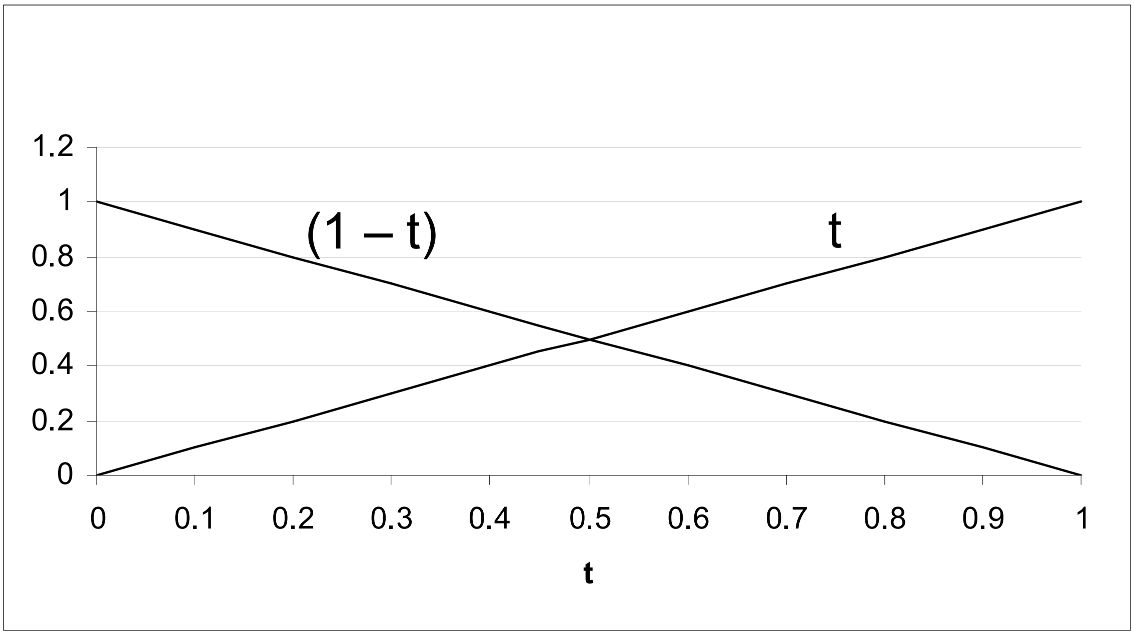



***Fig. 8.1*** *The graphs of (1-t) and t over the range 0 to 1.*

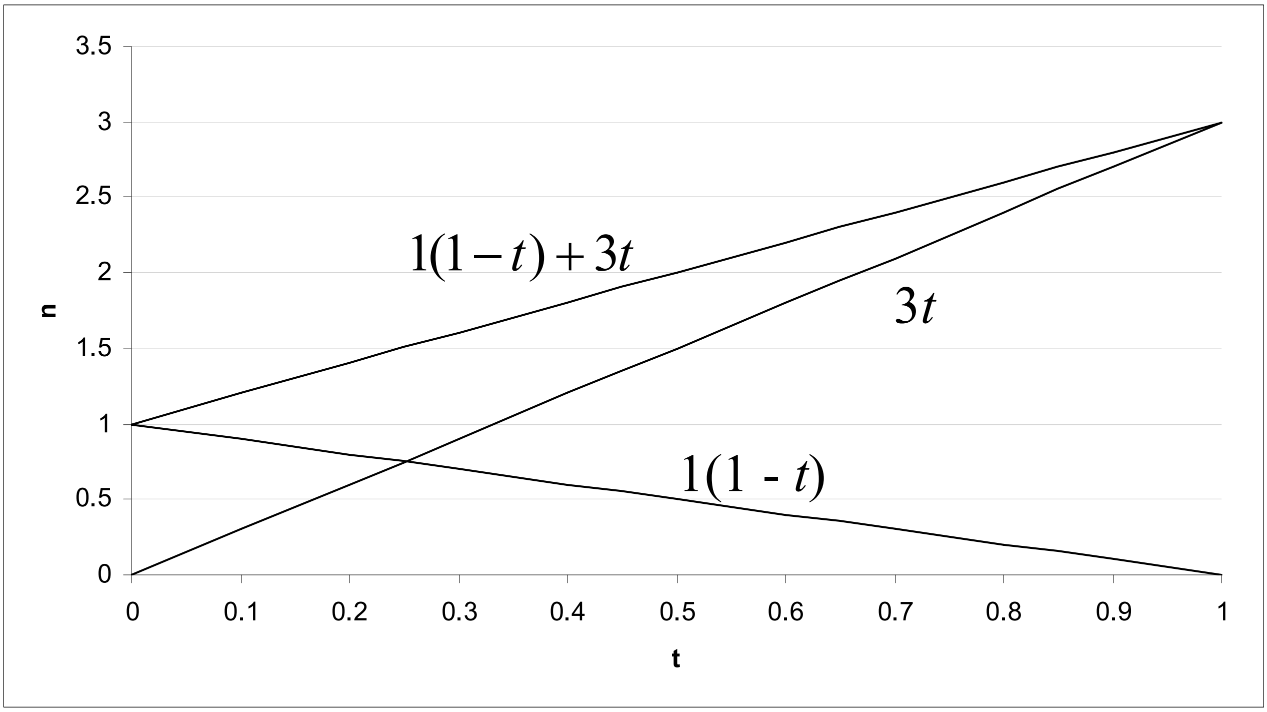
# Linear interpolation

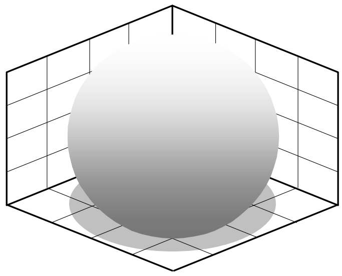
Interpolating between 1 and 3 using

$$n = n_1(1-t) + n_2 t$$

where $n_1 = 1$ and $n_2 = 3$

i.e. $n = 1(1-t) + 3t \qquad 0 \le t \le 1$





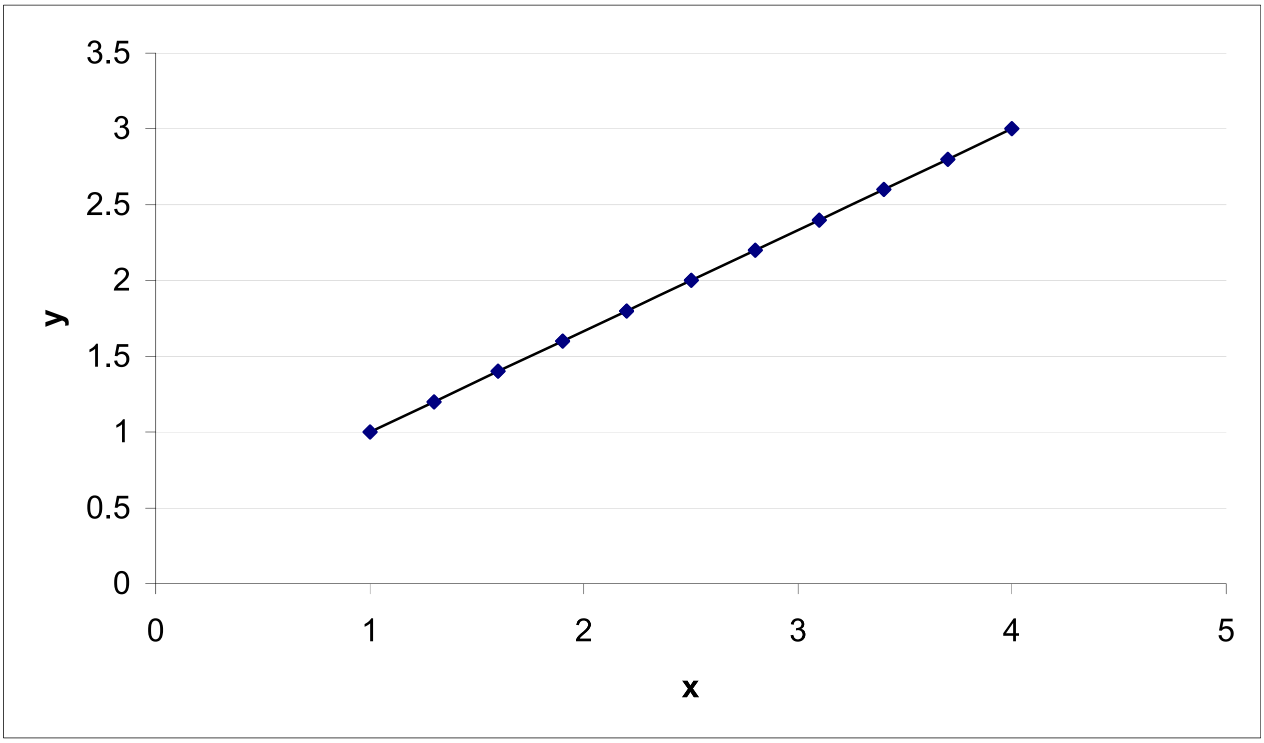
# Linear interpolating coordinates

Although the *lerp* is extremely simple, it is widely used in computer graphics software.

Consider interpolating between (1, 1) and (4, 3).

The interpolated coordinate $(x, y)$ is given by

$$x = 1(1-t) + 4t$$
$$y = 1(1-t) + 3t$$
$$\text{for } 0 \le t \le 1$$



# Matrix notation

$$n = (1 - t)n_1 + tn_2$$

$$n = \begin{bmatrix} (1 - t) & t \end{bmatrix} \cdot \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

$$n = \begin{bmatrix} t & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

## Non-linear interpolation

A linear interpolant ensures that equal steps in the parameter *t* give rise to equal steps in the interpolated values.

It is often required that equal steps in *t* give rise to unequal steps in the interpolated values.

We can achieve this using a variety of mathematical techniques.

For example, we could use trigonometric functions or polynomials to achieve this.

# Maths for Computer Graphics

## Trigonometric interpolation

$$\sin^2(\beta) + \cos^2(\beta) = 1$$

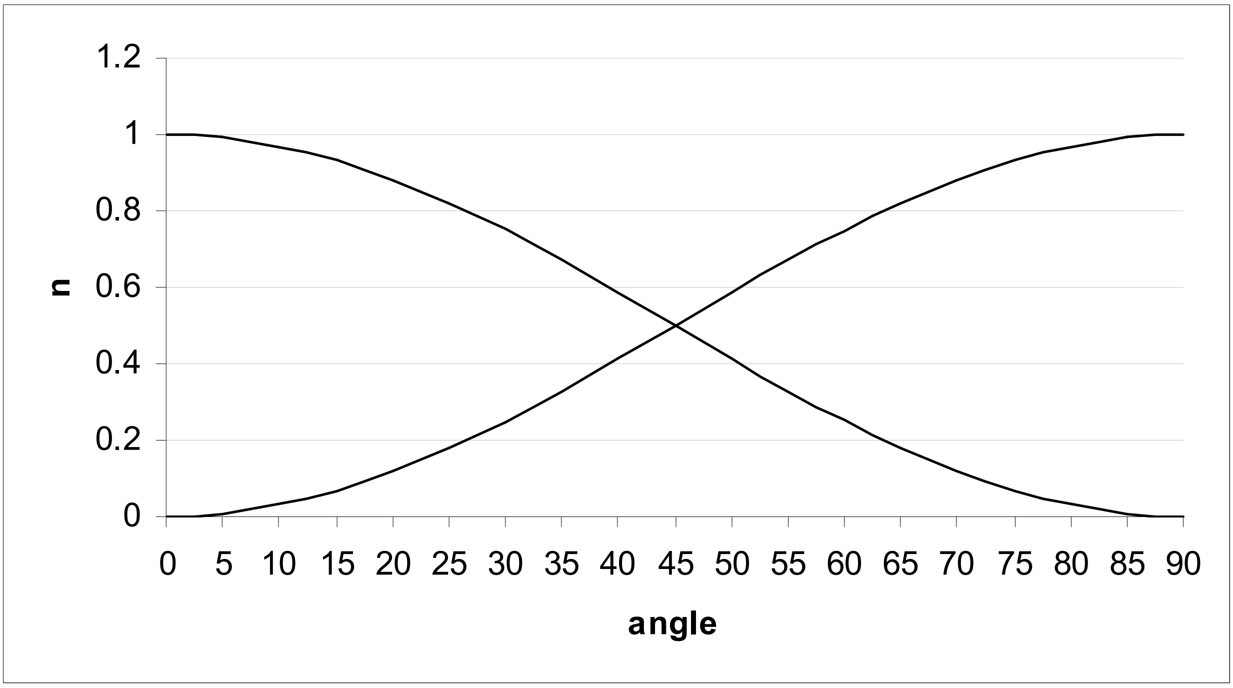
If $\beta$ varies between 0 and π/2

$\cos^2(\beta)$ varies between 1 and 0
$\sin^2(\beta)$ varies between 0 and 1

which can be used to modify the two interpolated values $n_1$ and $n_2$ as follows

$$n = n_1 \cos^2(t) + n_2 \sin^2(t)$$

$$\text{for} \quad 0 \le t \le \pi / 2$$



Maths for Computer Graphics

# Trigonometric interpolation

$$n = n_1 \cos^2(t) + n_2 \sin^2(t)$$

Let $n_1 = 1$ and $n_2 = 3$

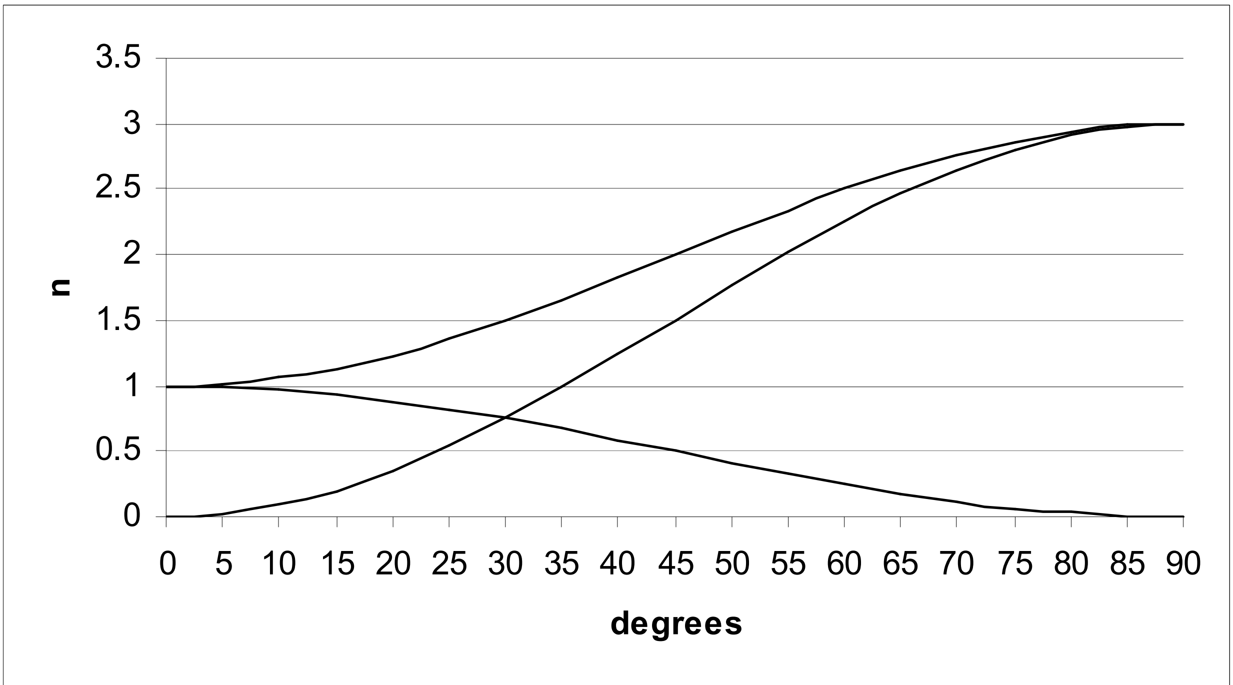



***Fig. 8.5** Interpolating between 1 and 3 using a trigonometric interpolant.*

# Maths for Computer Graphics

## Trigonometric interpolation

$$n = n_1 \cos^2(t) + n_2 \sin^2(t)$$

Interpolating between (1, 1) and (4, 3)

$$x = 1\cos^2(t) + 4\sin^2(t)$$
$$y = 1\cos^2(t) + 3\sin^2(t)$$

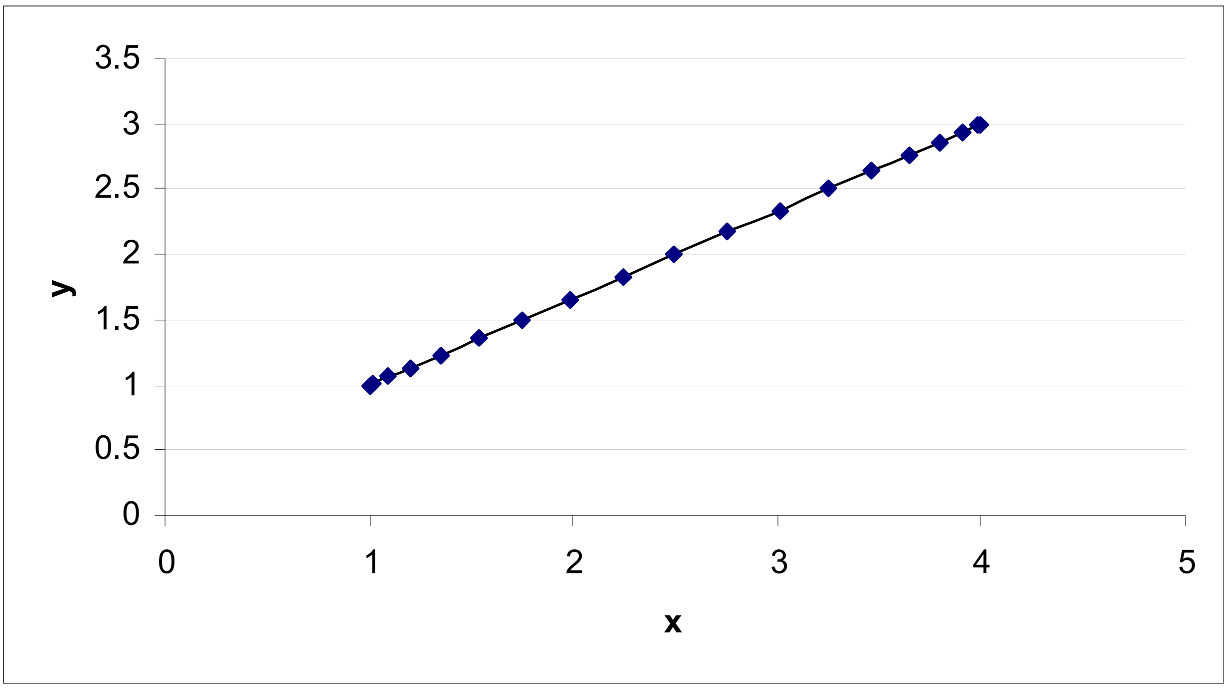



***Fig. 8.6*** *Interpolating between two points (1,1) and (4,3). Note the non-linear distribution of points*

# Maths for Computer Graphics

## Cubic interpolation

$$V_1 = 2t^3 - 3t^2 + 1$$

$$V_2 = -2t^3 + 3t^2$$

$$V_1 + V_2 = 1$$

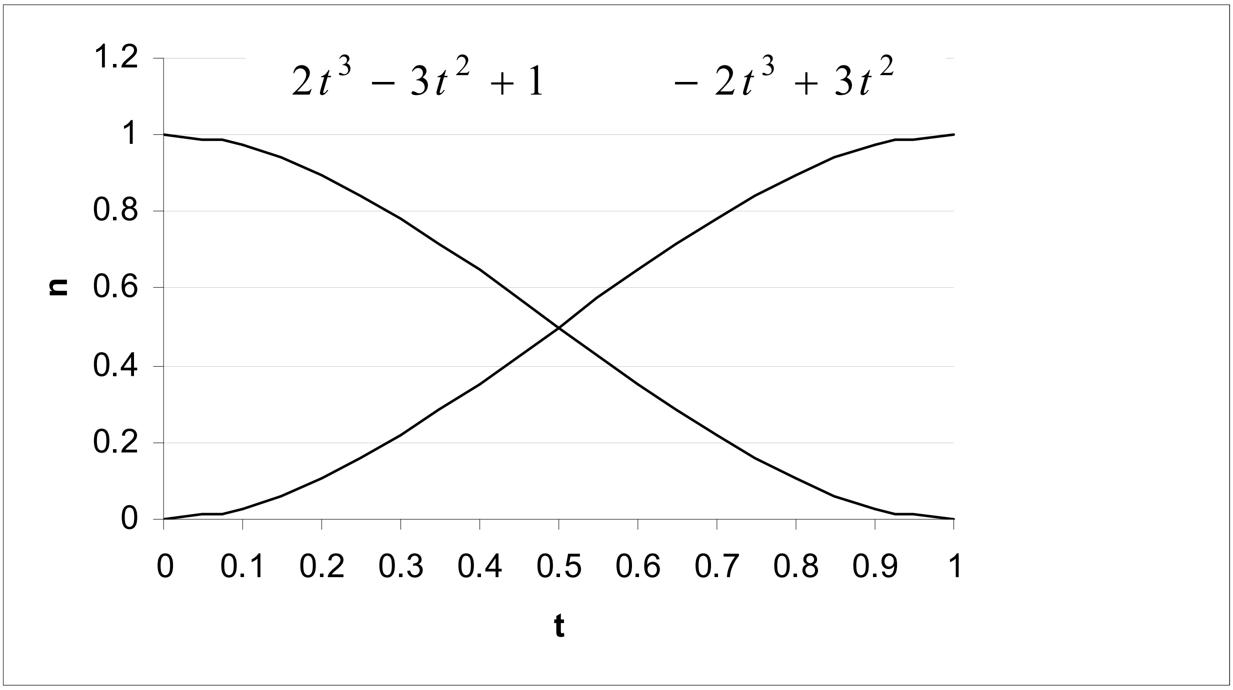



***Fig. 8.7*** *Two cubic interpolants.*

# Cubic interpolation

$$V_1 = 2t^3 - 3t^2 + 1$$

$$V_2 = -2t^3 + 3t^2$$

$$n = n_1 V_1 + n_2 V_2$$

$$n = \begin{bmatrix} 2t^3 - 3t^2 + 1 & -2t^3 + 3t^2 \end{bmatrix} . \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

$$n = \begin{bmatrix} t^3 & t^2 & t^1 & 1 \end{bmatrix} . \begin{bmatrix} 2 & -2 \\ -3 & 3 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} . \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

# Cubic interpolation

$$V_1 = 2t^3 - 3t^2 + 1$$

$$V_2 = -2t^3 + 3t^2$$

$$n = n_1 V_1 + n_2 V_2$$

Let $n_1 = 1$ and $n_2 = 3$

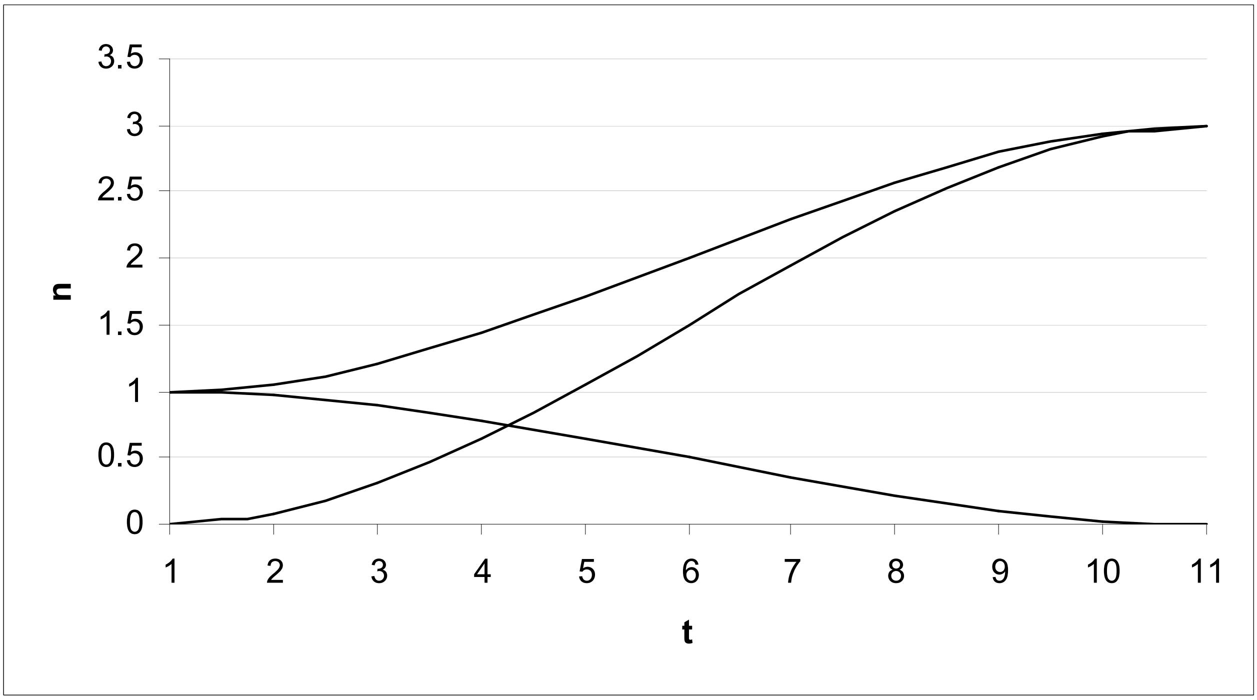



***Fig. 8.8*** *Interpolating between 1 and 3 using a cubic interpolant.*

# Maths for Computer Graphics

## Cubic interpolation of coordinates

$$V_1 = 2t^3 - 3t^2 + 1$$

$$V_2 = -2t^3 + 3t^2$$

$$x = x_1V_1 + x_2V_2$$
$$y = y_1V_1 + y_2V_2$$

Let $P_1 = (1, 1)$ and $P_2 = (4, 3)$

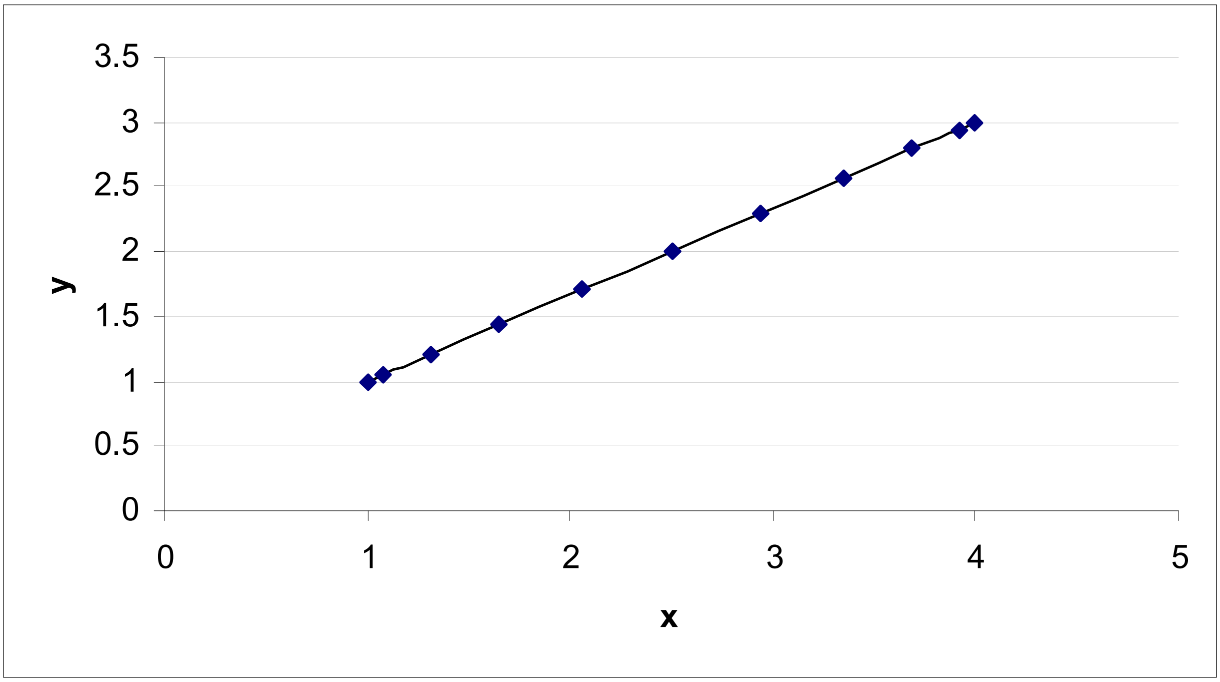



***Fig. 8.9*** *A cubic interpolant between points (1, 1) and (4, 3).*