APPENDIX CONTENTS

Appendix Contents

Appendix A1 Matrix multiplication

Two matrices can only be multiplied if they have the proper sizes. To multiply matrix "**A**" by matrix "**B**", A must have the same number of columns as B has rows. Order of multiplication is important since, in general, **A*****B** ≠ **B*****A**. Each single entry of the product matrix is a combination of an entire row from the first matrix and an entire column of the second matrix. Each entry results from a sum of products.

A is r by c **B** is s by d c = s **A*B** is r by d

As an example, we will use these two matrices:

$$\mathbf{A} = \begin{array}{c} \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{vmatrix} \begin{array}{l} \text{row 1} \\ \text{row 2} \end{array} \\ \text{column } 1 \quad 2 \quad 3 \end{array} \qquad \mathbf{B} = \begin{array}{c} \begin{vmatrix} a & d & g \\ b & e & h \\ c & f & i \end{vmatrix} \begin{array}{l} \text{row 1} \\ \text{row 2} \\ \text{row 3} \end{array} \\ \text{column } 1 \quad 2 \quad 3 \end{array}$$

The numbers shown in matrix A are symbols that represent the entries for easy identification in the product, they are not to be considered actual numbers.

A is 2 by 3 **B** is 3 by 3 **A*B** will be 2 by 3

The entries in the product are calculated thus:

$$\mathbf{A*B} = \begin{array}{c} \begin{vmatrix} 1a+2b+3c & 1d+2e+3f & 1g+2h+3i \\ 4a+5b+6c & 4d+5e+6f & 4g+5h+6i \end{vmatrix} \begin{array}{l} \text{row 1} \\ \text{row 2} \end{array} \\ \text{column } 1 \qquad 2 \qquad 3 \end{array}$$

Note that in the product, the entry in row r, column c comes from row r of the first matrix and column c of the second matrix. This makes an entry in the product, call it Pij, come from row "i" of A and column "j" of B. The more general way of writing it is this:

$$\mathbf{A} = \begin{vmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \end{vmatrix}$$

$$\mathbf{B} = \begin{vmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \end{vmatrix}$$

A * B =

$$\begin{vmatrix} A_{11}B_{11}+A_{12}B_{21}+A_{13}B_{31} & A_{11}B_{12}+A_{12}B_{22}+A_{13}B_{32} & A_{11}B_{13}+A_{12}B_{23}+A_{13}B_{33} \\ A_{21}B_{11}+A_{22}B_{21}+A_{23}B_{31} & A_{21}B_{12}+A_{22}B_{22}+A_{23}B_{32} & A_{21}B_{13}+A_{22}B_{23}+A_{23}B_{33} \end{vmatrix}$$

Appendix A2 Matrix Inverse

A matrix must be square to have an inverse. However, not all square matrices have an inverse. The inverse of a matrix is another matrix which, when multiplied by the original matrix, results in the identity matrix. The identity matrix is the equivalent to the number 1 in the normal number world. that is:

In normal numbers $1/A = A^{-1}$ $\quad A * A^{-1} = 1$

In matrices inverse = $\mathbf{A}^{-1}$ $\quad \mathbf{A} * \mathbf{A}^{-1}$ = Identity matrix

The identity matrix is also square (n by n) and has a diagonal of all ones with all other entries equal to zero.

$$\mathbf{A}*\mathbf{A}^{-1} = \begin{vmatrix} 1 & 0 & 0 & 0 & ..0 \\ 0 & 1 & 0 & 0 & ..0 \\ 0 & 0 & 1 & 0 & ..0 \\ 0 & 0 & 0 & 1 & ..0 \\ : & : & : & : & ::: \\ 0 & 0 & 0 & 0 & ...1 \end{vmatrix} \quad \text{The identity matrix.}$$

This can be done with matrix math, but by looking at the rules for matrix multiplication, algebra can be used as well. To derive the inverse of a matrix, we first set up an equation with the product of the matrix times its inverse set equal to the identity matrix. (the numbers here are actual numbers):

$$\underset{\mathbf{A}}{\begin{vmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{vmatrix}} \quad * \quad \underset{\mathbf{A}^{-1}\ (\mathbf{B})}{\begin{vmatrix} a & d & g \\ b & e & h \\ c & f & i \end{vmatrix}} \quad = \quad \underset{\text{Identity}}{\begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}}$$

Then we solve for the entries in the inverse matrix using algebra. The quadratic Bézier basis matrix is used for an example. The inverse matrix (**B**) starts with unknowns that are solved for. Using the multiplication rules, write the equation for each entry in the product matrix, which is the identity matrix. Solve one column of the identity matrix at a time. There will be an equation for each row, or three in this case. The first column of the identity matrix comes from row 1, 2 & 3 of A and column 1 of **B** ($\mathbf{A}^{-1}$).

Solving for the first column in **B**:

$$\begin{array}{l} \underline{1\text{->}} \\ \underline{2\text{->}} \\ \underline{3\text{->}} \end{array} \underset{\mathbf{A}}{\begin{vmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{vmatrix}} \quad * \quad \underset{\mathbf{B}}{\begin{vmatrix} \underline{\mathbf{a}} & d & g \\ \underline{\mathbf{b}} & e & h \\ \underline{\mathbf{c}} & f & i \end{vmatrix}} \quad = \quad \begin{vmatrix} \underline{\mathbf{1}} & 0 & 0 \\ \underline{\mathbf{0}} & 1 & 0 \\ \underline{\mathbf{0}} & 0 & 1 \end{vmatrix}$$

A row 1, **B** col 1	**A** row 2, **B** col 1	**A** row 3, **B** col 1
$1a - 2b + 1c = 1$	$-2a + 2b + 0c = 0$	$1a + 0b + 0c = 0$

From the right hand equation it is clear that: **$\underline{a = 0}$**

Substituting this into the left and center equations gives:

$-2b + c = 1$ $\quad 2b = 0$or **$\underline{b = 0}$**

Therefore **$\underline{c = 1}$**

Solving for the second column in **B**:

$$\begin{array}{l} \underline{1\text{->}} \\ \underline{2\text{->}} \\ \underline{3\text{->}} \end{array} \underset{\mathbf{A}}{\begin{vmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{vmatrix}} \quad * \quad \underset{\mathbf{B}}{\begin{vmatrix} a & \underline{\mathbf{d}} & g \\ b & \underline{\mathbf{e}} & h \\ c & \underline{\mathbf{f}} & i \end{vmatrix}} \quad = \quad \begin{vmatrix} 1 & \underline{\mathbf{0}} & 0 \\ 0 & \underline{\mathbf{1}} & 0 \\ 0 & \underline{\mathbf{0}} & 1 \end{vmatrix}$$

A row 1, **B** col 2	**A** row 2, **B** col 2	**A** row 3, **B** col 2
$1d - 2e + 1f = 0$	$-2d + 2e + 0f = 1$	$1d + 0e + 0f = 0$

From the right hand equation it is clear that: **$\underline{d = 0}$**

Substituting this into the left and center equations gives:

$-2e + f = 0$ $\quad 2e = 1$or **$\underline{e = 1/2}$**

Therefore

$-2(1/2) + f=0$ $\quad -1 + f = 0$ $\quad$ **$\underline{f = 1}$**

Solving for the third column in **B**:

$$\begin{array}{l} \underline{1\text{->}} \\ \underline{2\text{->}} \\ \underline{3\text{->}} \end{array} \underset{\mathbf{A}}{\begin{vmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{vmatrix}} \quad * \quad \underset{\mathbf{B}}{\begin{vmatrix} a & d & \underline{\mathbf{g}} \\ b & e & \underline{\mathbf{h}} \\ c & f & \underline{\mathbf{i}} \end{vmatrix}} \quad = \quad \begin{vmatrix} 1 & 0 & \underline{\mathbf{0}} \\ 0 & 1 & \underline{\mathbf{0}} \\ 0 & 0 & \underline{\mathbf{1}} \end{vmatrix}$$

A row 1, **B** col 3	**A** row 2, **B** col 3	**A** row 3, **B** col 3
$1g - 2h + 1i = 0$	$-2g + 2h + 0i = 0$	$1g + 0h + 0i = 1$

From the right hand equation it is clear that: **$\underline{g = 1}$**

Substituting this into the left and center equations gives:

$1 - 2h + i = 0$ $\quad -2 + 2h = 0$ $\quad$ **$\underline{h = 1}$**

Therefore

$1 - 2 + i = 0$ **$\underline{i = 1}$**

Putting these into the matrix gives the Quadratic Bézier basis matrix inverse:

$$\mathbf{A}^{-1} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 1/2 & 1 \\ 1 & 1 & 1 \end{vmatrix} \quad \text{or} \quad \begin{vmatrix} 0 & 0 & 2 \\ 0 & 1 & 2 \\ 2 & 2 & 2 \end{vmatrix} * \frac{1}{2}$$

As a test for the correct solution, the original matrix and its inverse are multiplied and the identity matrix should result. If the identity matrix does not result, the location of the bad entries helps identify which unknowns are in error.

This is a good place to put to use a spreadsheet program. Modern spreadsheet programs have many built in functions for matrix operations. If not, they can easily be built up with formulas.

Appendix A3 Conversions Between Types

The conversions between many of the more common types of interpolation are shown here. They are shown in two forms; matrix and equation. The matrix is in Noskowicz notation and has been normalized to remove fractions. When no normalization was required, 1/1 is shown explicitly to avoid ambiguity. The equation forms have been reduced to the lowest common denominator. They have been spaced into a columnar form for easier readability in order to help avoid mis-reading coefficients.

Converting between types in this manner is a conversion between control points for the master curve of individual segments only. The conversion is done for each segment of the original curve and must, of course, use the control points for that segment. The converted piecewise curve <u>will</u> be identical. However, because of differences in control point arrangements, the resulting control point sets may not always be handled in the normal way when stringing segments together.

An example of this is a conversion from cubic Bézier control points to cubic B-spline. Adjacent Bézier segments have only the end control points in common. Adjacent B-spline segments have three control points in common. However, Bézier segments converted to B-spline will have no common control points between segments. See Chapter 16 for a full explanation.

Pa Pb Pc & Pd are the original geometry vector entries (starting control points).
P_0, P_1, P_2 & P_3 are the new geometry vector entries (desired control points).

Bézier to Hermite (in Bézier-like form)

$$\begin{array}{c} \\ P_0 \\ T_0 \\ T_3 \\ P_3 \end{array} \begin{array}{c} \begin{matrix} Pa & Pb & Pc & Pd \end{matrix} \\ \left|\begin{matrix} 3 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 \\ 0 & 0 & -1 & 3 \\ 0 & 0 & 0 & 3 \end{matrix}\right| \end{array} \frac{1}{3}$$

$$\begin{aligned} P_0 &= Pa \\ T_0 &= (3Pa + Pb)/3 \\ T_3 &= (-Pc + 3Pd)/3 \\ P_3 &= Pd \end{aligned}$$

Hermite (in Bézier-like form) to Bézier

$$\begin{array}{c} \\ P_0 \\ P_1 \\ P_2 \\ P_3 \end{array} \begin{array}{c} \begin{matrix} Pa & Ta & Td & Pd \end{matrix} \\ \left|\begin{matrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \\ 0 & 0 & 0 & 1 \end{matrix}\right| \end{array} \frac{1}{1}$$

$$\begin{aligned} P_0 &= Pa \\ P_1 &= -3Pa + 3Ta \\ P_2 &= -3Td + 3Pd \\ P_3 &= Pd \end{aligned}$$

Golden to Quadratic Bézier

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 2 & 0 & 0 \\ -1 & 4 & -1 \\ 0 & 0 & 2 \end{matrix}\right| \end{array} \frac{1}{2}$$

$$\begin{aligned} P_0 &= Pa \\ P_1 &= (-Pa + 4Pb - Pc)/2 \\ P_2 &= Pc \end{aligned}$$

Quadratic Bézier to Golden

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 4 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 4 \end{matrix}\right| \end{array} \frac{1}{4}$$

$$\begin{aligned} P_0 &= Pa \\ P_1 &= (Pa + 2Pb + Pc)/4 \\ P_2 &= Pc \end{aligned}$$

Parabolic to Quadratic Bézier

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 1 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 1 & 1 \end{matrix}\right| \end{array} \frac{1}{2}$$

$$\begin{aligned} P_0 &= (Pa + Pb)/2 \\ P_1 &= Pb \\ P_2 &= (Pb + Pc)/2 \end{aligned}$$

Quadratic Bézier to Parabolic

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 2 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 2 \end{matrix}\right| \end{array} \frac{1}{1}$$

$$\begin{aligned} P_0 &= 2Pa - Pb \\ P_1 &= Pb \\ P_2 &= -Pb + 2Pc \end{aligned}$$

Parabolic to Golden

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 4 & 4 & 0 \\ 1 & 6 & 1 \\ 0 & 4 & 4 \end{matrix}\right| \end{array} \frac{1}{8}$$

$$\begin{aligned} P_0 &= (Pa + Pb)/2 \\ P_1 &= (Pa + 6Pb + Pc)/8 \\ P_2 &= (Pb + Pc)/2 \end{aligned}$$

Golden to Parabolic

$$\begin{array}{c} \begin{matrix} Pa & Pb & Pc \end{matrix} \\ \left|\begin{matrix} 5 & -4 & 1 \\ -1 & 4 & -1 \\ 1 & -4 & 5 \end{matrix}\right| \end{array} \frac{1}{2}$$

$$\begin{aligned} P_0 &= (5Pa - 4Pb + Pc)/2 \\ P_1 &= (-Pa + 4Pb - Pc)/2 \\ P_2 &= (Pa - 4Pb + 5Pc)/2 \end{aligned}$$

B-Spline to Bézier

$$\begin{array}{c|cccc|c} & Pa & Pb & Pc & Pd & \\ P_0 & 1 & 4 & 1 & 0 & \\ P_1 & 0 & 4 & 2 & 0 & \frac{1}{6} \\ P_3 & 0 & 2 & 4 & 0 & \\ P_4 & 0 & 1 & 4 & 1 & \end{array}$$

$$\begin{aligned} P_0 &= (\ Pa + 4Pb + Pc\ )/6 \\ P_1 &= (\ 2Pb + Pc\ )/3 \\ P_2 &= (\ Pb + 2Pc\ )/3 \\ P_3 &= (\ Pb + 4Pc + Pd\ )/6 \end{aligned}$$

Bézier to B-Spline

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 6 & -7 & 2 & 0 & \\ 0 & 2 & -1 & 0 & \frac{1}{1} \\ 0 & -1 & 2 & 0 & \\ 0 & 2 & -7 & 6 & \end{array}$$

$$\begin{aligned} P_0 &= 6Pa - 7Pb + 2Pc \\ P_1 &= 2Pb - Pc \\ P_2 &= -Pb + 2Pc \\ P_3 &= 2Pb - 7Pc + 6Pd \end{aligned}$$

Bézier to Catmul

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 6 & -6 & 0 & 1 & \\ 1 & 0 & 0 & 0 & \frac{1}{1} \\ 0 & 0 & 0 & 1 & \\ 1 & 0 & -6 & 6 & \end{array}$$

$$\begin{aligned} P_0 &= 6Pa - 6Pb + Pd \\ P_1 &= Pa \\ P_2 &= Pd \\ P_3 &= Pa - 6Pc + 6Pd \end{aligned}$$

Catmul to Bézier

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 0 & 6 & 0 & 0 & \\ -1 & 6 & 1 & 0 & \frac{1}{6} \\ 0 & 1 & 6 & -1 & \\ 0 & 0 & 6 & 0 & \end{array}$$

$$\begin{aligned} P_0 &= Pb \\ P_1 &= (\ -Pa + 6Pb + Pc\ )/6 \\ P_2 &= (\ +Pb + 6Pc - Pd\ )/6 \\ P_3 &= (\ Pc \end{aligned}$$

B-Spline to Catmul

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 6 & 1 & -2 & 1 & \\ 1 & 4 & 1 & 0 & \frac{1}{6} \\ 0 & 1 & 4 & 1 & \\ 1 & -2 & 1 & 6 & \end{array}$$

$$\begin{aligned} P_0 &= (6Pa + Pb - 2Pc + Pd\ )/6 \\ P_1 &= (\ Pa + 4Pb + Pc\ )/6 \\ P_2 &= (\ Pb + 4Pc + Pd\ )/6 \\ P_3 &= (\ Pa - 2Pb + Pc + 6Pd\ )/6 \end{aligned}$$

Catmul to B-Spline

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 7 & -4 & 5 & -2 & \\ -2 & 11 & -4 & 1 & \frac{1}{6} \\ 1 & -4 & 11 & -2 & \\ -2 & 5 & -4 & 7 & \end{array}$$

$$\begin{aligned} P_0 &= (\ 7Pa - 4Pb + 5Pc - 2Pd\ )/6 \\ P_1 &= (-2Pa + 11Pb - 4Pc + Pd\ )/6 \\ P_2 &= (\ Pa - 4Pb + 11Pc - 2Pd\ )/6 \\ P_3 &= (-2Pa + 5Pb - 4Pc + 7Pd\ )/6 \end{aligned}$$

Bézier to Cubic 4pt

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 27 & 0 & 0 & 0 & \\ 8 & 12 & 6 & 1 & \frac{1}{27} \\ 1 & 6 & 12 & 8 & \\ 0 & 0 & 0 & 27 & \end{array}$$

$$\begin{aligned} P_0 &= Pa \\ P_1 &= (\ 8Pa + 12Pb + 6Pc + Pd\ )/27 \\ P_2 &= (\ Pa + 6Pb + 12Pc + 8Pd\ )/27 \\ P_3 &= Pd \end{aligned}$$

Cubic 4pt to Bézier

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 6 & 0 & 0 & 0 & \\ -5 & 18 & -9 & 2 & \frac{1}{6} \\ 2 & -9 & 18 & -5 & \\ 0 & 0 & 0 & 6 & \end{array}$$

$$\begin{aligned} P_0 &= Pa \\ P_1 &= (-5Pa + 18Pb - 9Pc + 2Pd\ )/6 \\ P_2 &= (\ 2Pa - 9Pb + 18Pc - 5Pd\ )/6 \\ P_3 &= Pd \end{aligned}$$

B-Spline to Cubic 4pt

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 27 & 108 & 27 & 0 & \\ 8 & 93 & 60 & 1 & \frac{1}{162} \\ 1 & 60 & 93 & 8 & \\ 0 & 27 & 108 & 27 & \end{array}$$

$$\begin{aligned} P_0 &= (\ Pa + 4Pb + Pc\ )/6 \\ P_1 &= (8Pa + 93Pb + 60Pc + Pd\ )/162 \\ P_2 &= (\ Pa + 60Pb + 93Pc + 8Pd\ )/162 \\ P_3 &= (\ Pb + 4Pc + Pd\ )/6 \end{aligned}$$

Cubic 4pt to B-Spline

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 25 & -48 & 33 & -8 & \\ -4 & 15 & -12 & 3 & \frac{1}{2} \\ 3 & -12 & 15 & -4 & \\ -8 & 33 & -48 & 25 & \end{array}$$

$$\begin{aligned} P_0 &= (25Pa - 48Pb + 33Pc - 8Pd\ )/2 \\ P_1 &= (-4Pa + 15Pb - 12Pc + 3Pd\ )/2 \\ P_2 &= (\ 3Pa - 12Pb + 15Pc - 4Pd\ )/2 \\ P_3 &= (-8Pa + 33Pb - 48Pc + 25Pd\ )/2 \end{aligned}$$

Catmul to Cubic 4pt

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 0 & 27 & 0 & 0 & \\ -2 & 21 & 9 & -1 & \frac{1}{27} \\ -1 & 9 & 21 & -2 & \\ 0 & 0 & 27 & 0 & \end{array}$$

$$\begin{aligned} P_0 &= Pb \\ P_1 &= (-2Pa + 21Pb + 9Pc - Pd\ )/27 \\ P_2 &= (\ -Pa + 9Pb + 21Pc - 2Pd\ )/27 \\ P_3 &= Pc \end{aligned}$$

Cubic 4pt to Catmul

$$\begin{array}{|cccc|c} Pa & Pb & Pc & Pd & \\ 11 & -18 & 9 & -1 & \\ 1 & 0 & 0 & 0 & \frac{1}{1} \\ 0 & 0 & 0 & 1 & \\ -1 & 9 & -18 & 11 & \end{array}$$

$$\begin{aligned} P_0 &= 11Pa - 18Pb + 9Pc - Pd \\ P_1 &= Pa \\ P_2 &= Pd \\ P_3 &= -Pa + 9Pb - 18Pc + 11Pd \end{aligned}$$

Appendix A4 Bernstein Functions

Here are the Bernstein Functions, how to interpret them and convert them to polynomials. For a given degree (d), the Bernstein Functions provide a group of (d+1) equations. Each of these equations is a weighting function for one control point. Starting from the standard Bernstein definition requires a little algebra to get to our standard polynomial form.

The standard Bernstein definition:

$$J_f^d(t) = (B) * t^f * (1-t)^{(d-f)}$$

Where:
d is the degree (2=Quadratic) (3=Cubic)
f is the function number and goes from 0 to d (f = 0, 1, 2, ..., d)
B is called the binomial coefficient (explained in any college algebra book) that is:

$$B = \frac{d!}{f! * (d-f)!}$$

The "!" is the symbol for factorial. The factorials are this:

```
0! = 1  (this is a definition, don't try to understand)
1! = 1
2! = 2*1 = 2
3! = 3*2*1 = 6
4! = 4*3*2*1 = 24
```

As an example of the algebraic manipulation required, take the first equation (f=0) for the second degree (d=2) Bernstein function.

Substituting values (d=2 and f=0):

$$J_0^2(t) = \frac{2!}{0! * (2-0)!} * t^0 * (1-t)^{(2-0)}$$

Evaluating the factorials:

$$J_0^2(t) = \frac{2}{1 * 2} * t^0 * (1-t)^{(2)}$$

Simplifying further and expanding the squared quantity:

$$\begin{aligned} J_0^2(t) &= 1 * t^0 * (1-t)^2 = \\ &\quad 1 * 1 * (1-2t+t^2) = \\ &\quad +t^2 -2t +1 \end{aligned}$$

Doing the same type of manipulations with the equation for f=1 and 2 until standard polynomials are obtained gives:
(d=2, so there are three functions f=0, 1, 2):

$$J_0^2(t) = (1) * t^0 * (1-t)^2 = t^2 - 2t + 1$$
$$J_1^2(t) = (2) * t^1 * (1-t)^1 = -2t^2 + 2t$$
$$J_2^2(t) = (1) * t^2 * (1-t)^0 = t^2$$

(compare with equations 6.1a, b & c)

The same procedure is followed for any degree Bézier.

Appendix A5 Lagrange Interpolating Functions

For a series of points defined as $P_n = (x_n, y_n)$, n = 0 to d , the Lagrange function gives an equation of degree d that passes through all the points.

Chapter 15 describes the parameterized version of the Lagrange.

The general form of the Lagrange Functions for interpolation:

$$y = y_0L_0^d(t) + y_1L_1^d(t) + y_2L_2^d(t) + y_3L_3^d(t) \ldots$$

Where:

$$L_f^d(x) = \text{Product of all } \frac{(x-xj)}{(xf-xj)}$$

d is the degree (2=Quadratic) (3=cubic)
f is the function number (f=0, 1,...,d)
j multiplicand number (j = 0, 1, 2...d; but never=f).

For a given degree, d, you will have:

- d+1 known data points.
- d+1 "L_f^d" equations (f=0, 1,...,d)
- d factors in each equation (j = 0, 1, 2...d; but j not=f)

The pattern looks like this.

$$L_0^d(x) = \text{omit} * \frac{(x-x1)}{(x0-x1)} * \frac{(x-x2)}{(x0-x2)} * \frac{(x-x3)}{(x0-x3)} * \frac{(x-x4)}{(x0-x4)} \ldots \frac{(x-xd)}{(x0-xd)}$$

$$L_1^d(x) = \frac{(x-x0)}{(x1-x0)} * \text{omit} * \frac{(x-x2)}{(x1-x2)} * \frac{(x-x3)}{(x1-x3)} * \frac{(x-x4)}{(x1-x4)} \ldots \frac{(x-xd)}{(x1-xd)}$$

$$L_2^d(x) = \frac{(x-x0)}{(x2-x0)} * \frac{(x-x1)}{(x2-x1)} * \text{omit} * \frac{(x-x3)}{(x2-x3)} * \frac{(x-x4)}{(x2-x4)} \ldots \frac{(x-xd)}{(x2-xd)}$$

$$L_3^d(x) = \frac{(x-x0)}{(x3-x0)} * \frac{(x-x1)}{(x3-x1)} * \frac{(x-x2)}{(x3-x2)} * \text{omit} * \frac{(x-x4)}{(x3-x4)} \ldots \frac{(x-xd)}{(x3-xd)}$$

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$L_M^d(x) = \frac{(x-x0)}{(xd-x0)} * \frac{(x-x1)}{(xd-x1)} * \frac{(x-x2)}{(xd-x2)} * \frac{(x-x3)}{(xd-x3)} * \frac{(x-x4)}{(xd-x4)} * \text{omit}$$

Appendix A6 Derivative

The derivative of a function is the slope of that function. It is useful to have the derivative of a polynomial for setting constraints on the slope of a function as well as determining end slope values of existing curves and determining continuity. The first derivative of a (position) function is another function that defines the slope (velocity) of the function for any value of t. The second derivative of position function is another function that defines the change in slope (acceleration or change in velocity) of the function for any value of t. The method to obtain the derivative of the type of polynomial used in interpolation is described here.

There are two common ways to designate a derivative, the prime mark (') or the df/dt notation. The first derivative of f(t) is shown as f'(t) or df/dt. The second derivative is shown as f"(t) or d^2f/d^2t. These are not exponents, but part of the special notation for derivative. We use the prime notation here.

The derivative of a sum is the sum of the individual derivatives of those terms. Since a polynomial is a sum of terms, the individual terms are done independently. Each has a derivative that is a term in the resulting derivative function, or if:

$$f(t) = g(t) + h(t) + i(t) + j(t) \quad \text{then}$$
$$f'(t) = g'(t) + h'(t) + i'(t) + j'(t)$$

In general, the derivative looks like this:

$$\text{Derivative of } At^n = n * At^{(n-1)}$$

The three terms that have powers of t are all handled the same way. The derivative of these terms is a simple two step process.

1- Multiply the term by the exponent of "t' in that term.(put the exponent of t as another factor "in front" of the term)

2- Subtract one from the exponent.

Constant terms, such as "D", have a slope of zero and drop out. This is not a special case. This can be seen by considering the "D" term to actually have "t" to the zero power or:

$$Dt^0$$

The derivative then is: $0 * D\,t^{-1}$ that is 0.

We will use a degree three polynomial as an example.

$$f(t) = At^3 + Bt^2 + Ct + D$$

This makes our example:

$$f\,'(t) = 3At^2 + 2Bt^1 + 1Ct^0 + 0 \quad \text{or just}$$
$$f\,'(t) = 3At^2 + 2Bt + C$$

Repeating the procedure for the second derivative:

$$f''(t) = 2*3A^{2-1} + 1*2Bt^{1-1}$$

$$f''(t) = 6At + 2B$$

Matrix Notation Derivative
]
Taking the derivative of a basis matrix is quite simple. The same two steps are performed, however in the basis matrix this becomes a "multiply and move down" operation. Multiply the matrix entry by the exponent of "t" to its left. Then place it down one row in the derivative matrix. The bottom row disappears and zeros in any location move down.

Note that because in the second row from the bottom the exponent of t is one, the entries in this row move down unchanged. The (g h i) row in the matrix appears in the bottom row of the first derivative. The (2d 2e 2f) row in the first derivative appears in the bottom row of the second derivative. This characteristic makes finding and comparing end tangents easy. See Appendix A9.

Basis Matrix

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{ccc} a & b & c \\ d & e & f \\ g & h & i \\ j & k & l \end{array}\right|$$

First Derivative

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{ccc} 0 & 0 & 0 \\ 3a & 3b & 3c \\ 2d & 2e & 2f \\ g & h & i \end{array}\right|$$

Second Derivative

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 6a & 6b & 6c \\ 2d & 2e & 2f \end{array}\right|$$

A constant factor, such as the 1/6 of the B-spline, is unaffected since it appears as a coefficient of all terms. Just like the basis matrix, the derivatives will also have symmetric beginning (bottom row) and end (sum of columns) coefficients. The sum of columns will have the same pattern as the bottom row, just shifted to the align with the end of the segment. Examples of the Quadratic Bézier, B-spline and Cubic Bézier are shown.

Quadratic Bézier

$$\begin{array}{c} t^2 \\ t \\ 1 \end{array}\left|\begin{array}{ccc} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{array}\right|$$

Quadratic Bézier Derivative

$$\begin{array}{c} t^2 \\ t \\ 1 \end{array}\left|\begin{array}{ccc} 0 & 0 & 0 \\ 2 & -4 & 2 \\ -2 & 2 & 0 \end{array}\right|$$

B-spline

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{cccc} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{array}\right| \frac{1}{6}$$

B-spline Derivative

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{cccc} 0 & 0 & 0 & 0 \\ -3 & 9 & -9 & 3 \\ 6 & -12 & 6 & 0 \\ -3 & 0 & 3 & 0 \end{array}\right| \frac{1}{6}$$

Cubic Bézier

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{cccc} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right|$$

Cubic Bézier Derivative

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{cccc} 0 & 0 & 0 & 0 \\ -3 & 9 & -9 & 3 \\ 6 & -12 & 6 & 0 \\ -3 & 3 & 0 & 0 \end{array}\right|$$

Second Derivative

$$\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}\left|\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -6 & 18 & -18 & 6 \\ 6 & -12 & 6 & 0 \end{array}\right|$$

Appendix A7a Mirror by Substitution

The term "mirror function" used herein refers to the function that is a reflection, of a given function, about the t = 0.5 line. The method of deriving a mirror function by substituting (1-t) for t is shown here. The equation used as the example is the cubic function with end-slopes of zero (equation 4.7).

$$F = -2t^3 + 3t^2 \qquad (4.7)$$

Every occurrence of "t" is replaced by (1-t) to give the mirror function "Fm".

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

Evaluating the powers of (1-t) gives:

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

Expanding these in the equation gives:

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

Multiplying to remove the parentheses:

$$Fm = 2t^3 - 6t^2 + 6t - 2 + 3t^2 - 6t + 3$$

Arranging like terms for combining:

$$Fm = 2t^3 - 6t^2 + 3t^2 + 6t - 6t - 2 + 3$$

Combining gives:

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

Also, substituting (1-t) for t in this equation would return us to the original function.

Appendix A7b Mirror By Constraints

Another way to derive a mirror function is to define the constraints to mirror the original ones. The procedure used here is identical to that in Chapter 4 except the constraints are changed to reflect the mirror requirements. The places where differences occur from Chapter 4. are shown as **bold underline**.

A cubic, or third degree function has a general form with the third power of t:

$$F = At^3 + Bt^2 + Ct + D \qquad (4.1)$$

Again, the goal is to find the four constants (A, B, C, D). We proceed in the same manner starting with the two end values of **one/zero** then incorporating the end slopes.

For the end values:

When $t = 0$, $F = \underline{\mathbf{1}} = 0 + 0 + 0 + D$ Therefore $D = \underline{\mathbf{1}}$ (4.2m)

When $t = 1$, $F = \underline{\mathbf{0}} = A + B + C + D$

Therefore $A+B+C\underline{\mathbf{+1}}=\underline{\mathbf{0}}$ (4.3m)

The slope of equation 4.1m is the first derivative or:

$$Slope = dF/dt = 3At^2 + 2Bt + C \qquad (4.4m)$$

The mirror function also has zero slope at both ends. Substitute into equation 4.4m to get:

When $t = 0$, $dF/dt = 0 = 0 + 0 + C$ Therefore $C = 0$ (4.5m)

When $t = 1$, $dF/dt = 0 = 3A + 2B + 0$ (4.6m)

Again, with two unknowns and two equations (4.3m & 4.6m) we solve for A & B. From equation 4.3m with C = 0 and D = 1

$$A + B + 1 = 0$$
$$B = \underline{\mathbf{-A-1}}.$$

Substituting for B in 4.6m gives:

$$0 = 3A + 2(\underline{\mathbf{-A-1}})$$
$$0 = 3A - 2A - 2 \qquad \text{or} \qquad \underline{\mathbf{A = + 2}}$$
then since $B = \underline{\mathbf{-A-1}}$ $\qquad \underline{\mathbf{B = - 3}}$

Substituting all the constants into equation 4.1m gives the mirror function:

$$MF = 2t^3 - 3t^2 + 1 \qquad (4.7m)$$

Appendix A8 End-Slope Values for Center slope=0

This is a derivation of the end-slope value that causes the slope in the center to be zero for the "adjustable end-slope cubic function" of Chapter 4 equation 4.10.
The slope of a function is the first derivative. To find the value of end slope (S) that gives a slope of zero at t = 0.5, we set t equal to 0.5 and the whole slope equation equal to zero, then solve for S.
Starting with the function (equation 4.10)

$$F = (2S-2)\ t^3 + (3-3S)\ t^2 + S\ t \qquad (4.10)$$

The slope or first derivative is:

$$dF/dt = 3(2S-2)\ t^2 + 2(3-3S)\ t^1 + S$$

Set this equal to zero and t equal to 0.5 then solve for the value of S.

$$0 = 3(2S-2)\ /4 + 2(3-3S)\ /2 + S$$

Carry out the multiplications indicated:

$$0 = (6S-6)\ /4 + (6-6S)\ /2 + S$$

Multiply by 4 to clear fractions:

$$0 = 6S-6 + 2(6-6S) + 4S$$

Multiply as indicated:

$$0 = 6S - 6 + 12 - 12S + 4S$$

Rearrange like terms:

$$0 = - 6 + 12 + 6S - 12S + 4S$$

Combine like terms:

$$0 = 6 - 2S$$

Subtract 6:

$$-6 = - 2S$$

Divide by -2:

$$3 = S$$

This is the value of S that causes the slope to be zero at t = 0.5.

Appendix A9 Observable Matrix Characteristics

Direct observation of the basis matrix allows some characteristics of the curve to be easily determined. These characteristics are valid for both the fixed and variable curves, however, since the Hermite basis matrix has the tangents directly, these methods are not needed.
It is useful to remind the reader that the basis matrix is the collection of coefficients of a single polynomial equation and, as a result, we may handle them as we would handle terms in an equation. The presence of the powers of "t" and the control point designations in Noskowicz Notation allows a number of characteristics to be found by observation of the basis matrix.

Start Point

The bottom row (t^0) of the basis matrix shows us where the curve segment begins. At the start of a segment, $t = 0$. This makes all terms containing "t" also zero. Therefore, since there are no "t" terms in the bottom row, only the bottom row of the basis matrix has any effect on the interpolated value.
A one in this row is under the control point the segment begins on.

For example, we see the following types have a one in the first row under P_1 and therefore start there - the Linear, Simple Cubic, Quadratic Bézier, Quadratic Lagrange and Catmul-Rom (remember the 1/2). For the cubic Bézier the curve starts at P_0 because the 1 in the bottom row is below it.

If there are two or three entries in this row, the curve does not pass through a point. In this case, the relative value of the coefficients in the row shows the relative weight or contribution of each control point to the location of the segment start. This can be seen in the Parabolic and cubic B-spline.

For all types, you must be careful to include the "true" values of the basis matrix since it is common practice to make them integers by having an external factor as a fraction; such as the Catmul-Rom with its 1/2. This can be easily seen by looking at the Parabolic basis matrix. There are two 1's in the bottom row and the main factor is 1/2. This means that the two basis matrix values are 0.5. The P_0 and P_1 weighting functions both have a value of 0.5. This is the formula for a linear interpolation half way between the ends - therefore, the Parabolic starts half way between P_0 and P_1.

The B-spline has `[1   4   1]` in the bottom row (divided by 6). This shows that the segment start is nearest to P_1 (weight of 4/6), with 1/6 of its location due to P_0 and 1/6 due to P_2.

Linear

$$\begin{array}{c|cc|l} & P_1 & P_2 & \\ t & -1 & 1 & \\ 1 & 1 & 0 & \texttt{<-- Start location} \end{array}$$

Simple Cubic

$$\begin{array}{c|cc|l} & P_1 & P_2 & \\ t^3 & 2 & -2 & \\ t^2 & -3 & 3 & \\ t & 0 & 0 & \\ 1 & 1 & 0 & \texttt{<-- Start location} \end{array}$$

Quadratic Bézier

$$\begin{array}{c|ccc|l} & P_1 & P_2 & P_3 & \\ t^2 & 1 & -2 & 1 & \\ t & -2 & 2 & 0 & \\ 1 & 1 & 0 & 0 & \texttt{<-- Start location} \end{array}$$

Quadratic Lagrange

$$\begin{array}{c|ccc|l} & P_1 & P_2 & P_3 & \\ t^2 & n1 & n2 & n3 & \\ t & -n1-1 & -n2 & -n3*t2 & \\ 1 & 1 & 0 & 0 & \texttt{<-- Start location} \end{array}$$

Catmul-Rom

$$\begin{array}{c|cccc|l} & P_0 & P_1 & P_2 & P_3 & \\ t^3 & -1 & 3 & -3 & 1 & 1 \\ t^2 & 2 & -5 & 4 & -1 & \text{---} \\ t & -1 & 0 & 1 & 0 & 2 \\ 1 & 0 & 2 & 0 & 0 & \texttt{<-- Start location} \end{array}$$

Cubic Bézier

$$\begin{array}{c|cccc|l} & P_0 & P_1 & P_2 & P_3 & \\ t^3 & -1 & 3 & -3 & 1 & \\ t^2 & 3 & -6 & 3 & 0 & \\ t & -3 & 3 & 0 & 0 & \\ 1 & 1 & 0 & 0 & 0 & \texttt{<-- Start location} \end{array}$$

Parabolic

$$\begin{array}{c|ccc|l} & P_0 & P_1 & P_2 & \\ t^2 & 1 & -2 & 1 & \\ t & -2 & 2 & 0 & 1/2 \\ 1 & 1 & 1 & 0 & \texttt{<-- Start location} \end{array}$$

B-spline

$$\begin{array}{c|cccc|l} & P_0 & P_1 & P_2 & P_3 & \\ t^3 & -1 & 3 & -3 & 1 & 1 \\ t^2 & 3 & -6 & 3 & 0 & \text{---} \\ t & -3 & 0 & 3 & 0 & 6 \\ 1 & 1 & 4 & 1 & 0 & \texttt{<-- Start location} \end{array}$$

Observable Characteristics A9

Stop Point

At the finish of a segment, t = 1 and the powers of t are also = 1. All terms appear at face value. Therefore, to see where the segment stops, we look at the sum of columns of the basis matrix. The same characteristics apply here as at the segment start described above. Since all interpolation types described in this book are symmetrical, the column sum will have the same pattern as the bottom row, just shifted to the segment stop point.

End point Tangent Vectors

The slope of a function is its first derivative. We can determine tangent vectors at the segment end points, just as we did with the start and stop locations, by examining the derivative of the basis matrix.

Start Tangent Vector

At the start, t = 0. The only remaining terms are in the bottom row of the derivative matrix, therefore, this shows the tangent vector. However, this row is identical to the second row of the basis matrix itself due to the way the derivative works. This is due to the fact that the exponent of t in the second row of the basis matrix is one, leaving the coefficients unchanged when moved to the bottom row of the derivative. Therefore, the second row of the basis matrix also shows the tangent vector at the segment start. We do not need to take the derivative to see this. See Appendix A6.

For the case where the curve is tangent to the line connecting two control points, these coefficients have a specific pattern. There will only be two entries in this row and they will be equal and opposite in sign, the negative one being to the left. The curve will be tangent to a straight line connecting these two points. The magnitude of the entries gives the magnitude of the tangent vector relative to the spacing of these two control points.

A trivial example is the linear that has [-1 1] in the second row. These numbers are under P_1 and P_2. Therefore, the tangent vector is parallel to the line connecting P_1 and P_2 and equal to that distance.

The Parabolic has [-2 2] 1/2 in the second row. These are under P_0 and P_1. Therefore, the start tangent vector is parallel to the line connecting P_0 and P_1 and equal to that distance.

The Quadratic Bézier has [-2 2] in the second row. These numbers are under P_1 and P_2. Therefore, the start tangent vector is parallel to the line connecting P_1 and P_2 and equal to twice that distance.

Linear

	P_1	P_2	
t	-1	1	<-- Start tangent
1	1	0	<-- Start location
SUM->	0	1	<-- Stop location

Parabolic

	P_0	P_1	P_2	
t^2	1	-2	1	1/2
t	-2	2	0	<-- Start tangent
1	1	1	0	<-- Start location
SUM->	0	1	1	<-- Stop location

Quadratic Bézier

	P_1	P_2	P_3	
t^2	1	-2	1	
t	-2	2	0	<-- Start tangent
1	1	0	0	<-- Start location
SUM->	0	0	1	<-- Stop location

The Cubic Bézier has [-3 3] in the second row. These numbers are under P_0 and P_1. Therefore, the start tangent vector is parallel to the line connecting P_0 and P_1 and equal to three times that distance.

The B-spline and Catmul-Rom both have `[-0.5  0.5]` in the second row (remember the fraction outside the matrix). However, notice that these numbers are under P_0 and P_2. Therefore, both of these have start tangent vectors parallel to the line connecting P_0 and P_2 that are equal to one half that distance.

This can also be seen in variable curves such as the two Bézier family curves with end velocity control. Looking at the quadratic, we see `[-2-V  +2+V]` in the second row. These numbers are under P_1 and P_2. Therefore, the start tangent vector is parallel to the line connecting P_1 and P_2 and equal to 2+V times that distance.

Stop Tangent Vector

As with the start point, the tangent vector at the segment stop would be found by examining the sum of the columns in the <u>derivative</u> of the basis matrix. The same characteristics apply here as at the segment start above. Since all interpolation types described in this book are symmetrical, the column sum will have the same pattern as the bottom row, just shifted to the segment stop point.

This method can be used to determine the level of continuity that is exhibited at the joints by examination of only the basis matrix derivative.

Determining Continuity

We can determine the degree of parametric continuity at joints by a relatively simple analysis of the basis matrix. From the above end position analysis we found where the segments start and stop from the basis matrix itself. For higher levels of continuity we move to the derivatives of the basis matrix. Since the first derivative of the basis matrix gives the tangent, we use an identical analysis on it, giving us the tangent at the segment end point rather than the locations at the segment end.

Of course, each segment is determined by a different set of control points. As we move from segment to segment, we may shift one, two, three or more in the list of control points. To compare the derivatives on each side of the joint, we must therefore use the proper control points that determine the specific end conditions that we are comparing. We examine two types here.

Cubic Bézier

	P_0	P_1	P_2	P_3	
t^3	-1	3	-3	1	
t^2	3	-6	3	0	
t	-3	3	0	0	
1	1	0	0	0	<-- Start location

B-spline

	P_0	P_1	P_2	P_3	
t^3	-1	3	-3	1	
t^2	3	-6	3	0	1/6
t	-3	0	3	0	<--- Start Tangent
1	1	4	1	0	<--- Start location
SUM->	0	1	4	1	<--- Stop location

Cubic B-spline <u>derivative</u>

	P_0	P_1	P_2	P_3	
t^3	0	0	0	0	
t^2	-3	9	-9	3	1/6
t	6	-12	6	0	
1	-3	0	3	0	<--- Start Tangent
SUM->	0	-3	0	3	<--- Stop Tangent

Cubic Bézier

	P_0	P_1	P_2	P_3	
t^3	-1	3	-3	1	
t^2	3	-6	3	0	
t	-3	3	0	0	<--- Start Tangent
1	1	0	0	0	<--- Start location
SUM->	0	0	0	1	<--- Stop location

Catmul-Rom

	P_0	P_1	P_2	P_3	
t^3	-1	3	-3	1	
t^2	2	-5	4	-1	1/2
t	-1	0	1	0	
1	0	2	0	0	<-- Start location

Quadratic Bézier family with end velocity control

	P_0	P_1	P_2	P_3	
t^3	0	-V	0	V	
t^2	0	1+2V	-2-V	1-V	
t	0	-2-V	2+V	0	<--- Start Tangent
1	0	1	0	0	

We start with an analysis of the cubic Bézier. The first segment starts on P_0 and stops on P_3. The next segment starts on P_3 and stops on P_6. Similar to the end position analysis, the bottom row of the derivative matrix gives the start tangent and the sum of columns gives the stop tangent. From the arrangement shown at the right, we see that the first segment's stop tangent is determined by points P_2 & P_3 (the -3 3 in the SUM row). However, the second segment's start tangent is determined by points P_3 & P_4 (the -3 3 in the bottom row). Because these two end tangents are determined by different control points, they are not equal. Therefore, the Cubic Bézier does not have C^1 continuity at the joints. There is no need to look at the second derivative.

For analyzing the B-spline, we can temporarily omit the 1/6 factor. The first segment starts near P_1 and stops near P_2. The next segment starts near P_2 and stops near P_3. From the arrangement shown at the right, we see that the first segment's stop tangent is determined by points P_1 & P_3 (from the -3 0 3 0 in the SUM row). The second segment's start tangent is also determined by points P_1 & P_3 (from the 0 -3 0 3 in the bottom row). Because these two end tangents are determined by the same control points and the weighting function values are the same, they are equal. Therefore, the B-spline has C^1 continuity at the joints. The second derivative can be analyzed similarly to show C^2 continuity.

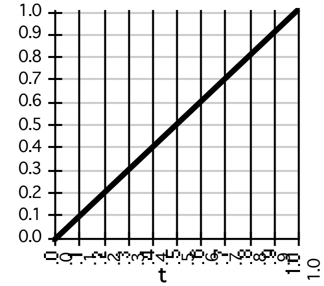
This same kind of analysis can be done on the variable interpolation types as well. The Quadratic Bézier family W/ End velocity control is shown on the right. We see that the end tangent of segment #1 is determined by P_2 & P_3, but the start tangent of segment #2 is determined by P_3 & P_4. Because these two end tangents are determined by different control points, they are not equal. Therefore, it does not have C^1 continuity at the joints.


```
Cubic Bézier derivative
                    P0   P1   P2   P3
          t^3 |   0    0   0    0 |
          t^2 |  -3    9  -9    3 |
          t   |   6  -12   6    0 |
          1   |  -3    3   0    0 |<--- Start tangent

      SUM->   0    0  -3    3  <--- Segment #1 stop tangent
                       |    |
                           -3   3   0   0 <--- Segment #2 start
tangent
                       |    |   |
                    P0   P1   P2   P3   P4   P5   P6


Cubic B-spline derivative
                    P0   P1   P2    P3
          t^3 |   0     0    0    0 |
          t^2 |  -3     9   -9    3 |  1/6
          t   |   6   -12    6    0 |
          1   |  -3     0    3    0 |     <--- Start Tangent

      SUM->   0    -3    0    3  <--- Segment #1 stop tangent
                    |         |
                   -3    0    3    0 <--- Segment #2 start tangent
                    |         |
                    P0   P1    P2   P3   P4   P5   P6


Quadratic Bézier Family W/End Velocity Control
                         P1    P2    P3
            t^3 |   -V     0     V  |
            t^2 |  1+2V  -2-V   1-V |
            t   |  -2-V   2+V    0  |    <--- Start Tangent
            1   |    1     0     0  |    <--- Start location

        SUM->      0      0     1      <--- Stop location


First derivative
                         P1    P2     P3
            t^2 |   -3V     0     3V  |
            t   |  2+4V  -4-2V  2-2V  |
            1   |  -2-V   2+V    0    |    <--- Start Tangent

        SUM->      0   -2-V   2+V      <--- Segment #1 stop tangent
                         |     |
                              -2-V    2+V     0 <--- Segment #2 start
                         |     |      |
                         P1    P2     P3     P4     P5
```

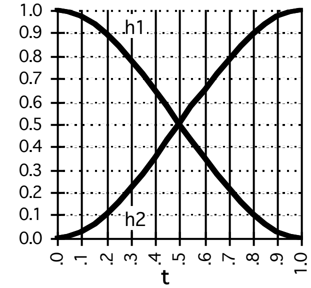
Appendix A10 Component Functions

Functions that have desirable characteristics can be considered as fundamental building blocks. They can be used for constructing and modifying quadratic and cubic weighting functions. They are shown with the values of the functions and derivatives at the start (t=0) and stop (t=1).

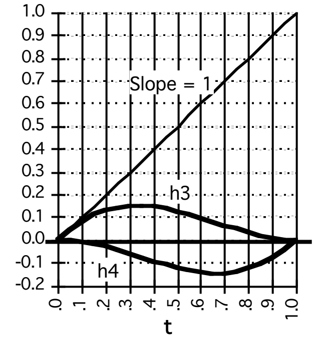
	f(0)	f(1)	f'(0)	f'(1)	f"(0)	f"(1)
Linear t	0	1	1	1	0	0
Hermite h1 $+2t^3-3t^2+1$	1	0	0	0	-6	6
Hermite h2 $-2t^3+3t^2$ Also = Linear - "S"	0	1	0	0	6	-6
Hermite h3 $+1t^3-2t^2+t$	0	0	1	0	-4	-2
Hermite h4 $+1t^3 -1t^2$	0	0	0	1	-2	4
"S" $+2t^3-3t^2+t$ Also = h3 + h4	0	0	1	1	-6	6
"Center" $-t^2+t$ Also = h3 - h4	0	0	1	-1	-2	-2
quintic 1 $-6t^5+15t^4-10t^3+1$	1	0	0	0	0	0
quintic 2 $+6t^5-15t^4+10t^3$	0	1	0	0	0	0
quintic 3 $-3t^5+8t^4-6t^3+t$	0	0	1	0	0	0
quintic 4 $-3t^5+7t^4-4t^3$	0	0	0	1	0	0
quintic 5 $-0.5t^5+1.5t^4-1.5t^3+0.5t^2$	0	0	0	0	1	0
quintic 6 $0.5t^5-t^4+0.5t^3$	0	0	0	0	0	1



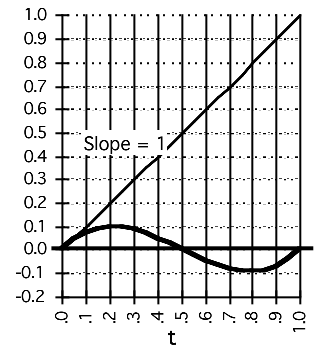
Linear f(t) = t



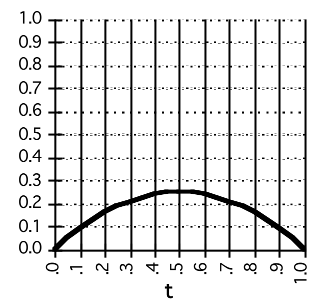
Hermite h1 f(t)=+2t3-3t2+1
Hermite h2 f(t)=-2t3+3t2



Hermite h3 f(t)=+1t3-2t2+t
Hermite h4 f(t)=+1t3-1t2



"S" Function f(t)=+2t3-3t2+t



"Center" Function f(t) = -t2 + t

Quintic graphs to be added

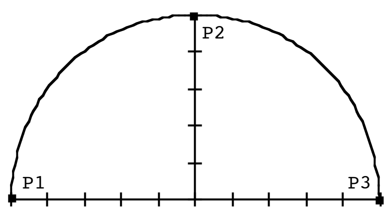
Appendix A11 Circle Approximations

Four polynomial methods are shown that can be used to approximate a circle. Ellipses are obtained by skewing the control points as appropriate.

Three Control Point Version

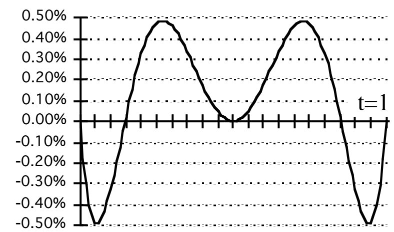
The first curve, using the quadratic Golden with cubic tension added, produces a half circle with less than 0.5% (5000 ppm) radius error and only requires three control points. The interpolated points are not uniformly spaced around the circle, but are slightly closer together at the center control point. The three control points are at the arc center and ends. The value of tension was chosen empirically to minimize the peak radius error.

Golden W/Tension



Radius Error

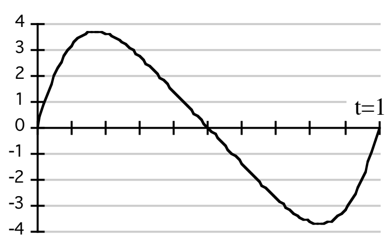
Tension -0.899



Basis Matrix

$$\begin{array}{c|ccc|}
 & P_1 & P_2 & P_3 \\
t^3 & -2T & 0 & 2T \\
t^2 & 2+3T & -4 & 2-3T \\
t & -3-T & 4 & T-1 \\
1 & 1 & 0 & 0
\end{array} \quad \text{Where } T = -0.899$$

Angle error (deg.) vs. t

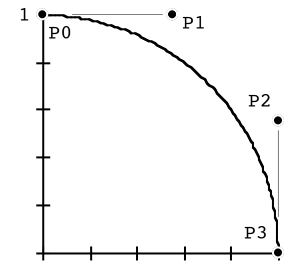


This plot shows the error of the actual location of the curve relative to an angle proportional to the parameter.

Cubic Bézier Version

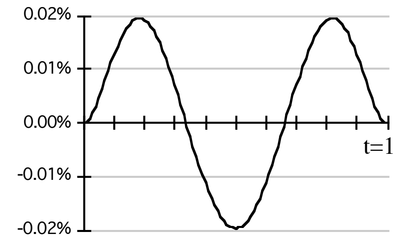
The second method, using the cubic Bézier, produces a quarter circle with less than 0.02% (200ppm) radius error and requires four control points. This approximation was derived to place the arc mid point and ends on the circle, however the radius is greater everywhere else. By applying a correction factor to reduce the value of "k", the peak radius error and angle error are reduced. The interpolated points are not uniformly spaced along the arc, but are slightly closer together at the arc ends.

Cubic Bézier



Bézier Radius Error

With correction

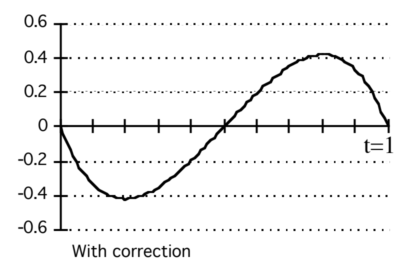


The standard cubic Bézier basis matrix is used, but the location of control points for a unit circle is as follows:

	P_0	P_1	P_2	P_3
X	0	k*c	1	1
Y	1	1	k*c	0

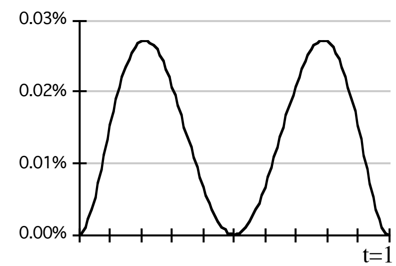
Where $k = 4(\sqrt{2} - 1)/3$
and correction factor c = 0.99933
Uncorrected c = 1

Angle error (deg.) vs. t



Bézier Radius Error

Without correction



This plot shows the error of the actual location of the curve relative to the angle proportional to the parameter.

Hermite Method

The third method, using Hermite functions, is a generalized form for an arc defined by half the angle between the arc end points.

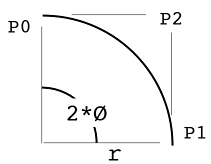
P_0 & P_1 are the arc end points.
P_2 is the point where the two tangents from P_0 & P_1 meet -
For the quarter circle in the first quadrant:
$P_0 = (0,1)$ $P_1 = (1,0)$ $P_2 = (1,1)$

Ø = theta = half the angle between the two end points P_0, P_1.
r = radius.
H = is the Hermite Basis functions (this is with the parameter) = |h1 h2 h3 h4|
He also appears to say that the center point of the curve, p(0.5), is on the circle.

The geometry vector in the standard Hermite form:

$$
p = H * \left|\begin{array}{c} P_0 \\ P_1 \\ (P_2-P_0)*\dfrac{4*\mathrm{Cos}(th)}{1+\mathrm{Cos}(th)} \\ (P_1-P_2)*\dfrac{4*\mathrm{Cos}(th)}{1+\mathrm{Cos}(th)} \end{array}\right| \begin{array}{c} P_0 \\ P_1 \\ T_0 \\ T_1 \end{array}
$$

For a quarter circle (Ø=45°) $\dfrac{4*\mathrm{Cos}(th)}{1+\mathrm{Cos}(th)} = 4(\sqrt{2} - 1)$



Hermite arc diagram

This version was derived with a geometric construction described in Mortenson who reports that is has less then 10^{-6} error from the circle, but an analysis by the author shows it to be 2 x 10^{-4}. This version at a quarter circle (Ø= 45°) is equivalent to the Cubic Bézier. They both produce equivalent representations of the same polynomial. They produce a quarter circle and require four control points. The whole circle is obtained by mirroring the coordinates. The

correction of 0.99933 can be applied to the tangents to obtain reduced error just as in the Bézier version above.

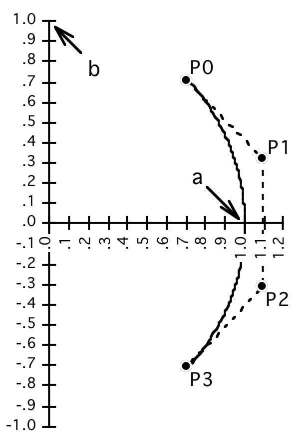
Zero Crossing Method

The fourth method, using the cubic Bézier, is based on the zero crossings of the circle. This version defines segments that join at the 45° points. This version was derived using the cubic Bézier subdivision equations (equations 9.13 through 9.18). It produces a quarter circle with less than 0.02% (200ppm) radius error and requires four control points. This approximation was derived to place the arc mid point and ends on the circle, however the radius is greater everywhere else. The interpolated points are not uniformly spaced along the arc, but are slightly closer together at the arc ends. These characteristics are the same as shown for the Cubic Bézier version above.

The X axis crossing is "a" and the Y axis crossing is "b".

	X	Y
P_0	$a/\sqrt{2}$	$b/\sqrt{2}$
P_1	$a*(8-\sqrt{2})/6$	$b*(7*\sqrt{2}-8)/6$
P_2	$a*(8-\sqrt{2})/6$	$-b*(7*\sqrt{2}-8)/6$
P_3	$a/\sqrt{2}$	$-b/\sqrt{2}$

Cubic Bézier



Insert the random arc approximation using Bezier.

Appendix A12 Basis Matrices, Fixed Types

This appendix contains basis matrices for fixed interpolation types. Included with each is information on the curve behavior, the inverse matrix, weighting function plots, and samples of the interpolated curve. For a few types, the building block component functions are given.

For interpolating long lists of control points, the start, step and end points are given so that segments connect end to end. Each basis matrix is shown in a 4x4 format because they have been extracted from a program that uses a common block of code to draw all of the curve types so they can be studied and compared. Because of this, references to the start and end point should be closely examined. Sometimes the first point used is called P_0 and sometimes it is called P_1. Types that do not have a 4x4 basis matrix usually start on the point called P_1. For these types P_0 may not exist.
The rows, columns and entries of any specific type that are zero can be omitted from the calculation. Some inverse matrices are shown their normal size. *Only some types have inverses and not all inverses are shown here.*

For the sample curves, the number of points per segment, the number of segments shown and the arrangement of control points_ were selected to best show the characteristics of each type. *Note that some of the interpolated points can be obscured by the larger control points.*

Fixed Interpolation Types

- Linear
- Simple Cubic
- Parabolic (quadratic B-spline)
- Quadratic Bézier
- Quadratic Three Point (Golden)
- Cubic Bézier
- Cubic Four Point
- Hermite
- B-spline
- Catmul-Rom
- Quintic Super Segment
    - Improved Acceleration Factor
    - Equal Factors
    - Variable Factors
    - Cubic Bézier-like Acceleration
    - Hermite-like Form

Linear

Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) is constant within each segment. The "curve" passes through both control points. This is not a spline.

Basis Matrix

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 0 & 0 & 0 \\ t^2 & 0 & 0 & 0 & 0 \\ t & 0 & -1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

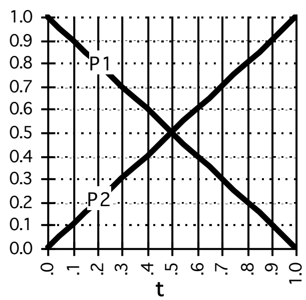
Inverse Matrix

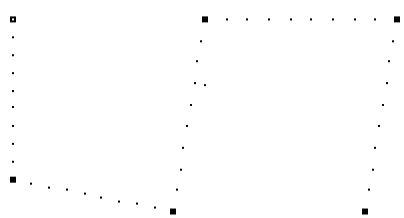
$$\begin{vmatrix} 0 & 1 \\ 1 & 1 \end{vmatrix}$$

As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the first control point P_1.
- Step one at a time.
- End at next to last.

This type needs two control points to interpolate a segment. One segment spans those two control points.





5 segments with 10 points per segment. End points are covered by the control points.

Simple Cubic

Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) decreases to zero at the control points. The "curve" passes through both control points. This is not a spline. The weighting functions are the Hermite functions h1 & h2.

Derived from source code found on the net written by Toby Orloff & Jim Larson. U. of Minn. Geometry Supercomputer Project "omni_interp".

Basis Matrix

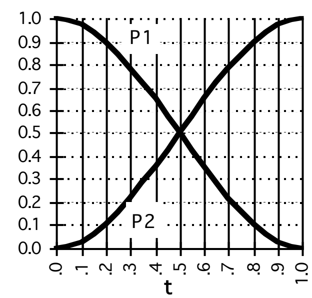
No inverse matrix exists

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & +2 & -2 & 0 \\ t^2 & 0 & -3 & +3 & 0 \\ t & 0 & 0 & 0 & 0 \\ 1 & 0 & +1 & 0 & 0 \end{array}$$

As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the first control point P_1.
- Step one at a time.
- End at next to last.

This type needs two control points to interpolate a segment. One segment spans those two control points. The first derivative and therefore the velocity is always zero at control points. This curve is a special case of the three variable Simple Cubic types with slope control.



5 segments with 8 new points per segment. End points are covered by the control points.

Component functions: P_1 = h1; P_2 = h2

Quadratic B-spline / Parabolic

Produces interpolated points that are on a curve that goes from the midpoint of line P_0P_1 to the midpoint of line P_1P_2 where it is tangent to those lines. The curve does not pass through any control points. This is a spline because the first derivatives are equal at the joints and it is a second degree function. This is the quadratic B-spline
Provided by Leon de Boer.

Basis Matrix

$$\begin{array}{c|cccc|c} & P_0 & P_1 & P_2 & P_3 & \\ \hline t^3 & 0 & 0 & 0 & 0 & \\ t^2 & 1 & -2 & 1 & 0 & 1 \\ t & -2 & 2 & 0 & 0 & \overline{} \\ 1 & 1 & 1 & 0 & 0 & 2 \end{array}$$

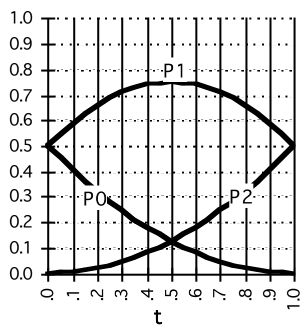
Inverse Matrix

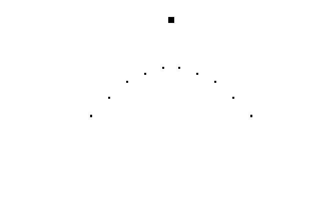
$$\begin{vmatrix} 0 & -1 & 2 \\ 0 & 1 & 2 \\ 4 & 3 & 2 \end{vmatrix} \frac{1}{2}$$

As t goes from 0 to 1 the curve goes from the midpoint of line P_0P_1 to the midpoint of line P_1P_2 .

Piecewise loop parameters:
 Start at the second control point P_1.
 Step one at a time.
 End at next to last.

This type needs three control points to interpolate a segment. The curve spans the two midpoints. For the curve to intercept the end points, there must be two co-located points.

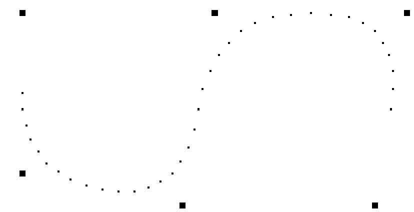




1 segment with 10 points per segment.
This includes the points at t = 0 and t = 1.

Component functions: P_1 = 0.5 + Center Function

Quadratic B-spline / Parabolic



4 segments with 10 points per segment. This includes the two points at t = 0 and t = 1. Since the end points of each segment coincide, there are 37 total unique points.

Quadratic Bézier

Produces interpolated points that are on a curve starting on P_1 and ending on P_3. At P_1 the curve is tangent to the line P_1P_2. At P_3 the curve is tangent to the line P_2P_3 . The result of this is that it appears that the curve goes "near" or is attracted to P_2. This is not a spline.

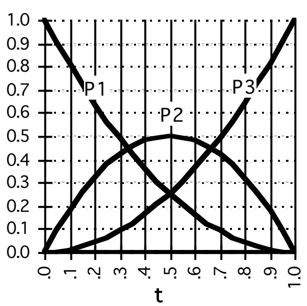
Basis Matrix — Inverse Matrix

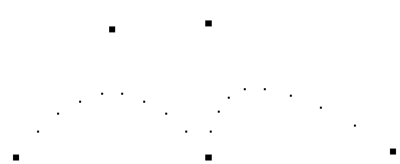
$$\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}\begin{array}{c} \begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \end{array} \\ \left|\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & -2 & 2 & 0 \\ 0 & 1 & 0 & 0 \end{array}\right| \end{array} \qquad \left|\begin{array}{ccc} 0 & 0 & 2 \\ 0 & 1 & 2 \\ 2 & 2 & 2 \end{array}\right| \frac{1}{2}$$

As t goes from 0 to 1 the curve goes from P_1 passing near P_2 to P_3

Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

This type needs three control points to interpolate a segment. The segment spans the three control points.





2 segments with 8 new points per segment. End points are covered by the control points.

Component functions: P_2 = 2 * Center function

Quadratic Three Point (Golden)

Produces interpolated points that are on a curve through P_1, P_2 and P_3. The center point P_2 is at t = 0.5 on the curve. This is not a spline. Derived by Sean Palmer and obtained in private e-mails.

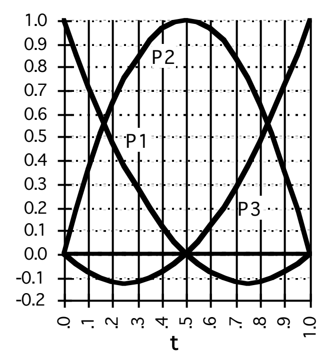
Basis Matrix — Inverse Matrix

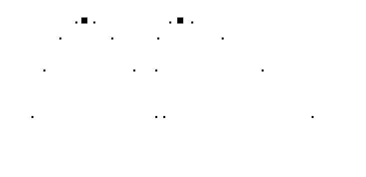
$$\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}\begin{array}{c} \begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \end{array} \\ \left|\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 2 & -4 & 2 \\ 0 & -3 & 4 & -1 \\ 0 & 1 & 0 & 0 \end{array}\right| \end{array} \qquad \left|\begin{array}{ccc} 0 & 0 & 4 \\ 1 & 2 & 4 \\ 4 & 4 & 4 \end{array}\right| \frac{1}{4}$$

As t goes from 0 to 1 the curve goes from P_1 through P_2 to P_3

Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

This type needs three control points to interpolate a segment. The segment spans the three control points. This is also the quadratic Lagrange with t2 = 0.5.





2 segments with 8 new points per segment. End points are covered by the control points.

Component functions: P_2 = 4 * Center function
P_3 = linear -2*h3 + 2*h4

Cubic Bézier

Produces interpolated points that are on a curve starting on P_0 and ending on P_3. At P_0 the curve is tangent to the line P_0P_1. At P_3 the curve is tangent to the line P_2P_3. The result of this is that it appears that the curve goes near or is drawn toward P_1 and P_2. This is not a spline.

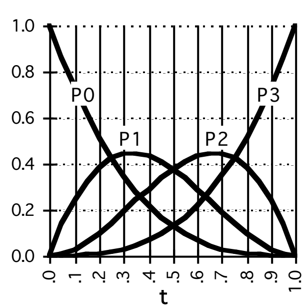
Basis Matrix — Inverse Matrix

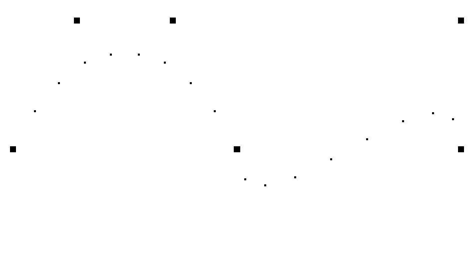
$$
\begin{array}{c}
\\ t^3 \\ t^2 \\ t \\ 1
\end{array}
\begin{array}{cccc}
P_0 & P_1 & P_2 & P_3 \\
-1 & 3 & -3 & 1 \\
3 & -6 & 3 & 0 \\
-3 & 3 & 0 & 0 \\
1 & 0 & 0 & 0
\end{array}
\qquad
\left|\begin{array}{cccc}
0 & 0 & 0 & 3 \\
0 & 0 & 1 & 3 \\
0 & 1 & 2 & 3 \\
3 & 3 & 3 & 3
\end{array}\right| \frac{1}{3}
$$

As t goes from 0 to 1 the curve goes from P_0, passes near P_1 and P_2 to P_3.

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

This type needs four control points to interpolate a segment. The segment spans the four control points.





2 segments with 8 new points per segment. End points are covered by the control points.

Component functions:
P_1 = 3*h3
P_2 = -3*h4
P_3 = linear - h3 + 2*h4

Cubic Four Point

Produces interpolated points that are on a curve passing through P_0, P_1, P_2 and P_3. This is not a spline.

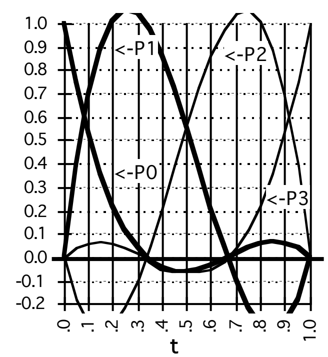
Basis Matrix — Inverse Matrix

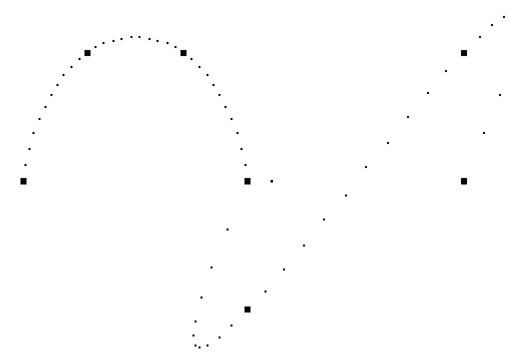
$$
\begin{array}{c}
\\ t^3 \\ t^2 \\ t \\ 1
\end{array}
\begin{array}{cccc}
P_0 & P_1 & P_2 & P_3 \\
-9 & 27 & -27 & 9 \\
18 & -45 & 36 & -9 \\
-11 & 18 & -9 & 2 \\
2 & 0 & 0 & 0
\end{array}
\frac{1}{2}
\qquad
\left|\begin{array}{cccc}
0 & 0 & 0 & 27 \\
1 & 3 & 9 & 27 \\
8 & 12 & 18 & 27 \\
27 & 27 & 27 & 27
\end{array}\right| \frac{1}{27}
$$

As t goes from 0 to 1 the curve goes from P_0, through P_1 and P_2 to P_3.

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

This type needs four control points to interpolate a segment. The segment spans the four control points. This is also the quadratic Lagrange with t1 = 1/3 and t2 = 2/3.





2 segments with 32 new points per segment End points are covered by the control points and two coincide with the inner control points).

Hermite

Produces interpolated points that are on a curve starting on P_0 and ending on P_3. T_0 is the tangent vector at P_0 in Cartesian form (X_1, Y_1, Z_1). T_3 is the slope or tangent vector at P_3 in Cartesian form (X_2, Y_2, Z_2). T_0 and T_3 are not part of the image, but are tangent vectors relative to zero. This is not a spline. The end slopes are given directly by T_0 and T_3. The first format accentuates the similarity to the cubic Bézier. It differs from what is usually shown for the Hermite (see below) where T_0 & T_3 are the right two columns.

Basis Matrix

	h1	h3	h4	h2
	P_0	T_0	T_3	P_3
t^3	2	1	1	-2
t^2	-3	-2	-1	3
t	0	1	0	0
1	1	0	0	0

As t goes from 0 to 1 the curve goes from P_0 to P_3

Piecewise loop parameters:

Start at the first control point P_0.
Step three at a time.
End three from the last.

This type needs four control points to interpolate a segment. The segment spans two of the control points (P_0 P_3). T_1 and T_2 are the end tangent values. They are absolute values and are not relative to the end points. This arrangement was developed to allow it to fit in a piecewise interpolation loop like the other interpolation types.
This arrangement can be related to the cubic Bézier. On the cubic Bézier, the central two control points indirectly determine the end point tangents, relative to the end points.

This is the more common format:

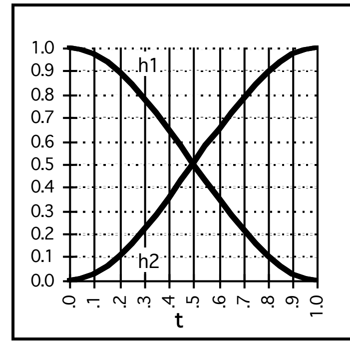
	h1	h2	h3	h4
	P_0	P_1	T_0	T_1
t^3	2	-2	1	1
t^2	-3	3	-2	-1
t	0	0	1	0
1	1	0	0	0

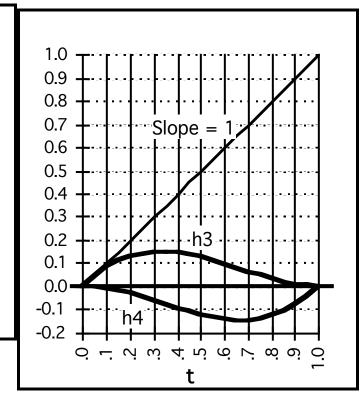
As t goes from 0 to 1 the curve goes from P_0 to P_1

Piecewise loop parameters:

N/A
This arrangement does not easily fit into a piecewise loop like the arrangement above.

Hermite





Add curves

Cubic B-Spline

Produces interpolated points that pass near the control points. This is a spline.

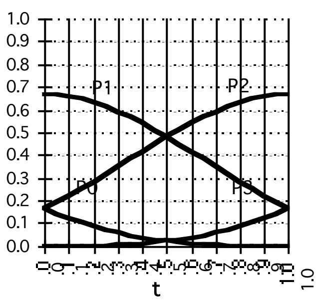
Basis Matrix — Inverse Matrix

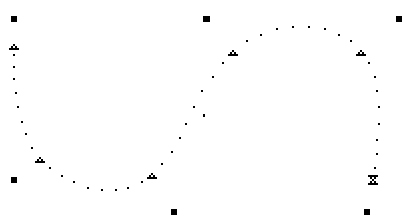
$$\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{array} \;\; \frac{1}{6} \qquad \begin{array}{cccc} 0 & 2 & -3 & 3 \\ 0 & -1 & 0 & 3 \\ 0 & 2 & 3 & 3 \\ 18 & 11 & 6 & 3 \end{array} \;\; \frac{1}{3}$$

As t goes from 0 to 1 the curve goes from near P_1 to near P_2

Piecewise loop parameters:
- Start at the second control point P_1.
- Step one at a time
- End two from the last

This type needs four control points to interpolate a segment. The segment spans the two inner control points although it does not pass through them. The end points are repeated to make the curve intercept them.





5 segments with 10 points per segment. This includes the two points at t = 0 and t = 1. Since the end points of each segment coincide, there are 46 total points. The joint points are marked.

Catmul-Rom

Produces interpolated points that are on a smooth curve through all control points.

Basis Matrix — Inverse Matrix

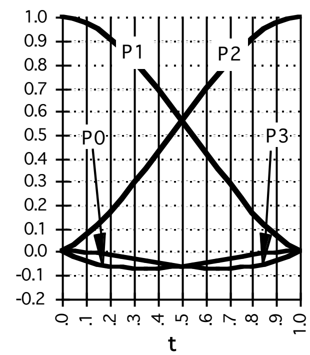
$$\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{array} \;\; \frac{1}{2} \qquad \begin{array}{cccc} 1 & 1 & -1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 6 & 4 & 2 & 1 \end{array} \;\; \frac{1}{1}$$

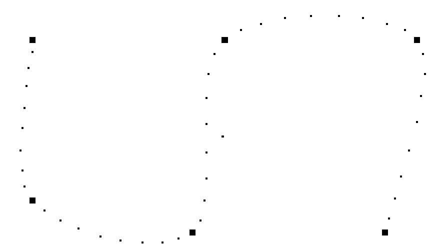
As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the second control point P_1.
- Step one at a time
- End two from the last

This type needs four control points to interpolate a segment. The segment spans the two inner control points. The slope at any control point is parallel to the line connecting the two adjacent control points. The end points are repeated to make the curve intercept them.

Add non-end repeat curve here

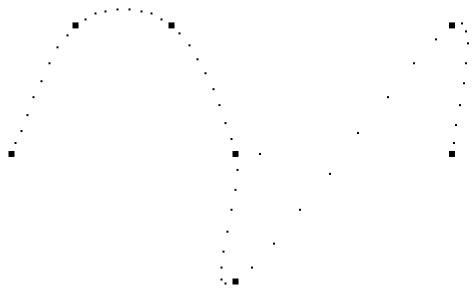




5 segments with 8 new points per segment.

Component Functions:
P_0 = -h3
P_1 = 1-linear + h3 + 0.5*h4
P_2 = linear - 0.5*h3 - h4
P_3 = h4

Catmul-Rom



6 segments with 8 new points per segment.

Quintic Super segment

Improved Acceleration Factor
Produces interpolated points that are on a curve starting on P_0 and ending on P_5. The characteristics of this curve parallel the cubic Bezier with the addition of two control points that control the second derivative (acceleration) at the joints. Points P_1 and P_4 are the end tangent (velocity) control points just as in the cubic Bézier. At P_0 the curve is tangent to the line P_0P_1. At P_5 the curve is tangent to the line P_4P_5. Points P_2 and P_3 are the end acceleration control points and are manipulated like the tangent control points. At P_0 the acceleration is tangent to the line P_0P_2. At P_5 the acceleration is tangent to the line P_3P_5.

The tangents (velocities) at the ends are given by:
$V_0 = 3(P_1 - P_0)$ $V_5 = 3(P_5 - P_4)$

The accelerations at the ends are given by:
$A_0 = 24(P_2 - P_0)$ $A_5 = 24(P_5 - P_3)$

The result of this is that it appears that the curve goes near or is drawn toward P_1 and P_4. This is not a spline.

Basis Matrix V=3, A=24

$$\begin{array}{c|rrrrrr|c}
 & P_0 & P_1 & P_2 & P_3 & P_4 & P_5 & \\
t^5 & 15 & -9 & -12 & 12 & 9 & -15 & \\
t^4 & -45 & 24 & 36 & -24 & -21 & 30 & 1 \\
t^3 & 44 & -18 & -36 & 12 & 12 & -14 & --- \\
t^2 & -12 & 0 & 12 & 0 & 0 & 0 & 1 \\
t & -3 & 3 & 0 & 0 & 0 & 0 & \\
1 & 2 & 0 & 0 & 0 & 0 & 0 &
\end{array}$$

As t goes from 0 to 1 the curve goes from P_0, passes near P_1 and P_4 and stops at P_5. The end tangents are equal to three times the spacing between the end point and the respective control point just as in the cubic Bézier. The end accelerations are equal to 24 times the spacing between the end point and the respective control point. This factor was chosen to keep the acceleration control points near the respective end points (compared to the 3x version shown below).

Piecewise loop parameters:
 Start at the first control point P_0.
 Step five at a time
 End five from the last

This type needs six control points to interpolate a segment. The segment spans the six control points.

Quintic Super segment

Equal factors

In the following variation, the acceleration is three times the difference between the acceleration control point (P_1P_4) and the respective end point.

The tangents (velocities) at the ends are given by:
$V_0 = 3(P_1 - P_0)$ $V_5 = 3(P_5 - P_4)$

The accelerations at the ends are given by:
$A_0 = 3(P_2 - P_0)$ $A_5 = 3(P_5 - P_3)$

Basis Matrix V=3, A=3

	P_0	P_1	P_2	P_3	P_4	P_5	
t^5	9	-18	-3	3	18	-9	
t^4	-27	48	9	-6	-42	18	1
t^3	25	-36	-9	3	24	-7	---
t^2	-3	0	3	0	0	0	2
t	-6	6	0	0	0	0	
1	2	0	0	0	0	0	

As t goes from 0 to 1 the curve goes from P_0, passes near P_1 and P_4 and stops at P_5. The end tangents are equal to three times the spacing between the end point and the respective control point just as in the cubic Bézier. The end accelerations are also equal to 3 times the spacing between the end point and the respective control point. This factor was initially chosen to be the same as the velocity factor, but graphically the acceleration control points wind up rather far from the end points for reasonable changes in acceleration.

Piecewise loop parameters:
 Start at the first control point P_0.
 Step five at a time
 End five from the last

This type needs six control points to interpolate a segment. The segment spans the six control points.

Needs sample curves.

Quintic Super segment

Variable factors

The following variation is the more general form. The acceleration factor and velocity factor are variables. This implementation allows the user to adjust the factors A and V to place the graphics control points as desired.

Basis Matrix

	P_0	P_1 (V_0)	P_2 (A_0)	P_3 (A_5)	P_4 (V_5)	P_5	
t^5	-12 +6V +A	-6V	-A	A	6V	12 -6V -A	
t^4	30-16V-3A	16V	3A	-2A	-14V	-30+14V+2A	1
t^3	-20+12V+3A	-12V	-3A	A	8V	20 -8V -A	---
t^2	-A	0	A	0	0	0	2
t	-2V	2V	0	0	0	0	
1	2	0	0	0	0	0	

As t goes from 0 to 1 the curve goes from P_0, passes near P_1 and P_4 and stops at P_5. The end tangents are equal to the spacing between the end point and the respective control point times "V". In the cubic Bézier this factor is three. The end accelerations are equal to the spacing between the end point and the respective control point times "A". A reasonable range for A is 6 to 30. By making the As and Vs under the left three control points different from those under the right three control points, independant factors are obtained for each end.

The tangents (velocities) at the ends are given by:
$V_0 = V(P_1 - P_0)$ $V_5 = A(P_5 - P_4)$

The accelerations at the ends are given by:
$A_0 = A(P_2 - P_0)$ $A_5 = A(P_5 - P_3)$

Piecewise loop parameters:
 Start at the first control point P_0.
 Step five at a time
 End five from the last

This type needs six control points to interpolate a segment. The segment spans the six control points.

Cubic Bézier-like acceleration

If it is desired to start with a segment equivalent to the cubic Bézier, use a velocity factor of 3 and place the acceleration control points as follows.

$P_2 = [\ (6+A)P_0 - 12P_1 + 6P_4\ ]\ /\ A$

$P_3 = [\ (6+A)P_5 - 12P_4 + 6P_1\ ]\ /\ A$

Quintic Super segment

Hermite-like form
In the following variation, the velocity and acceleration are given directly by the internal control values as in the Hermite.
The tangents (velocities) at the ends are given by:
V_0 = Start tangent (velocity). V_5 = End tangent (velocity).

The accelerations at the ends are given by:
A_0 = Start acceleration. A_5 = End acceleration.

Basis Matrix

$$
\begin{array}{c}
 \\ t^5 \\ t^4 \\ t^3 \\ t^2 \\ t \\ 1
\end{array}
\begin{array}{c}
\begin{array}{rrrrrr} P_0 & V_0 & A_0 & A_5 & V_5 & P_5 \end{array} \\
\left|\begin{array}{rrrrrr}
-12 & -6 & -1 & 1 & -6 & 12 \\
30 & 16 & 3 & -2 & 14 & -30 \\
-20 & -12 & -3 & 1 & -8 & 20 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 2 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0
\end{array}\right|
\end{array}
\frac{1}{2}
$$

As t goes from 0 to 1 the curve goes from P_0 to P_5. The end tangents (velocities) are equal to V_0 and V_5 just as in the Hermite. The end accelerations are equal to A_0 and A_5

Piecewise loop parameters:
- Start at the first control point P_0.
- Step five at a time
- End five from the last

This type needs two control points to interpolate a segment. The segment spans the two control points. Four additional controls are needed for each segment to define the end tangents (velocities) and accelerations.

Appendix A13 Basis Matrices, Variable Types

This appendix contains basis matrices for variable interpolation types. These types contain additional parameters that can be varied to alter the shape of the curve without moving the control points. Included with each is information on the curve behavior, the effect of the variable parameter and samples of the interpolated curve. The inverse matrix and weighting function plots are not shown.
Each basis matrix is shown in a 4x4 format because they have been extracted from a program that uses a common block of code to draw all of the curve types so they can be studied and compared. Because of this, references to the start and end point should be closely examined. Sometimes the first point required for a particular curve is called P_0 and sometimes it is called P_1. Types that do not have a 4x4 basis matrix usually start on the point called P_1. For these types P_0 may *or may not* exist.
The rows, columns and entries of any specific type that are zero can be omitted from the calculation. –
For interpolating long lists of control points, the start, step and end points are given so that segments connect end to end.

For the sample curves, the number of points per segment, the number of segments shown and the arrangement of control points were selected to best show the characteristics of each type. *Note that some of the interpolated points can be obscured by the larger control points.*

Variable Interpolation Types

- Simple Cubic W/ End Slope Control
- Simple Cubic W/ Midpoint Slope Control
- Simple Cubic W/ Independent End Slope Control
- Quadratic Bézier Family W/End Velocity Control
- Quadratic Lagrange W/Bias
- Cubic Lagrange W/Two Bias Controls
- Beta Spline W/Tension & Bias Controls
- Kochanek-Bartles W/Tension, Bias & Continuity Controls
- Cubic Bézier Family W/End Velocity Control
- Quadratic three point W/Bias & Tension Control (Palmer)

Simple Cubic with End Slope Control

Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) goes to the value of S_0 at the control points. The curve passes through both control points. This is not a spline.

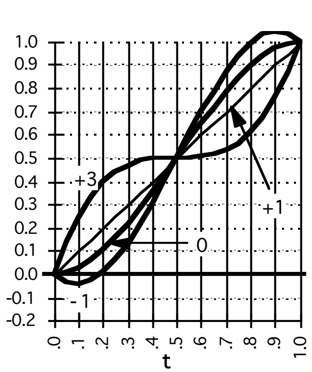
Basis Matrix   No inverse matrix exists

$$
\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}
\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ \hline
0 & 2-2S_0 & 2S_0-2 & 0 \\
0 & 3S_0-3 & 3-3S_0 & 0 \\
0 & -S_0 & S_0 & 0 \\
0 & 1 & 0 & 0
\end{array}
$$

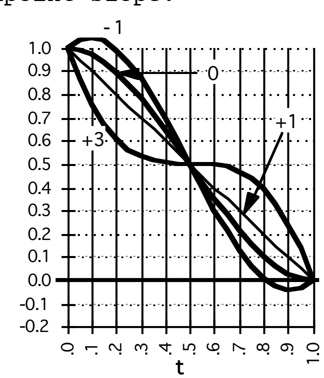
As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the first control point P_1.
- Step one at a time.
- End at next to last.

This type only needs two control points to interpolate a segment. One segment spans those two control points. The join points will connect, and the first derivative/velocity at the control points is specified by S_0. Because of this, the curve can over shoot the second control point in a segment ($S_0 > 9$) or start headed away from the second control point ($S_0 < 0$). For a value of $S_0 = 3$ the velocity drops to zero half way between control points. Please note that the values of S_0 shown for these weighting function plots were not selected to give all the same weighting function plots as shown for the simple-cubic-with-variable-midpoint-slope.



P_2 weighting Function

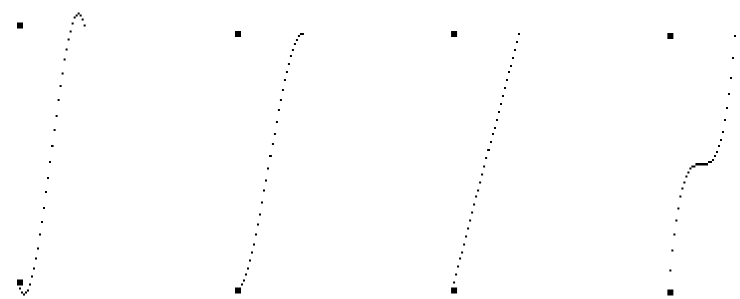


P_1 weighting Function

With various values of S_0

Simple Cubic with End Slope Control

Note: Because this type interpolates in a straight line between control points, the interpolated points can double back on themselves. The weighting function plots give a good indication of the curve path. To help visualize the curves, these plots are displaced to the right as t goes from 0 to 1 to eliminate the overlap.



$S_0 = -1$ $S_0 = 0$ & Simple Cubic $S_0 = 1$ & linear $S_0 = 3$

1 segment with 32 new points per segment.

Simple Cubic with Midpoint Slope Control

Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) goes to the value of S_m at the midpoint between control points. The curve passes through both control points. This is not a spline.

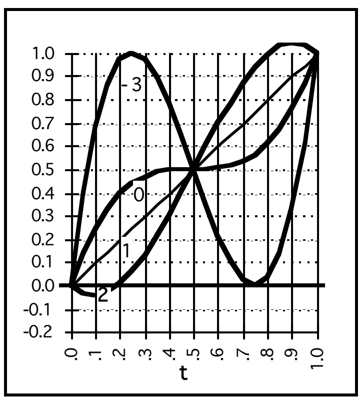
Basis Matrix   No inverse matrix exists

	P_0	P_1	P_2	P_3
t^3	0	$4S_m-4$	$4-4S_m$	0
t^2	0	$6-6S_m$	$6S_m-6$	0
t	0	$2S_m-3$	$3-2S_m$	0
1	0	1	0	0

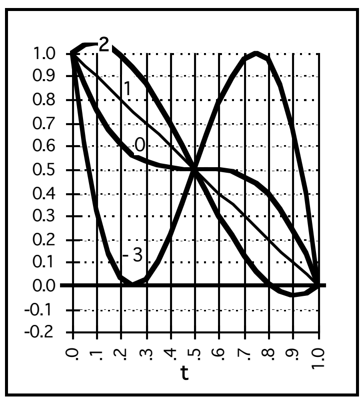
As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the first control point P_1.
- Step one at a time.
- End at next to last.

This type only needs two control points to interpolate a segment. One segment spans those two control points. The join points will connect, and the first derivative/velocity at the midpoint of the segment is specified by S_m. Because of this, the curve can over shoot the second control point in a segment ($S_m < -3$) or start headed away from the second control point ($S_m > 1$). For a value of $S_m = 0$ the velocity drops to zero half way between control points.



P_2 weighting Function

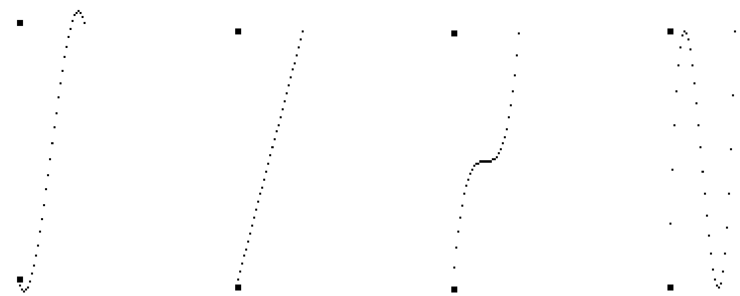


P_1 weighting Function

With various values of S_m

Simple Cubic with Midpoint Slope Control

Note: Because this type interpolates in a straight line between control points, the interpolated points can double back on themselves. The weighting function plots give a good indication of the curve path. To help visualize the curves, these plots are displaced to the right as t goes from 0 to 1 to eliminate the overlap.



$S_m = 2$ $S_m = 1$ & linear $S_m = 0$ $S_m = -3$

1 segment with 32 new points per segment.

Simple Cubic with Independent End Slope Control

Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) goes to the value of S_0 at the segment start and S_1 at the segment finish. The curve passes through both control points. This is not a spline.

Basis Matrix　　　　No inverse matrix exists

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 2-S_0-S_1 & S_0+S_1-2 & 0 \\ t^2 & 0 & 2S_0+S_1-3 & 3-2S_0-S_1 & 0 \\ t & 0 & -S_0 & S_0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

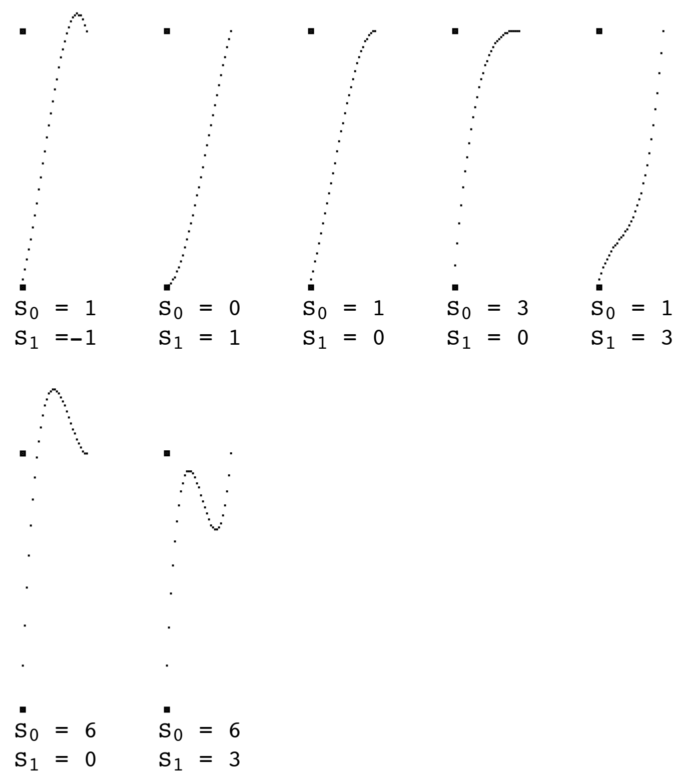
As t goes from 0 to 1 the curve goes from P_1 to P_2

Piecewise loop parameters:
- Start at the first control point P_1.
- Step one at a time.
- End at next to last.

This type only needs two control points to interpolate a segment. One segment spans those two control points. The join points will connect, and the first derivative/velocity at the start of the segment is specified by S_0 and the first derivative/velocity at the finish of the segment is specified by S_1.

Simple Cubic with Independent End Slope Control

Note: Because this type interpolates in a straight line between control points, the interpolated points can double back on themselves. The weighting function plots give a good indication of the curve path. To help visualize the curves, these plots are displaced to the right as t goes from 0 to 1 to eliminate the overlap.



1 segment with 32 new points per segment.

Quadratic Bézier Family With End Velocity Control

Produces interpolated points that are on a curve starting on P_1 and ending on P_3. At P_1 the curve is tangent to the line P_1P_2. At P_3 the curve is tangent to the line P_2P_3. The parameter V changes the end tangent vector magnitude. To get this control, the polynomial is cubic yet it still spans three control points.

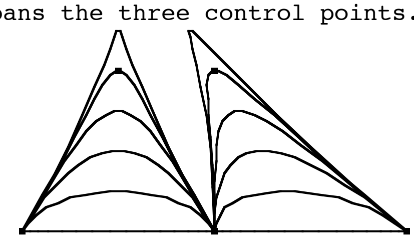
Basis Matrix

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & -V & 0 & V \\ t^2 & 0 & 1+2V & -2-V & 1-V \\ t & 0 & -2-V & 2+V & 0 \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

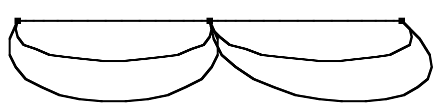
As t goes from 0 to 1 the curve goes from P_1 passing near P_2 to P_3. When V is 0, the curve is the quadratic Bézier. When V is -2 this is a simple cubic between P_1 & P_3. When V is 2 the curve intercepts the center control point.

Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

This type needs three control points to interpolate a segment. The segment spans the three control points.

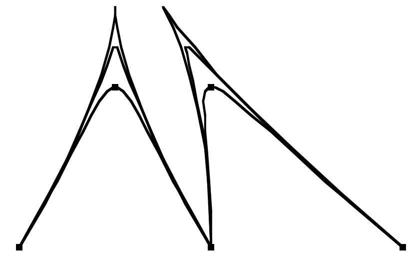


Quadratic Velocity Bézier $-2 \le V \le 3$ steps of 1
or $0 \le S \le 5$



Quadratic Velocity Bézier $-4 \le V \le -2$ steps of 1
or $-2 \le S \le 0$

Quadratic Bézier with End Velocity Control



Quadratic Velocity Bézier $+2 \le V \le +4$ steps of 1
or $4 \le S \le +6$

A simple substitution of the variable gives a curve referenced to zero end velocity.

Basis Matrix (zero end Speed referenced)

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 2-S & 0 & S-2 \\ t^2 & 0 & 2S-3 & -S & 3-S \\ t & 0 & -S & S & 0 \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

As t goes from 0 to 1 the curve goes from P_1 passing near P_2 to P_3. When S is 0 this is a simple cubic between P_1 & P_3. When S is 2, the curve is the quadratic Bézier. When S is 4 the curve intercepts the center control point.

Quadratic Lagrange

Produces interpolated points that are on a curve through P_1, P_2 and P_3. The value of t2 determines at what value of t the curve passes through P_2. This is not a spline, it only has G^0 continuity at the joints.

Basis Matrix

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 0 & 0 & 0 \\ t^2 & 0 & n1 & n2 & n3 \\ t & 0 & -n1-1 & -n2 & -n3*t2 \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

Inverse Matrix
not attempted.

Where t2 is the value of t at P_2. $0 < t2 < 1$

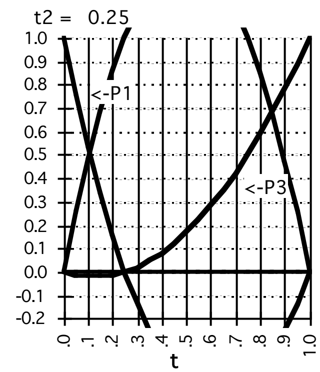
$$n1 = \frac{1}{t2} \qquad n2 = \frac{1}{t2(t2-1)} \qquad n3 = \frac{1}{1-t2}$$

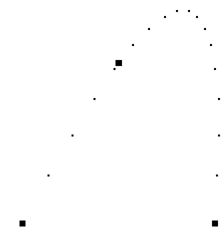
As t goes from 0 to 1 the curve goes from P_1 through P_2 to P_3.

Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

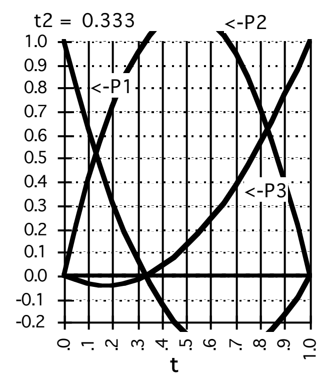
This type needs 3 control points to interpolate a segment. The segment spans the three control points. When t2 = 0.5, this is the quadratic three point (Golden). Bias causes the curve to pass through the center control point P_2 at different values of "t". The result is that the curve "peaks" before (preshoot) or after (post shoot) the center control point P_2. The peak, however, is always at t = 0.5.

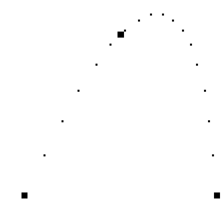
Quadratic Lagrange





High Post shoot.
1 segment with 16 new points per segment. End points are covered by the control points.

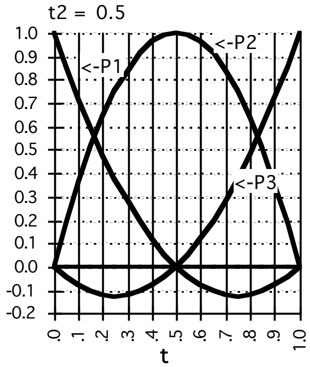


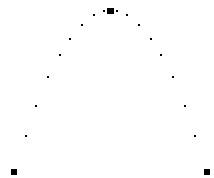


Moderate Post shoot

Variable Types A13

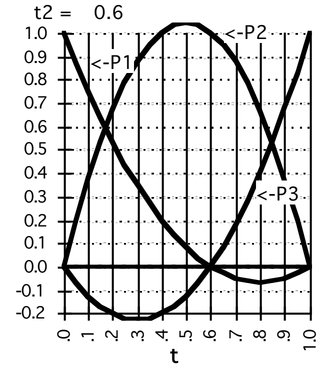
Quadratic Lagrange

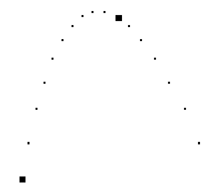




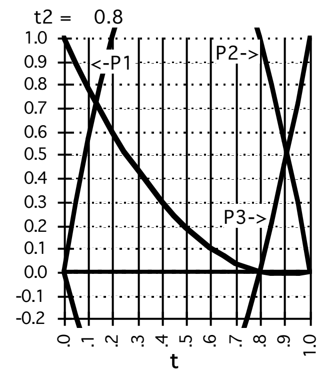
Nominal Bias

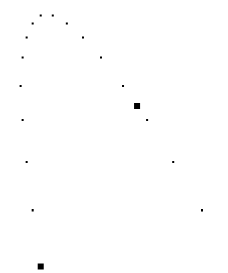
Quadratic Lagrange





Moderate Pre shoot





High Pre shoot

Cubic Lagrange

Produces interpolated points that are on a curve passing through P_0, P_1, P_2 and P_3. The value of t1 determines at what value of t the curve passes through P_1. The value of t2 determines at what value of t the curve passes through P_2. This is not a spline, it only has G0 continuity at the joints.

Basis Matrix

Inverse Matrix not attempted

$$
\begin{array}{c|cccc|}
 & P_0 & P_1 & P_2 & P_3 \\
t^3 & -n0 & n1 & n2 & n3 \\
t^2 & +n0(t1+t2+1) & n1(t2+1) & n2(t1+1) & -n3(t1+t2) \\
t^1 & -n0(t1+t2+t1t2) & -n1t2 & -n2t1 & n3t1t2 \\
t^0 & 1 & 0 & 0 & 0
\end{array}
$$

As t goes from 0 to 1 the curve goes from P_0 through P_1 & P_2 to P_3. Where t1 is the value of t at P_1 and t2 is the value of t at P_2.

$$0 < t1 < t2 < 1$$

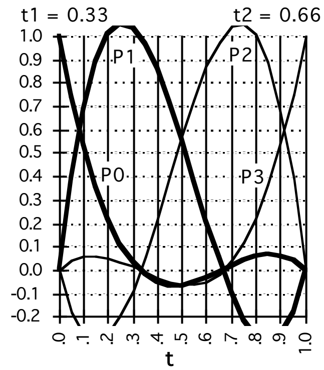
$$n0 = \frac{1}{t1t2} \qquad n1 = \frac{1}{t1(t1-t2)(1-t1)}$$

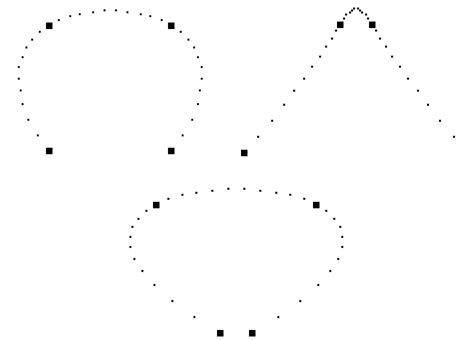
$$n2 = \frac{1}{t2(t2-t1)(1-t2)} \qquad n3 = \frac{1}{(1-t1)(1-t2)}$$

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

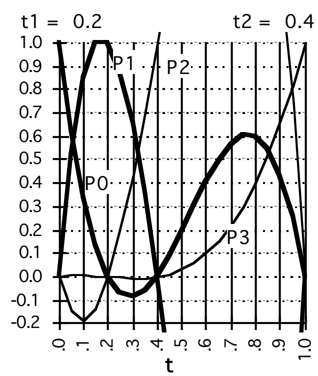
This type needs four control points to interpolate a segment. The segment spans the four control points. When t1 = 1/3 and t2 = 2/3, this is the cubic four point. Bias causes the curve to pass through P_1 when t = t1 and P_2 when t = t2. The result is that the curve "peaks" before (preshoot) or after (post shoot) the center control points P_2 and P_3.

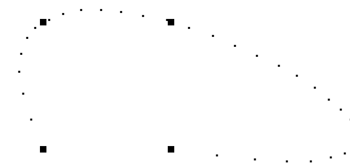
Cubic Lagrange

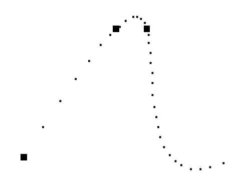




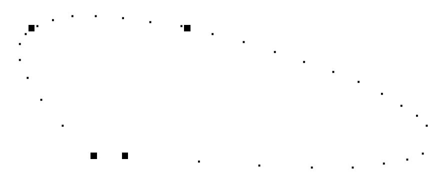
Nominal Bias.
1 segment with 34 total points per segment. End points and two inner points are covered by the control points.



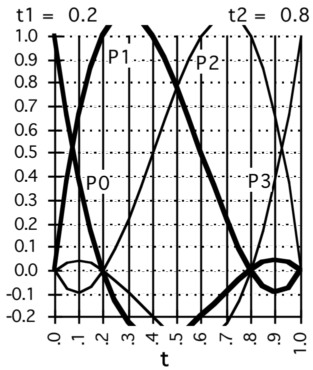


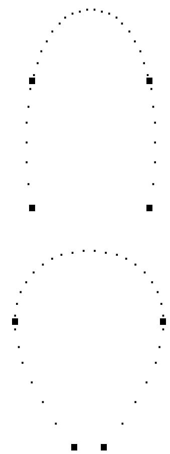


Both biases early.

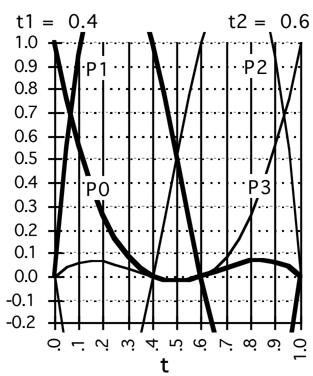


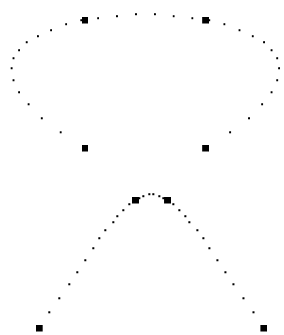
Cubic Lagrange



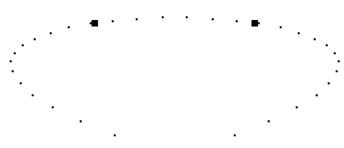


First Bias early,
second late.





First Bias late,
second early.



Beta Spline

This is the cubic B-Spline with tension and bias control. As tension goes from 0 to infinity, attraction is toward control point P_1. As tension goes to about -6, curve "repels" from control point P_1. As Bias goes from 1 to infinity attraction moves the joint from point P_1 earlier toward P0. As bias goes from 1 to 0, attraction moves the joint later toward P_2.

Basis Matrix

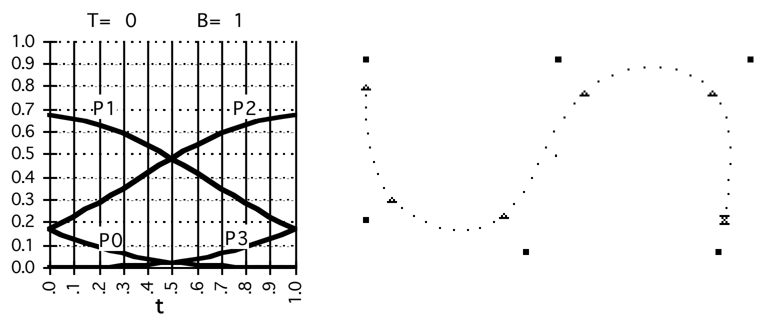
	P_0	P_1	P_2	P_3	
t^3	$-2B^3$	$2(T+B^3+B^2+B)$	$-2(T+B^2+B+1)$	2	
t^2	$+6B^3$	$-3(T+2B^3+2B^2)$	$3(T+2B^2)$	0	1
t	$-6B^3$	$6(B^3-B)$	6B	0	--------------
1	$2B^3$	$T+4(B^2+B)$	2	0	$T+2B^3+4B^2+4B+2$

As t goes from 0 to 1 the curve goes from near P_1 to near P_2. For B=1 & T=0 this traces the B-Spline.

Piecewise loop parameters:
- Start at the second control point P_1.
- Step one at a time
- End two from the last

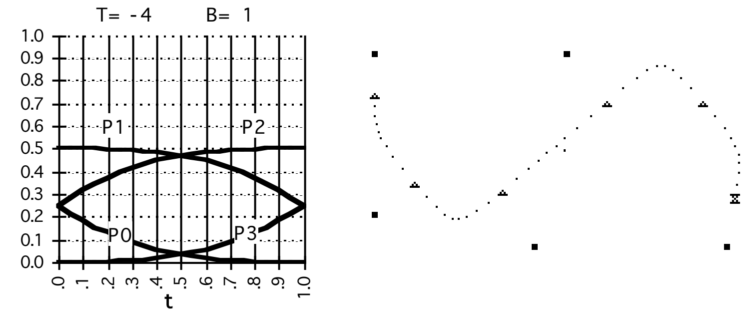
This type needs four control points to interpolate a segment. The segment spans the two inner control points although it does not pass through them. Tension pulls the curve toward the control points. Bias moves the joints backward toward the previous control point or forward toward the next control point.

These curves have 5 segments with 10 points per segment. This includes the two points at t = 0 and t = 1. Since the end points of each segment coincide, there are 46 total points. The knot points are marked.

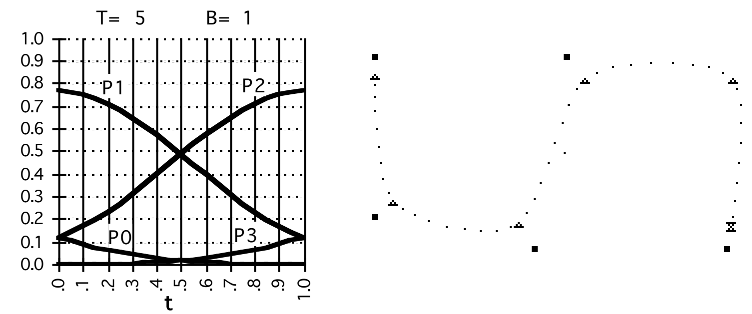


Neutral Tension & Bias (B-Spline

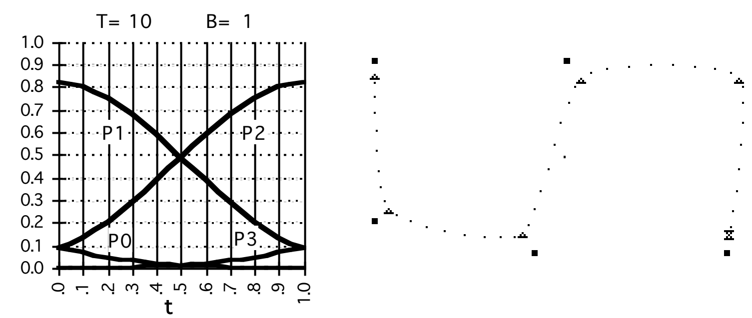
Beta Spline



Negative Tension.

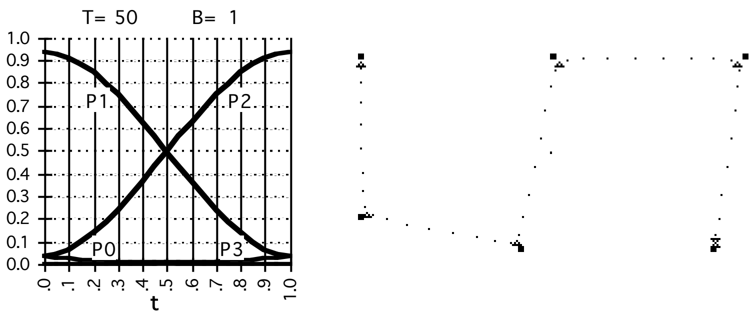


Low Tension.

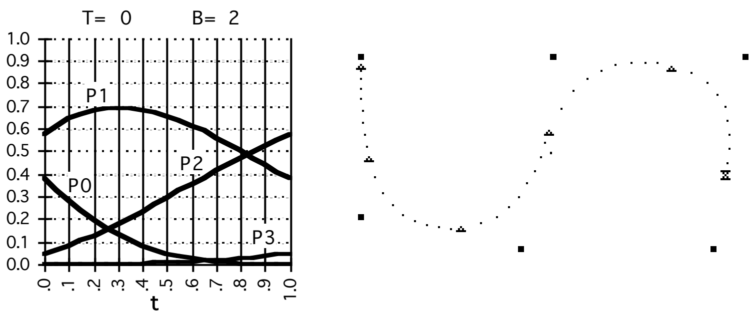


Moderate Tension.

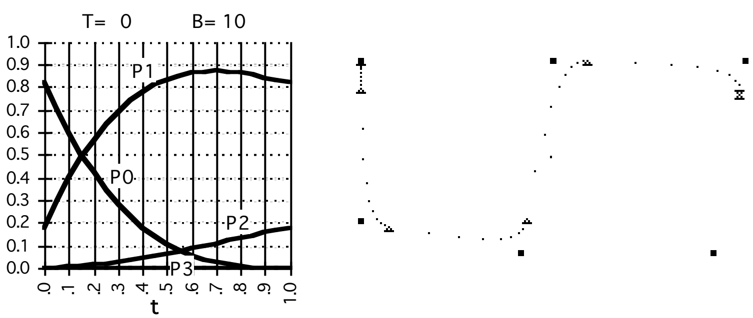
Beta Spline



High Tension

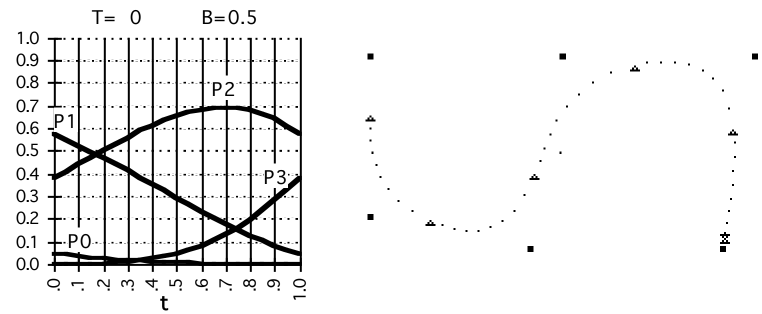


Moderate early Bias.

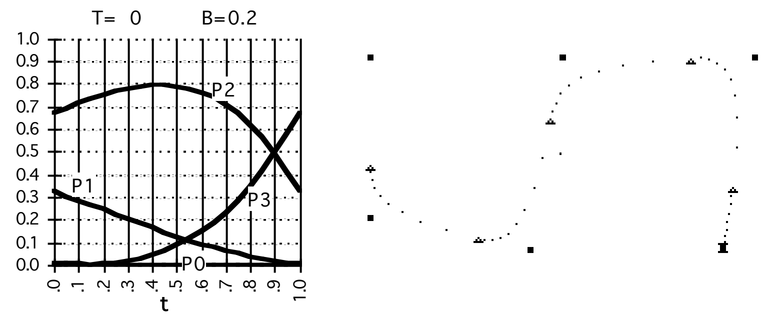


Very early Bias

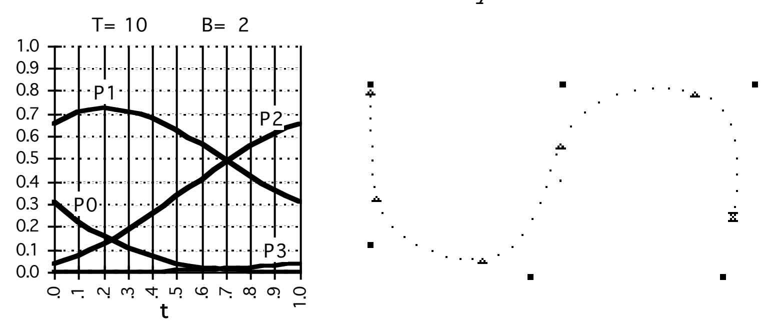
Beta Spline



Moderate late Bias.

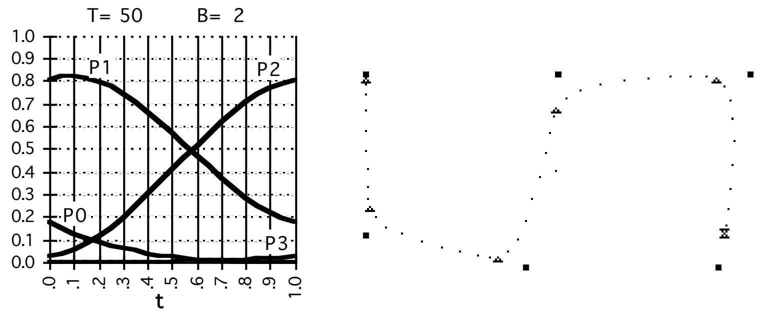


Very late Bias.

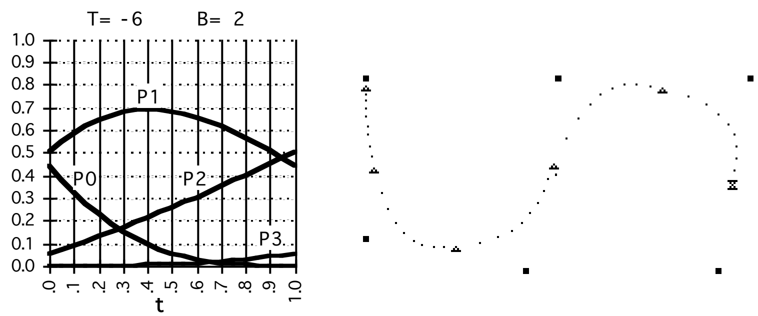


Moderate Tension, Early Bias.

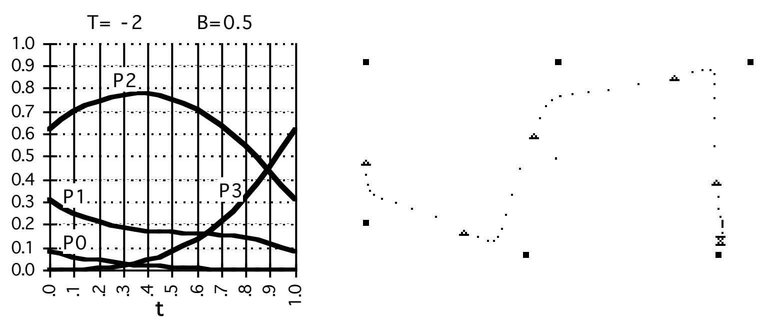
Beta Spline



High Tension, Early Bias



Negative Tension, Early Bias.



Negative Tension, late Bias.

Kochanek-Bartels

Produces interpolated points that are on a smooth curve through all control points. There is control of the tension, bias and continuity at the control points. The tension parameter controls the magnitude of the tangent vector (speed) at the control points. The Bias parameter controls the direction of the tangent vector at the control points by changing the relative effect of the two adjacent control points. The continuity parameter causes the incoming tangent vector to differ in direction from the out going tangent vector in a symmetrical manner.

Basis Matrix

	P_0	P_1	P_2	P_3	
t^3	-A	4+A-B-C	-4+B+C-D	D	
t^2	+2A	-6-2A+2B+C	6-2B-C+D	-D	1
t	-A	A-B	B	0	---
1	0	2	0	0	2

Where A, B, C and D are defined as:
A=(1-Te)*(1+Co)*(1+Bi)
B=(1-Te)*(1-Co)*(1-Bi)
C=(1-Te)*(1-Co)*(1+Bi)
D=(1-Te)*(1+Co)*(1-Bi)

As t goes from 0 to 1 the curve goes from P_1 to P_2

Tension	Te=+1-->Tight	Te=-1--> Round
Bias	Bi=+1-->Post Shoot	Bi=-1--> Pre shoot
Continuity	Co=+1-->Inverted corners	Co=-1--> Box corners

When Te = Bi = Co = 0 this is the Catmul-Rom.
When Te = 1 this is the Simple Cubic (Bi & Co are don't care)
When Te = Bi = 0 & Co = -1 this is the linear interp.

Piecewise loop parameters:
 Start at the second control point P_1.
 Step one at a time
 End two from the last

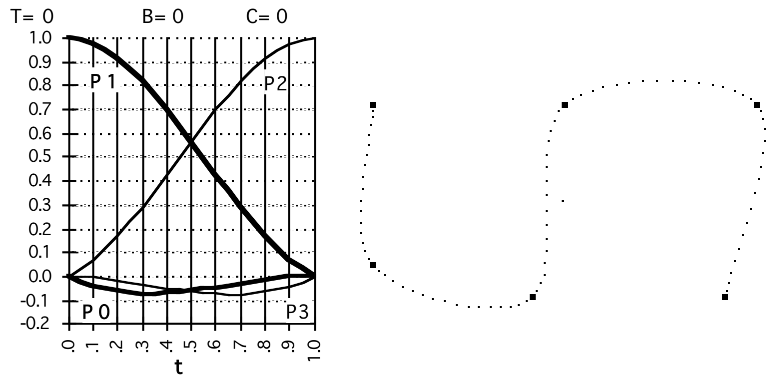
This type needs four control points to interpolate a segment. The segment spans the two inner control points. Additional controls are required for Tension Bias and Continuity. This can be done globally as shown here, requiring three additional controls or locally requiring three controls for each segment.

To obtain independent local control of the segment start and end for tension, bias and continuity, make two T, C & Bs. One for A & B (segment beginning) and one for C & D (segment end).

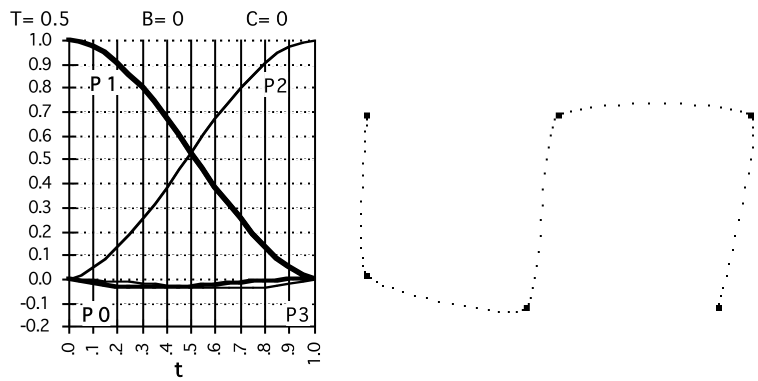
Kochanek-Bartels

There are two ways to do local control of tension, bias and continuity.
For local control of each point as shown on the video or in the paper, you'll need an array of T, C & Bs for each individual point. You use the "A" & "B" for the current segment and the "C" & "D" for the previous segment.
For local control of both ends of each segment together, use "A", "B", "C" & "D" from the same segment. This can be considered a generalization of the Catmul-Rom. The end control points are not intercepted unless they are repeated.

These are shown with 5 segments of 16 new points per segment. Some of these points are covered by the control point dots.

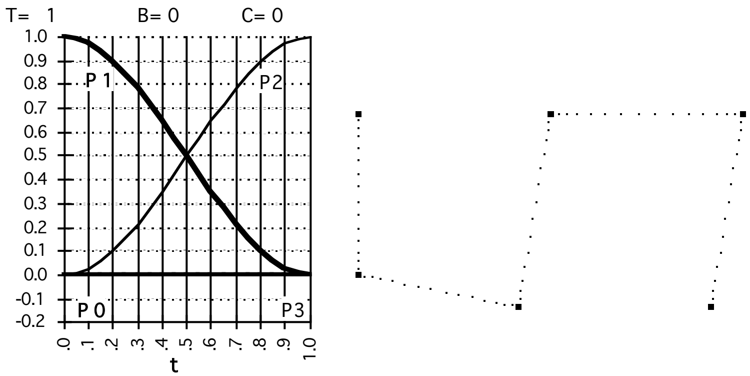


Equivalent to Catmul-Rom

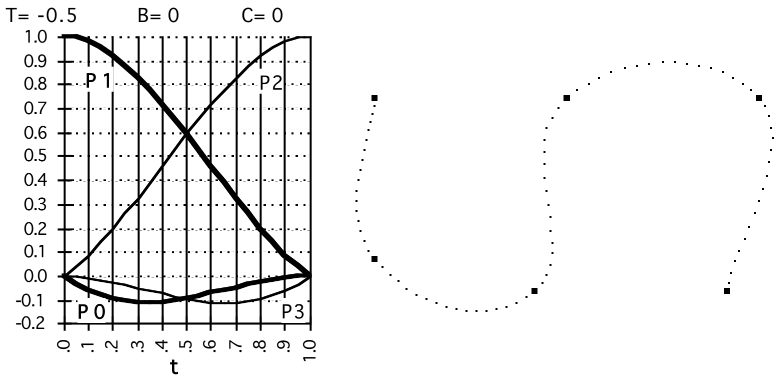


Moderate tension.

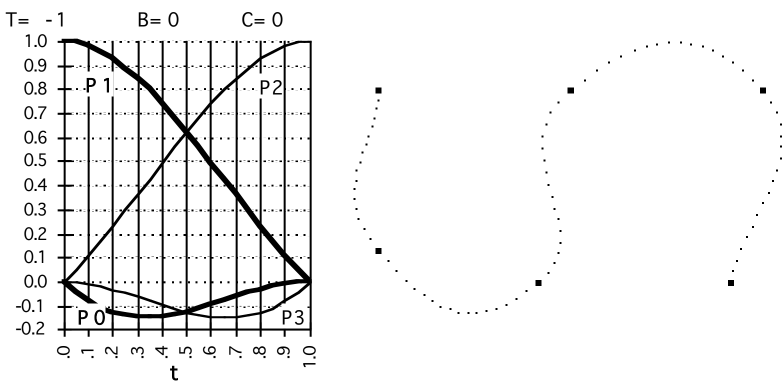
Kochanek-Bartels



High tension, equivalent to Simple Cubic

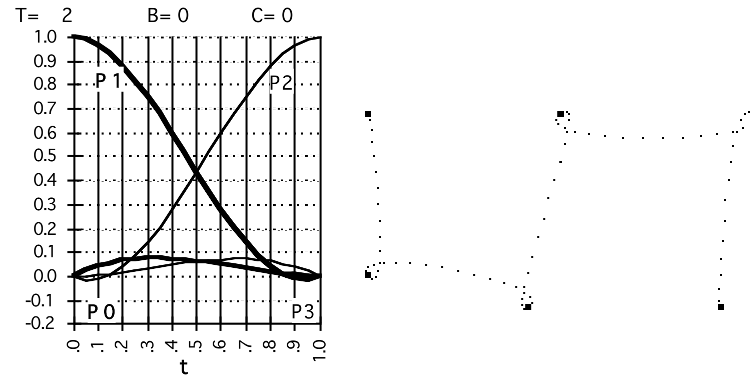


Moderate negative Tension

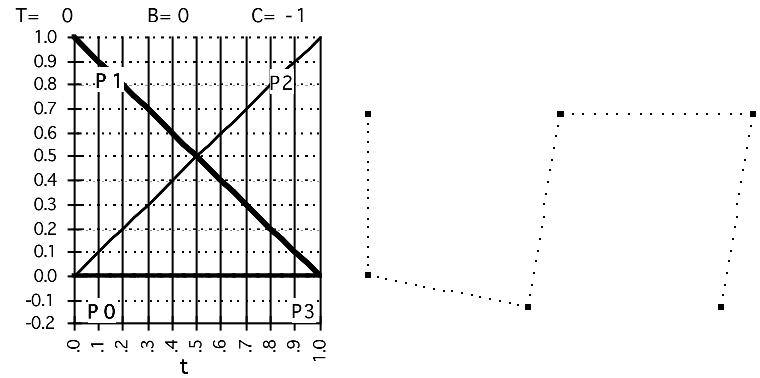


High negative Tension

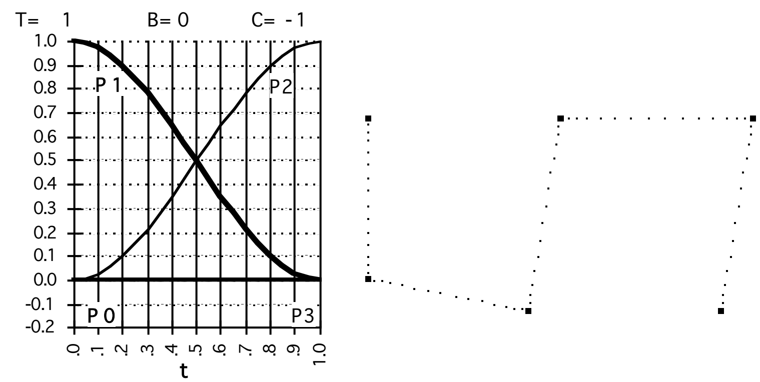
Kochanek-Bartels



Excessive Tension.

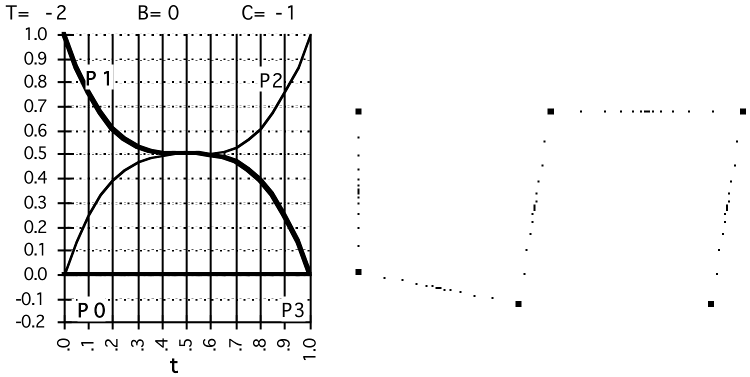


Equivalent to Linear

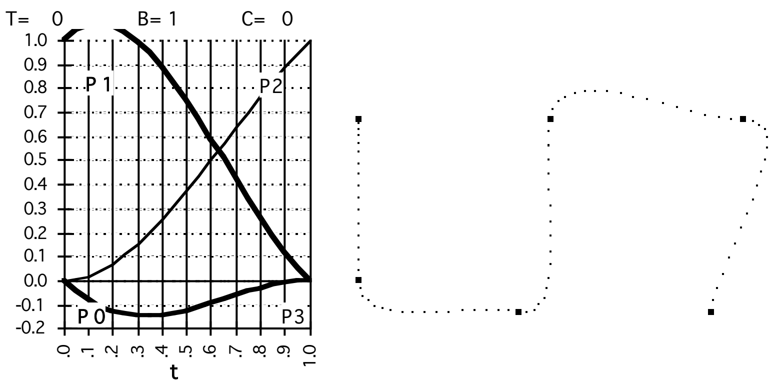


Equivalent to Simple Cubic.

Kochanek-Bartels

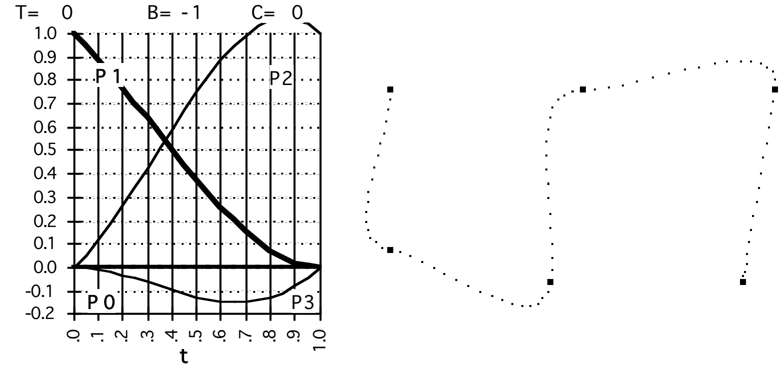


Zero velocity at mid points.

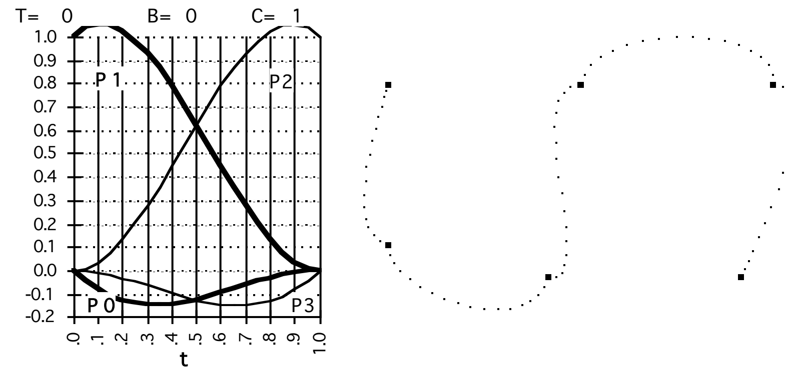


Positive Bias.

Kochanek-Bartels



Negative Bias.



Negative Continuity

Cubic Bézier Family With End Velocity Control

Produces interpolated points that are on a curve starting on P_0 and ending on P_3. At P_0 the curve remains tangent to the line P_0P_1. At P_3 the curve remains tangent to the line P_2P_3.

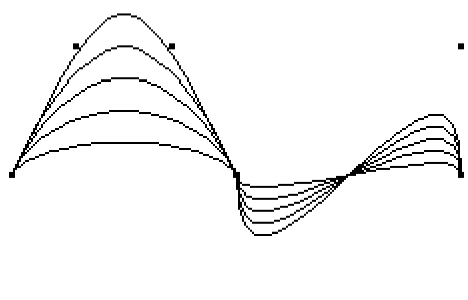
Basis Matrix

	P_0	P_1	P_2	P_3
t^3	-1-V	3+V	-3-V	1+V
t^2	3+2V	-6-2V	3+V	-V
t	-3-V	3+V	0	0
1	1	0	0	0

As t goes from 0 to 1 the curve goes from P_0 to P_3. The parameter V controls the end tangent magnitude.

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

This type needs four control points to interpolate a segment. The segment spans the four control points.



Cubic Velocity Bézier $-2 \le V \le +2$ steps of 1

Now a non-bitmapped PICTure which should show dots?

Palmer

Produces interpolated points that are on a curve from P_1, near P_2 to P_3. This is a generalized form of the Golden with bias control and a crude form of tension control. The curve intercepts P_2 at a parameter value of t1 if T = 0.0 . This is not a spline. Derived by Sean Palmer and obtained in private e-mails.

Basis Matrix

	P_0	P_1	P_2	P_3
t^3	0	-2T	0	2T
t^2	0	n0+3T	-n1	n2-3T
t	0	-1-n0-T	n1	T-n2/n0
1	0	1	0	0

n0 = 1/t1
n1 = 1/(t1*(1-t1))
n2 = 1/(1-t1)

As t goes from 0 to 1 the curve goes from P_1 passing near/through P_2 to P_3.
T is a tension-like parameter. The curve becomes sharper at t = 0.5 when T is positive and less sharp when T is negative.
The curve only intercepts P_2 when T = 0.0.
t1 is Bias. The curve passes near / through P_2 when t = t1.

Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

This type needs three control points to interpolate a segment. The segment spans the three control points. When t2 = 0.5 and T = 0.0 this is the Golden. When T = 0.0 this is the quadratic Lagrange.

Needs sample curves

Appendix A14 Basis Matrices, Morph Types

This appendix contains basis matrices for morph interpolation types. Included with each is information on the curve behavior, the effect of the morph parameter and samples of the interpolated curve. The inverse matrix and weighting function plots are not shown.
Sometimes the first point required for a particular curve is called P_0 and sometimes it is called P_1. Types that do not have a 4x4 basis matrix usually start on the point called P_1. For these types P_0 may *or may not* exist. The rows, columns and entries of any specific type that are zero can be omitted from the calculation. –

The example curves are drawn as lines to better show the range of curve behavior. Since they are based on other types of curves, the parametric speed (dot spacing) can be deduced from the foundation curves or the reader can calculatr thr derivative or code them up and observe them.

Morph Interpolation Types

- B-Simp
- B-Cat
- Quadratic W/Attraction / Pressure control
- Cubic W / Attraction Control #1
- Cubic W / Attraction Control #2
- Linear W/Phase Control
- Parabolic W/Phase Control
- Parabolic W/ Attraction Control
- Quadratic Bézier W/Phase Control
- Catmul-Rom W/Phase Control
- B-spline W/Phase Control
- Quadratic to Cubic Bézier Morph
- Cubic to Quartic Bézier morph (soft degree elevation)

B-Simp
(B-Spline --> Simple Cubic)

Produces interpolated points that pass near the control points. The parameter A, called attraction, causes the curve to be attracted toward the control points.

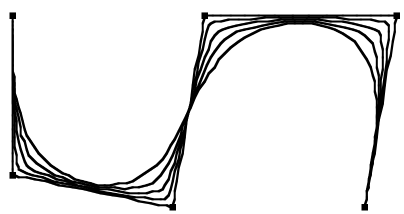
Linear Morph Basis Matrix

$$
\begin{array}{c|cccc|c}
 & P_0 & P_1 & P_2 & P_3 & \\
t^3 & A-1 & 3+9A & -3-9A & 1-A & \\
t^2 & 3-3A & -6-12A & 3+15A & 0 & 1 \\
t & 3A-3 & 0 & 3-3A & 0 & \text{---} \\
1 & 1-A & 4+2A & -A & 0 & 6
\end{array}
$$

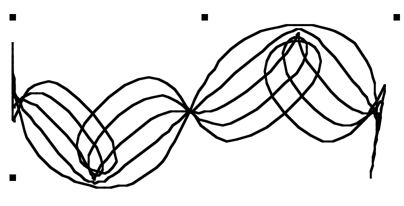
As t goes from 0 to 1, the curve goes from near P_1 to near P_2. As A goes from 0 to 1 the curve goes from the B-spline to the Simple Cubic.

Piecewise loop parameters:
- Start at the second control point P_1.
- Step one at a time
- End two from the last

This type needs four control points to interpolate a segment. The segment spans two control points although it does not necessarily pass through them.

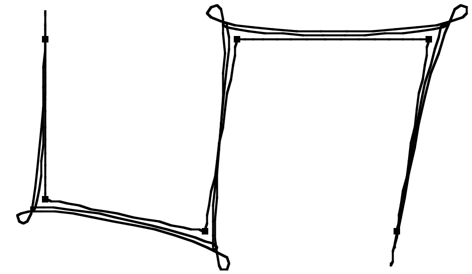


B-Spline --> Simple Cubic, Linear $0 \le A \le 1$ steps of 0.5



B-Spline --> Simple Cubic, linear $-2 \le A \le 0$ steps of 0.5

B-Spline --> Simple Cubic



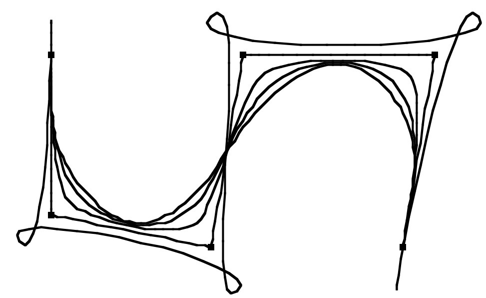
B-Spline --> Simple Cubic, linear $1 \le A \le 2$ steps of 0.5

An interesting variation is obtained when the divisor is also morphed.

Non-Linear Morph Basis Matrix

$$
\begin{array}{c|cccc|c}
 & P_0 & P_1 & P_2 & P_3 & \\
t^3 & A-1 & 3-A & A-3 & 1-A & \\
t^2 & 3-3A & 3A-6 & 3 & 0 & 1 \\
t & 3A-3 & 0 & 3-3A & 0 & \text{----} \\
1 & 1-A & 4-3A & 1-A & 0 & 6-5A
\end{array}
$$

As t goes from 0 to 1 the curve goes from near P_1 to near P_2. As A goes from 0 to 1 the curve goes from the B-spline to the simple cubic. The practical range of A is -60 to 1.19. The function is undefined at A = 1.2 due to the 6-5A in the divisor.



B-Spline -> Simple cubic, non-linear A = -60, 0, .75, 1, 1.1

B-Cat
B-Spline --> Catmul-Rom

Produces interpolated points that pass near the control points. The parameter A, called attraction, causes the curve to be attracted toward the control points.

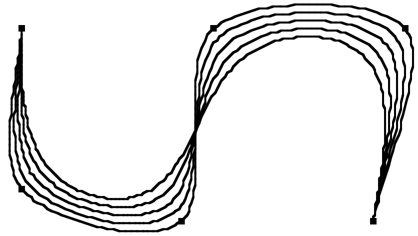
Linear Morph Basis Matrix

$$
\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}
\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ \left.\begin{array}{cccc} -1-2A & 3+6A & -3-6A & 1+2A \\ 3+3A & -6-9A & 3+9A & -3A \\ -3 & 0 & 3 & 0 \\ 1-A & 4+2A & 1-A & 0 \end{array}\right| \end{array}
\ \frac{1}{6}
$$

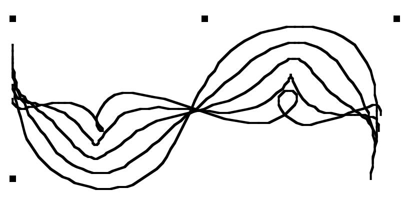
As t goes from 0 to 1, the curve goes from near P_1 to near P_2
As A goes from 0 to 1 the curve goes from the B-spline to the Catmul-Rom.

Piecewise loop parameters:
- Start at the second control point P_1.
- Step one at a time
- End two from the last

This type needs four control points to interpolate a segment. The segment spans two control points although it does not pass through them except when A = 1. Note that since the endpoints of the B-spline and Catmul-Rom do not coincide, they will drift as Attraction varies unless end control points are duplicated

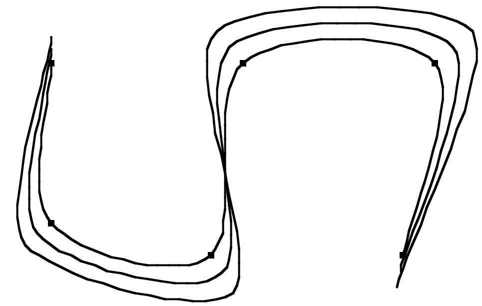


B-Spline --> Catmul, linear $0 \le A \le 1$ steps of 0.5



B-Spline --> Catmul, linear $-2 \le A \le 0$ steps of 0.5

B-Spline --> Catmul-Rom



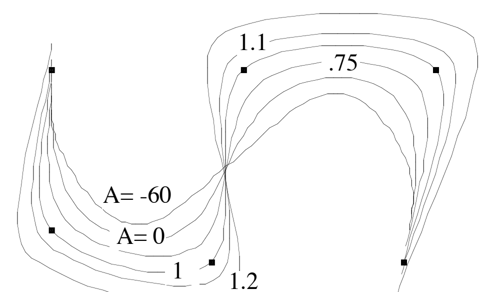
B-Spline --> Catmul, linear $1 \le A \le 2$ steps of 0.5

An interesting variation is obtained when the divisor is also morphed.

Non-Linear Morph Basis Matrix

$$
\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}
\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ \left|\begin{array}{cccc} -1 & 3 & -3 & 1 \\ 3-A & A-6 & 3+A & -A \\ 2A-3 & 0 & 3-2A & 0 \\ 1-A & 4-2A & 1-A & 0 \end{array}\right| \end{array}
\ \frac{1}{6-4A}
$$

As t goes from 0 to 1, the curve goes from near P_1 to near P_2
As A goes from 0 to 1 the curve goes from the B-spline to the Catmul-Rom. The practical range of A is -60 to 1.49. The function is undefined at A = 1.5 due to the 6-4A in the divisor.



B-Spline --> Catmul, non-linear A = -60, 0, .75, 1, 1.1, 1.2

Quadratic Bézier based w/Attraction Control

Produces interpolated points that are on a curve starting on P_1 and ending on P_3. The parameter A, called attraction, causes the curve to be attracted toward the center control point.

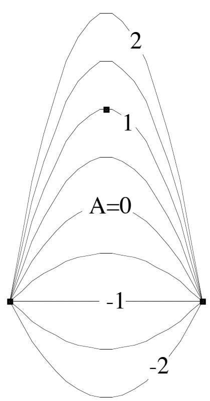
Basis Matrix

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 0 & 0 & 0 \\ t^2 & 0 & 1+A & -2-2A & 1+A \\ t & 0 & -2-A & 2+2A & -A \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

As t goes from 0 to 1 the curve goes from P_1 passing near P_2 to P_3. When A is 0, the curve is the quadratic Bézier. When A is 1, this passes through P_2 (quadratic Golden). When A is -1, this is linear between P_1 and P_3.

For both variations:
Piecewise loop parameters:
- Start at the first control point P_1.
- Step two at a time
- End two from the last

This type needs three control points to interpolate a segment. The segment spans the three control points.



Quadratic with Attraction $-2 \le A \le 2$ steps of 1
or $-1 \le P \le 3$

Morph Types A14

Quadratic Bézier based w/Attraction Control

Defining a variable "P" that is zero when A=-1 (straight line) gives behavior roughly analogous to pressure. P = A+1

Basis Matrix (zero pressure referenced)

$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^3 & 0 & 0 & 0 & 0 \\ t^2 & 0 & P & -2P & P \\ t & 0 & -1-P & 2P & 1-P \\ 1 & 0 & 1 & 0 & 0 \end{array}$$

As t goes from 0 to 1 the curve goes from P_1 passing near P_2 to P_3. When P is 0 this is linear between P_1 & P_3. When P is 1, this is the quadratic Bézier. When P is 2, this passes through P_2 (quadratic Golden).

Cubic w/ Attraction Control #1

Produces interpolated points that are on a curve starting on P_0 and ending on P_3. The curve goes "near" or is attracted to P_1 and P_2.

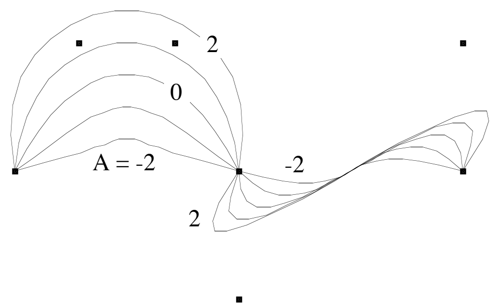
Basis Matrix

$$
\begin{array}{c|cccc|}
 & P_0 & P_1 & P_2 & P_3 \\
t^3 & A-1 & 3+A & -3-A & 1-A \\
t^2 & 3-A & -6-2A & 3+A & 2A \\
t & -3 & 3+A & 0 & -A \\
1 & 1 & 0 & 0 & 0
\end{array}
$$

As t goes from 0 to 1 the curve goes from P_0 to P_3 When A is 0, this is the cubic Bézier.

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

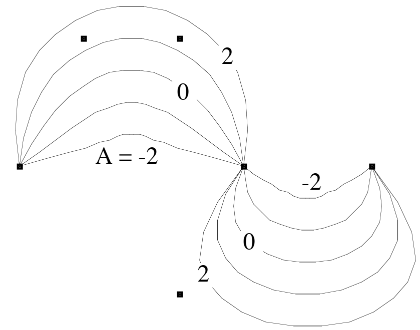
This type needs four control points to interpolate a segment. The segment spans the four control points.



Cubic Attraction #1 $-2 \le A \le 2$ steps of 1

Note that this curve has irregular behavior. The behavior around the inner control points is not consistent as the control points move.

Cubic With Attraction Control #1



Cubic Attraction #1 $-2 \le A \le 2$ steps of 1
The behavior near the inner control points is not consistent.

Cubic w/ Attraction Control #2

Produces interpolated points that are on a curve starting on P_0 and ending on P_3. The curve goes near P_1 and P_2.

Basis Matrix

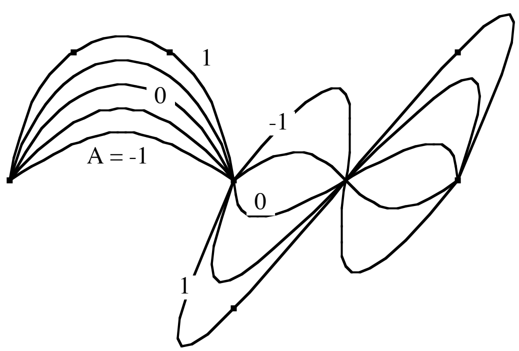
$$
\begin{array}{c} \\ t^3 \\ t^2 \\ t \\ 1 \end{array}
\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ \hline \end{array}
$$

$$
\begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array}
\left|\begin{array}{cccc} -2-7A & 6+21A & -6-21A & 2+7A \\ 6+12A & -12-33A & 6+30A & -9A \\ -6-5A & 6+12A & 9A & 2A \\ 2 & 0 & 0 & 0 \end{array}\right| \frac{1}{2}
$$

As t goes from 0 to 1, the curve goes from P_0 to P_3. When A is zero, this is the Cubic Bézier. When A is 1, this is the Cubic Four Point and the curve passes through the inner control points.

Piecewise loop parameters:
- Start at the first control point P_0.
- Step three at a time
- End three from the last

This type needs four control points to interpolate a segment. The segment spans the four control points.

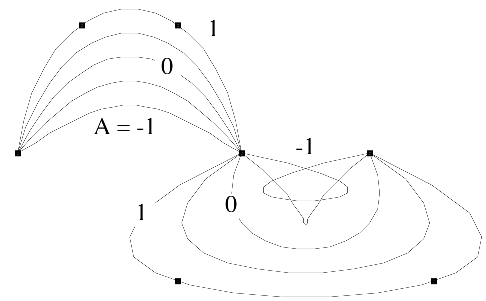


Cubic with Attraction #2 $-1 \le A \le 1$ steps of 0.5

Note that this curve has irregular behavior. The behavior around the inner control points is not consistent as the control points move.

Show the other set of control points like type #1

Cubic w/ Attraction Control #2



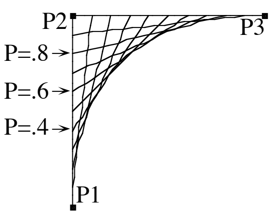
Cubic with Attraction #2 $-1 \le A \le 1$ steps of 0.5

Linear w/ Phase Control

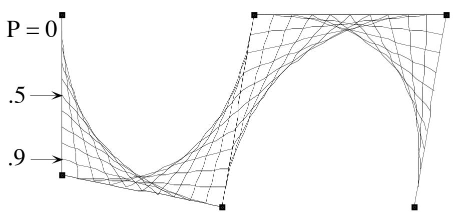
Produces linear interpolated points that are in a straight line. The Phase parameter shifts from one set of control points to the next that is shifted by one in the sequence. It is a morph of the line from the P_0P_1 line to the P_1P_2 line.

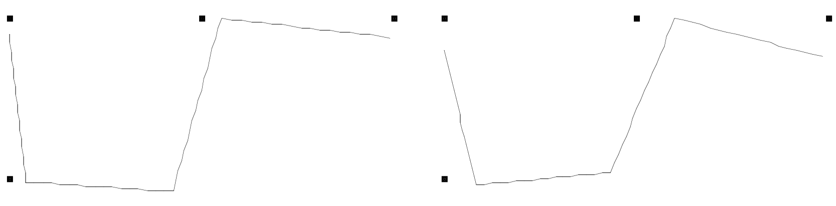
Basis Matrix

$$
\begin{array}{c} \\ t \\ 1 \end{array}
\begin{array}{c}
\begin{array}{ccc} P_0 & P_1 & P_2 \end{array} \\
\left|\begin{array}{ccc} -1+P & 1-2P & P \\ 1-P & P & 0 \end{array}\right|
\end{array}
$$



Linear with Phase in steps of 0.2





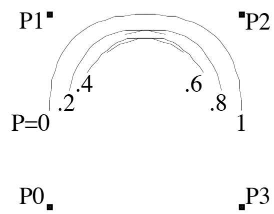
Phase at 0.1. Phase at 0.2.

These show quite clearly the apparent motion of control points to the right in the sequence. The joints move on a straight line to the next control point.

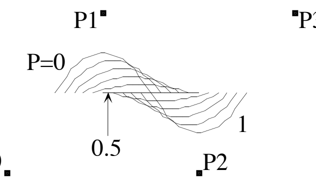
Parabolic (quadratic B-spline) w/ Phase Control

Produces a parabolic curve. The Phase parameter shifts from one set of control points to the next that is shifted by one in the sequence. It is a morph from the curve with control points $P_0P_1P_2$ to the curve with control points $P_1P_2P_3$.

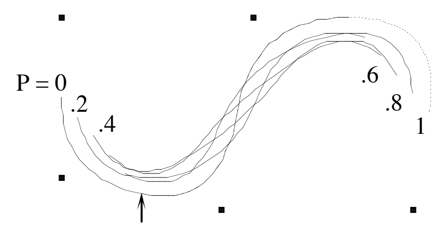
$$
\begin{array}{c} \\ t^2 \\ t \\ 1 \end{array}
\begin{array}{c}
\begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \end{array} \\
\left|\begin{array}{cccc} 1-P & -2+3P & 1-3P & P \\ -2+2P & 2-4P & 2P & 0 \\ 1-P & 1 & P & 0 \end{array}\right|
\end{array}
\; \frac{1}{2}
$$



Steps of 0.2



Steps of 0.1



Steps of 0.2

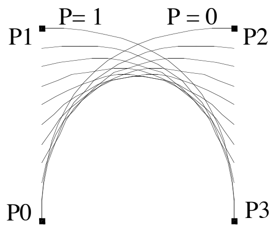
The small difference between the phase-shifted curves makes this a rather un-interesting variation. The start of the P=1 curve is at the arrow and most of it is hidden by the P=0 curve until it emerges as a dotted line at the upper right.

Quadratic Bézier w/ Phase Control

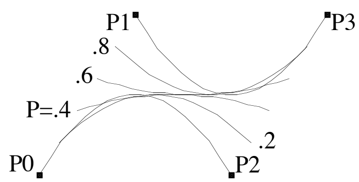
Basis Matrix

$$\begin{array}{c} \\ t^2 \\ t \\ 1 \end{array} \begin{array}{cccc} P_0 & P_1 & P_2 & P_3 \\ \hline \end{array}$$

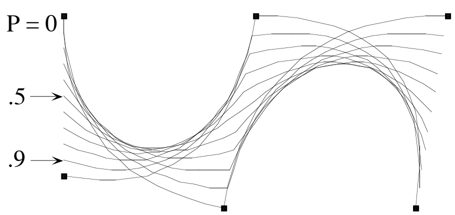
$$\begin{array}{c|cccc|} & P_0 & P_1 & P_2 & P_3 \\ t^2 & 1-P & -2+3P & 1-3P & P \\ t & -2-P & 2-4P & 2P & 2P \\ 1 & 1-P & P & 0 & 0 \end{array}$$



Steps of 0.1



Steps of 0.2

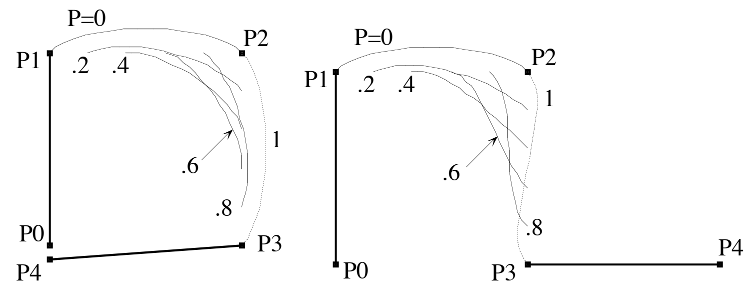


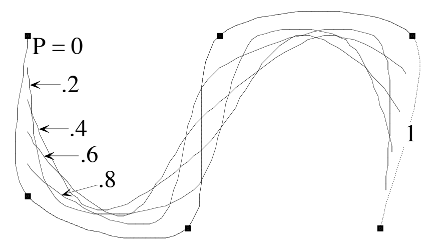
Steps of 0.1

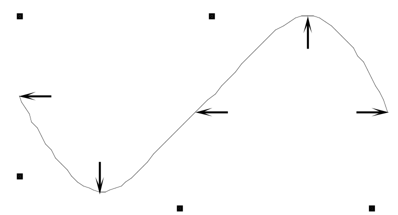
Catmul-Rom w/ Phase Control

Basis Matrix

$$\begin{array}{c|ccccc|c} & P_0 & P_1 & P_2 & P_3 & P_4 & \\ t^3 & 1-P & 3-4P & -3+6P & 1+4P & P & \\ t^2 & 2-2P & -5+7P & 4-9P & -1+5P & -P & 1 \\ t & -1+P & -P & 1-P & -P & 0 & --- \\ 1 & 0 & 2-2P & 2P & 0 & 0 & 2 \end{array}$$



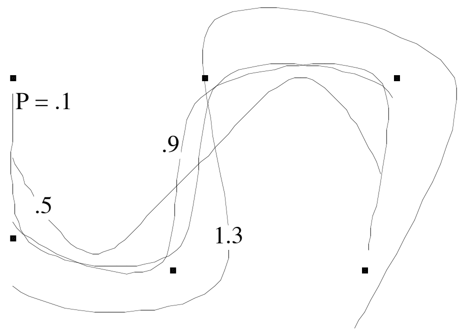




Phase at 0.5

Arrows indicate the location half way to the next control point. This is where the effective control points are located due to the phase shift of 0.5. With control points at these locations, the standard Catmul-Rom technique would draw this same curve.

Catmul-Rom w/ Phase Control

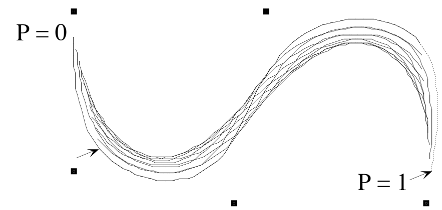


With phase values greater than 1, the location of the interpolated control points can be seen to extend past the next in line.

Cubic B-Spline w/ Phase Control

Basis Matrix

$$\begin{array}{cc} & \begin{array}{ccccc} P_0 & P_1 & P_2 & P_3 & P_4 \end{array} \\ \begin{array}{c} t^3 \\ t^2 \\ t \\ 1 \end{array} & \left|\begin{array}{ccccc} -1+P & 3-4P & -3+6P & 1+4P & P \\ 3-3P & -6+9P & 3-9P & -3P & 0 \\ -3+3P & -3P & 3-3P & -3P & 0 \\ 1-P & 4-3P & 1+3P & -P & 0 \end{array}\right| \dfrac{1}{6} \end{array}$$



The start of the P=1 curve is at the arrow and most of it coincides with the P=0 curve. The small difference between the phase-shifted curves makes this a rather un-interesting variation.

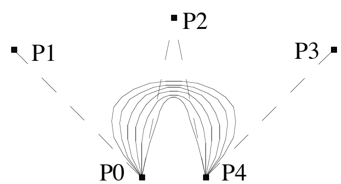
Quadratic to Cubic Bézier Morph

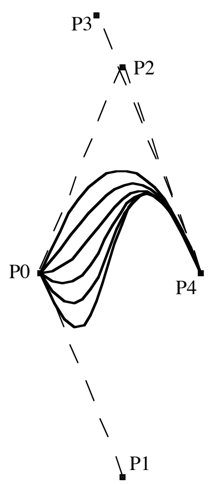
Produces points on a curve that changes from a quadratic Bézier to cubic Bézier.

Basis Matrix

	P_0	P_1	P_2	P_3	P_4
t^3	-M	3M	0	-3M	M
t^2	1+2M	-6M	2+2M	3M	1-M
t	-2-M	3M	2-2M	0	0
1	1	0	0	0	0

As t goes from 0 to 1 the curve goes from P_0 to P_4. As the Morph parameter "M" goes from 0 to 1, the curve goes from quadratic Bézier to cubic Bézier. For the quadratic curve (m=0), P_2 is the inner control point. For the cubic, P_1 and P_3 are the inner control points. This may be considered to be like a "soft" degree elevation, though the curve is quadratic only when M = 0.





Cubic to Quartic Bézier Morph

Produces points on a curve that changes from a cubic Bézier to quartic Bézier.

Basis Matrix

	P_0	P_1	P_2	P_3	P_4
t^4	M	-4M	6M	-4M	M
t^3	-1-3M	3+9M	-2M	-3+7M	1-M
t^2	3+3M	-6-M	6M	3-3M	0
t	-3-M	3+M	0	0	0
1	1	0	0	0	0

As t goes from 0 to 1 the curve goes from P_0 to P_4. As the Morph parameter "M" goes from 0 to 1, the curve goes from cubic Bézier to quartic Bézier. For the cubic curve, P_1 and P_3 are the inner control points. For the quartic, P_1, P_2 and P_3 are the inner control points. This may be considered to be like a "soft" degree elevation, though the curve is cubic only when M = 0.

Needs sample curves

Appendix A15 Trigonometric Weighting Functions

This appendix contains trigonometric basis matrices. Though not covered in the text, these are derived using trig functions. The parameter is the angle used in the trig functions and instead of the parameter vector (matrix) containing powers of the parameter "t", it contains the trig functions as shown in Noskowicz Notation, to the left of the basis matrix.

Included with each is information on the curve behavior and samples of the interpolated curve.
The rows, columns and entries of any specific type that are zero can be omitted from the calculation. -

For the sample curves, the number of points per segment, the number of segments shown and the arrangement of control points were selected to best show the characteristics of each type. *Note that some of the interpolated points can be obscured by the larger control points.*

The example curves are drawn as dots to better show the range of curve behavior.

NOTE: The basis matrix for trigonometric curves contains only ones. The methods for finding start/end points and tangents differ from the polynomial curves. The full weighting function for each point must be evaluated as follows. Each trig function value (evaluated at the appropriate parameter value of t=0 or t=1) is multiplied by the basis matrix entry to its right to obtain the "true" entry value. Then the "true" values are always summed by columns. To take the derivative in order to find the end tangents, the derivative of the trig functions in the parameter vector (to the left of the basis matrix) must be used.
The derivative of $\mathrm{Cos}(\pi t) = -\pi \mathrm{Sin}(\pi t)$.
The derivative of $\mathrm{Sin}(\pi t) = \pi \mathrm{Cos}(\pi t)$.
The derivative of $\mathrm{Sin}(\pi t)*\mathrm{Cos}(\pi t) = \pi \mathrm{Cos}(2\pi t)$

Trig. Interpolation Types

Simple
Cubic Bézier like

Simple Trig

This type is very similar to the Simple Cubic. Produces interpolated points that are on a straight line between adjacent control points. The spacing of the interpolated points (the velocity / first derivative) decreases to zero at the control points. The "curve" passes through both control points. This is not a spline. The weighting functions are a portion of the cosine function.

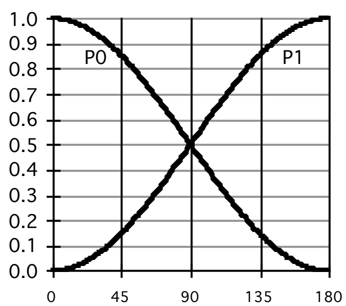
Basis Matrix

$$\begin{array}{c|cc|c} & P_0 & P_1 & 1 \\ \mathrm{COS}(\pi t) & 1 & -1 & --- \\ 1 & 1 & 1 & 2 \end{array}$$

As t goes from 0 to 1 the curve goes from P_0 to P_1

Piecewise loop parameters:
- Start at the first control point P_0.
- Step one at a time.
- End at next to last.

This type needs two control points to interpolate a segment. One segment spans those two control points. The first derivative and therefore the velocity is always zero at control points.



Position basis matrix showing sum of columns for determining the start and end point of the curve.

$$\begin{array}{c|cc|c} & P_0 & P_1\ 1 & \\ \mathrm{COS}(\pi t) & 1 & -1 & --- \\ 1 & 1 & 1 & 2 \end{array}$$

Sum of collumns.
t=0,COS(0)=1 -> 2 0 start is P_0
t=1,COS(1)=-1 -> 0 2 finish is P_1

Simple Trig

Derivative of basis matrix showing sum of columns for determining the start and end tangents.

$$\begin{array}{c|cc|c} & P_0 & P_1 & \pi \\ -\mathrm{Sin} & 1 & -1 & --- \\ 0 & 1 & 1 & 2 \end{array}$$

t=0,-SIN(0)=0 -> 0 0 start slope is zero.
t=1,-SIN(1)=0 -> 0 0 finish slope is zero.

Needs sample curves.

Imitation of Cubic Bézier

This type is very similar to the Cubic Bézier.
Produces interpolated points that are on a curve starting on P_0 and ending on P_3. At P_0 the curve is tangent to the line P_0P_1. At P_3 the curve is tangent to the line P_2P_3. The result of this is that it appears that the curve goes near or is drawn toward P_1 and P_2. This is not a spline. Whereas the Cubic Bézier end tangents are equal to three times the distance between the end and adjacent control point, this tangent is π times the distance.

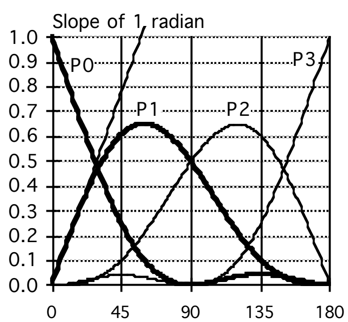
Basis Matrix

	P_0	P_1	P_2	P_3	
$SIN(\pi t)*COS(\pi t)$	-1	1	-1	1	1
$SIN(\pi t)$	-1	1	1	-1	---
$COS(\pi t)$	1	0	0	-1	2
1	1	0	0	1	

As t goes from 0 to 1 the curve goes from P_0, passes near P_1 and P_2 to P_3.

Piecewise loop parameters:
Start at the first control point P_0.
Step three at a time
End three from the last

This type needs four control points to interpolate a segment.
The segment spans the four control points.



Horizontal axis is t, not π * t.

Imitation of Cubic Bézier

Derivative of basis matrix showing sum of columns for determining the start and end tangents.

	P_0	P_1	P_2	P_3	
$COS(2\pi t)$	-1	1	-1	1	π
$COS(\pi t)$	-1	1	1	-1	---
$-SIN(\pi t)$	1	0	0	-1	2
0	1	0	0	1	

Sum of collumns.
t=0, $COS(2\pi t)=1$
$COS(\pi t)=1$
$-SIN(\pi t)=0$ -2 2 0 0 start tangent = $\pi*(P_1-P_0)$

t=1, $COS(2\pi t)=1$
$COS(\pi t)=-1$
$-SIN(\pi t)=0$ 0 0 -2 2 start tangent = $\pi*(P_3-P_2)$

Needs sample curves.