Common Cubic Curves

These curves are constructed using just addition and multiplication. However, because they require a *lot* of addition and multipication, they are summarised here in matrix notation.

1. Cubic Bezier curves

$$P(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} p_1 \\ p_{c1} \\ p_{c2} \\ p_2 \end{pmatrix}$$

   This defines a curve from p_1 (at t = 0) to p_2 (at t = 1), using control points p_{c1} and p_{c2}. The other curves are splines - which means you can join a series of points with a series of curves, and the whole thing will be nice and smooth. The curve $P_i(t)$ runs from p_i (at t=0) to p_{i+1} (at t = 1). p_{i-1} is where the previous curve starts and p_{i+2} is where the next curve ends. Beziers can easily be splined by offsetting p_{c1} from p_1 by the same amount and in the opposite direction from which the previous curve offset p_{c2}.

2. Uniform cubic B-splines

$$P_i(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \frac{1}{6} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{pmatrix} p_{i-1} \\ p_i \\ p_{i+1} \\ p_{i+2} \end{pmatrix}$$

3. Catmull Rom splines

$$P_i(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} -\tau & 2-\tau & \tau-2 & \tau \\ 2\tau & \tau-3 & 3-2\tau & -\tau \\ -\tau & 0 & \tau & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} p_{i-1} \\ p_i \\ p_{i+1} \\ p_{i+2} \end{pmatrix}$$

   0.5 is said to be a good value for the curve-tightness τ to start with when playing with Catmull Rom splines.

Calculating Beziers from known points on curve

Here's the formula for a cubic Bezier curve

$$v(t) = v_1(1-t)^3 + 3v_{c1}(1-t)^2t + 3v_{c2}(1-t)t^2 + v_2t^3$$

The curve starts at v_1 and ends at v_2. Given two additional known points on the curve $v(t_a)$ and $v(t_b)$, we can write this

$$\begin{pmatrix} v_{c1} & v_{c2} \end{pmatrix} \begin{pmatrix} 3(1-t_a)^2t_a & 3(1-t_a)t_a^2 \\ 3(1-t_b)^2t_b & 3(1-t_b)t_b^2 \end{pmatrix} = \begin{pmatrix} v(t_a) - v_1(1-t_a)^3 - v_2t_a^3 \\ v(t_b) - v_1(1-t_b)^3 - v_2t_b^3 \end{pmatrix}$$

Which means we can compute our Bezier's control points v_{c1} and v_{c2} by performing Guassian elimination

$$\left(\begin{array}{cc|c} 3(1-t_a)^2t_a & 3(1-t_a)t_a^2 & v(t_a) - v_1(1-t_a)^3 - v_2t_a^3 \\ 3(1-t_b)^2t_b & 3(1-t_b)t_b^2 & v(t_b) - v_1(1-t_b)^3 - v_2t_b^3 \end{array}\right)$$