Alexander M. Nicoara

CS 411: Computer Graphics

Professor Gady Agam

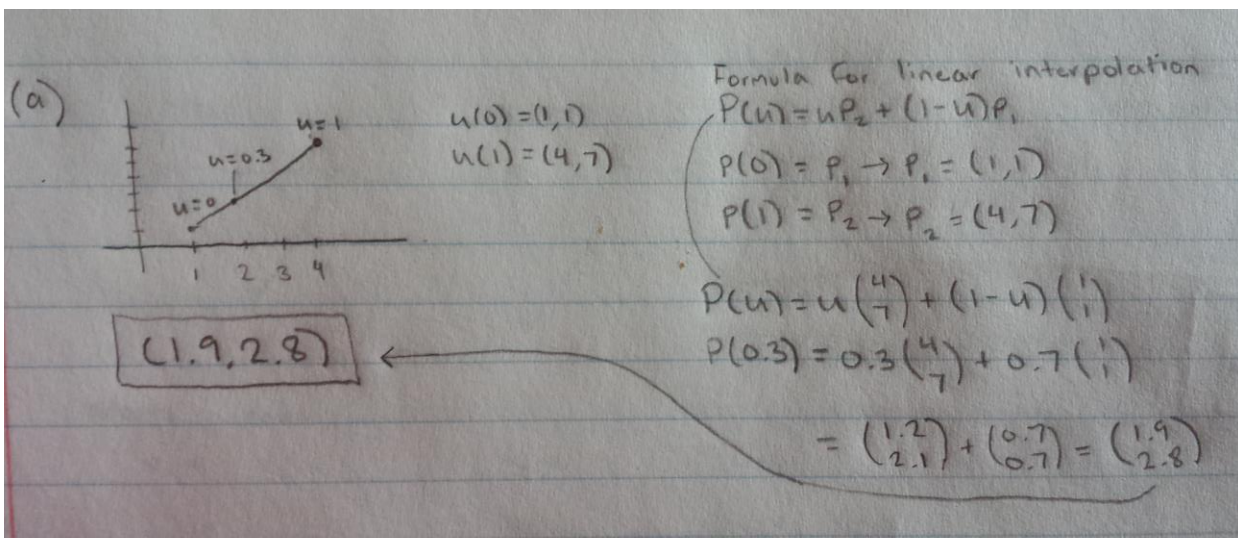
Assignment 3: Curve and Surface Interpolation

Introduction

This assignment explores the topic of curve and surface interpolation that was taught in class a few weeks back. In the first part of the assignment, I will be using interpolation techniques to solve the assigned problems that will include different types of splines, curves, and lines. For the second part of this assignment, I will be using WebGL to create a program that will be using interpolation to draw a path of curve segments (using generated points from mouse clicks) for an object to travel on. Some of the properties of this program will feature tension (how tight the curves are), speed, and toggling animation.

Questions

(a) Let (1,1) (4,7) be two points. Assume an interpolation parameter u = 0 at (1,1) and u = 1 at (4,7). Find the coordinates of the 2D point between them using linear interpolation with an interpolation parameter of u = 0.3.



(b) Explain the difference between parametric and geometric continuity. Explain the advantages and challenges that are associated with piecewise interpolation.

The difference between parametric and geometric continuity is that:

- **In parametric continuity, the <u>magnitude and direction must be the same</u> for the ending point of a curve P at P'(1) and at the starting point of another curve Q at Q'(0) (for connected curves P and Q). (ie. P'(1) = Q'(0))**
- **However, in geometric continuity, <u>only the direction must be the same</u> for the ending point of a curve P at P'(1) and at the starting point of another curve Q at Q'(0), but the magnitude is different. (ie. P'(1) = aQ'(0) for some positive constant a)**

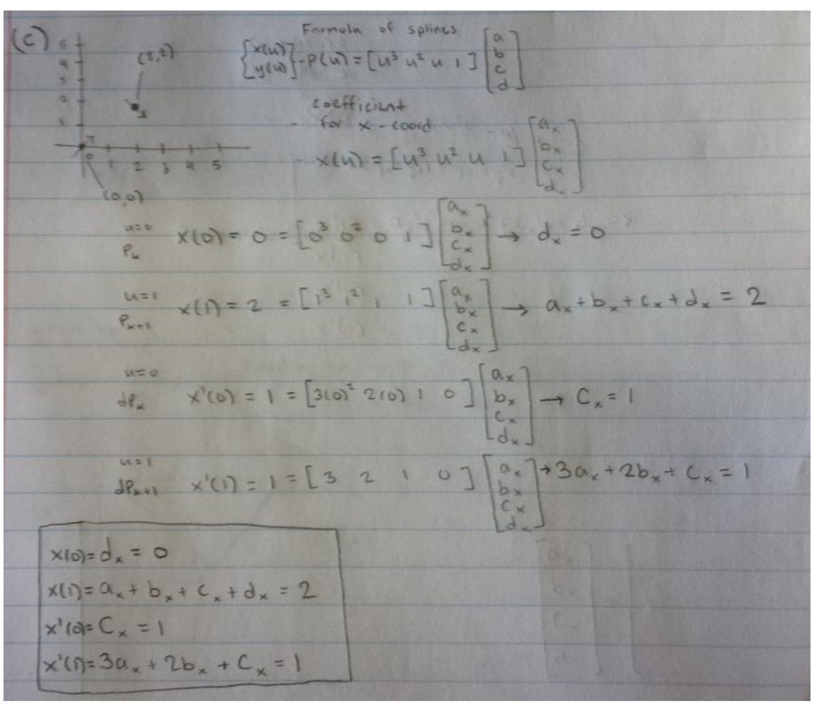
Advantages of piecewise interpolation:

- **A large number of data points can be connected together using low degree polynomials.**
- **It's faster to use (than high degree interpolation) and the nodes adapt to the behavior of the function we are trying to draw.**

Challenges of piecewise interpolation:

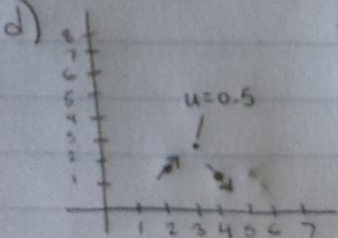
- **The piecewise functions are not always smoothly connected (they have "kinks" or connecting points with different tangents).**

(c) Given the points (0,0) (2,2) and corresponding tangents (1,1) (1,-1) respectively, write the 4 constraint equations that are used to compute the Hermite interpolation coefficients for the x coordinate of the interpolation curve between the points.



(d) Given the 2D control points (2,2)(4,2) with a tangent of (1,1) at the point (2,2) and a tangent of (1,-1) at the point (4,2), compute the coordinate of the point at parameter u = 0.5 using Hermite splines. Assume a parameter of u = 0 at (2,2) and of u = 1 at (4,2). Use the matrix form for the computations.

d)



$P(0) = (2,2) \quad P'(0) = (1,1) \quad u = 0$

$P(1) = (4,2) \quad P'(1) = (1,-1) \quad u = 1$

<u>find u = 0.5</u>

$$x(0.5) = \begin{bmatrix} 0.5^3 & 0.5^2 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5^3 & 0.5^2 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 2\cdot 2 + 4\cdot -2 + 1\cdot 1 + 1\cdot 1 \\ 2\cdot -3 + 4\cdot 3 + 1\cdot -2 + 1\cdot -1 \\ 0 + 0 + 1\cdot 1 + 0 \\ 2\cdot 1 + 0 + 0 + 0 \end{bmatrix} = \begin{bmatrix} 0.5^3 & 0.5^2 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 3 \\ 1 \\ 2 \end{bmatrix}$$

$$= (0.5)^3(-2) + (0.5)^2(3) + (0.5)(1) + (1)(2)$$

$$= -\frac{1}{4} + \frac{3}{4} + \frac{2}{4} + \frac{8}{4} = \frac{12}{4} = 3$$

$$y(0.5) = \begin{bmatrix} 0.5^3 & 0.5^2 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0.5^3 & 0.5^2 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -1 \\ 1 \\ 2 \end{bmatrix}$$

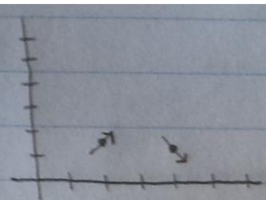
$$= \cancel{(0.5)^3(0)} + (0.5)^2(-1) + (0.5)(1) + (1)(2)$$

$$= -\frac{1}{4} + \frac{2}{4} + \frac{8}{4} = \frac{9}{4} = 2.25$$

$$\boxed{P(0.5) = (3, 2.25)}$$

(e) Repeat the previous question but this time using the blending function form of the computations.

(e)



$P(0) = (2,2) \quad P'(0) = (1,1) \quad u=0$

$P(1) = (4,2) \quad P'(1) = (1,-1) \quad u=1$

Find $u=0.5$ using Blending functions

$$P(u) = H_0(u)P_k + H_1(u)P_{k+1} + H_2(u)dP_k + H_3(u)dP_{k+1}$$

$$H_0(u) = [u^3 \; u^2 \; u \; 1]\begin{bmatrix}2\\-3\\0\\1\end{bmatrix} = 2u^3-3u^2+1 \qquad H_0(0.5) = 2(0.5)^3 - 3(0.5)^2 + 1 = 0.5$$

$$H_1(u) = [u^3 \; u^2 \; u \; 1]\begin{bmatrix}-2\\3\\0\\0\end{bmatrix} = -2u^3+3u^2 \qquad H_1(0.5) = -2(0.5)^3+3(0.5)^2 = 0.5$$

$$H_2(u) = [u^3 \; u^2 \; u \; 1]\begin{bmatrix}1\\-2\\1\\0\end{bmatrix} = u^3-2u^2+u \qquad H_2(0.5) = (0.5)^3-2(0.5)^2+0.5 = 0.125$$

$$H_3(u) = [u^3 \; u^2 \; u \; 1]\begin{bmatrix}1\\-1\\0\\0\end{bmatrix} = u^3-u^2 \qquad H_3(0.5) = 0.5^3 - 0.5^2 = -0.125$$

$$P(0.5) = 0.5\begin{pmatrix}2\\2\end{pmatrix} + 0.5\begin{pmatrix}4\\2\end{pmatrix} + 0.125\begin{pmatrix}1\\1\end{pmatrix} - 0.125\begin{pmatrix}1\\-1\end{pmatrix}$$

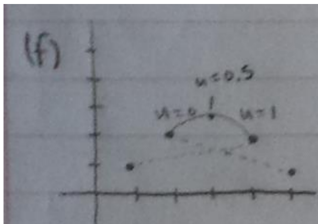
$x(0.5) = 1 + 2 + 0.125 - 0.125 = 3$

$y(0.5) = 1 + 1 + 0.125 + 0.125 = 2.25$

$$\boxed{P(0.5) = (3, 2.25)}$$

(f) Given a set 2D control points (1,1)(2,2)(4,2)(5,1) with parameter u = 0 at (2,2) and parameter u = 1 at (4,2) find the coordinate of the point at u = 0.5 when using Cardinal splines. Use the matrix form for the computations. Assume a tension parameter of 0.5.

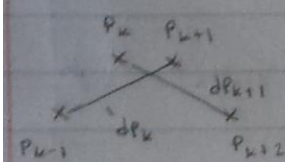
(f)



Control Points

(1,1) (2,2) (4,2) (5,1)   Tension = 0.5

u=0 u=1



Find u = 0.5

$$P(u) = [u^3\ u^2\ u\ 1]\, M_c \begin{bmatrix} P_{k-1} \\ P_k \\ P_{k+1} \\ P_{k+2} \end{bmatrix}$$

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

$$\downarrow$$

$$\begin{Bmatrix} x(0.5) \\ y(0.5) \end{Bmatrix}$$

$$x(0.5) = [0.5^3\ 0.5^2\ 0.5\ 1] \begin{array}{c} \begin{array}{cccc} -1/2 & 3 & -6 & 2.5 \end{array} \\ \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 5 \end{bmatrix}$$

$$= [0.5^3\ 0.5^2\ 0.5\ 1] \begin{bmatrix} -0.5\cdot 1 + 1.5\cdot 2 + (-1.5\cdot 4) + 0.5\cdot 5 \\ 1\cdot 1 + -2.5\cdot 2 + 2\cdot 4 + -0.5\cdot 5 \\ -0.5\cdot 1 + 0 + 0.5\cdot 4 + 0 \\ 0 + 2 + 0 + 0 \end{bmatrix}$$

$$= [0.5^3\ 0.5^2\ 0.5\ 1] \begin{bmatrix} -1 \\ 1.5 \\ 1.5 \\ 2 \end{bmatrix} = -(0.5)^3 + (0.5)^2(1.5) + (0.5)(1.5) + (1)(2)$$

$$= -0.125 + 0.375 + 0.75 + 2$$

$$= 3$$

$$y(0.5) = [0.5^3\ 0.5^2\ 0.5\ 1] \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

$$= [0.5^3\ 0.5^2\ 0.5\ 1] \begin{bmatrix} 0 \\ -0.5 \\ 0.5 \\ 2 \end{bmatrix} = (0.5)^2(-0.5) + (0.5)(0.5) + (1)(2)$$

$$= -0.125 + 0.25 + 2$$

$$= 2.125$$

$$\boxed{P(0.5) = (3, 2.125)}$$

(g) Repeat the previous question but this time using the blending function form.

(g) $P_{k-1} = (1,1)$ $\quad T = 0.5$

$P_k = (2,2)$ $\quad u = 0$

$P_{k+1} = (4,2)$ $\quad u = 1$

$P_{k+2} = (5,1)$

$$M_c = \begin{bmatrix} -s & 2-s & s-2 & s \\ 2s & s-3 & 3-2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

BLENDING FUNCTION

$$P(u) = C_0(u)P_{k-1} + C_1(u)P_k + C_2(u)P_{k+1} + C_3(u)P_{k+2}$$

Find $P(0.5)$

$$C_0(u) = -0.5u^3 + u^2 - 0.5u$$
$$= -0.5(0.5)^3 + (0.5)^2 - 0.5(0.5) = -0.0625$$

$$C_1(u) = 1.5u^3 - 2.5u^2 + 1$$
$$= 1.5(0.5)^3 - 2.5(0.5)^2 + 1 = 0.5625$$

$$C_2(u) = -1.5u^3 + 2u^2 + 0.5u$$
$$= -1.5(0.5)^3 + 2(0.5)^2 + 0.5(0.5) = 0.5625$$

$$C_3(u) = 0.5u^3 - 0.5u^2$$
$$= 0.5(0.5)^3 - 0.5(0.5)^2 = -0.0625$$

$$P(0.5) = -0.0625\begin{pmatrix}1\\1\end{pmatrix} + 0.5625\begin{pmatrix}2\\2\end{pmatrix} + 0.5625\begin{pmatrix}4\\2\end{pmatrix} - 0.0625\begin{pmatrix}5\\1\end{pmatrix}$$

$$x(0.5) = -0.0625(1) + 0.5625(2) + 0.5625(4) - 0.0625(5)$$
$$= 3$$

$$y(0.5) = -0.0625(1) + 0.5625(2) + 0.5625(2) - 0.0625(1)$$
$$= 2.125$$

$$\boxed{P(0.5) = (3, 2.125)}$$

(h) Given a set 2D control points (1,1)(2,2)(4,2)(5,1) with parameter u = 0 at (1,1) and parameter u = 1 at (5,1) find the coordinate of the point at u = 0.5 when using Cubic Bezier curves. Use the matrix form for the computations.

(h) Control points:

(1,1) (2,2) (4,2) (5,1)

u=0 u=1

$$M_B = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Find P(0.5)

$$P(u) = [u^3 \ u^2 \ u \ 1] M_B \begin{bmatrix} P_{k-1} \\ P_k \\ P_{k+1} \\ P_{k+2} \end{bmatrix}$$

$$\begin{Bmatrix} x(0.5) \\ y(0.5) \end{Bmatrix}$$

$$x(0.5) = [(0.5)^3 \ (0.5)^2 \ (0.5) \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 5 \end{bmatrix}$$

$$= [0.5^3 \ 0.5^2 \ 0.5 \ 1] \begin{bmatrix} -2 \\ 3 \\ 3 \\ 1 \end{bmatrix}$$

$$= (-2)(0.5)^3 + (3)(0.5)^2 + (3)(0.5) + (1)(1)$$

$$= -\frac{1}{4} + \frac{3}{4} + \frac{6}{4} + \frac{4}{4} = \frac{12}{4} = 3$$

$$y(0.5) = [(0.5)^3 \ (0.5)^2 \ (0.5) \ 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

$$= [0.5^3 \ 0.5^2 \ 0.5 \ 1] \begin{bmatrix} 0 \\ -3 \\ 3 \\ 1 \end{bmatrix}$$

$$= 0.5^2(-3) + (0.5)(3) + (1)(1) = -\frac{3}{4} + \frac{6}{4} + \frac{4}{4} = \frac{7}{4} = 1.75$$

$$\boxed{P(0.5) = (3, 1.75)}$$

(i) Repeat the previous question but this time using the blending function form.

(i)

$$P(u) = \sum_{i=0}^{3} B_i^3(u) P_i$$

Find $P(0.5)$

$B_0^3(u) = (1-u)^3$ $\qquad$ $B_0^3(0.5) = (1-0.5)^3 = 0.5^3 = 0.125$

$B_1^3(u) = 3u(1-u)^2$ $\quad \Rightarrow \quad$ $B_1^3(0.5) = 3(0.5)(1-0.5)^2 = \frac{3}{2}(0.5)^2 = 0.375$

$B_2^3(u) = 3u^2(1-u)$ $\qquad$ $B_2^3(0.5) = 3(0.5)^2(0.5) = 0.375$

$B_3^3(u) = u^3$ $\qquad$ $B_3^3(u) = 0.5^3 = 0.125$

$$P(0.5) = 0.125\begin{pmatrix}1\\1\end{pmatrix} + 0.375\begin{pmatrix}2\\2\end{pmatrix} + 0.375\begin{pmatrix}4\\2\end{pmatrix} + 0.125\begin{pmatrix}5\\1\end{pmatrix}$$

$x(0.5)$

$y(0.5)$

$$x(0.5) = 0.125(1) + 0.375(2) + 0.375(4) + 0.125(5) = 3$$

$$y(0.5) = 0.125(1) + 0.375(2) + 0.375(2) + 0.125(1) = 1.75$$

$$\boxed{P(0.5) = (3, 1.75)}$$

(j) Assuming that we want to add another cubic Bezier curve segment that will connect to the cubic Bezier curve segment in the previous questions smoothly (with C1 continuity), compute the coordinates of the first control point in the second curve segment.

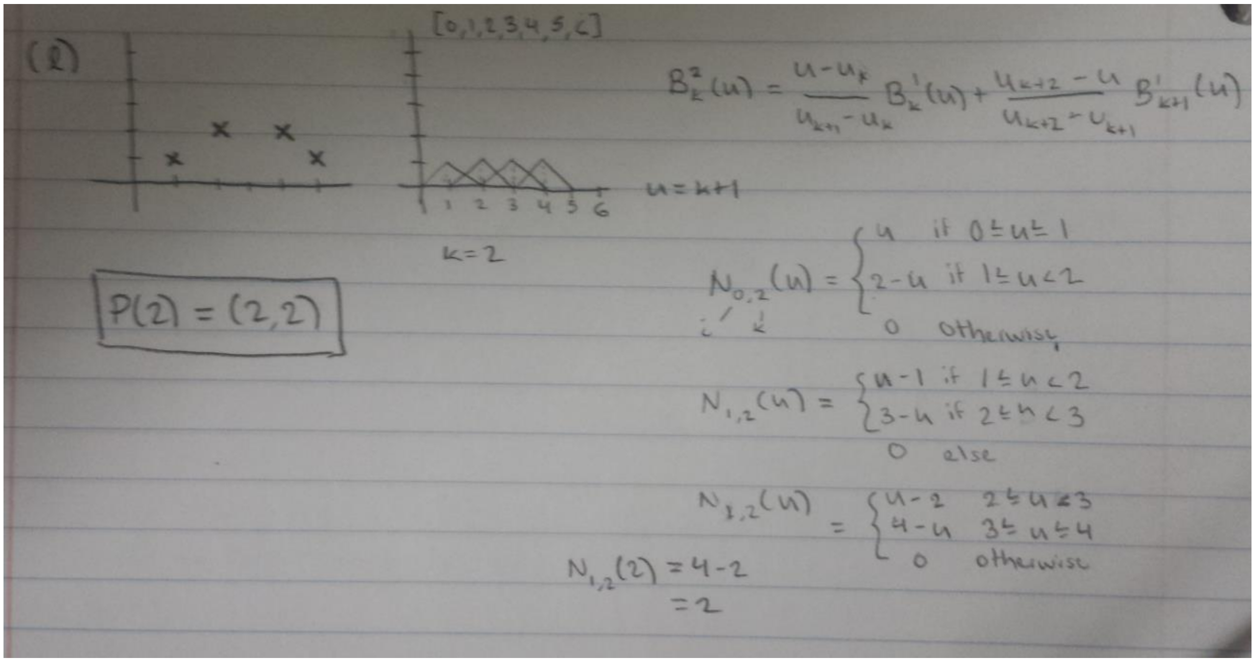
The first control point of the new Bezier curve must be (5,1) since Bezier curves pass through their first and last points. The previous curve ends at these coordinates and must start here in order to be passed through in the new segment. For the sake of C_1 continuity, the point (5,1) sits on the tangent line formed by the last 2 points of the previous Bezier curve, which is needed so that the first slope of the new curve is equal to the ending slope of the previous curve.

Answer: (5,1)

(k) Explain the advantage of the Bezier curve blending functions (Bernstein polynomials).

Blending functions eliminate the need to use tangents.

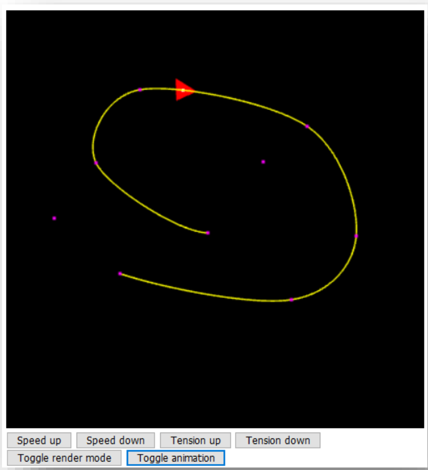
(l) Given a set 2D control points (1,1)(2,2)(4,2)(5,1) and a knot vector [0,1,2,3,4,5,6] find the coordinate of the point at u = 2 when using uniform quadratic B-splines.



(m) *Did not attempt*

WebGL Program

Introduction

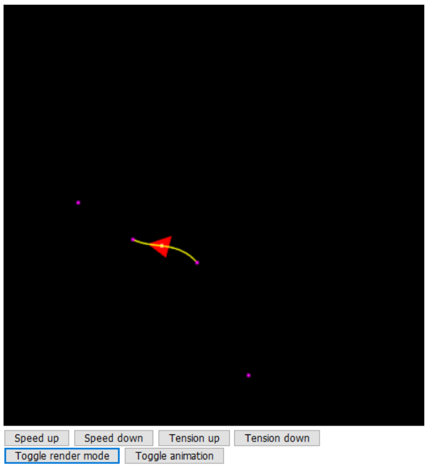


In the second part of the assignment, I was tasked with creating a WebGL program that would draw cardinal splines and curves using the coordinates from mouse clicks. In this program, the curves you generate are a path for an object (the red triangle) to follow, and you can change properties such as object speed (how fast the triangle moves), the curve tension (which either constricts the curve or smooths it out), and toggle the animation (pause/play).

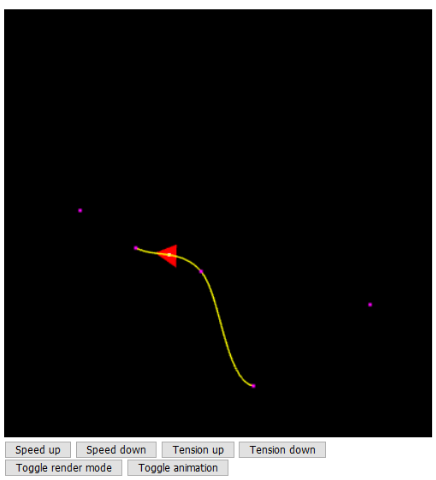
How does it work?

To use the program, you first need to click 4 times (preferably in random locations) to generate 4 control points in order to generate the cardinal spline. You must have at least one curve to use as a path for the object to travel along.

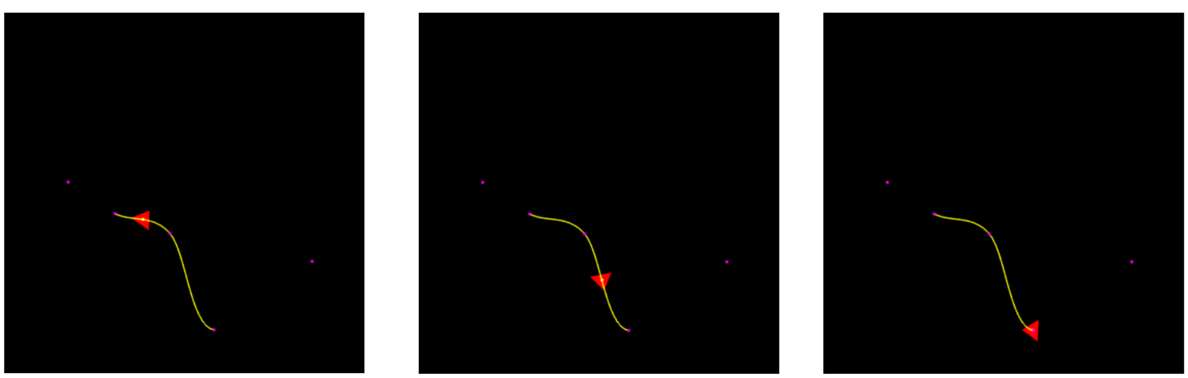
If you want to spawn the object that moves along the curve, press Toggle render mode to generate the red triangle object.



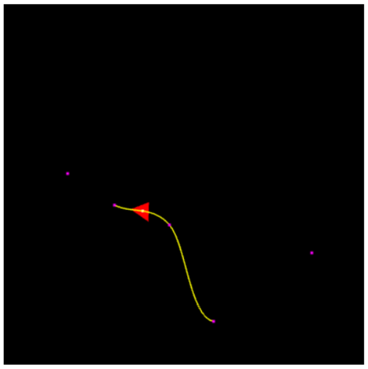
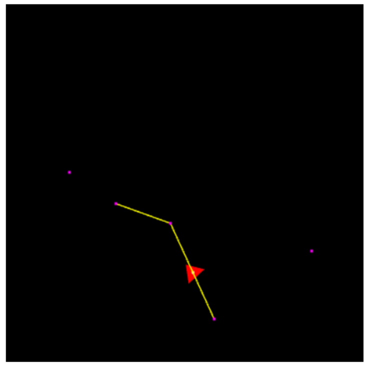
To add curves, click anywhere on the canvas. This will add another control point and thus another curve.



To adjust the speed of the object, press either Speed Up/Speed Down button to make the object travel faster or slower along the path.

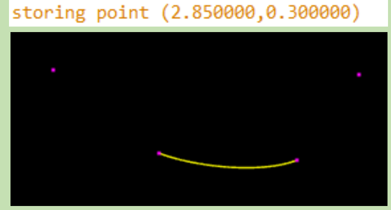
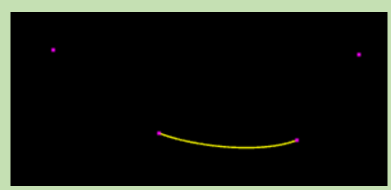
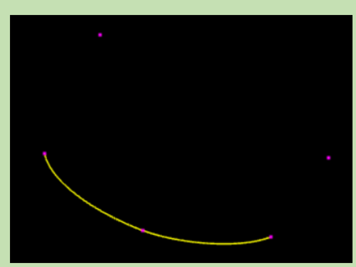


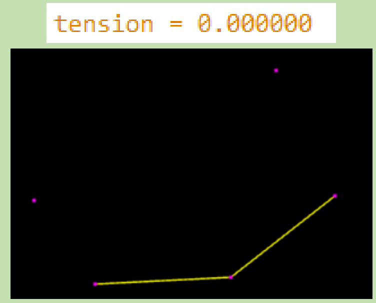
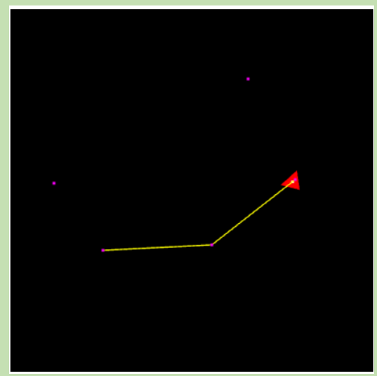
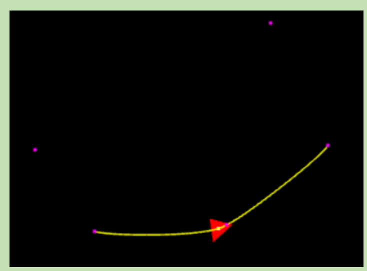
To modify the tension parameter, press either Tension Up/Tension down. This will redraw all of the curves the program generated using the same control points. The path will change more (appear to look either constricting or expanding) the more you alter the tension parameters.

(Before and after when clicking the Tension down button a few times)

Test cases

Case	Expected Result	Pass/Fail
The program stores and displays clicked locations as points.	The program stores the mouse click locations into the control points array and draws purple dots indicating the points.	storing point (-3.800000,0.200000) storing point (-2.450000,-1.650000) storing point (0.550000,-1.500000) storing point (2.850000,0.300000)  **Pass**
Upon receiving 4 points, draw a cardinal spline between the middle 2.	After receiving 4 points, the program will draw a yellow line representing the cardinal spline onto the canvas.	 **Pass**
Upon receiving any additional points, a curve is generated while preserving C1 continuity.	After receiving 4 points and then after that another point, a new yellow curve segment is drawn onto the canvas smoothly attaching to the previous curve.	 **Pass**

When either Tension Up or Tension Down is pressed, the curve is updated with new calculations from the new value of the tension parameter.	The program changes the global variable ‘tension’ and empties the intrPts array. The interpolation function is called using all the same control points and the new tension value. After that, the new curves are drawn and will either appear to constrict or expand.	 **Pass**
When the toggle animation button is pressed, the object stops moving. When pressed again, the object begins moving again.	The program halts the movement of the red triangle when the toggle animation button is pressed. When the toggle animation button is pressed again, the red triangle resumes moving.	 **Pass**
When speed up is pressed, the object’s speed increases. When speed down is pressed, the object’s speed decreases.	The program will change the speed of the red triangle (either increasing or decreasing depending on the speed button pressed).	 **Pass**

Assignment Conclusion

Challenges and how I overcame them

While doing this assignment, one of the challenges I faced was making sure everything ran correctly in my WebGL program. Initially, my cardinal spline interpolation algorithm did not compute the correct coordinates, so I had to trace back and find out what went wrong in my calculations. It appeared to be that I did not order my control points correctly, which resulted in a miscalculation. After ordering them pk-1, pk, pk+1, pk+2, my program was able to compute the correct coordinates. Another issue I faced in my WebGL program was obtaining the correct points needed to redrawing the curves. At first this was really frustrating because the ctrlPts array consisted of both x and y values in individual indexes for all the coordinates, so I had to find a way to traverse through the array in order to obtain the correct coordinates I needed to use as parameters for the interpolation function. After looking at the main function's method of indexing and storing values into the ctrlPts array, I knew what was going on and how to correctly reference the x and y values for the coordinates.

Summary

In conclusion, this assignment really helped me understand the process of interpolation and how curves are parametrized and drawn. At first, this assignment was a challenge for me because I knew very little about splines and curves prior, but afterwards I feel much more confident and have a firmer understanding about them. Doing both the word problems and the WebGL program allowed me to learn both mathematically and visually, which was also very helpful.