Title: Length of curve experiment

Background: The motivation behind the experiment is severalfold.

- Recreate an *equivalent* (not same) Runge's numerical experiment that showed that higher-order interpolation is a bad idea. As shown in class, as you use higher order interpolants to approximate $f(x) = \frac{1}{1+25x^2}$ in the domain $[-1,1]$, the differences between the original function and the polynomial interpolant became bigger.
- Show why we use spline interpolation rather than polynomial interpolation to find a smooth path to travel through several discrete data points, such as in the example of the path of a robot (Refer to Chapter 05.10 for further information).
- Develop programming skills related to numerical methods.

What to do to set it up: A flexible curve (Figure 1) made of lead-core construction with inch graduations from 0 to $\mathbf{12}$" is provided. Using the flexible curve, trace a $\mathbf{12}$" length curve similar (not same) in shape to the Runge's curve on the graphing paper as shown.

Items needed: A charged laptop, MATLAB loaded on the laptop, ruler, three sheets of graph paper, pencil, eraser.

Exercises to do:

1. Write your last name and first name on the graphing paper.
2. Put the graphing paper in landscape mode. Draw the *x*-axis and *y*-axis as you please. Putting the *x*-axis close to the bottom and the *y*-axis in the middle is an option chosen by most. See Figure 1 only for illustrative purposes for coordinates.
3. Trace a $\mathbf{12}$" curve (mind the scale – the flexible curve is longer than $\mathbf{12}$") in a similar shape as Runge's curve using the flexible curve.
4. Pick 1 point at each of the two ends of the curve.
5. Pick 6 more points distributed throughout the curve. They do not have to be equally spaced.
6. Find and mark the $(\boldsymbol{x}, \boldsymbol{y})$ coordinates of the 8 data points on the plot in decimal notation. See Figure 1.
7. On the graph paper, do the following
    a. Using the ruler, draw piecewise linear polynomials (connecting the dots) through the 8 data points.
    b. Use the ruler to measure the length of the piecewise linear polynomials to estimate the length of the curve.
    c. Put the length of each piecewise linear polynomial on the curve.
    d. Put the length of the curve on your graph paper as "Estimated length of the curve using a ruler for linear splines = _____________". Show your calculation.

8. Open a new MATLAB mfile. Put your first name and last name as comments.

9. Put %% Problem 9 as a comment. Use the coordinates of the 8 data points and estimate the length of the curve using linear spline interpolation and the concept of loops in MATLAB. Display the length of the curve using the *disp* statement. You can use the length of a straight-line equation to do this. How close is your answer to 12″? How far is your answer from that of Problem 7 that you measured by hand?

10. Put %% Problem 10 as a comment. Find the polynomial interpolant that passes through the 8 data points using MATLAB functions such as *polyfit*. **There is no need to display the polynomial expression**.

11. Put %% Problem 11 as a comment. Calculate the length of the polynomial interpolant from Problem 10 using MATLAB functions such as *vpaintegral* and *diff*. Display the length of the curve using the disp statement. How close is your answer to 12″? How far is your answer from those of Problem 9?

    Hint: $S = \int_a^b \sqrt{1 + (dy/dx)^2}\, dx$ is the formula for the length of a curve $y(x)$ from a to b.

12. Put %% Problem 12 as a comment. Calculate the length of the above polynomial interpolant from Problem 10 using MATLAB by breaking the curve into small-length straight lines and finding their lengths. Display the length of the curve using the *disp* statement. How close is your answer to 12″? How far is your answer from those of Problem 9?

13. Put %% Problem 13 as a comment. Find the cubic spline interpolant that passes through the 8 data points using MATLAB functions such as *spline*. **There is no need to display the spline expression**. Calculate the length of the cubic spline interpolant using MATLAB by any scientific method of your choice. Display the length of the curve using the disp statement. How close is your answer to 12″? How far is your answer from those of Problems 9, 11, and 12?

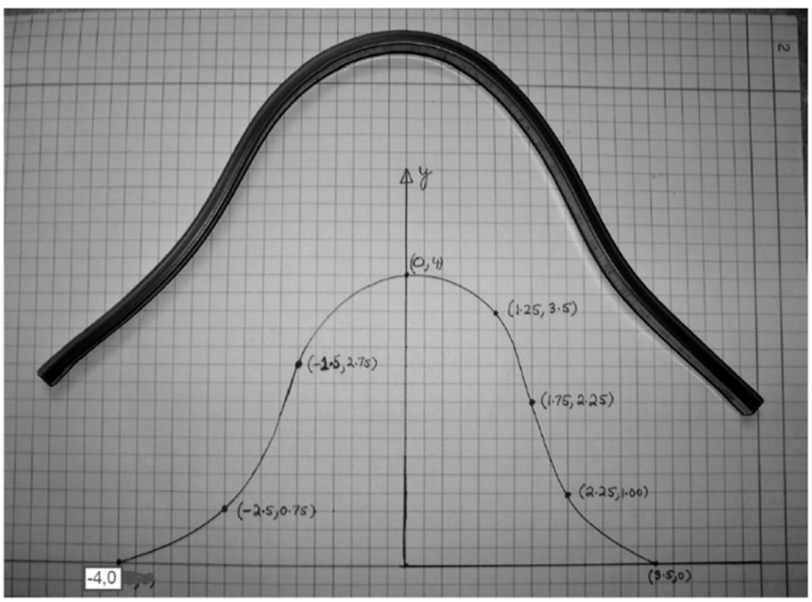

Figure 1. Using a flexible curve to draw a curve of known length. This figure only shows a sample curve – do not use these coordinates. Your curve should have different coordinates and have 8 points chosen, including the endpoints.

14. Bonus Question Who Want a Challenge or Finish Early. Not Required for the Course.

    The cubics found by MATLAB is of a different form as given. For example, given $(x_1, y_1), \ldots (x_n, y_n)$, the first cubic of the $n-1$ cubics is given as $a(x-x_1)^3 + b(x-x_1)^2 + c(x-x_1) + d$, where *a, b, c, d* are the coefficients in the first row of the output matrix, *p* of the *p=spline*(*x,y*) MATLAB command, and so on. You can use this information to at least calculate the integrand exactly of the $L = \int_a^b \sqrt{1 + (dy/dx)^2}$ integral. Then use *vpaintegral* over each cubic domain to calculate length of each cubic. The addition of all the lengths would be the length of the cubic spline. Display the length of the curve using the disp statement. How close is your answer to 12″? How far is your answer from those of Problems 9, 11, 12 and 13?