Path And Surface Specification

David M. Auslander

Mechanical Engineering
University of California at Berkeley

September 23,2007

Object Dimensionality in 3D Cartesian Space

- For simplicity, consider 3D Cartesian only; however, the concepts generalize (see below!)
- All objects are characterized by a location and other parameters
- 0D Object: a point. Location, no other parameters
- 1D Object: a curve. Location, distance along the curve Special case - a line.
- 2D Object: a surface. Location, two axes on the surface Special case - a plane
- While distance along a curve is an obvious parameter, the axes used on a surface are anything but obvious!

Applications in Mechanical Control

- Points: motion end point
- Curves: robot motion for painting, welding, etc.; machine tool path; robot obstacle avoidance; coordinate measurement (CMM) approach path; vehicles on a track
- Surfaces: machining; free-roaming vehicles (not on a track); navigation on earth

Higher Dimensional Spaces

- Seems abstract, but let's look at typical robots
- To control position and orientation takes 6 degrees-of-freedom (DOF)
- Thus, a 6th dimensional space
- Point-to-point move requires specification of 6 coordinates
- General robot to do this must have 6 independent axes
- Scorbots in lab are 5 DOF

Mechanical Control: The Supervisor

- A Supervisor is a module that gives setpoints to feedback control modules
- To move along a path, it must represent the path internally and feed appropriate setpoints to several modules controlling physical motions
- This leads to two situations:
  - Close (tight) coordination: One supervisor giving setpoints to several axes
  - Loose coordination: One supervisor for each axis
- Close coordination is used where there is a strict specification on how far the motion can deviate from the desired path (following error)
- Loose coordination works where the motion must only be roughly coordinated, for example, all axes should start and end together.

A Convenient Form to Represent Curves

- Focus on curves — most important for motion control
- Example: $x^2 + y^2 = r^2$ is a circle on the XY plane of an XYZ space
- It's correct but not very useful.
- For motion control, we generally know a starting point and need to find subsequent positions and velocities along the desired curve
- That is, how do we "walk" the curve?

Parametric Circle

- Strategy: find explicit equations to compute x, y, and z
- How? Use a "parametric variable"

$$x = \cos(s)$$
$$y = \sin(s)$$
$$z = 0$$

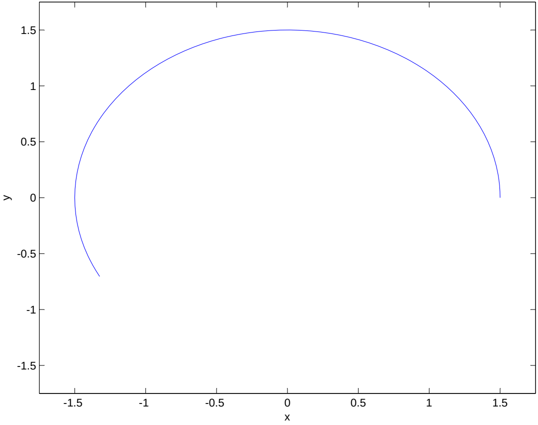
- "s" is the parametric variable
- But, it doesn't have much meaning relative to the path (it's the angle in radians but that's unique to circles) and constant radius
- Easy to convert to distance on the path:

$$x = r\cos(\frac{d}{r})$$
$$y = r\sin(\frac{d}{r})$$
$$z = 0$$

Example

Radius = 1.5
Draw circle segment from
d = 0 to d = 5.5

- Issues:
- Always starts at y=0
- Center always at origin



Curves as Tables

- Not all curves of interest have analytical representations
- Any curve can be approximated by a set of arrays of values
- The issue is how to get values between those in the table
- Interpolation - many methods!
- 0-th order: enough points in the table so that it is OK to just use the nearest
- 1-st order: use linear equation between points
- n-th order: various polynomial methods including cubic spline, NURBS
- FFT: uses Fourier transform, fills in additional points, then does inverse FFT

Interpolation Considerations: Smoothness

Positional continuity (G0) holds whenever the end positions of two curves or surfaces are coincidental. The curves or surfaces may still meet at an angle, giving rise to a sharp corner or edge and causing broken highlights.

Tangential continuity (G1) requires the end vectors of the curves or surfaces to be parallel, ruling out sharp edges. Because highlights falling on a tangentially continuous edge are always continuous and thus look natural, this level of continuity can often be sufficient.

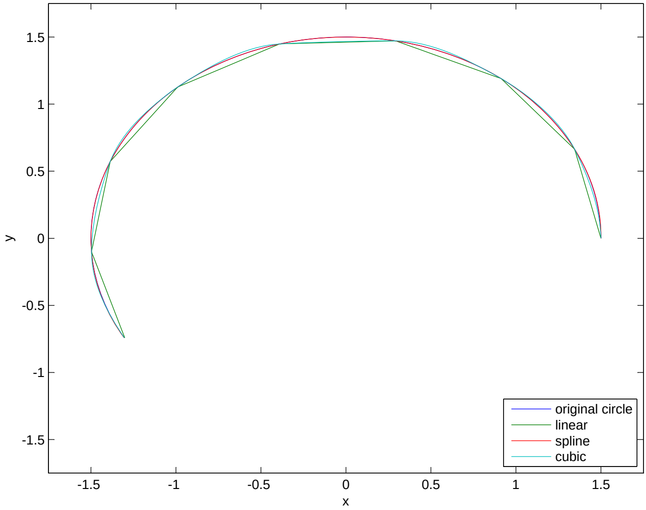
Curvature continuity (G2) further requires the end vectors to be of the same length and rate of length change. Highlights falling on a curvature-continuous edge do not display any change, causing the two surfaces to appear as one. This can be visually recognized as perfectly smooth.

From: `http://en.wikipedia.org/wiki/NURBS`

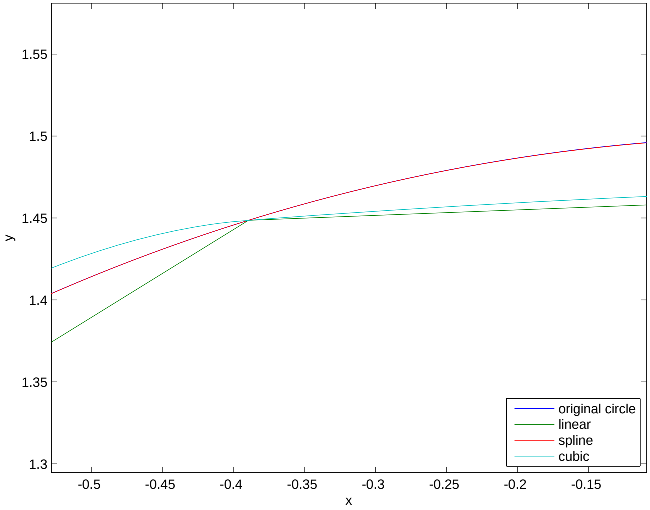
Other Interpolation Considerations

- Computational efficiency
- Memory usage
- Properties with various types of curves (this one can be very important because all of the methods have "interesting" vagaries)

Segment of a circle example with several interpolators using Matlab (blow-up of top section on next slide)



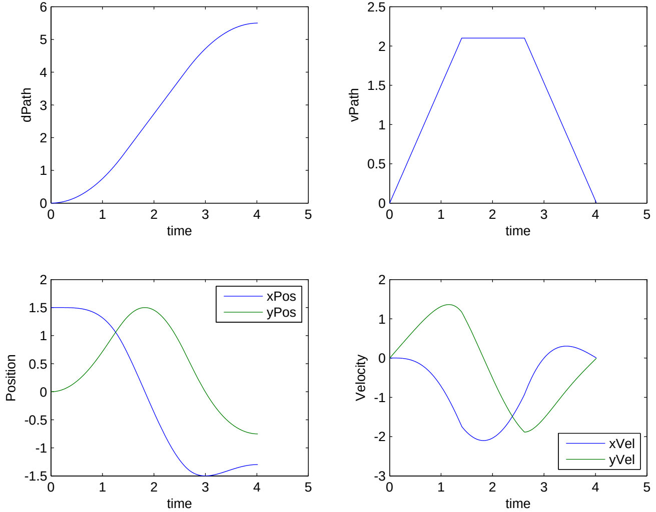
Blow-up of top part of the circle



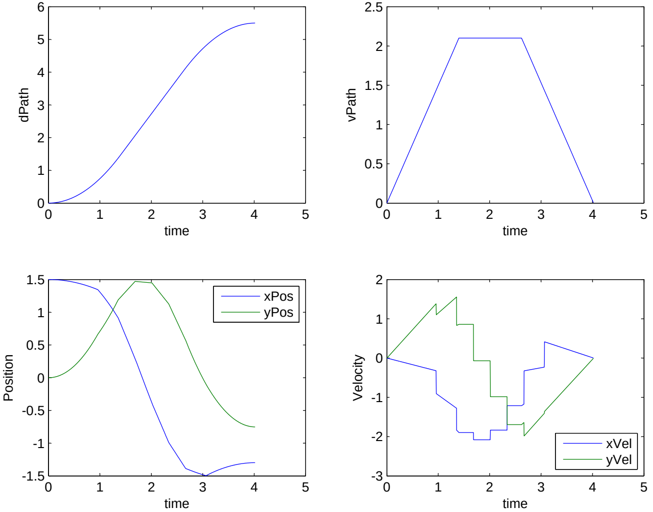
The Supervisor, Continued

- Transform to proper coordinate frame to get mathematical description to match starting position-orientation of the target object
- Establish a proper profile for motion along the curve
- Determine whether velocities for the individual axes are within reasonable limits
- Must generate appropriate position and velocity setpoints to follow the defined curve

Setpoints for Circle Segment: Spline Interpolation-8



Setpoints for Circle Segment: Linear Interpolation-8



Setpoints for Circle Segment: Linear Interpolation-16