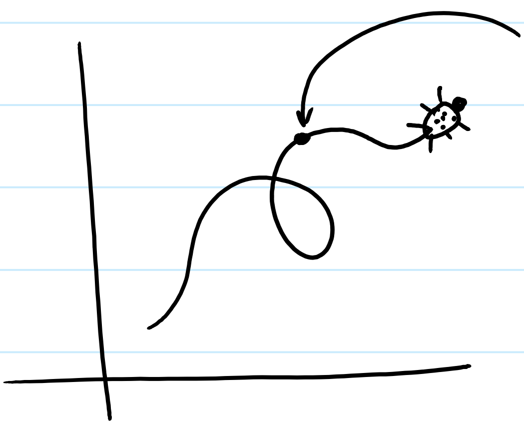
L10: Parametric Curves

Goal: To learn an alternate and more efficient representation for cubic splines.

Imagine a bug crawling around on a table... Oh, and you happen to have a stopwatch.



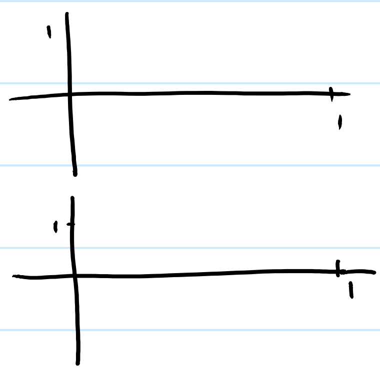
Location of bug at time t is $(x(t), y(t))$

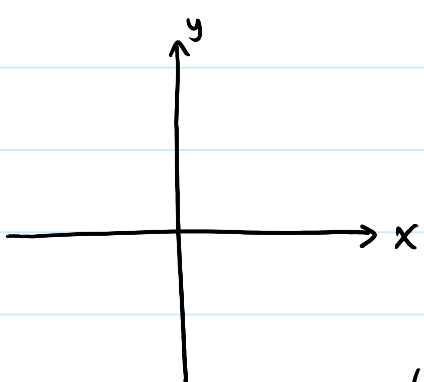
This a parametric curve, where t is the parameter.

A parametric curve is given by two (or more) functions $x(t)$ and $y(t)$ of a common parameter t, $a \le t \le b$.
In the case of the bug, t could be thought of as time.

e.g. Circle

$$x(t) = \cos(2\pi t)$$
$$y(t) = \sin(2\pi t)$$
$$0 \le t \le 1$$





t	
x	
y	

(Matlab demo - crawling_bug.m)

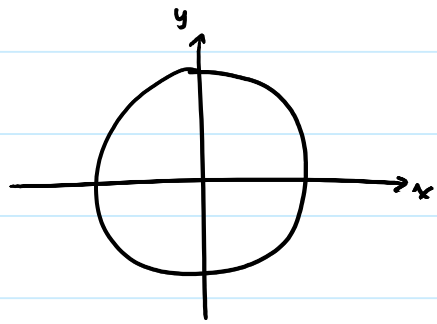
Different parameterizations can give the same parametric curve.

For example, the bug's speed could be different for different traversals of the same path. They bug could even speed up or slow down, but stay on the same curve.

Consider,

$$x(t) = \cos(2\pi t^4)$$
$$y(t) = \sin(2\pi t^4)$$

t	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	1
x	1	0.9997	0.9239	-0.4052	1
y	0	$-0.000\ldots$	0.3827	0.9142	0



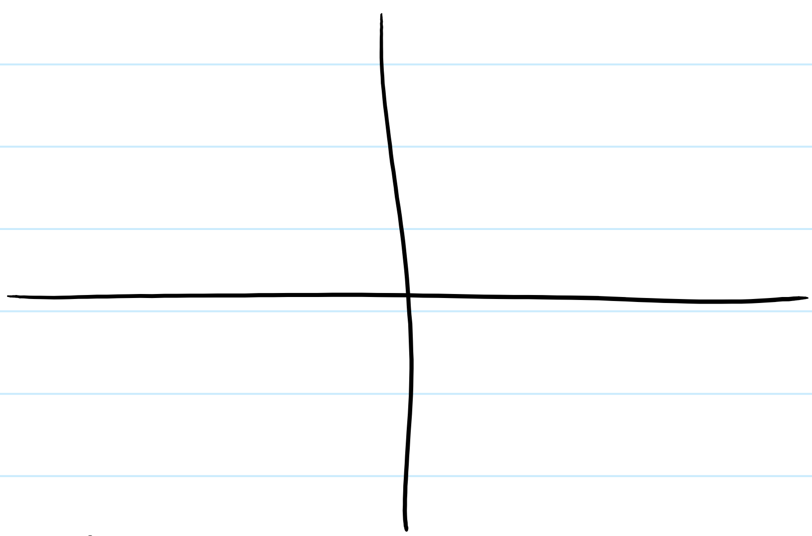
Same curve, different parameterization.

The Reverse Process

Given a curve, find and that approximately draw that curve.

Here's how we'll approach it.

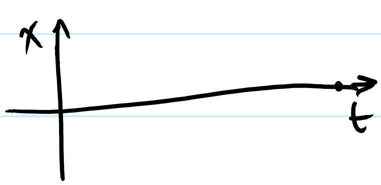
1) Draw the curve
2) Choose points
3) Assign parameter values

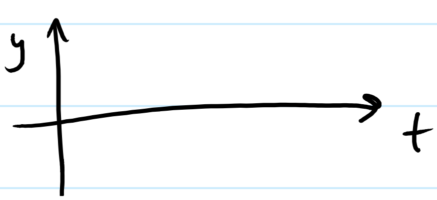


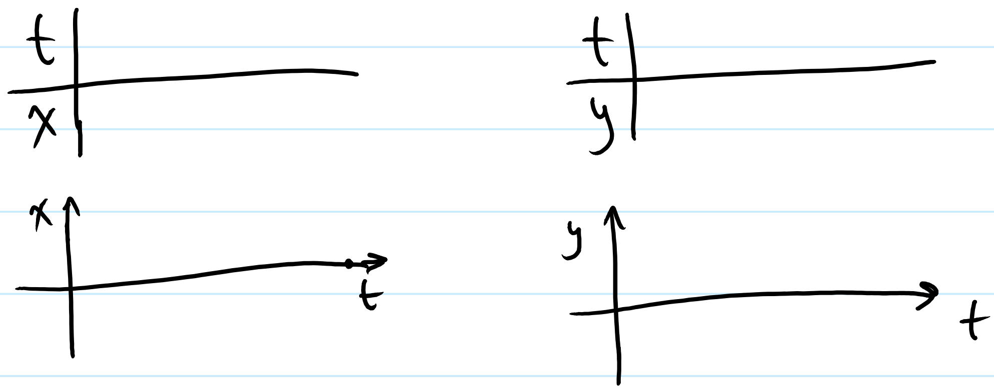
4) Separate x and y components

t			
x			

t			
y			







5) Interpolate each of the sets of points.

6) Draw $(x(t), y(t))$ as a parametric curve.

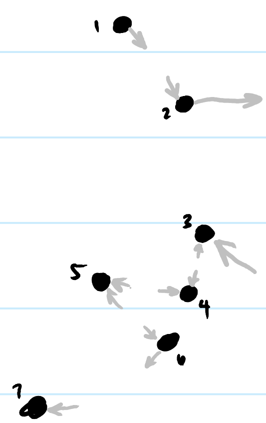
L11: Bézier Curves

Goal: To find out about a common type of interpolatin curve used in industry, Bezier curves.

Bézier Curves

Suppose we have a set of interpolation points, and we also want to specify the direction at which our curve approaches each point.

eg.



Introducing... Bézier Curves.

Bézier curves are based on Bernstein polynomials.

$$B_{i,N}(t) =$$

Properties... See Matlab worksheet.

A Bezier curve for the set of points $\{x_0, x_1, \dots, x_N\}$ is a sum of Bernstein polynomials

$$P(t) =$$

Then:

- $P(0) =$ since and for $i = 1, \dots, N$
- $P(1) =$ since and for $i = 0, \dots, N-1$

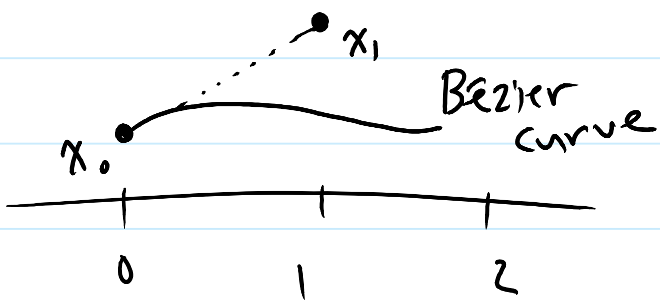
$$\bullet\ P'(t) = \frac{d}{dt}\sum_{i=0}^{N} x_i B_{i,N}(t)$$

$$= \sum_{i=0}^{N} x_i N\left(B_{i-1,N-1}(t) - B_{i,N-1}(t)\right)$$

At $t=0 \Rightarrow P'(0) =$

$=$

$=$

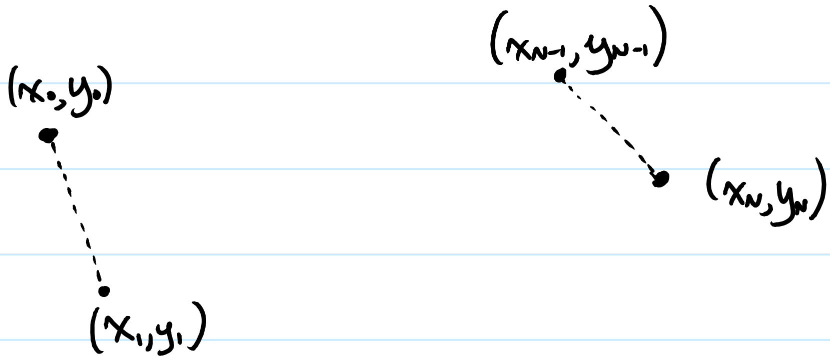


$$\bullet\ P'(t)\Big|_{t=1} =$$

Hence, for a set of points $\{(x_0, y_0), (x_1, y_1), \dots, (x_N, y_N)\}$
The Bezier curve:
- Passes through

- Is tangent to

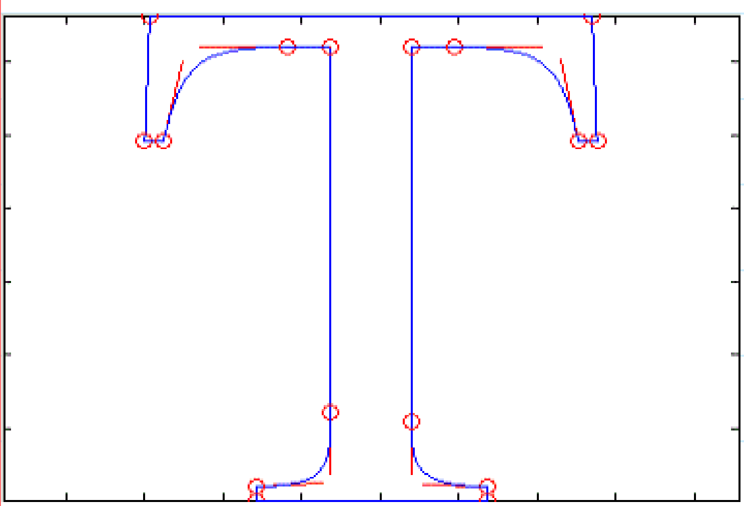
- Is tangent to



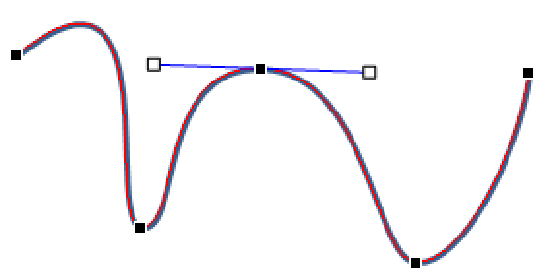
Moreover, the positions of the points "pull" the curve, although the curve does not necessarily pass through the points (except for the first and last).

<u>Application</u>

Bézier curves are used in TrueType fonts.



Microsoft Word, and other drawing applications.



(Matlab demo - Bezier.m)