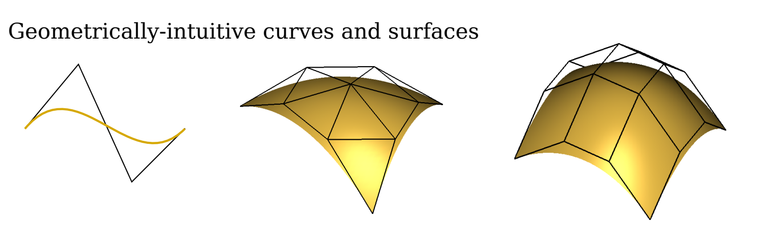
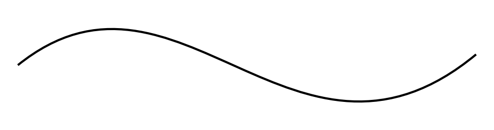
Bézier Curves and Surfaces

Geometrically-intuitive curves and surfaces





A Little History

o **Bernstein** polynomials form the basis of the Bézier representation

o Bézier representation extended to surfaces by **Paul de Casteljau** (physicist, mathematician working for Citroen)

o Popularized by **Pierre Bézier** (automotive engineer at Renault)

Sergei Natanovich Bernstein **Pierre Bézier**

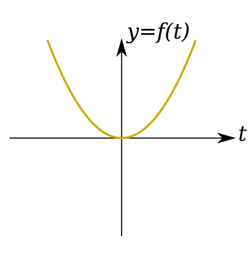
Photos from
http://en.wikipedia.org/wiki/Sergei_Natanovich_Bernstein
http://www.photoshopessentials.com/basics/pen-tool-selections/

Polynomial functions and parametric curves

Power series representation:

$$f(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + ... + a_d t^d$$

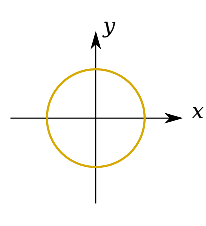
d is the degree of the curve
f(t) is defined by *d+1* coefficients $a_0, a_1, ..., a_d$



Graph of $f(t) = t^2$

Graphs of functions f(t) are limited in the types of curves they can produce. The curves always go left to right. For example, it cannot produce a circle.

We can describe a more general class of curves using the *parametric representation (x(t), y(t))*.



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

Why are polynomials useful?

Very easy to evaluate! We know how to multiply and add (something that computers can do very efficiently).

Evaluate *f(0.14)* in the following cases:

- $f(t) = t^3 + 3t$ ← 3 mult and 1 add
- $f(t) = \sqrt{t}$
- $f(t) = \cos(t)$
- $f(t) = e^t$

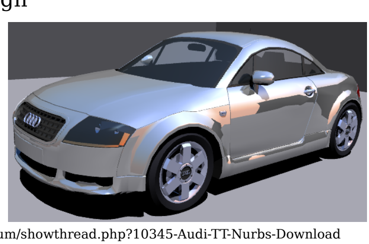
Uses
o Many real-world constraints, e.g. **(squared) distances**, are polynomial - lots of work done in solving system of polynomial constraints

o Many **physical systems** are (rational) polynomials
  o missle trajectory (parabola)
  o gravity, charge potential

o **Approximations** to smooth functions
  o approximate cos and sin as above
  o approximate smooth functions on surfaces

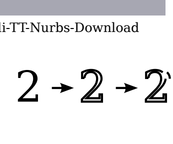
o **Cryptography**

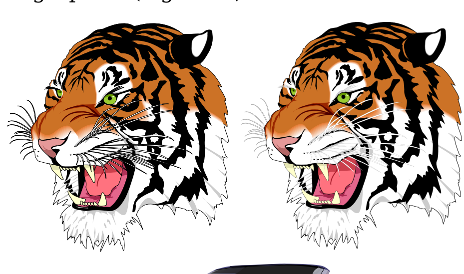
o **Represent smooth curves** (Bézier really good for this)
  o Computer-Aided Design
    o chassis of cars
    o outer hull of planes

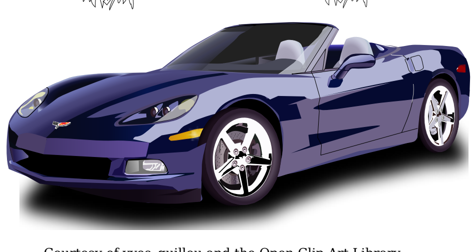


From
http://blenderartists.org/forum/showthread.php?10345-Audi-TT-Nurbs-Download

  o fonts (almost everything on this page!)
  o vector graphics (e.g. SVG)





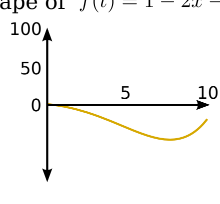


Courtesy of yves_guillou and the Open Clip Art Library
http://www.openclipart.org/detail/sport-car-by-yves_guillou-21763

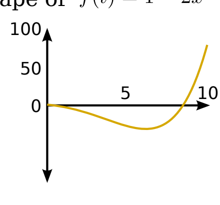
Why not use power representation?

What's wrong with the power-series representation?

Not geometrically intuitive
What is the shape of $f(t) = 1 - 2x - x^2 + 0.001x^5$?



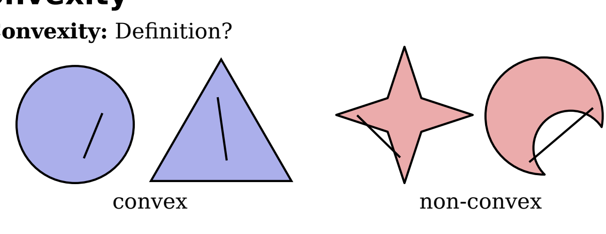
Not numerically stable
What is the shape of $f(t) = 1 - 2x - x^2 + 0.002x^5$?



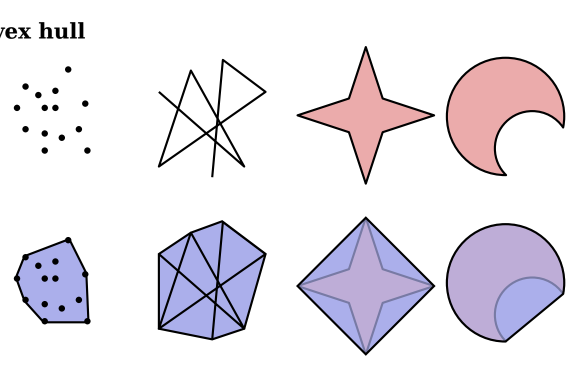
Small change in coefficient
→ huge change in output

Convexity

Convexity: Definition?



Convex hull



Linear combinations

Linear combination w of a_1, a_2, a_3, a_4 with weights b_1, b_2, b_3, b_4

$$w = a_1 b_1 + a_2 b_2 + a_3 b_3 + a_4 b_4$$

(Here w is also a linear combination of b_1, b_2, b_3, b_4)

Example: Polynomials in power form are linear combinations

$$f(t) = a_0 1 + a_1 t + a_2 t^2 + a_3 t^3$$
$$g(t) = a_0 1 + a_1 (t-1) + a_2 (t-1)^2 + a_3 (t-1)^3$$

cubic Bézier → $h(t) = a_0 (1-t)^3 + a_1 3(1-t)^2 t + a_2 3(1-t) t^2 + a_3 t^3$

Example: Trigonometric polynomial

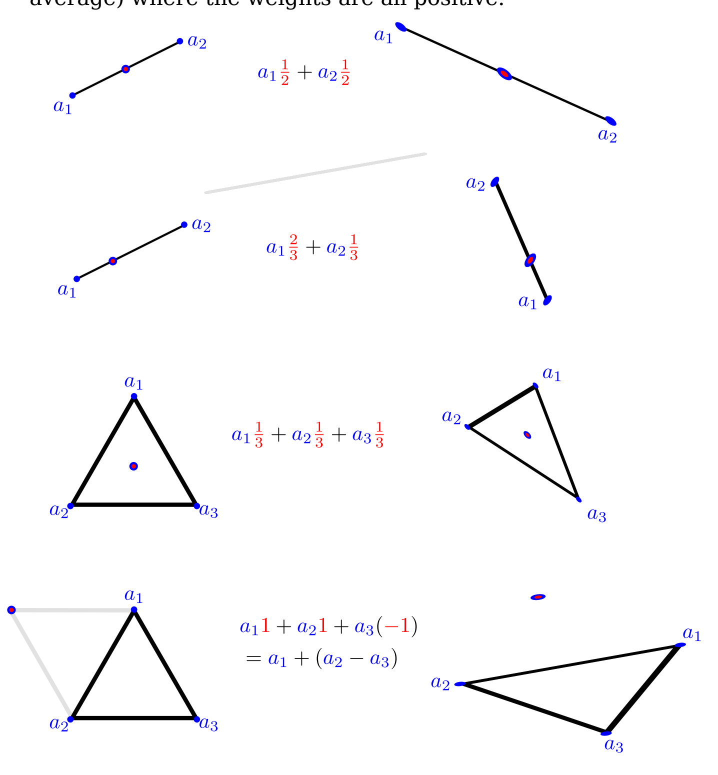
$$f(t) = a_0 \cos(t) + a_1 \sin(t) + a_2 \cos(2t) + a_3 \sin(2t)$$

Affine combination w of a_1, a_2, a_3, a_4 is a linear combination where the weights add to 1:

$$b_1 + b_2 + b_3 + b_4 = 1$$

(Think weighted average.)

Convex combination is an affine combination (weighted average) where the weights are all positive.



Affine combinations are geometrically meaningful, independent of the location of the original points.

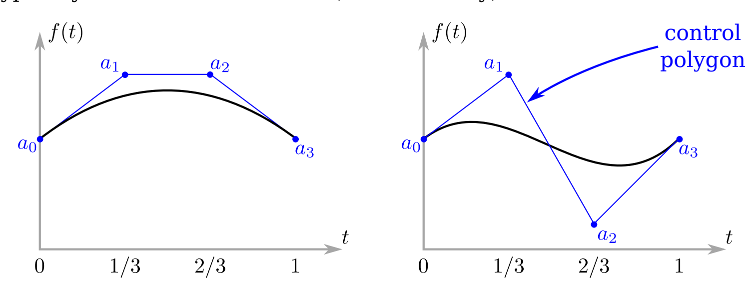
Convex combinations have the additional property that the result in is in the **convex hull**.

Bézier representation

Cubic (degree 3) polynomial in **Bézier form** with coefficients a_1, a_2, a_3, a_4 is the function

$$f(t) = a_0 (1-t)^3 + a_1 3(1-t)^2 t + a_2 3(1-t) t^2 + a_3 t^3$$

Typically restricted to $0 \leq t \leq 1$ (not necessary)



Properties:
o **Interpolates** the end points
o The curve is **tangent** to the control polygon at each end

o **Affine invariant:** The Bézier basis

$(1-t)^3$ $3(1-t)^2 t$ $3(1-t) t^2$ t^3

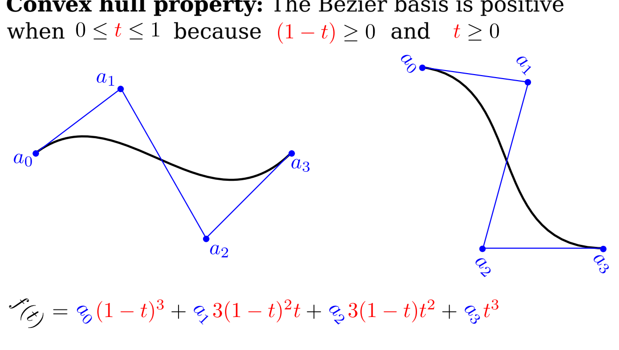
adds up to 1 for *any t*. How would you prove this?

$$(x+y)^3 = x^3 + 3x^2 y + 3xy^2 + y^3$$
$$1 = 1^3 = ((1-t) + t)^3 = (1-t)^3 + 3(1-t)^2 t + 3(1-t) t^2 + t^3$$

So, *f(t)* is an affine combination of a_1, a_2, a_3, a_4

$$f(t) = a_0 (1-t)^3 + a_1 3(1-t)^2 t + a_2 3(1-t) t^2 + a_3 t^3$$

o **Convex hull property:** The Bézier basis is positive when $0 \leq t \leq 1$ because $(1-t) \geq 0$ and $t \geq 0$

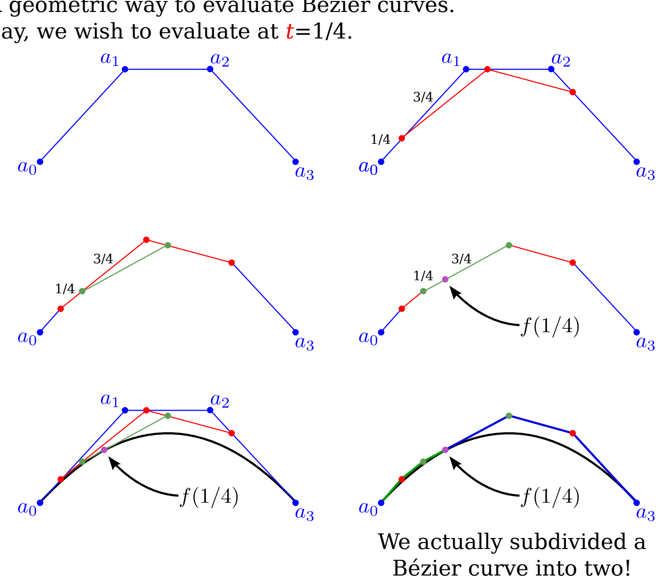


$$f(t) = a_0 (1-t)^3 + a_1 3(1-t)^2 t + a_2 3(1-t) t^2 + a_3 t^3$$

o **Numerically-stable** to compute! Output (function) cannot change more than input (control polygon).

De Casteljau's Algorithm

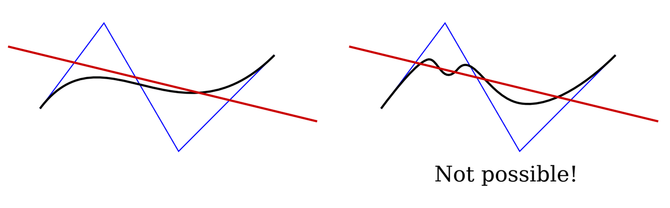
A geometric way to evaluate Bézier curves.
Say, we wish to evaluate at *t*=1/4.



We actually subdivided a Bézier curve into two!

Variation Diminishing Property

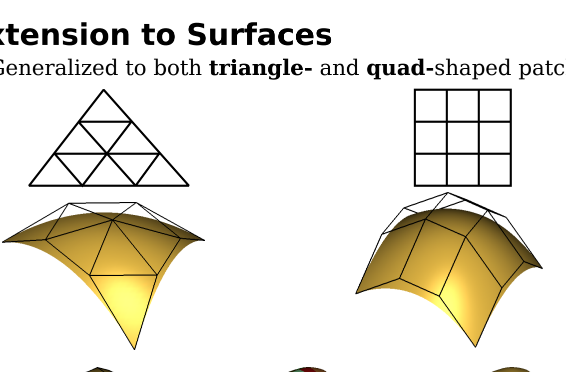
o Follows as a consequence of iterative corner-cutting.
o "**Shape-preserving**"
  o Curve will not wiggle more than control polygon.
  o No line will intersect the curve more times than it intersects the control polygon.

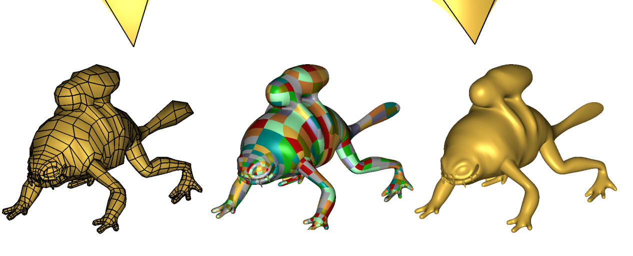


Not possible!

Extension to Surfaces

Generalized to both **triangle-** and **quad-**shaped patches.





Summary

Bézier representation

o Fun and **intuitive** way to manipulate polynomials

o Important and useful **geometric properties**
  o curve defined by a **control polygon** (exaggeration of shape)
  o **shape preserving**
  o **numerically-stable** to compute between 0 and 1
    (consequence of the convex hull property)

o Used everywhere, from fonts, vector graphics, CAD

o Can be further **generalized** to represent and manipulate piecewise-polynomials efficiently
  o B-spline
  o NURBS