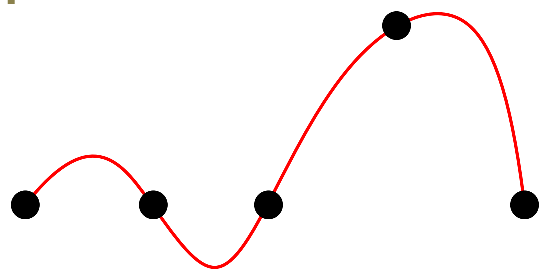
HERMITE INTERPOLATION

Rodrigo Silveira

Curve and Surface Design
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

Recall: Issues with polynomial interpolation



- The high degree of the polynomial produces a curve with higher roughness (i.e., it can wiggle a lot) than probably desired

  $\rightarrow$ The variation diminishing property is not satisfied!

- Intuitively: adding data points should improve the resulting polynomial curve. But that is not always the case! This is known as *Runge's phenomenon*

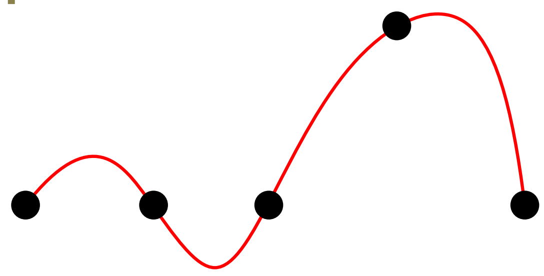
- Lagrange's formula requires $\Theta(n^2)$ additions and products, which is quite a lot (although more efficient versions exist)

- If one has computed $\gamma(t)$ for n points and needs to add one extra point, everything needs to be recomputed

- Lagrange's formula is not numerically stable: small variations in the input points can produce large variations in the final curve

- The method is not easy to make interactive: if the curve is not what one wants, (and you cannot modify the data points) all you can do is to add more points

Recall: Issues with polynomial interpolation



- The high degree of the polynomial produces a curve with higher roughness (i.e., it can wiggle a lot) than probably desired

  → The variation diminishing property is not satisfied!

- Intuitively: adding data points should improve the resulting polynomial curve. But that is not always the case! This is known as *Runge's phenomenon*

- Lagrange's formula requires $\Theta(n^2)$ additions and products, which is quite a lot (although more efficient versions exist)

- If one has computed $\gamma(t)$ for n points and needs to add one extra point, everything needs to be recomputed

- Lagrange's formula is not numerically stable: small variations in the input points can produce large variations in the final curve

- The method is not easy to make interactive: if the curve is not what one wants, (and you cannot modify the data points) all you can do is to add more points

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

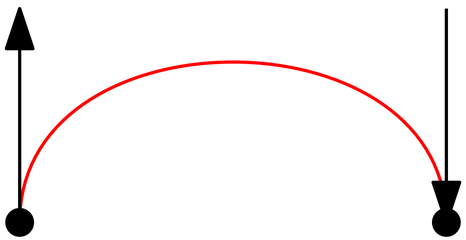
Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points

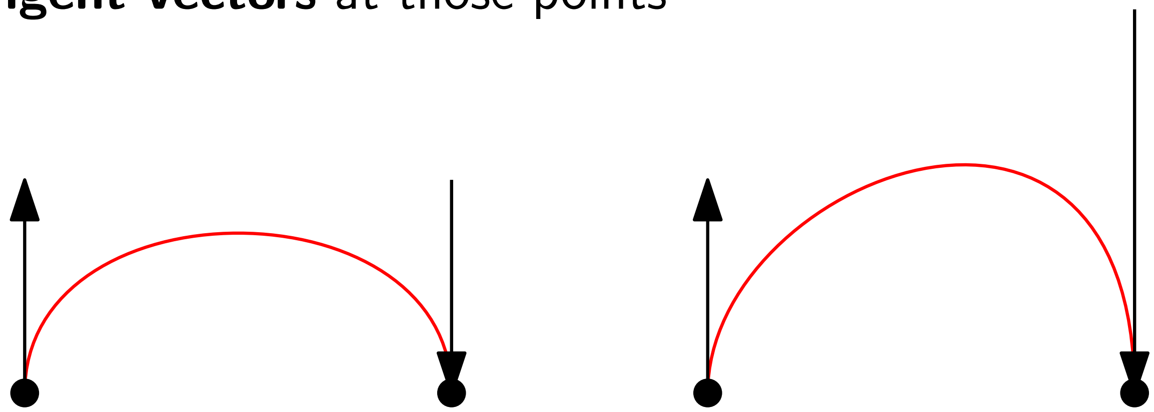


A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points



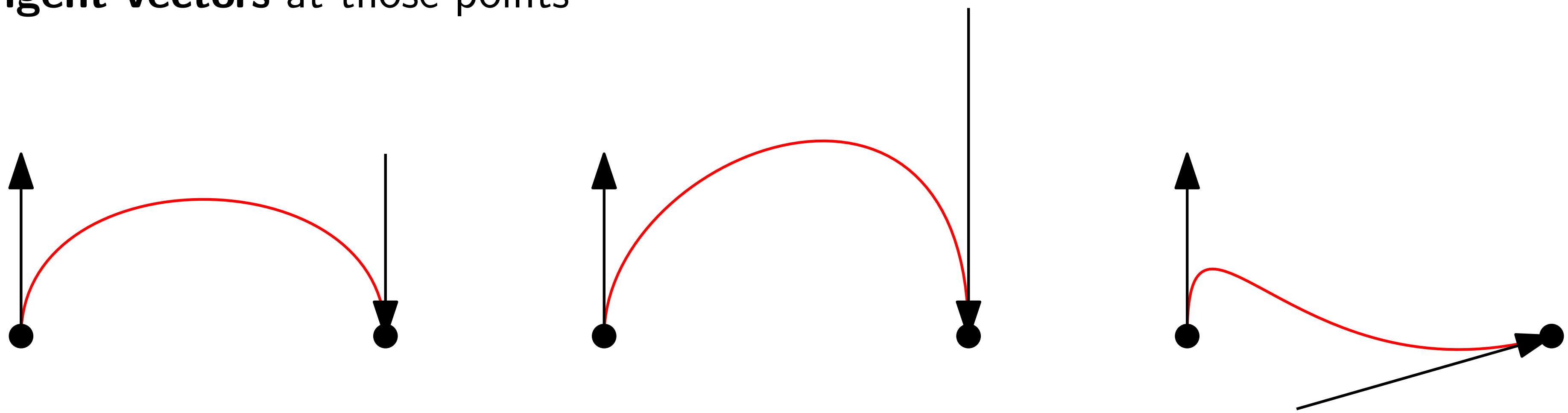
Note the effect of modifying one of the tangent vectors

A more interactive interpolation method

Practical curve design methods need to be interactive:

- Based on user-controlled parameters that modify the shape of the curve in an intuitive (and thus predictable) way
- The first of such methods that we will see is **Hermite interpolation**

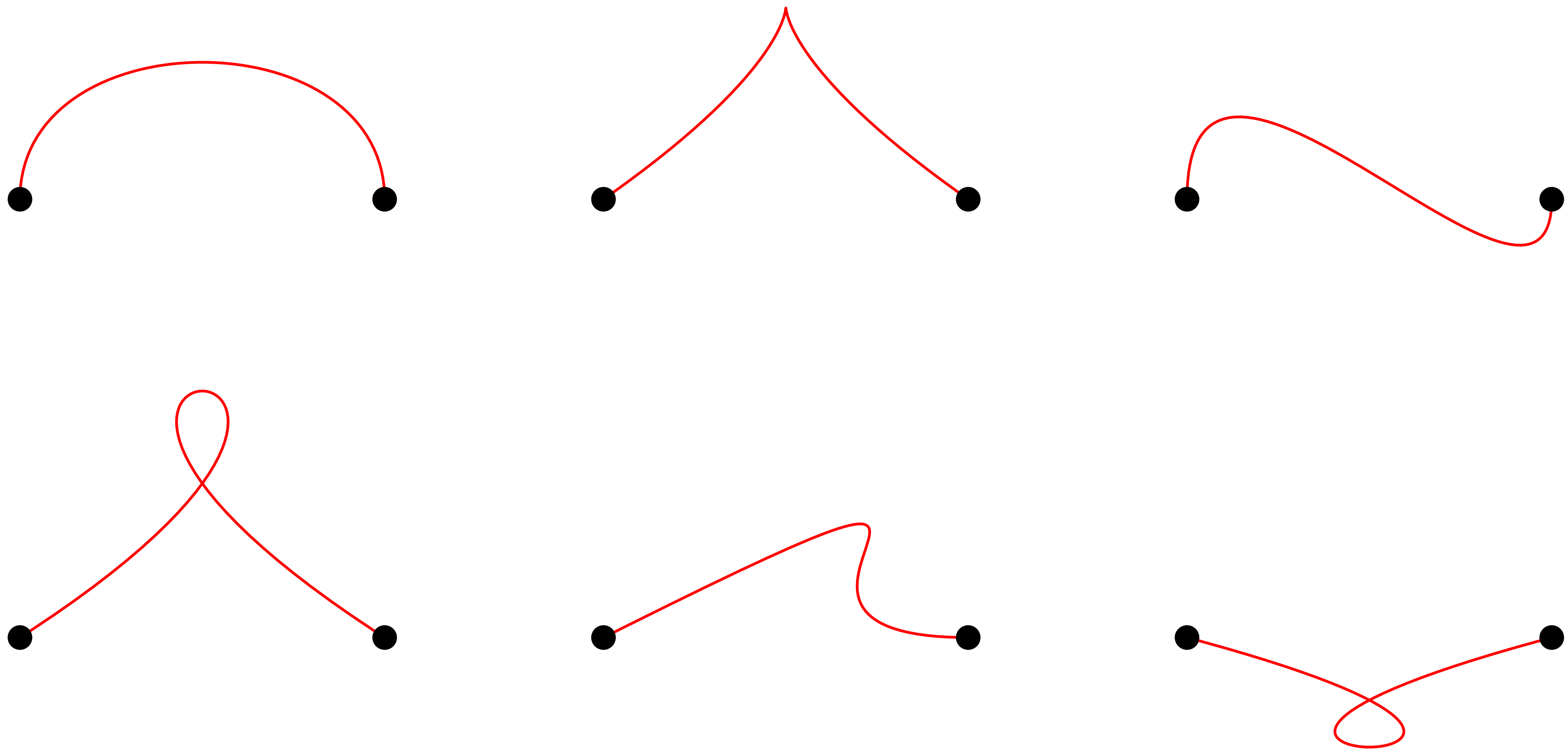
Idea: design a curve that interpolates **two** points, and whose shape is controlled by the **tangent vectors** at those points



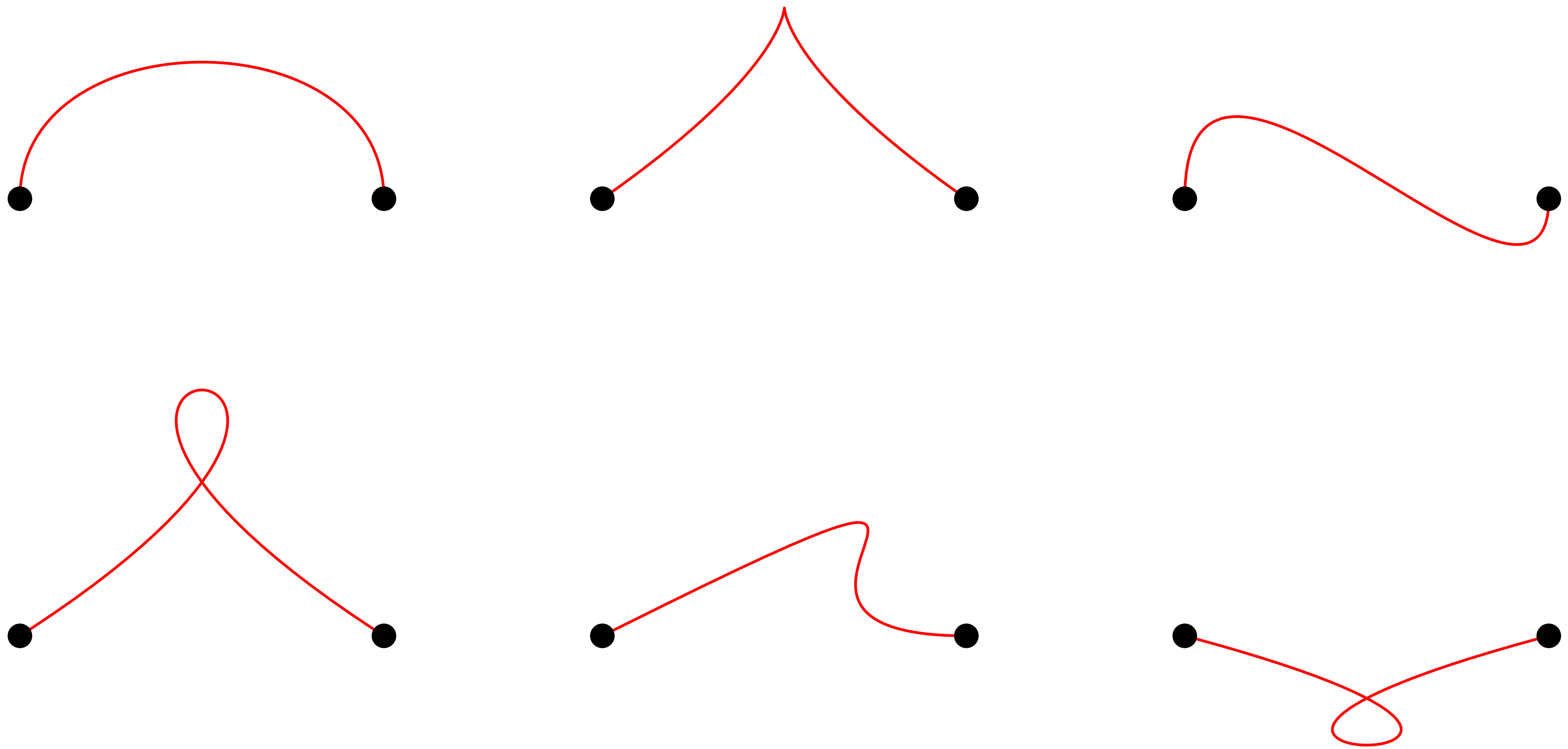
Note the effect of modifying one of the tangent vectors

HERMITE INTERPOLATION

A single Hermite curve can take many different shapes

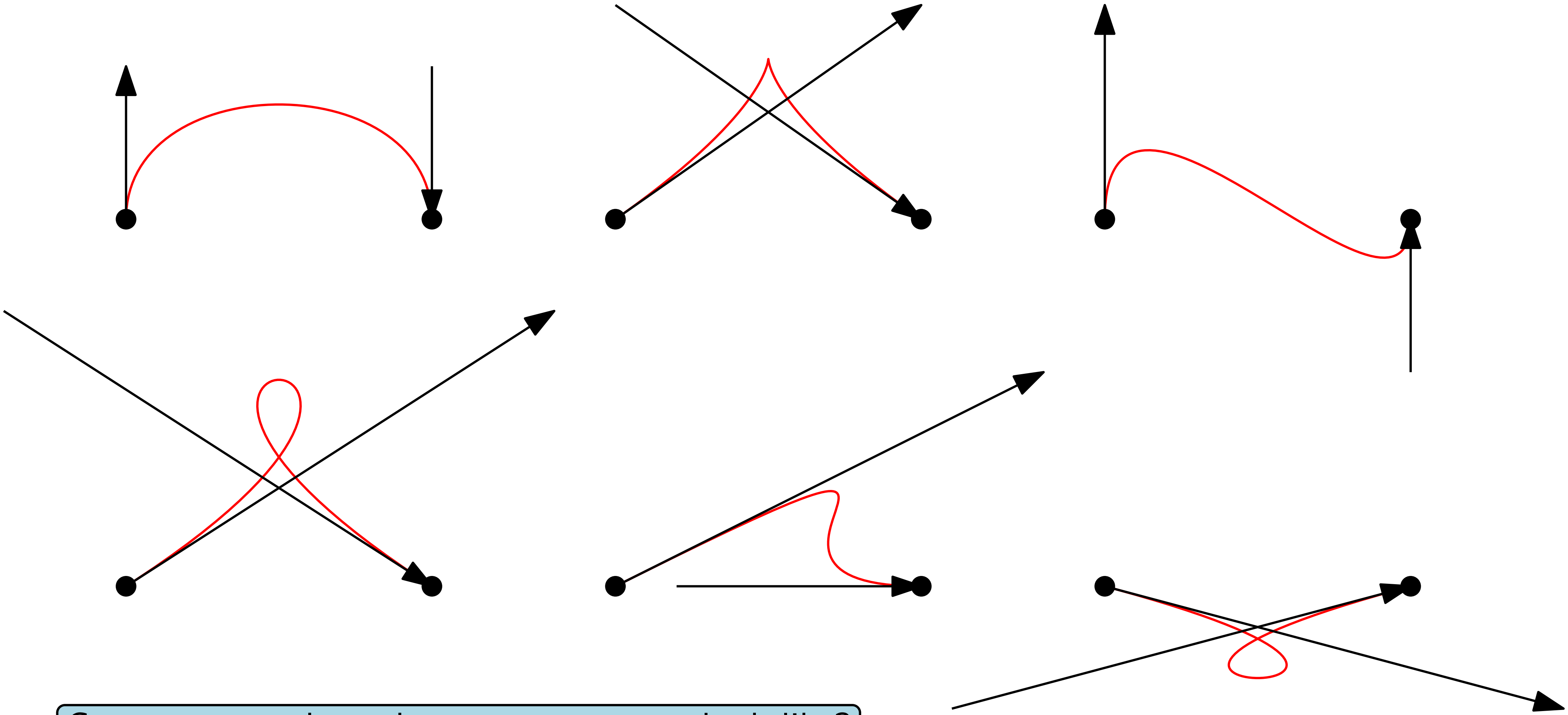


A single Hermite curve can take many different shapes



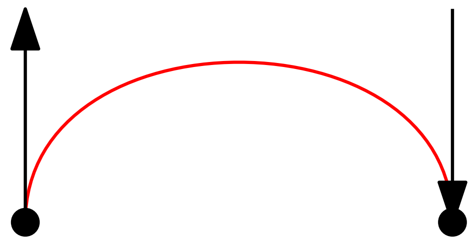
Can you guess how the tangent vectors look like?

A single Hermite curve can take many different shapes

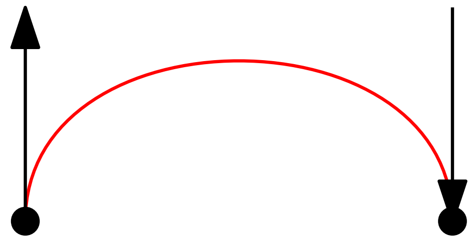


Can you guess how the tangent vectors look like?

Cubic Hermite interpolation

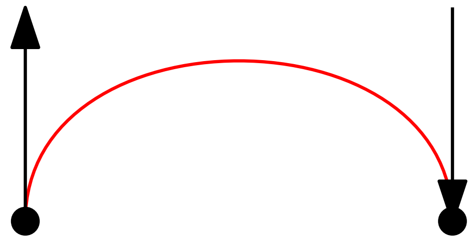


Cubic Hermite interpolation



Each curve is a (parametric) cubic polynomial

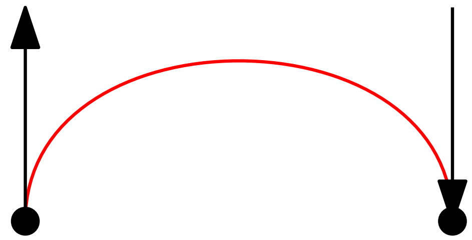
Cubic Hermite interpolation



Each curve is a (parametric) cubic polynomial

- We know that we can define a cubic polynomial based on four points
- But we can also define it based on two points and two tangent vectors!

Cubic Hermite interpolation



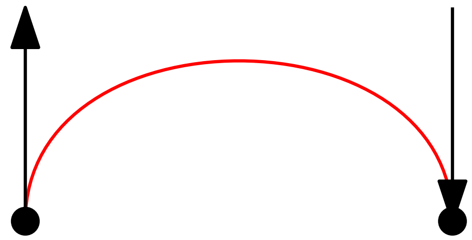
Each curve is a (parametric) cubic polynomial

- We know that we can define a cubic polynomial based on four points
- But we can also define it based on two points and two tangent vectors!

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Cubic Hermite interpolation



Each curve is a (parametric) cubic polynomial

- We know that we can define a cubic polynomial based on four points
- But we can also define it based on two points and two tangent vectors!

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Secondly, we prove that it exists, by deducing an expression for it.

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Let γ and δ be two curves that satisfy the constraints above, and consider a third curve r defined as $r(t) = \gamma(t) - \delta(t)$. Clearly, $r(t)$ is a polynomial of degree at most three. Since $r(0) = 0$ and $r(1) = 0$, we can write it as $r(t) = at(t-1)(t-t_0)$, for two unknown values a and t_0.

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Let γ and δ be two curves that satisfy the constraints above, and consider a third curve r defined as $r(t) = \gamma(t) - \delta(t)$. Clearly, $r(t)$ is a polynomial of degree at most three. Since $r(0) = 0$ and $r(1) = 0$, we can write it as $r(t) = at(t-1)(t-t_0)$, for two unknown values a and t_0.

Now consider $r'(t)$. We know that $r'(t) = \gamma'(t) - \delta'(t)$, so we have that $r'(0) = 0$ and $r'(1) = 0$. We can also write $r'(t)$ as follows

$$r'(t) = at(t-1) + at(t-t_0) + a(t-1)(t-t_0)$$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Let γ and δ be two curves that satisfy the constraints above, and consider a third curve r defined as $r(t) = \gamma(t) - \delta(t)$. Clearly, $r(t)$ is a polynomial of degree at most three. Since $r(0) = 0$ and $r(1) = 0$, we can write it as $r(t) = at(t-1)(t-t_0)$, for two unknown values a and t_0.

Now consider $r'(t)$. We know that $r'(t) = \gamma'(t) - \delta'(t)$, so we have that $r'(0) = 0$ and $r'(1) = 0$. We can also write $r'(t)$ as follows

$$r'(t) = at(t-1) + at(t-t_0) + a(t-1)(t-t_0)$$

Therefore, since $r'(0) = 0$, we have $0 = a(-1)(-t_0) = at_0$

Similarly, since $r'(1) = 0$, we have $0 = a(1)(1-t_0) = a(1-t_0)$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\vec{v_0}, \vec{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \vec{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \vec{v_1}$$

Proof. First we prove uniqueness, as we did with Lagrange interpolation.

Let γ and δ be two curves that satisfy the constraints above, and consider a third curve r defined as $r(t) = \gamma(t) - \delta(t)$. Clearly, $r(t)$ is a polynomial of degree at most three. Since $r(0) = 0$ and $r(1) = 0$, we can write it as $r(t) = at(t-1)(t-t_0)$, for two unknown values a and t_0.

Now consider $r'(t)$. We know that $r'(t) = \gamma'(t) - \delta'(t)$, so we have that $r'(0) = 0$ and $r'(1) = 0$. We can also write $r'(t)$ as follows

$$r'(t) = at(t-1) + at(t-t_0) + a(t-1)(t-t_0)$$

$$\left.\begin{array}{l}\text{Therefore, since } r'(0) = 0\text{, we have } 0 = a(-1)(-t_0) = at_0 \\ \text{Similarly, since } r'(1) = 0\text{, we have } 0 = a(1)(1-t_0) = a(1-t_0)\end{array}\right] \longrightarrow$$

iff a is 0, thus $r(t) = 0$ for all t, and therefore $\gamma(t) = \delta(t)$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. Now we will show that it exists. Recall that $\gamma(t)$ is a cubic polynomial in t.

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. Now we will show that it exists. Recall that $\gamma(t)$ is a cubic polynomial in t.

Hence $\gamma(t)$ can be written as $\gamma(t) = At^3 + Bt^2 + Ct + D$, for $A, B, C, D \in \mathbb{R}$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. Now we will show that it exists. Recall that $\gamma(t)$ is a cubic polynomial in t.

Hence $\gamma(t)$ can be written as $\gamma(t) = At^3 + Bt^2 + Ct + D$, for $A, B, C, D \in \mathbb{R}$

Then $\gamma'(t) = 3At^2 + 2Bt + C$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. Now we will show that it exists. Recall that $\gamma(t)$ is a cubic polynomial in t.

Hence $\gamma(t)$ can be written as $\gamma(t) = At^3 + Bt^2 + Ct + D$, for $A, B, C, D \in \mathbb{R}$

Then $\gamma'(t) = 3At^2 + 2Bt + C$

We have four constraints that give us four equations on A, B, C, D

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof. Now we will show that it exists. Recall that $\gamma(t)$ is a cubic polynomial in t.

Hence $\gamma(t)$ can be written as $\gamma(t) = At^3 + Bt^2 + Ct + D$, for $A, B, C, D \in \mathbb{R}$

Then $\gamma'(t) = 3At^2 + 2Bt + C$

We have four constraints that give us four equations on A, B, C, D

- $P_0 = \gamma(0) = D$, thus $D = P_0$
- $\overrightarrow{v_0} = \gamma'(0) = C$, thus $C = \overrightarrow{v_0}$
- $P_1 = \gamma(1) = A + B + C + D$, thus $B = P_1 - P_0 - \overrightarrow{v_0} - A$
- $\overrightarrow{v_1} = \gamma'(1) = 3A + 2B + C = 3A + 2(P_1 - P_0 - \overrightarrow{v_0} - A) + \overrightarrow{v_0} = A + 2P_1 - 2P_0 - \overrightarrow{v_0}$

$$\to A = \overrightarrow{v_1} + \overrightarrow{v_0} + 2P_0 - 2P_1$$
$$\to B = P_0 - P_1 - \overrightarrow{v_0} - \overrightarrow{v_1} - \overrightarrow{v_0} - 2P_0 + 2P_1 = 3P_1 - 3P_0 - 2\overrightarrow{v_0} - \overrightarrow{v_1}$$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof (cont'd). Replacing the values of A, B, C, D we obtain:

$$\begin{aligned}\gamma(t) &= At^3 + Bt^2 + Ct + D \\ &= (\overrightarrow{v_1} + \overrightarrow{v_0} + 2P_0 - 2P_1)t^3 + (3P_1 - 3P_0 - 2\overrightarrow{v_0} - \overrightarrow{v_1})t^2 + \overrightarrow{v_0}t + P_0\end{aligned}$$

After simplifying and grouping by the input points and vectors, this is:

$$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}, \text{ for } t \in [0, 1]$$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof (cont'd). Replacing the values of A, B, C, D we obtain:

$$\begin{aligned}\gamma(t) &= At^3 + Bt^2 + Ct + D \\ &= (\overrightarrow{v_1} + \overrightarrow{v_0} + 2P_0 - 2P_1)t^3 + (3P_1 - 3P_0 - 2\overrightarrow{v_0} - \overrightarrow{v_1})t^2 + \overrightarrow{v_0}t + P_0\end{aligned}$$

After simplifying and grouping by the input points and vectors, this is:

$$\boxed{\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}, \text{ for } t \in [0,1]}$$

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof (cont'd). Replacing the values of A, B, C, D we obtain:

$$\begin{aligned}\gamma(t) &= At^3 + Bt^2 + Ct + D \\ &= (\overrightarrow{v_1} + \overrightarrow{v_0} + 2P_0 - 2P_1)t^3 + (3P_1 - 3P_0 - 2\overrightarrow{v_0} - \overrightarrow{v_1})t^2 + \overrightarrow{v_0}t + P_0\end{aligned}$$

After simplifying and grouping by the input points and vectors, this is:

$$\boxed{\gamma(t) = \underline{(2t^3 - 3t^2 + 1)}P_0 + \underline{(-2t^3 + 3t^2)}P_1 + \underline{(t^3 - 2t^2 + t)}\overrightarrow{v_0} + \underline{(t^3 - t^2)}\overrightarrow{v_1}, \text{ for } t \in [0, 1]}$$

Hermite *blending functions*

Cubic Hermite interpolation

Theorem: Given two points P_0, P_1 and two vectors $\overrightarrow{v_0}, \overrightarrow{v_1}$, there exists a unique curve $\gamma(t)$ parametrized as a cubic polynomial in t such that:

$$\gamma(0) = P_0,\ \gamma'(0) = \overrightarrow{v_0}$$
$$\gamma(1) = P_1,\ \gamma'(1) = \overrightarrow{v_1}$$

Proof (cont'd). Replacing the values of A, B, C, D we obtain:

$$\begin{aligned}\gamma(t) &= At^3 + Bt^2 + Ct + D \\ &= (\overrightarrow{v_1} + \overrightarrow{v_0} + 2P_0 - 2P_1)t^3 + (3P_1 - 3P_0 - 2\overrightarrow{v_0} - \overrightarrow{v_1})t^2 + \overrightarrow{v_0}t + P_0\end{aligned}$$

After simplifying and grouping by the input points and vectors, this is:

$$\boxed{\gamma(t) = \underline{(2t^3 - 3t^2 + 1)}P_0 + \underline{(-2t^3 + 3t^2)}P_1 + \underline{(t^3 - 2t^2 + t)}\overrightarrow{v_0} + \underline{(t^3 - t^2)}\overrightarrow{v_1}, \text{ for } t \in [0, 1]}$$

Hermite *blending functions*

Exercise: verify that $\gamma(t)$ satisfies the four constraints of the theorem

Cubic Hermite blending functions

The concept of blending functions is fundamental for many curve design methods

$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}$, for $t \in [0, 1]$

Cubic Hermite blending functions

The concept of blending functions is fundamental for many curve design methods

$$\gamma(t) = (\underbrace{2t^3 - 3t^2 + 1}_{F_1})P_0 + (\underbrace{-2t^3 + 3t^2}_{F_2})P_1 + (\underbrace{t^3 - 2t^2 + t}_{F_3})\overrightarrow{v_0} + (\underbrace{t^3 - t^2}_{F_4})\overrightarrow{v_1}, \text{ for } t \in [0,1]$$

These are the four blending functions in Hermite interpolation:

$F_1(t) = 2t^3 - 3t^2 + 1$ $\quad$ $F_3(t) = t^3 - 2t^2 + t$

$F_2(t) = -2t^3 + 3t^2$ $\quad$ $F_4(t) = t^3 - t^2$

Cubic Hermite blending functions

The concept of blending functions is fundamental for many curve design methods

$$\gamma(t) = (\underbrace{2t^3 - 3t^2 + 1}_{F_1})P_0 + (\underbrace{-2t^3 + 3t^2}_{F_2})P_1 + (\underbrace{t^3 - 2t^2 + t}_{F_3})\overrightarrow{v_0} + (\underbrace{t^3 - t^2}_{F_4})\overrightarrow{v_1}, \text{ for } t \in [0,1]$$

These are the four blending functions in Hermite interpolation:

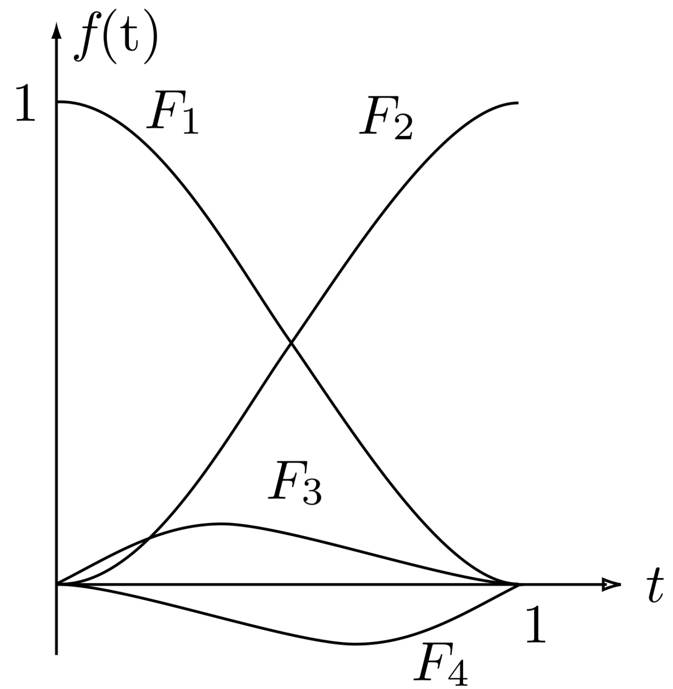
$F_1(t) = 2t^3 - 3t^2 + 1$

$F_2(t) = -2t^3 + 3t^2$

$F_3(t) = t^3 - 2t^2 + t$

$F_4(t) = t^3 - t^2$

Let's see how these functions look like:



Cubic Hermite blending functions

The concept of blending functions is fundamental for many curve design methods

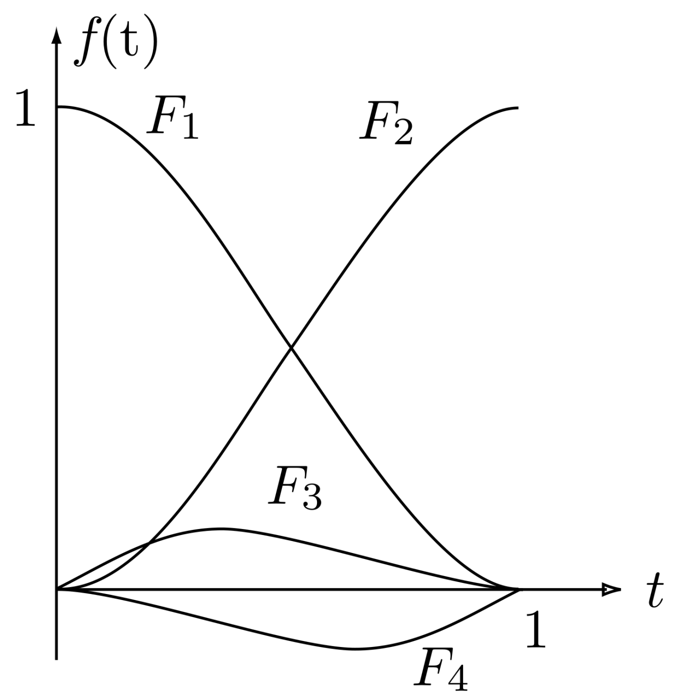
$$\gamma(t) = (\underbrace{2t^3 - 3t^2 + 1}_{F_1})P_0 + (\underbrace{-2t^3 + 3t^2}_{F_2})P_1 + (\underbrace{t^3 - 2t^2 + t}_{F_3})\vec{v_0} + (\underbrace{t^3 - t^2}_{F_4})\vec{v_1}, \text{ for } t \in [0,1]$$

These are the four blending functions in Hermite interpolation:

$F_1(t) = 2t^3 - 3t^2 + 1$ $\qquad$ $F_3(t) = t^3 - 2t^2 + t$

$F_2(t) = -2t^3 + 3t^2$ $\qquad$ $F_4(t) = t^3 - t^2$

Let's see how these functions look like:



The functions control the weights of $P_0, P_1, \vec{v_0}, \vec{v_1}$:

- For $t = 0$, $F_1(t) = 1$, and all others are 0: this makes the curve start at P_0. Similarly, at $t = 1$, $F_2(t) = 1$, and all others are 0, so the curve ends at P_1.
- $F_3(t)$ has a less clear behavior: for small values of t, it has little effect (the curve stays close to P_0). For t around $1/3$, $F_3(t)$ has its maximum influence, pulling the curve in direction $\vec{v_0}$. For larger t, $F_3(t)$ again has almost no effect.
- $F_4(t)$ behaves in a symmetric way to $F_3(t)$.

Affine invariance

As we would expect, Hermite interpolation is affine invariant

Affine invariance

As we would expect, Hermite interpolation is affine invariant

$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}$, for $t \in [0, 1]$

- The weights of the points add up to 1:

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

- The weights of the tangent vectors vanish at $t = 0$ and $t = 1$
- This implies the curve is affine invariant, as one can verify:

Affine invariance

As we would expect, Hermite interpolation is affine invariant

$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}$, for $t \in [0, 1]$

- The weights of the points add up to 1:

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

- The weights of the tangent vectors vanish at $t = 0$ and $t = 1$
- This implies the curve is affine invariant, as one can verify:

Consider $f(x) = Ax + W$. Recall only linear part of f applies to a vector, i.e., $f(\overrightarrow{u}) = A\overrightarrow{u}$

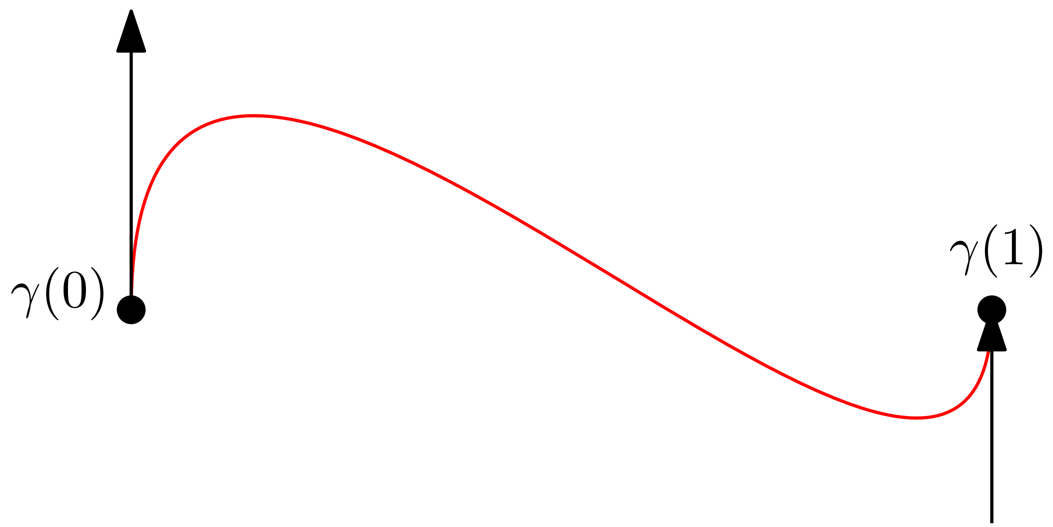
$\gamma(t) = a(t)P_0 + (1 - a(t))P_1 + \alpha(t)\overrightarrow{v_0} + \beta(t)\overrightarrow{v_1}$, for some functions $a(t), \alpha(t), \beta(t)$

$$\begin{aligned}
f(\gamma(t)) &= A(a(t)P_0 + (1 - a(t))P_1 + \alpha(t)\overrightarrow{v_0} + \beta(t)\overrightarrow{v_1}) + W \\
&= a(t)AP_0 + (1 - a(t))AP_1 + \alpha(t)A\overrightarrow{v_0} + \beta(t)A\overrightarrow{v_1} + W \quad \text{(using } W = (a(t) + (1 - a(t))W\text{)} \\
&= a(t)AP_0 + (1 - a(t))AP_1 + \alpha(t)A\overrightarrow{v_0} + (a(t) + (1 - a(t))W + \beta(t)A\overrightarrow{v_1} \\
&= a(t)(AP_0 + W) + (1 - a(t))(AP_1 + W) + \alpha(t)A\overrightarrow{v_0} + \beta(t)A\overrightarrow{v_1} \\
&= a(t)f(P_0) + (1 - a(t))f(P_1) + \alpha(t)f(\overrightarrow{v_0}) + \beta(t)f(\overrightarrow{v_1}) \qquad \therefore \text{ it is affine invariant!}
\end{aligned}$$

Clipping Hermite curves

Clipping is a basic operation with curves: extract a continuous part of an Hermite curve $\gamma(t)$ into a new Hermite curve

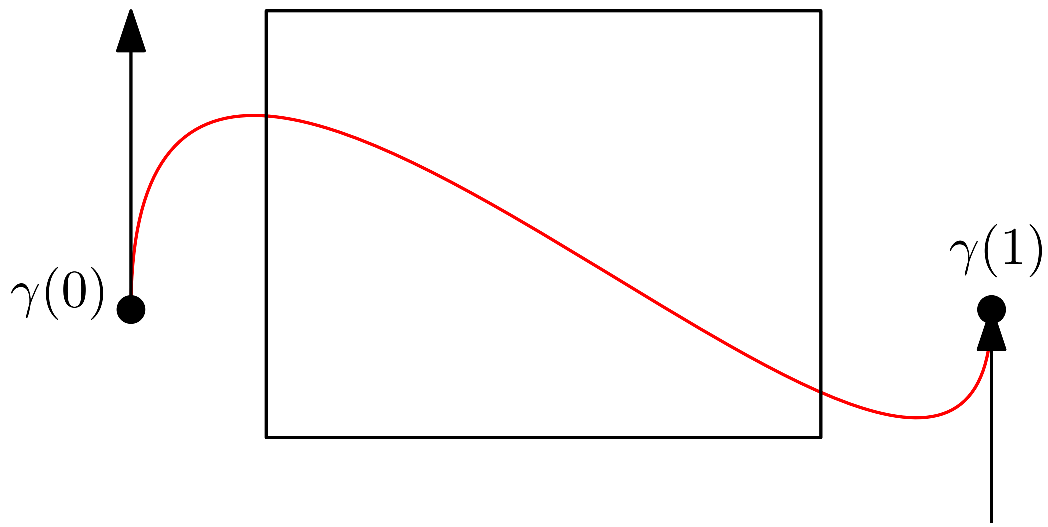
- $\gamma(t)$ is parametrized in $[0, 1]$



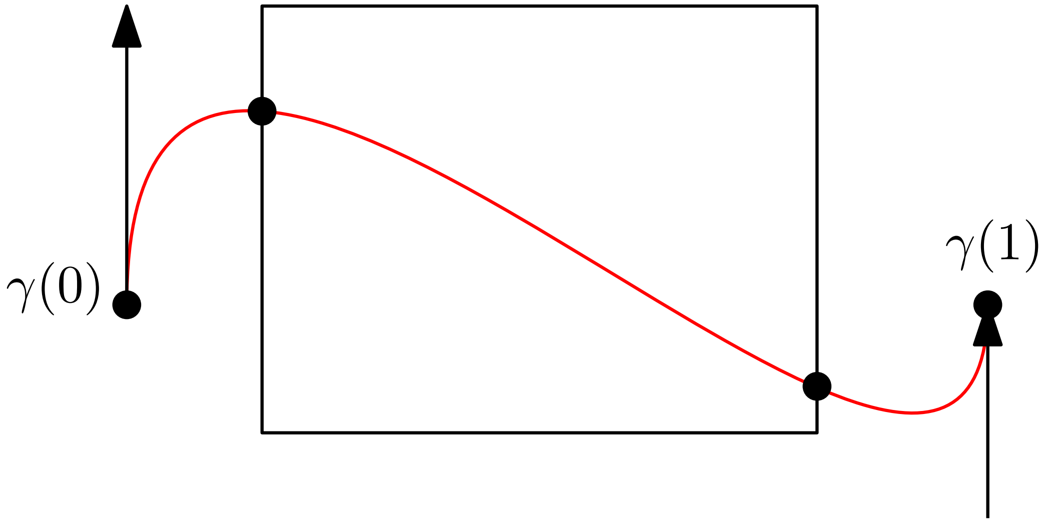
Clipping Hermite curves

Clipping is a basic operation with curves: extract a continuous part of an Hermite curve $\gamma(t)$ into a new Hermite curve

- $\gamma(t)$ is parametrized in $[0,1]$



Clipping Hermite curves



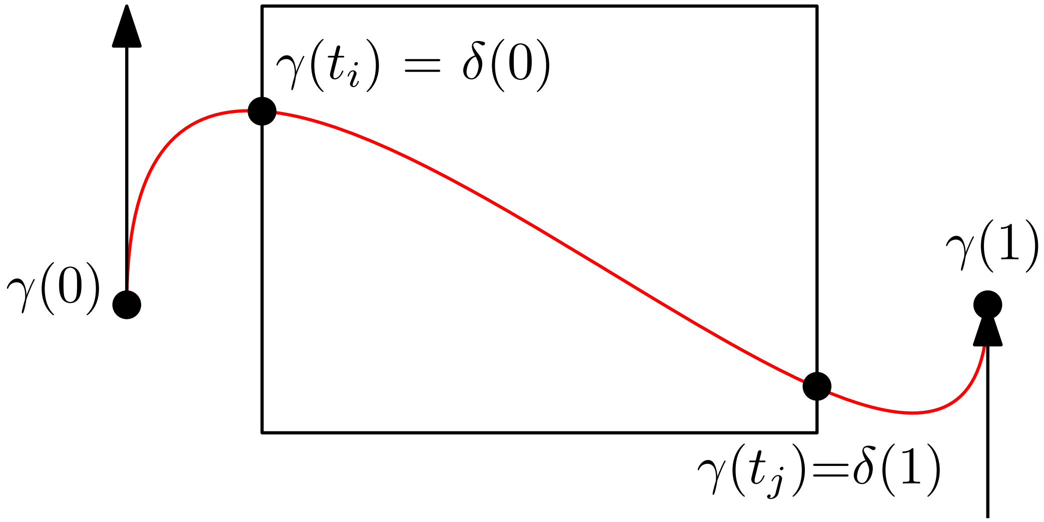
Clipping is a basic operation with curves: extract a continuous part of an Hermite curve $\gamma(t)$ into a new Hermite curve

- $\gamma(t)$ is parametrized in $[0, 1]$
- we want a new curve δ that is equal to γ from t_i to t_j (for some t_i, t_j of our choice)
- $\delta(s)$ should be a Hermite curve (parametrized by $s \in [0, 1]$, with its two tangent vectors)

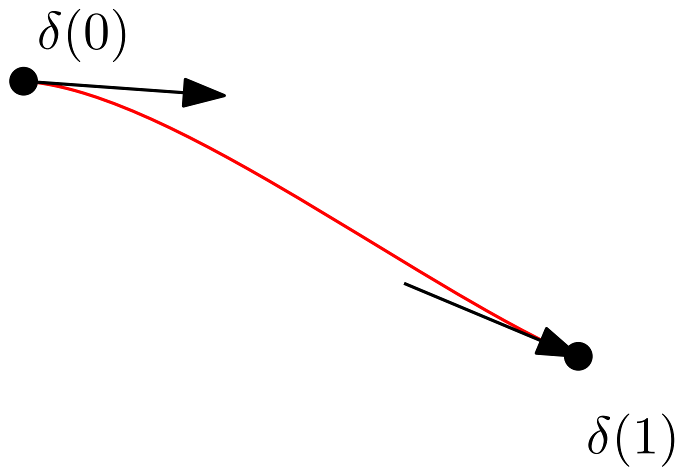
Clipping Hermite curves

Clipping is a basic operation with curves: extract a continuous part of an Hermite curve $\gamma(t)$ into a new Hermite curve

- $\gamma(t)$ is parametrized in $[0, 1]$
- we want a new curve δ that is equal to γ from t_i to t_j (for some t_i, t_j of our choice)

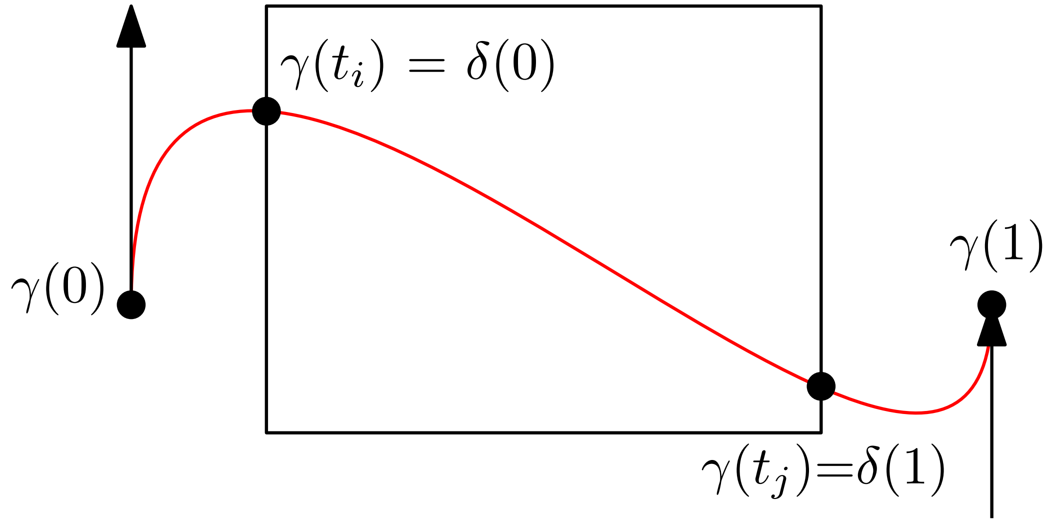


- $\delta(s)$ should be a Hermite curve (parametrized by $s \in [0, 1]$, with its two tangent vectors)



Clipping Hermite curves

Clipping is a basic operation with curves: extract a continuous part of an Hermite curve $\gamma(t)$ into a new Hermite curve



- $\gamma(t)$ is parametrized in $[0, 1]$
- we want a new curve δ that is equal to γ from t_i to t_j (for some t_i, t_j of our choice)
- $\delta(s)$ should be a Hermite curve (parametrized by $s \in [0, 1]$, with its two tangent vectors)

We need to find the two points and two vectors that define δ

We need to reparametrize that portion of γ:

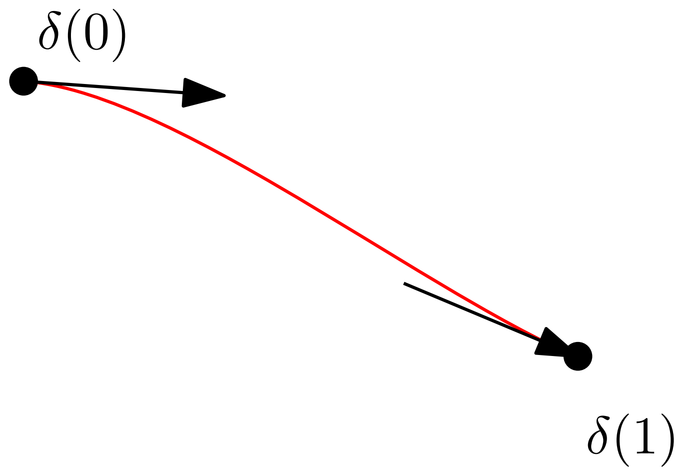
$$[0, 1] \longleftrightarrow [t_i, t_j]$$

$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$



Clipping Hermite curves

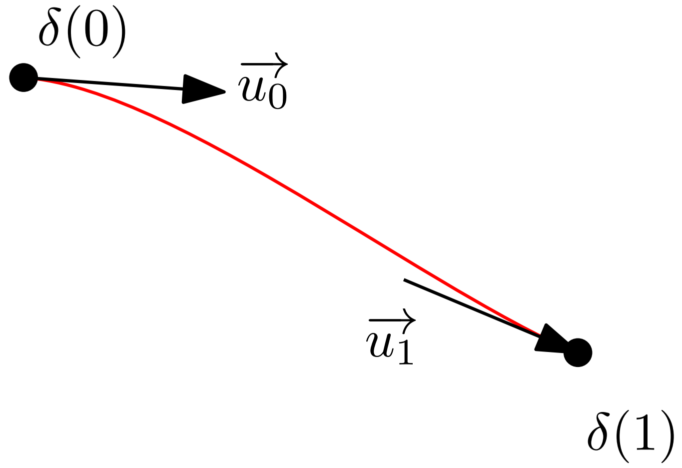
We need to reparametrize that portion of γ:

$$[0,1] \longleftrightarrow [t_i, t_j]$$
$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$



Clipping Hermite curves

We need to reparametrize that portion of γ:

$$[0,1] \longleftrightarrow [t_i, t_j]$$

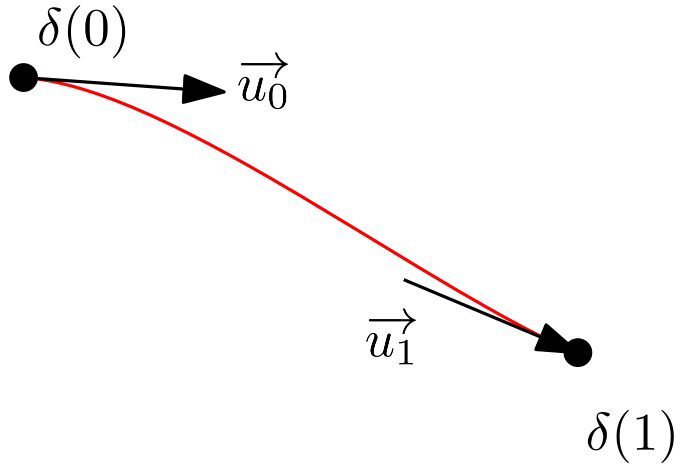
$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$

$$\delta(s) \longleftrightarrow \gamma(t(s))$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$



Clipping Hermite curves

We need to reparametrize that portion of γ:

$[0,1] \longleftrightarrow [t_i, t_j]$

$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$

$\delta(s) \longleftrightarrow \gamma(t(s))$

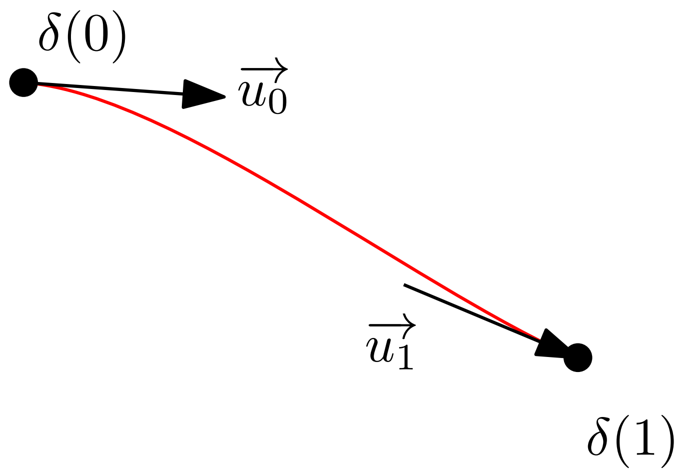
and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$

The values of $\delta(0)$ and $\delta(1)$ we know: for instance, $\delta(0) = \gamma(t(0)) = \gamma(t_i)$
We just need to find the right values for the tangent vectors.

$\delta'(s) = \frac{\partial}{\partial s}\gamma(t(s)) = \gamma'(t(s))t'(s) = \gamma'(t(s))(t_j - t_i)$



Clipping Hermite curves

We need to reparametrize that portion of γ:

$$[0,1] \longleftrightarrow [t_i, t_j]$$

$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$

$$\delta(s) \longleftrightarrow \gamma(t(s))$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$

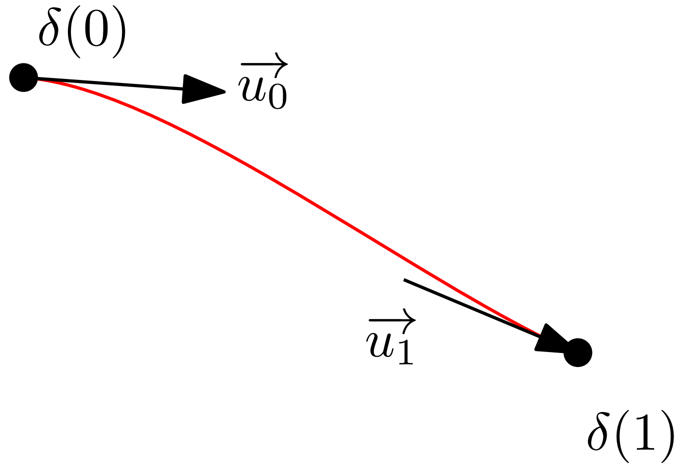
The values of $\delta(0)$ and $\delta(1)$ we know: for instance, $\delta(0) = \gamma(t(0)) = \gamma(t_i)$
We just need to find the right values for the tangent vectors.

$$\delta'(s) = \frac{\partial}{\partial s}\gamma(t(s)) = \gamma'(t(s))t'(s) = \gamma'(t(s))(t_j - t_i)$$

Therefore we have

$$\overrightarrow{\mu_0} = \delta'(0) = \gamma'(t_i)(t_j - t_i)$$

$$\overrightarrow{\mu_1} = \delta'(1) = \gamma'(t_j)(t_j - t_i)$$



Clipping Hermite curves

We need to reparametrize that portion of γ:

$$[0,1] \longleftrightarrow [t_i, t_j]$$

$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$

$$\delta(s) \longleftrightarrow \gamma(t(s))$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$

The values of $\delta(0)$ and $\delta(1)$ we know: for instance, $\delta(0) = \gamma(t(0)) = \gamma(t_i)$
We just need to find the right values for the tangent vectors.

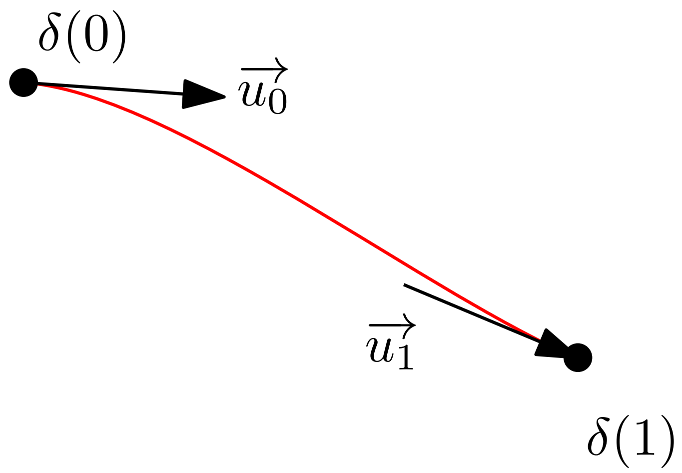
$$\delta'(s) = \frac{\partial}{\partial s}\gamma(t(s)) = \gamma'(t(s))t'(s) = \gamma'(t(s))(t_j - t_i)$$

Therefore we have

$\overrightarrow{\mu_0} = \delta'(0) = \gamma'(t_i)(t_j - t_i)$

$\overrightarrow{\mu_1} = \delta'(1) = \gamma'(t_j)(t_j - t_i)$

The tangent vectors need to be scaled by the length of the parameter interval $[t_i, t_j]$



Clipping Hermite curves

We need to reparametrize that portion of γ:

$$[0,1] \longleftrightarrow [t_i, t_j]$$
$$s \longleftrightarrow t(s) = t_i + s(t_j - t_i)$$
$$\delta(s) \longleftrightarrow \gamma(t(s))$$

and make sure that:

$\delta(0) = \gamma(t_i)$, $\delta'(0) = \gamma'(t_i)$

$\delta(1) = \gamma(t_j)$, $\delta'(1) = \gamma'(t_j)$

The values of $\delta(0)$ and $\delta(1)$ we know: for instance, $\delta(0) = \gamma(t(0)) = \gamma(t_i)$
We just need to find the right values for the tangent vectors.

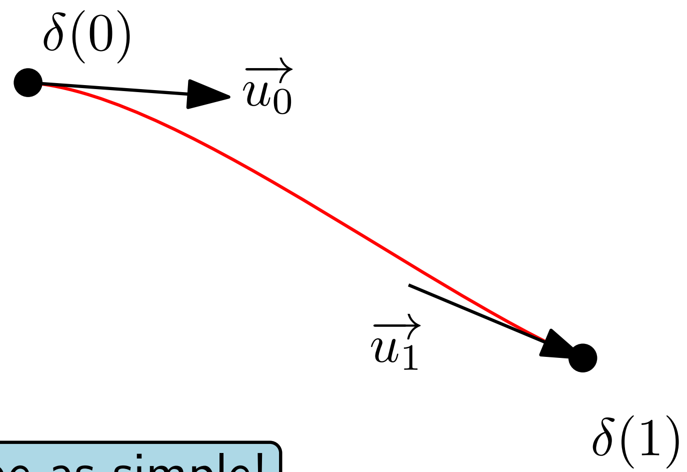
$$\delta'(s) = \frac{\partial}{\partial s}\gamma(t(s)) = \gamma'(t(s))t'(s) = \gamma'(t(s))(t_j - t_i)$$

Therefore we have

$\overrightarrow{\mu_0} = \delta'(0) = \gamma'(t_i)(t_j - t_i)$

$\overrightarrow{\mu_1} = \delta'(1) = \gamma'(t_j)(t_j - t_i)$

The tangent vectors need to be scaled by the length of the parameter interval $[t_i, t_j]$



Notice that clipping in a Lagrange polynomial would not be as simple!

Matrix formulation

Similarly to most curve design methods, Hermite interpolation can be expressed in terms of matrices. This is sometimes convenient.

Matrix formulation

Similarly to most curve design methods, Hermite interpolation can be expressed in terms of matrices. This is sometimes convenient.

$$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}$$

Matrix formulation

Similarly to most curve design methods, Hermite interpolation can be expressed in terms of matrices. This is sometimes convenient.

$$\gamma(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)\overrightarrow{v_0} + (t^3 - t^2)\overrightarrow{v_1}$$

is equivalent to:

$$\gamma(t) = (t^3, t^2, t, 1)\begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} P_0 \\ P_1 \\ \overrightarrow{v_0} \\ \overrightarrow{v_1} \end{pmatrix}$$

Adding some *tension*

One can also define *non-uniform* Hermite polynomials, which depend on a parameter Δ that controls the *tension* of the curve

Adding some *tension*

One can also define *non-uniform* Hermite polynomials, which depend on a parameter Δ that controls the *tension* of the curve

The parameter Δ (for some $\Delta > 0$) scales the two tangent vectors

Adding some *tension*

One can also define *non-uniform* Hermite polynomials, which depend on a parameter Δ that controls the *tension* of the curve

The parameter Δ (for some $\Delta > 0$) scales the two tangent vectors

The formula is modified accordingly:

$$\gamma(t) = (t^3, t^2, t, 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ \Delta\overrightarrow{v_0} \\ \Delta\overrightarrow{v_1} \end{pmatrix}$$

Adding some *tension*

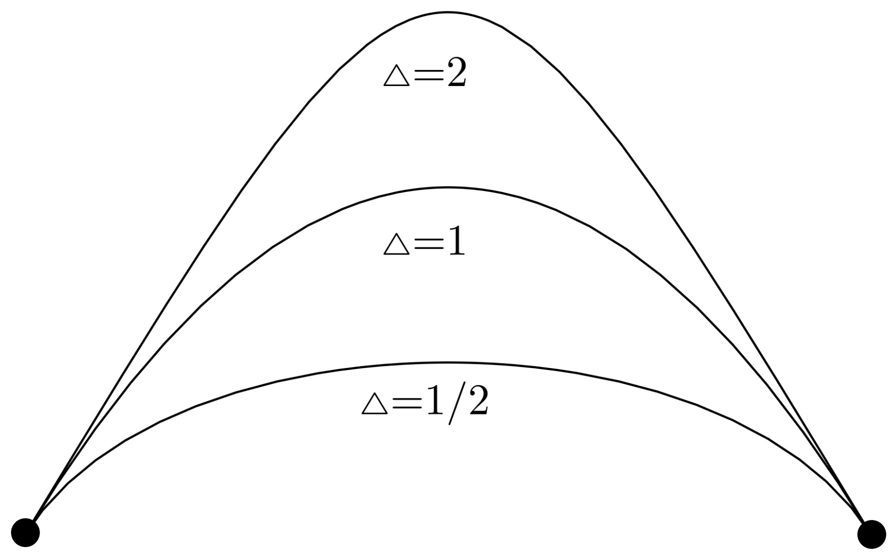
One can also define *non-uniform* Hermite polynomials, which depend on a parameter Δ that controls the *tension* of the curve

The parameter Δ (for some $\Delta > 0$) scales the two tangent vectors

The formula is modified accordingly:

$$\gamma(t) = (t^3, t^2, t, 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ \Delta\overrightarrow{v_0} \\ \Delta\overrightarrow{v_1} \end{pmatrix}$$

Note that for $\Delta = 1$ we obtain the standard (uniform) Hermite polynomial



Adding some *tension*

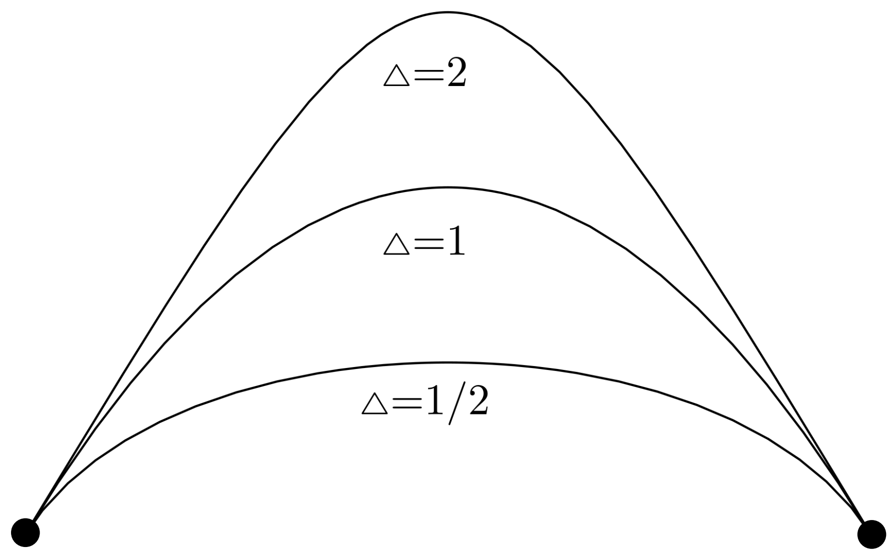
One can also define *non-uniform* Hermite polynomials, which depend on a parameter Δ that controls the *tension* of the curve

The parameter Δ (for some $\Delta > 0$) scales the two tangent vectors

The formula is modified accordingly:

$$\gamma(t) = (t^3, t^2, t, 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ \Delta\overrightarrow{v_0} \\ \Delta\overrightarrow{v_1} \end{pmatrix}$$

Note that for $\Delta = 1$ we obtain
the standard (uniform) Hermite polynomial



The smaller the Δ, the higher the tension in the curve

Higher degree Hermite polynomials

The idea of the cubic Hermite polynomial can be extended to polynomials of higher degree

- For degree-3 we used the two endpoints $(P_0,\, P_1)$ and two tangent vectors at them $(\overrightarrow{v_0},\, \overrightarrow{v_1})$

Higher degree Hermite polynomials

The idea of the cubic Hermite polynomial can be extended to polynomials of higher degree

- For degree-3 we used the two endpoints (P_0, P_1) and two tangent vectors at them ($\overrightarrow{v_0}$, $\overrightarrow{v_1}$)
- For degree-5 we can use two endpoints, two tangent vectors, and two second derivative vectors (i.e., principal normal vectors) at the endpoints

Higher degree Hermite polynomials

The idea of the cubic Hermite polynomial can be extended to polynomials of higher degree

- For degree-3 we used the two endpoints (P_0, P_1) and two tangent vectors at them $(\overrightarrow{v_0}, \overrightarrow{v_1})$
- For degree-5 we can use two endpoints, two tangent vectors, and two second derivative vectors (i.e., principal normal vectors) at the endpoints
- In general, for degree $2k+1$ we can use two endpoints and the first k derivatives at each of them $(2k+2$ items)

  The formulas for such polynomials can be derived as we did for degree 3

  However, higher degree Hermite polynomials are not of much use in practice!