Lecture 16:

# Keyframe Interpolation

FUNDAMENTALS OF COMPUTER GRAPHICS

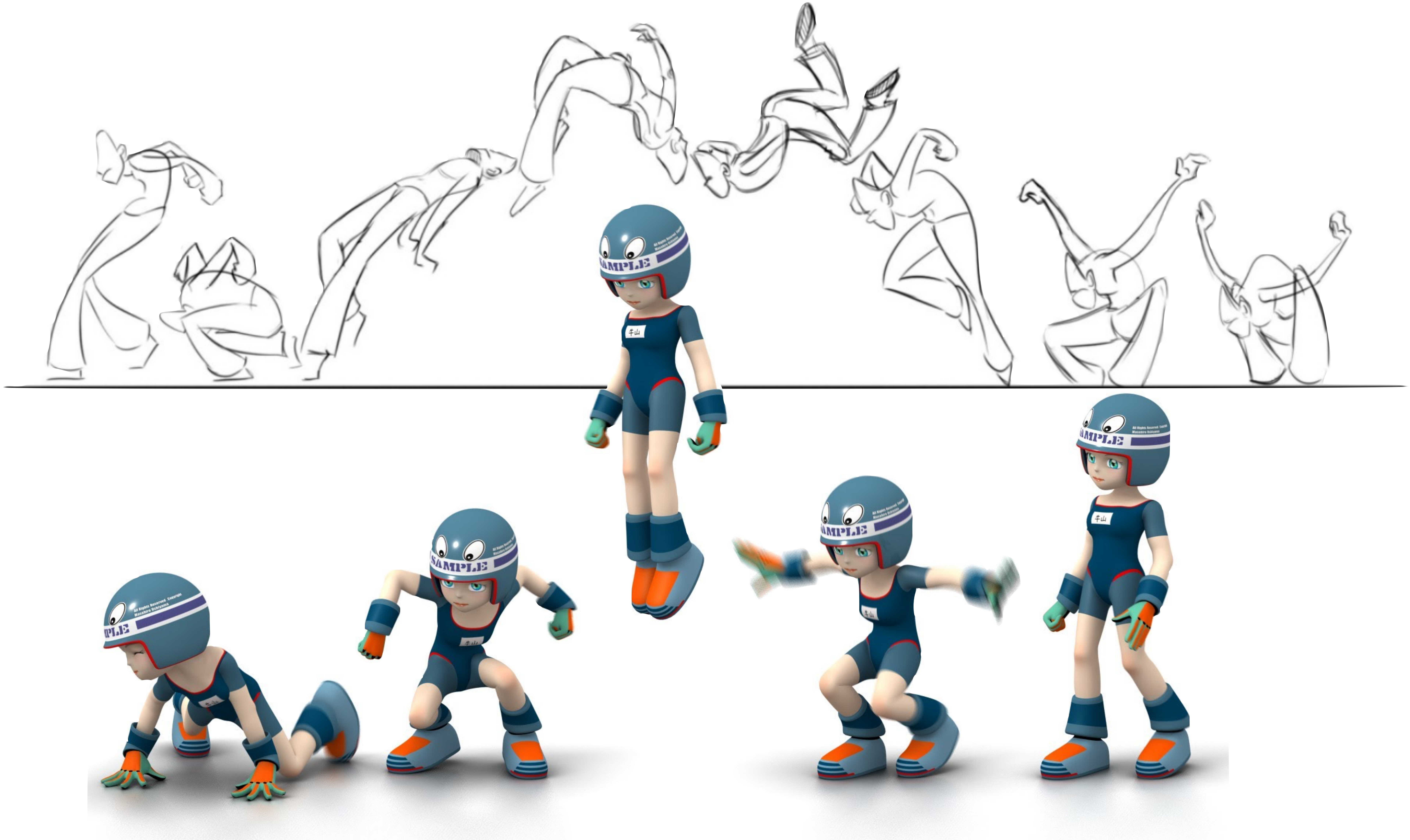
**Animation & Simulation**

**Stanford CS248B**

# 19th century keyframe animation

- Two conditions to make moving images in 19th century
  - at least 10 frames per second
  - a period of blackness between images

# Keyframes



# Modern Zoetrope

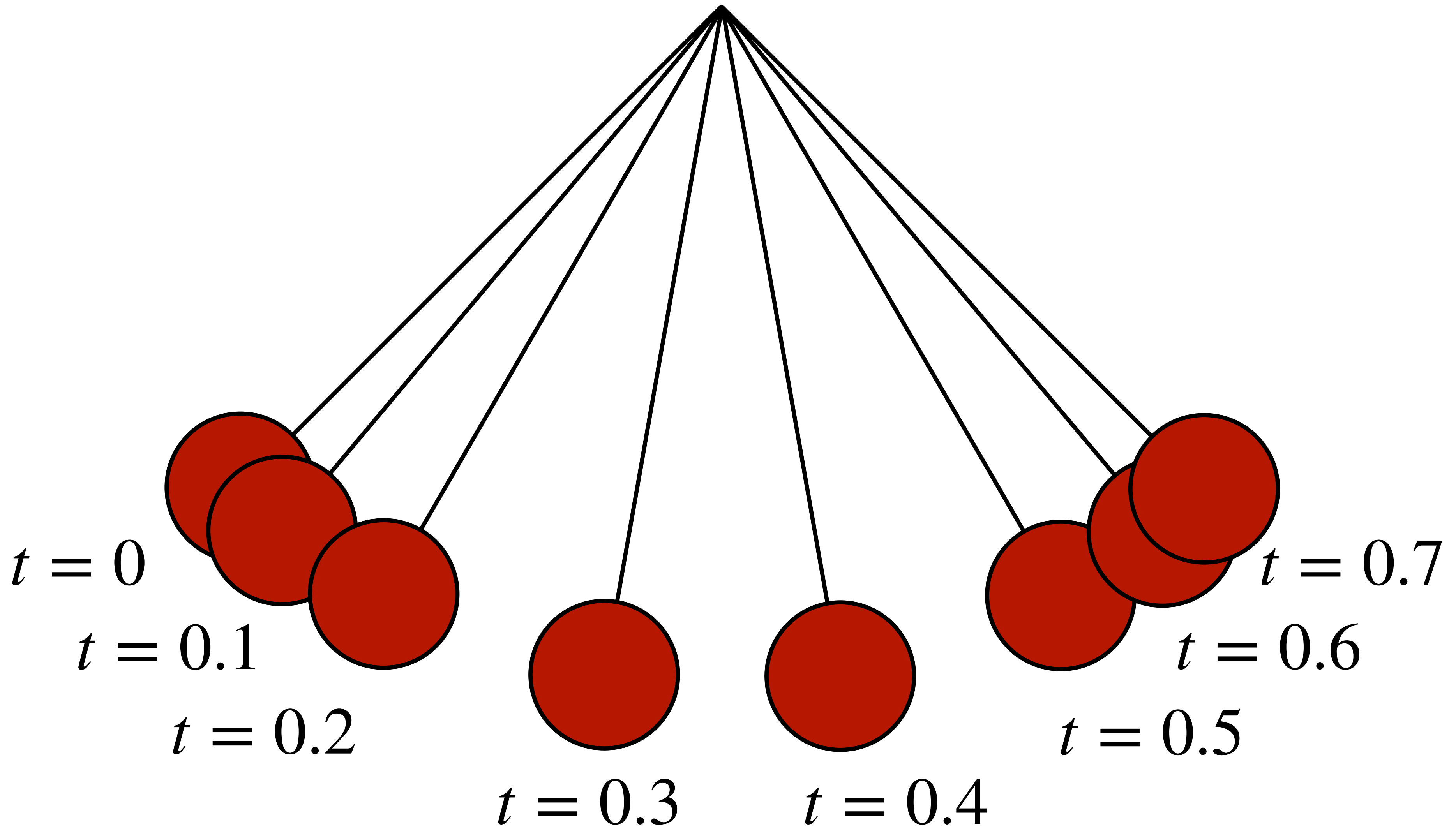
- Instead of drawing figures, animators specify keyframes in 3D.
- Each keyframe is defined by a set of parameters, such as body position and joint angles.
- Zoetrope uses a physical device to interpolate keyframes. In computer animation, we need an algorithm to interpolate keyframes.

# Interpolation keyframes

- Use simple linear interpolation to create in-between frames.

$$x = x_0 + \frac{t - t_0}{t_1 - t_0}(x_1 - x_0)$$

- What is the problem?
  - Motion is not smooth, especially when it's fast.
  - The string length is not constant.



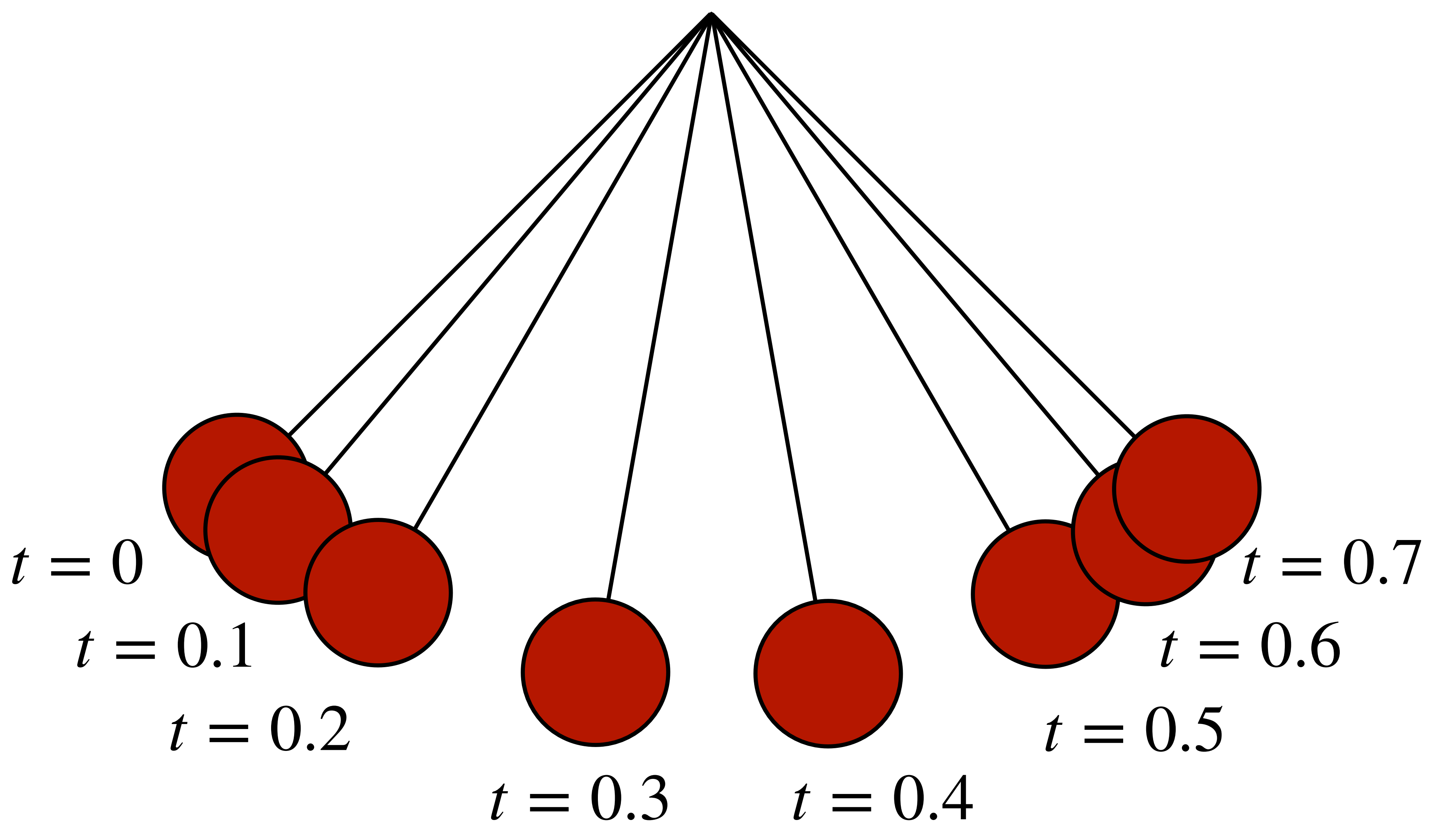
# Cubic interpolation

- **Use three cubic curves to interpolate between two consecutive 3D positions.**

$$x(t') = a_x t'^3 + b_x t'^2 + c_x t' + d_x$$

$$y(t') = a_y t'^3 + b_y t'^2 + c_y t' + d_y$$

$$z(t') = a_z t'^3 + b_z t'^2 + c_z t' + d_z$$




- **Normalize the time between two keyframes such that** $0 \leq t' \leq 1$.

$$t' = \frac{t - t_0}{t_1 - t_0}$$

# Compact representation

- **Put it in a more compact representation**

$$x(t') = a_x t'^3 + b_x t'^2 + c_x t' + d_x$$
$$y(t') = a_y t'^3 + b_y t'^2 + c_y t' + d_y$$
$$z(t') = a_z t'^3 + b_z t'^2 + c_z t' + d_z$$

Put them in the matrix representation and factor out the time variable.

$$\mathbf{Q}(t') = [x(t')\ y('t)\ z(t')]$$

# Compact representation

- **Put it in a more compact representation**

$$x(t') = a_x t'^3 + b_x t'^2 + c_x t' + d_x$$

$$y(t') = a_y t'^3 + b_y t'^2 + c_y t' + d_y$$

$$z(t') = a_z t'^3 + b_z t'^2 + c_z t' + d_z$$

It's nice to have a representation that factors out the time because we can easily compute the gradient of the curve.

$$\mathbf{Q}(t') = [x(t')\ y('t)\ z(t')] = [t'^3\ t'^2\ t'\ 1] \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix} = \mathbf{TC}$$

# Quiz

- **It's nice to factor the time**

**What is the first derivative function of $\mathbf{Q}(t')$?**

$$\dot{\mathbf{Q}}(t') = \dot{\mathbf{T}}\mathbf{C} = \frac{d}{dt'}[t'^3 \; t'^2 \; t' \; 1]\mathbf{C} = [3t'^2 \; 2t' \; 1 \; 0]\mathbf{C}$$

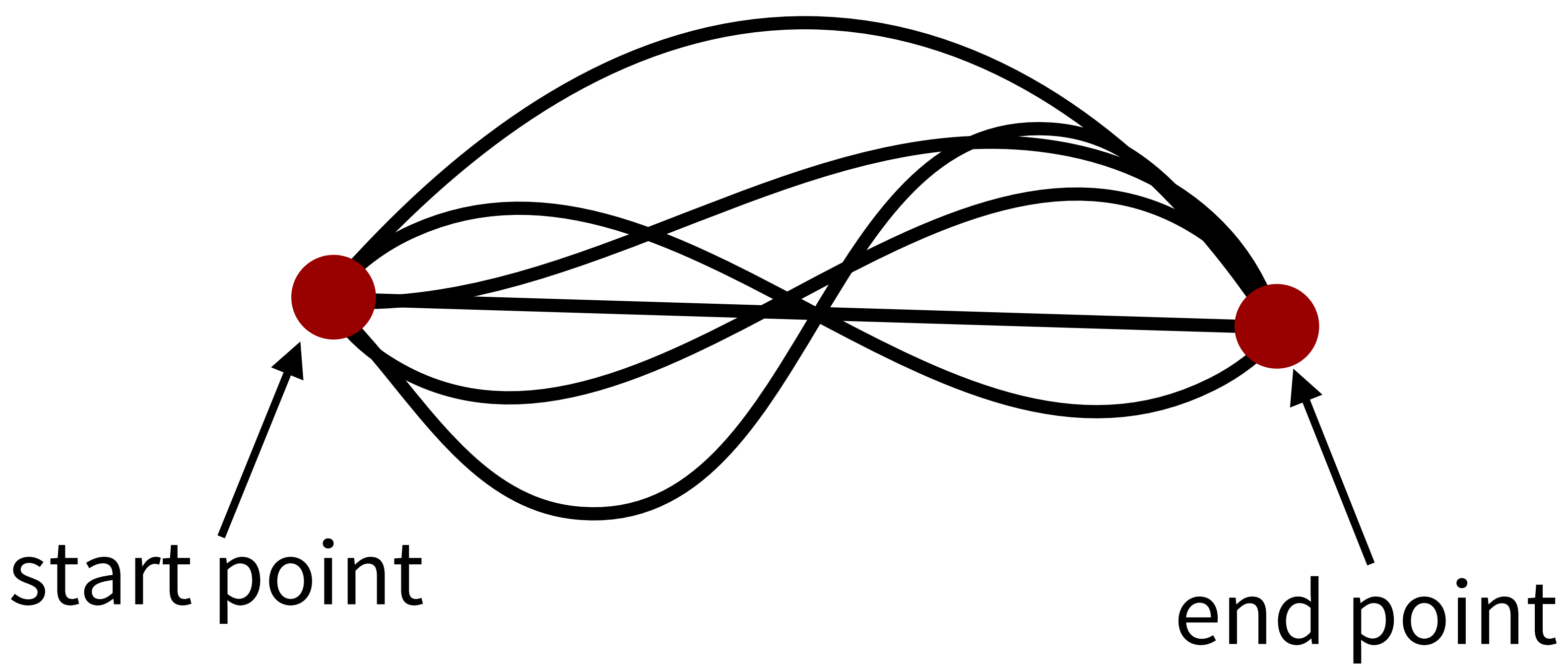
**What about $\ddot{\mathbf{Q}}(t')$?**

$$\ddot{\mathbf{Q}}(t') = \ddot{\mathbf{T}}\mathbf{C} = [6t' \; 2 \; 0 \; 0]\mathbf{C}$$

These derivative functions will be useful later…

# Compact representation

- How do we determine the matrix $\mathbf{C}$?
  - For each cubic function, we need four constraints to determine it.
- Two constraints come from end points, what about other two constraints?
  - Desired shape of the curve.



# Compact representation

- **How do we enforce these geometry constraints in $\mathbf{C}$? The current representation of $\mathbf{C}$ is a bit inconvenient.**

$$\mathbf{C} = \begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{bmatrix}$$

Basis matrix $\mathbf{M}$ falls out of algebra manipulation.

Geometry matrix $\mathbf{G}$ stores parameters of desired geometry constraints. We need **four** of them.

- **We can reparameterize $\mathbf{C}$ as a product of two matrices.**

$$\mathbf{C} = \mathbf{MG} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} g_{1x} & g_{1y} & g_{1z} \\ g_{2x} & g_{2y} & g_{2z} \\ g_{3x} & g_{3y} & g_{3z} \\ g_{4x} & g_{4y} & g_{4z} \end{bmatrix}$$

# Compact representation

Basis matrix $\mathbf{M}$

Geometry matrix $\mathbf{G}$

$$\mathbf{Q} = \mathbf{TMG} = \begin{bmatrix} t'^3 & t'^2 & t' & 1 \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} g_{1x} & g_{1y} & g_{1z} \\ g_{2x} & g_{2y} & g_{2z} \\ g_{3x} & g_{3y} & g_{3z} \\ g_{4x} & g_{4y} & g_{4z} \end{bmatrix}$$

# Hermite curves

- A Hermite curve is determined by
  - Two end points $\mathbf{P}_1$ and $\mathbf{P}_4$
  - Two tangent vectors $\mathbf{R}_1$ and $\mathbf{R}_4$


- Use these elements to construct geometry matrix

$$\mathbf{G}_h = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_4 \\ \mathbf{R}_1 \\ \mathbf{R}_4 \end{bmatrix} = \begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{4x} & p_{4y} & p_{4z} \\ r_{1x} & r_{1y} & r_{1z} \\ r_{4x} & r_{4y} & r_{4z} \end{bmatrix}$$

$$\mathbf{Q}(t') = \mathbf{T}\mathbf{M}_h \begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{4x} & p_{4y} & p_{4z} \\ r_{1x} & r_{1y} & r_{1z} \\ r_{4x} & r_{4y} & r_{4z} \end{bmatrix}$$

What about $\mathbf{M}_h$ ?

# Hermite basis matrix

- To find basis matrix $\mathbf{M}_h$, we need to enforce desired geometry constraints

recall: $\dot{\mathbf{Q}}(t') = \dot{\mathbf{T}}\mathbf{C} = \frac{d}{dt'}[t'^3\ t'^2\ t'\ 1]\mathbf{C} = [3t'^2\ 2t'\ 1\ 0]\mathbf{C}$

  - End points meet $\mathbf{P}_1$ and $\mathbf{P}_4$

$$\mathbf{Q}(0) = [0 \quad 0 \quad 0 \quad 1]\,\mathbf{M}_h\mathbf{G}_h$$

$$\mathbf{Q}(1) = [1 \quad 1 \quad 1 \quad 1]\,\mathbf{M}_h\mathbf{G}_h$$

  - Tangent vectors meet $\mathbf{R}_1$ and $\mathbf{R}_4$

$$\dot{\mathbf{Q}}(0) = [0 \quad 0 \quad 1 \quad 0]\,\mathbf{M}_h\mathbf{G}_h$$

$$\dot{\mathbf{Q}}(1) = [3 \quad 2 \quad 1 \quad 0]\,\mathbf{M}_h\mathbf{G}_h$$

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \mathbf{M}_h\mathbf{G}_h = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_4 \\ \mathbf{R}_1 \\ \mathbf{R}_4 \end{bmatrix}$$

# Quiz

Given $\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \mathbf{M}_h \mathbf{G}_h = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_4 \\ \mathbf{R}_1 \\ \mathbf{R}_4 \end{bmatrix}$

What is $\mathbf{M}_h$?

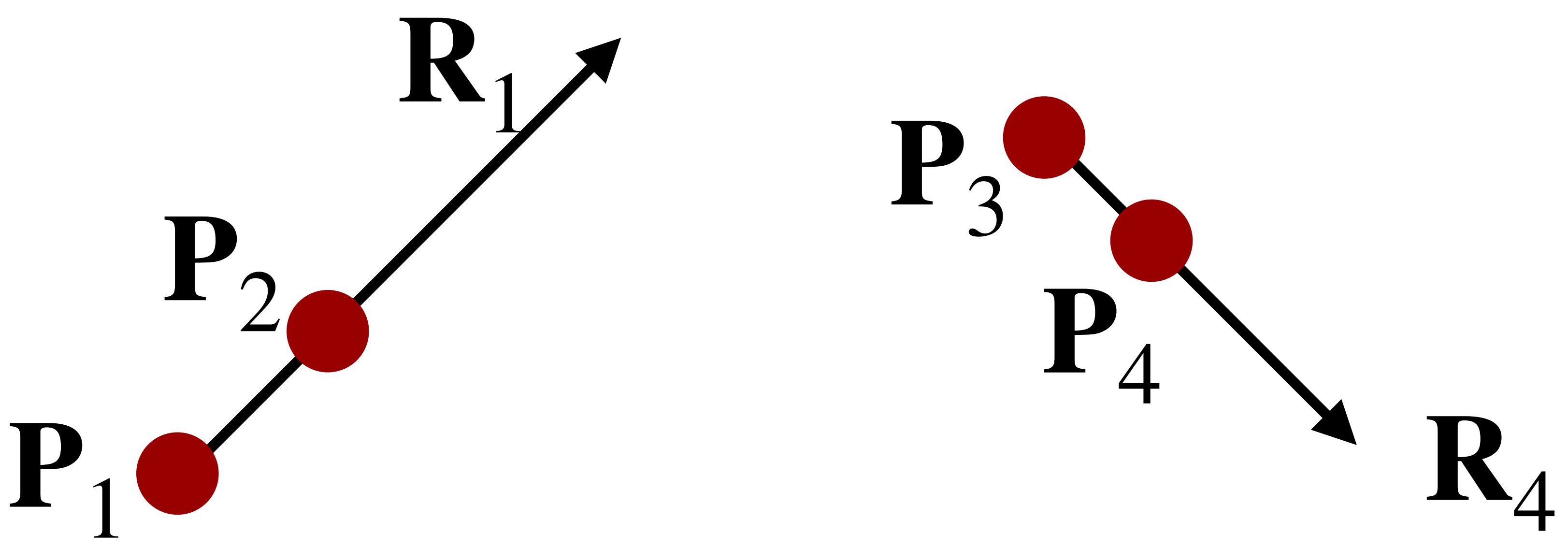
$$\mathbf{M}_h = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

# Bézier curves

- **Indirectly specify tangent vectors by specifying two intermediate points**

$$\mathbf{R}_1 = 3(\mathbf{P}_2 - \mathbf{P}_1)$$

$$\mathbf{R}_4 = 3(\mathbf{P}_4 - \mathbf{P}_3)$$




- **Use these elements to construct geometry matrix**

$$\mathbf{G}_b = \begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{2x} & p_{2y} & p_{2z} \\ p_{3x} & p_{3y} & p_{3z} \\ p_{4x} & p_{4y} & p_{4z} \end{bmatrix}$$

$$\mathbf{Q}(t') = \mathbf{T}\mathbf{M}_b \begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{2x} & p_{2y} & p_{2z} \\ p_{3x} & p_{3y} & p_{3z} \\ p_{4x} & p_{4y} & p_{4z} \end{bmatrix}$$

What about $\mathbf{M}_b$ ?

# Bézier basis matrix

- **Exploit the relation between Hermite and Bezier geometry matrices to find the basis matrix for Bézier curve,** $\mathbf{M}_b$**.**

$$\mathbf{G}_h = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_4 \\ \mathbf{R}_1 \\ \mathbf{R}_4 \end{bmatrix} = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \\ \mathbf{P}_4 \end{bmatrix}$$

recall:

$$\mathbf{R}_1 = 3(\mathbf{P}_2 - \mathbf{P}_1)$$
$$\mathbf{R}_4 = 3(\mathbf{P}_4 - \mathbf{P}_3)$$

- $\mathbf{Q} = \mathbf{T}\mathbf{M}_h\mathbf{G}_h = \mathbf{T}\mathbf{M}_h(\mathbf{M}_{hb}\mathbf{G}_b) = \mathbf{T}(\mathbf{M}_h\mathbf{M}_{hb})\mathbf{G}_b = \mathbf{T}\mathbf{M}_b\mathbf{G}_b$

$$\mathbf{M}_b = \mathbf{M}_h\mathbf{M}_{hb}$$

# Bézier basis matrix

- **Exploit the relation between Hermite and Bezier geometry matrices to find the basis matrix for Bézier curve, $\mathbf{M}_b$.**

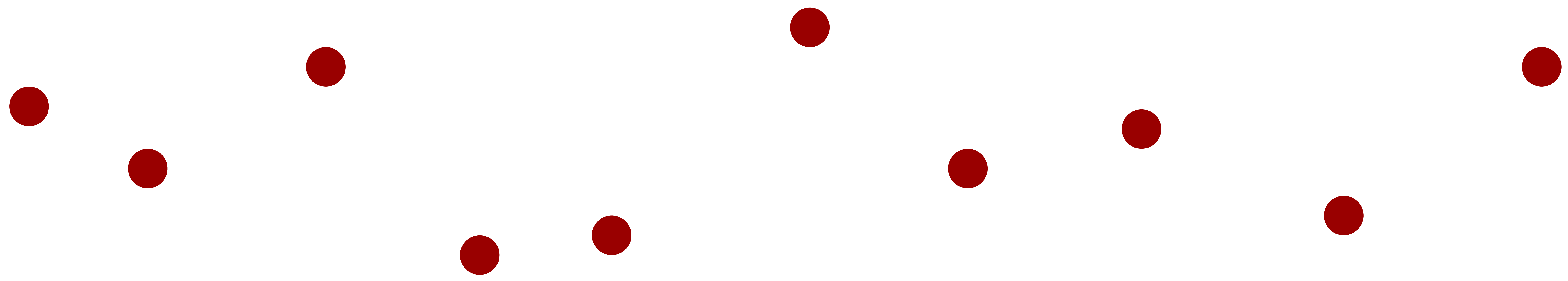
$$\mathbf{G}_h = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_4 \\ \mathbf{R}_1 \\ \mathbf{R}_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \\ \mathbf{P}_4 \end{bmatrix} = \mathbf{M}_{hb}\mathbf{G}_b$$

- $\mathbf{Q} = \mathbf{T}\mathbf{M}_h\mathbf{G}_h = \mathbf{T}\mathbf{M}_h(\mathbf{M}_{hb}\mathbf{G}_b) = \mathbf{T}(\mathbf{M}_h\mathbf{M}_{hb})\mathbf{G}_b = \mathbf{T}\mathbf{M}_b\mathbf{G}_b$

$$\mathbf{M}_b = \mathbf{M}_h\mathbf{M}_{hb} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

# Complex curves

- What if we want to model a curve that passes through these points?



- Problem with higher order polynomials
  - Wiggly curves
  - No local control

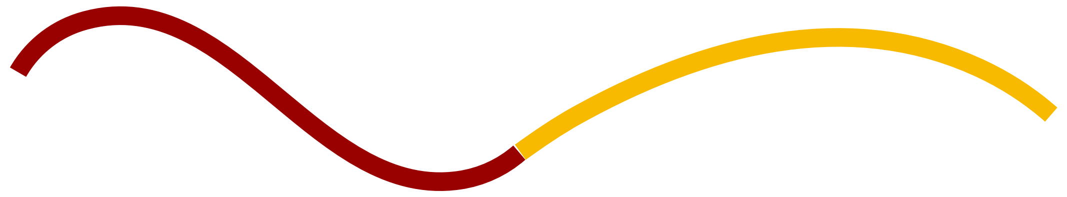
# Splines

- A piecewise polynomial that has locally very simple form, yet be globally flexible and smooth
- There are three nice properties of splines we'd like to have
  - Continuity
  - Local control
  - Interpolation

# Continuity

- $C^0$: positions coincide, velocities don't

- $C^1$: positions and velocities coincide



- $C^2$: positions, velocities and accelerations coincide

Often the difference between $C^1$ and $C^2$ is not that obvious.

# Local control

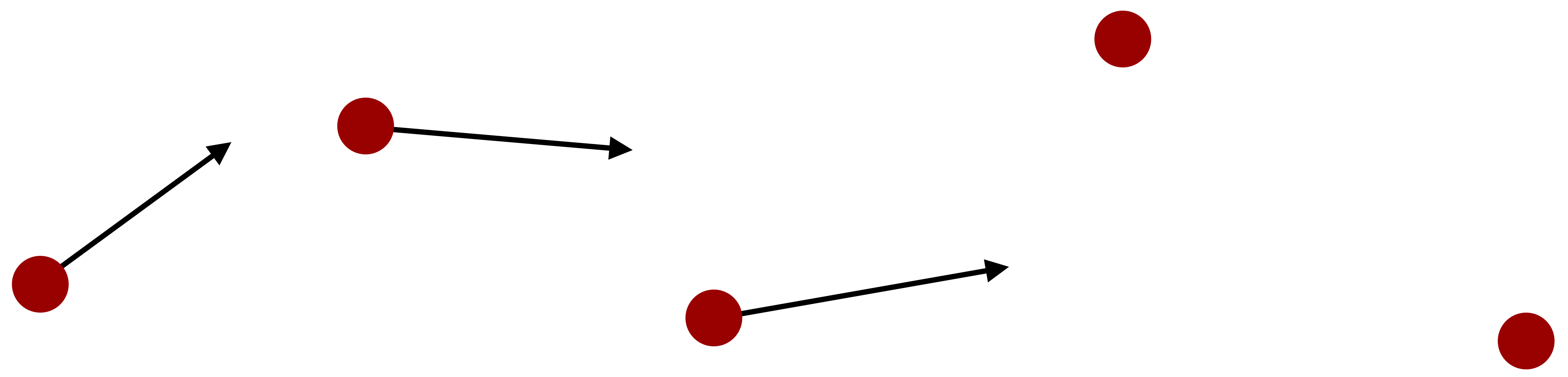
- We'd like to have each control point on the spline only affect some well-defined neighborhood around that point.
- Polynomial functions don't have local control; moving a single keyframe affects the whole curve.

# Interpolation

- We'd like to have a spline interpolating the control points so that the spline always passes through every control points.
- Bézier curves do not necessarily pass through all the control points.

# Catmull-Rom splines

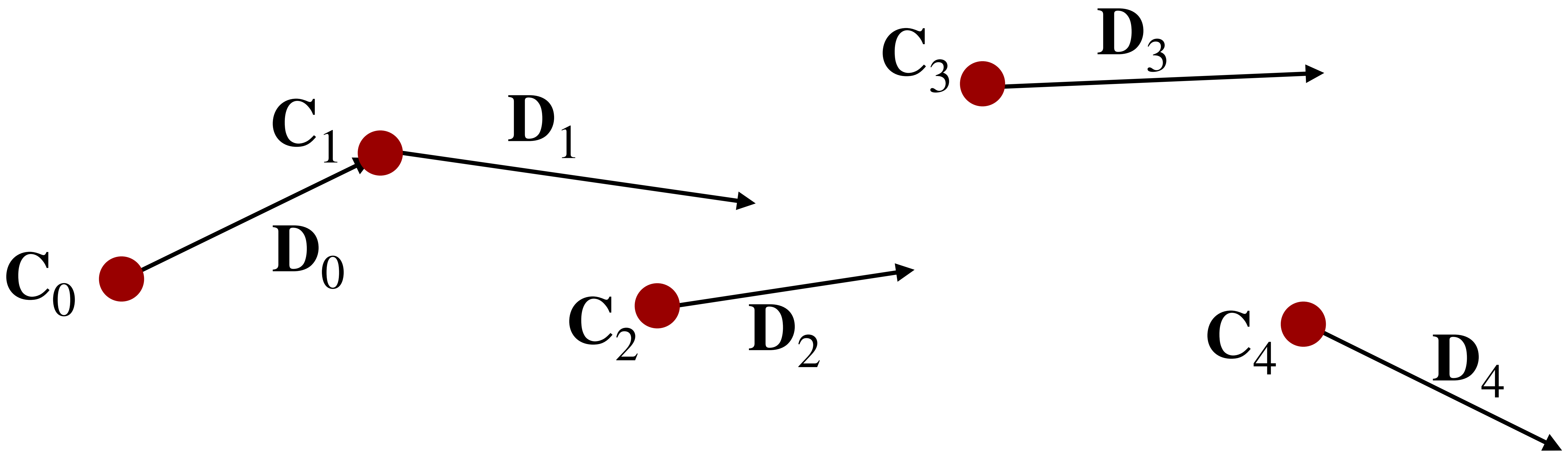
- Each polynomial in a spline can be a Hermite curve.



- We need a rule to determine tangents shared by two consecutive Hermite curves.

# Catmull-Rom splines

- Each polynomial in a spline can be a Hermite curve.




$$\mathbf{D}_0 = \mathbf{C}_1 - \mathbf{C}_0$$

$$\mathbf{D}_1 = \frac{1}{2}(\mathbf{C}_2 - \mathbf{C}_0)$$

$$\mathbf{D}_2 = \frac{1}{2}(\mathbf{C}_3 - \mathbf{C}_1)$$

$$\vdots$$

$$\mathbf{D}_n = (\mathbf{C}_n - \mathbf{C}_{n-1})$$

- We need a rule to determine tangents shared by two consecutive Hermite curves.

# Catmull-Rom basis matrix

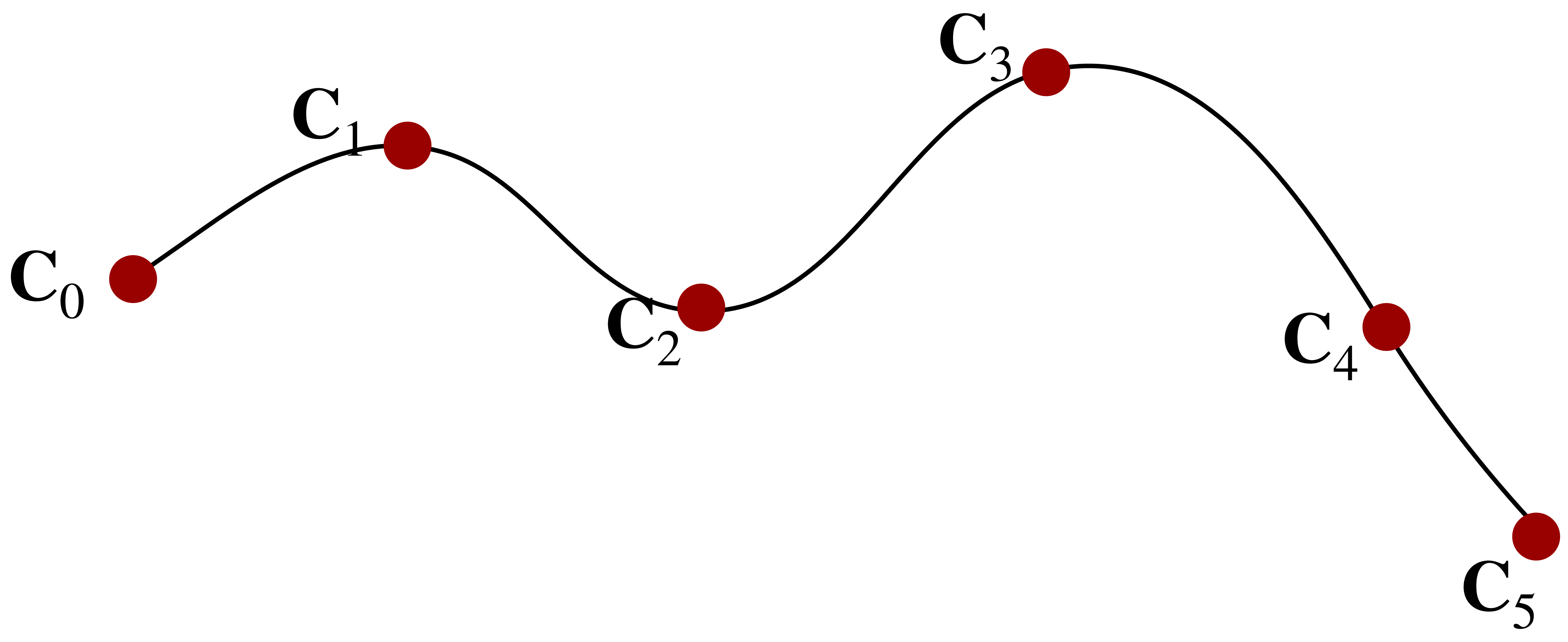
- For interior portion of Catmull-Rom spline, we can derive the basis matrix and use four neighboring keyframes to form the geometry matrix.

$$\mathbf{Q} = \mathbf{T}\begin{bmatrix} \frac{-1}{2} & \frac{3}{2} & \frac{-3}{2} & \frac{1}{2} \\ 1 & \frac{-5}{2} & 2 & \frac{-1}{2} \\ \frac{-1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} \mathbf{C_1} \\ \mathbf{C_2} \\ \mathbf{C_3} \\ \mathbf{C_4} \end{bmatrix}$$

- For the boundary portion of Catmull-Rom, we can simply use the Hermite curve formulation.

# Quiz

**Which portion of the Catmull-Rom spline is drawn by this equation?**




$$\mathbf{Q} = \mathbf{T}\begin{bmatrix} \frac{-1}{2} & \frac{3}{2} & \frac{-3}{2} & \frac{1}{2} \\ 1 & \frac{-5}{2} & 2 & \frac{-1}{2} \\ \frac{-1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} \mathbf{C_1} \\ \mathbf{C_2} \\ \mathbf{C_3} \\ \mathbf{C_4} \end{bmatrix}$$

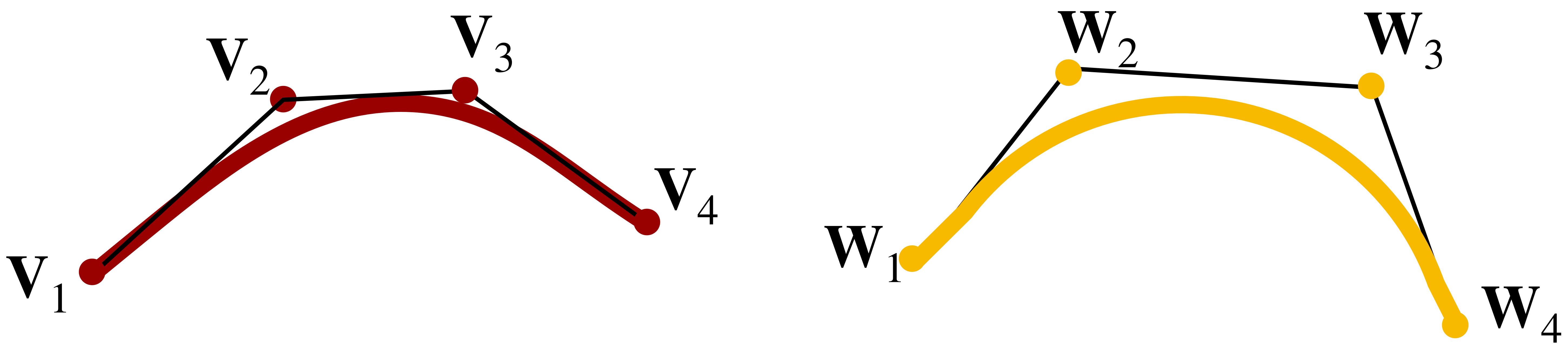
The portion between $\mathbf{C}_2$ and $\mathbf{C}_3$

# Properties of Catmll-Rom Splines

- $C^2$ continuity ✗
- Local control ✓
- Interpolation ✓

# B-splines

- We can join multiple Bézier curves to create B-splines.




- We will do it in such way that $C^2$ continuity is enforced.

Recall $\ddot{\mathbf{Q}}(t') = [6t' \; 2 \; 0 \; 0]\mathbf{M}_b \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \\ \mathbf{P}_4 \end{bmatrix}$

Positions: $\mathbf{Q}_v(1) = \mathbf{Q}_w(0)$

Velocities: $\dot{\mathbf{Q}}_v(1) = \dot{\mathbf{Q}}_w(0)$

Accelerations: $\ddot{\mathbf{Q}}_v(1) = \ddot{\mathbf{Q}}_w(0)$

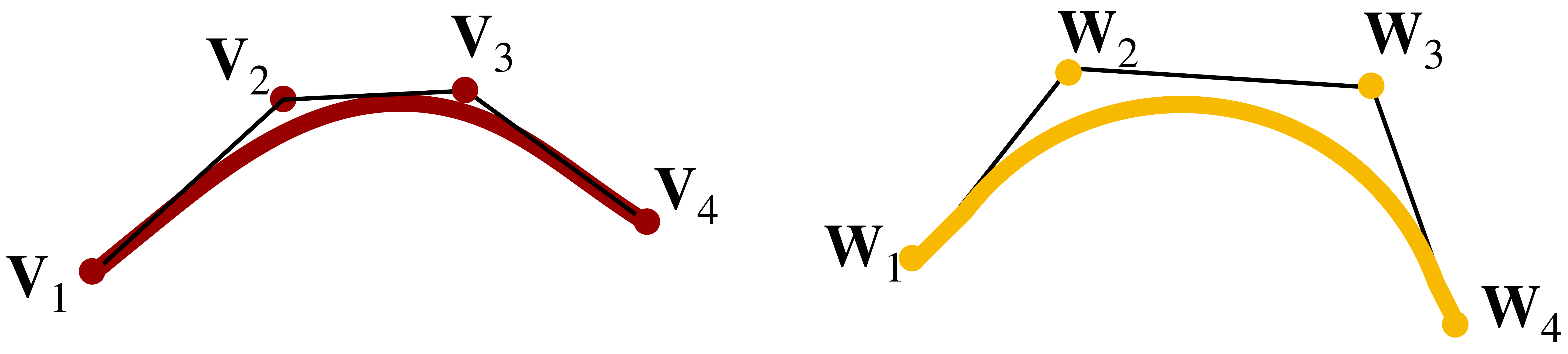
$$\mathbf{V}_4 = \mathbf{W}_1$$

$$\mathbf{V}_4 - \mathbf{V}_3 = \mathbf{W}_2 - \mathbf{W}_1$$

$$\mathbf{V}_2 - 2\mathbf{V}_3 + \mathbf{V}_4 = \mathbf{W}_1 - 2\mathbf{W}_2 + \mathbf{W}_3$$

# B-splines

- We can join multiple Bézier curves to create B-splines.

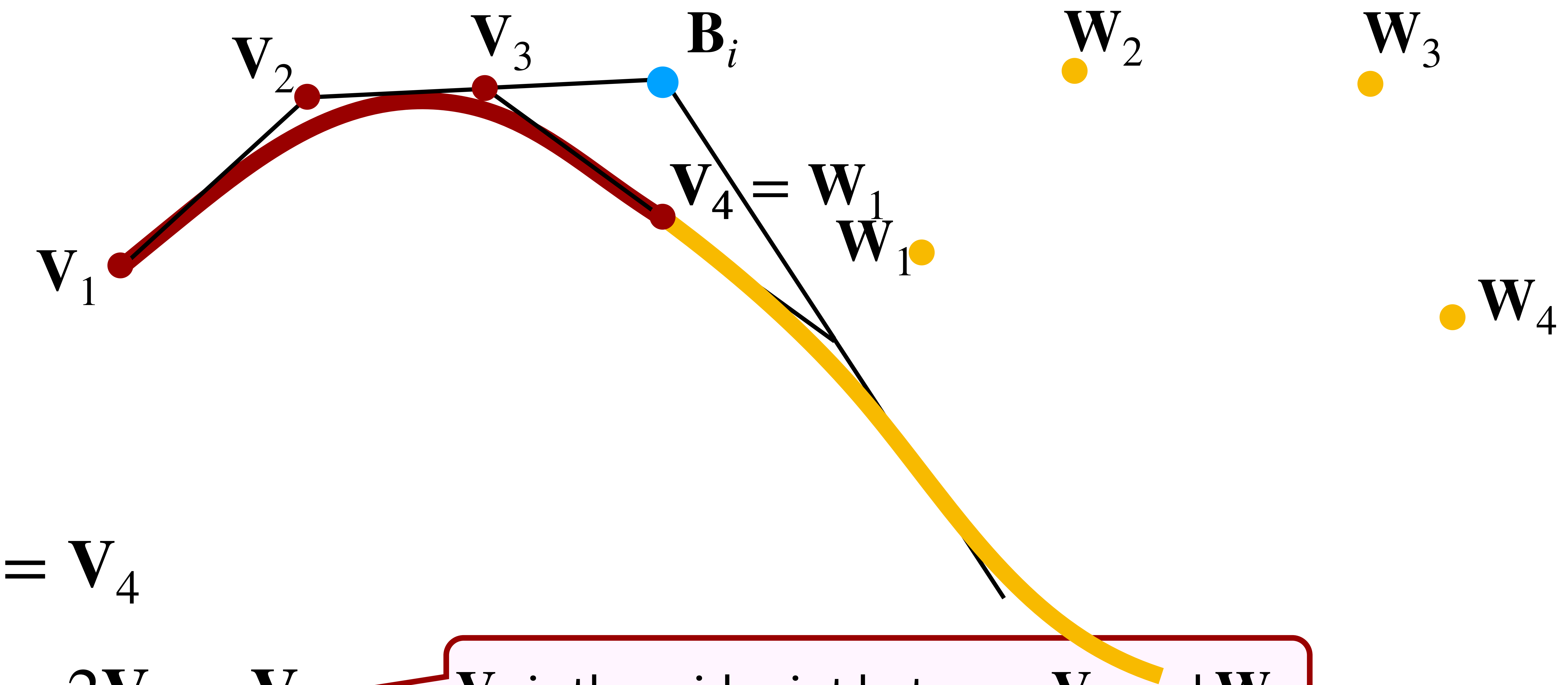



- We will do it in such way that $C^2$ continuity is enforced.

$$W_1 = V_4$$
$$W_2 = 2V_4 - V_3$$
$$W_3 = 2W_2 - (2V_3 - V_2)$$

$$V_4 = W_1$$
$$V_4 - V_3 = W_2 - W_1$$
$$V_2 - 2V_3 + V_4 = W_1 - 2W_2 + W_3$$

# B-splines




$$\mathbf{W}_1 = \mathbf{V}_4$$

$$\mathbf{W}_2 = 2\mathbf{V}_4 - \mathbf{V}_3$$

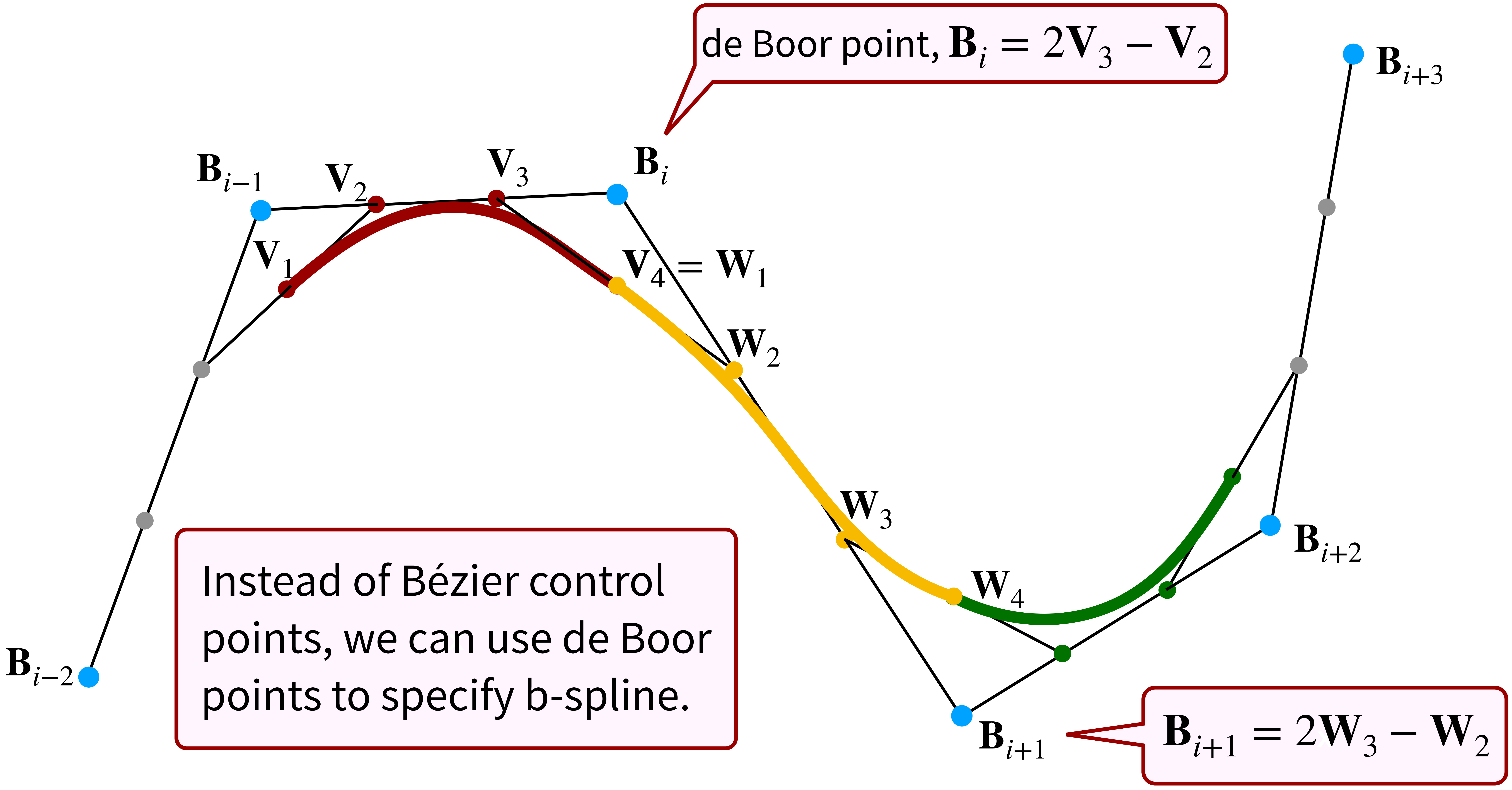
$\mathbf{V}_4$ is the midpoint between $\mathbf{V}_3$ and $\mathbf{W}_2$

$$\mathbf{W}_3 = 2\mathbf{W}_2 - (2\mathbf{V}_3 - \mathbf{V}_2)$$

$$= 2\mathbf{W}_2 - \mathbf{B}_i, \text{ where } \mathbf{B}_i = 2\mathbf{V}_3 - \mathbf{V}_2$$

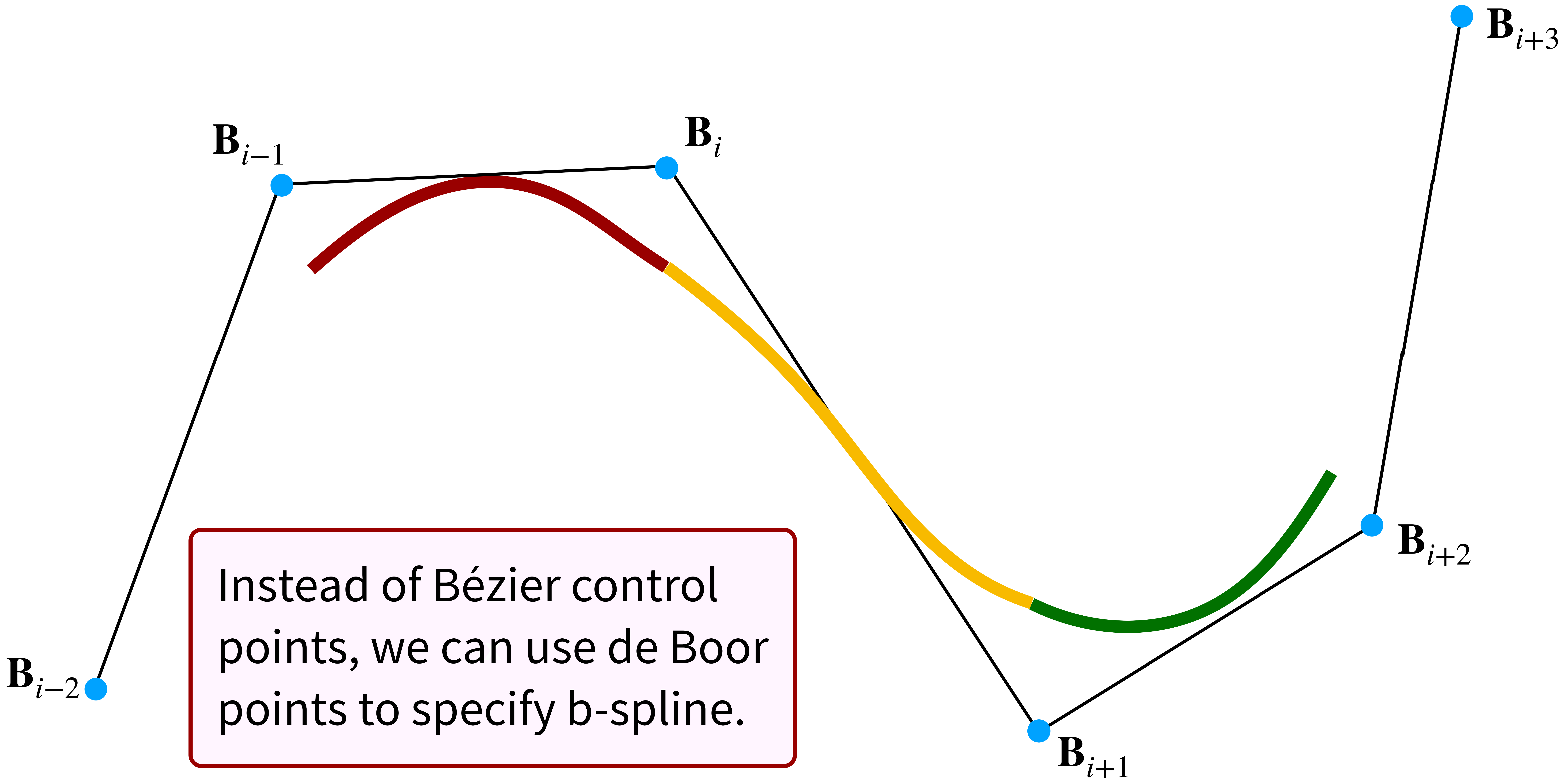
$\mathbf{V}_3$ is the midpoint between $\mathbf{V}_2$ and $\mathbf{B}_i$

# de Boor points



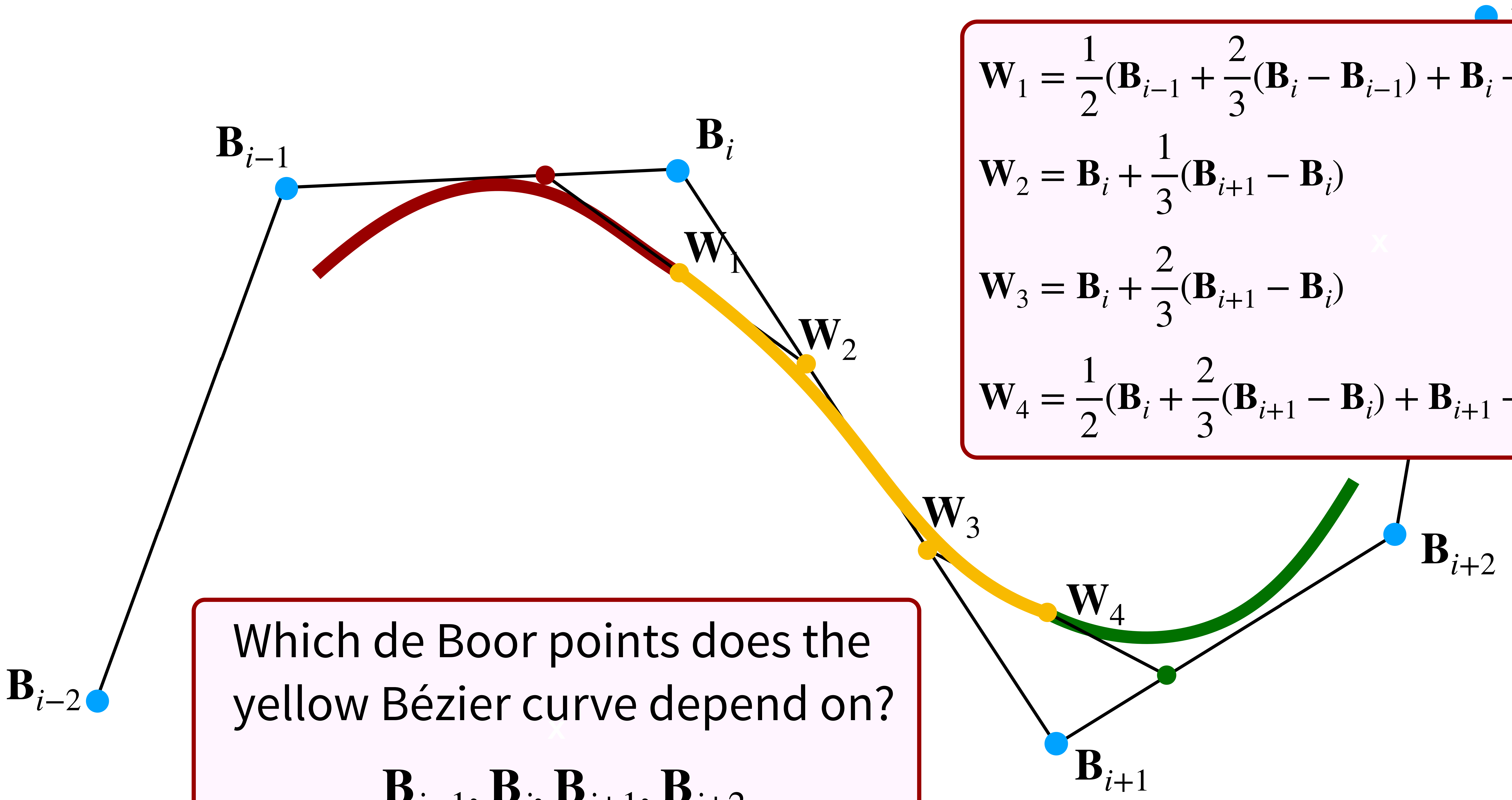

Instead of Bézier control points, we can use de Boor points to specify b-spline.

# de Boor points




Instead of Bézier control points, we can use de Boor points to specify b-spline.

# Relation between de Boor points and Bézier points




$$\mathbf{W}_1 = \frac{1}{2}(\mathbf{B}_{i-1} + \frac{2}{3}(\mathbf{B}_i - \mathbf{B}_{i-1}) + \mathbf{B}_i + \frac{1}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i))$$

$$\mathbf{W}_2 = \mathbf{B}_i + \frac{1}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i)$$

$$\mathbf{W}_3 = \mathbf{B}_i + \frac{2}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i)$$

$$\mathbf{W}_4 = \frac{1}{2}(\mathbf{B}_i + \frac{2}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i) + \mathbf{B}_{i+1} + \frac{1}{3}(\mathbf{B}_{i+2} - \mathbf{B}_{i+1}))$$

Which de Boor points does the yellow Bézier curve depend on?

$\mathbf{B}_{i-1}, \mathbf{B}_i, \mathbf{B}_{i+1}, \mathbf{B}_{i+2}$

# B-splines

$$\begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \mathbf{W}_3 \\ \mathbf{W}_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{i-1} \\ \mathbf{B}_i \\ \mathbf{B}_{i+1} \\ \mathbf{B}_{i+2} \end{bmatrix}$$

$$\mathbf{W}_1 = \frac{1}{2}(\mathbf{B}_{i-1} + \frac{2}{3}(\mathbf{B}_i - \mathbf{B}_{i-1}) + \mathbf{B}_i + \frac{1}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i))$$

$$\mathbf{W}_2 = \mathbf{B}_i + \frac{1}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i)$$

$$\mathbf{W}_3 = \mathbf{B}_i + \frac{2}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i)$$

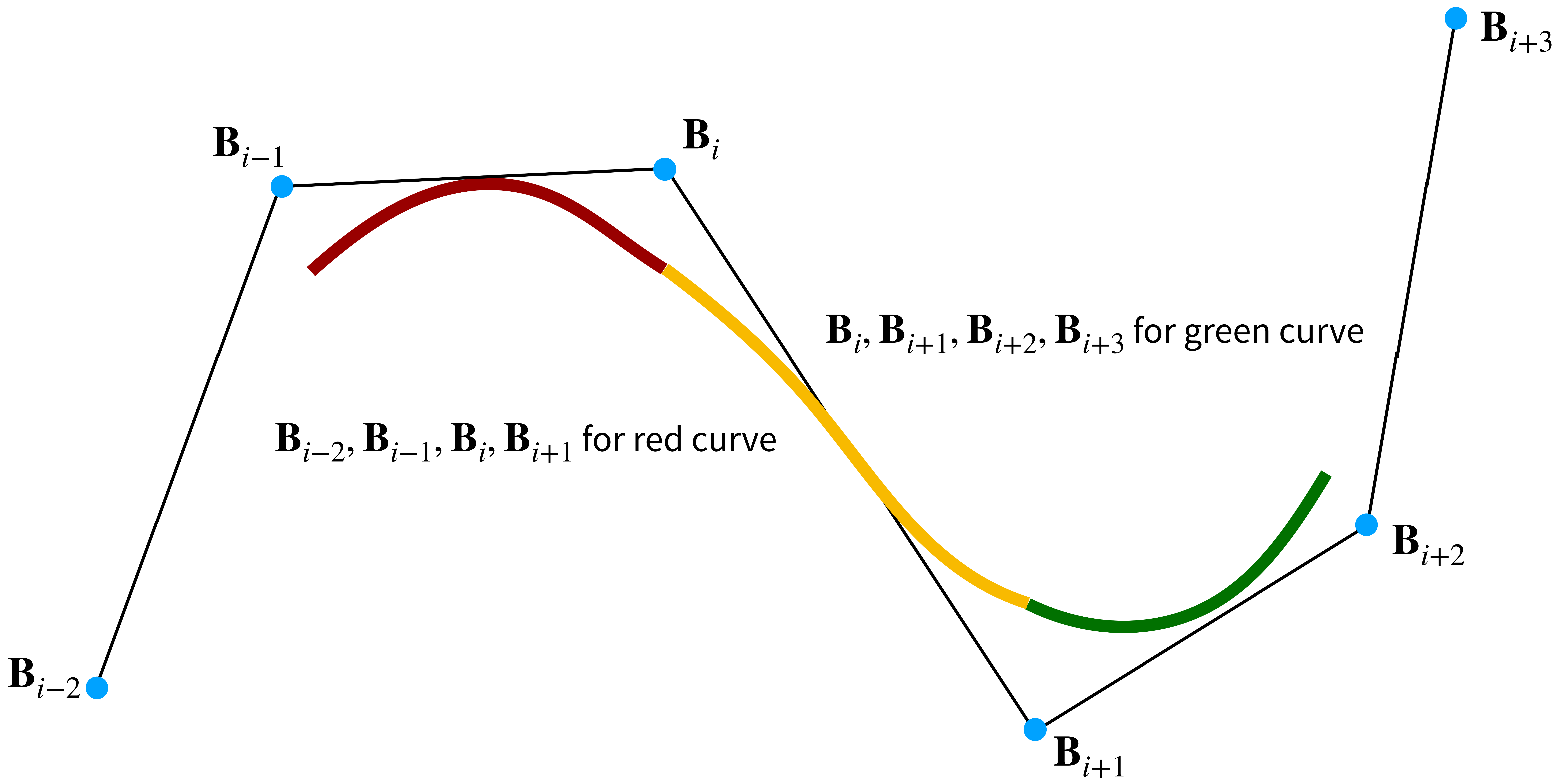
$$\mathbf{W}_4 = \frac{1}{2}(\mathbf{B}_i + \frac{2}{3}(\mathbf{B}_{i+1} - \mathbf{B}_i) + \mathbf{B}_{i+1} + \frac{1}{3}(\mathbf{B}_{i+2} - \mathbf{B}_{i+1}))$$

Basis matrix for B-splines defined by de Boor points

$$\mathbf{Q} = \mathbf{T}\mathbf{M}_b \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \mathbf{W}_3 \\ \mathbf{W}_4 \end{bmatrix}$$

# Quiz

- What de Boor points are used to compute the red and green Bézier curves?



# Properties of B-Splines

- $C^2$ continuity ✓
- Local control ✓
- Interpolation ✗

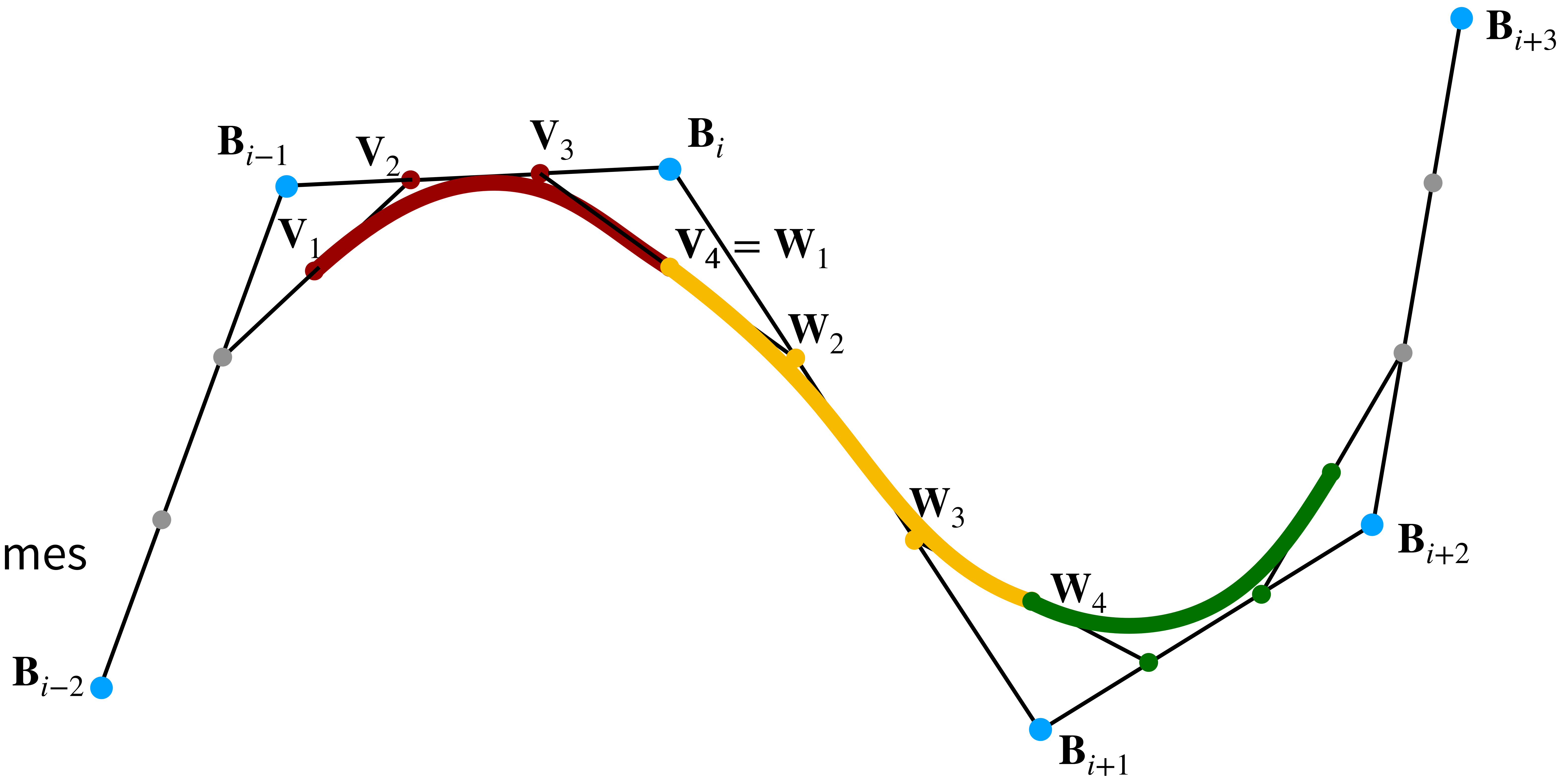
# Endpoints

**It would be nice if we could at least control the endpoints of the splines explicitly.**

$$\begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{i-2} \\ \mathbf{B}_{i-1} \\ \mathbf{B}_i \\ \mathbf{B}_{i+1} \end{bmatrix}$$

If we repeat the de Boor endpoints 3 times

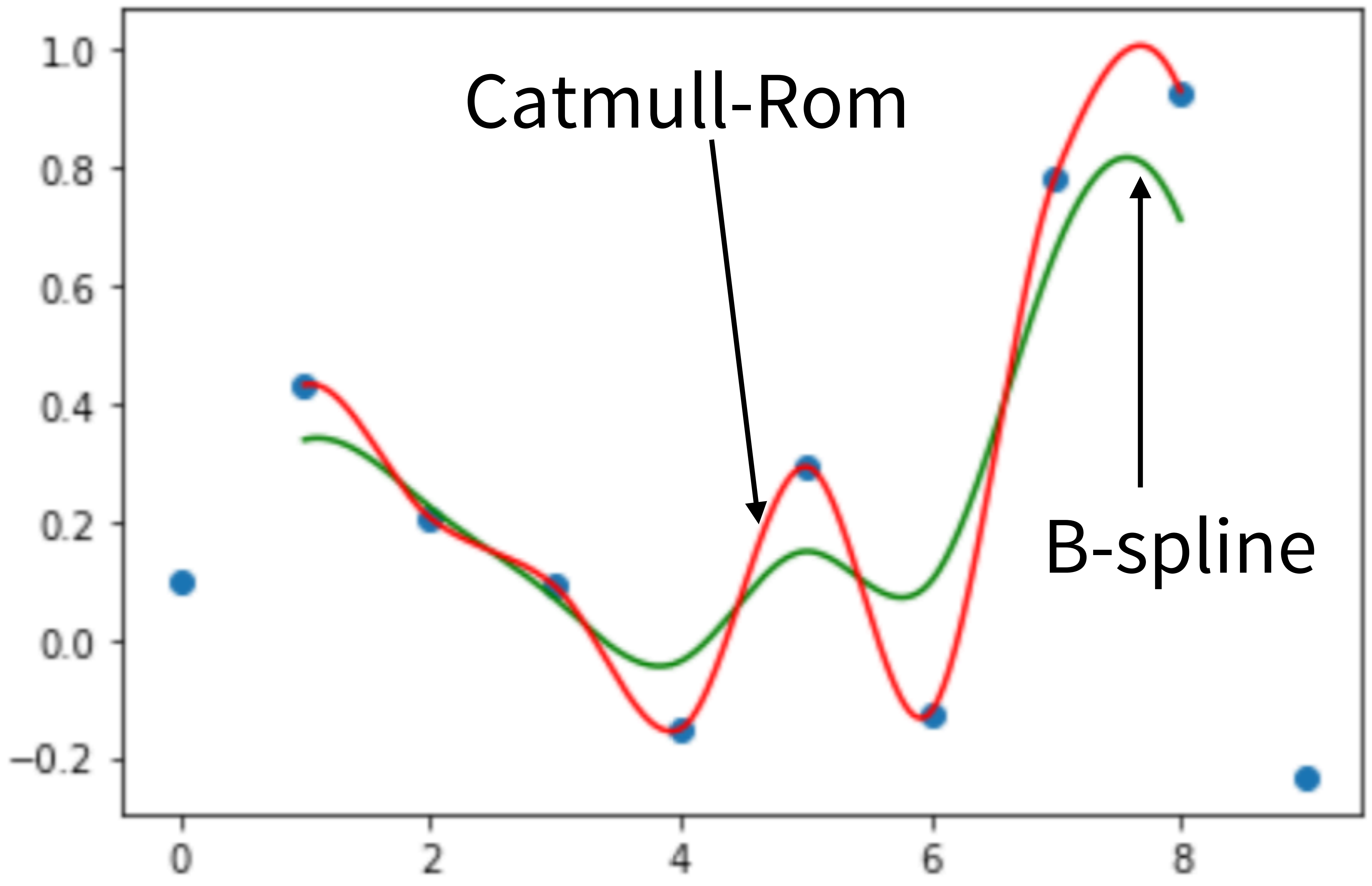
$$\begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix} \begin{bmatrix} \mathbf{B}_{i-2} \\ \mathbf{B}_{i-2} \\ \mathbf{B}_{i-2} \\ \mathbf{B}_{i-1} \end{bmatrix}$$




$\mathbf{V}_1 = \mathbf{B}_{i-2}$ so the spline will start from $\mathbf{B}_{i-2}$

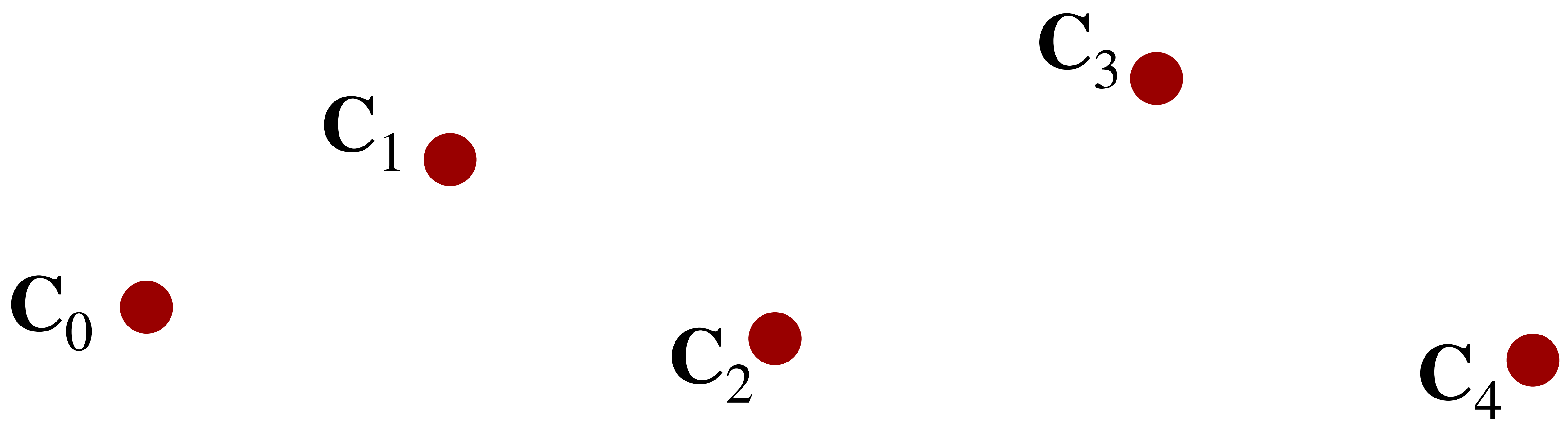
# Quiz

Which one is Catmull-Rom? Which one is B-spline?



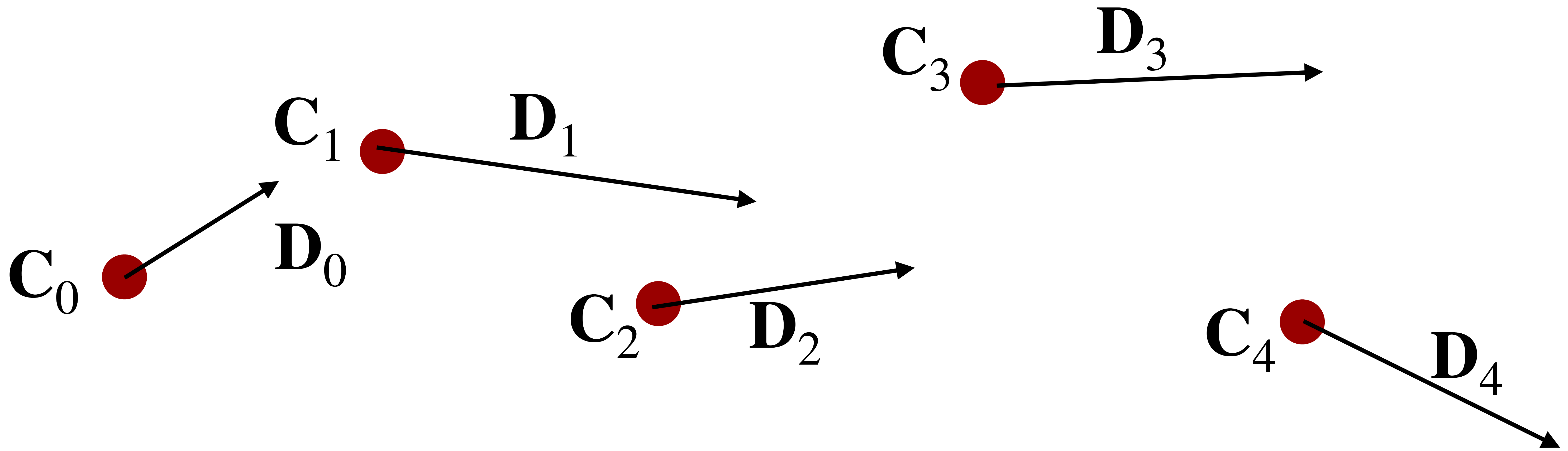
# $C^2$ interpolating splines

- How can we keep the C2 continuity of B-splines but get interpolation property as well?
- Suppose we have a set of points representing keyframes, our goal is to find a C2 spline that passes through all the points.



# $C^2$ interpolating splines

- Make each pair of segments share an arbitrary tangent will only give you C1.
- Need to solve for D's such that C2 continuity is enforced between segments.



# $C^2$ interpolating splines

- **If we want to create a Bezier curve between each pair of consecutive points, what are Bezier control points in terms of C's and D's?**

$$\mathbf{V}_0 = \mathbf{C}_0$$

$$\mathbf{V}_1 = \mathbf{C}_0 + \frac{1}{3}\mathbf{D}_0$$

$$\mathbf{V}_2 = \mathbf{C}_1 - \frac{1}{3}\mathbf{D}_1$$

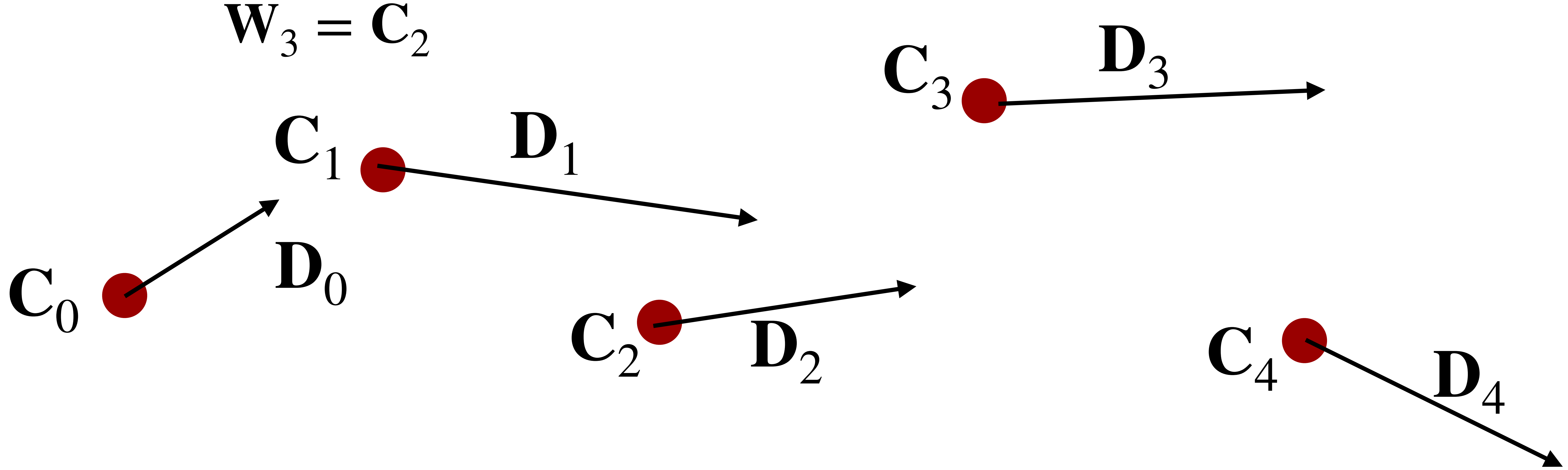
$$\mathbf{V}_3 = \mathbf{C}_1$$

$$\mathbf{W}_0 = \mathbf{C}_1$$

$$\mathbf{W}_1 = \mathbf{C}_1 + \frac{1}{3}\mathbf{D}_1$$

$$\mathbf{W}_2 = \mathbf{C}_2 - \frac{1}{3}\mathbf{D}_2$$

$$\mathbf{W}_3 = \mathbf{C}_2$$



# $C^2$ interpolating splines

- **If we want to create a Bezier curve between each pair of consecutive points, what are Bezier control points in terms of C's and D's?**

$$\mathbf{V}_0 = \mathbf{C}_0 \qquad \mathbf{W}_0 = \mathbf{C}_1$$

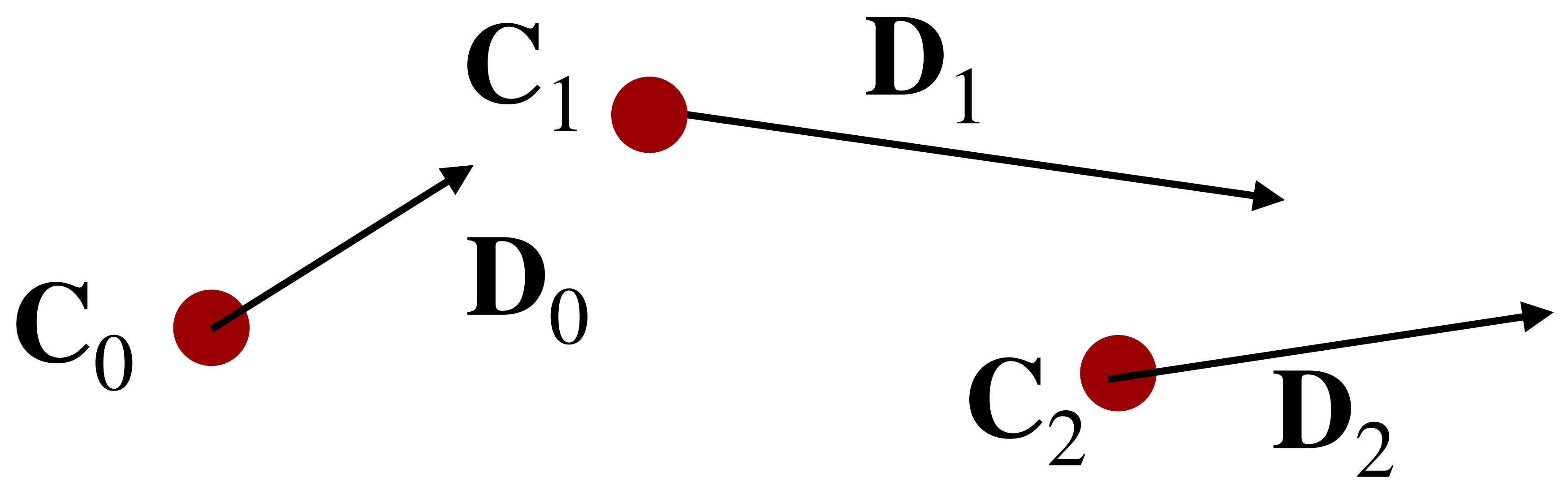
$$\mathbf{V}_1 = \mathbf{C}_0 + \frac{1}{3}\mathbf{D}_0 \qquad \mathbf{W}_1 = \mathbf{C}_1 + \frac{1}{3}\mathbf{D}_1$$

$$\mathbf{V}_2 = \mathbf{C}_1 - \frac{1}{3}\mathbf{D}_1 \qquad \mathbf{W}_2 = \mathbf{C}_2 - \frac{1}{3}\mathbf{D}_2$$

$$\mathbf{V}_3 = \mathbf{C}_1 \qquad \mathbf{W}_3 = \mathbf{C}_2$$

Recall C2 continuity constraint for a Bezier curve: $\mathbf{V}_2 - 2\mathbf{V}_3 + \mathbf{V}_4 = \mathbf{W}_1 - 2\mathbf{W}_2 + \mathbf{W}_3$

$$\mathbf{D}_0 + 4\mathbf{D}_1 + \mathbf{D}_2 = 3(\mathbf{C}_2 + \mathbf{C}_0)$$



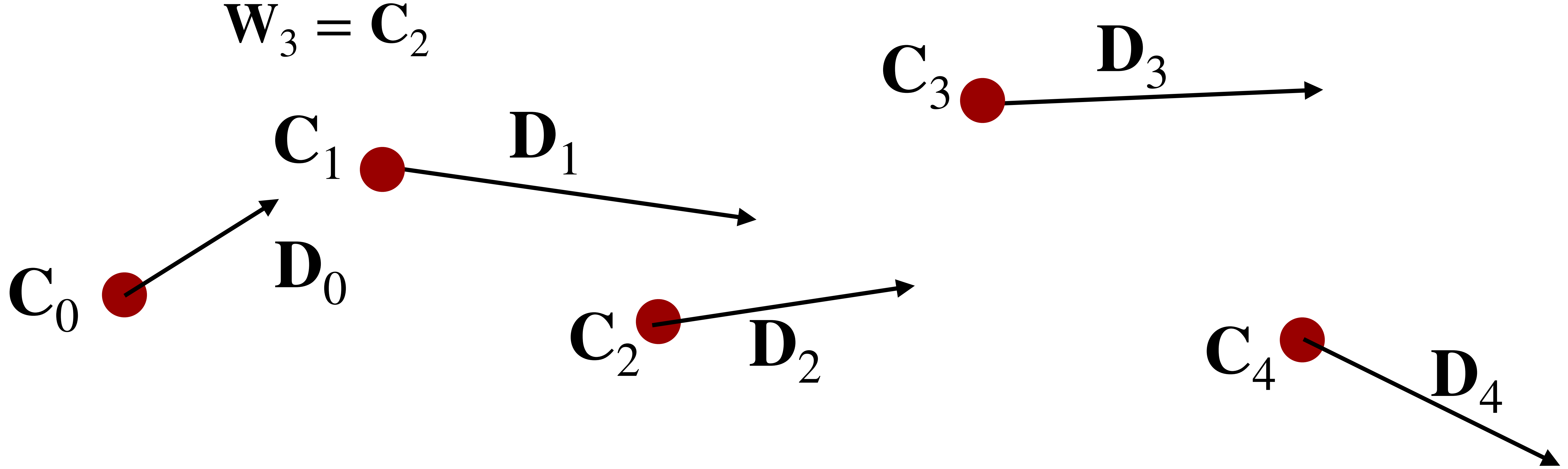
# $C^2$ interpolating splines

- If we want to create a Bezier curve between each pair of consecutive points, what are Bezier control points in terms of C's and D's?

$$\mathbf{V}_0 = \mathbf{C}_0$$
$$\mathbf{V}_1 = \mathbf{C}_0 + \frac{1}{3}\mathbf{D}_0$$
$$\mathbf{V}_2 = \mathbf{C}_1 - \frac{1}{3}\mathbf{D}_1$$
$$\mathbf{V}_3 = \mathbf{C}_1$$

$$\mathbf{W}_0 = \mathbf{C}_1$$
$$\mathbf{W}_1 = \mathbf{C}_1 + \frac{1}{3}\mathbf{D}_1$$
$$\mathbf{W}_2 = \mathbf{C}_2 - \frac{1}{3}\mathbf{D}_2$$
$$\mathbf{W}_3 = \mathbf{C}_2$$

$$\mathbf{D}_0 + 4\mathbf{D}_1 + \mathbf{D}_2 = 3(\mathbf{C}_2 - \mathbf{C}_0)$$
$$\mathbf{D}_1 + 4\mathbf{D}_2 + \mathbf{D}_3 = 3(\mathbf{C}_3 - \mathbf{C}_1)$$
$$\mathbf{D}_2 + 4\mathbf{D}_3 + \mathbf{D}_4 = 3(\mathbf{C}_4 - \mathbf{C}_2)$$



# $C^2$ interpolating splines

- **Given m keyframes, how many equations do we have?**

m-1

$$\mathbf{D}_0 + 4\mathbf{D}_1 + \mathbf{D}_2 = 3(\mathbf{C}_2 - \mathbf{C}_0)$$
$$\mathbf{D}_1 + 4\mathbf{D}_2 + \mathbf{D}_3 = 3(\mathbf{C}_3 - \mathbf{C}_1)$$
$$\mathbf{D}_2 + 4\mathbf{D}_3 + \mathbf{D}_4 = 3(\mathbf{C}_4 - \mathbf{C}_2)$$
$$\vdots$$
$$\mathbf{D}_{m-2} + 4\mathbf{D}_{m-1} + \mathbf{D}_m = 3(\mathbf{C}_m - \mathbf{C}_{m-2})$$

- How many variables are we trying to solve?

m+1

# Boundary conditions

- We can impose more conditions on the spline to solve the two extra degrees of freedom.
- Natural C2 interpolating splines require second derivative to be zero at the endpoints.

$$\mathbf{V}_2 - 2\mathbf{V}_3 + \mathbf{V}_4 = 0$$

$$\begin{aligned}
2\mathbf{D}_0 - \mathbf{D}_1 &= 3(\mathbf{C}_1 - \mathbf{C}_0)\\
\mathbf{D}_0 + 4\mathbf{D}_1 + \mathbf{D}_2 &= 3(\mathbf{C}_2 - \mathbf{C}_0)\\
\mathbf{D}_1 + 4\mathbf{D}_2 + \mathbf{D}_3 &= 3(\mathbf{C}_3 - \mathbf{C}_1)\\
\mathbf{D}_2 + 4\mathbf{D}_3 + \mathbf{D}_4 &= 3(\mathbf{C}_4 - \mathbf{C}_2)\\
&\vdots\\
\mathbf{D}_{m-2} + 4\mathbf{D}_{m-1} + \mathbf{D}_m &= 3(\mathbf{C}_m - \mathbf{C}_{m-2})\\
\mathbf{D}_{m-1} - 2\mathbf{D}_m &= 3(\mathbf{C}_m - \mathbf{C}_{m-1})
\end{aligned}$$

# Solve for the tangents

- Collect m+1 equations into a linear system.
- Use forward elimination to zero out every thing below the diagonal, then back substitute to compute D's

$$
\begin{array}{l}
\times -\frac{1}{2} \\ \hookrightarrow \\ \vdots \\ \\ \\ \\
\end{array}
\begin{bmatrix}
2 & 1 & & & & \\
1 & 4 & 1 & & & \\
 & 1 & 4 & 1 & & \\
 & & \ddots & & & \\
 & & & 1 & 4 & 1 \\
 & & & & 1 & 2
\end{bmatrix}
\begin{bmatrix}
\mathbf{D}_0 \\ \mathbf{D}_1 \\ \vdots \\ \vdots \\ \mathbf{D}_{m-1} \\ \mathbf{D}_m
\end{bmatrix}
=
\begin{bmatrix}
3(\mathbf{C}_1 - \mathbf{C}_0) \\ 3(\mathbf{C}_2 - \mathbf{C}_0) \\ \vdots \\ \vdots \\ 3(\mathbf{C}_m - \mathbf{C}_{m-2}) \\ 3(\mathbf{C}_{m+1} - \mathbf{C}_m)
\end{bmatrix}
$$

$$0 \quad 3\frac{1}{2} \quad 1 \qquad 3\mathbf{C}_2 - 1.5\mathbf{C}_0 - 3\mathbf{C}_1$$

$$\mathbf{D}_m = \frac{\mathbf{C}'_m}{a_{mm}}$$

$$\mathbf{D}_{m-1} = \frac{\mathbf{C}'_{m-1} - a_{m-1,m}\mathbf{D}_m}{a_{m-1,m-1}}$$

$$\ldots$$

$$
\begin{bmatrix}
2 & 1 & & & \\
0 & a_{22} & a_{23} & & \\
 & & \ddots & & \\
 & & 0 & a_{m-1,m-1} & a_{m-1,m} \\
 & & & 0 & a_{mm}
\end{bmatrix}
\begin{bmatrix}
\mathbf{D}_0 \\ \mathbf{D}_1 \\ \vdots \\ \mathbf{D}_{m-1} \\ \mathbf{D}_m
\end{bmatrix}
=
\begin{bmatrix}
3(\mathbf{C}_1 - \mathbf{C}_0) \\ \mathbf{C}'_1 \\ \vdots \\ \mathbf{C}'_{m-1} \\ \mathbf{C}'_m
\end{bmatrix}
$$

# Properties of B-Splines

- $C^2$ continuity ✓
- Local control ✗
- Interpolation ✓

# Additional reading

- **Bézier curve: https://en.wikipedia.org/wiki/Bézier_curve**
- **Pixar 3D Zoetrope: https://www.youtube.com/watch?v=5khDGKGv088**