Barycentric Coordinates

Xiao-Ming Fu

Outlines

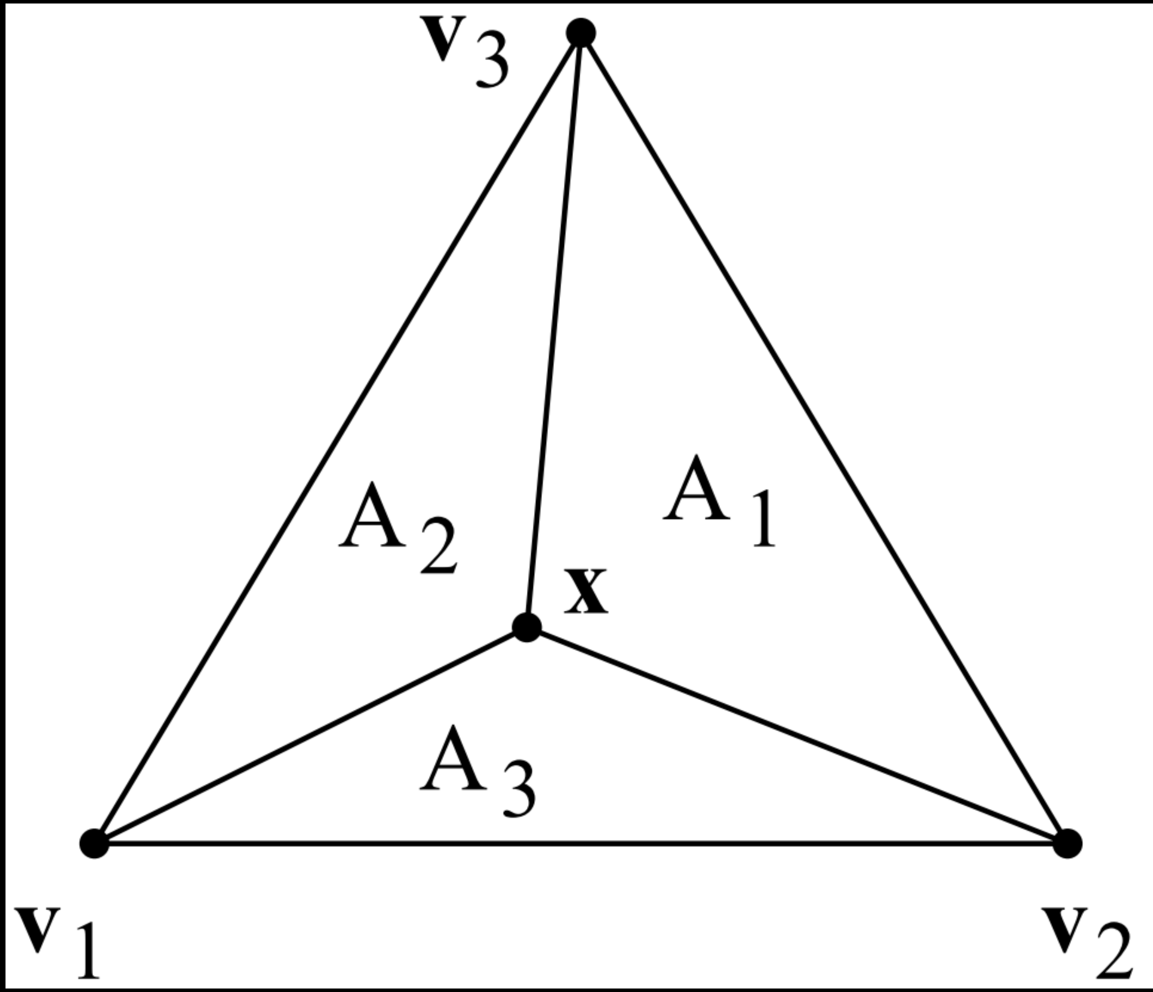
- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Outlines

- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Barycentric coordinates on triangles

$x = \phi_1 v_1 + \phi_2 v_2 + \phi_3 v_3$,
where $\phi_i = \frac{A_i}{A}$.



- Tetrahedron with four sub-tetrahedral.
- Any simplex.

Applications

- Function interpolation
- Function composite
- Defining Bernstein-Bézier polynomials over simplices
- …

Outlines

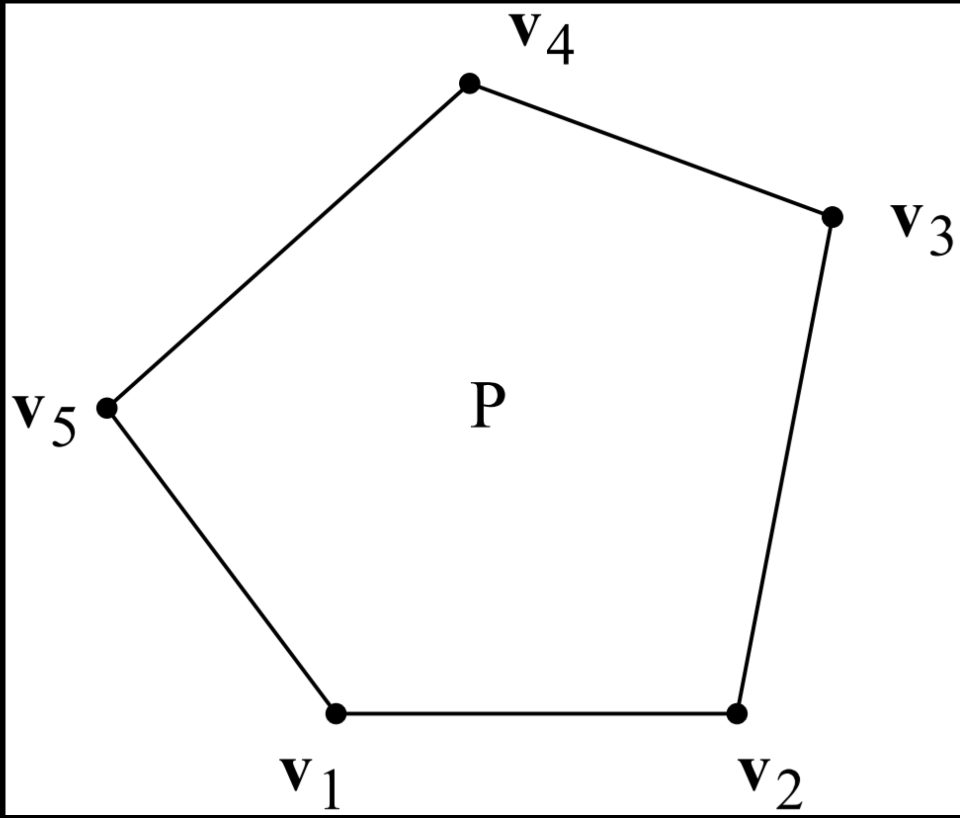
- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Generalized barycentric coordinates

- Let $P \subset R^2$ be a convex polygon, viewed as an open set, with vertices $v_1, v_2, \ldots, v_n, n \geq 3$, in some anticlockwise ordering.
- Any functions $\phi_i: P \to R, i = 1, \ldots, n$, will be called generalized barycentric coordinates if $\forall x \in P, \phi_i(x) \geq 0, i = 1, \ldots, n$, and

$$\sum_{i=1}^{n} \phi_i(x) = 1, \sum_{i=1}^{n} \phi_i(x) v_i = x$$

- ϕ_i: from any point in polygon P to R



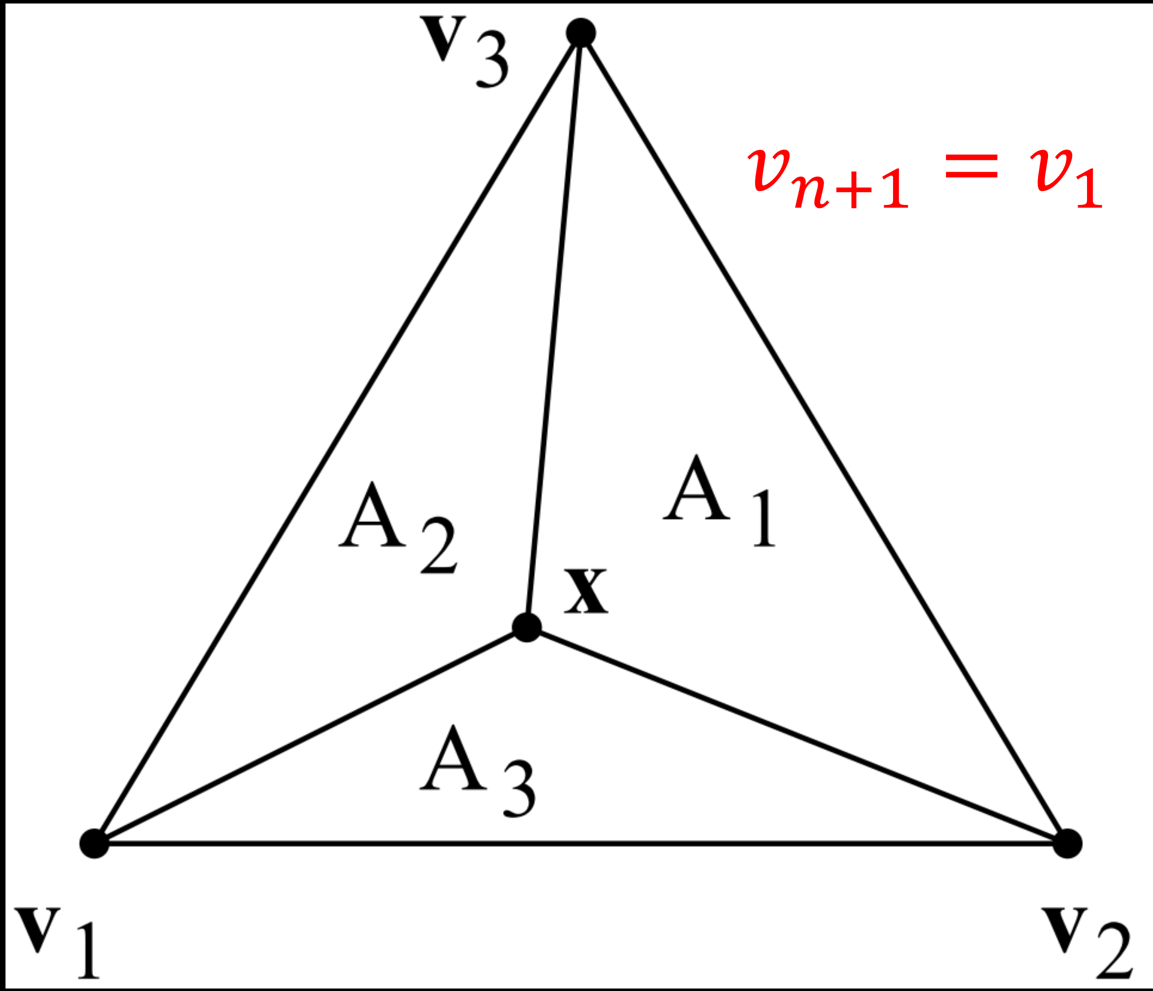
Triangular barycentric coordinates

$$\phi_i(x) = \frac{A(x, v_{i+1}, v_{i+2})}{A(v_1, v_2, v_3)}$$

Note: $A(p_1, p_2, p_3)$ is the signed area of the triangle with vertices $p_k = (x_k, y_k), k = 1, 2, 3,$

$$A(x_1, x_2, x_3) := \frac{1}{2}\begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix}$$

When $n \geq 4$, it is not unique.



$v_{n+1} = v_1$

Some basic properties/requirements

- The functions ϕ_i have a unique continuous extension to ∂P, the boundary of P.
- Lagrange property: $\phi_i(v_j) = \delta_{ij}$
- Piecewise linearity on ∂P
  - $\phi_i\left((1-\mu)v_j + \mu v_{j+1}\right) = (1-\mu)\phi_i(v_j) + \mu\phi_i(v_{j+1}), \mu \in [0,1]$.
- Interpolation
  - If $g(x) = \sum_{i=1}^{n} \phi_i(x) f(v_i)\, , x \in P$, then $g(v_i) = f(v_i)$. We call g a barycentric interpolant to f.
- Linear precision: if f is linear then $g = f$.

Some basic properties

- $l_i \leq \phi_i \leq L_i$ where $L_i, l_i: P \to R$ are the continuous, piecewise linear functions over the partitions of P satisfying $L_i(v_j) = l_i(v_j) = \delta_{ij}$. L_i is the least upper bound on ϕ_i and l_i the greatest lower bound.

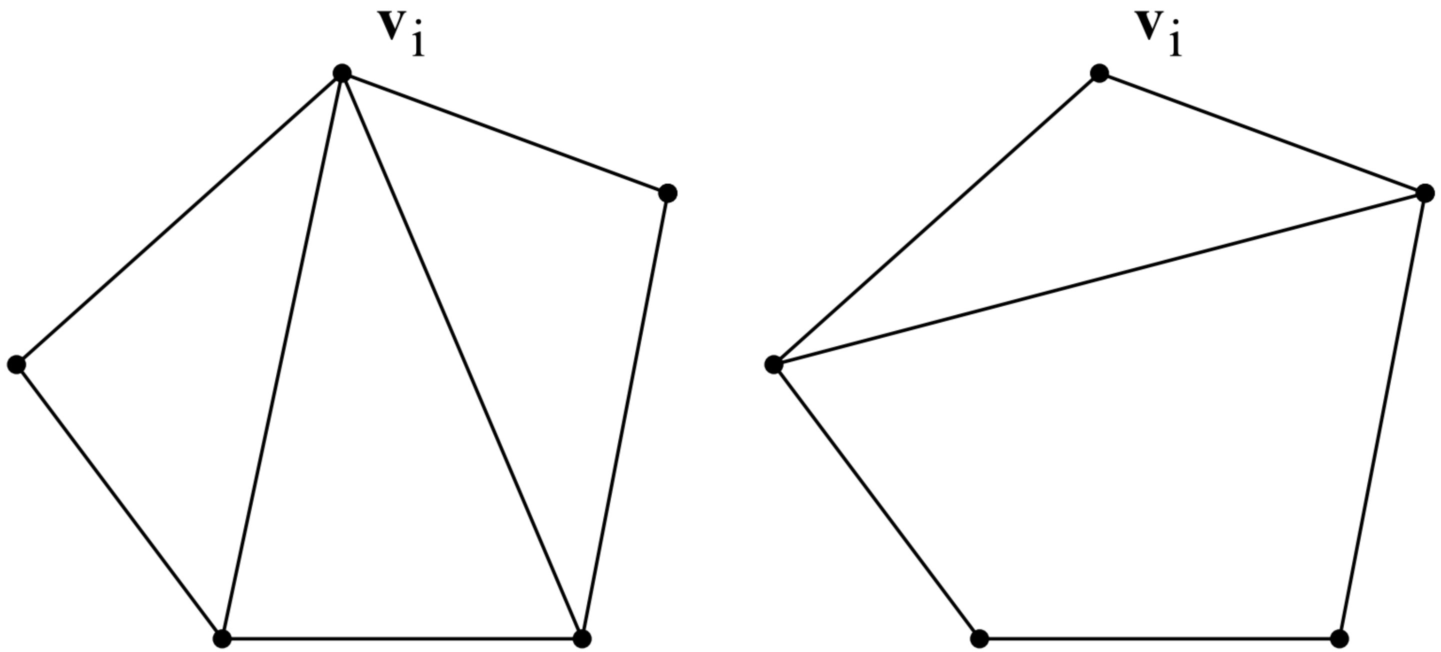


Figure 2.3. Partitions for L_i and ℓ_i.

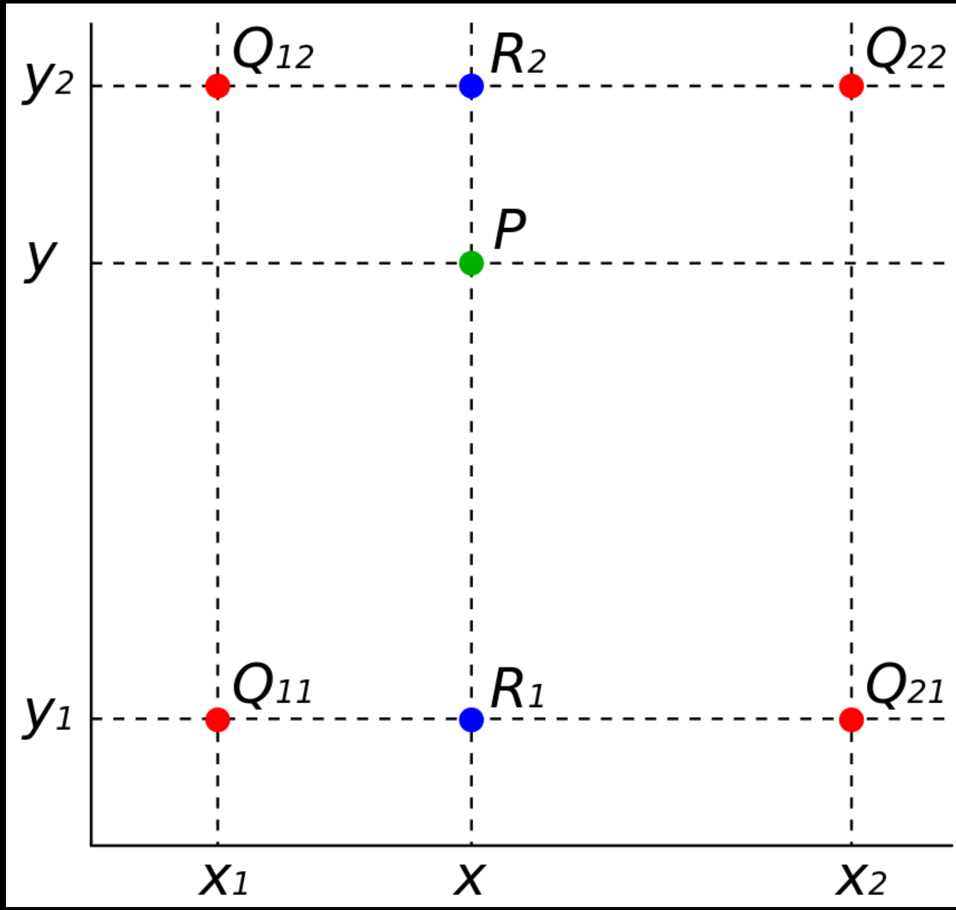
Outlines

- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Bilinear interpolation

https://en.wikipedia.org/wiki/Bilinear_interpolation

- Suppose that we want to find the value of the unknown function f at the point (x, y).
- It is assumed that we know the value of f at the four points $Q_{11} = (x_1, y_1), Q_{12} = (x_1, y_2), Q_{21} = (x_2, y_1)$, $Q_{22} = (x_2, y_2)$.
- Bilinear interpolation: The key idea is to perform linear interpolation first in one direction, and then again in the other direction.



Bilinear interpolation

https://en.wikipedia.org/wiki/Bilinear_interpolation

x-direction

$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21})$$

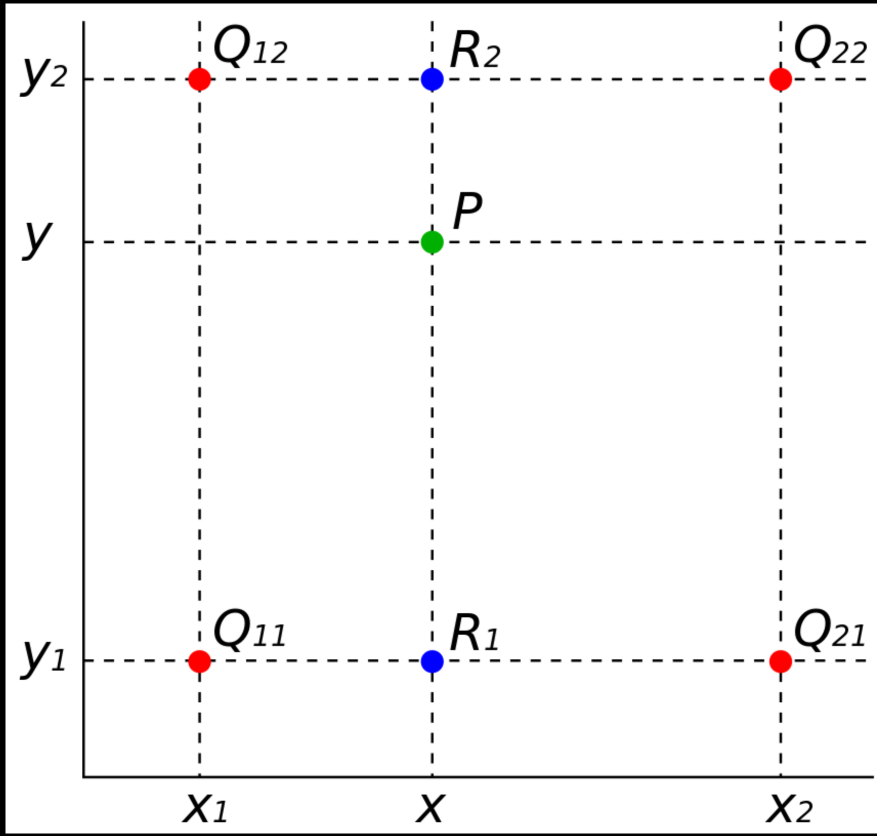
$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22})$$

y-direction

$$f(x, y) = \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2)$$

$$= \frac{(y_2 - y)(x_2 - x)}{(y_2 - y_1)(x_2 - x_1)} f(Q_{11}) + \frac{(y_2 - y)(x - x_1)}{(y_2 - y_1)(x_2 - x_1)} f(Q_{21})$$

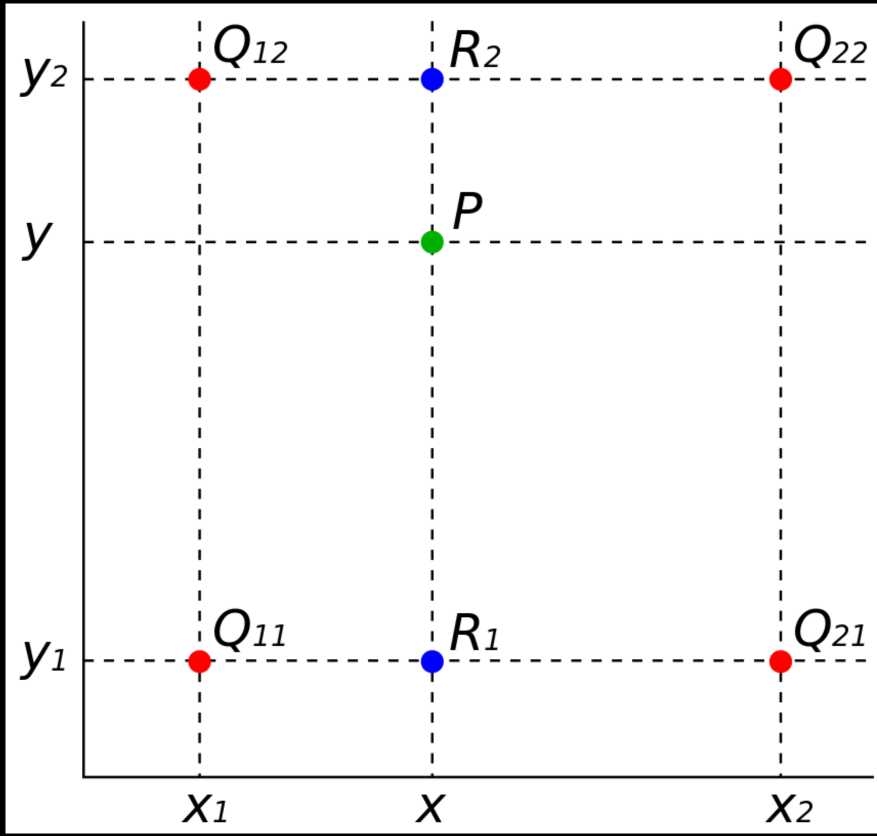
$$+ \frac{(y - y_1)(x_2 - x)}{(y_2 - y_1)(x_2 - x_1)} f(Q_{12}) + \frac{(y - y_1)(x - x_1)}{(y_2 - y_1)(x_2 - x_1)} f(Q_{22})$$



Unit square

- Suppose $x_1 = y_1 = 0, x_2 = y_2 = 1$

$$f(x, y) = (1 - x)(1 - y) \cdot f(0,0) + x(1 - y) \cdot f(1,0) + (1 - x)y \cdot f(0,1) + xy \cdot f(1,1)$$



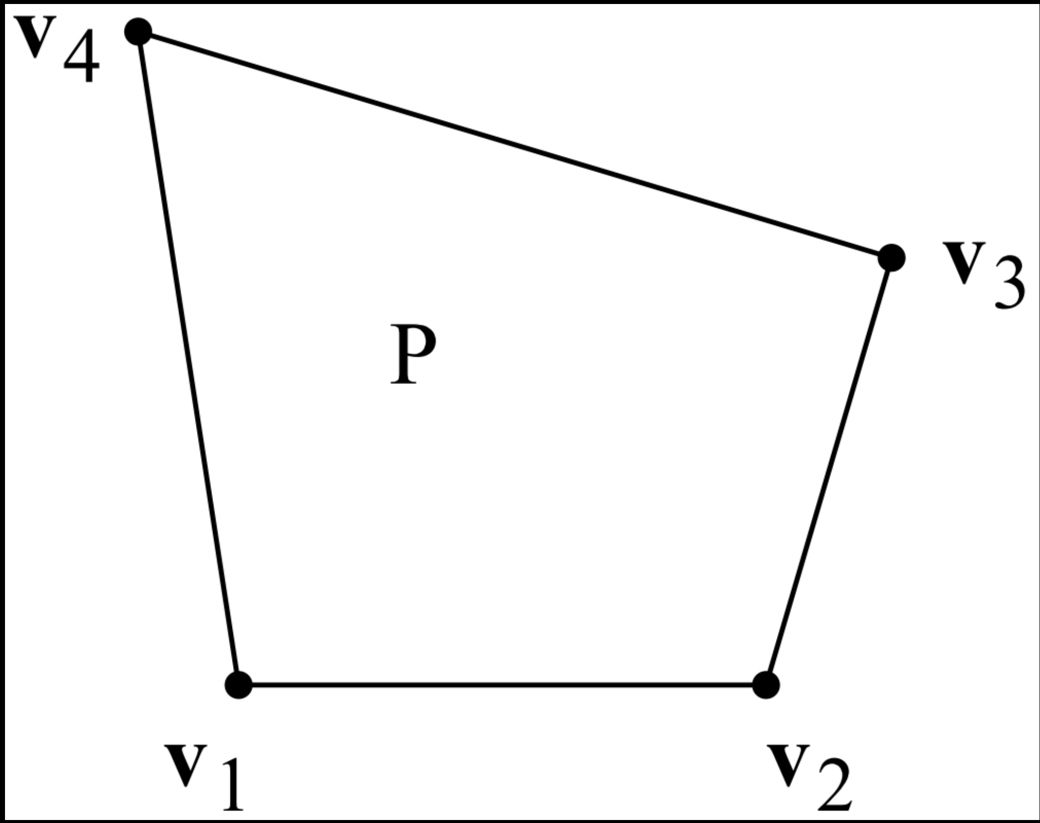
Convex quadrilaterals

- View P as the image of a bilinear map from the unit square $[0,1] \times [0,1]$.

- For each $x \in P$, there exist unique $\lambda, \mu \in (0, 1)$ such that

$(1-\lambda)(1-\mu)v_1 + \lambda(1-\mu)v_2 + \lambda\mu v_3 + (1-\lambda)\mu v_4 = x$ and so the four functions

$\phi_1(x) = (1-\lambda)(1-\mu)$, $\phi_2(x) = \lambda(1-\mu)$, $\phi_3(x) = \lambda\mu$, $\phi_4(x) = (1-\lambda)\mu$ are barycentric coordinates for x.



Inverse of the bilinear map

- $A_i(x) = A(x, v_i, v_{i+1}), B_i(x) = A(x, v_{i-1}, v_{i+1})$
- Theorem

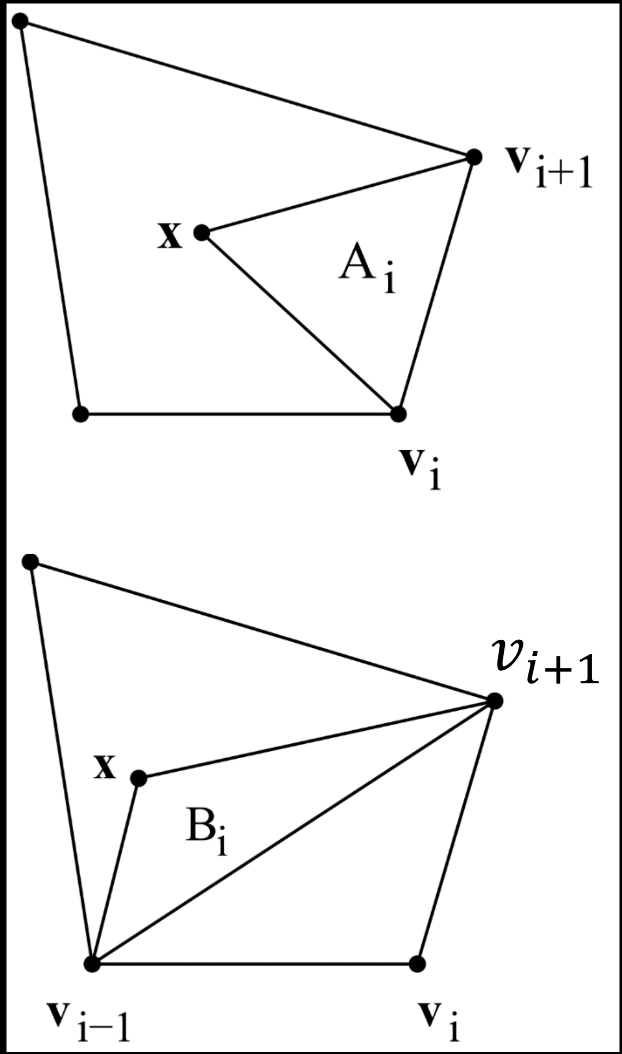
$$(\mu, 1-\lambda, 1-\mu, \lambda) = \left(\frac{2A_i}{E_i}\right)_{i=1,2,3,4}$$

where $E_i = 2A_i - B_i - B_{i+1} + \sqrt{D}$ and

$D = B_1^2 + B_2^2 + 2A_1A_3 + 2A_2A_4.$

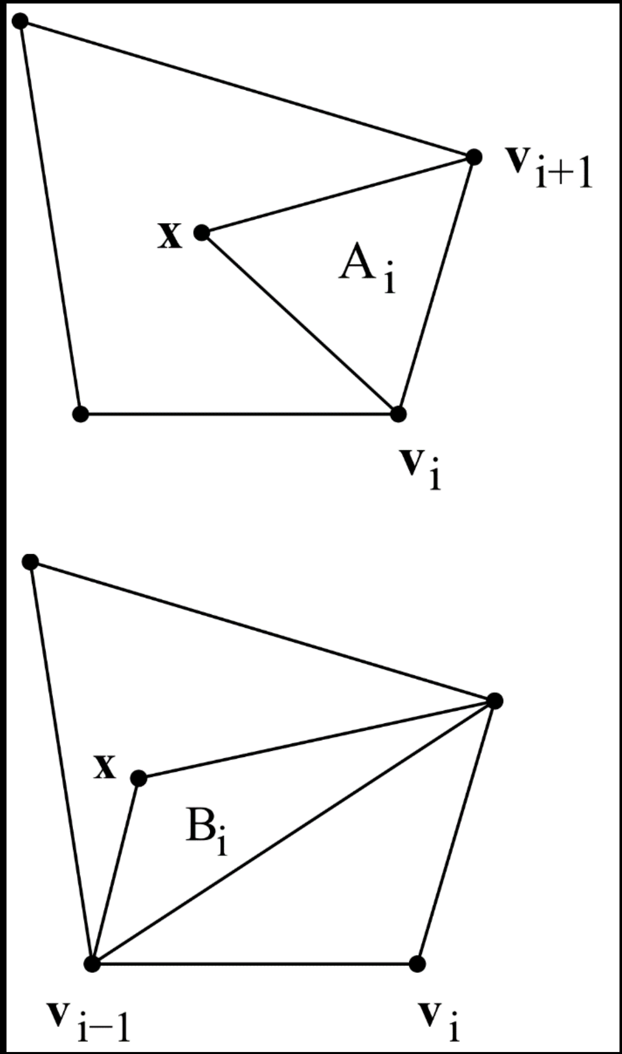
Therefore

$$\phi_i = \frac{4A_{i+1}A_{i+2}}{E_{i+1}E_{i+2}}$$



Inverse of the bilinear map

- Proof process, i.e., computational process:
- Known: $(1-\lambda)(1-\mu)v_1 + \lambda(1-\mu)v_2 + \lambda\mu v_3 + (1-\lambda)\mu v_4 = x$ and the convex quad P
- We want to solve λ, μ.



Proof. It is sufficient to show that

$$\mu = \frac{2A_1}{E_1} = \frac{2A_1}{2A_1 - B_1 - B_2 + \sqrt{D}}$$

as the derivation of the other three terms in (3.2) is similar. Defining the four vectors $\mathbf{d}_i = \mathbf{v}_i - \mathbf{x}$, $i = 1, 2, 3, 4$, (3.1) can be expressed as

$$(1-\lambda)(1-\mu)\mathbf{d}_1 + \lambda(1-\mu)\mathbf{d}_2 + \lambda\mu\mathbf{d}_3 + (1-\lambda)\mu\mathbf{d}_4 = 0.$$

Next, divide the equation by $\lambda\mu$, and defining

$$\alpha := \frac{1-\lambda}{\lambda}, \qquad \beta := \frac{1-\mu}{\mu},$$

the equation becomes

$$\alpha\beta\mathbf{d}_1 + \beta\mathbf{d}_2 + \mathbf{d}_3 + \alpha\mathbf{d}_4 = 0.$$

By writing this as

$$\alpha(\beta\mathbf{d}_1 + \mathbf{d}_4) + (\beta\mathbf{d}_2 + \mathbf{d}_3) = 0,$$

and taking the cross product of it with $\beta\mathbf{d}_1 + \mathbf{d}_4$ eliminates α:

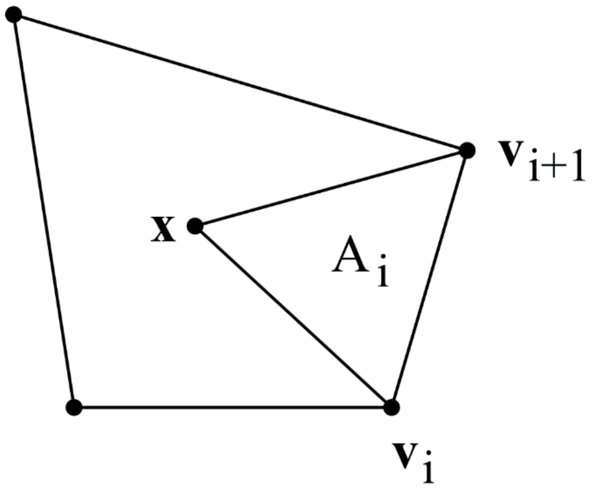
$$(\beta\mathbf{d}_1 + \mathbf{d}_4) \times (\beta\mathbf{d}_2 + \mathbf{d}_3) = 0.$$

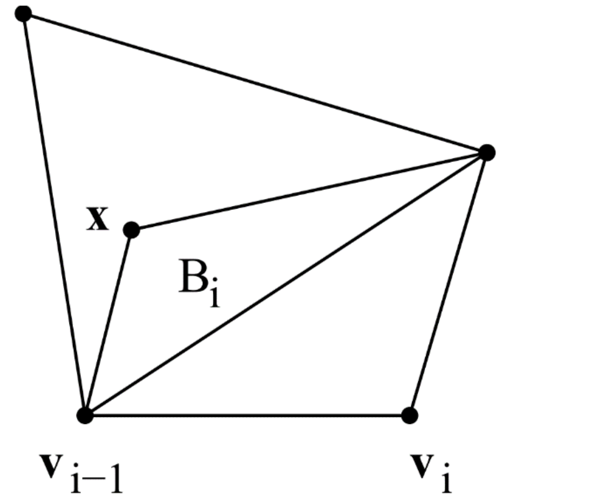
(Here, $\mathbf{a} \times \mathbf{b} = (a_1, a_2) \times (b_1, b_2) := a_1 b_2 - a_2 b_1$.) This is a quadratic equation in β, which, in terms of the A_i and B_i, is

$$A_1\beta^2 + (B_1 + B_2)\beta - A_3 = 0.$$

The discriminant is

$$D = (B_1 + B_2)^2 + 4A_1A_3 > 0,$$





and so the equation has real roots, and β is the positive one,

$$\beta = \frac{-B_1 - B_2 + \sqrt{D}}{2A_1}.$$

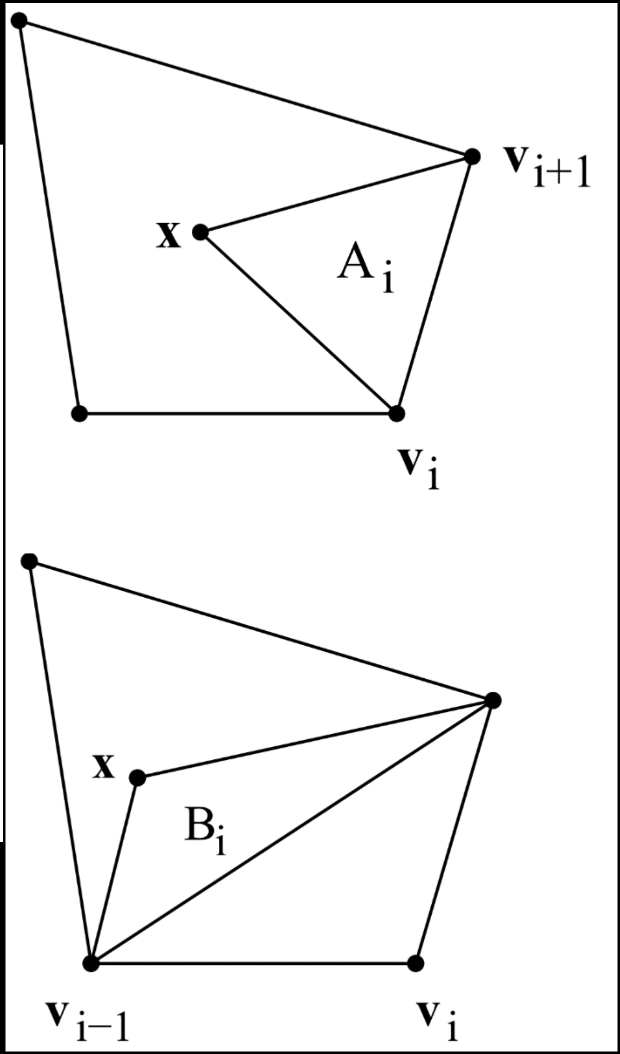
Next observe that

$$B_1 B_2 = A_2 A_4 - A_1 A_3,$$

which follows from taking the cross product of $\mathbf{d}_4$ with the well known equation

$$(\mathbf{d}_1 \times \mathbf{d}_2)\mathbf{d}_3 + (\mathbf{d}_2 \times \mathbf{d}_3)\mathbf{d}_1 + (\mathbf{d}_3 \times \mathbf{d}_1)\mathbf{d}_2 = 0.$$

From this we find that D can be expressed as (3.3). From β we now obtain $\mu = 1/(1+\beta)$. $\square$



Outlines

- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

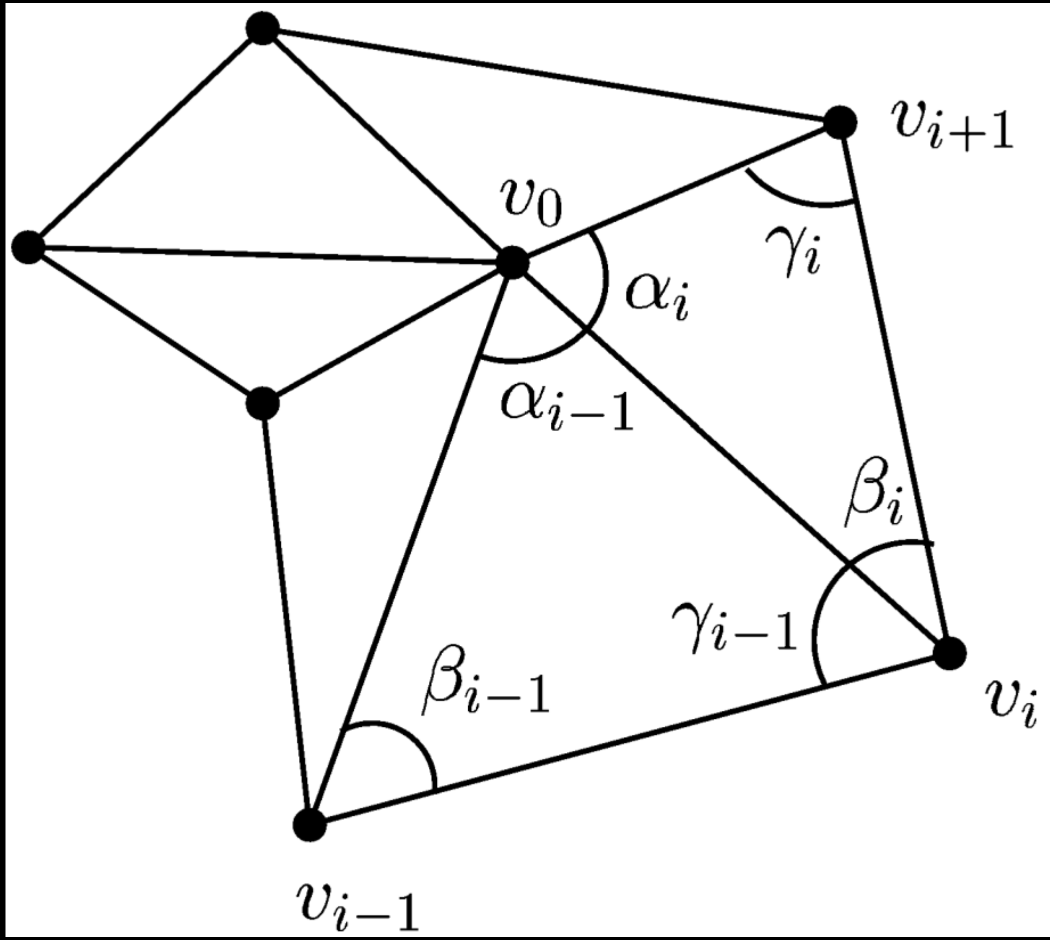
Mean value coordinates (MVC)

The weights

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

are coordinates for v_0 with respect to $v_1, \ldots, v_n$.



Three requirements

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$
$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

Since $0 < \alpha_i < \pi$, $\phi_i(x) \geq 0$

$\sum_{i=1}^{n} \phi_i(x) = 1$, by definition

$$\sum_{i=1}^{n} \phi_i(x) v_i = v_0 \Leftrightarrow \sum_{i=1}^{n} \phi_i(x)(v_i - v_0) = 0$$

Three requirements

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

- Proof.

$$v_i = v_0 + r_i(\cos\theta_i, \sin\theta_i)$$

Then we have

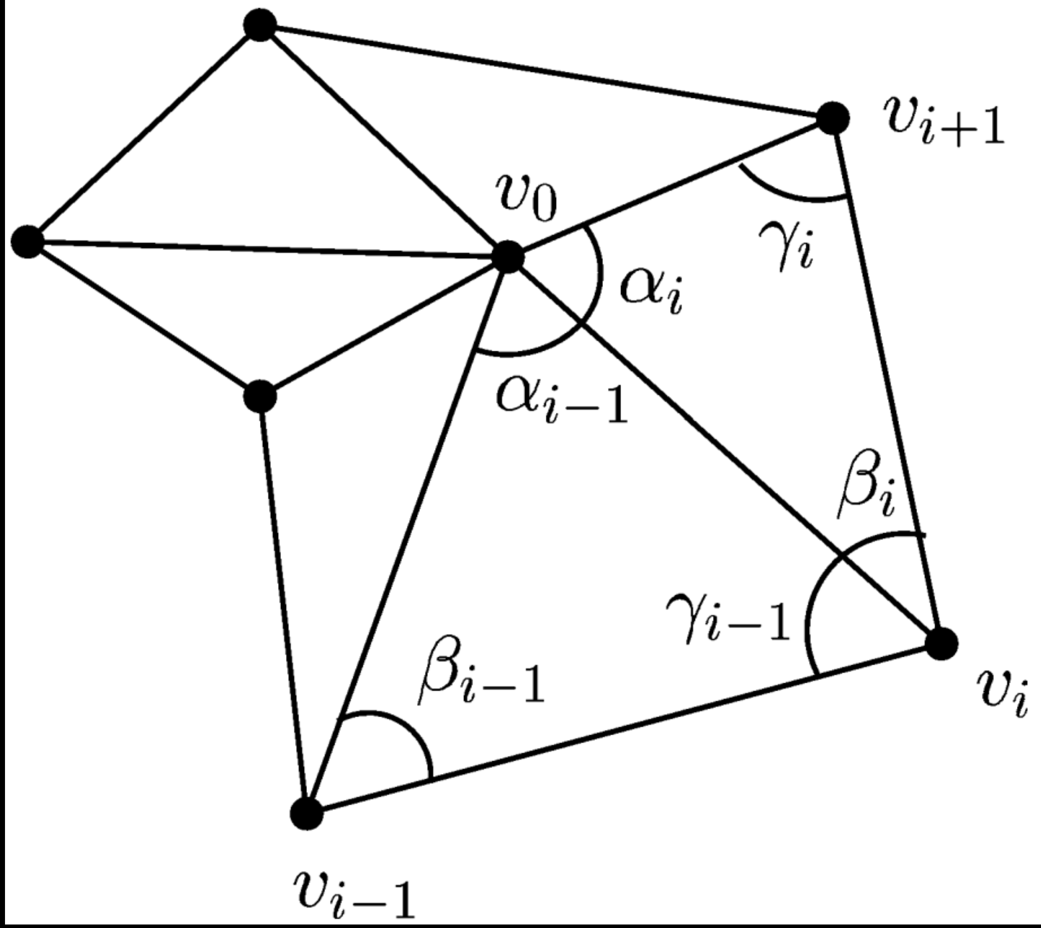
$$\frac{v_i - v_0}{\|v_i - v_0\|} = (\cos\theta_i, \sin\theta_i)$$

$$\alpha_i = \theta_{i+1} - \theta_i$$

Then,

$$\sum_{i=1}^{n} \phi_i(x)(v_i - v_0)$$

$$= \sum_{i=1}^{n} \left(\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)\right)(\cos\theta_i, \sin\theta_i)$$



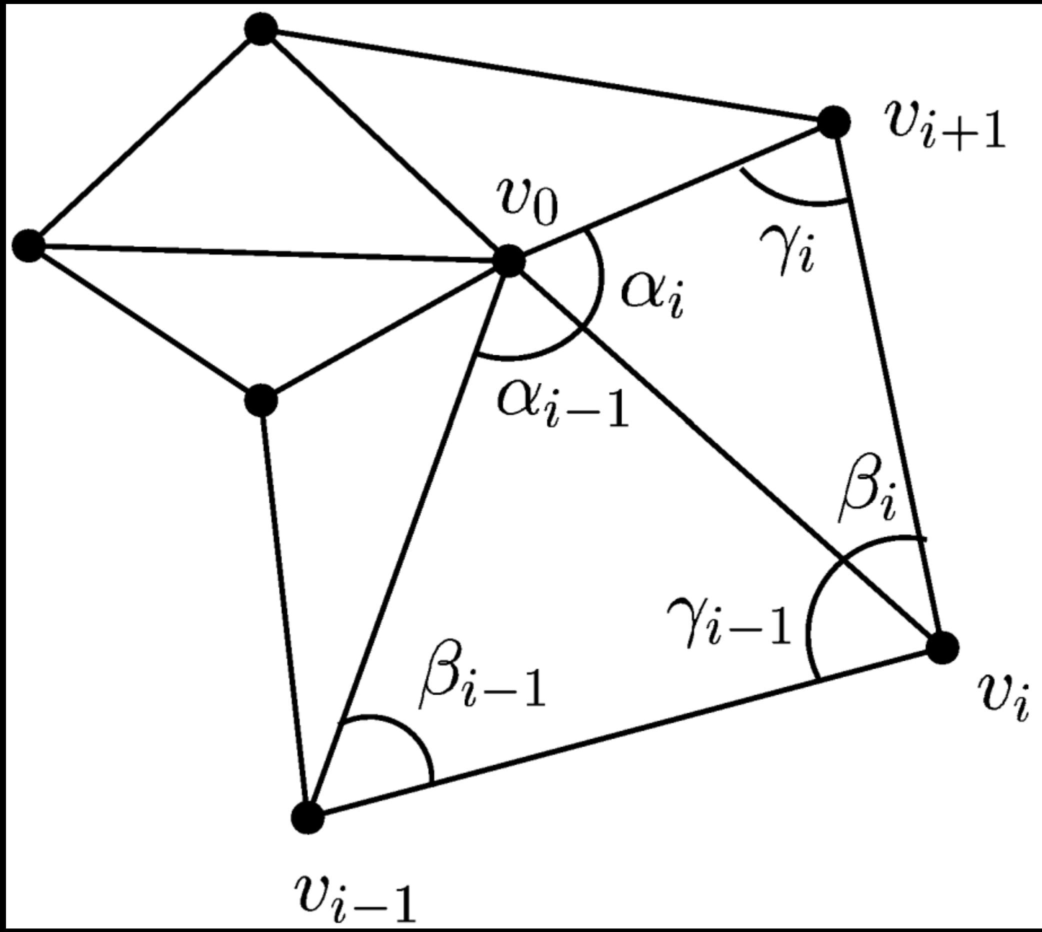
Proof

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

$$\sum_{i=1}^{n} \left(\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right) \right) (\cos\theta_i , \sin\theta_i)$$

$$= \sum_{i=1}^{n} \tan\left(\frac{\alpha_i}{2}\right) ((\cos\theta_i , \sin\theta_i)$$



Proof

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

Since

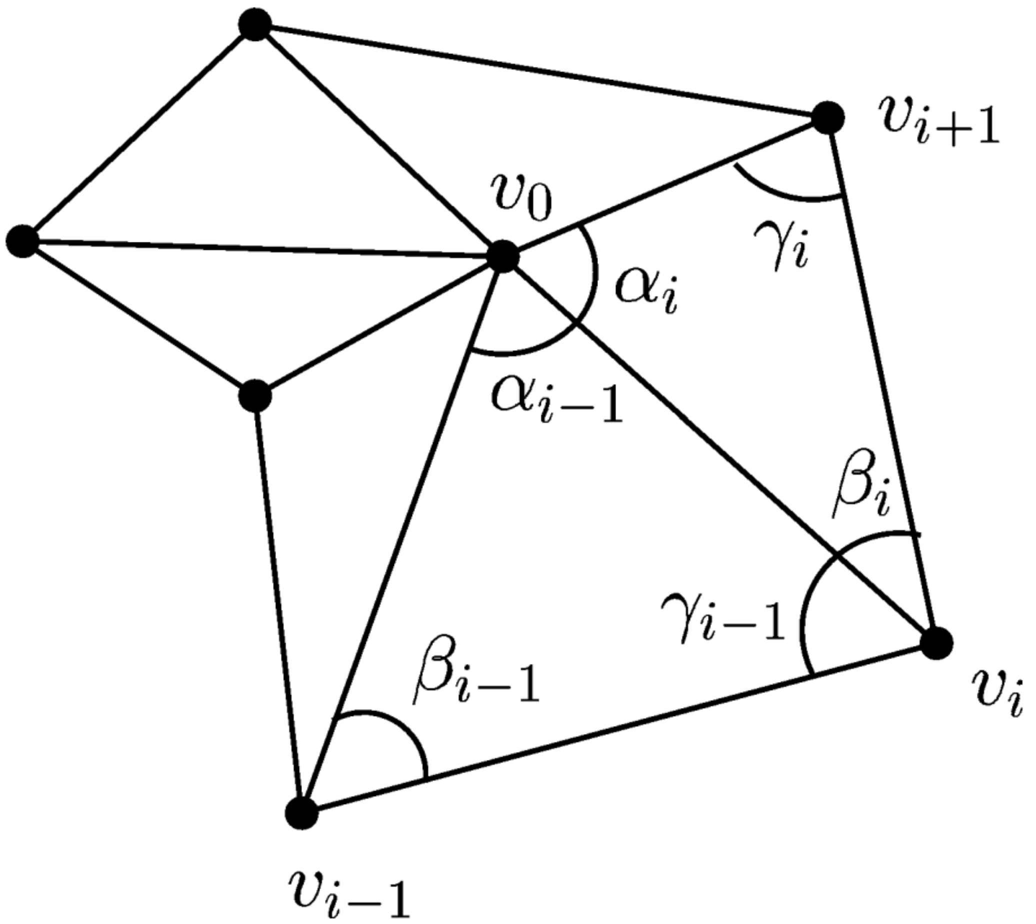
$$\cos\theta = \frac{\cos\theta \sin(\theta_{i+1} - \theta_i)}{\sin(\theta_{i+1} - \theta_i)}$$

$$= \frac{\cos\theta \sin(\theta_{i+1}) \cos\theta_i - \cos\theta \sin(\theta_i) \cos\theta_{i+1}}{\sin(\alpha_i)}$$

$$= \frac{\cos\theta \sin(\theta_{i+1}) \cos\theta_i - \sin\theta \cos\theta_i \cos\theta_{i+1}}{\sin(\alpha_i)}$$

$$+ \frac{\sin\theta \cos\theta_i \cos\theta_{i+1} - \cos\theta \sin(\theta_i) \cos\theta_{i+1}}{\sin(\alpha_i)}$$

$$= \frac{\sin(\theta_{i+1} - \theta) \cos\theta_i}{\sin(\alpha_i)} + \frac{\sin(\theta - \theta_i) \cos\theta_{i+1}}{\sin(\alpha_i)}$$



Proof

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

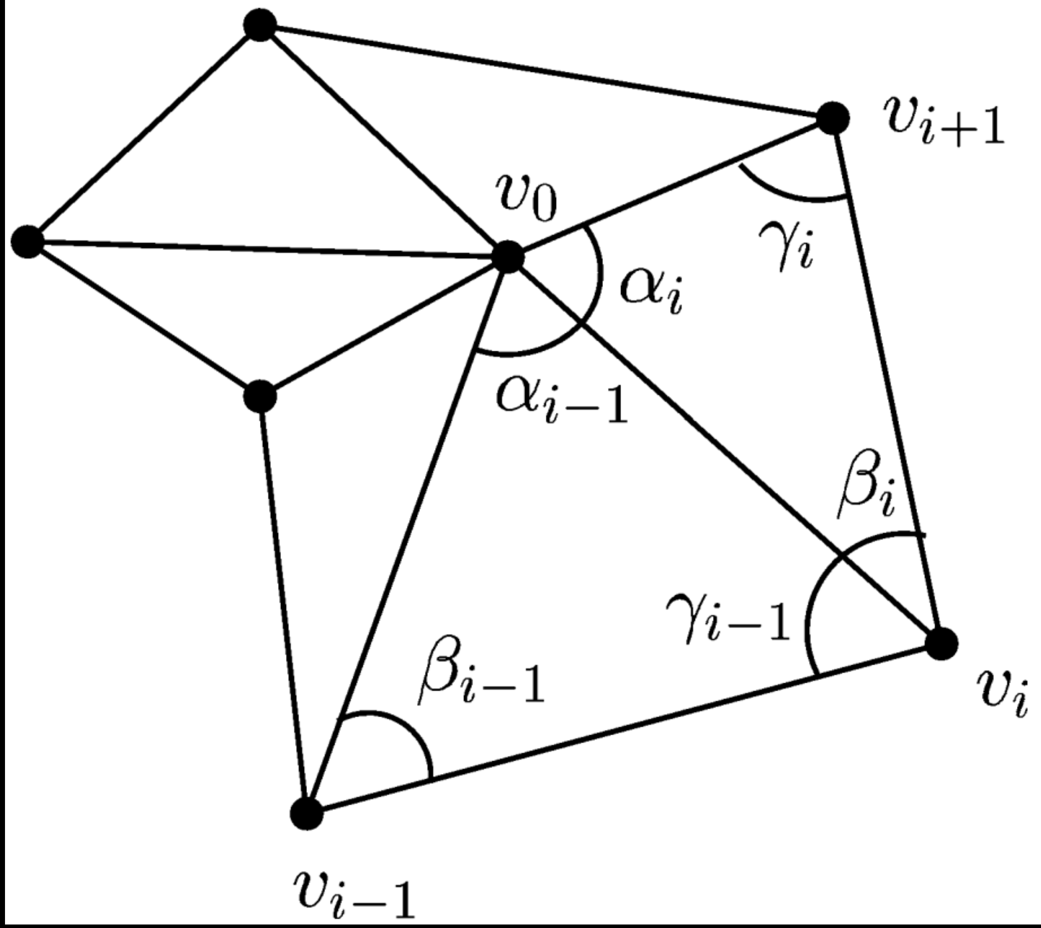
Similarly,

$$\sin\theta = \frac{\sin(\theta_{i+1} - \theta)\sin\theta_i}{\sin(\alpha_i)} + \frac{\sin(\theta - \theta_i)\sin\theta_{i+1}}{\sin(\alpha_i)}$$

As we know

$$0 = \int_0^{\pi} (\cos\theta, \sin\theta) d\theta = \sum_{i=1}^{n} \int_{\theta_i}^{\theta_{i+1}} (\cos\theta, \sin\theta) d\theta$$

$$= \sum_{i=1}^{n} \int_{\theta_i}^{\theta_{i+1}} \frac{\sin(\theta_{i+1} - \theta)}{\sin(\alpha_i)} (\cos\theta_i, \sin\theta_i) + \frac{\sin(\theta - \theta_i)}{\sin(\alpha_i)} (\cos\theta_{i+1}, \sin\theta_{i+1}) d\theta$$



Proof

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

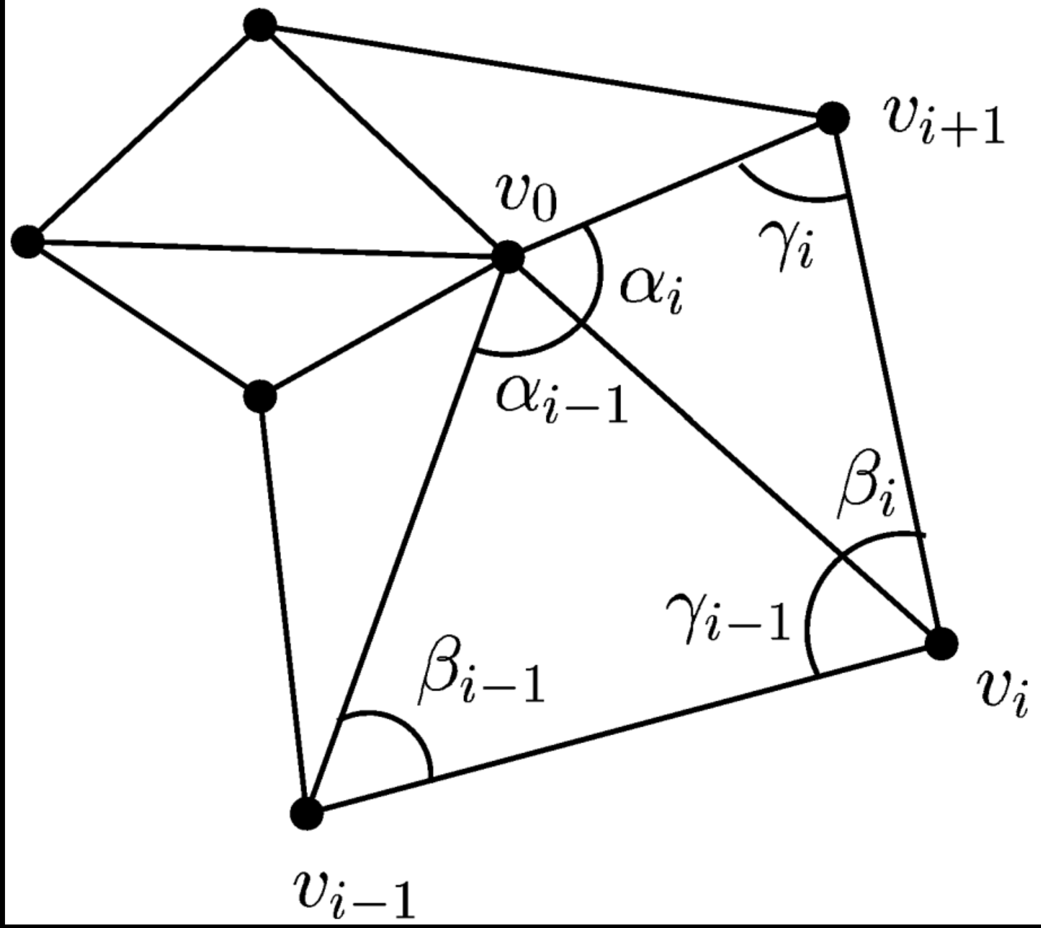
Since

$$\int_{\theta_i}^{\theta_{i+1}} \frac{\sin(\theta_{i+1} - \theta)}{\sin(\alpha_i)} d\theta = \int_{\theta_i}^{\theta_{i+1}} \frac{\sin(\theta - \theta_i)}{\sin(\alpha_i)} d\theta$$

$$= \frac{1 - \cos\alpha_i}{\sin\alpha_i} = \tan\frac{\alpha_i}{2}$$

Thus

$$\sum_{i=1}^{n} \tan\left(\frac{\alpha_i}{2}\right)((\cos\theta_i, \sin\theta_i)$$



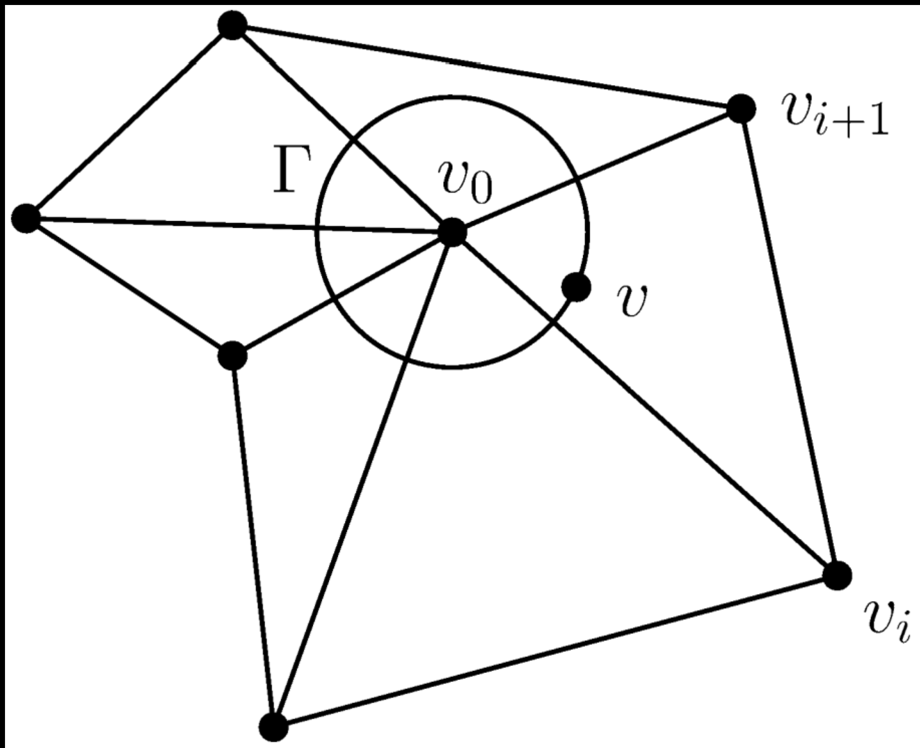
Motivation of MVC

- The motivation behind the coordinates was an attempt to approximate harmonic maps by piecewise linear maps over triangulations, in such a way that injectivity is preserved.
  - $u_{xx} + u_{yy} = 0$

Motivation of MVC

- Suppose we want to approximate the solution u with respect to Dirichlet boundary conditions, $u_{|\partial\Omega} = u_0$, by a piecewise linear function u_T over some triangulation T of Ω.
    - $\int_\Omega |\nabla \mathrm{u}_T|^2\, dx$
    - boundary conditions
    - a sparse linear system
    - $u_T(v_0) = \sum_{i=1}^{n} \phi_i\, u_T(v_i)$
    - Tutte's embedding
- Mean value theorem:

$$u(v_0) = \frac{1}{2\pi r}\int_\Gamma u(v)ds$$



Motivation of MVC

$$\phi_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j},$$

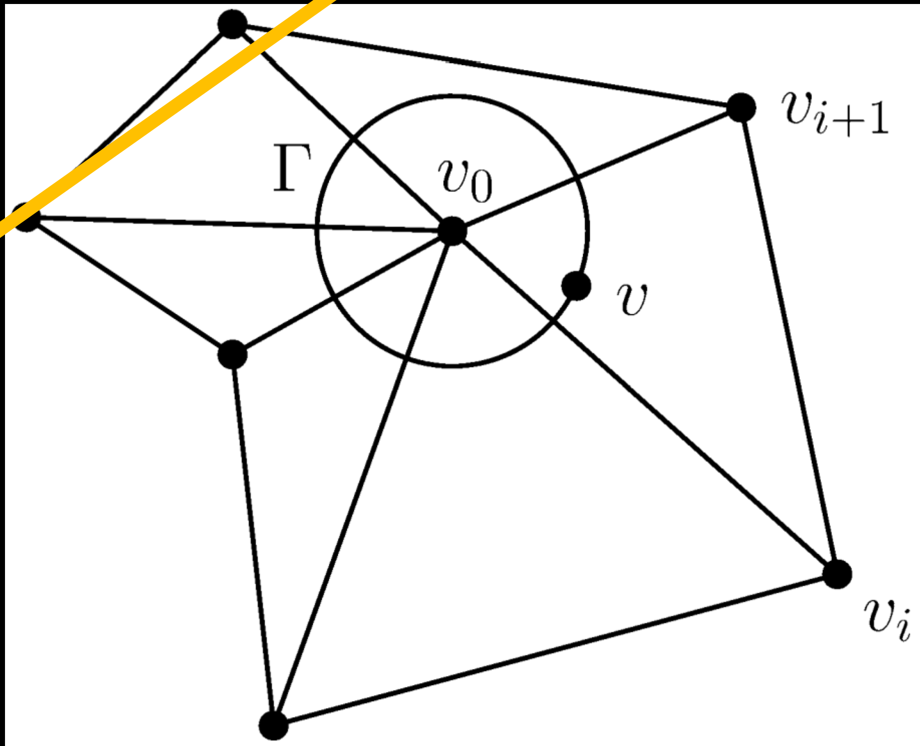
$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

- Thus, we want to find

$$u_T(v_0) = \frac{1}{2\pi r} \int_{\Gamma} u_T(v) ds$$

for r sufficiently small that the disc $B(v_0, r)$ is entirely contained in the union of the triangles containing v_0.

If above condition is satisfied $\rightarrow u_T(v_0) = \sum_{i=1}^{n} \phi_i \, u_T(v_i)$ where ϕ_i is, independent of the choice of r.



Motivation of MVC

- Lemma: if $f: T_i \to R$ is any linear function then

$$\int_{\Gamma_i} f(v)ds = r\alpha_i f(v_0) + r^2 \tan(\frac{\alpha_i}{2})\left(\frac{f(v_i) - f(v_0)}{||v_i - v_0||} + \frac{f(v_{i+1}) - f(v_0)}{||v_{i+1} - v_0||}\right)$$

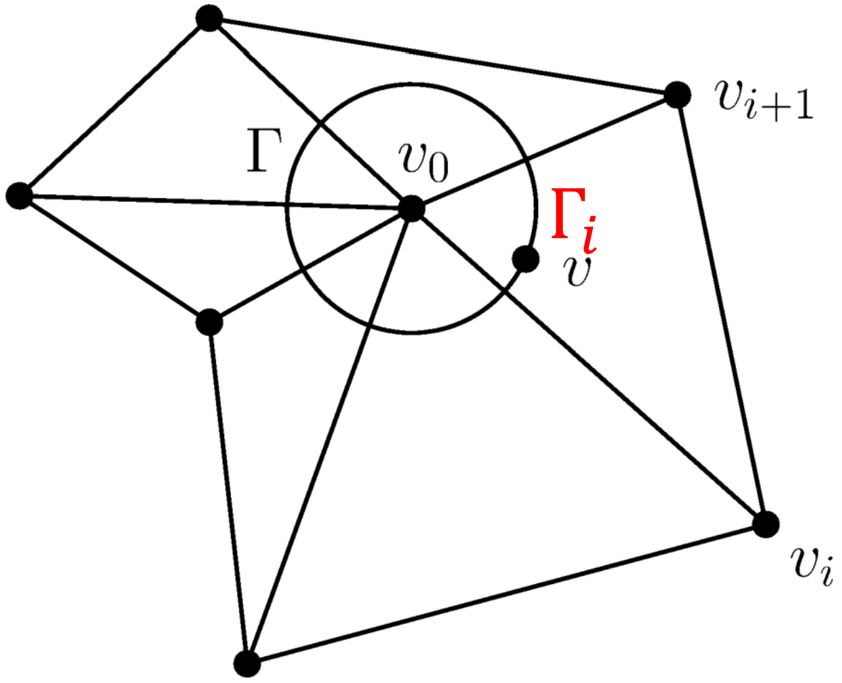
- Proof: $\forall v \in \Gamma_i, v = v_0 + r(\cos\theta, \sin\theta)$

and $v_j = v_0 + r_j(\cos\theta_j, \sin\theta_j)$.

Then, $\int_{\Gamma_i} f(v)ds = r\int_{\theta_i}^{\theta_{i+1}} f(v)d\theta$

Since f is linear, and using barycentric coordinates

we have: $f(v) = f(v_0) + \lambda_1\big(f(v_i) - f(v_0)\big)$
$+\lambda_2\big(f(v_{i+1}) - f(v_0)\big)$

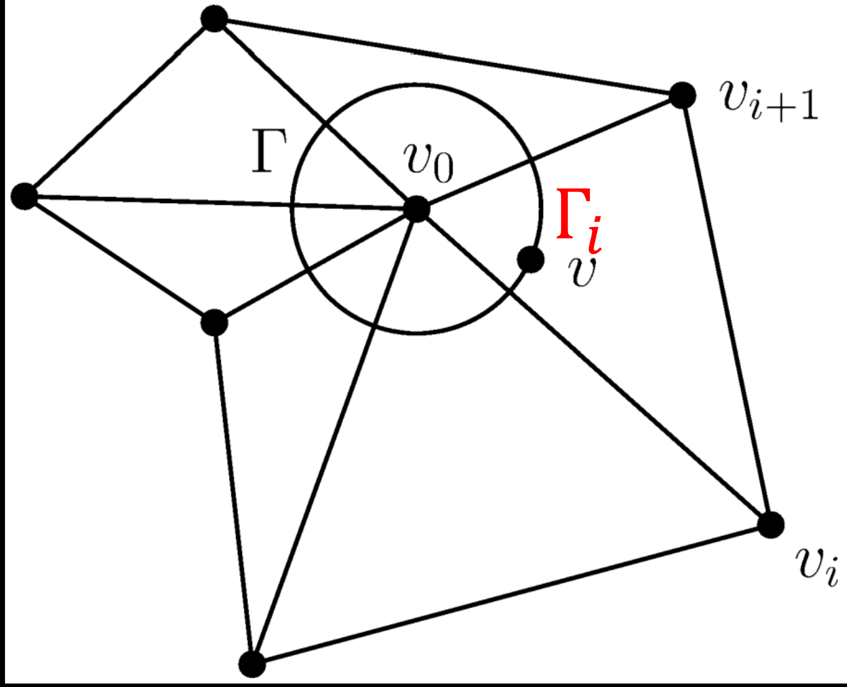


Motivation of MVC

- $\lambda_1 = \frac{A_1}{A}, \lambda_2 = \frac{A_2}{A}$
- $A_1: \Delta v_0 v v_{i+1}, A_2: \Delta v_0 v_i v$
- $A = \frac{1}{2} r_i r_{i+1} \sin \alpha_i, A_1 = \frac{1}{2} r r_{i+1} \sin(\theta_{i+1} - \theta),$
  $A_2 = \frac{1}{2} r_i r \sin(\theta - \theta_i)$
- $\lambda_1 = \frac{r \sin(\theta_{i+1} - \theta)}{r_i \sin \alpha_i}, \lambda_2 = \frac{r \sin(\theta - \theta_i)}{r_{i+1} \sin \alpha_i}$

$$\int_{\Gamma_i} f(v) ds = r \int_{\theta_i}^{\theta_{i+1}} f(v) d\theta$$

$$= r \int_{\theta_i}^{\theta_{i+1}} f(v_0) + \lambda_1 \big(f(v_i) - f(v_0)\big) + \lambda_2 \big(f(v_{i+1})$$



Motivation of MVC

- Proposition: Suppose the piecewise linear function $u_T : \Omega \to R$ satisfies the local mean value theorem, i.e., for each interior vertex v_0, it satisfies $u_T(v_0) = \frac{1}{2\pi r}\int_\Gamma u_T(v)ds$ for some $r > 0$ suitably small. Then $u_T(v_0)$ is given by the convex combination in $u_T(v_0) = \sum_{i=1}^n \phi_i u_T(v_i)$ with the weights ϕ_i is

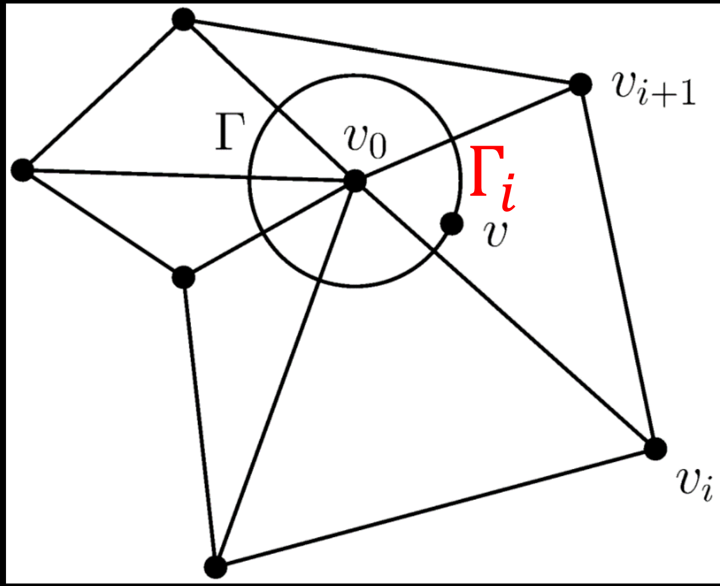
$$\phi_i = \frac{\omega_i}{\sum_{j=1}^n \omega_j},$$

$$\omega_i = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{\|v_i - v_0\|}$$

Proof of Proposition

$$u_T(v_0) = \frac{1}{2\pi r}\int_\Gamma u_T(v)ds = \frac{1}{2\pi r}\sum_{i=1}^{n}\int_{\Gamma_i} u_T(v)ds$$

$$u_T(v_0)$$
$$= \frac{1}{2\pi r}\sum_{i=1}^{n}\Big(r\alpha_i u_T(v_0)$$

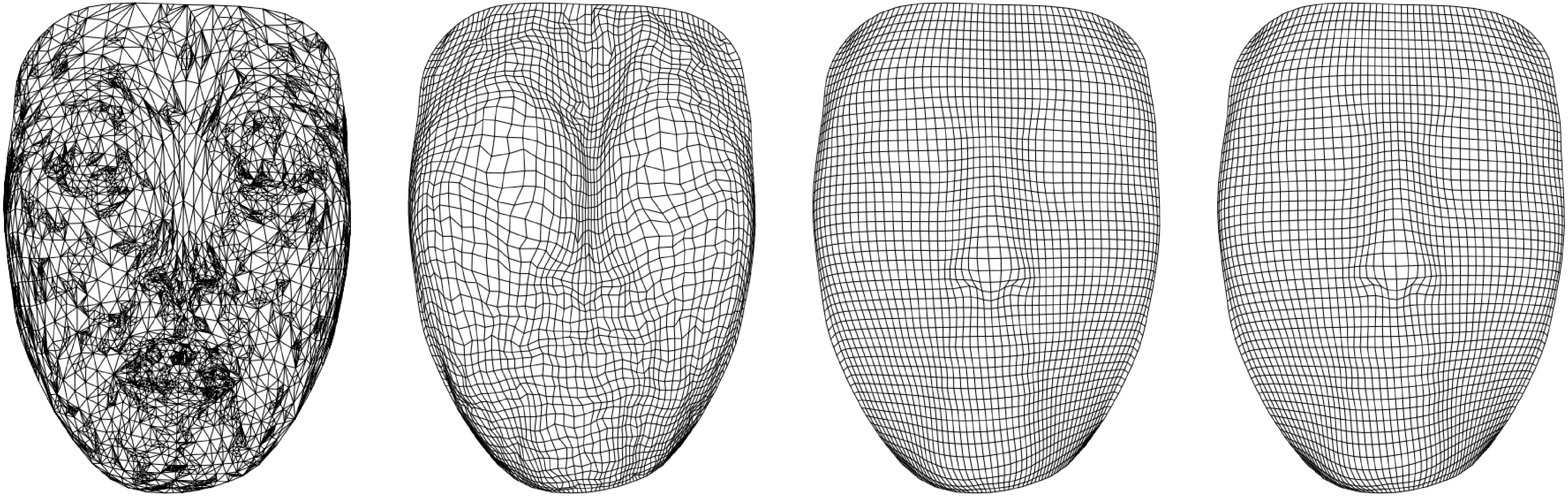


Applications of MVC

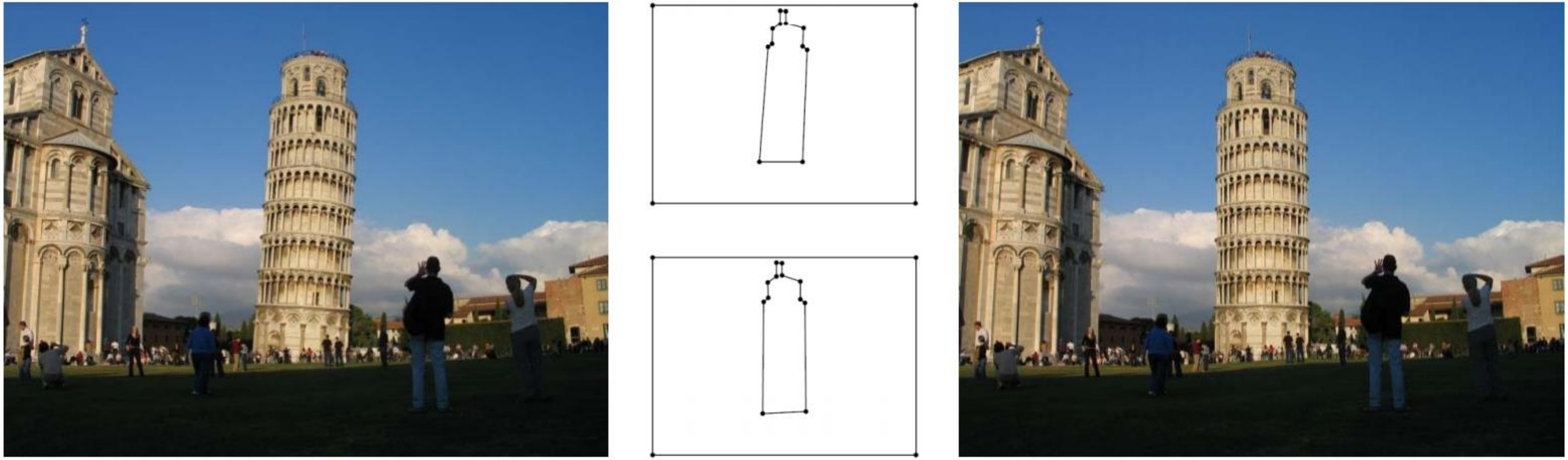
- Parameterization
  - Mean Value Coordinate
- Deformation
  - Mean Value Coordinates for Closed Triangular Meshes
- Poisson image editing
  - Coordinates for Instant Image Cloning
- Diffusion curves/surfaces
  - Volumetric Modeling with Diffusion Surfaces

Parameterization

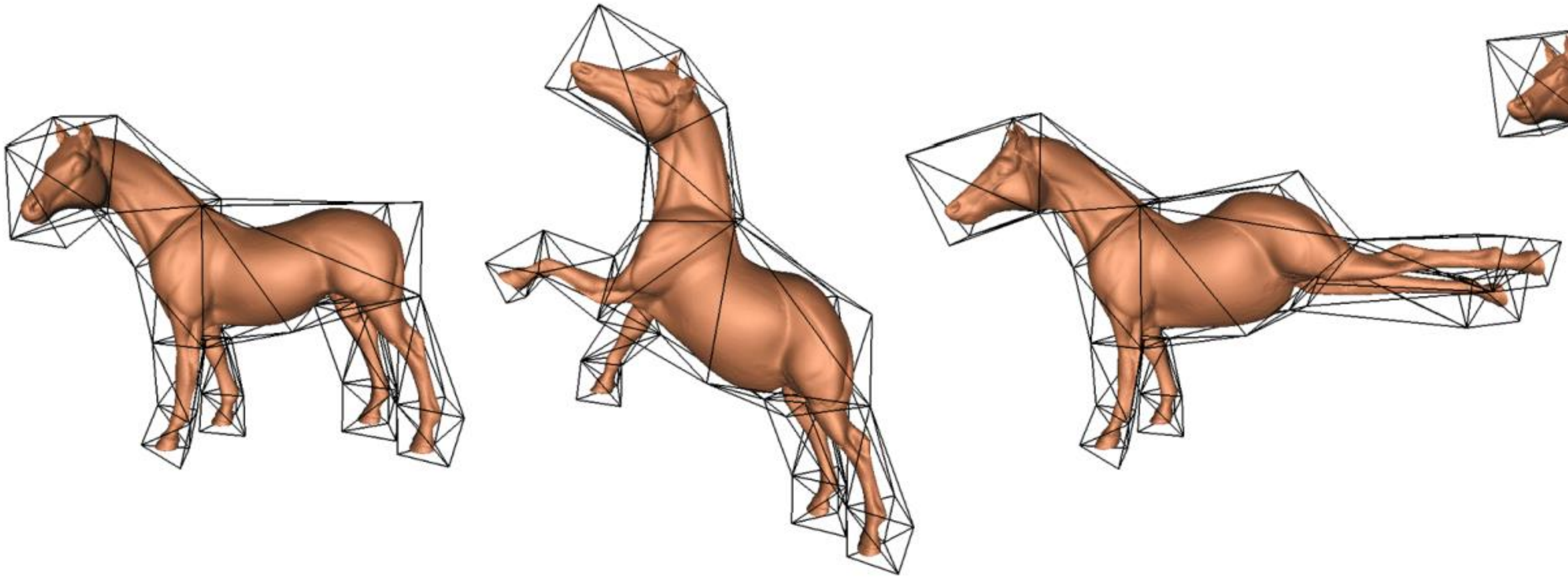
- Compute the coordinates ϕ_i directly from the vertices $v_0, \ldots, v_k \in R^3$



Deformation



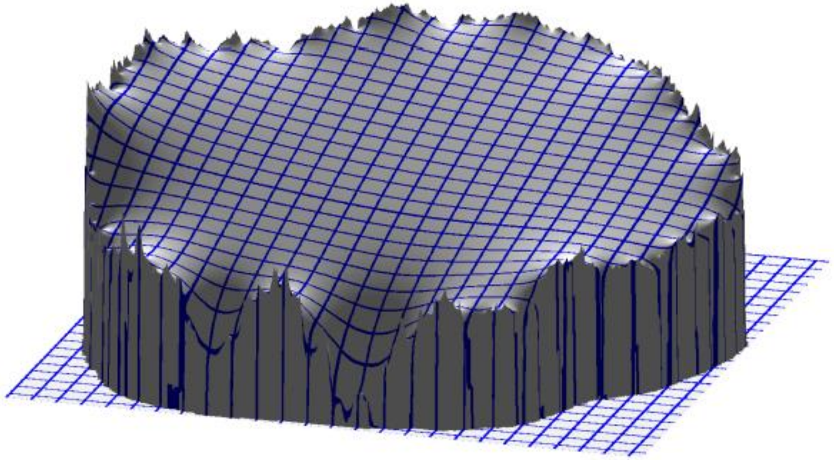
Deformation



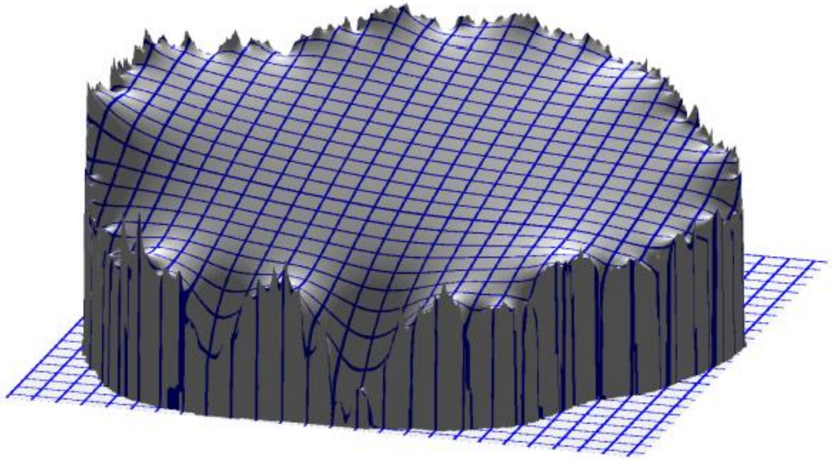
(a) (b) (c) (d)

Poisson image editing

(a) Source patch



(b) Laplace membrane



(c) Mean-value membrane

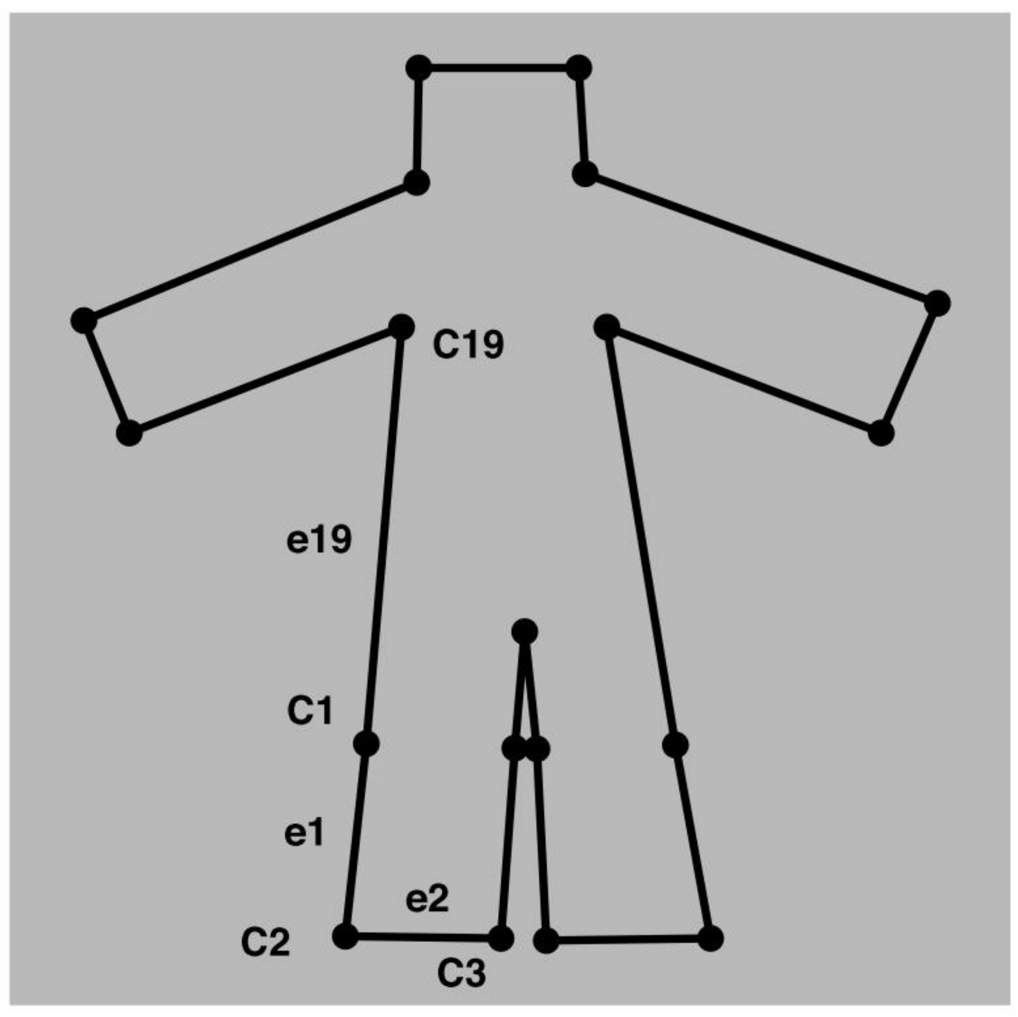
(d) Target image

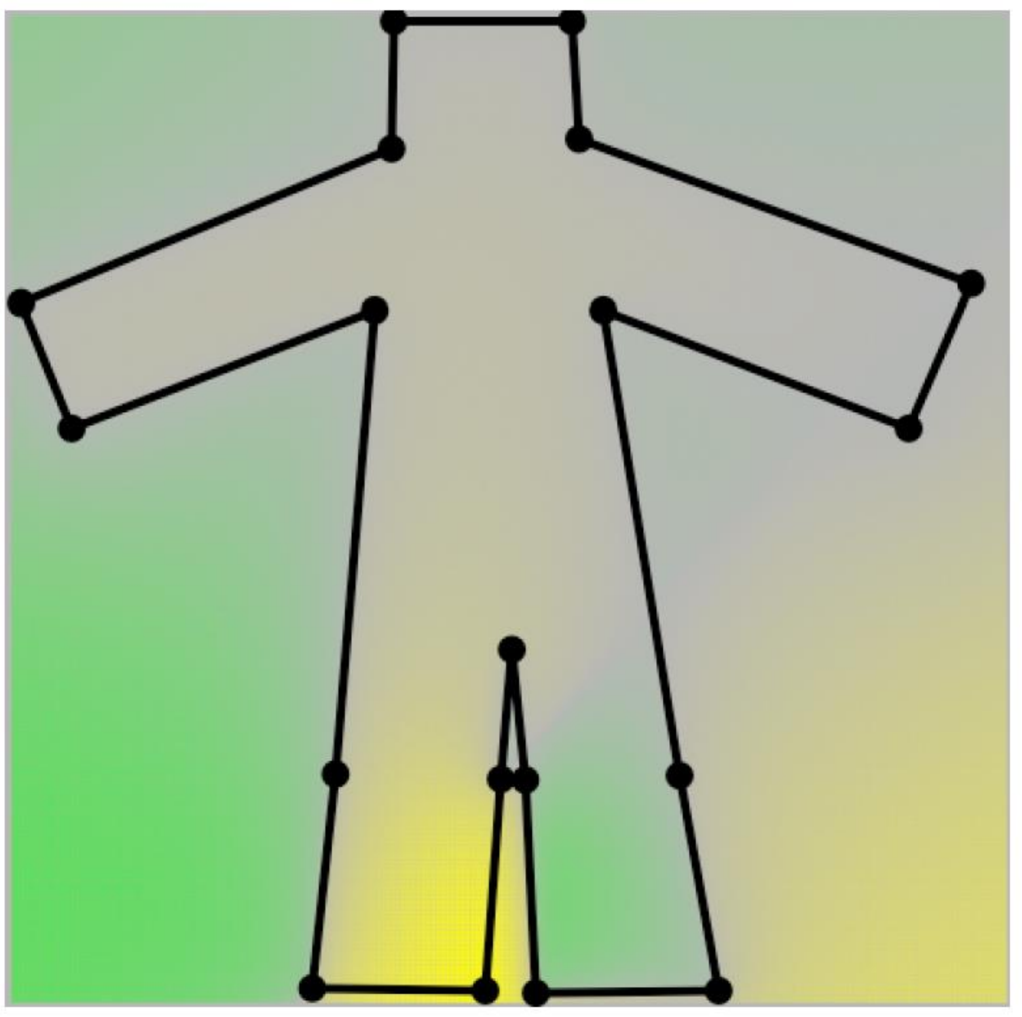
(e) Poisson cloning

(f) Mean-value cloning

Concave Polygon

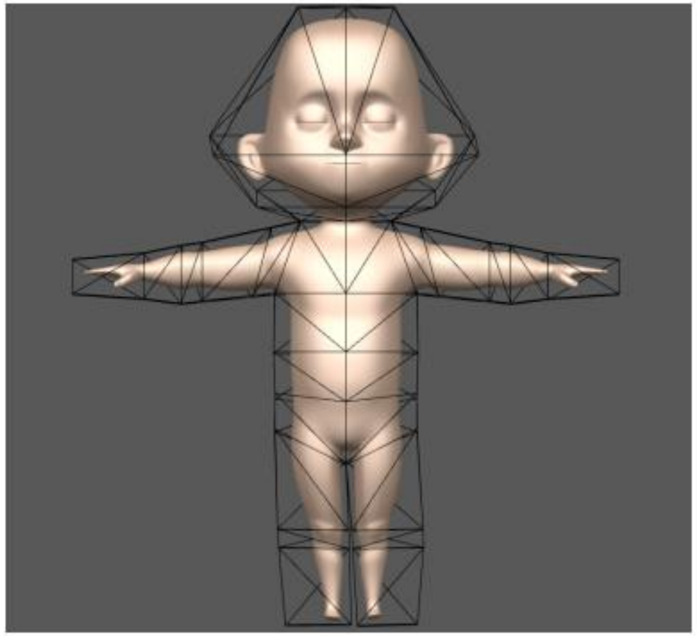
Yellow indicates positive values
Green indicates negative values



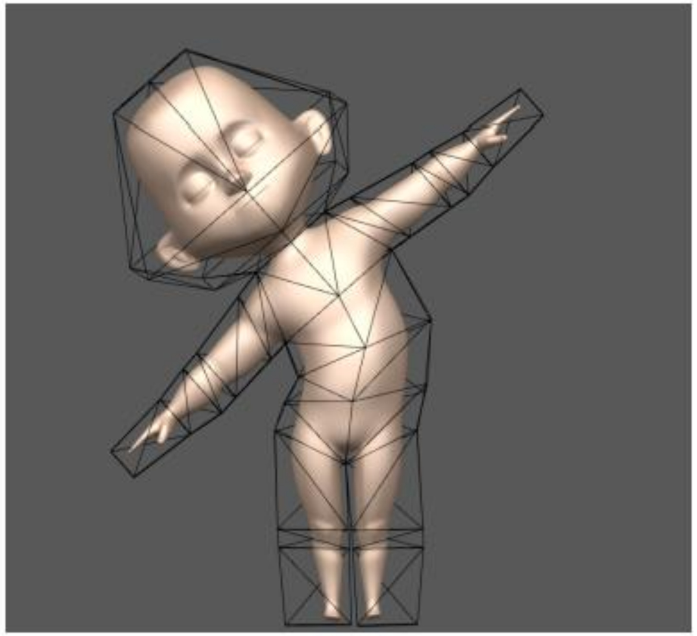


MVC doesn't have

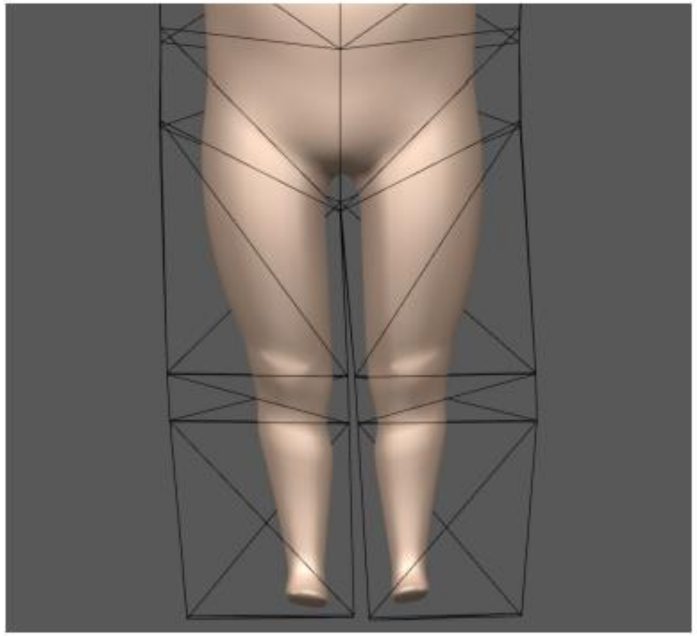
- Non-negativity
  - All weights are positive
- Interior locality
  - Interior locality holds, if, in addition to non-negativity, the coordinate functions have no interior extrema.



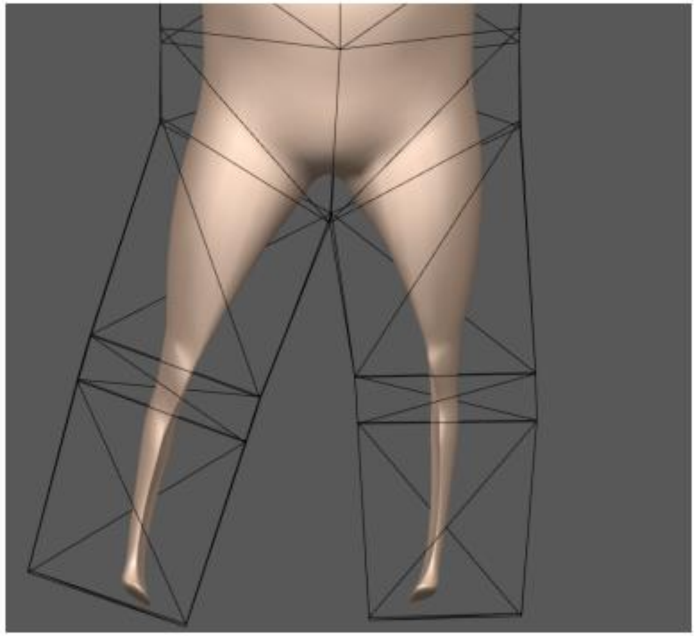
(a)



(b)



(d)



(e)

Outlines

- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Harmonic Coordinates

$$\nabla^2 \phi_i(x) = 0, \forall x \in P$$
$$s.t.\, \phi_i(\partial P) = h_i(\partial P)$$

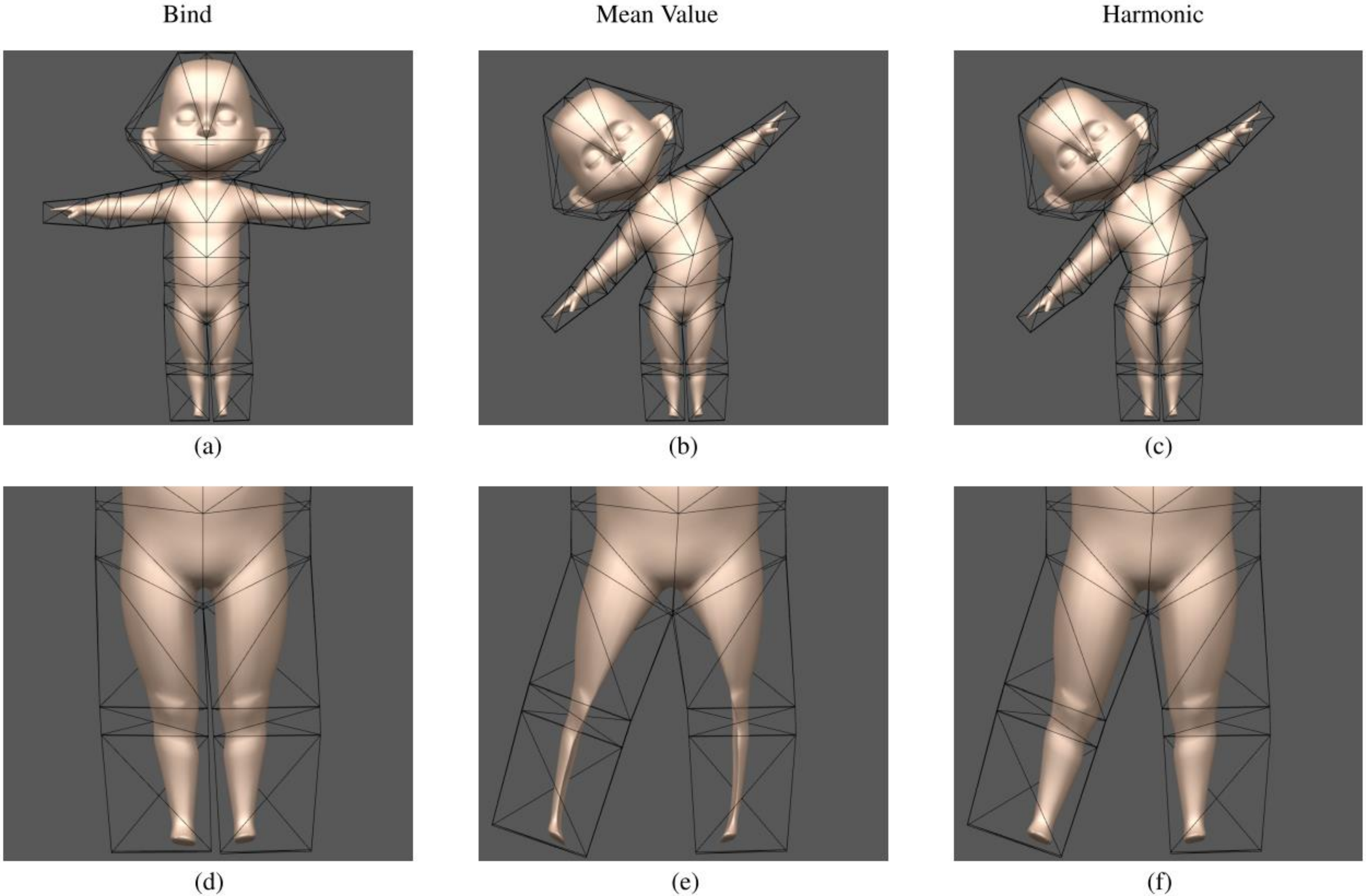
$h_i(\partial P)$: the (univariate) piecewise linear function such that $h_i(v_j) = \delta_{i,j}$.

- Non-negativity: harmonic functions achieve their extrema at their boundaries.
- Interior locality: follows from non-negativity and the fact that harmonic functions possess no interior extrema.

Numerical solution

- 1. Allocate a regular grid of cells that is large enough to enclose the cage.
- 2. Laplacian smooth: For each INTERIOR cell, replace the value of the cell with the average of the value of its neighbors. This Laplacian smoothing step is performed iteratively until the termination criterion is reached.

- A simple hierarchical finite difference solver
  - By first solving the problem at a lower resolution, better starting points for the iteration can be obtained.


Bind
Mean Value
Harmonic
(a)
(b)
(c)
(d)
(e)
(f)

More papers

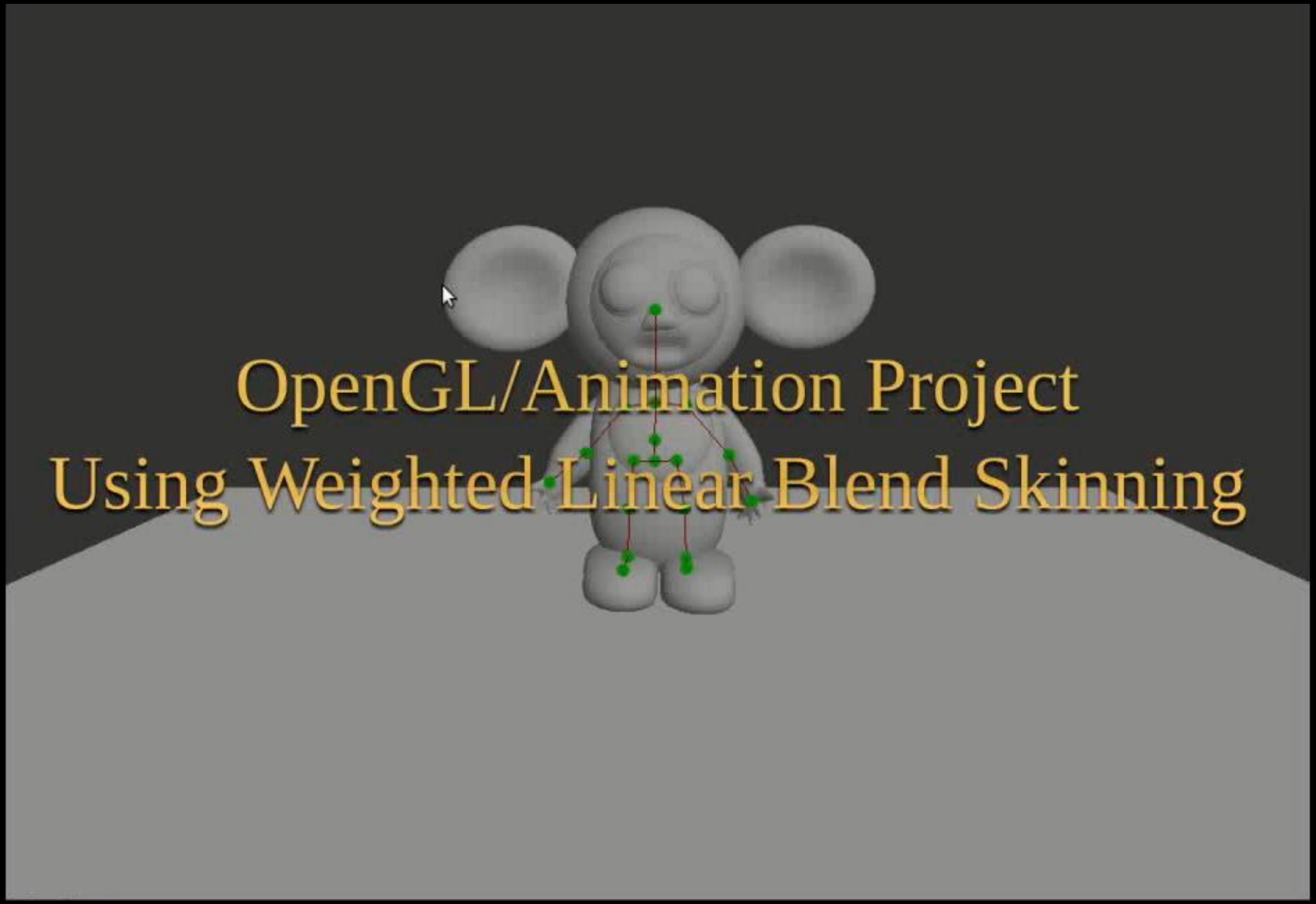
- Green Coordinates, 2008
- Complex Barycentric Coordinates with Applications to Planar Shape Deformation, 2009
- A Complex View of Barycentric Mappings, 2011
- Poisson Coordinates, 2013
- Cubic Mean Value Coordinates, 2013
- ……

Outlines

- Introduction
- Barycentric coordinates on convex polygons
- Inverse bilinear coordinates
- Mean value coordinates
- Harmonic Coordinates
- A general construction

Linear blend skinning

- Skeleton-subspace deformation



Linear blend skinning - input data

- Rest pose shape
  - Represented as a polygon mesh
  - The mesh connectivity is assumed to be constant, i.e., only vertex positions will change during deformations.
  - Rest-pose vertices: $v_1, \ldots, v_n \in R^3$
- Bone transformations
  - A list of matrices
  - Spatial transformations aligning the rest pose of bone i with its current (animated) pose.
- **Skinning weights**
  - For vertex v_i, we have weights $w_{i,1}, \ldots, w_{i,m} \in R$.
  - Each weight $w_{i,j}$ describes the amount of influence of bone j on vertex i.

Linear blend skinning - Bone transformations

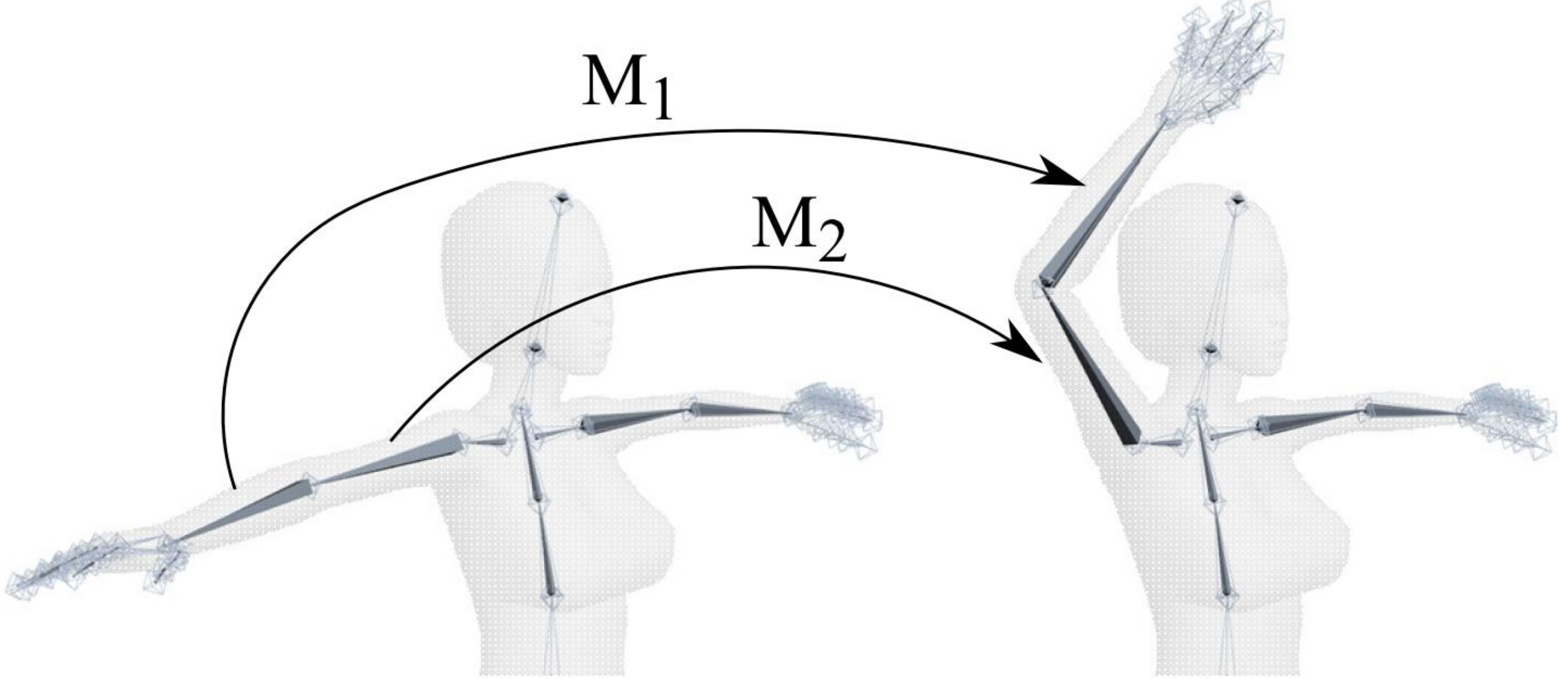
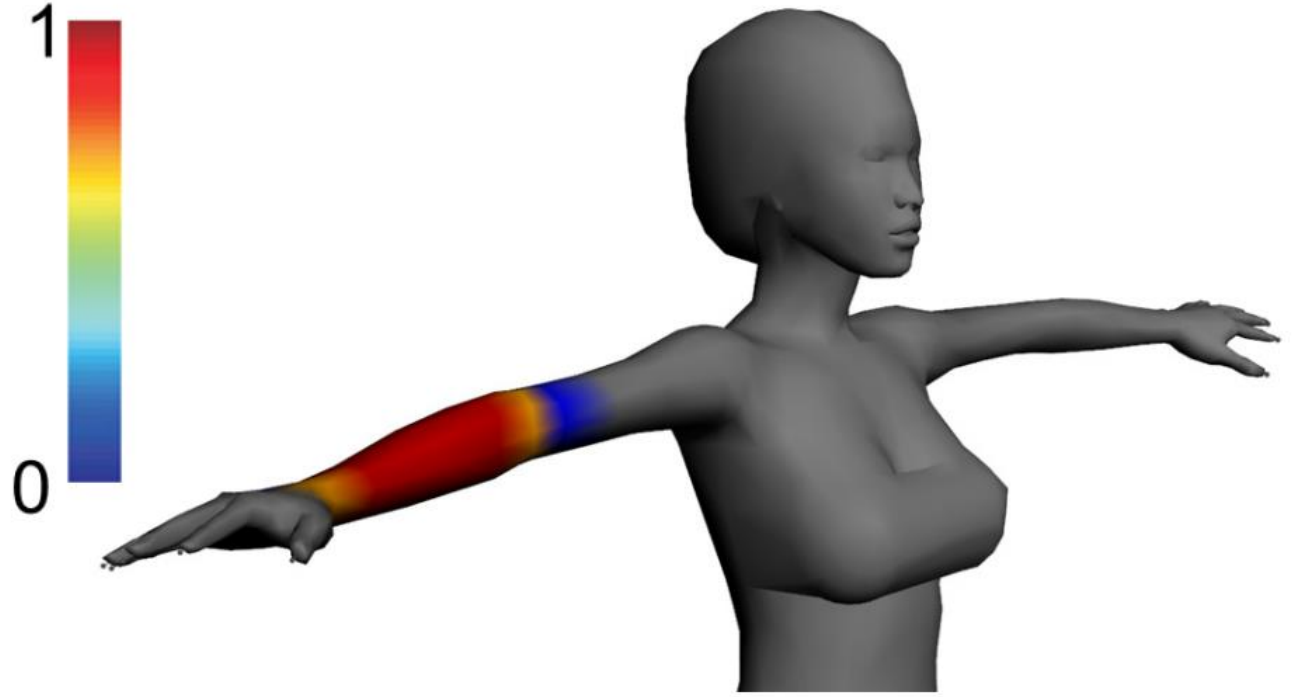


Figure 1: *Bone transformations (lower and upper arm bones) for one example deformed pose.*

Linear blend skinning - Skinning weights



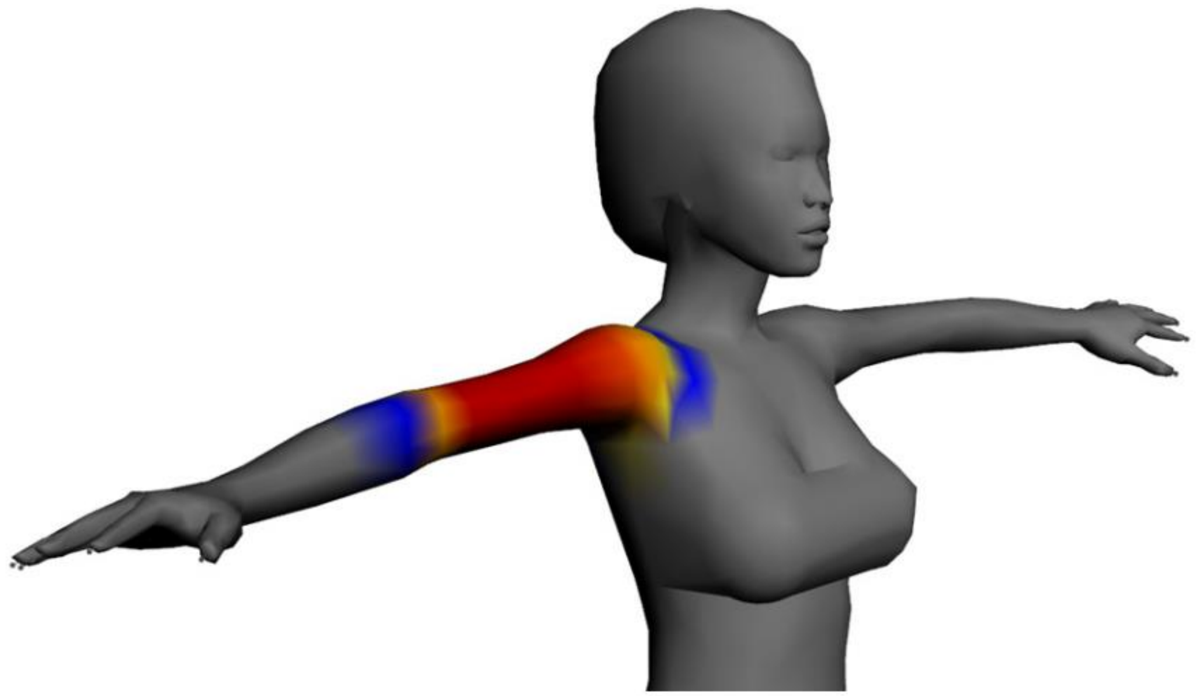


Figure 2: *Influence weights corresponding to lower and upper arm bones.*

Deformed vertex positions

$$v_i^{new} = \sum_{j=1}^{m} w_{i,j} T_j v_i = \left(\sum_{j=1}^{m} w_{i,j} T_j \right) v_i$$

The latter form highlights the fact that the rest pose vertex v_i is transformed by a linear combination (blend) of bone transformation matrices T_j.

Recap of properties

- Interpolation (Lagrange property)
- Smoothness
- Non-negativity ($\phi_i(x) \geq 0$)
- Interior locality
- Linear reproduction ($\sum_{i=1}^{n} \phi_i(x) v_i = x$)
- Affine-invariance = Partition of unity ($\sum_{i=1}^{n} \phi_i(x) = 1$)

Some papers

- Bounded Biharmonic Weights for Real-Time Deformation, 2011
- Local Barycentric Coordinates, 2014
- Linear Subspace Design for Real-Time Shape Deformation, 2015

Bounded Biharmonic Weights for Real-Time Deformation

- Real-time performance is critical for both interactive design and interactive animation.
- Among all deformation methods, linear blending and its variants dominate practical usage thanks to their speed
  - each point on the object is transformed by a linear combination of a small number of affine transformations.
- Real-time object deformations would be easier with support for all handle types: points, skeletons, and cages.
  - Goal: smooth and intuitive deformation

Various handles

Points
Bones
Cages

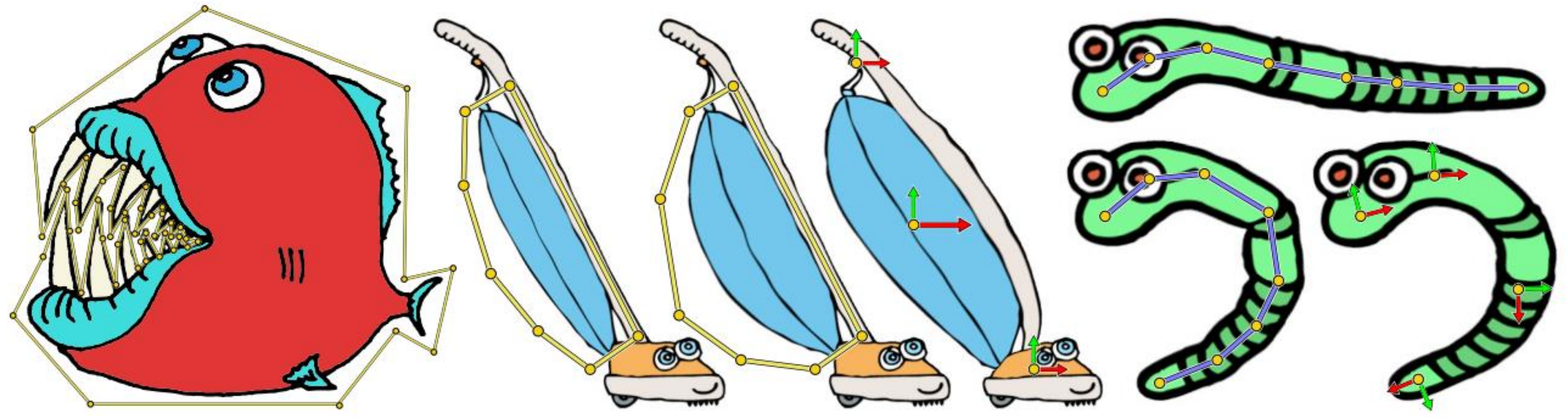
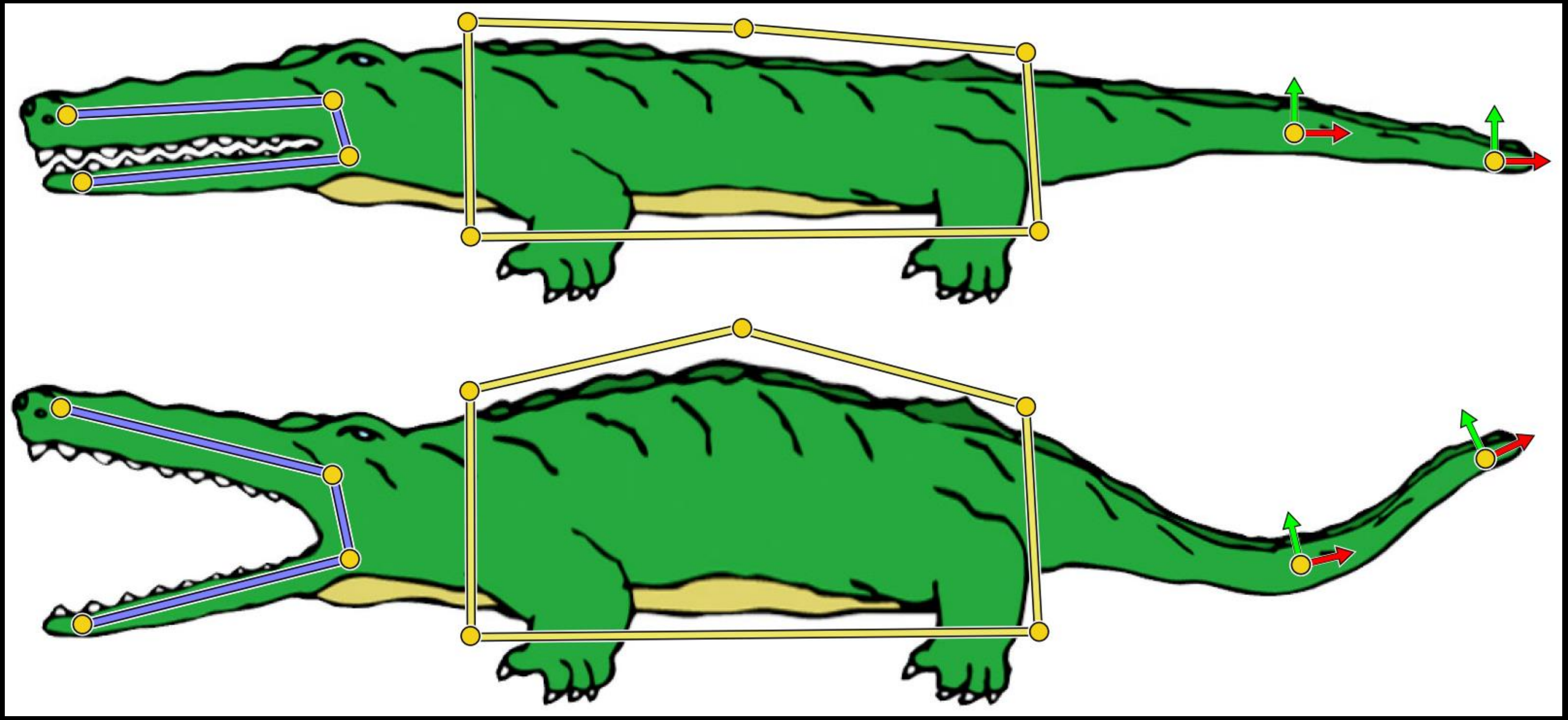


Figure 2: *Left to right: Although cages allow flexible control, setting up a closed cage can be both tedious and unintuitive: the* Pirahna*'s jaws require weaving around the teeth. In the case of the* Vacuum, *points can provide crude scaling effects, while cages provide precise scaling articulation. Point handles can provide loose and smooth control, while achieving the same effect with a skeleton results in an overly complex armature.*

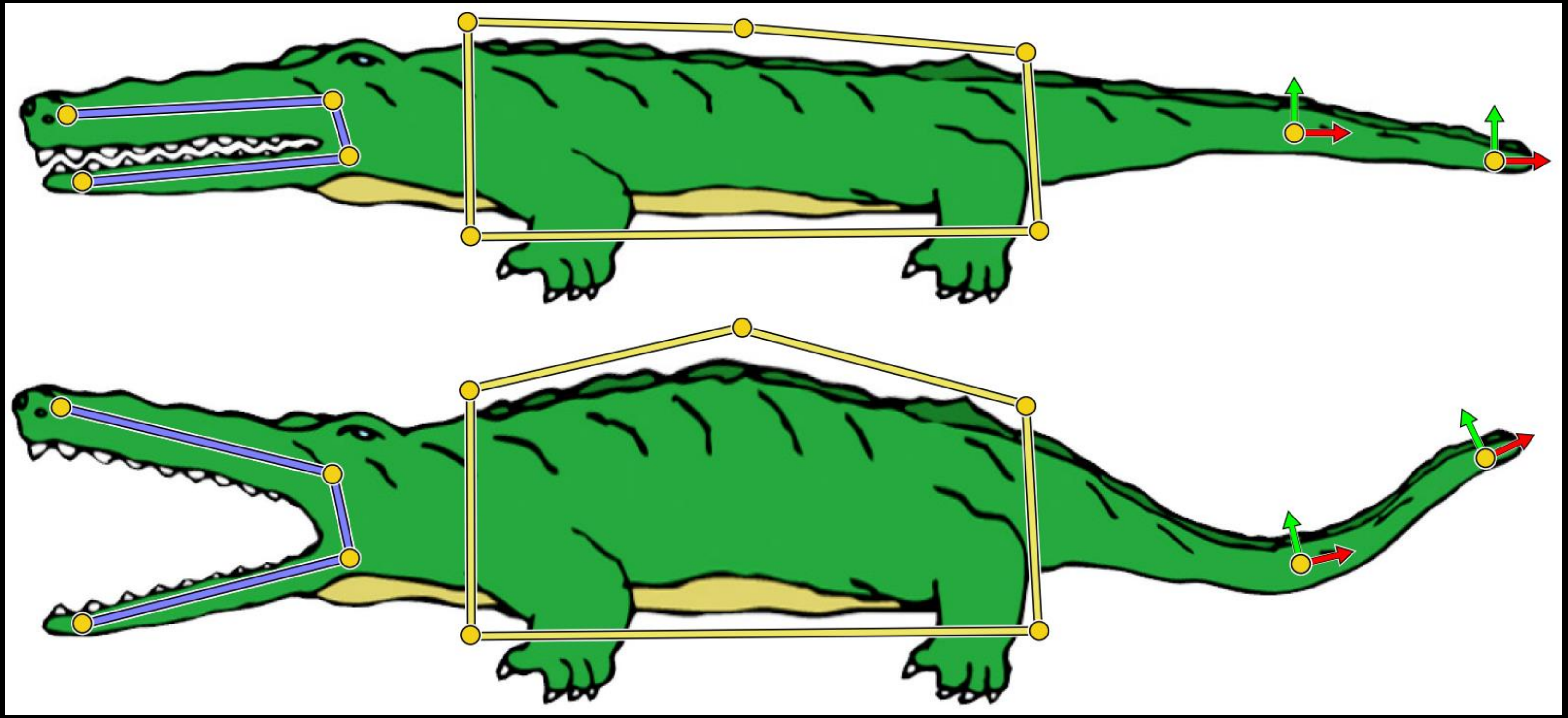
Handles - Points

- Points are quick to place and easy to manipulate.
- They specify local deformation properties (position, rotation and scaling) that smoothly propagate onto nearby areas of the object.



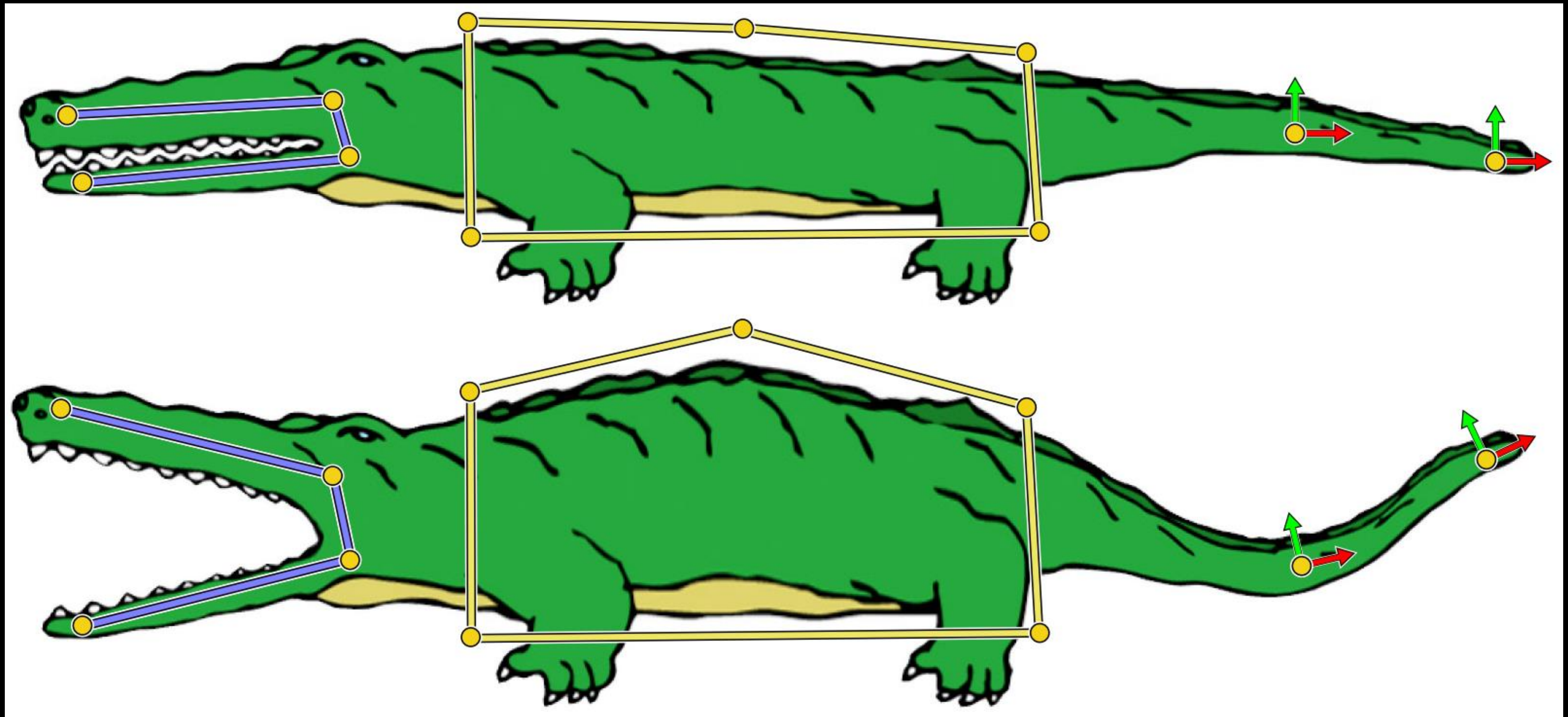
Handles - Bones

- Bones make some directions stiffer than others.
- If a region between two points appears too supple, bones can transform it into a rigid limb.



Handles - Cages

- Cages allow influencing a significant portion of the object at once, making it easier to control bulging and thinning in regions of interest.



Bounded biharmonic weights

$$\underset{w_j,\ j=1,\dots,m}{\arg\min} \sum_{j=1}^{m} \frac{1}{2} \int_{\Omega} \|\Delta w_j\|^2 dV \tag{2}$$

$$\text{subject to:}\quad w_j|_{H_k} = \delta_{jk} \tag{3}$$

$$w_j|_F \text{ is linear} \qquad \forall F \in \mathcal{F}_{\mathcal{C}} \tag{4}$$

$$\sum_{j=1}^{m} w_j(\mathbf{p}) = 1 \qquad \forall \mathbf{p} \in \Omega \tag{5}$$

$$0 \le w_j(\mathbf{p}) \le 1,\ \ j = 1, \dots, m, \qquad \forall \mathbf{p} \in \Omega, \tag{6}$$

Properties

- Smoothness ($\Delta^2 w_j = 0$)
  - The bounded biharmonic weights are C^1 at the handles and C^∞ everywhere else, provided that the posed boundary conditions are smooth.
- Non-negativity
- Shape-awareness: bi-Laplacian operator
- Partition of unity
- Locality and sparsity: just observation
- No local maxima: experimentally observed
- No Linear reproduction ($\sum_{i=1}^{n} \phi_i(x) v_i = x$)

Properties

Figure 3: *Bounded biharmonic weights are smooth and local: the blending weight intensity for each handle is shown in red with white isolines. Each handle has the maximum effect on its immediate region and its influence disappears in distant parts of the object.*

Properties

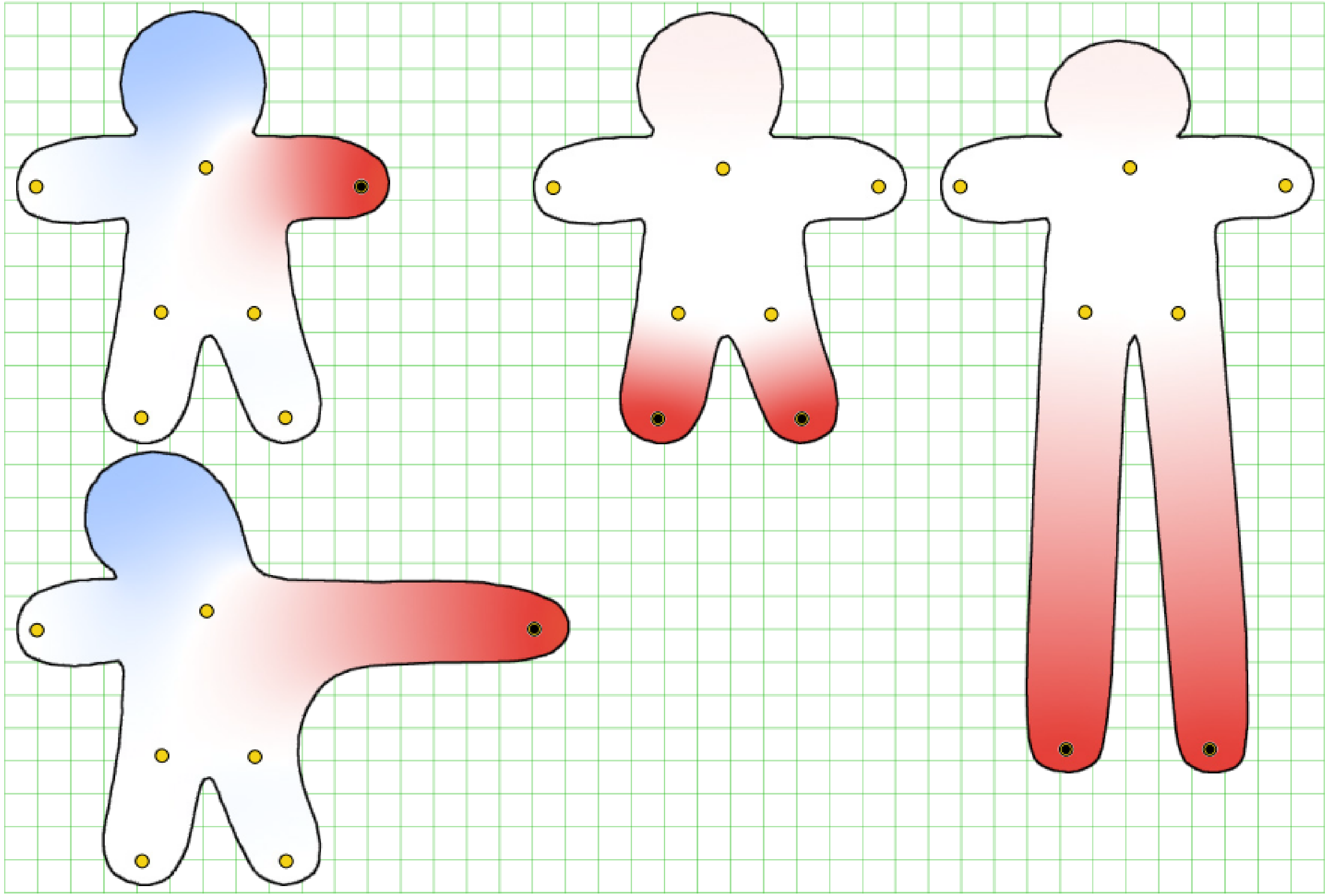


Figure 4: *Weights like unconstrained biharmonic functions that have negative weights (left) and extraneous local maxima (right) lead to undesirable and unintuitive behavior. Notice the shrinking of the head on the right.*

Bounded Biharmonic Weights for Real-Time Deformation

Alec Jacobson[1]
Ilya Baran[2]
Jovan Popović[3]
Olga Sorkine[1,4]

[1]New York University
[2]Disney Research, Zurich
[3]Adobe Systems, Inc.
[4]ETH Zurich

This video contains narration

Local Barycentric Coordinates

- A local change in the value at a single control point will create a global change by propagation into the whole domain.

- Global nature
  - The first one is the lack of locality and control over a deformation.
  - The second drawback is scalability.
    - Most practical applications store barycentric coordinates using one scalar value per control point for every vertex of the target domain.

Formulation

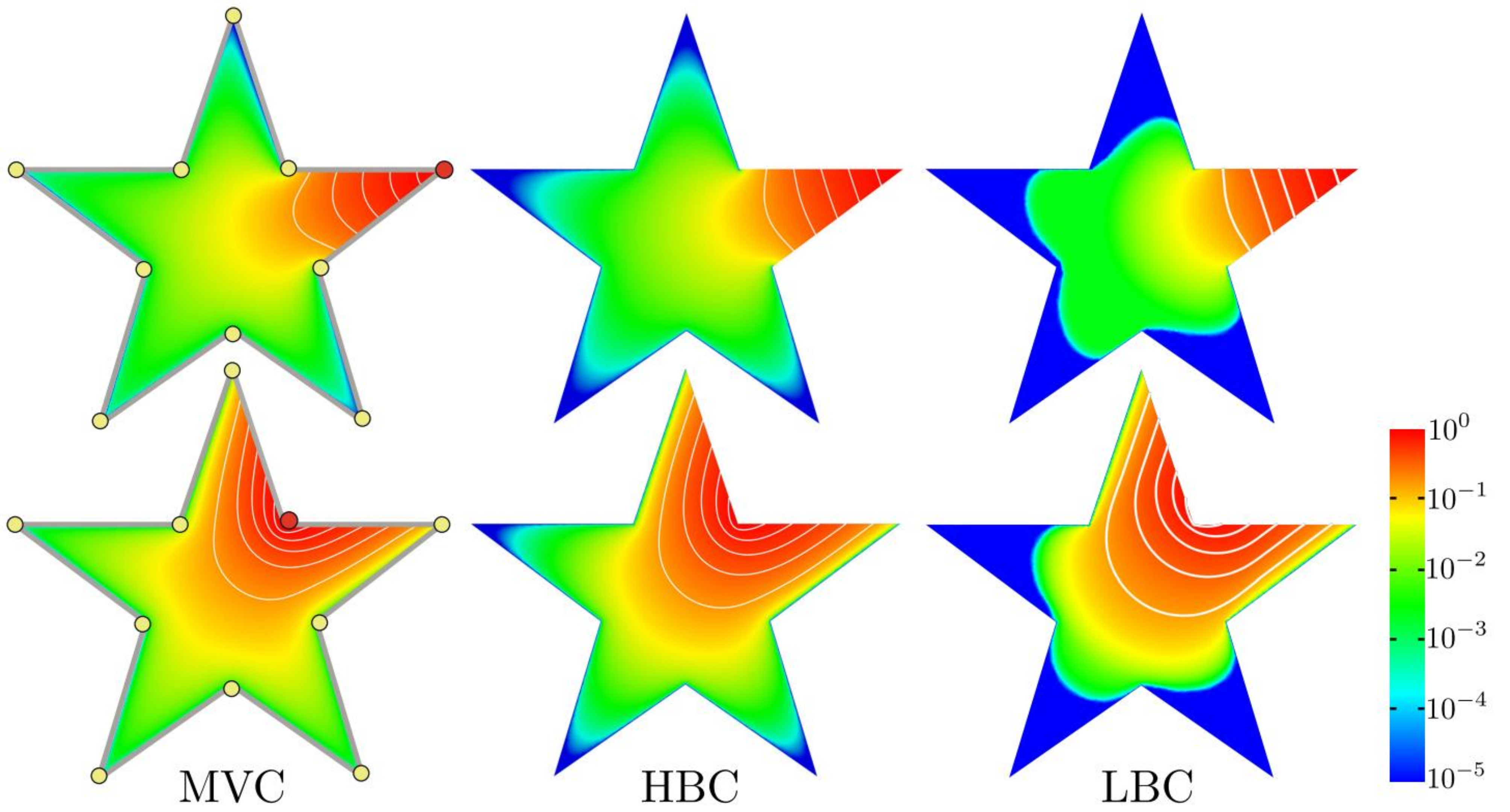
$$\min_{w_1,\ldots,w_n} \sum_{i=1}^{n} \int_{\Omega} |\nabla w_i|$$

$$\text{s.t.} \sum_{i=1}^{n} w_i(\mathbf{x})\mathbf{c}_i = \mathbf{x}, \; \sum_{i=1}^{n} w_i = 1, \; w_i \geq 0, \; \forall \mathbf{x} \in \Omega,$$

$$w_i(\mathbf{c}_j) = \delta_{ij} \; \forall i, j,$$

$$w_i \text{ is linear on cage edges and faces } \forall i. \tag{5}$$

Locality



Local extrema

- TV measures oscillation, and hence its minimization inhibits local extremal values.

Demo

Linear Subspace Design for Real-Time Shape Deformation

- Linear reproduction
  - Cot weights of Laplacian satisfy.
  - ?

Demo