Shape deformation (II)

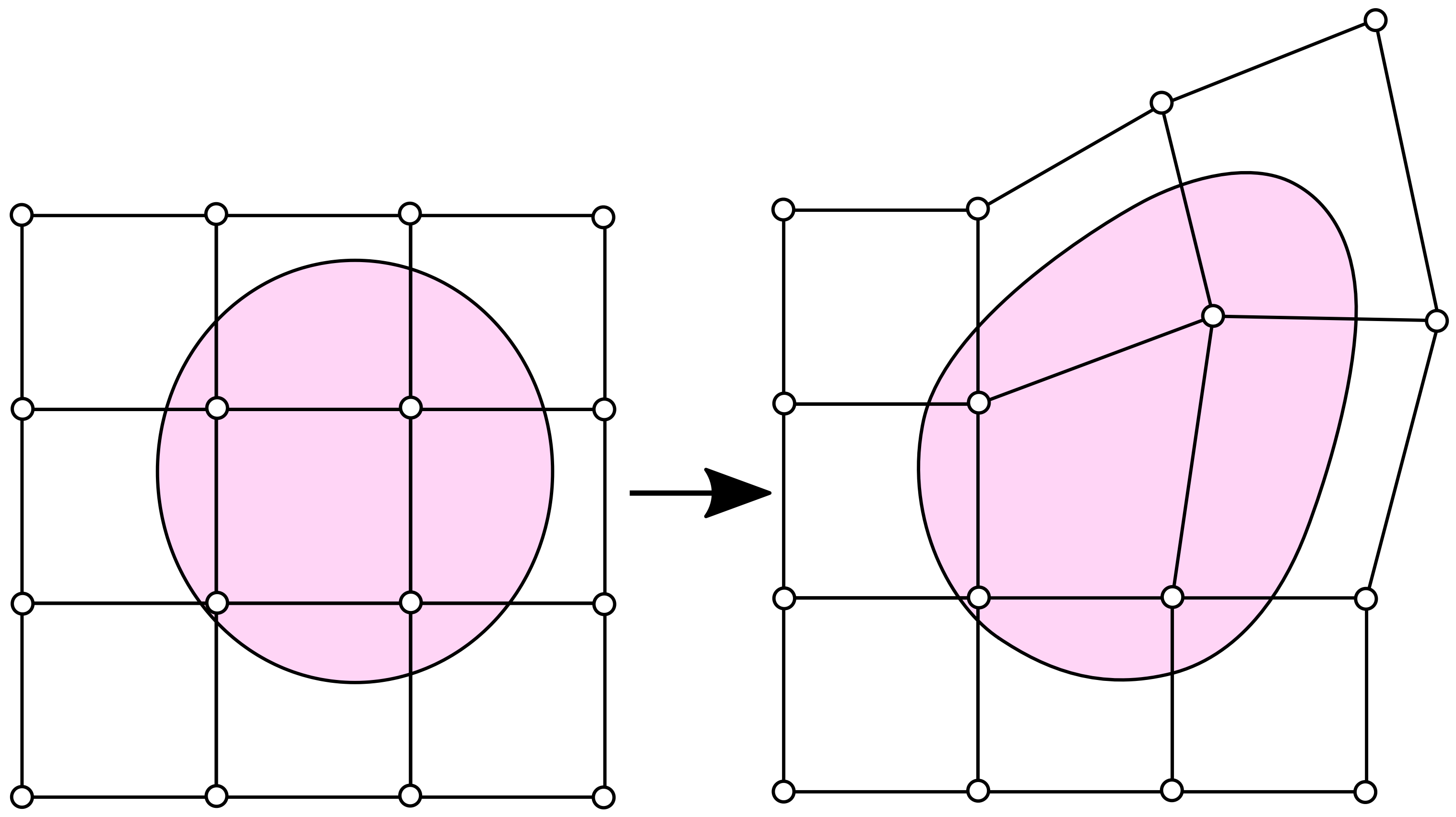
Volume-based deformation

- *Free Form Deformation*
- *Cages*
- *Vector field*

Free Form Deformation

FFD / Grid Based Deformation

- Embeed shape within a rectangular grid (lattice)
- Deform the grid $\rightarrow$ space deformation map



How to compute the space deformation ?

[redacted]

Kestrel Moon

Idea: Approximation function

Bezier, BSpline, NURBS, etc.

Reminder for 1D curve c approximating points $(p_i)_{i=0..N}$ (control polygon)

$$c(u) = \sum_{i=0}^{N} b_i(u)\, p_i$$

$$b_i(u) = \binom{N}{i}\, u^i\, (1-u)^{N-i} \;, \quad u \in [0,1]$$ (Bernstein Polynomial)

For a 2D surface S approximating points p_{ij}

$$S(u,v) = \sum_i \sum_j b_i(u)\, b_j(v)\; p_{ij}$$ (tensor-product)

For a 3D volume V approximating points on the lattice p_{ijk}

$$V(u,v,w) = \sum_i \sum_j \sum_k b_i(u) b_j(v) b_k(w)\; p_{ijk}$$

$\Rightarrow$ Use V as a spatial deformation on vertex coordinates (u,v,w), use p_{ijk} as grid.

Free Form Deformation

Given

- g_{ijk}: position of the grid coordinates
- $p_m = (x_m, y_m, z_m) \in [0,1]^3$: Initial vertex coordinates (expressed with respect to the grid)

$$\Rightarrow q_m = \sum_i \sum_j \sum_k b_i(x_m)\, b_j(y_m)\, b_k(z_m)\; g_{ijk}$$

Note:

$w_{m\,ijk} = b_i(x_m) b_j(y_m) b_k(z_m)$

is a scalar weights independant of the deformation

$\rightarrow$ can be precomputed

$$\Rightarrow q_m = \sum_{ijk} w_{m\,ijk}\; g_{ijk}$$

FFD limits and extensions

FFD introduced in 1986

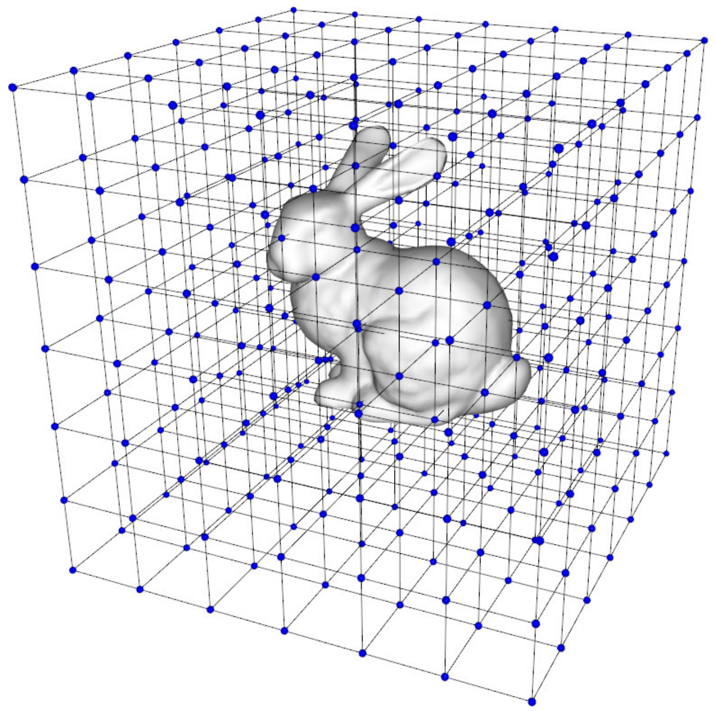
[Thomas Sederberg, Scott Parry. Free-Form deformation of solid geometric models, SIGGRAPH 1986.]

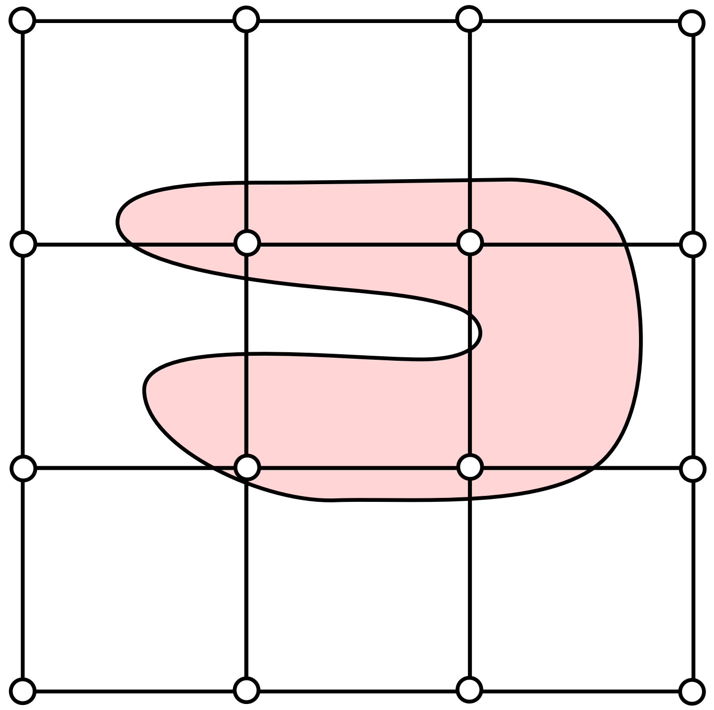
Initially described with Bezier polynomials

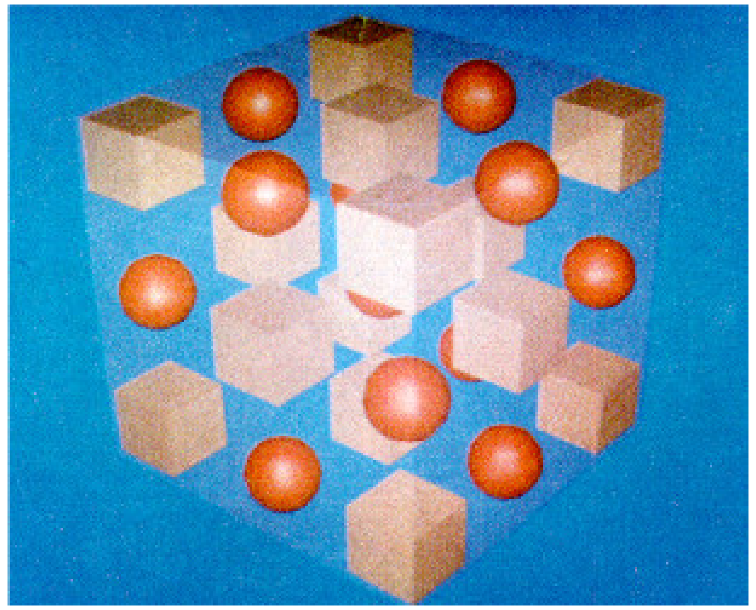
- But extends to Spline with any basis function
- *BSpline, NURBS, etc*

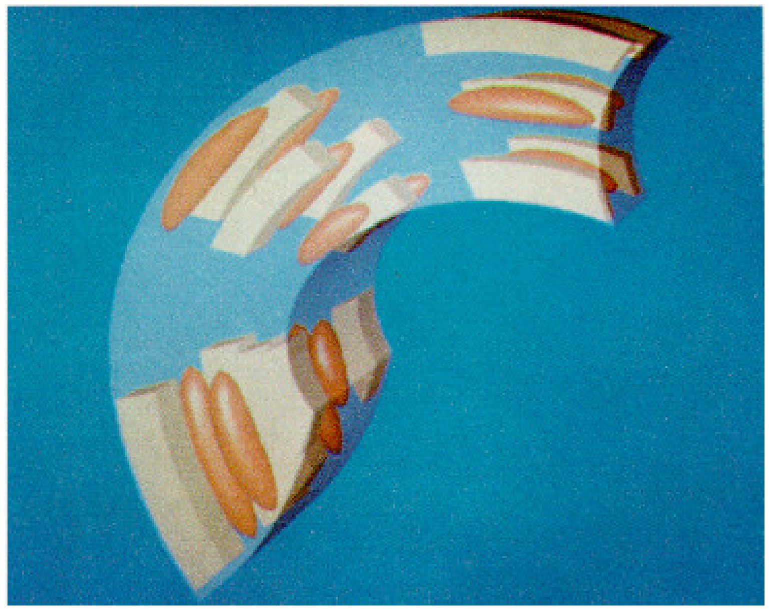
Limitations

- - Large grid is hard to manipulate
- - Doesn't take into account shape morphology





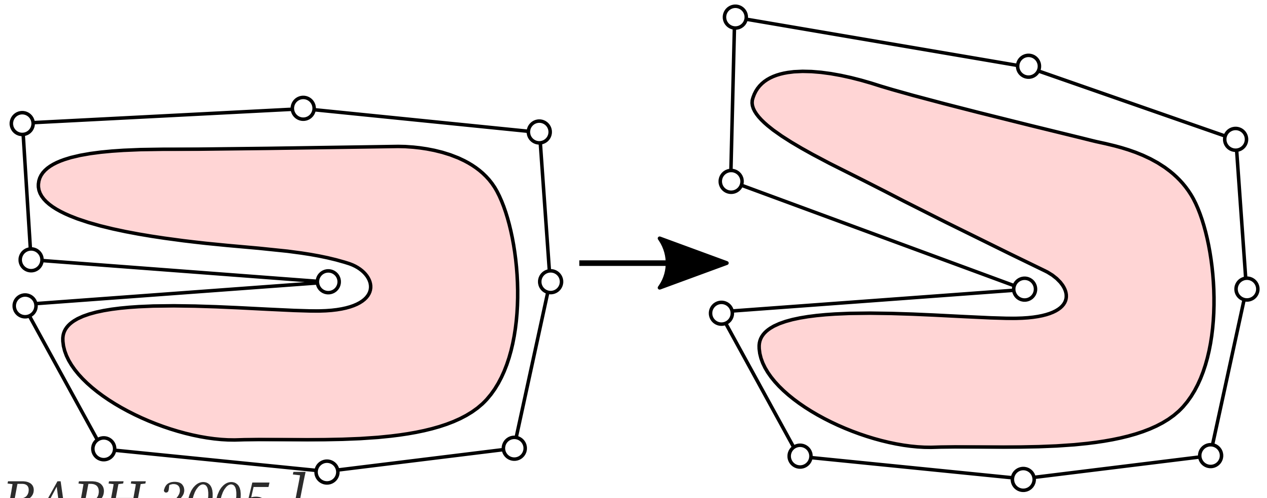




Cage-based Deformation

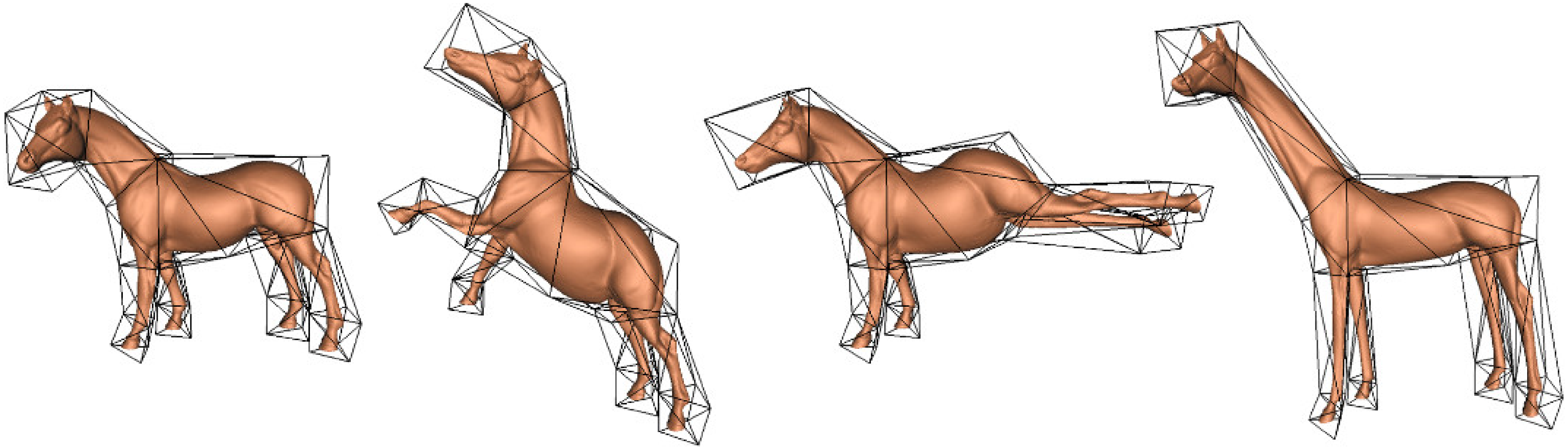
Use a cage surrounding the shape

(+) Adapts deformation locality to shape morphology/topology



Introduced for 3D meshes in 2005

[Tao Ju, Scott Schaefer, Joe Warren. Mean value coordinates for closed triangular meshes. SIGGRAPH 2005]



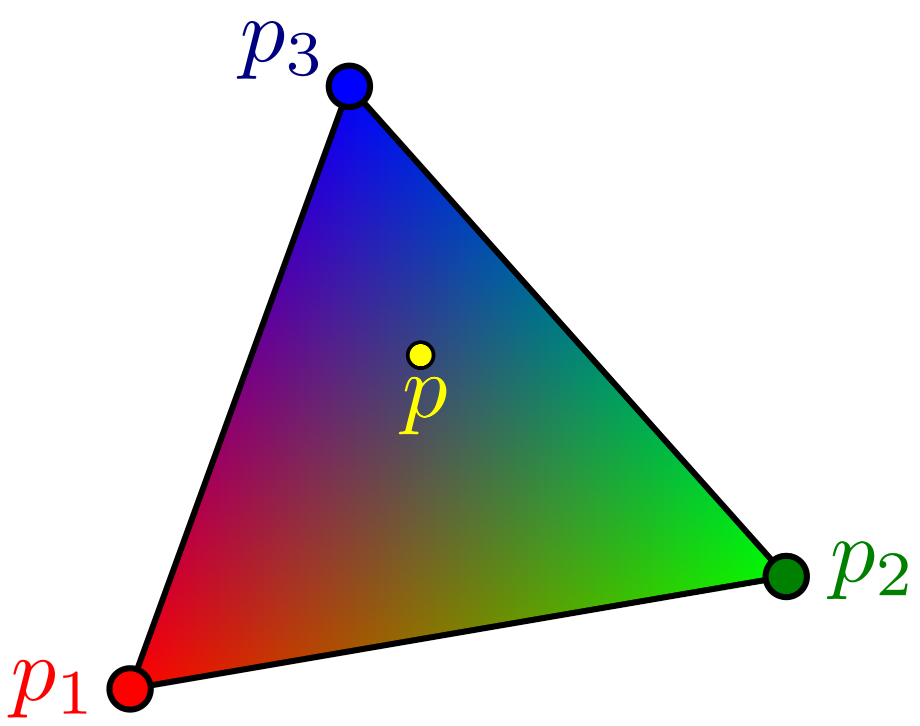
How to compute the spatial deformation from arbitrary cage ?

Barycentric coordinates

Reminder barycentric coordinates for Triangle

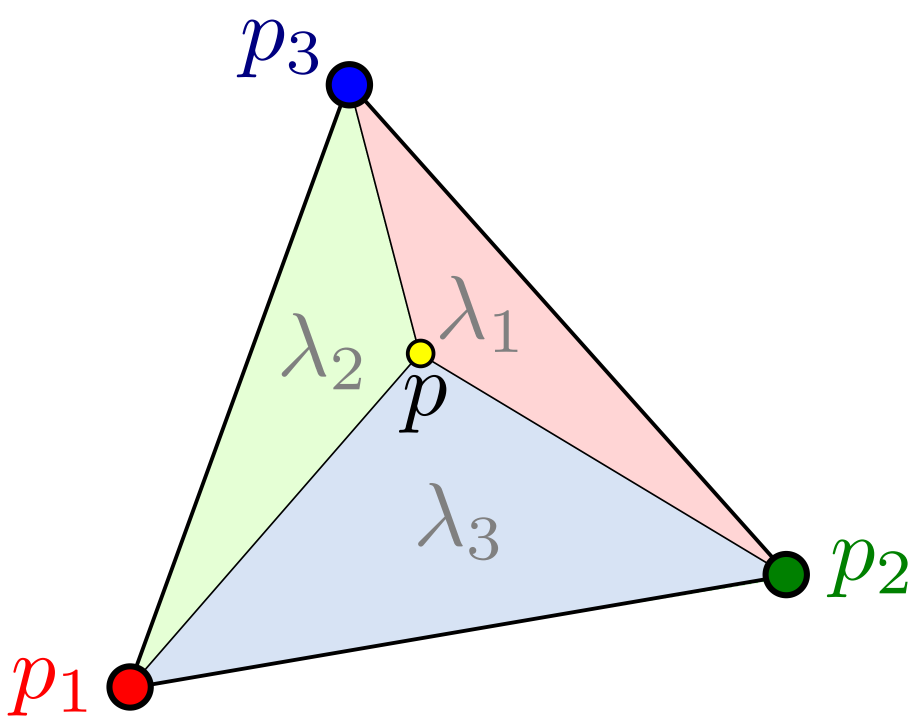
Triangle $(p_1\, p_2\, p_3)$ - barycentric coordinates $(\lambda_1, \lambda_2, \lambda_3)$

$$p \in (p_1\, p_2\, p_3) \Rightarrow p = \lambda_1\, p_1 + \lambda_2\, p_2 + \lambda_3\, p_3$$
$$\lambda_1 + \lambda_2 + \lambda_3 = 1\,, \quad (\lambda_1, \lambda_2, \lambda_3) \in [0, 1]^3$$



$$\lambda_1 = area(p_3\ p\ p_2)/\mathcal{A}$$
$$\lambda_2 = area(p_1\ p\ p_3)/\mathcal{A}$$
$$\lambda_3 = area(p_2\ p\ p_1)/\mathcal{A}$$
$$with,\ \mathcal{A} = area(p_1\ p_2\ p_3)$$
$$and,\ area(A\ B\ C) = \frac{1}{2}\|(p_2 - p_1) \times (p_3 - p_1)\|$$

Note. Similar for tetrahedron: $p = \lambda_1\, p_1 + \lambda_2\, p_2 + \lambda_3\, p_3 + \lambda_4\, p_4$



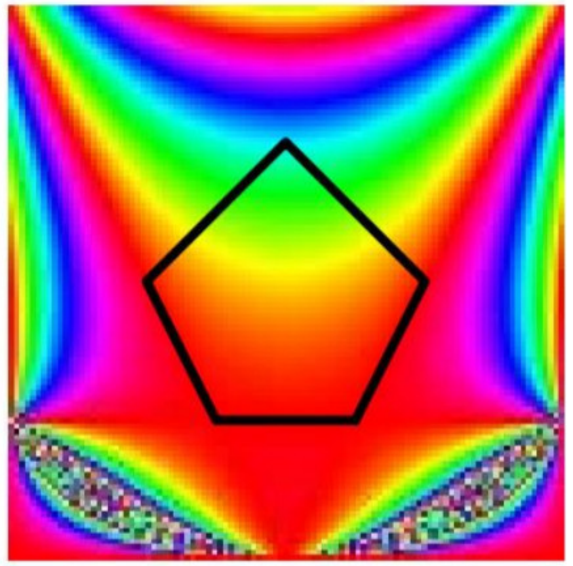
Idea: Generalize barycentric coordinates to arbitrary cages $p = \sum_i \lambda_i\, p_i = \sum_i \omega_i\, p_i / \sum_i \omega_i$

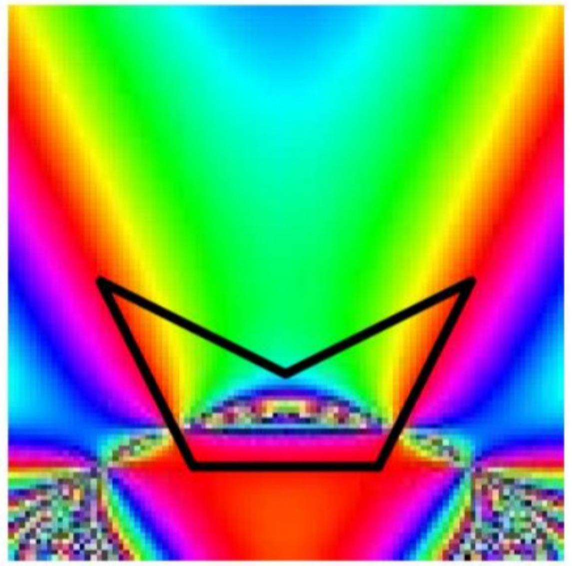
Wachspress Coordinates

$$w_i = \frac{A_{i+1} + A_i - B_i}{A_{i+1}\, A_i}$$

$$w_i = \frac{cot(\alpha_{i-1}) + cot(\beta_i)}{\|p_i - p\|}$$

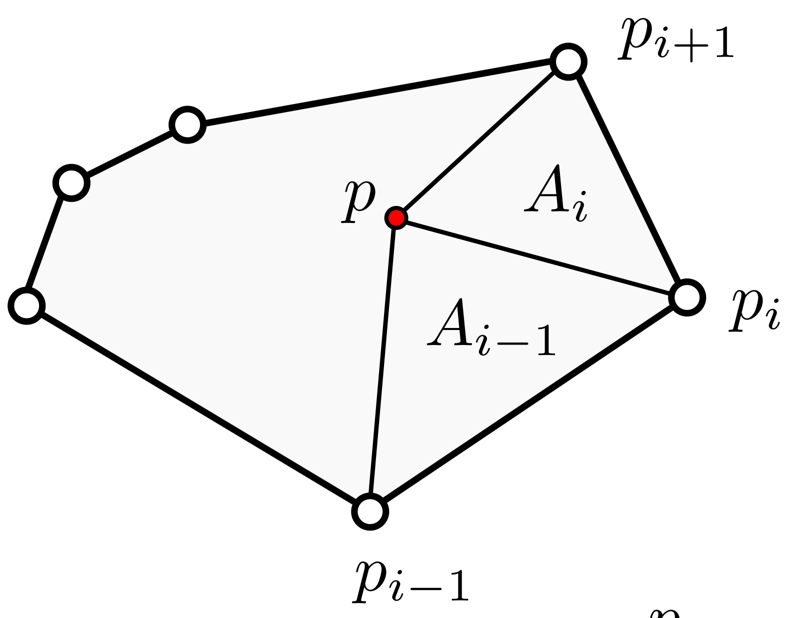
Introduced in 1975, Eugene Wachspress, A Rational Finite Element Basis.

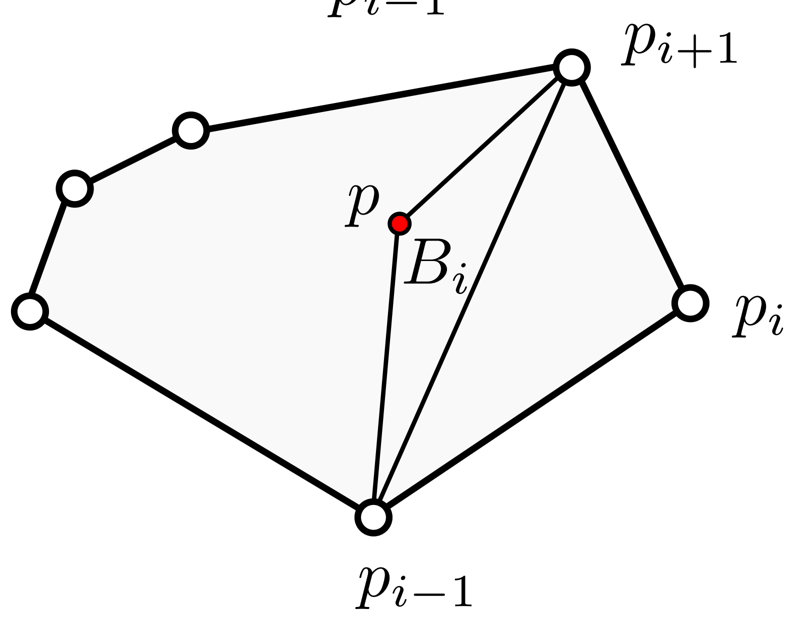


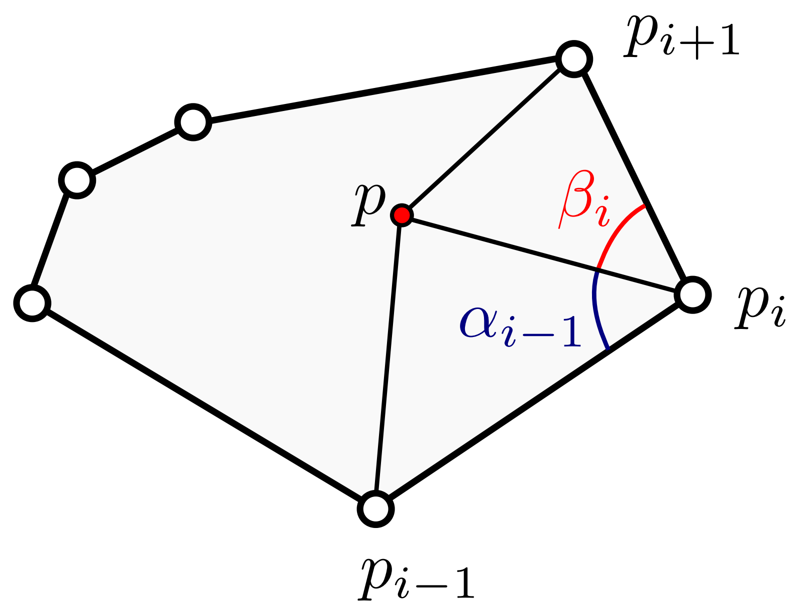


(-) Limited to 2D

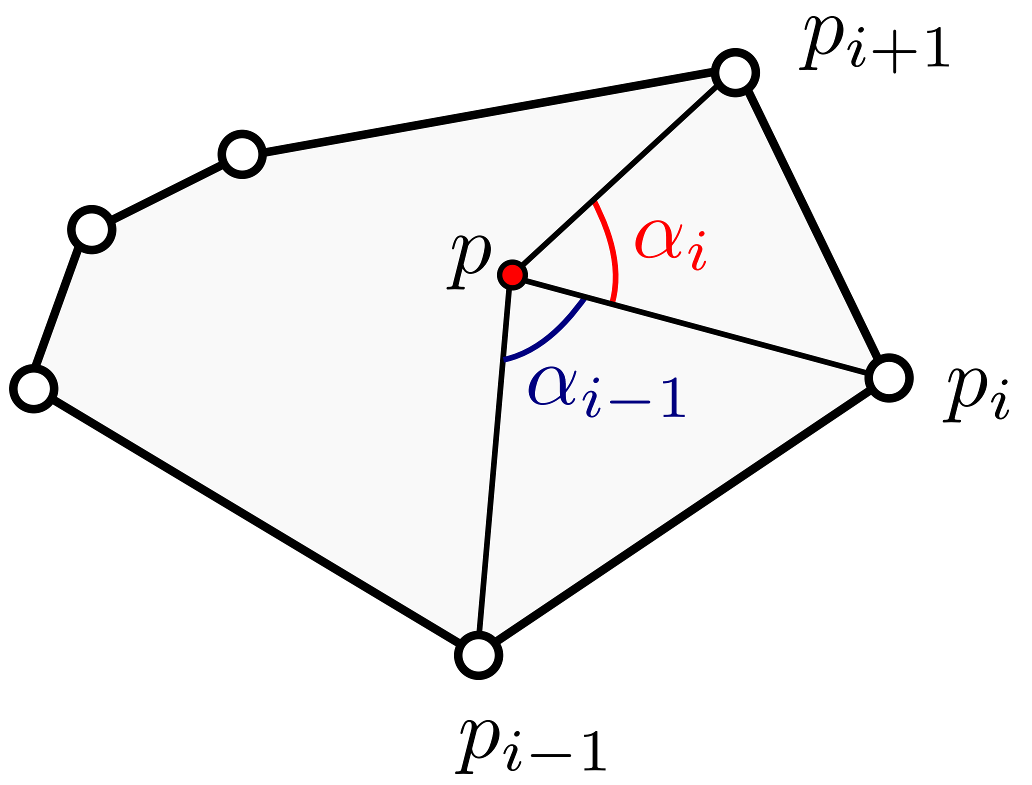
(-) Limited to convex polygons (negative weights, poles)







Mean Value



In 2D

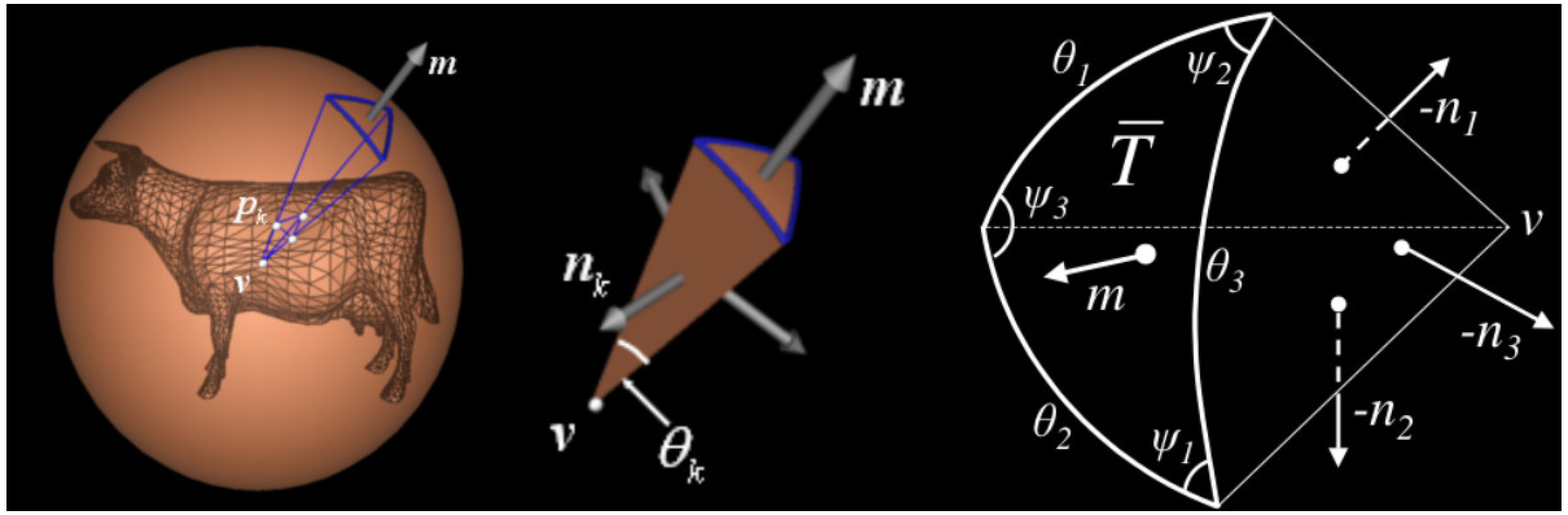
$$\omega_i = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|p_i - p\|}$$

[Michael S. Floater. Mean value coordinates. CAGD, 2003]

In 3D

$$\omega_i = \frac{n_i \cdot m}{n_i \cdot (p_i - v)}$$

$$m = \frac{1}{2} \sum_i \theta_i \, n_i$$



[Michael S. Floater, Géza Kos, Martin Reimers. Mean value coordinates in 3D. CAGD, 2005]

[Tao Ju, Scott Schaefer, Joe Warren. Mean Value Coordinates for Closed Triangular Meshes. SIGGRAPH 2005]

Mean Value - Idea in 2D

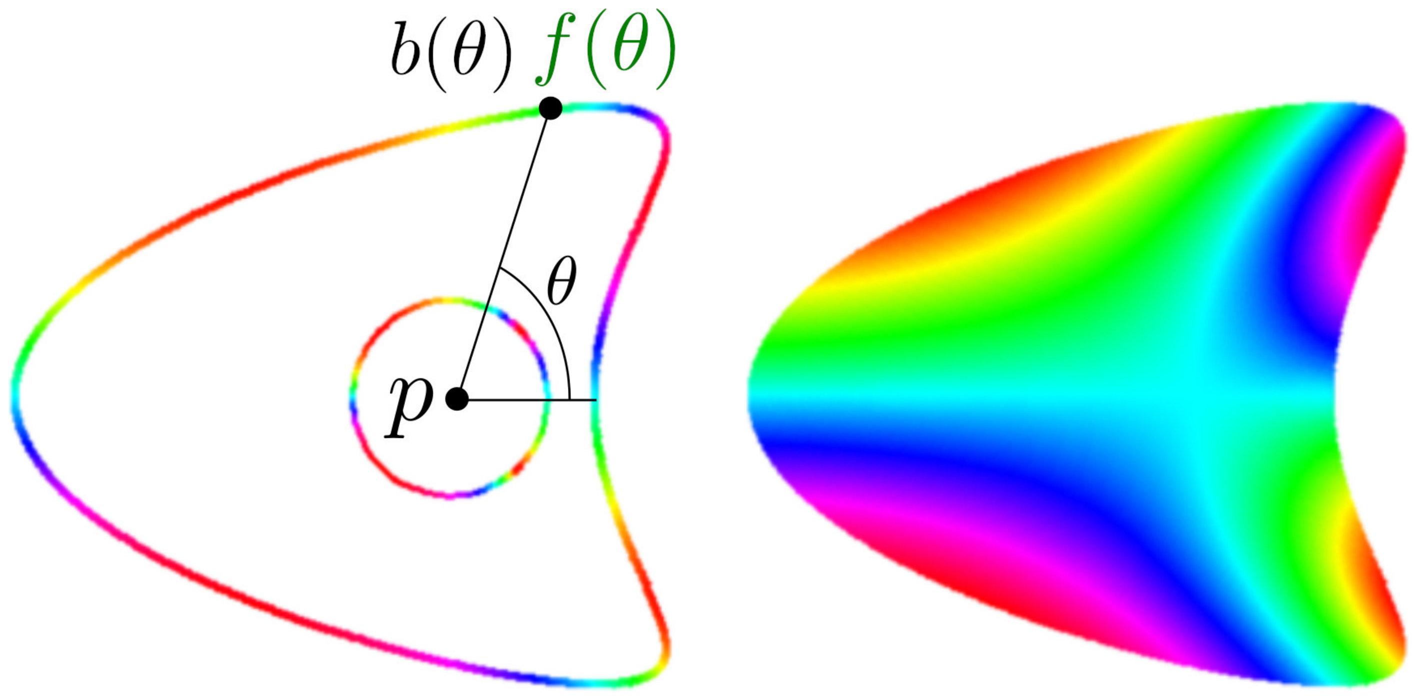
Look for an interpolant.

Given a 2D boundary curve $b(\theta)$, with values $f(\theta)$

Mean-value associated to point p:

$$\hat{f}(p) = \frac{\displaystyle\int_{\theta=0}^{2\pi} \kappa(\theta)\, f(\theta)\, \mathrm{d}\theta}{\displaystyle\int_{\theta=0}^{2\pi} \kappa(\theta)\, \mathrm{d}\theta}, \quad \kappa(\theta) = 1/\|b(\theta) - p\|$$

Averaged value projected around a unit circle, weighted by the inverse distance.

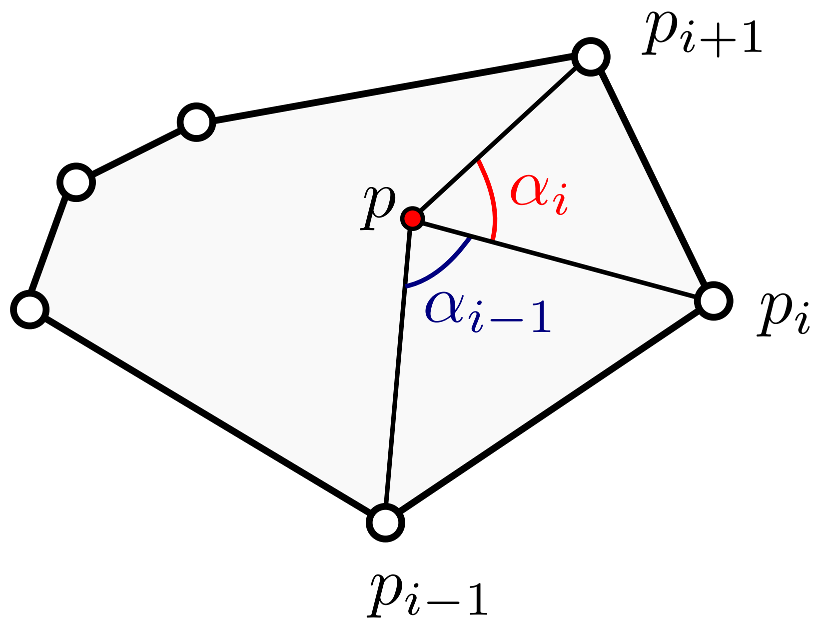


For a piecewise linear boundary

Over the edge $p_i\, p_{i+1}$

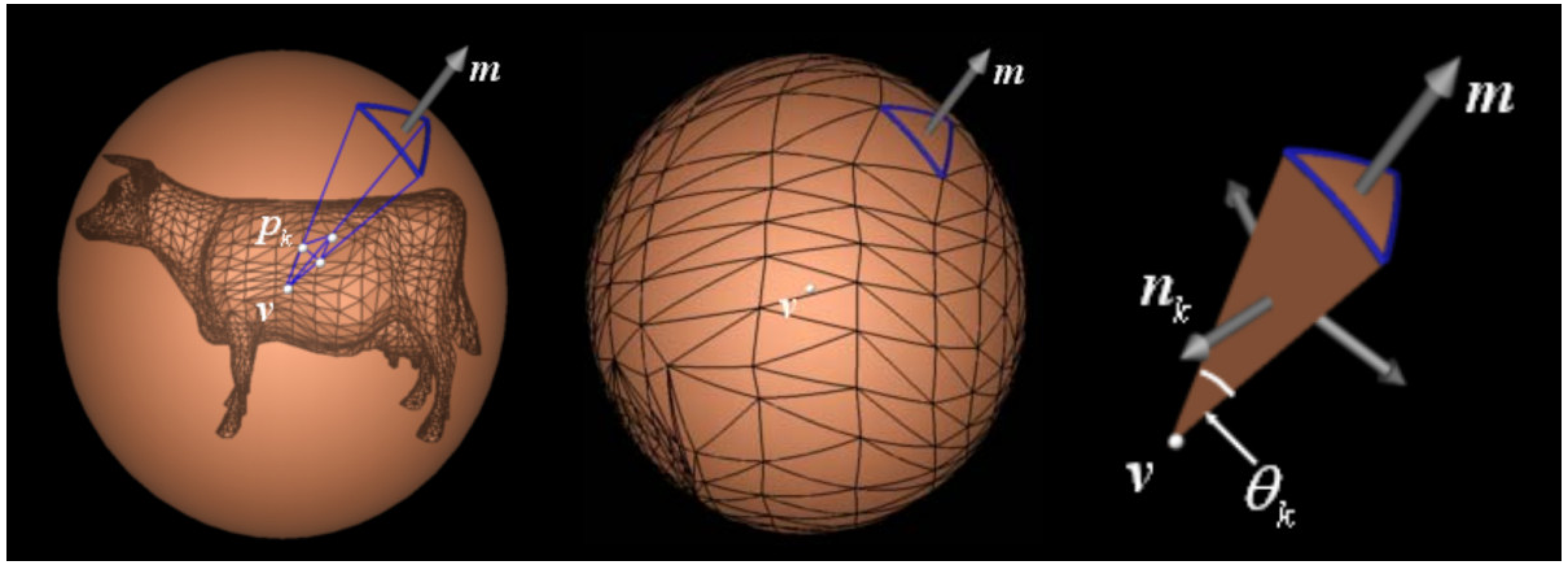
$$\int_{\theta=\theta_1}^{\theta=\theta_2} \kappa(\theta)\, f(\theta)\mathrm{d}\theta = \cdots = \left(\frac{f(p_i)}{\|p_i - p\|} + \frac{f(p_{i+1})}{\|p_{i+1} - p\|} \right) tan \left(\frac{\theta_2 - \theta_1}{2} \right)$$

$$\Rightarrow \omega_i = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|p_i - p\|}$$



Mean Value - Idea in 3D

In 3D: Integrate over a triangulated sphere



[Mean Value Coordinates for Closed Trianular Meshes. T. Ju et al. SIGGRAPH 2005]

// Robust evaluation on a triangular mesh

for each vertex p_j with values f_j

$\quad d_j \leftarrow \|p_j - x\|$

$\quad$ if $d_j < \varepsilon$ return f_j

$\quad u_j \leftarrow (p_j - x)/d_j$

$totalF \leftarrow 0$

$totalW \leftarrow 0$

for each triangle with vertices p_1, p_2, p_3 and values f_1, f_2, f_3

$\quad l_i \leftarrow \|u_{i+1} - u_{i-1}\|$ $\quad$ *// for $i = 1,2,3$*

$\quad \theta_i \leftarrow 2\arcsin[l_i/2]$

$\quad h \leftarrow (\sum \theta_i)/2$

$\quad$ if $\pi - h < \varepsilon$

$\quad\quad$ *// x lies on t, use 2D barycentric coordinates*

$\quad\quad w_i \leftarrow \sin[\theta_i] d_{i-1} d_{i+1}$

$\quad\quad$ return $(\sum w_i f_i)/(\sum w_i)$

$\quad c_i \leftarrow (2\sin[h]\sin[h - \theta_i])/(\sin[\theta_{i+1}]\sin[\theta_{i-1}]) - 1$

$\quad s_i \leftarrow \text{sign}[det[u_1, u_2, u_3]]\sqrt{1 - c_i^2}$

$\quad$ if $\exists i, |s_i| \leq \varepsilon$

$\quad\quad$ *// x lies outside t on the same plane, ignore t*

$\quad\quad$ continue

$\quad w_i \leftarrow (\theta_i - c_{i+1}\theta_{i-1} - c_{i-1}\theta_{i+1})/(d_i \sin[\theta_{i+1}] s_{i-1})$

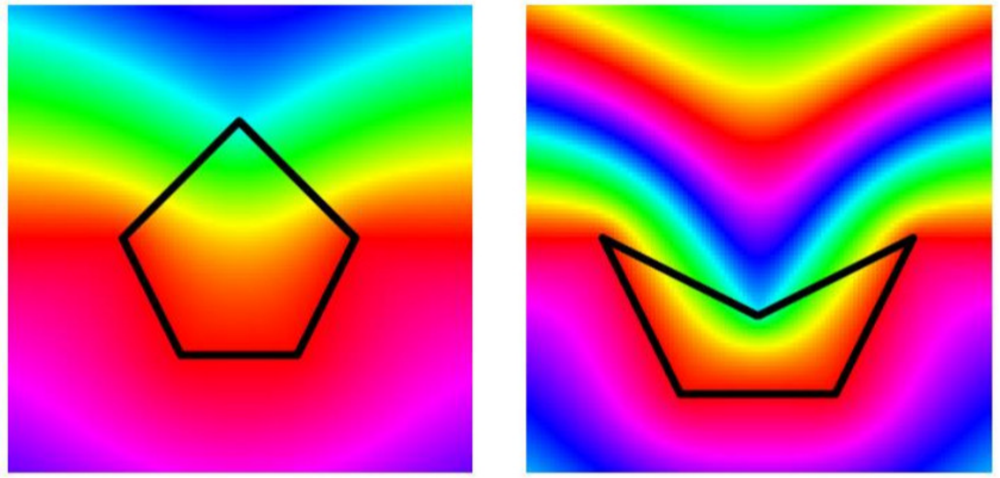
$\quad totalF += \sum w_i f_i$

$\quad totalW += \sum w_i$

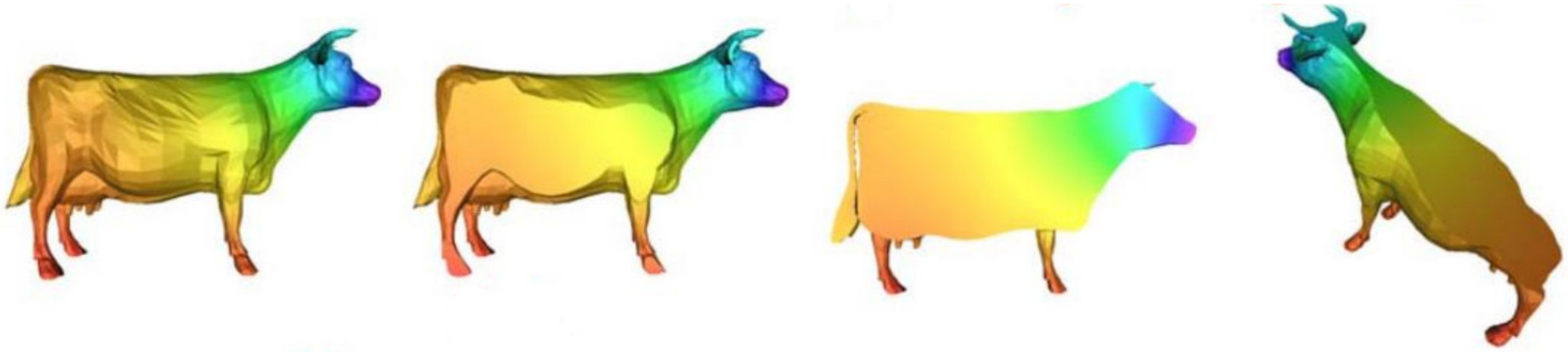
$f_x \leftarrow totalF/totalW$

Mean Value - Results

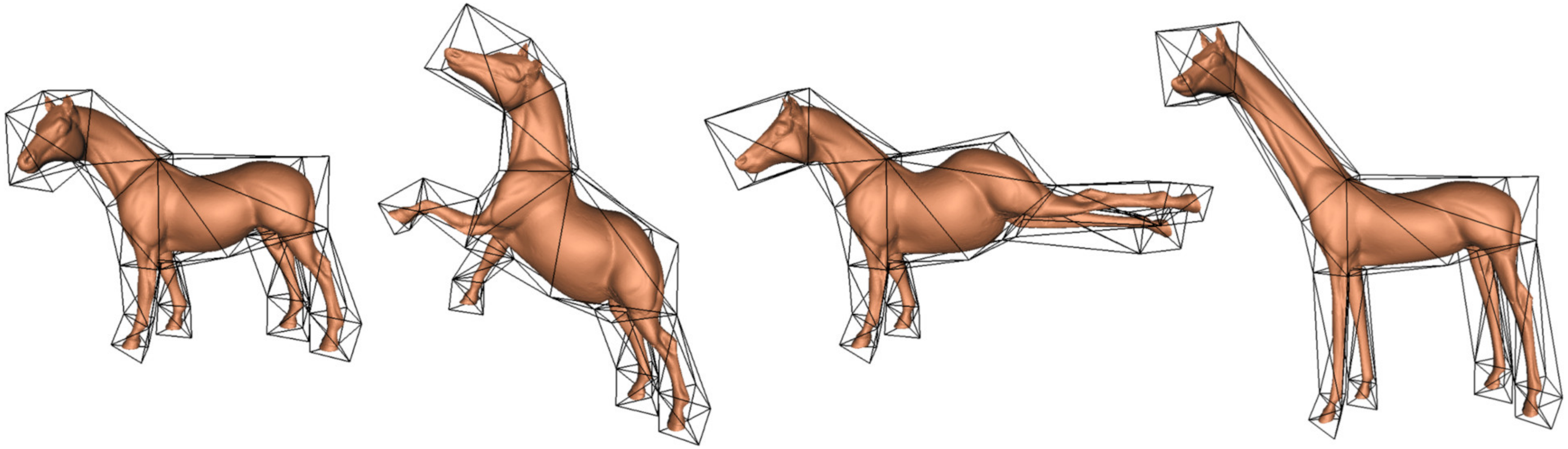
Coordinates in 2D



Coordinates in 3D



Application to shape deformation



Harmonic Coordinates

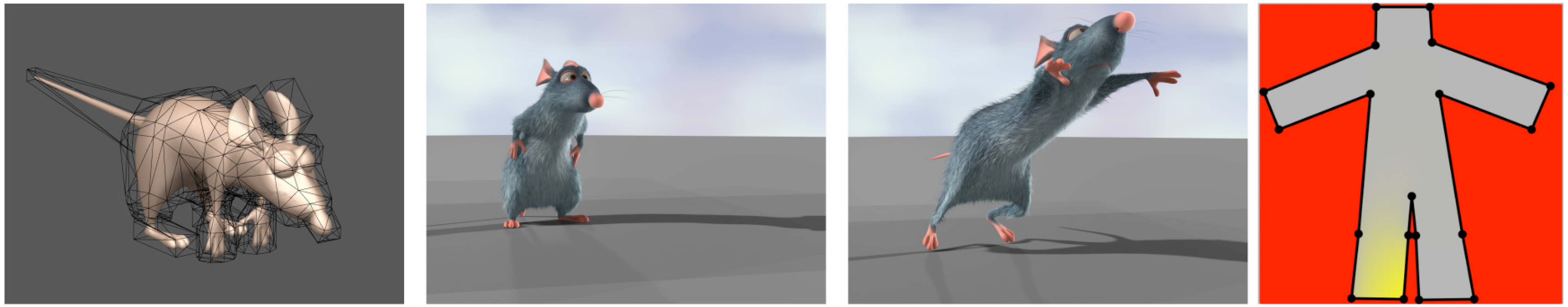
ω solution of the harmonic equation

$$\triangle\omega(p) = 0$$

$$\omega_i(p_j) = \delta_{ij}$$

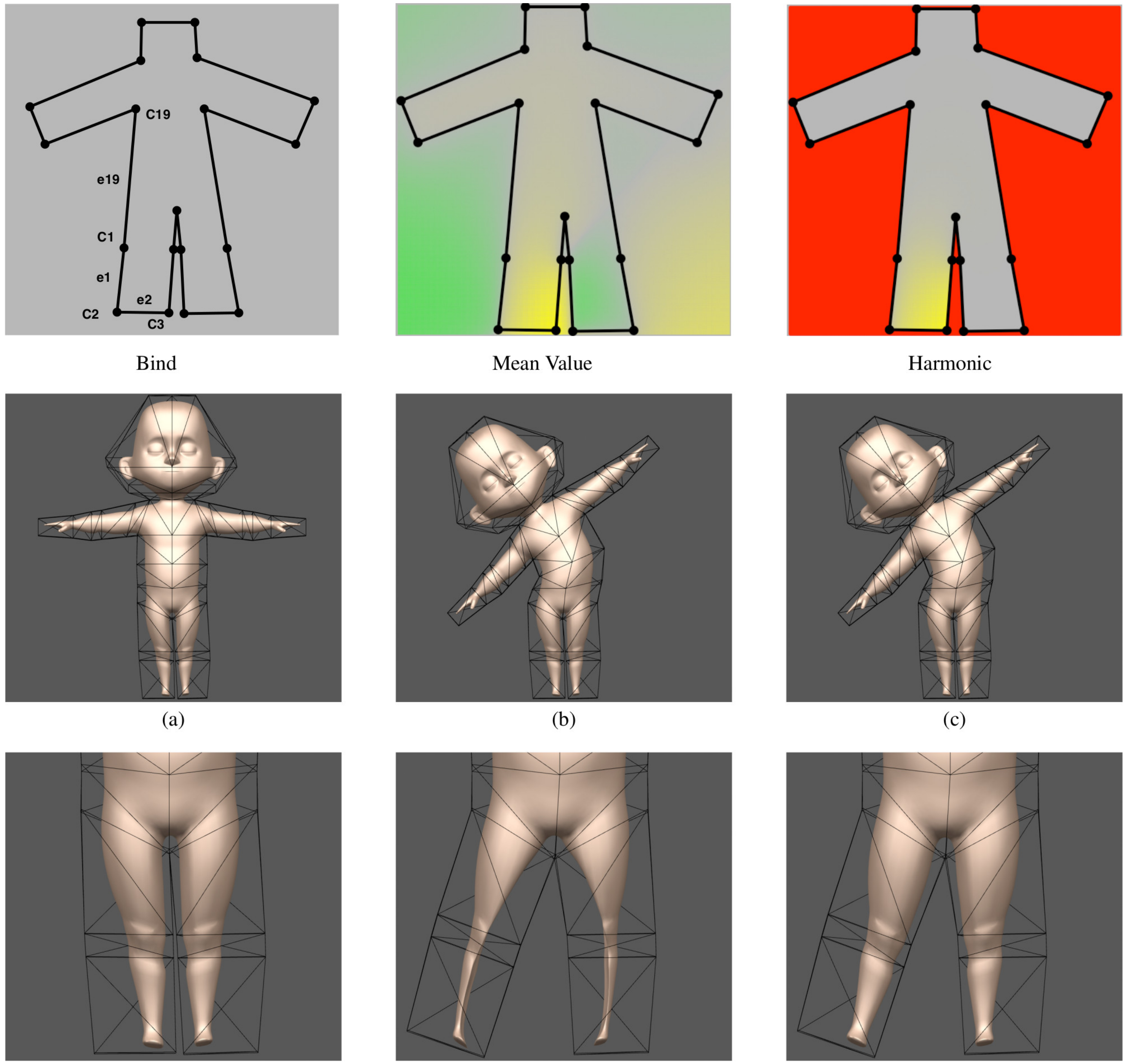
(-) No-closed form solution: requires numerical solver.

(+) Positive weights, even for concave cages



[Pushkar Joshi, Mark Meyer, Tony DeRose. Harmonic Coordinates for Character Articulation. SIGGRAPH 2007.]

Harmonic Coordinates VS Mean Value Coordinates



Extensions

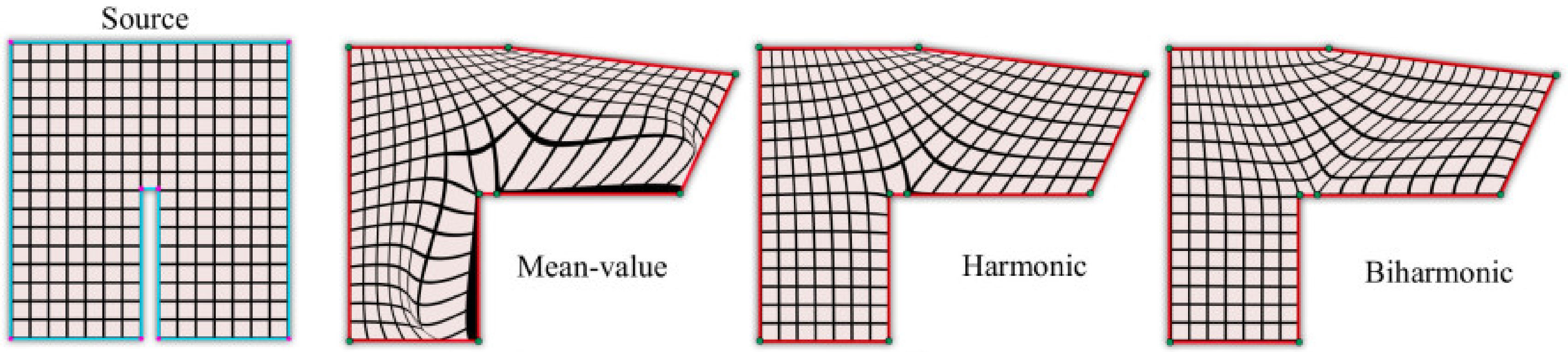
Other coordinates

Green coordinates

[Green Coordinates. Y. Lipman et al. SIGGRAPH 2008]

Bi-Harmonic Coordinates

[Biharmonic Coordinates. O. Weber et al. CGF 2012]



Multi-cage

*[*Cages: A multilevel, multi-cage-based system for mesh deformation. F. Garcia et al. ACM TOG 2013]*

MVC on quad meshes

[Mean value coordinates for quad cages in 3D. J.-M. Thiery et al. SIGGRAPH Asia 2018]

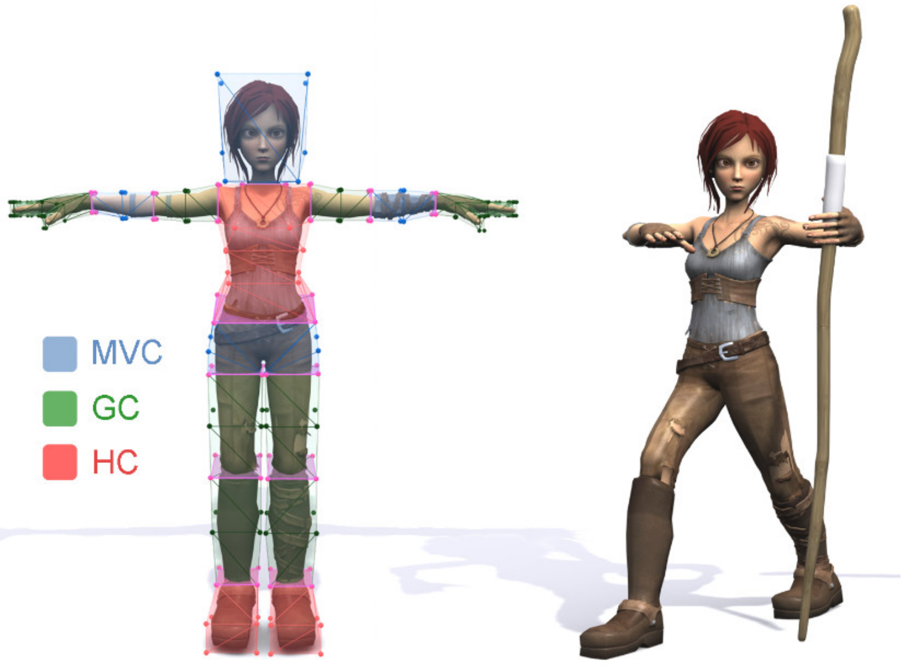
Cage creation

Building good cage is non-trivial task

Multiple works to avoid manual creation

[Skeleton Based Cage Generation Guided by Harmonic Fields. S. Casti et al. Computer & Graphics 2019]

[CageR: Cage-based Reverse Engineering of Animated 3D Shapes. J.-M. Thiery et al. CGF 2012.]



Vector field-based deformation

Smooth vector field (or velocity) $u : (x, y, z) \to (u_x(x, y, z), u_y(x, y, z), u_z(x, y, z))$

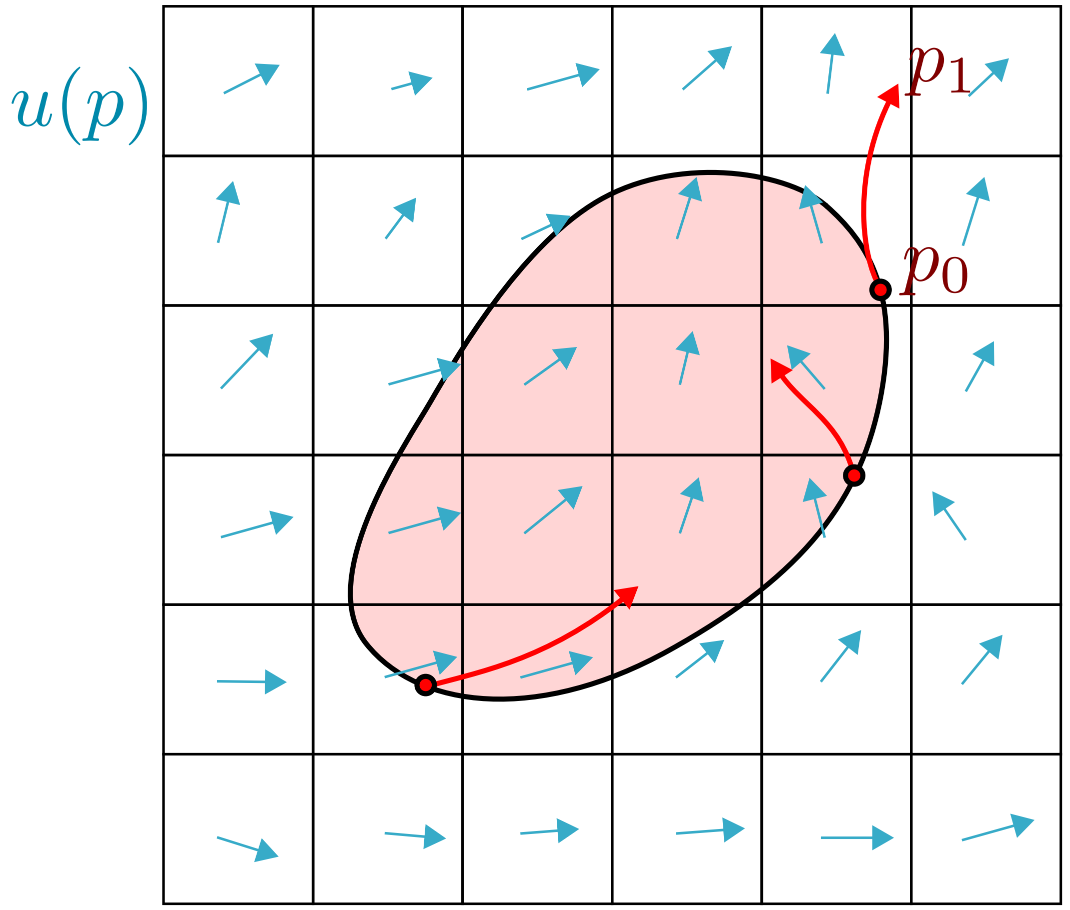
At arbitrary time t, $\dot{p}(t) = u(p(t))$

Applying deformation = integrate vertex position along streamlines

$$p_1 = p_0 + \int_t u(p(t))\,\mathrm{d}t$$

At a given time t, the instantaneous deformation at position p is

$$f(p, t) = p + u(p, t) \to f = I + u$$



(+) If $\|u\| \neq 0$ (and $\neq$ singularity) no-intersection

Streamlines do not intersect

(-) Arbitrary vector fields hard to define and control.

Note: All physically-based deformation can be seen as vector field deformations

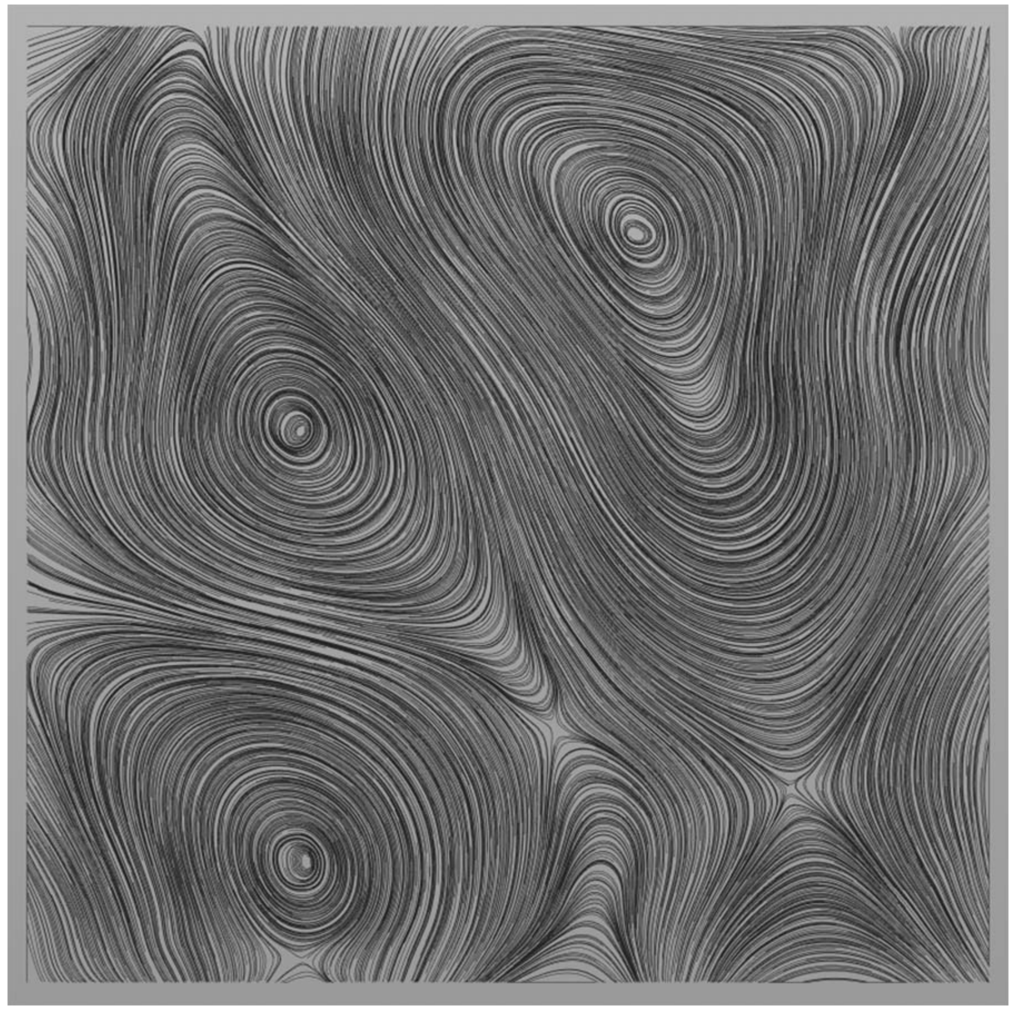
Divergence Free Vector Field

Divergence of u: $div(u) = \nabla \cdot u = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial u_y} + \frac{\partial u_z}{\partial u_z}$

Divergence free / solenoidal field : $div(u) = 0$

Ex. Magnetic fields, incompressible fluids velocity, are divergent free

$\Rightarrow$ Volume preserving deformation



Dem. (in 2D)

$f = I + u$

$$\Rightarrow J_f = \begin{pmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} \end{pmatrix} = \begin{pmatrix} 1 + \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} \\ \frac{\partial u_y}{\partial x} & 1 + \frac{\partial u_y}{\partial y} \end{pmatrix}$$

Volume preserving deformation $det(J_f) = 1$

$$\Rightarrow (1 + \frac{\partial u_x}{\partial x}) \left(1 + \frac{\partial u_y}{\partial y}\right) - \frac{\partial u_x}{\partial y} \frac{\partial u_y}{\partial x} = 1$$

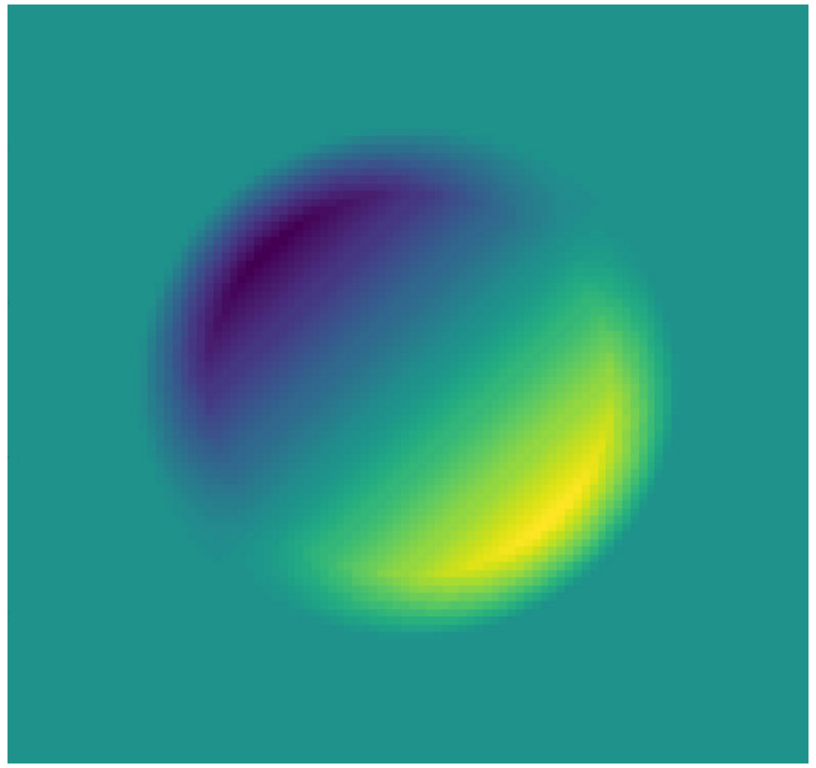
$$\Rightarrow \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} = 0$$

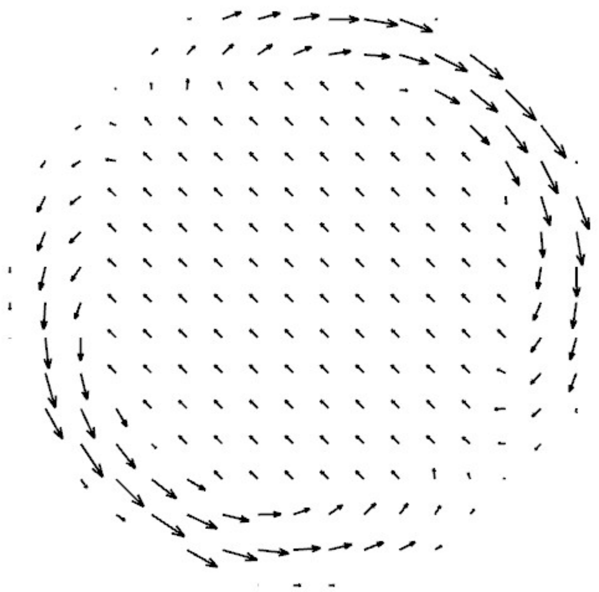
Building divergence free vector field

Example: Define 2 scalar fields (a, b)

Build vector field $u = \nabla a \times \nabla b$

$$div(u) = div(\nabla a \times \nabla b)$$
$$div(u) = \underbrace{curl(\nabla a)}_{=0} \cdot \nabla b - \nabla a \cdot \underbrace{curl(\nabla b)}_{=0}$$
$$div(u) = 0 \Rightarrow \text{divergence free vector field}$$



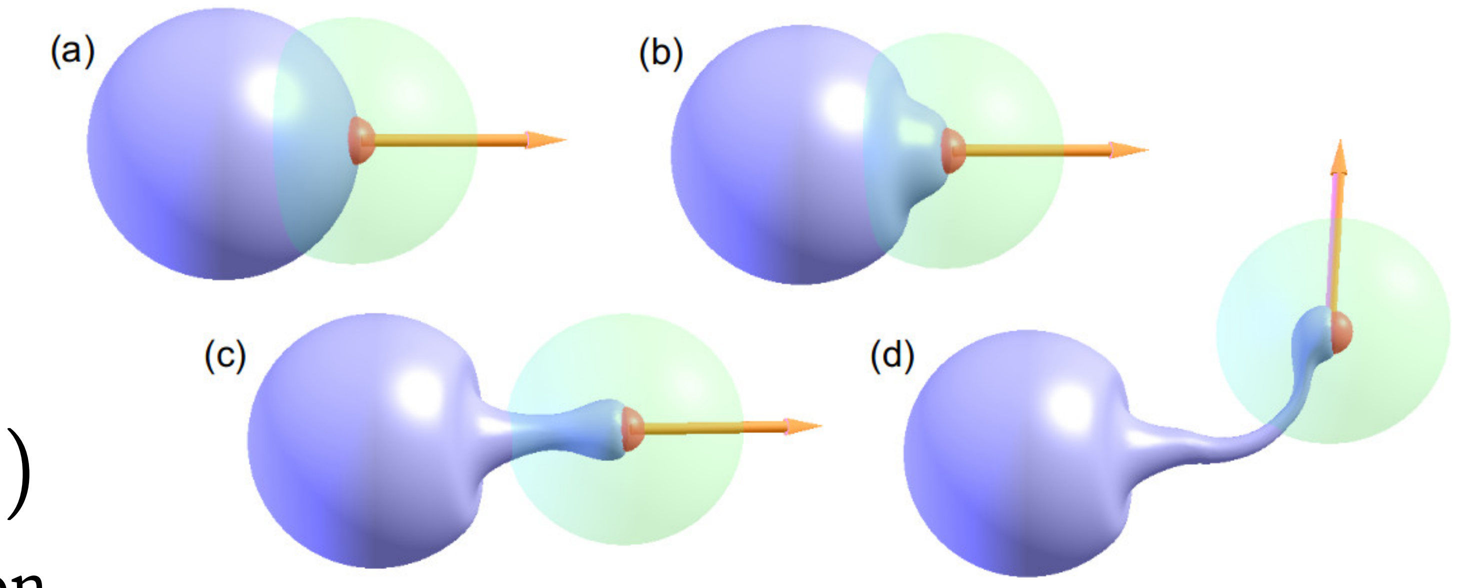


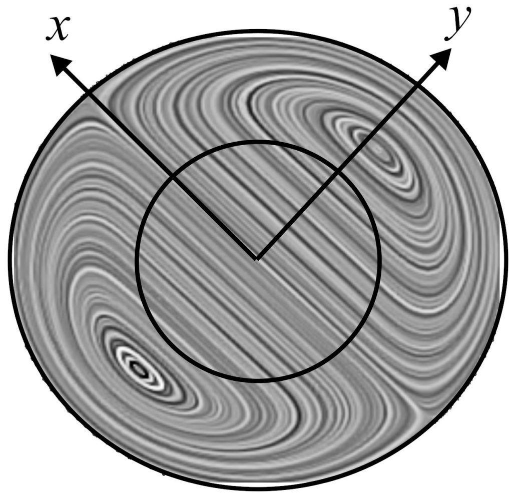
Divergence free vector field example

Example from [Vector Field Based Shape Deformations. W. von Funck et al. SIGGRAPH 2006]

$$a(p) = \begin{cases} a_0(p) & \|p\| < r_1 \\ (1-\alpha(p))\, a_0(p) & r_1 \le \|p\| \le r_2 \\ 0 & otherwise \end{cases}$$

Similarily with $b(p)$



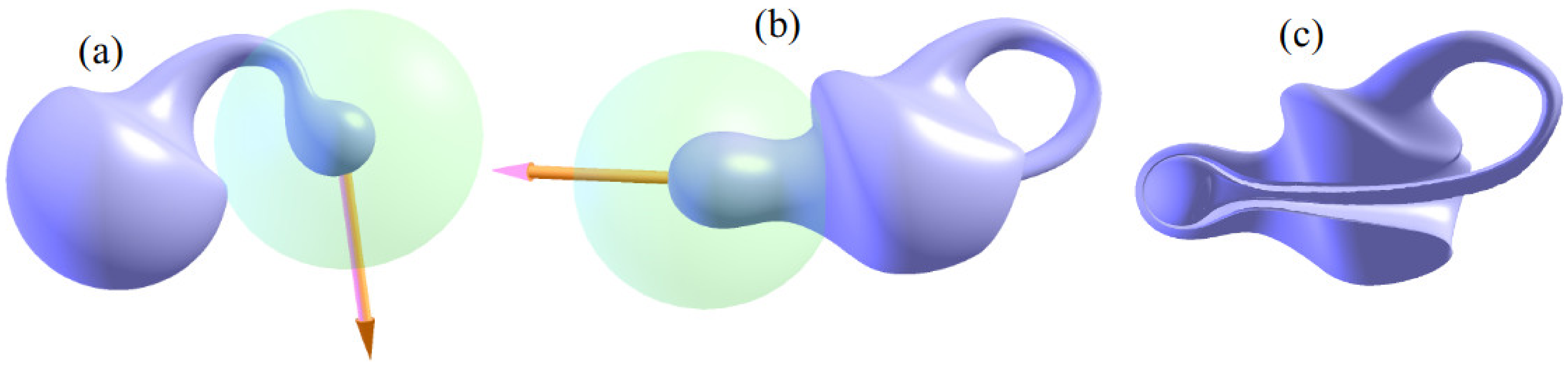


Translation: $a_0(p) = v_1 \cdot (p - p_0), b_0(p) = v_2 \cdot (p - p_0)$

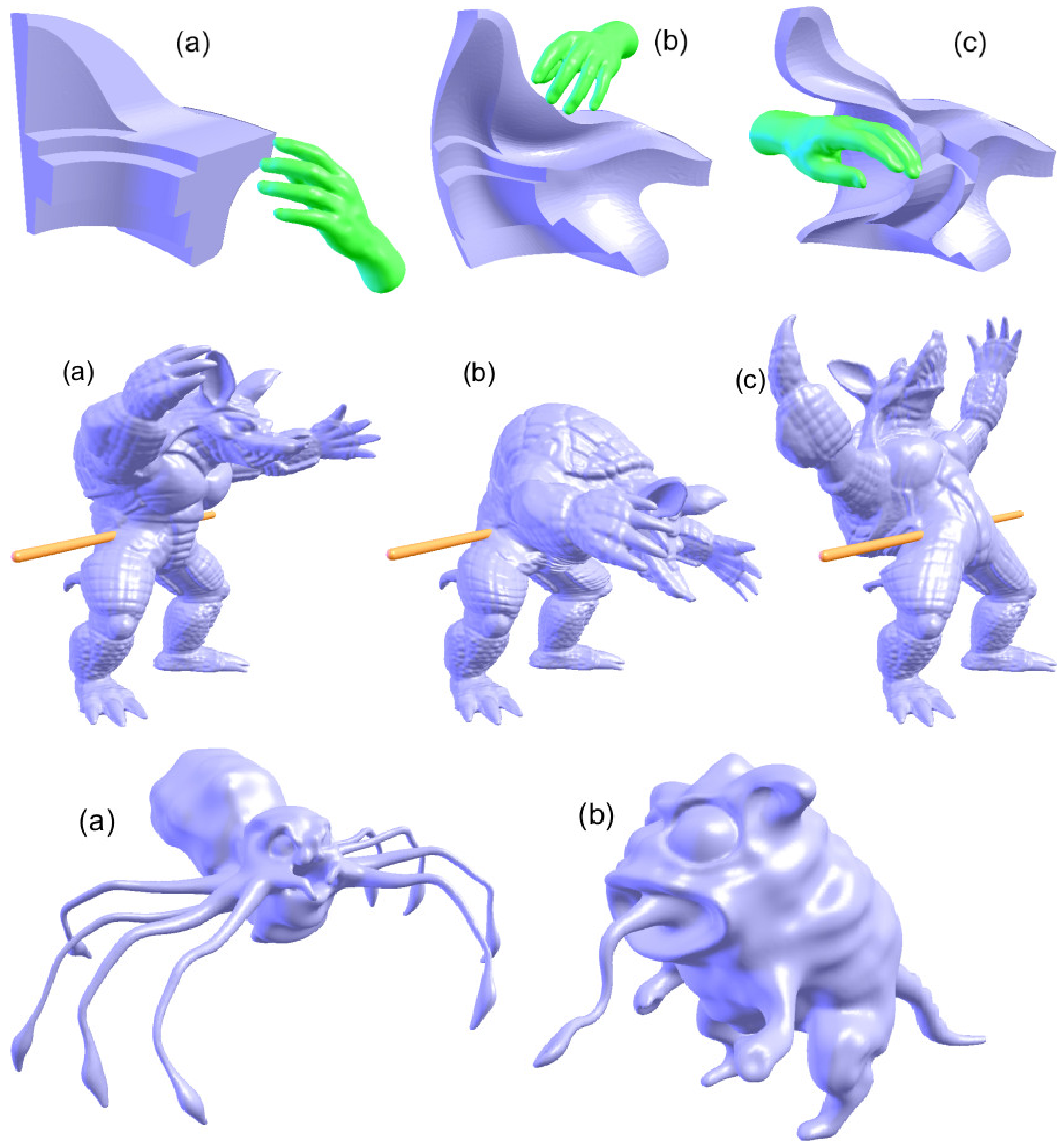
v_1, v_2 arbitrary orthogonal unit vector to the direction of translation.

Bending: $a_0(p) = a \cdot (p - p_0), b_0 = \|a \times (p - p_0)\|^2$

a rotation axis.



Example of results



Case of implicit surfaces

Undeformed implicit surface $S = \{p \in \mathbb{R}^3 \mid \; g(p) = 0\}$

Given a vector field u

The implicit surface S' advected by u is defined as

$$\frac{\partial g}{\partial t} + u \cdot \nabla g = 0$$

Note: Only velocity normal to the surface changes the shape

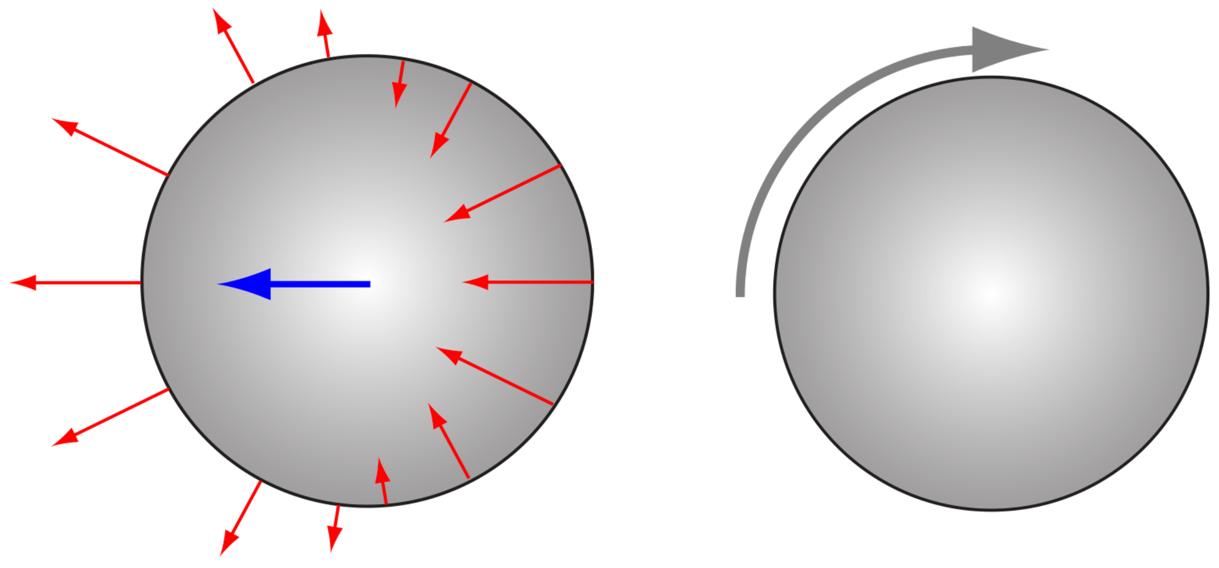
g is a space-time function $g(p, t)$

$g(p + u\mathrm{d}t, t + \mathrm{d}t) = g(p, t) = 0$

$\Rightarrow$ The differential $\mathrm{d}g = 0$

$\Rightarrow \frac{\partial g}{\partial t}\,\mathrm{d}t + \nabla g \cdot \mathrm{d}p = 0$

$\Rightarrow \frac{\partial g}{\partial t} + \nabla g \cdot \underbrace{\frac{\mathrm{d}p}{\mathrm{d}t}}_{u} = 0$



[On the Velocity of an Implicit Surface]