Title
Making green goop from polygon soup

Author
O'Brien, James F

Publication Date
2005-06-13

DOI
10.1145/1060244.1060246

Peer reviewed

Building Surfaces from Polygons and Tracking Surfaces with Polygons

Prof. James F. O'Brien

EECS, Computer Science
University of California, Berkeley

Collaborators

Students at U.C Berkeley

Okan Arikan (w/ DAF)
Adam Bargteil
Bryan Feldman
Tolga Goktekin
Hayley Iben
Adam Kirk
Bryan Klingner
Ravikrishna Kolluri (w/ JRS)
Chen Shen

Faculty at U.C Berkeley

Erik Demaine (MIT)
David Forsyth
Jonathan Shewchuk
John Strain (Mathematics)

Berkeley Computer Animation & Modeling

Overview

- Building implicit surfaces from “polygon soup”
- Tracking surfaces using polygonal surfaces
- Some thoughts tying the two together

Implicit Moving Least-Square

- Repairing defective polygon models
  - Holes, gaps, T-junctions, self-intersections, non-manifold structures
- Testing interior/exterior points
- Preprocessing for rapid prototyping machines
- Generating simulation envelopes

Shen, O'Brien, Shewchuk, *SIGGRAPH 2004*

Implicit Moving Least Squares Surfaces (ILMS)

- True normal constraints
  - No undesirable oscillatory behavior or spurious surfaces
- Integrated constraints over polygons
  - Avoids dimples and bumps
- Adjustment procedure
  - Tight fit, completely enclosed
- Hierarchical fast evaluation

Background

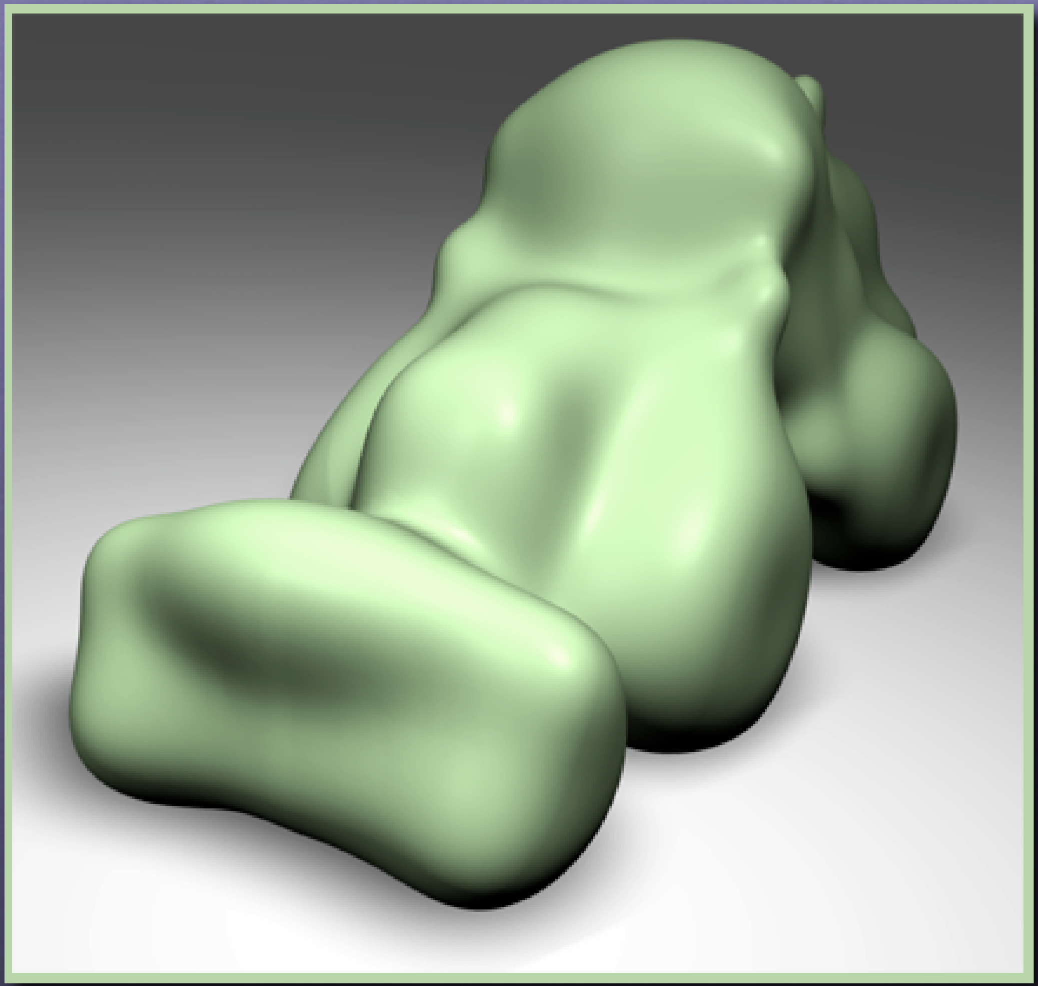
- Implicit Partition-of-Unity
  - Ohtake *et al.* 2003
- Moving Least Squares projection methods
  - Alexa *et al.* 2001, 2003, Fleishman *et al.* 2003, Amenta *et al.* 2004
- Other implicit techniques
- Delaunay/Voronoi based methods
- Other model fixing/smoothing methods
- Please see paper for details and others...

Example

Input polygons

Interpolating Implicit Surface

Example

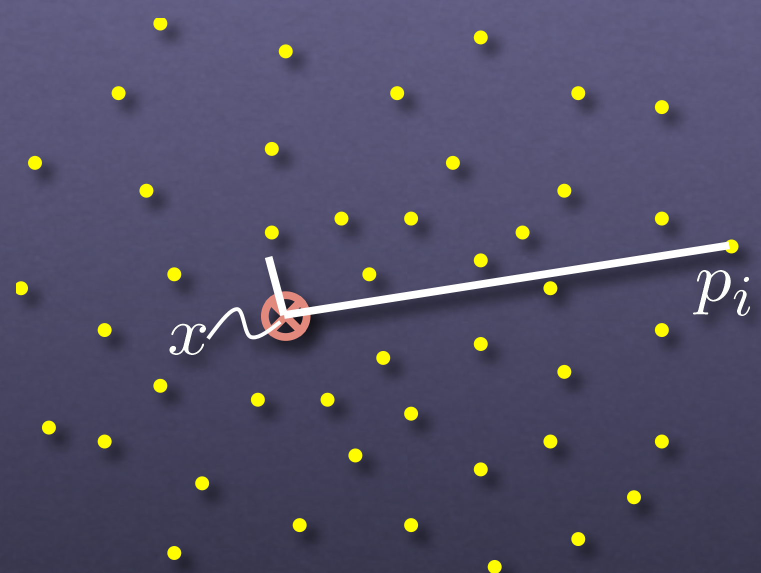


Approximating Implicit Surface

MLS Interpolation / Approximation

- Standard Least Square

$$\begin{bmatrix} b^\top(p_1) \\ \vdots \\ b^\top(p_N) \end{bmatrix} c = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix}$$

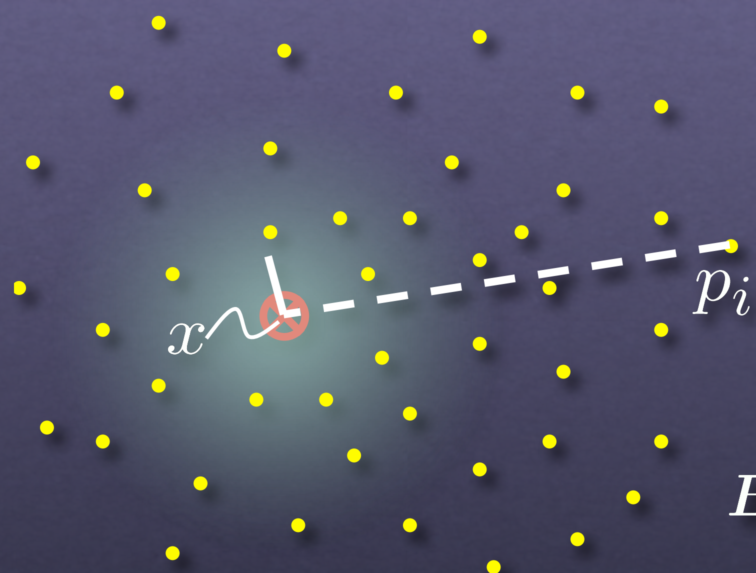


$$B^\top B\, c = B^\top \phi$$

MLS Interpolation / Approximation

- Moving Least Square

$$\begin{bmatrix} w(\boldsymbol{x},\boldsymbol{p}_1) & & \\ & \ddots & \\ & & w(\boldsymbol{x},\boldsymbol{p}_N) \end{bmatrix} \begin{bmatrix} \boldsymbol{b}^\mathsf{T}(\boldsymbol{p}_1) \\ \vdots \\ \boldsymbol{b}^\mathsf{T}(\boldsymbol{p}_N) \end{bmatrix} \boldsymbol{c} = \begin{bmatrix} w(\boldsymbol{x},\boldsymbol{p}_1) & & \\ & \ddots & \\ & & w(\boldsymbol{x},\boldsymbol{p}_N) \end{bmatrix} \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix}$$

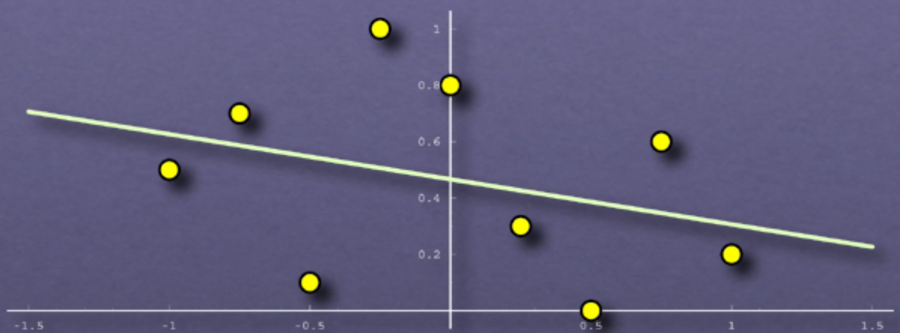


$$w(r) = \frac{1}{\left(r^2 + \epsilon^2\right)}$$

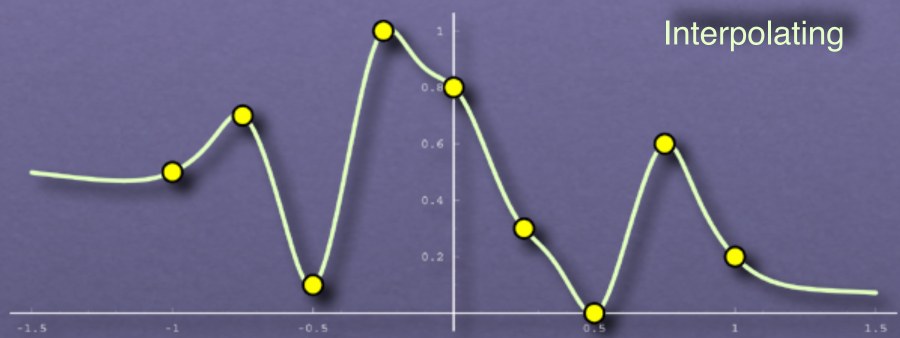
$$\boldsymbol{B}^\mathsf{T} \; (\boldsymbol{W}(\boldsymbol{x}))^2 \; \boldsymbol{B} \; \boldsymbol{c}(\boldsymbol{x}) = \boldsymbol{B}^\mathsf{T} \; (\boldsymbol{W}(\boldsymbol{x}))^2 \; \boldsymbol{\phi}$$

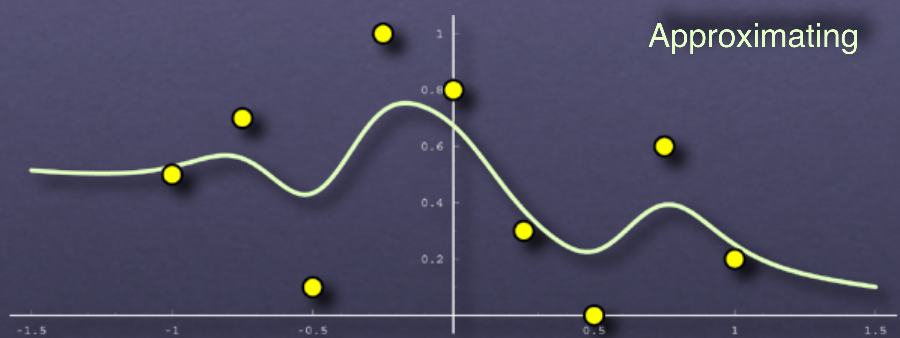
MLS Interpolation / Approximation

Least Square

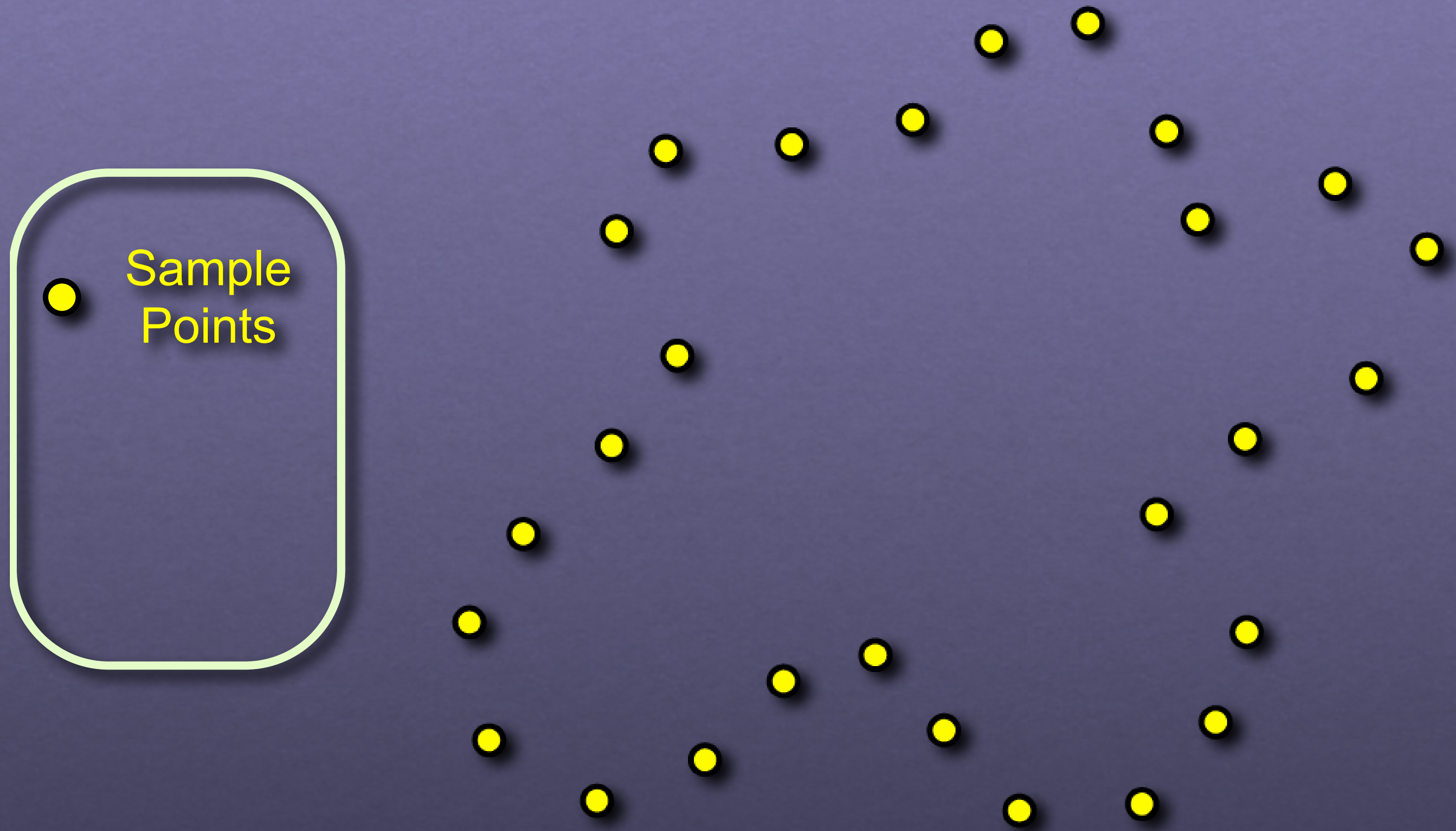


Moving Least Square

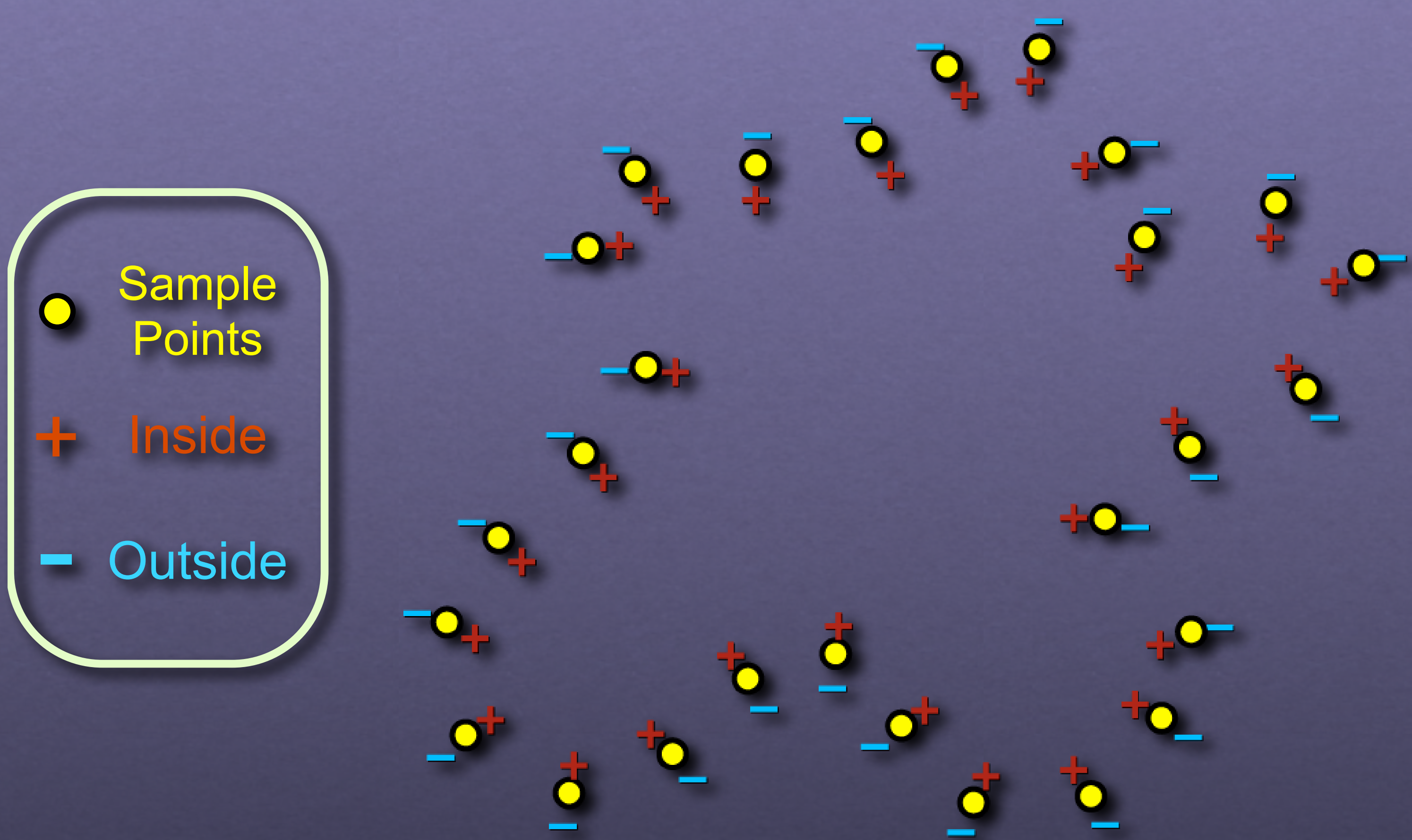




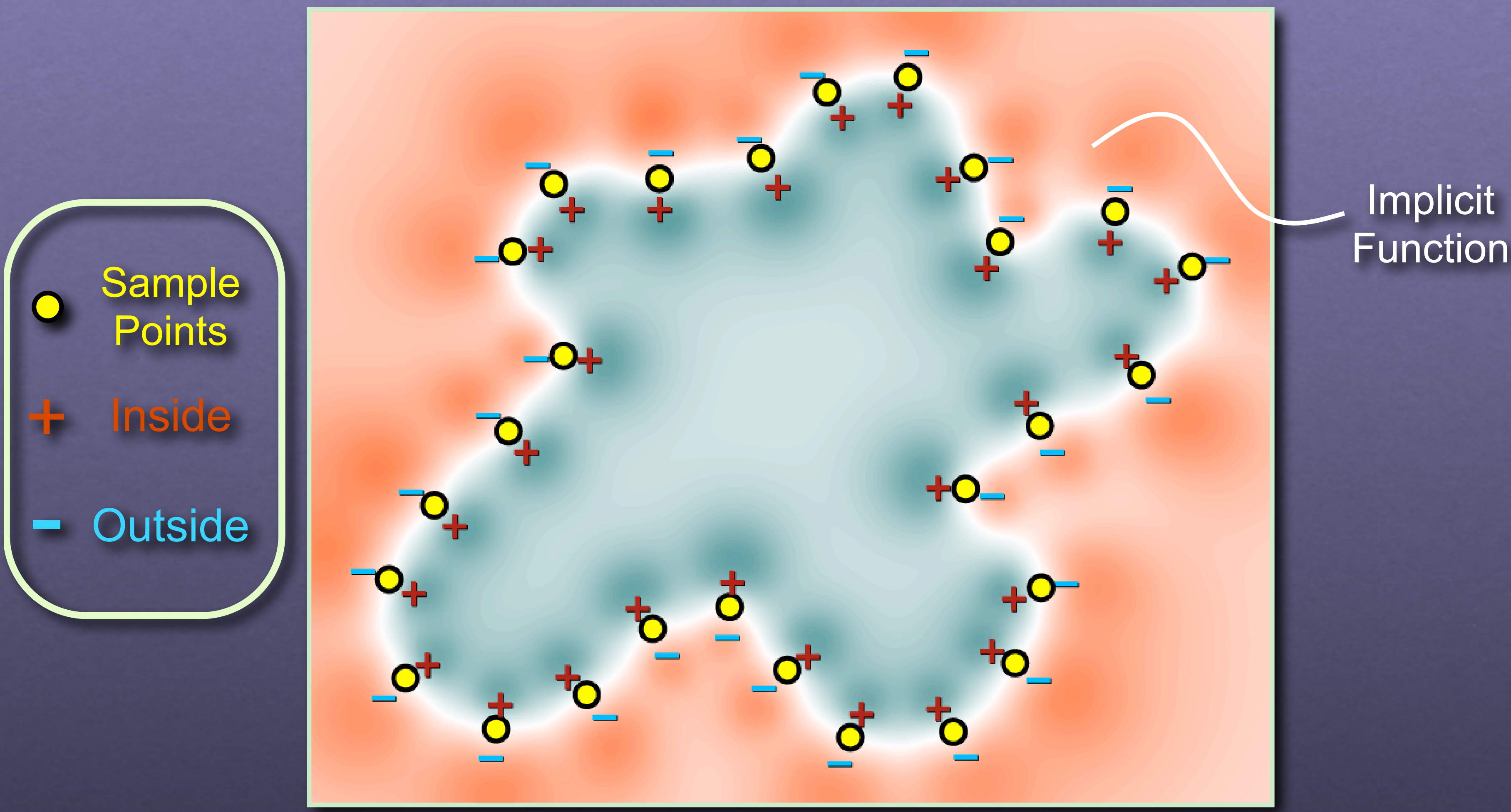
Implicit MLS Surfaces (Or curves in 2D)



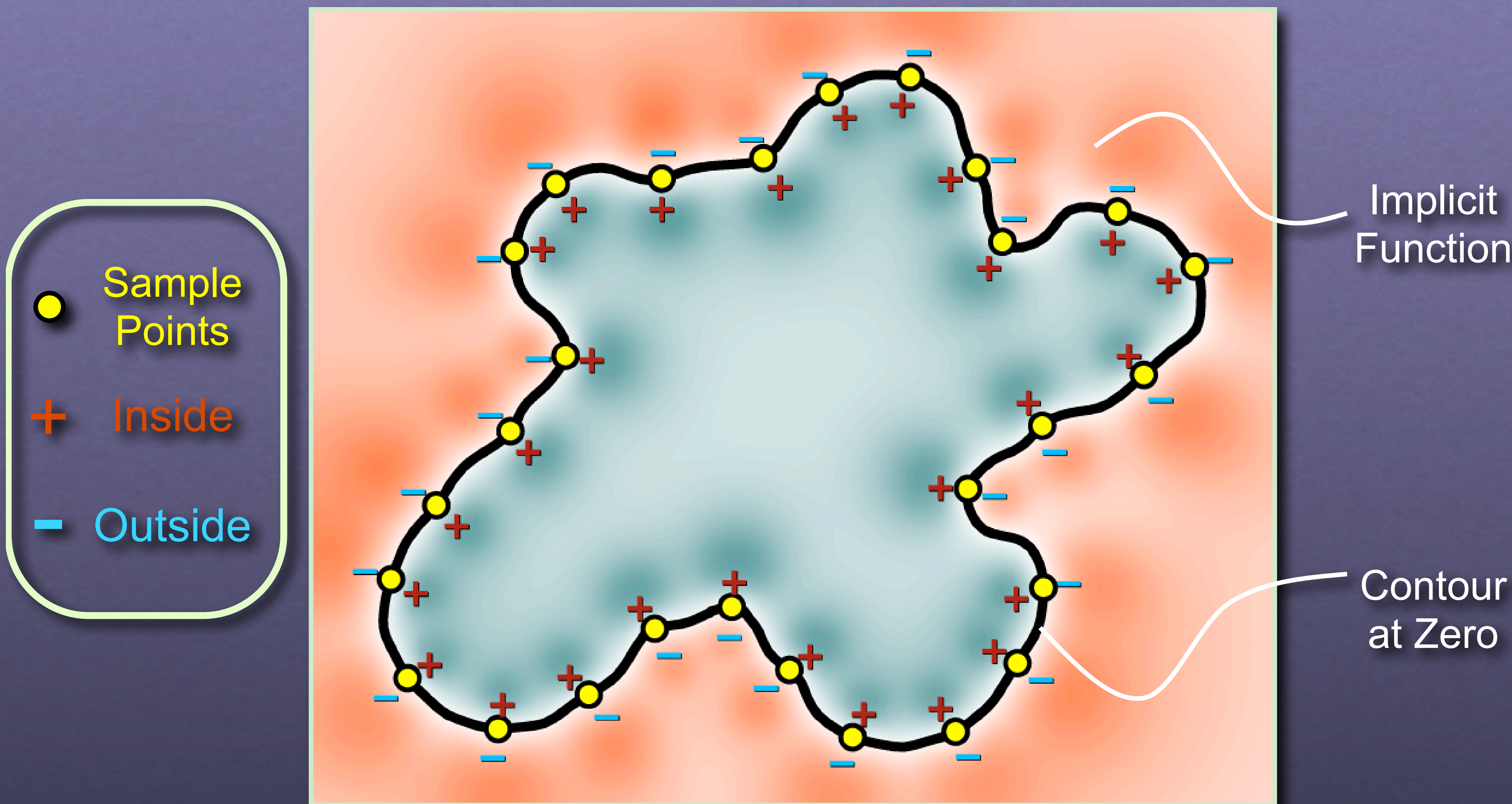
Implicit MLS Surfaces (Or curves in 2D)



Implicit MLS Surfaces (Or curves in 2D)

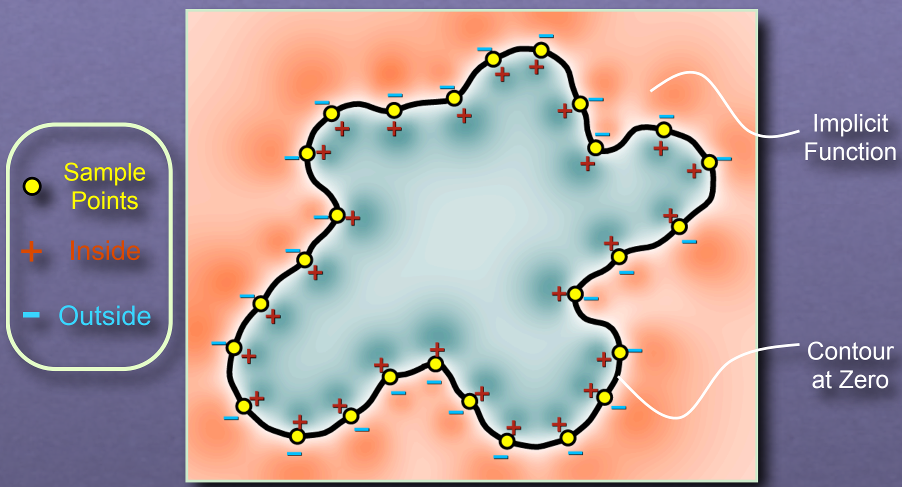


Implicit MLS Surfaces (Or curves in 2D)

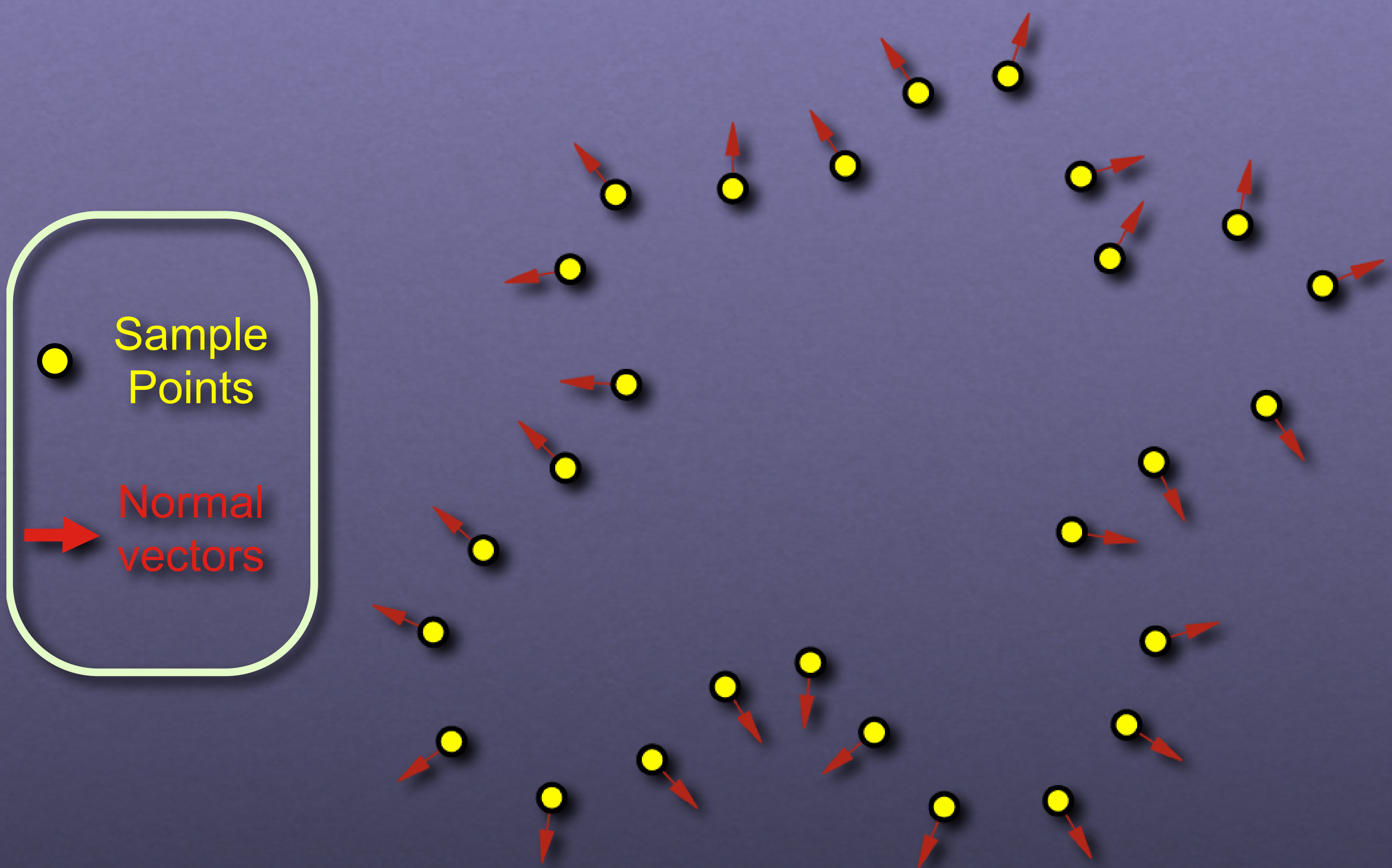


Ugly bumps

Implicit MLS Surfaces (Or curves in 2D)

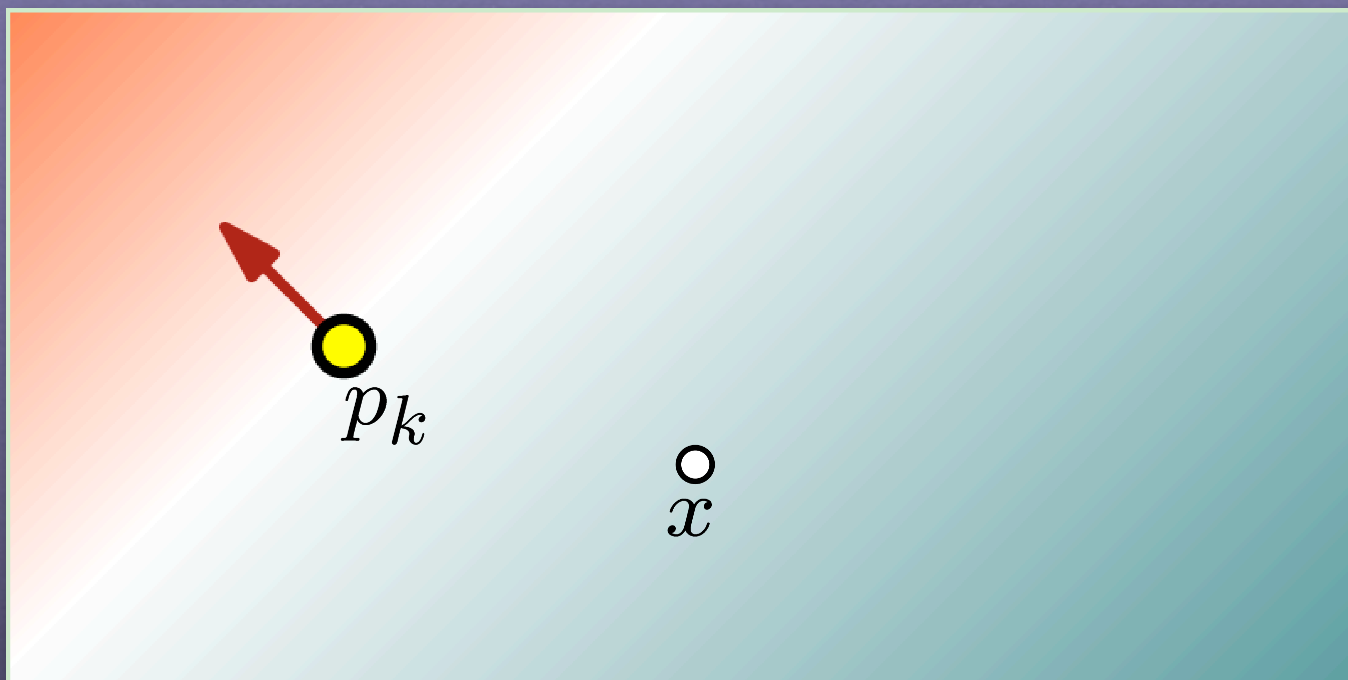


Implicit MLS Surfaces (Or curves in 2D)



Implicit MLS Surfaces (Or curves in 2D)

Point functions



$$S_k(\boldsymbol{x}) = \phi_k + (\boldsymbol{x} - \boldsymbol{p}_k)^\mathsf{T}\,\hat{\boldsymbol{n}}_k$$
$$= \psi_{0k} + \psi_{xk}\,x + \psi_{yk}\,y + \psi_{zk}\,z$$

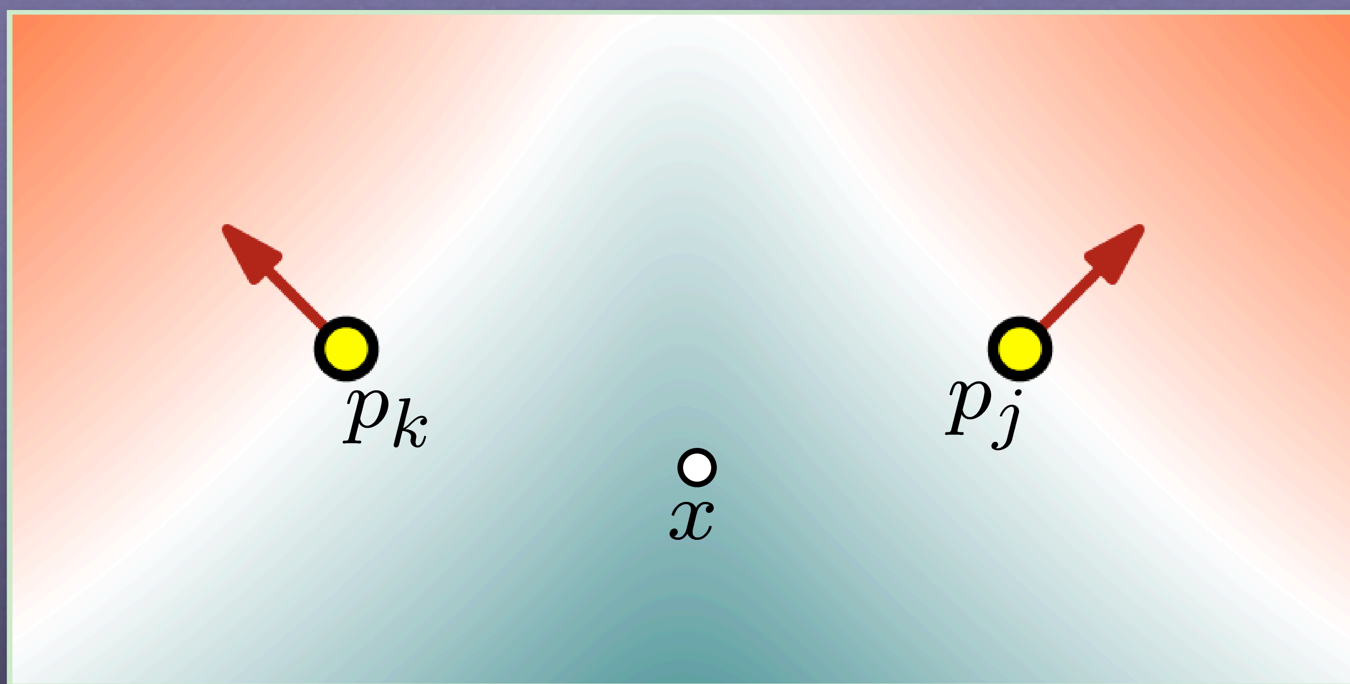
Implicit MLS Surfaces (Or curves in 2D)

Point functions

$$
\begin{aligned}
S_j(\boldsymbol{x}) &= \phi_k + (\boldsymbol{x} - \boldsymbol{p}_k)^{\mathsf{T}}\, \hat{\boldsymbol{n}}_k \\
&= \psi_{0k} + \psi_{xk}\, x + \psi_{yk}\, y + \psi_{zk}\, z
\end{aligned}
$$

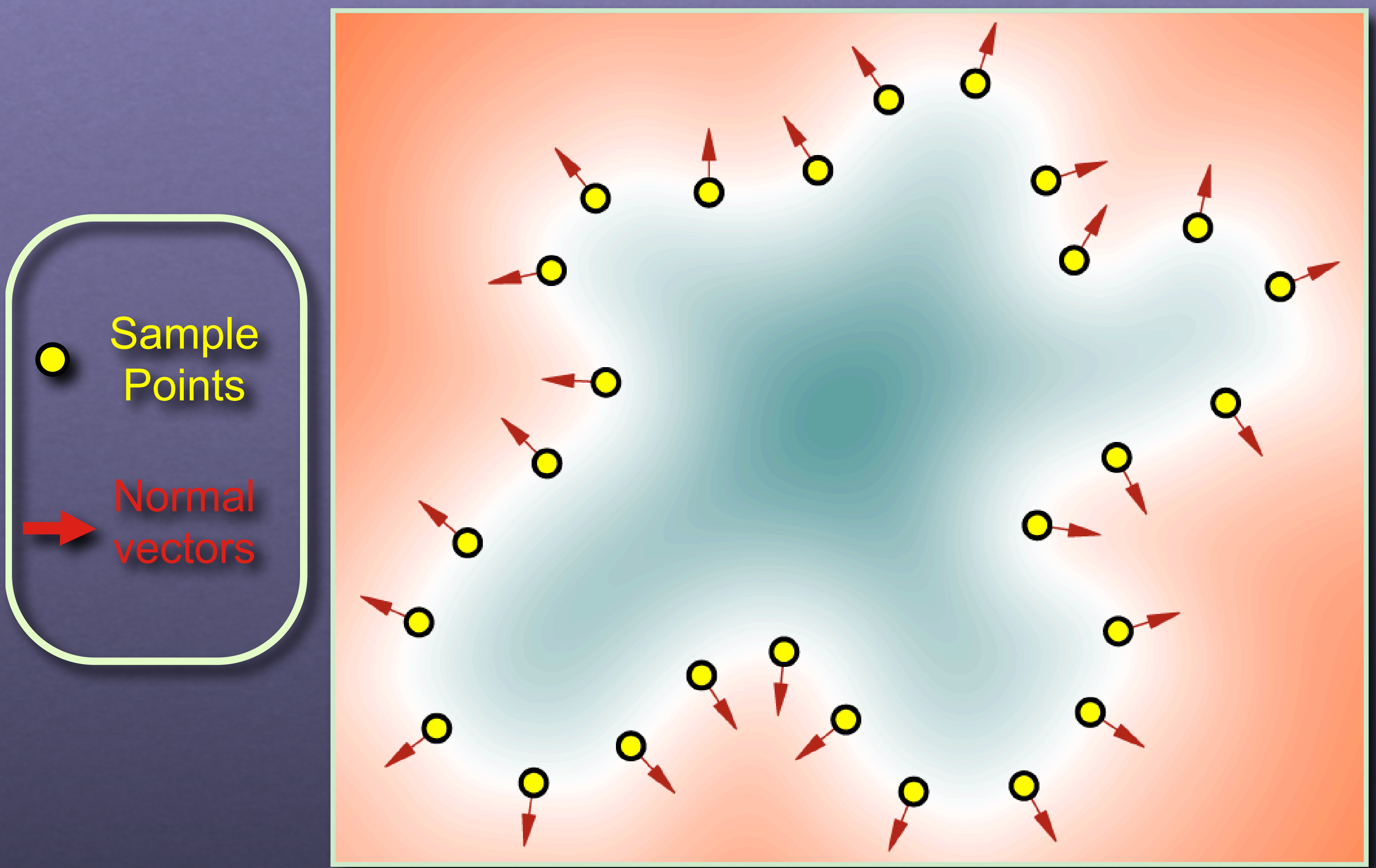
Implicit MLS Surfaces (Or curves in 2D)

Point functions

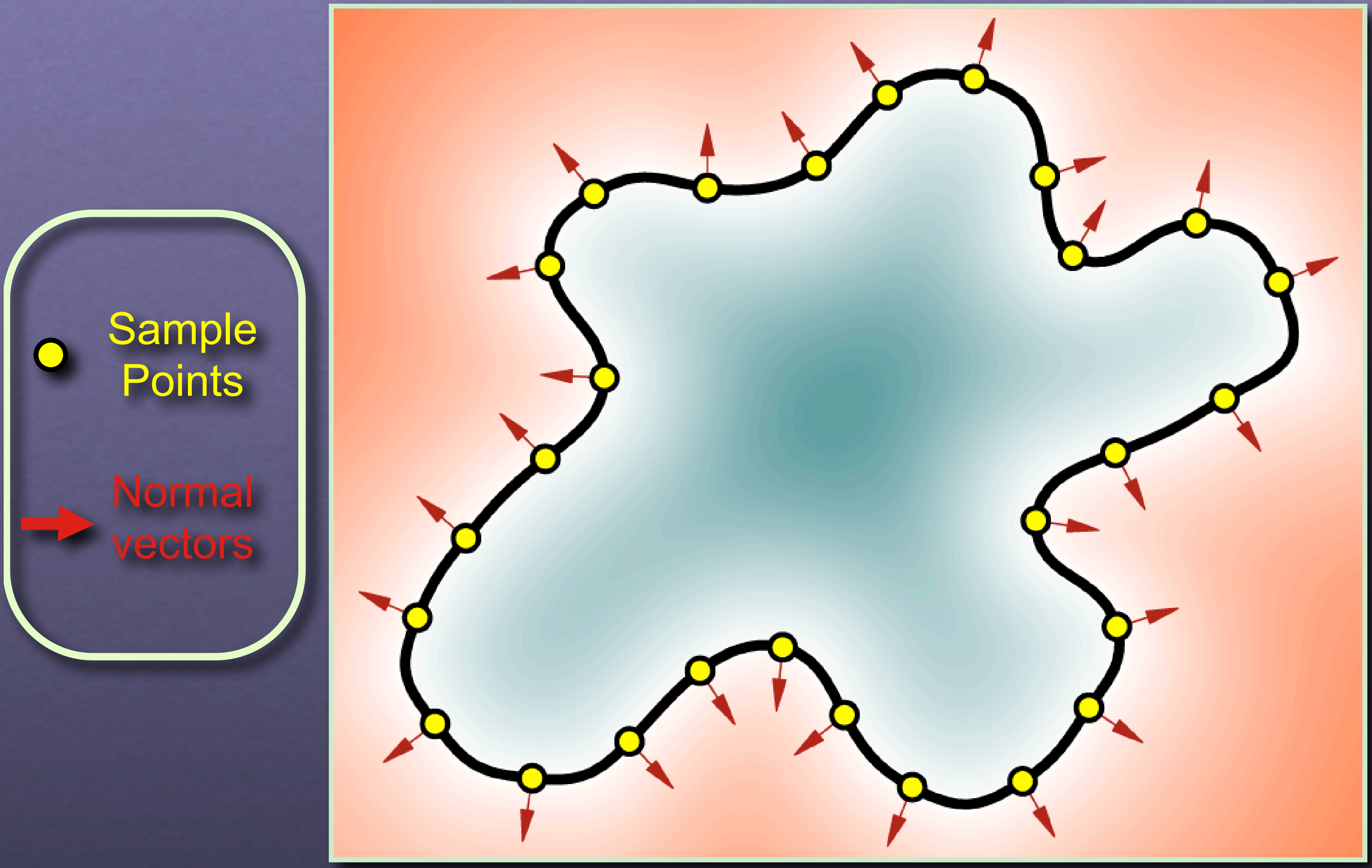


$$\begin{bmatrix} w(\boldsymbol{x}, \boldsymbol{p}_1) \\ \vdots \\ w(\boldsymbol{x}, \boldsymbol{p}_i) \end{bmatrix} c_1 = \begin{bmatrix} w(\boldsymbol{x}, \boldsymbol{p}_1) & & \\ & \ddots & \\ & & w(\boldsymbol{x}, \boldsymbol{p}_N) \end{bmatrix} \begin{bmatrix} S_1(\boldsymbol{x}) \\ \vdots \\ S_N(\boldsymbol{x}) \end{bmatrix}$$

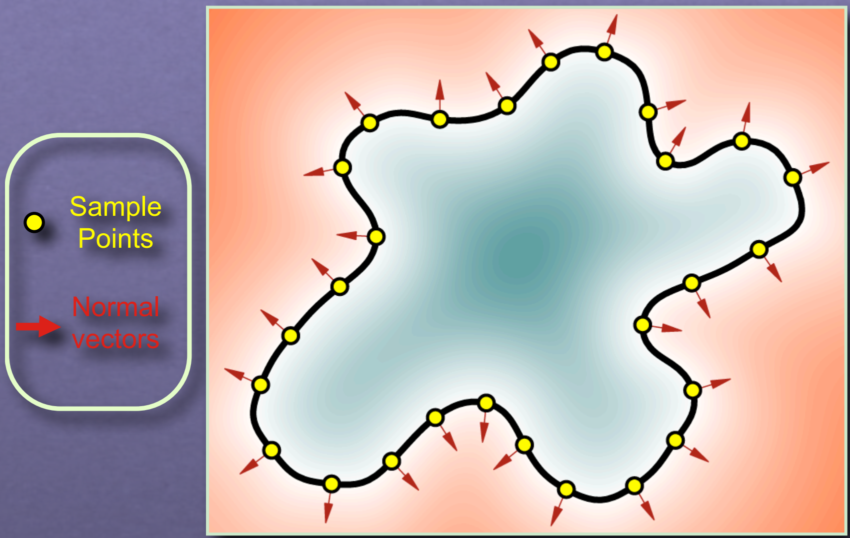
Implicit MLS Surfaces (Or curves in 2D)



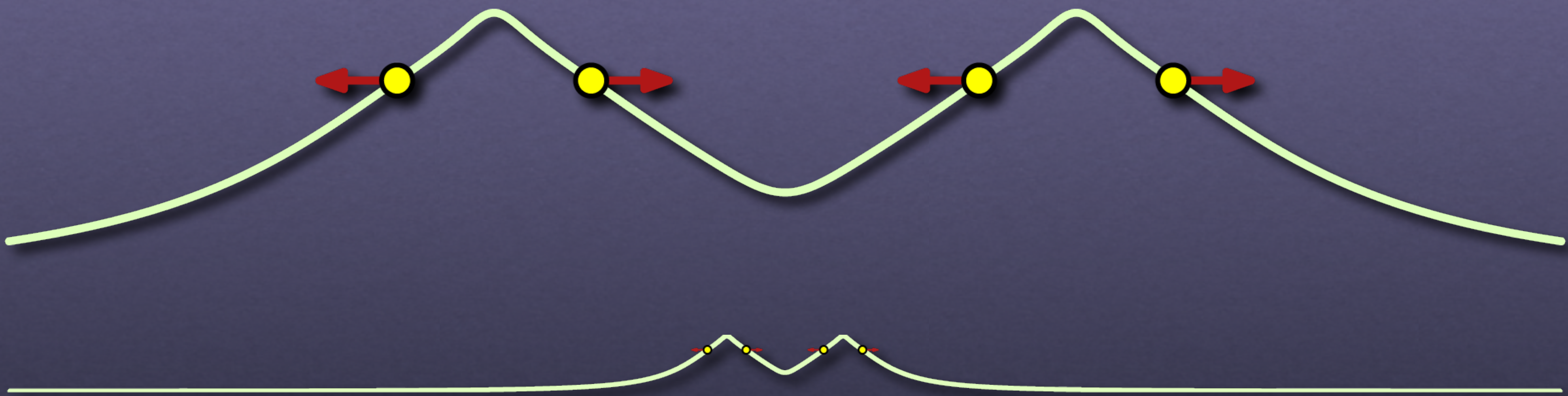
Implicit MLS Surfaces (Or curves in 2D)



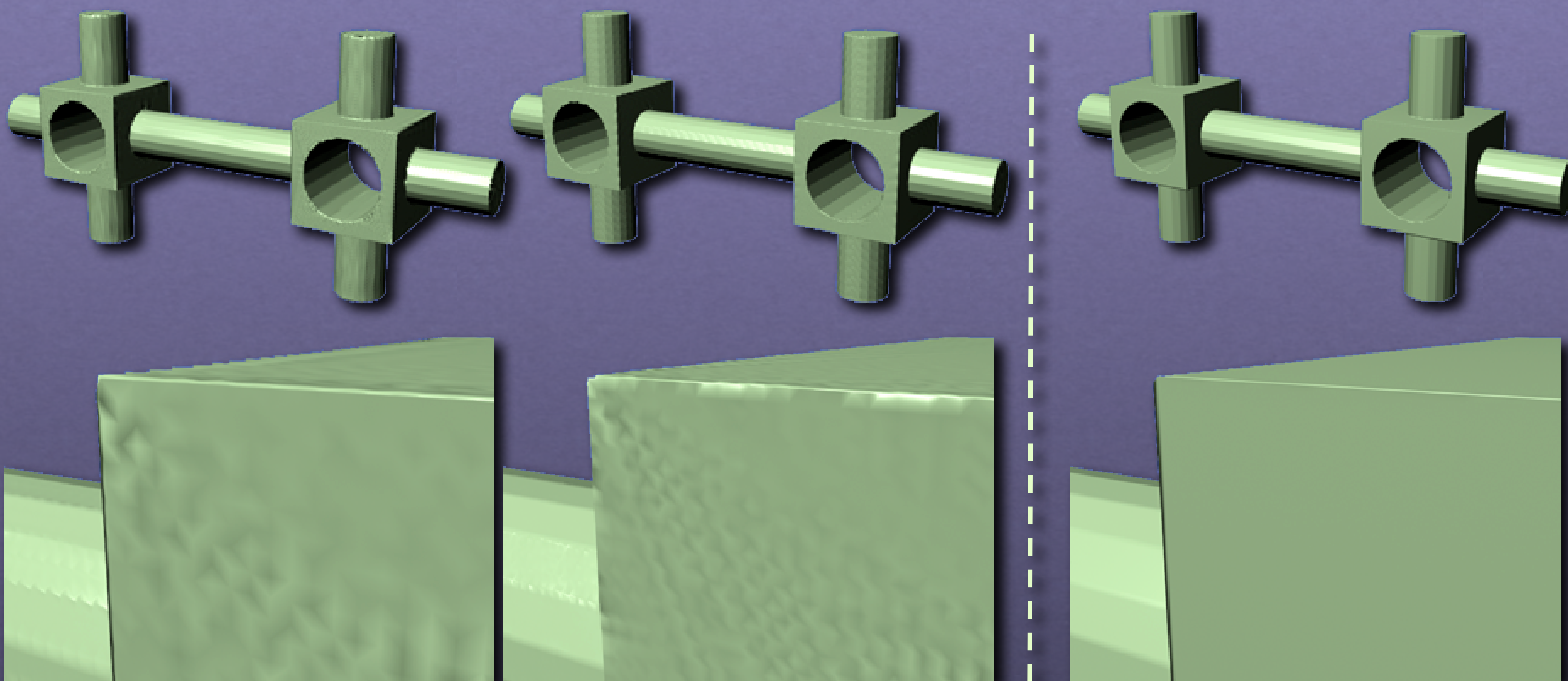
Implicit MLS Surfaces (Or curves in 2D)



Proof of good behavior in Kolluri SODA 2005



Integrated Constraints



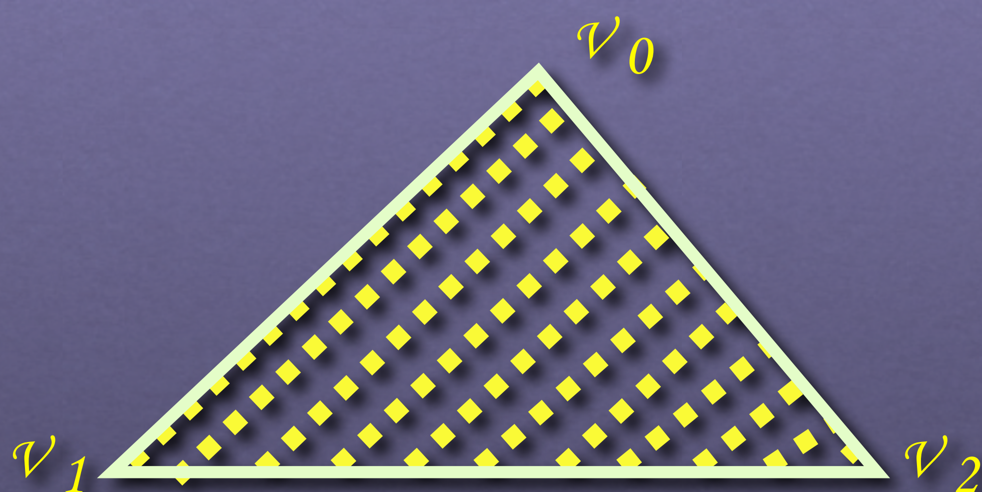
Fine Super Fine

Scattered point constraints

Integrated polygon constraints (new method)

Integrated Constraints

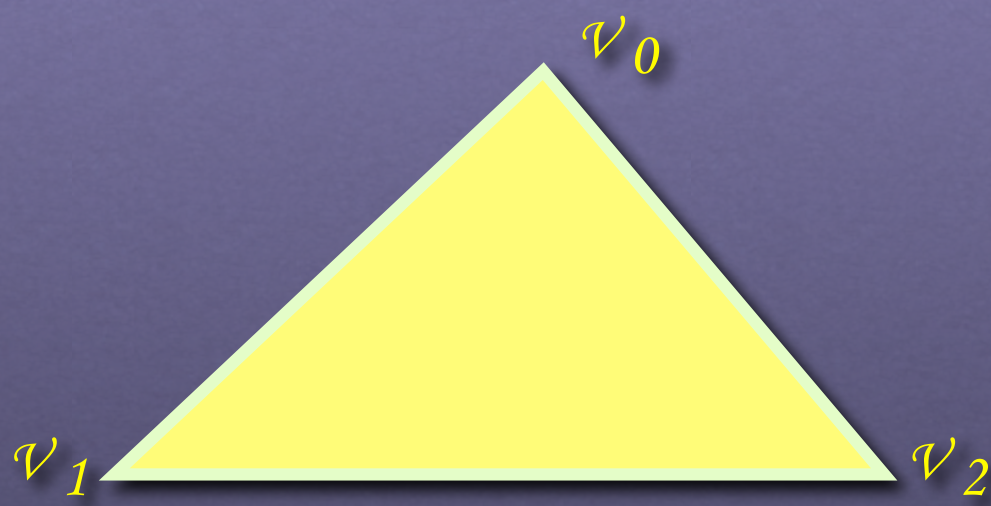
- Scattered point constraints



$$\left(\sum_{i=1}^{N} w^2(\boldsymbol{x}, \boldsymbol{p}_i)\, \boldsymbol{b}(\boldsymbol{p}_i)\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p}_i) \right) \boldsymbol{c}(\boldsymbol{x}) = \sum_{i=1}^{N} w^2(\boldsymbol{x}, \boldsymbol{p}_i)\, \boldsymbol{b}(\boldsymbol{p}_i)\, \phi_i$$

Integrated Constraints

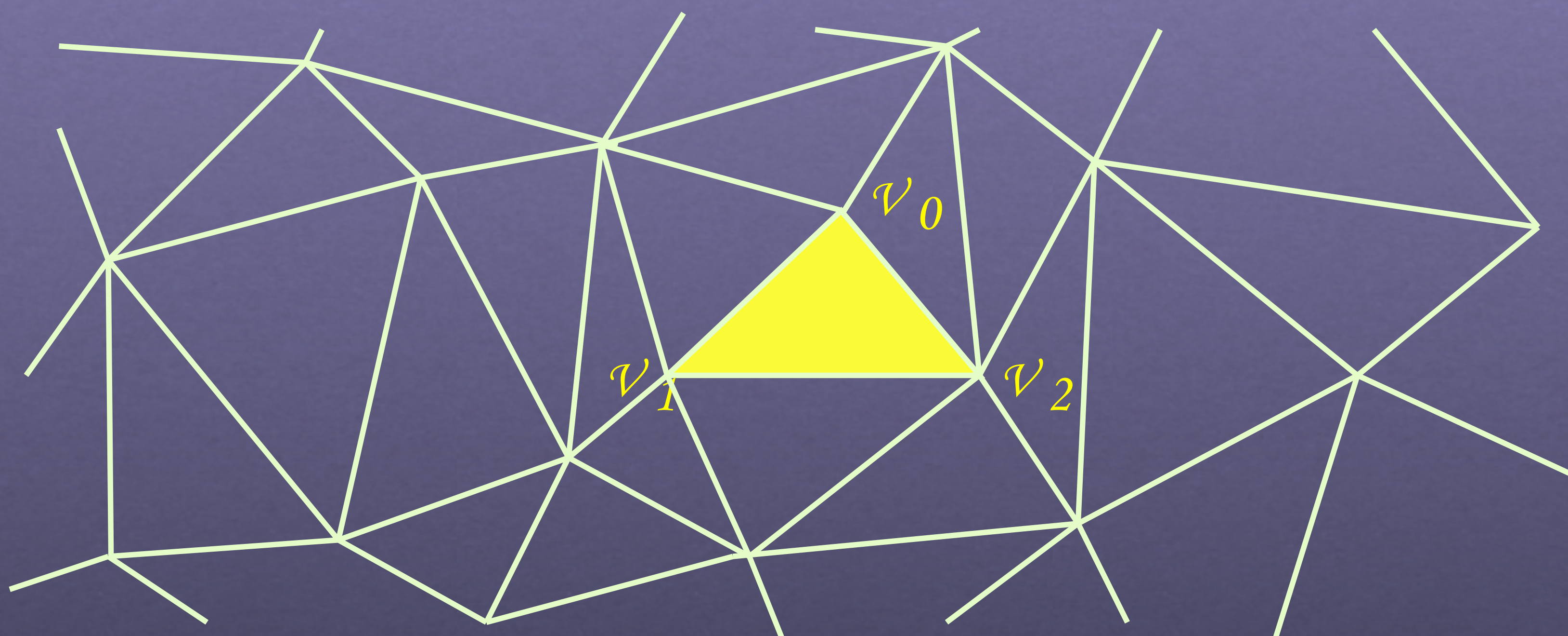
- Constraints integrated over polygons



$$\left(\int_{\Omega_k} w^2(\boldsymbol{x},\boldsymbol{p})\,\boldsymbol{b}(\boldsymbol{p})\,\boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p})\ \,\mathsf{d}\boldsymbol{p}\right)\boldsymbol{c}(\boldsymbol{x}) = \int_{\Omega_k} w^2(\boldsymbol{x},\boldsymbol{p})\,\boldsymbol{b}(\boldsymbol{p})\,\phi_k\ \,\mathsf{d}\boldsymbol{p}$$

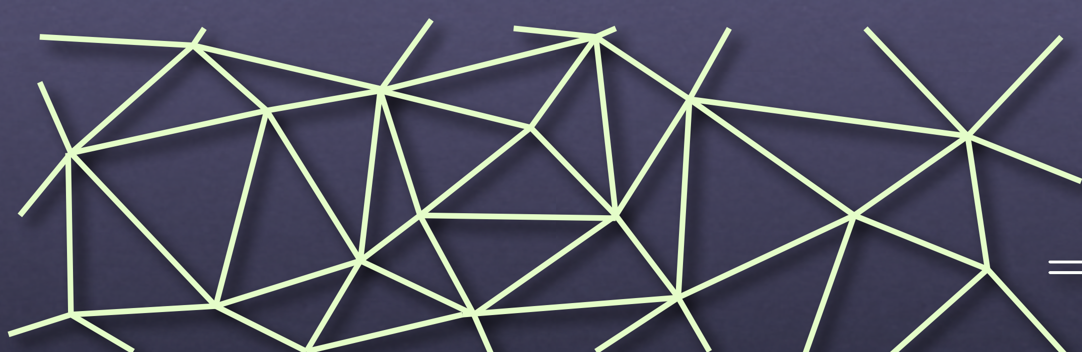
Integrated Constraints

- Sum integrals over mesh



$$\sum_{k=1}^{K} \left(\int_{\Omega_k} w^2(\boldsymbol{x},\boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p}) \;\; \mathsf{d}\boldsymbol{p} \right) \boldsymbol{c}(\boldsymbol{x}) = \sum_{k=1}^{K} \int_{\Omega_k} w^2(\boldsymbol{x},\boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \phi_k \;\; \mathsf{d}\boldsymbol{p}$$

Fast Evaluation

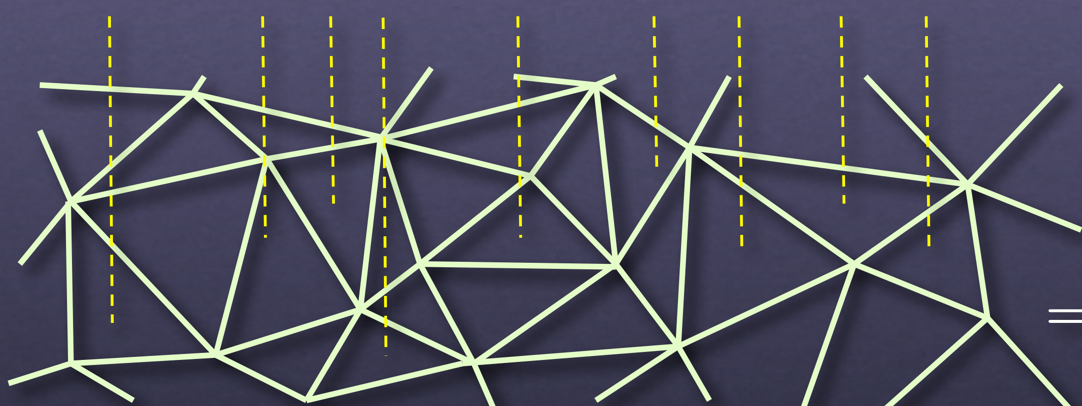


$$\sum_{k=1}^{K} \left(\int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p}) \;\, \mathrm{d}\boldsymbol{p} \right) \boldsymbol{c}(\boldsymbol{x})$$

$$= \sum_{k=1}^{K} \int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \phi_k \;\, \mathrm{d}\boldsymbol{p}$$

Fast Evaluation

$$\int_{\Omega_0} \quad \int_{\Omega_1} \quad \dots \quad \int_{\Omega_k} \quad \dots \quad \int_{\Omega_K} \qquad \int_{\Omega_k} = \int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p}) \ \mathrm{d}\boldsymbol{p}$$



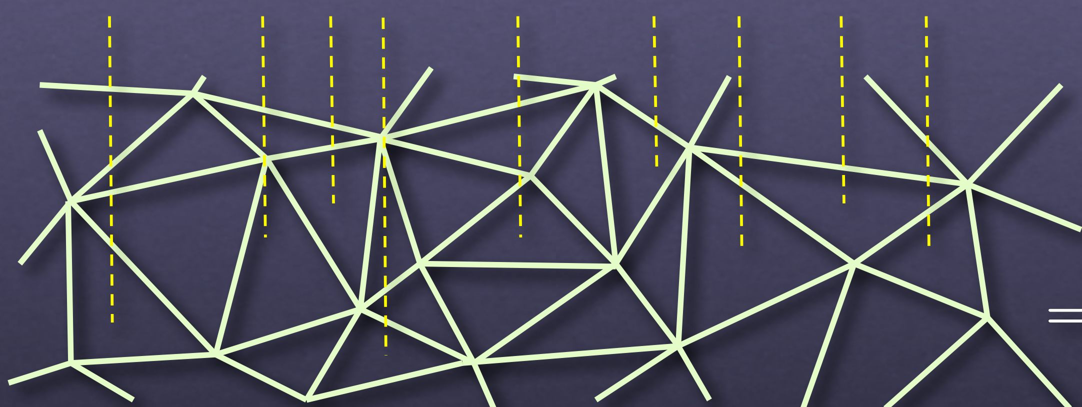
$$\sum_{k=1}^{K} \left(\int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p}) \ \mathrm{d}\boldsymbol{p} \right) \boldsymbol{c}(\boldsymbol{x})$$

$$= \sum_{k=1}^{K} \int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \phi_k \ \mathrm{d}\boldsymbol{p}$$

Fast Evaluation

A_0 A_1 … A_k … A_K

$\int_{\Omega_0}$ $\int_{\Omega_1}$ … $\int_{\Omega_k}$ … $\int_{\Omega_K}$



$$\boldsymbol{A}_k = \int_{\Omega_k} \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^\top(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p}$$

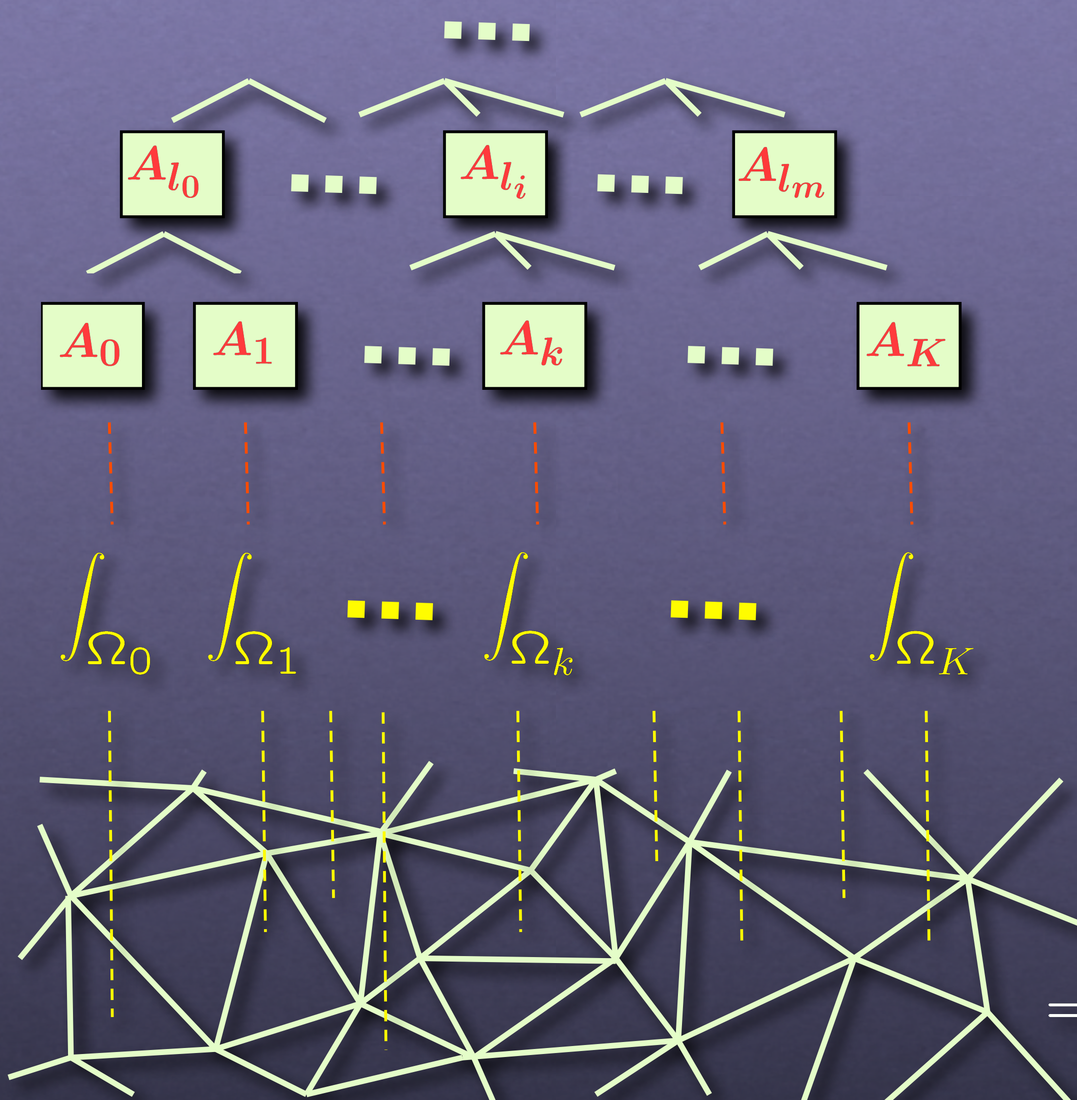
$$\int_{\Omega_k} \approx w^2(\boldsymbol{x}, \boldsymbol{p_c})\, \boldsymbol{A}_k$$

$$\int_{\Omega_k} = \int_{\Omega_k} \boxed{w^2(\boldsymbol{x}, \boldsymbol{p})}\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^\top(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p}$$

$$\sum_{k=1}^{K} \left(\int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^\top(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p} \right) \boldsymbol{c}(\boldsymbol{x})$$

$$= \sum_{k=1}^{K} \int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \phi_k\ \mathrm{d}\boldsymbol{p}$$

Fast Evaluation



K-D tree averaging

$$\boldsymbol{A_k} = \int_{\Omega_k} \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p}$$

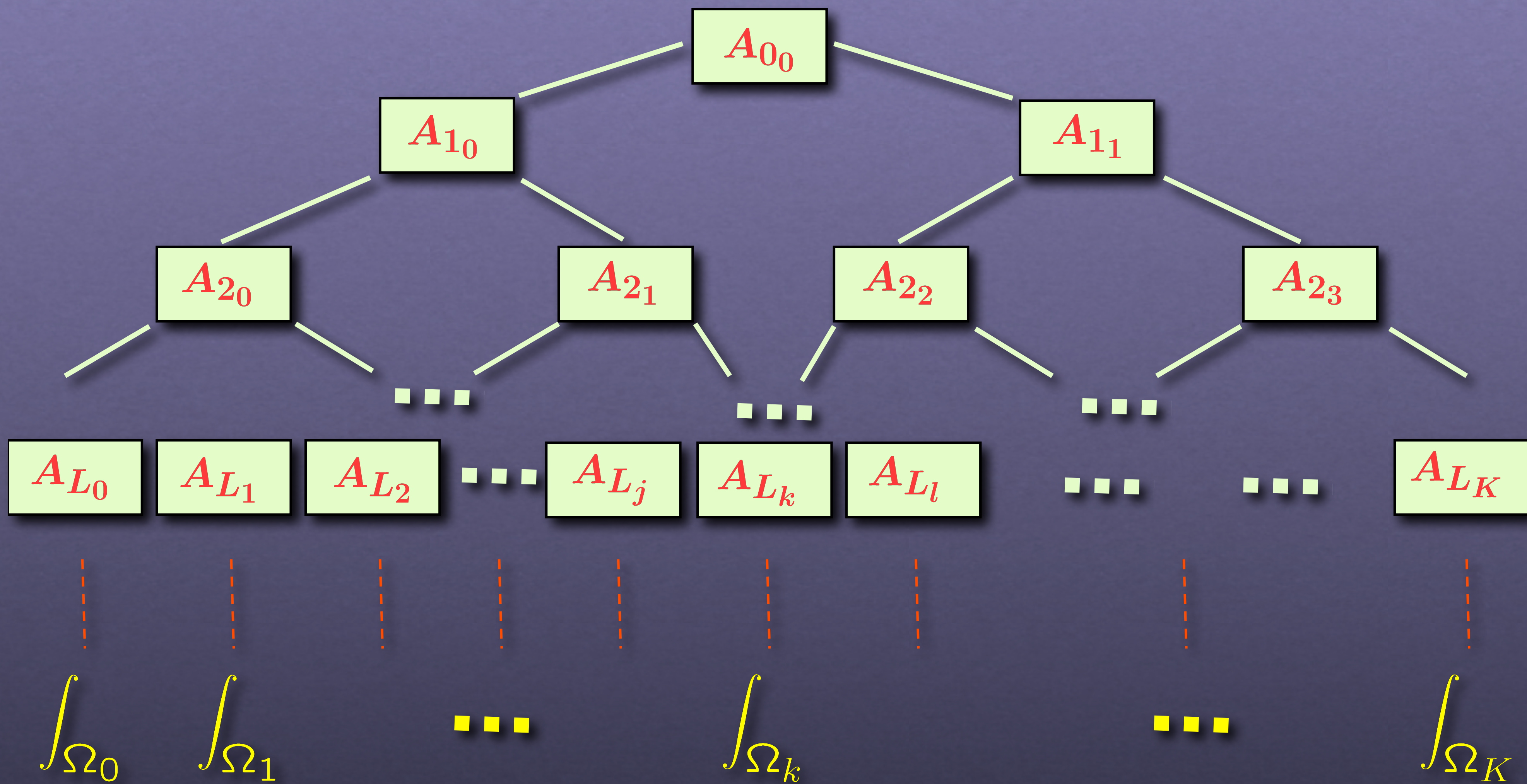
$$\int_{\Omega_k} \approx w^2(\boldsymbol{x}, \boldsymbol{p_c})\, \boldsymbol{A_k}$$

$$\int_{\Omega_k} = \int_{\Omega_k} \boxed{w^2(\boldsymbol{x}, \boldsymbol{p})}\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p}$$

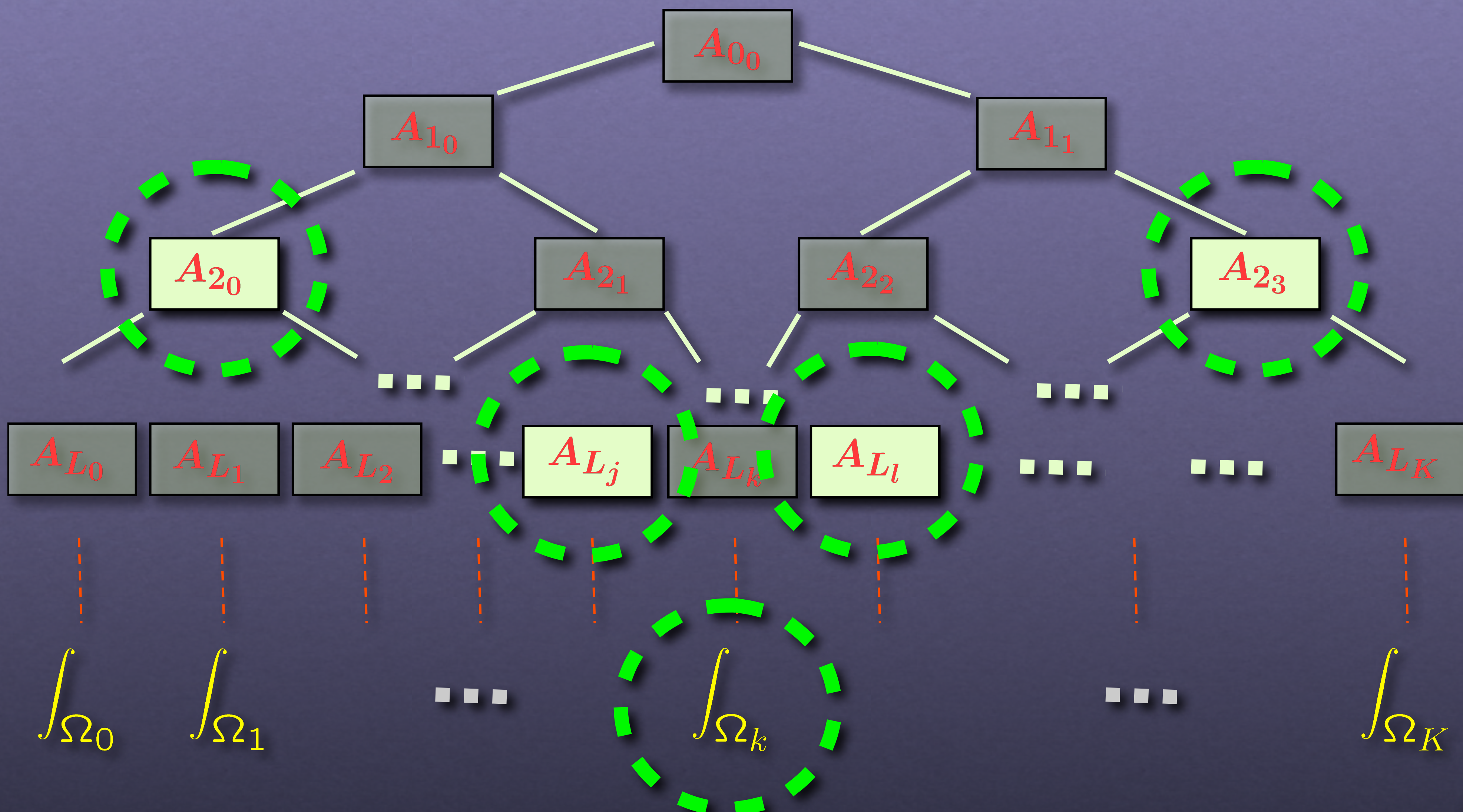
$$\sum_{k=1}^{K} \left(\int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \boldsymbol{b}^{\mathsf{T}}(\boldsymbol{p})\ \mathrm{d}\boldsymbol{p} \right) \boldsymbol{c}(\boldsymbol{x})$$

$$= \sum_{k=1}^{K} \int_{\Omega_k} w^2(\boldsymbol{x}, \boldsymbol{p})\, \boldsymbol{b}(\boldsymbol{p})\, \phi_k\ \mathrm{d}\boldsymbol{p}$$

Fast Evaluation



Fast Evaluation

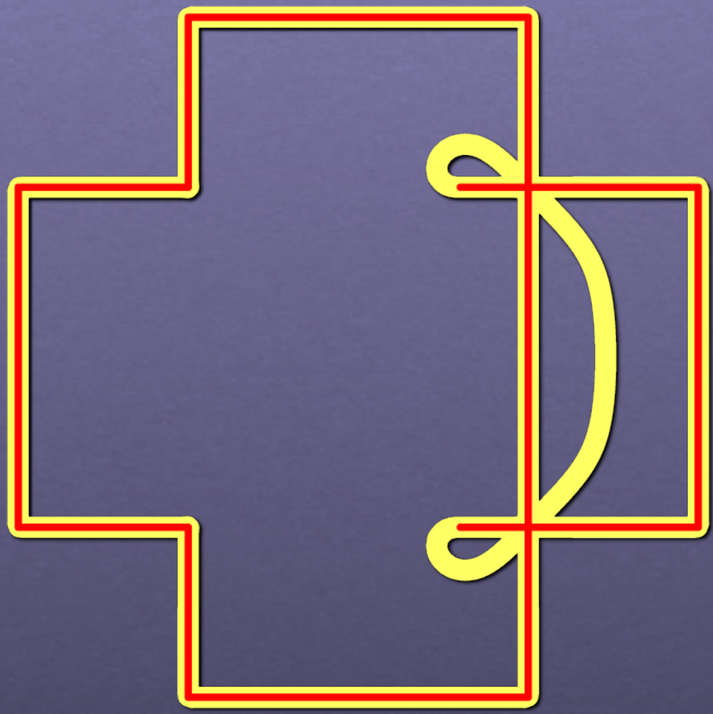


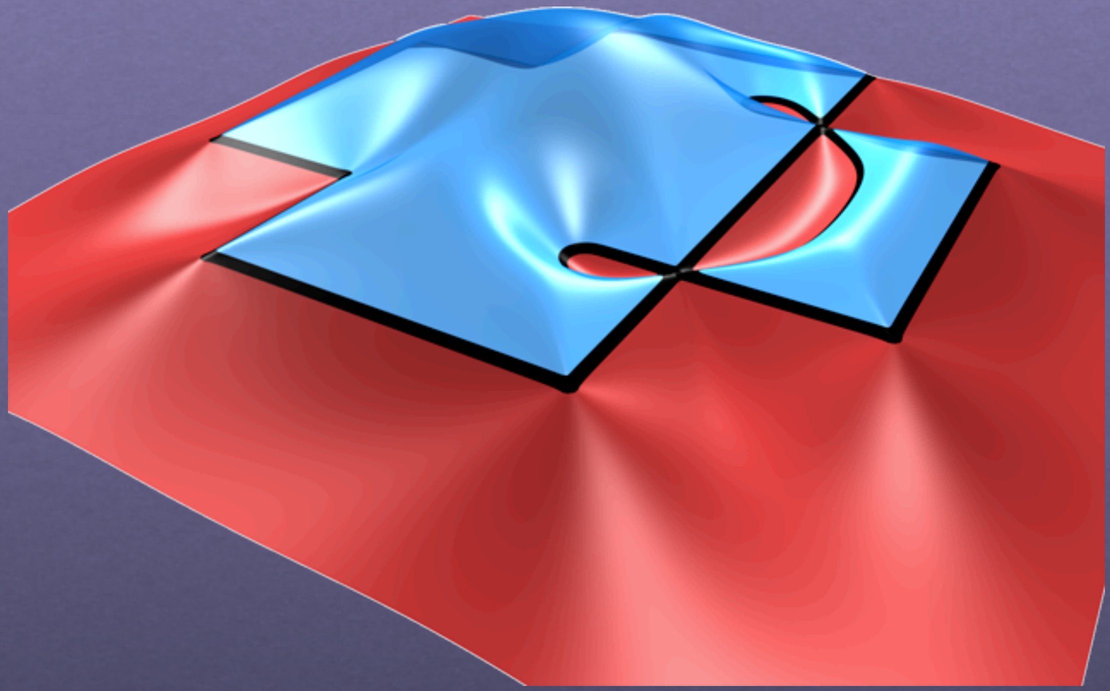
Approximation Adjustment

- Naive approximation

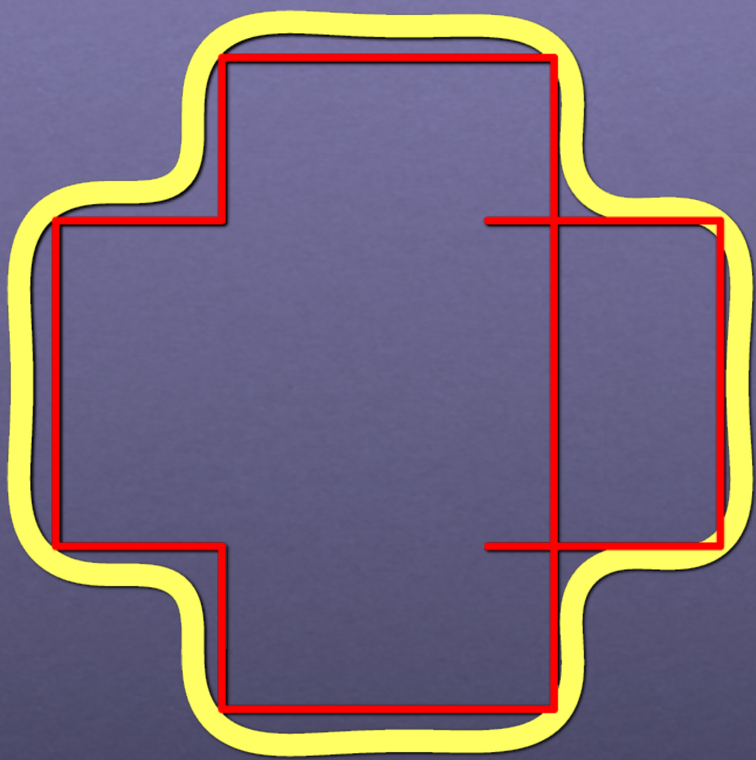
- With iterative value adjustment

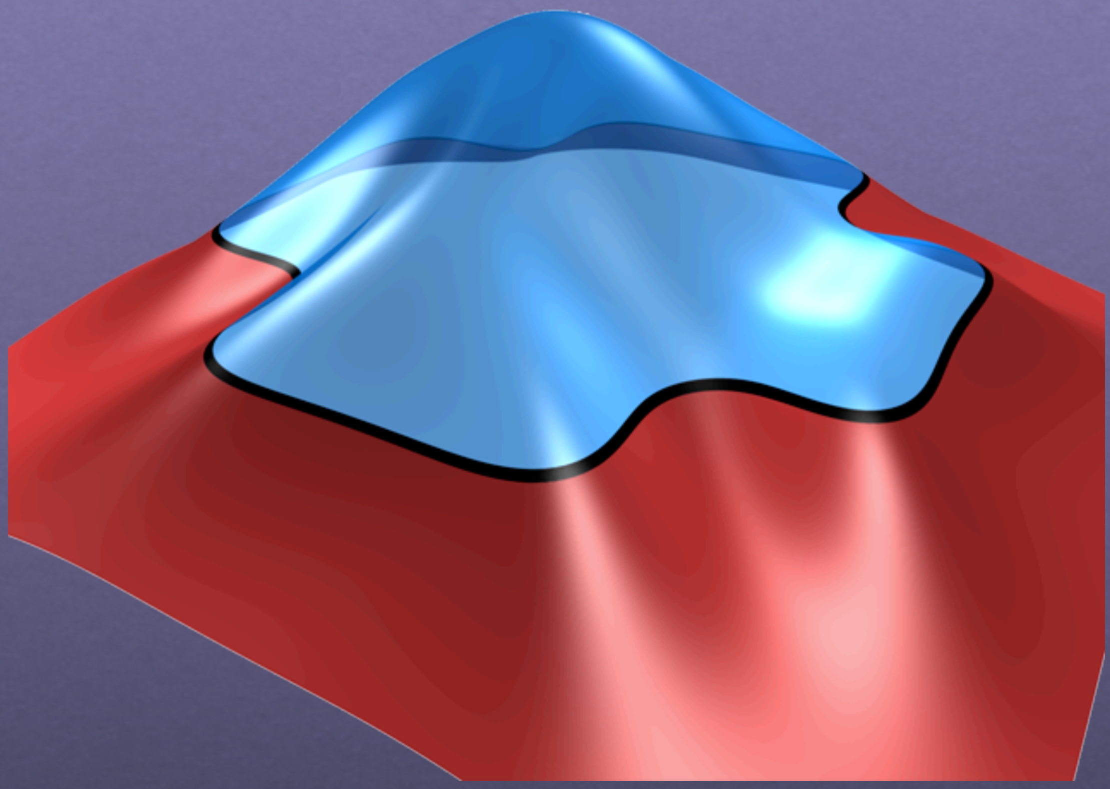
Interpolating/Approximation



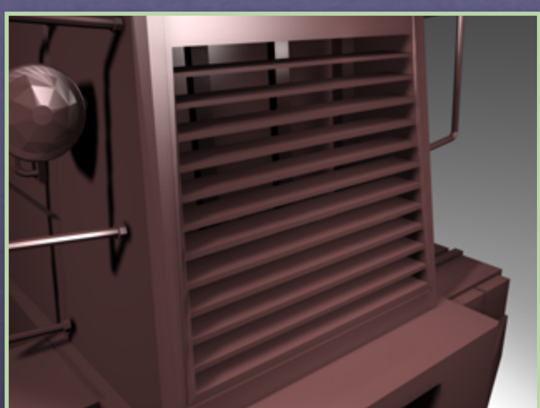


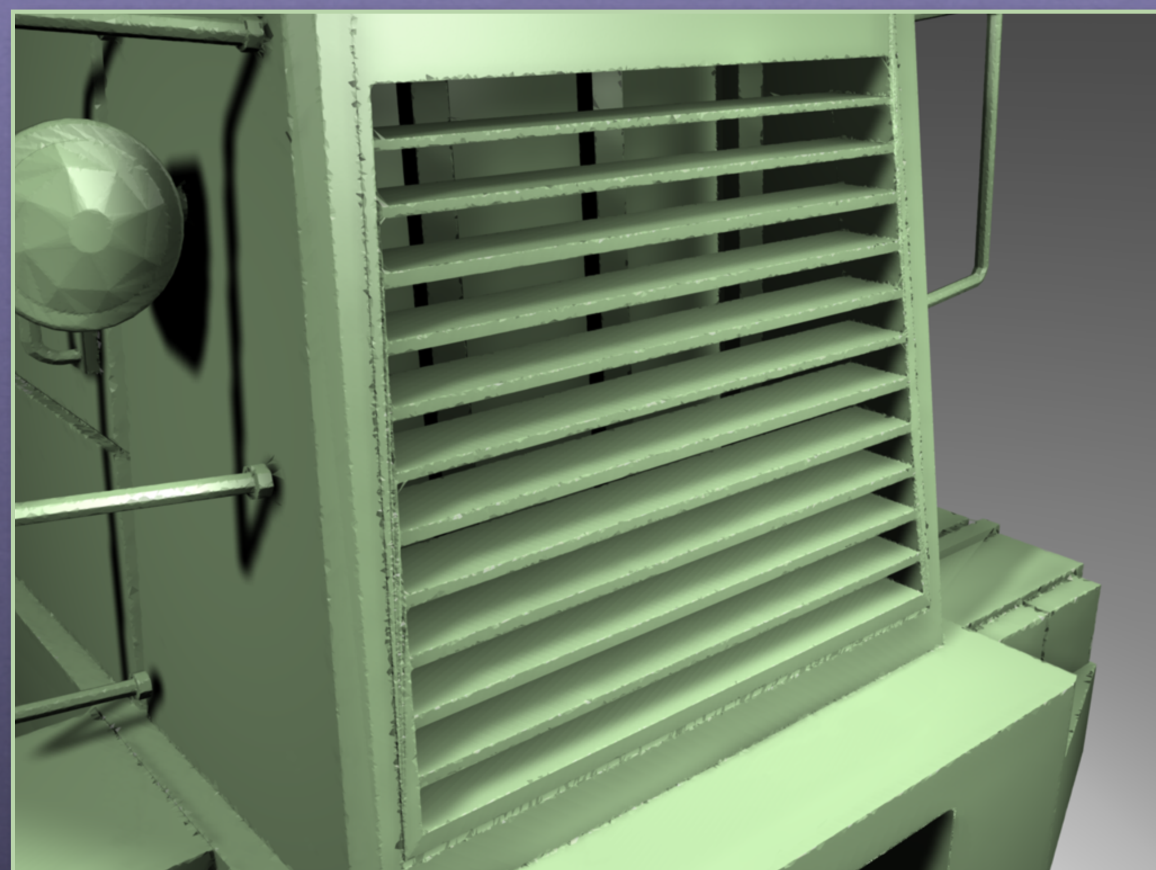
Interpolating/Approximation

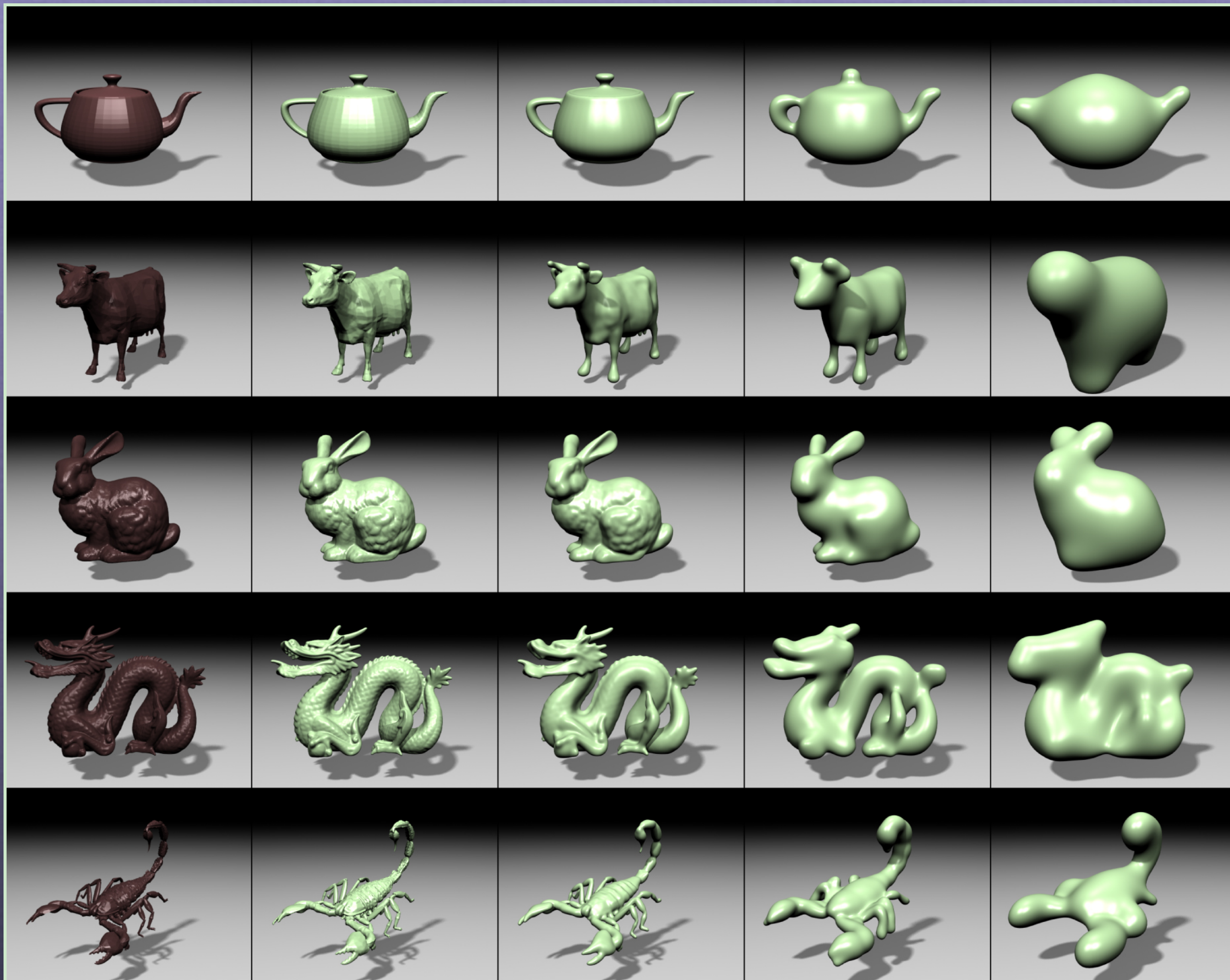


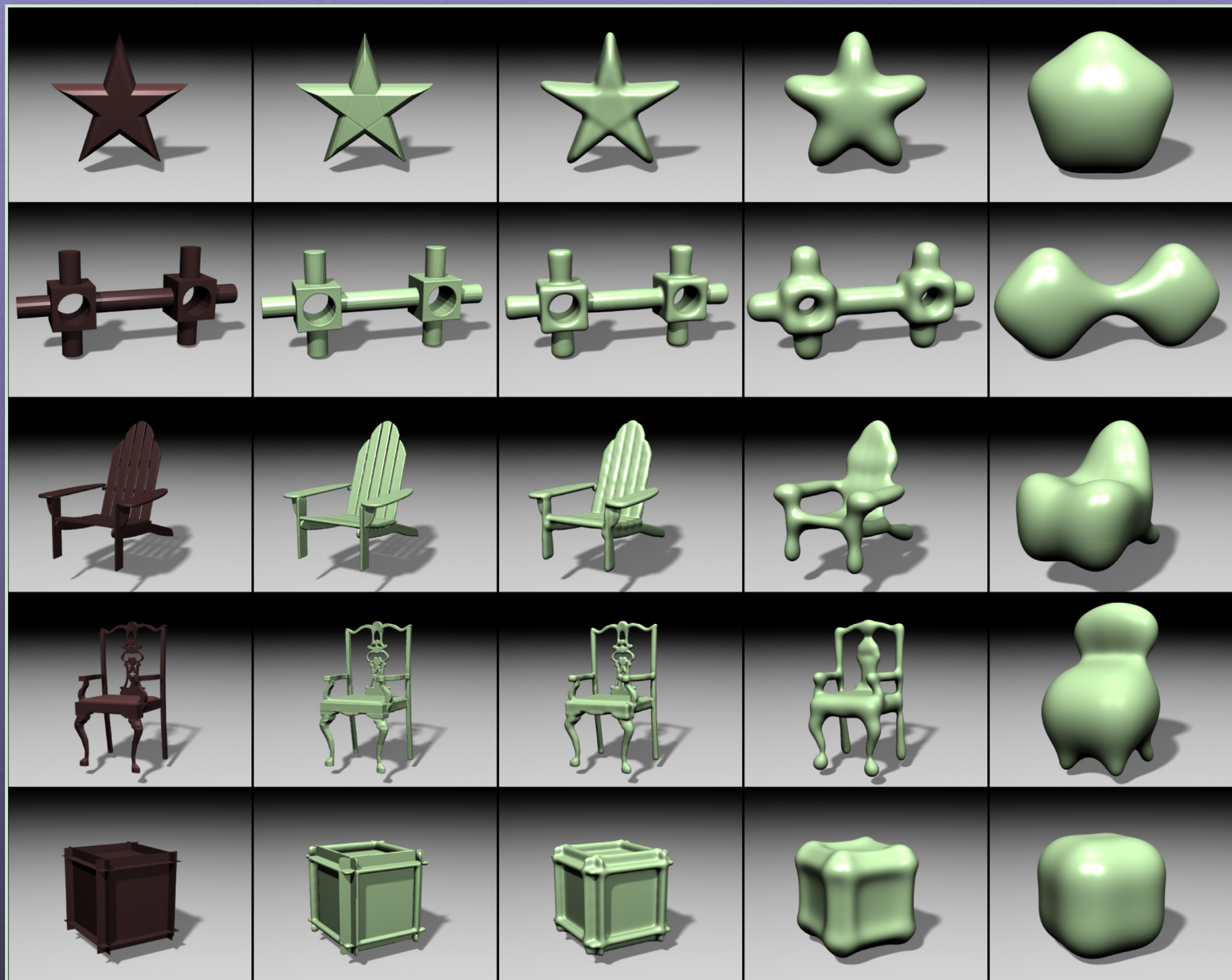


Close Up



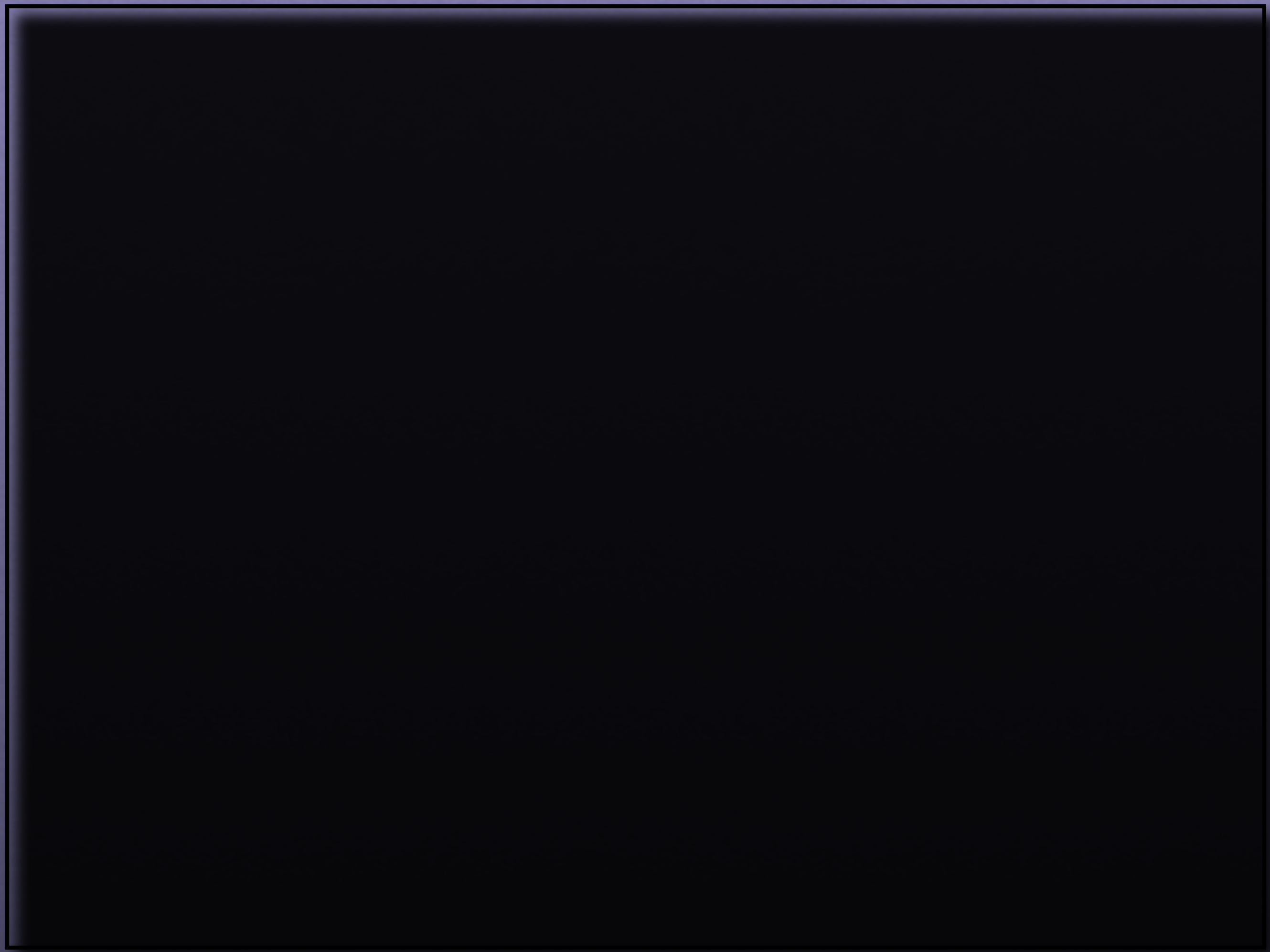






Rapid Prototyping

“Gratuitous Goop” from SIGGRAPH 2004 Electronic Theater



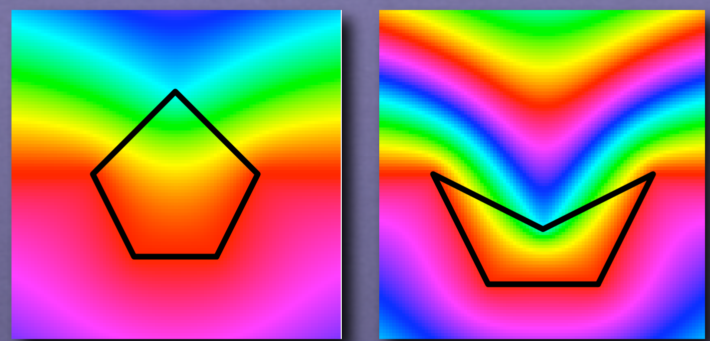
“Gratuitous Goop” from SIGGRAPH 2004 Electronic Theater

Barycentric Coordinates

- Barycentric coordinates defined for simplices are incredibly useful
- Various generalizations
  - Wachspress 1975, Loop & DeRose 1989, Meyer *et al.* 2002, Warren *et al.* 2004
  - Floater 2003, Malsch & Dasgupta 2003, Horman 2004
  - ... and others. (See Ju, Schaefer & Warren in SIGGRAPH 2005.)

Mean Value Coordinates

- Ju, Schaefer & Warren SIGGRAPH 2005



- Equivalent to integrated MLS
  - Constant basis and point functions
  - Weight function

$$w = \frac{\cos(\theta)}{r^3}$$

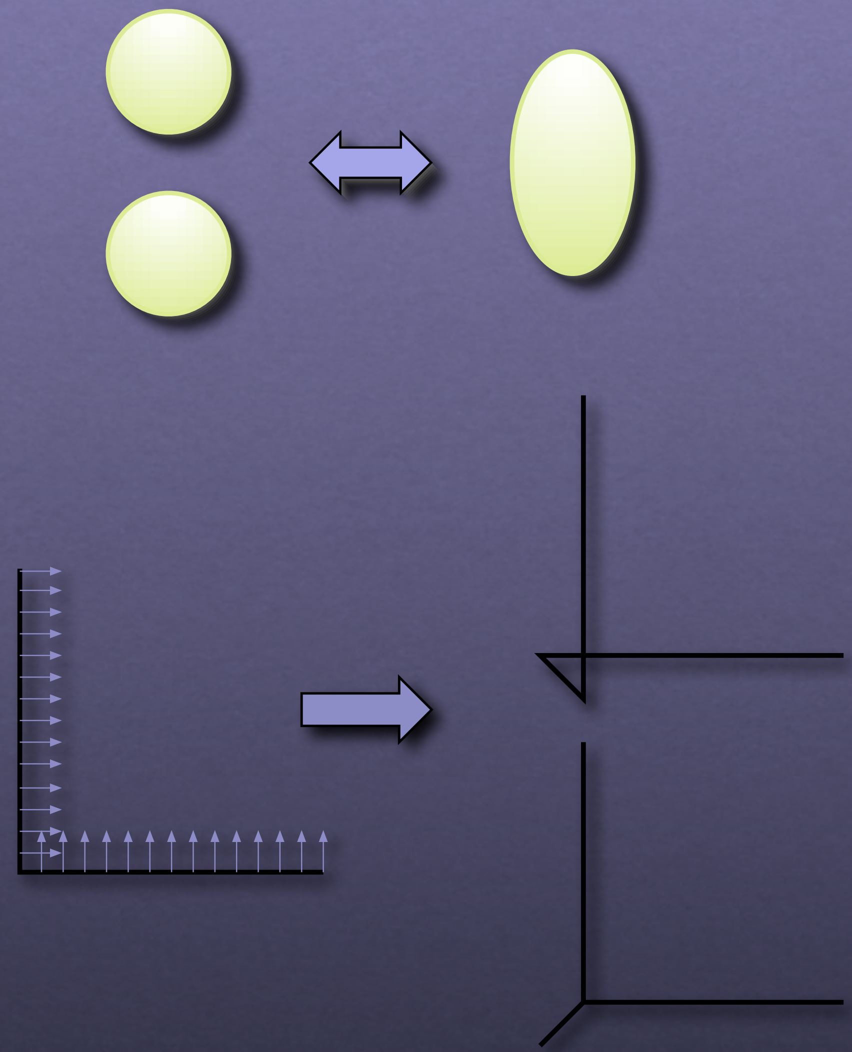
- Very nice properties... read their paper!

Surface Tracking

- Given a surface and velocity field
- Track surface as it moves over time
- Velocity field may be influenced by surface motion

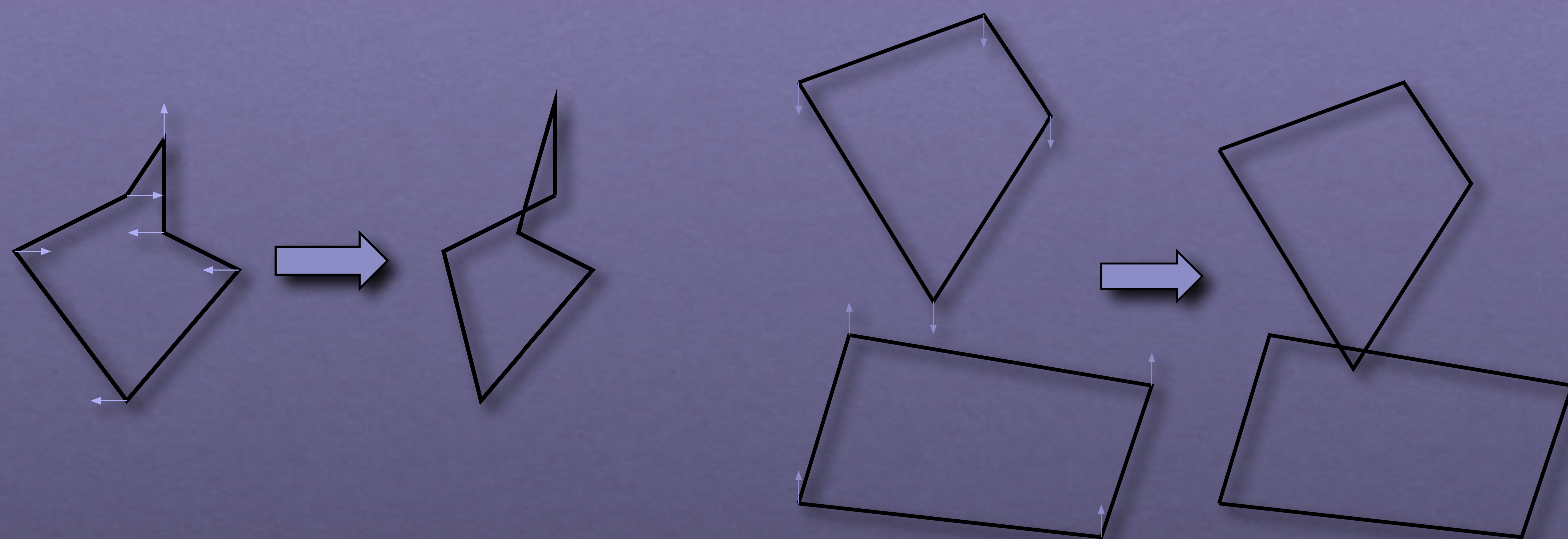
Surface Tracking

- Global topology
- Local structure



Surface Tracking

- Adverting polygons is difficult



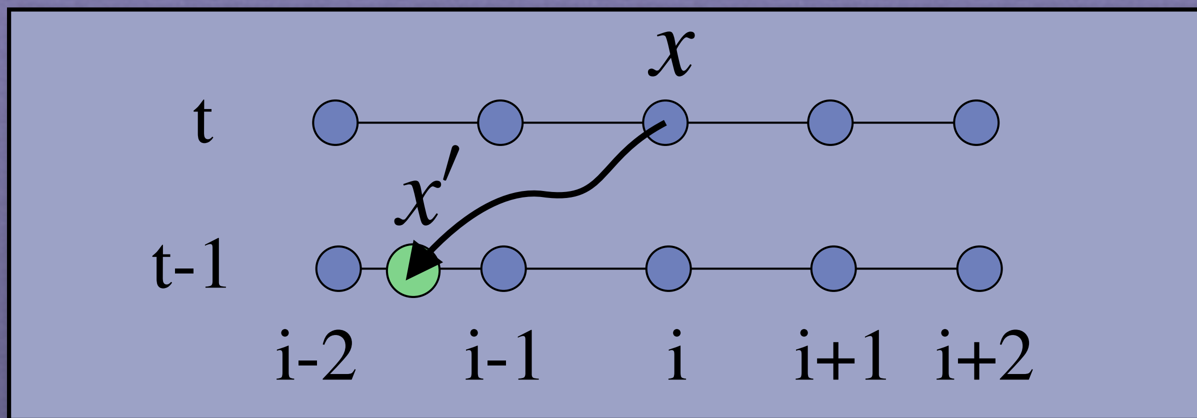
- Large family of level-set and related implicit methods have been developed
  - See paper or text by Osher and Fedkiw for detailed list...

Semi-Lagrangian Contouring

- Define a *composite-implicit function*:
  - Perform semi-Lagrangian backwards path tracing
  - Evaluate the exact(-ish) distance to the polygon mesh at the previous timestep
- The zero-set of composite function defines new surface

Strain, JCP 2001
Bargteil, Goktekin, O'Brien, Strain, *TOG 2005*

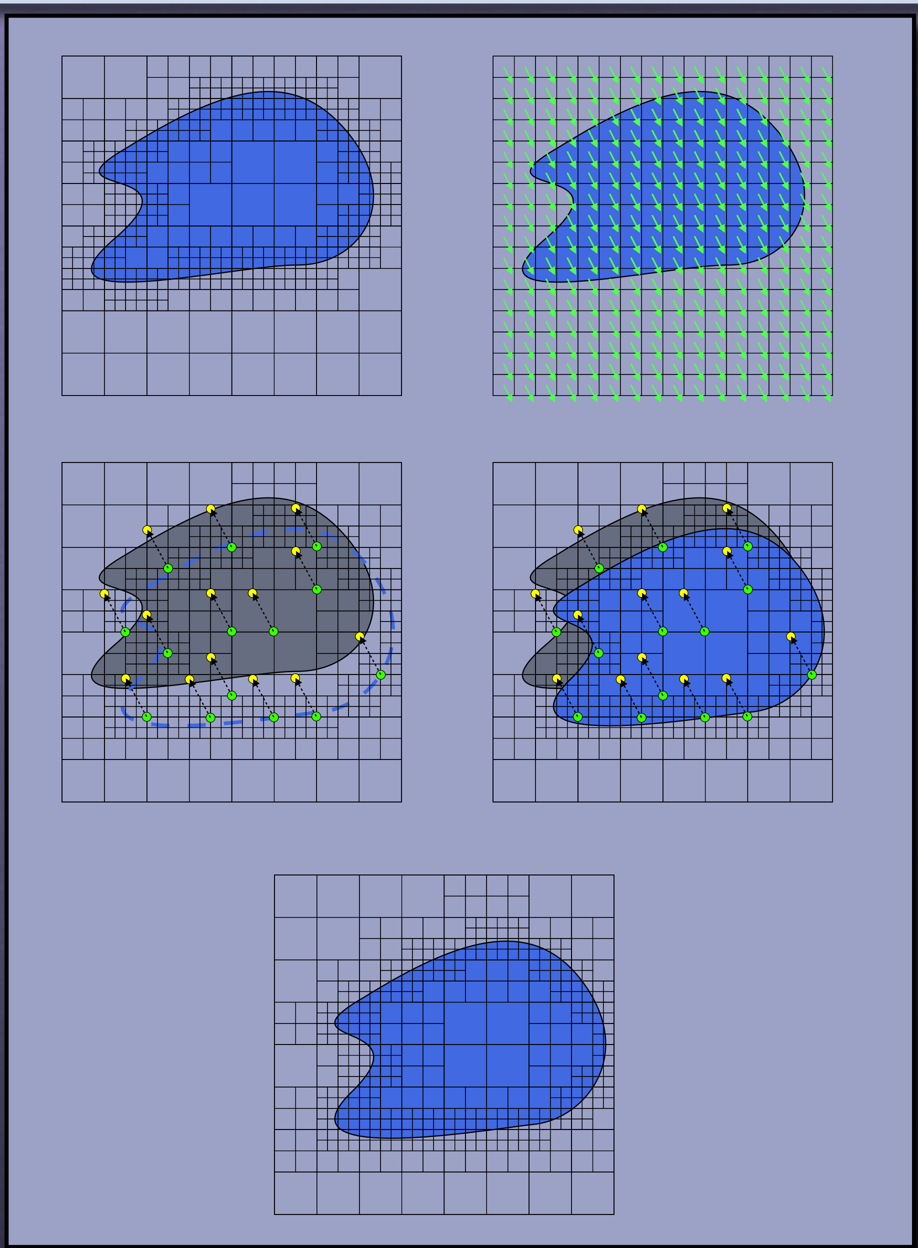
Semi-Lagrangian Advection



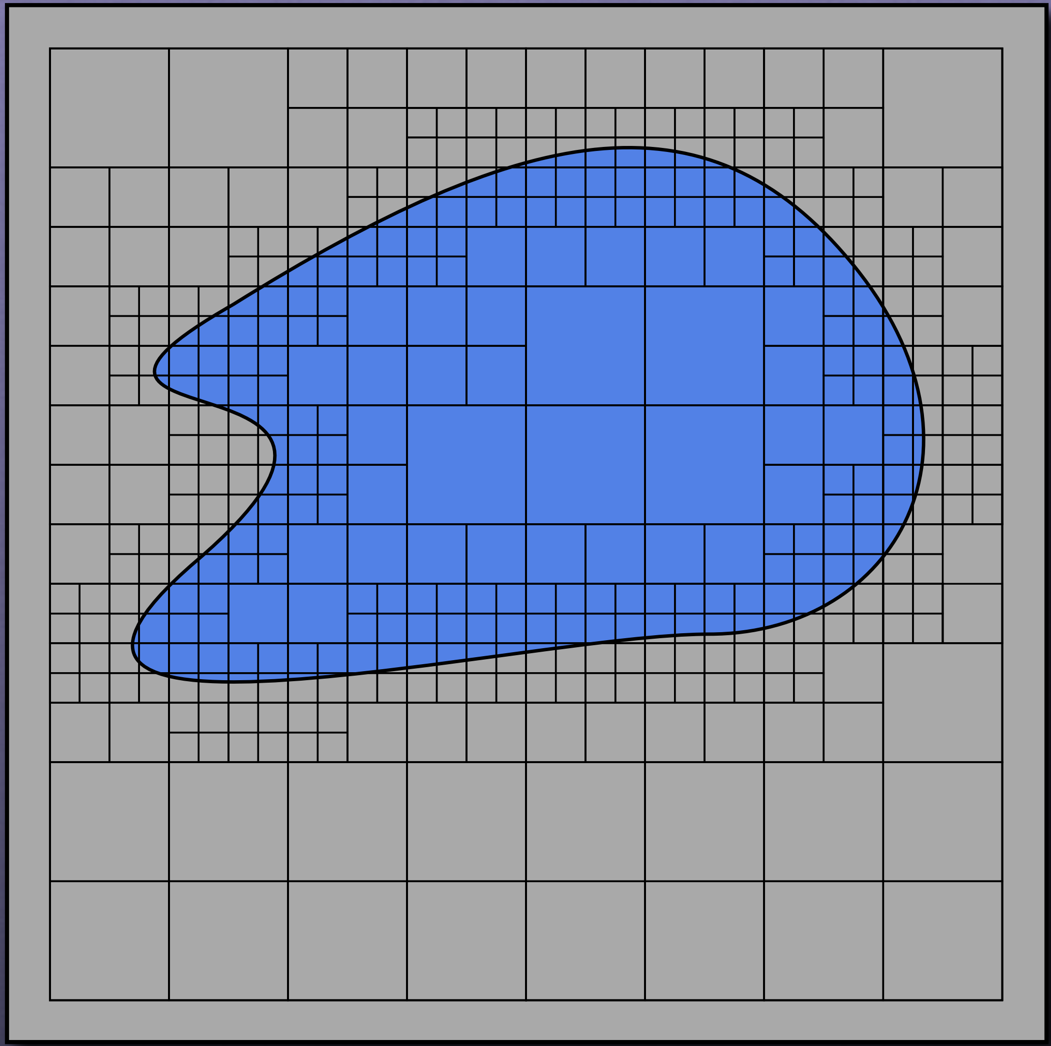
- Obtain new values by backward path tracing followed by interpolation
- Introduced to graphics by Stam in 1999

Algorithm Overview

- Start with polymesh, octree, and velocity field
- Build new octree
- Build new polymesh
- Redistance octree
- Repeat



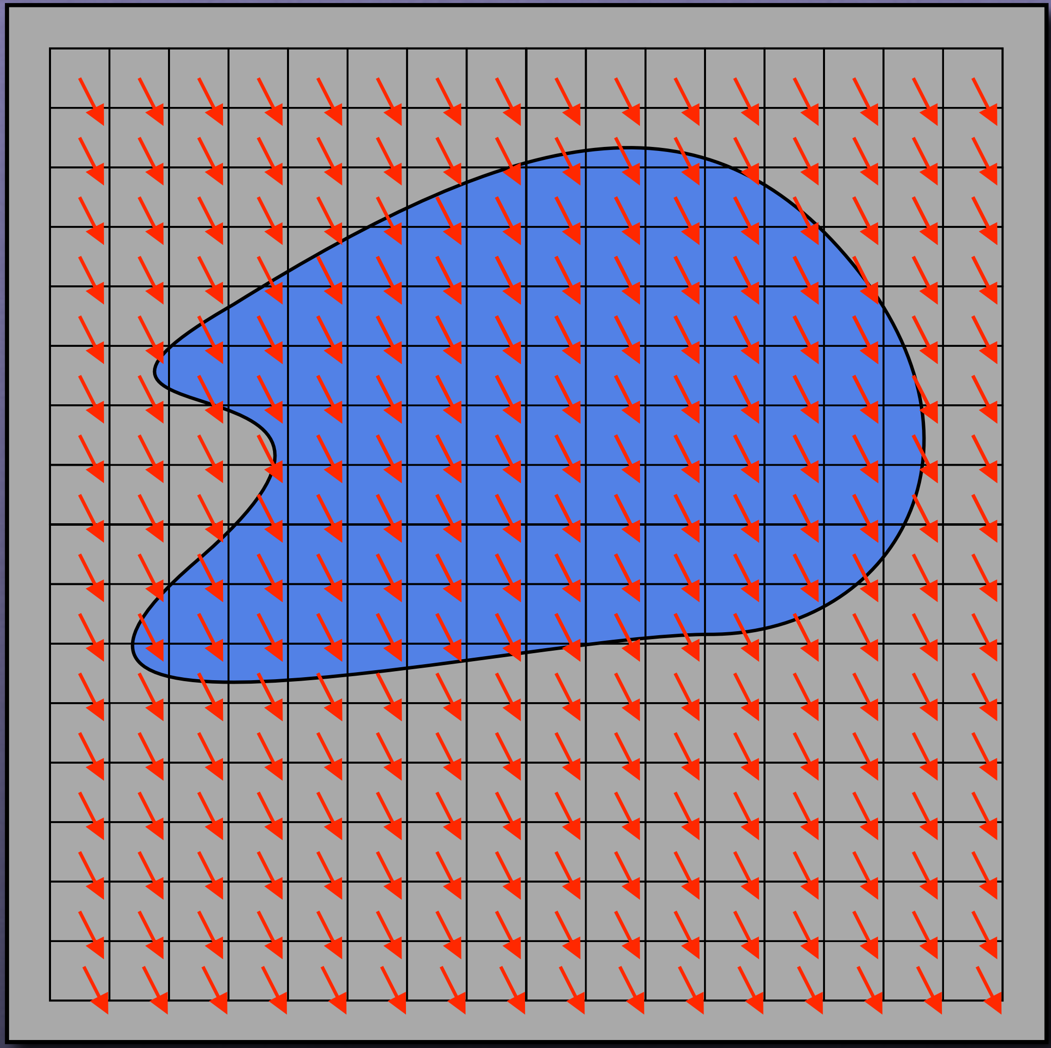
Surface Representation



- Polygon mesh
- Distance tree
  - Accelerated lookup
  - Approximate signed distance away from mesh

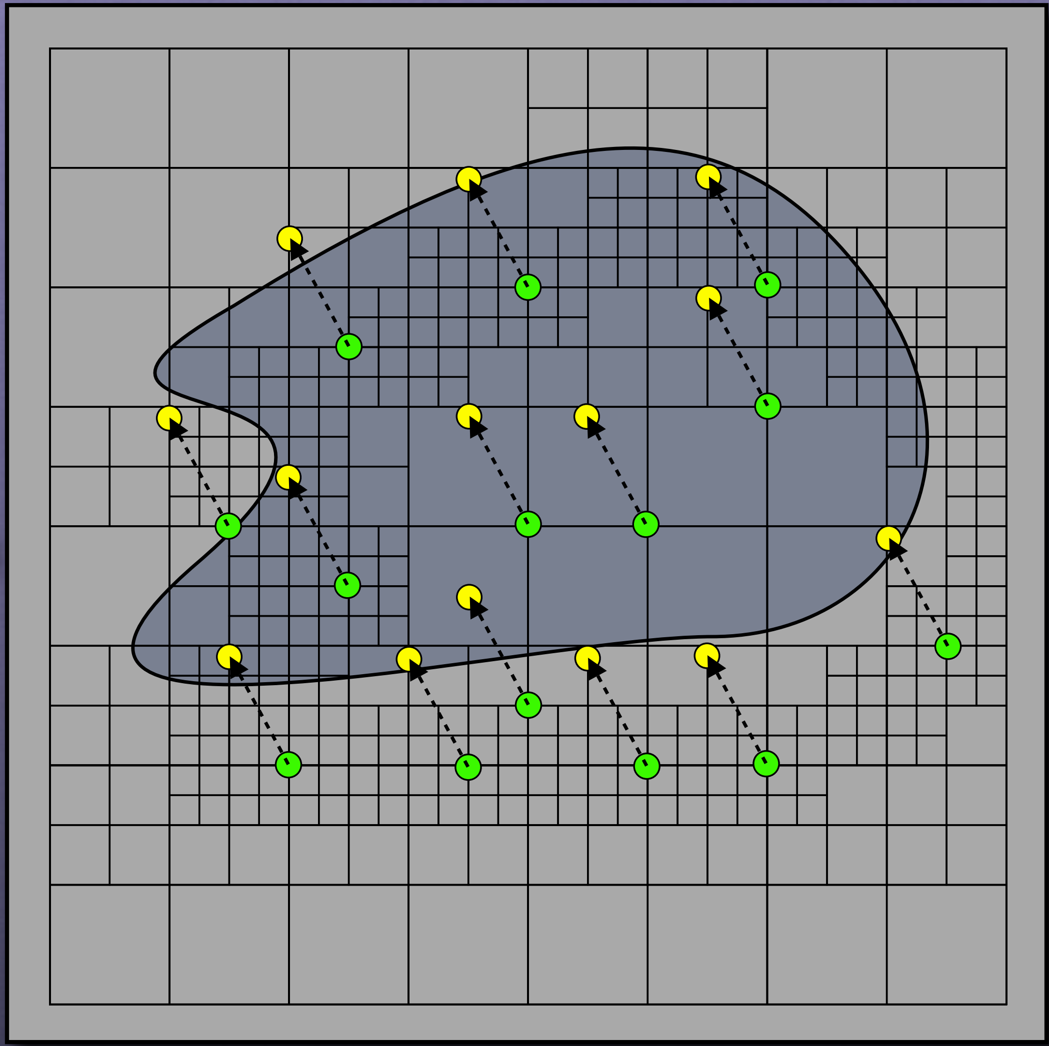
Building New Surface

- Compute velocities
  - e.g. Fluid simulation



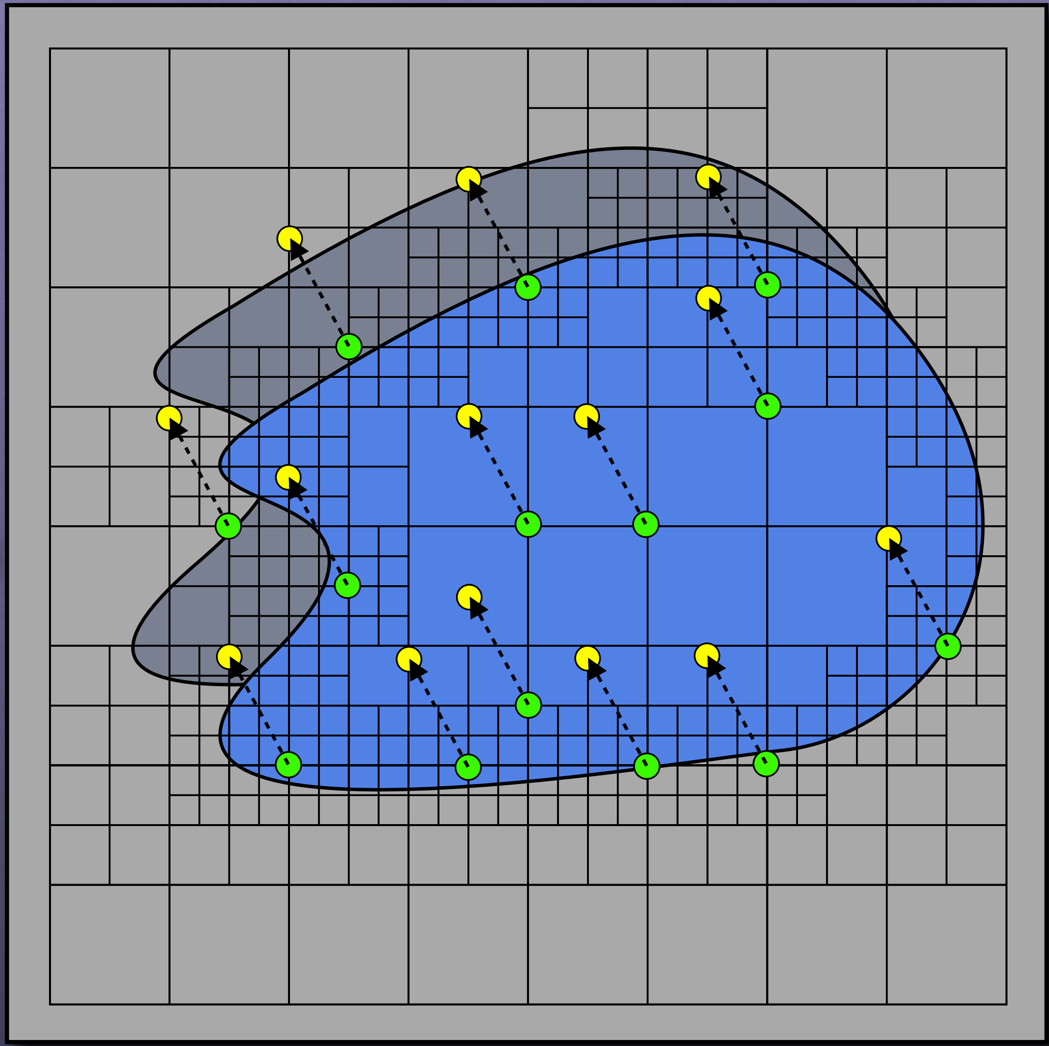
Building New Surface

- Compute velocities
  - e.g. Fluid simulation
- Build new octree
  - S.D. values are verts.
  - Trace back to old mesh
  - Adaptively refine



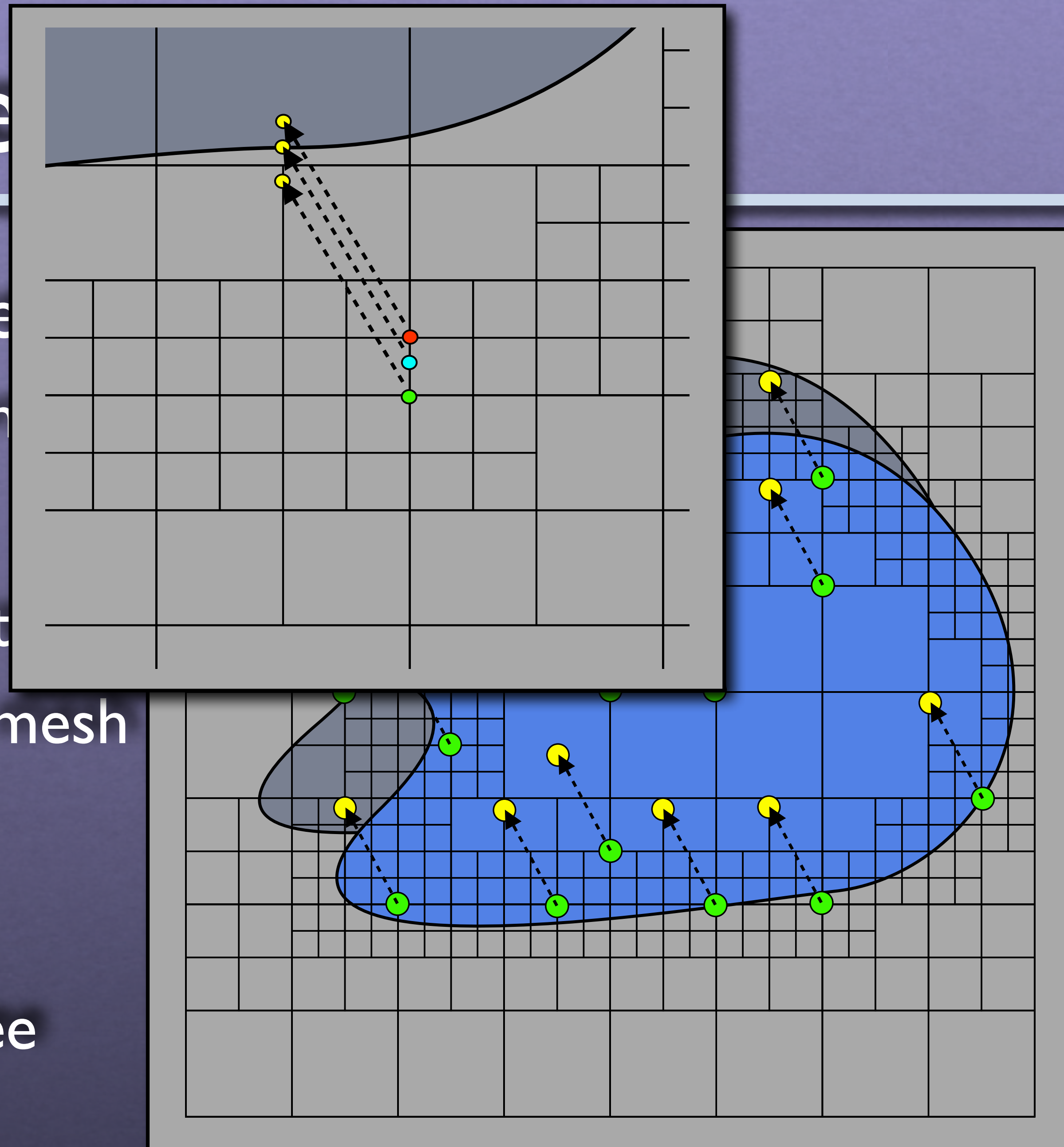
Building New Surface

- Compute velocities
  - e.g. Fluid simulation
- Build new octree
  - S.D. values are verts.
  - Trace back to old mesh
  - Adaptively refine
- Build new mesh
  - M.C. on new octree



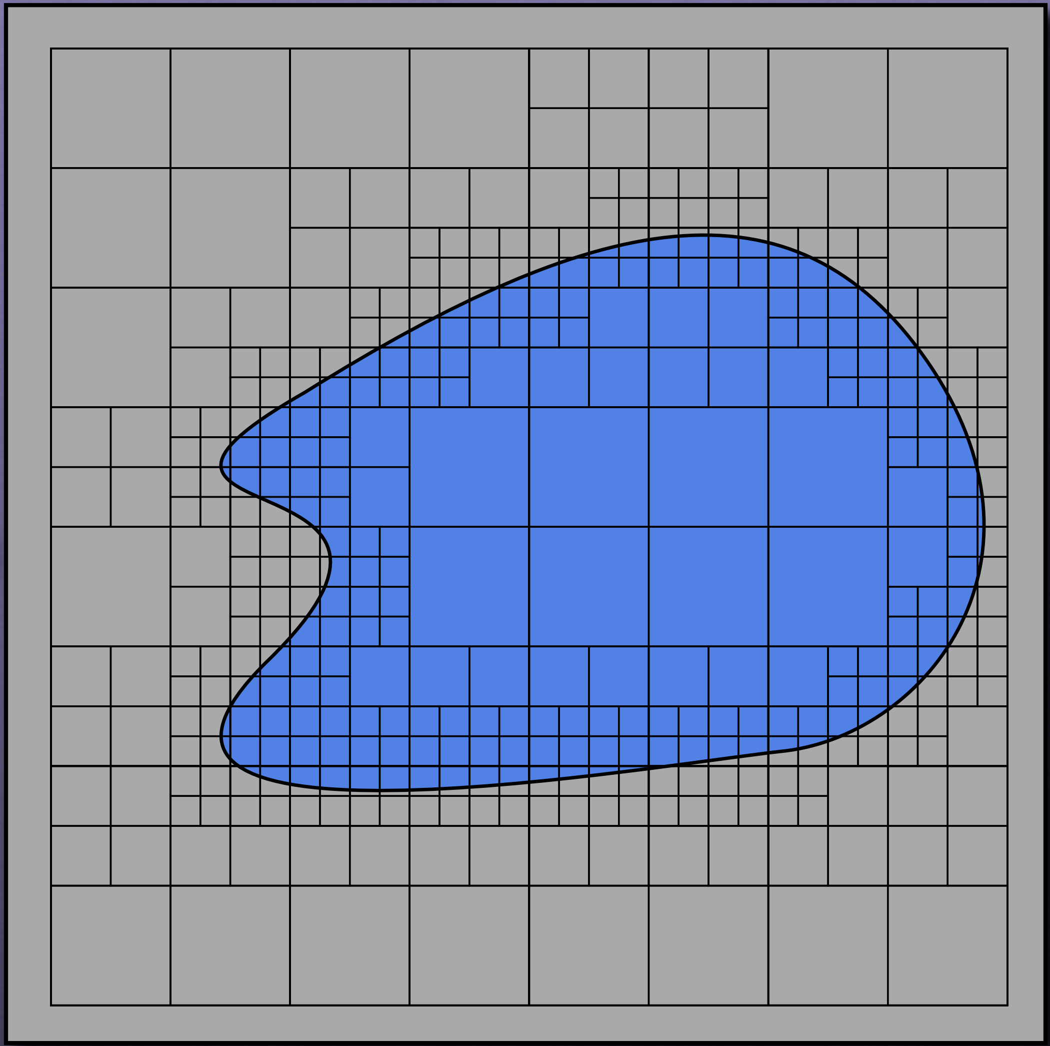
Building Ne

- Compute velocitie
  - e.g. Fluid simulation
- Build new octree
  - S.D. values are vert
  - Trace back to old mesh
  - Adaptively refine
- Build new mesh
  - M.C. on new octree
  - Exact distances

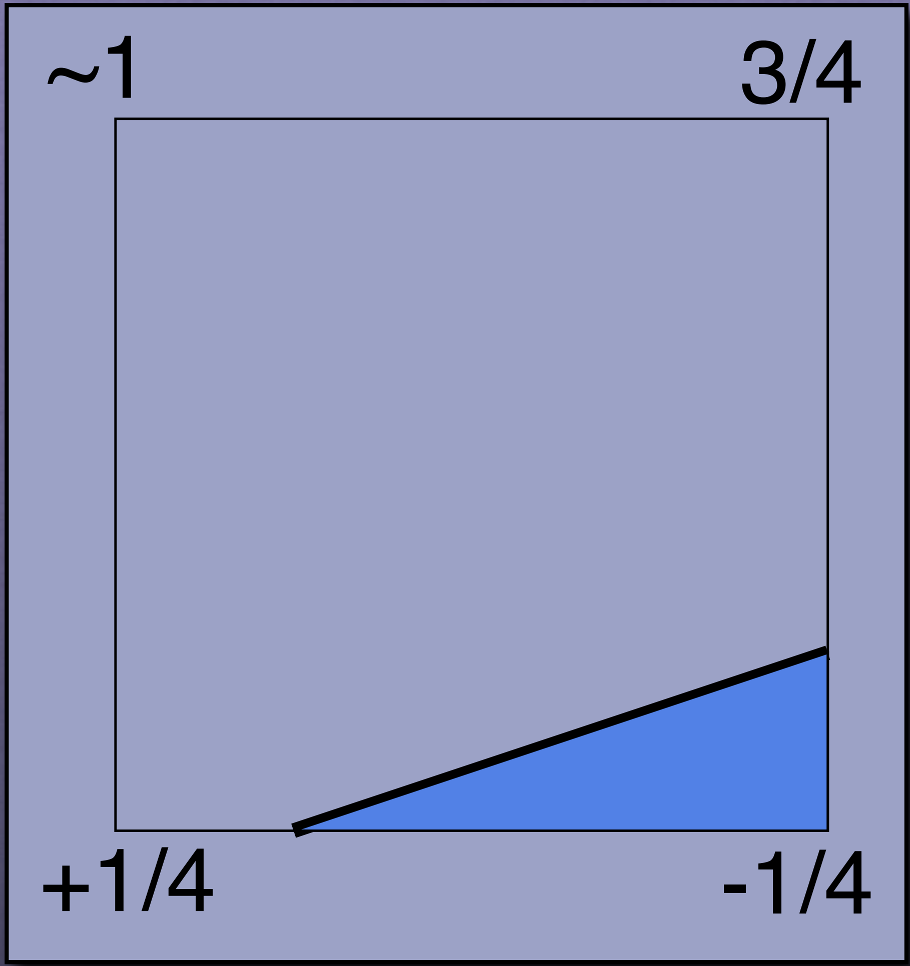


Building New Surface

- Compute velocities
  - e.g. Fluid simulation
- Build new octree
  - S.D. values are verts.
  - Trace back to old mesh
  - Adaptively refine
- Build new mesh
  - M.C. on new octree
- Re-distance octree

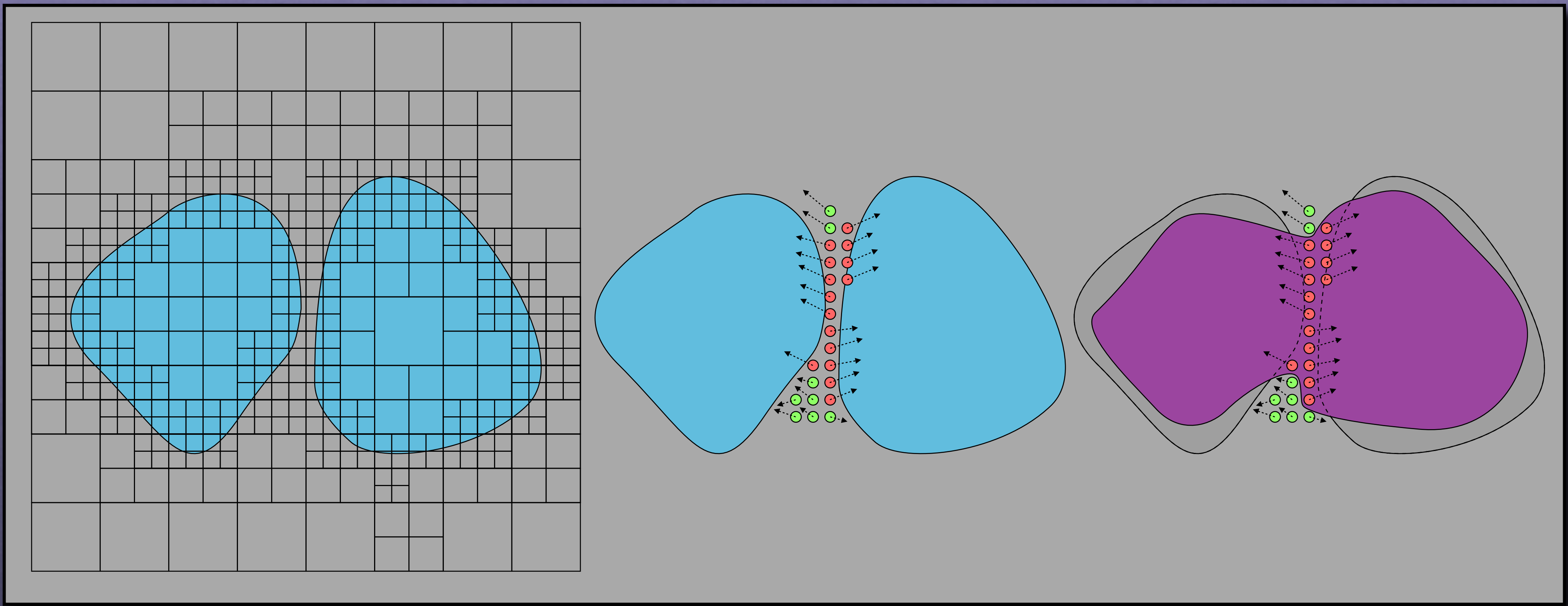


Why Exact Near Surface?



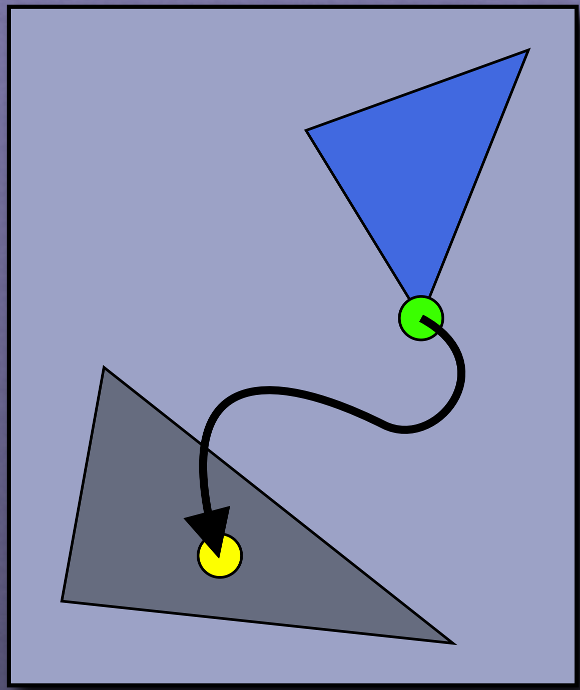
- Interpolation gets the wrong answer.

Surface Merging / Separating

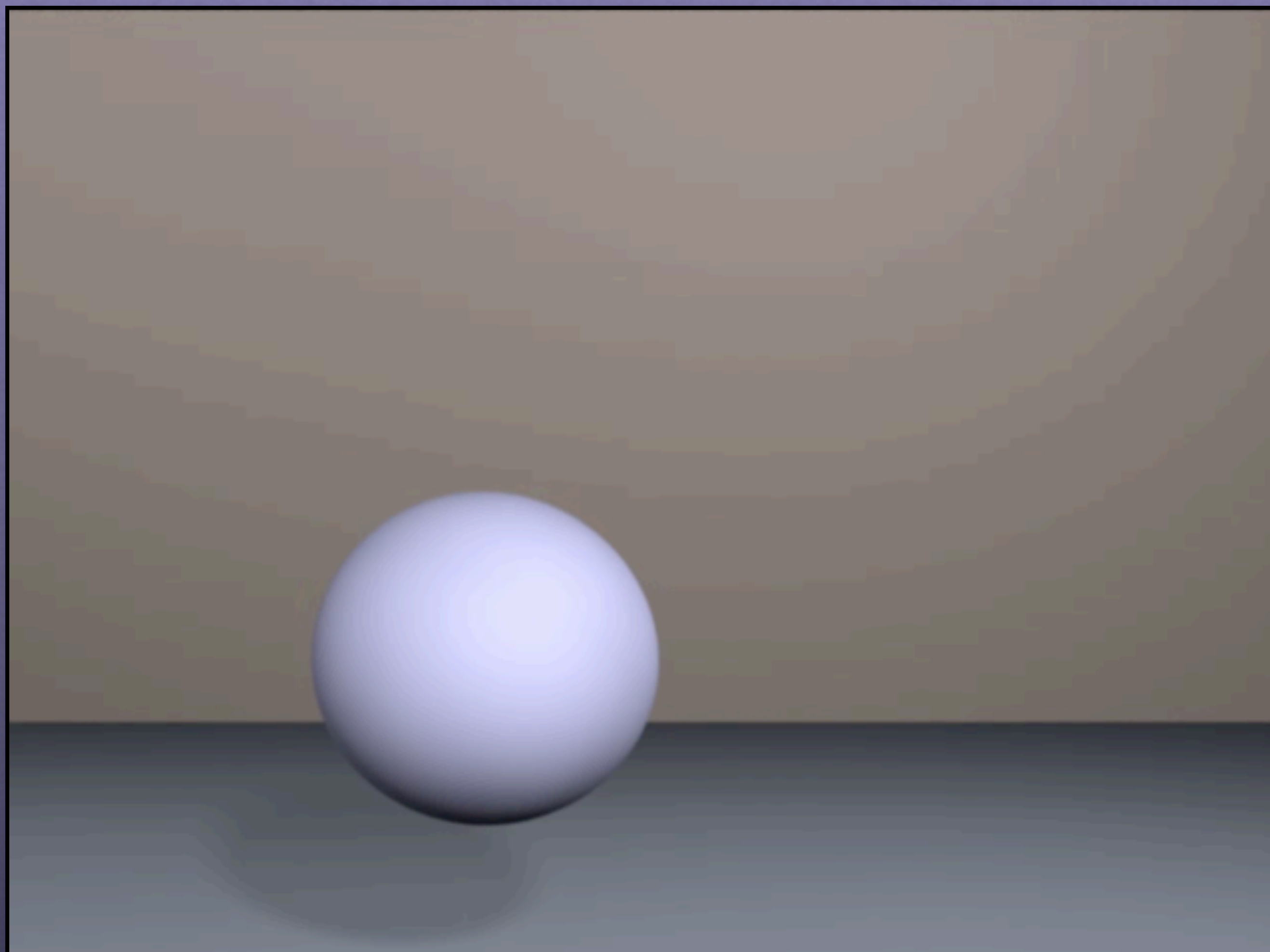


Tracking Surface Properties

- Semi-Lagrangian advection provides a mapping between surfaces at different timesteps.
- We can use this mapping to track surface properties.
- Surface signals get resampled at every step.



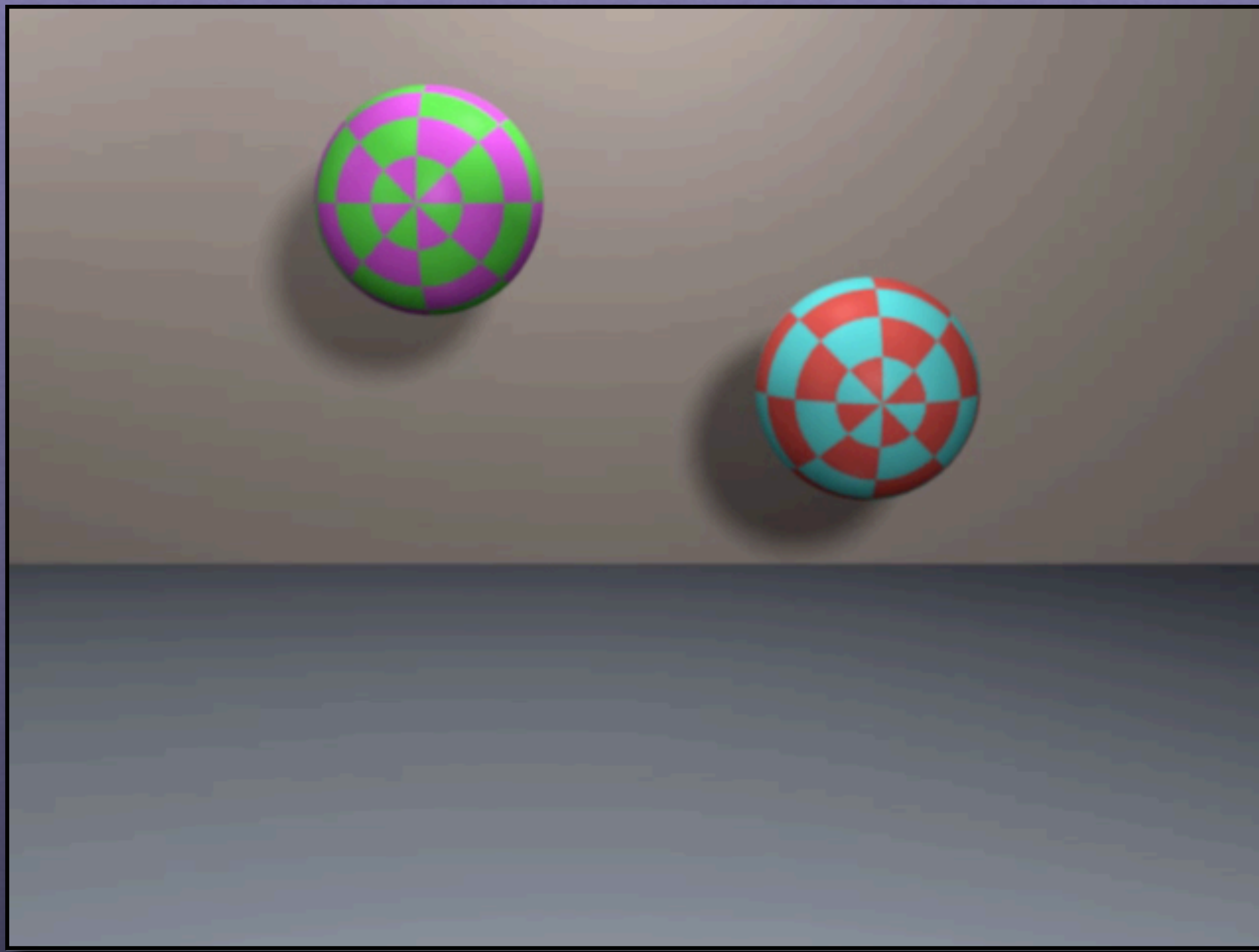
Results



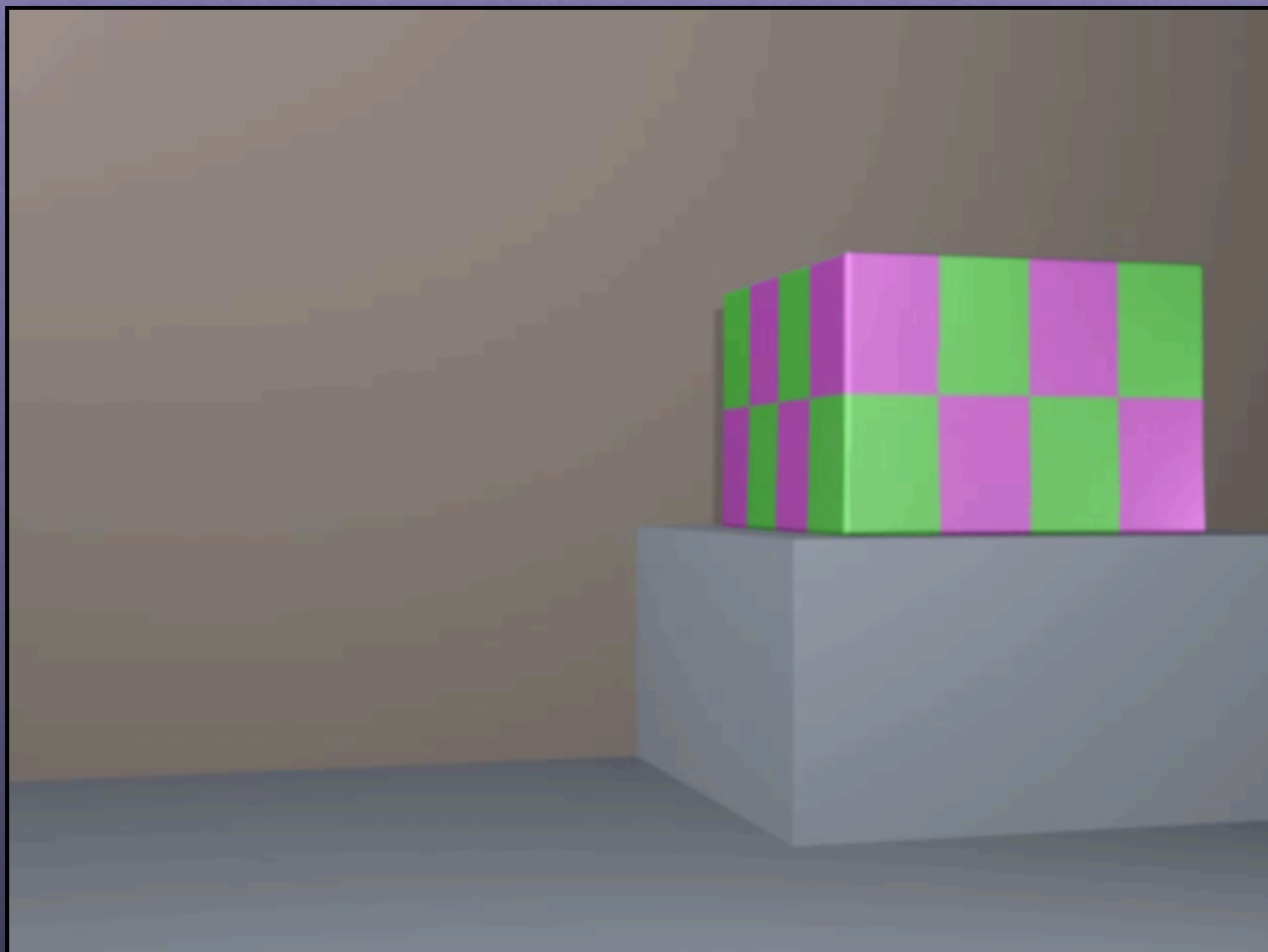
"Enright Test"

Results

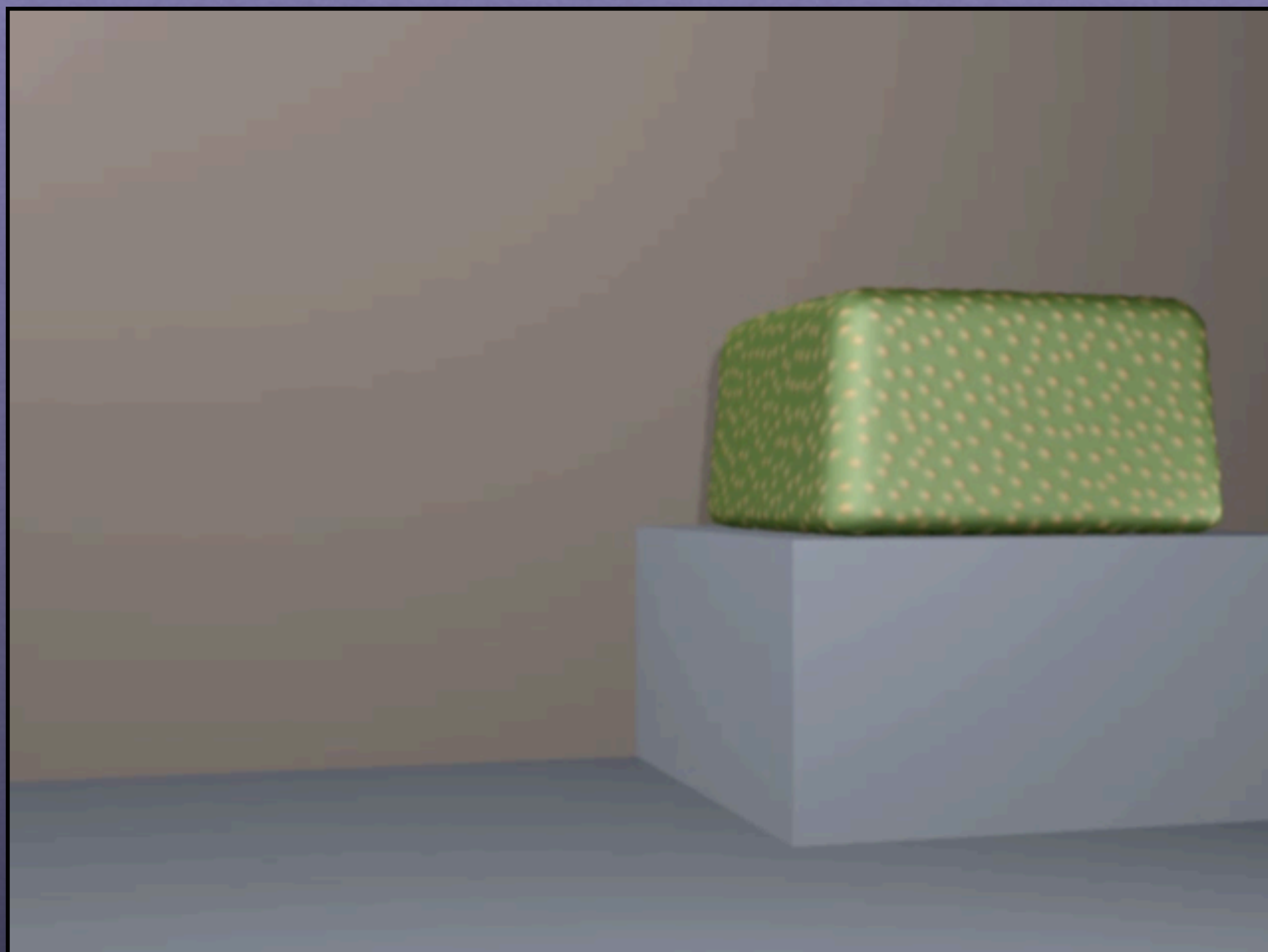
Results



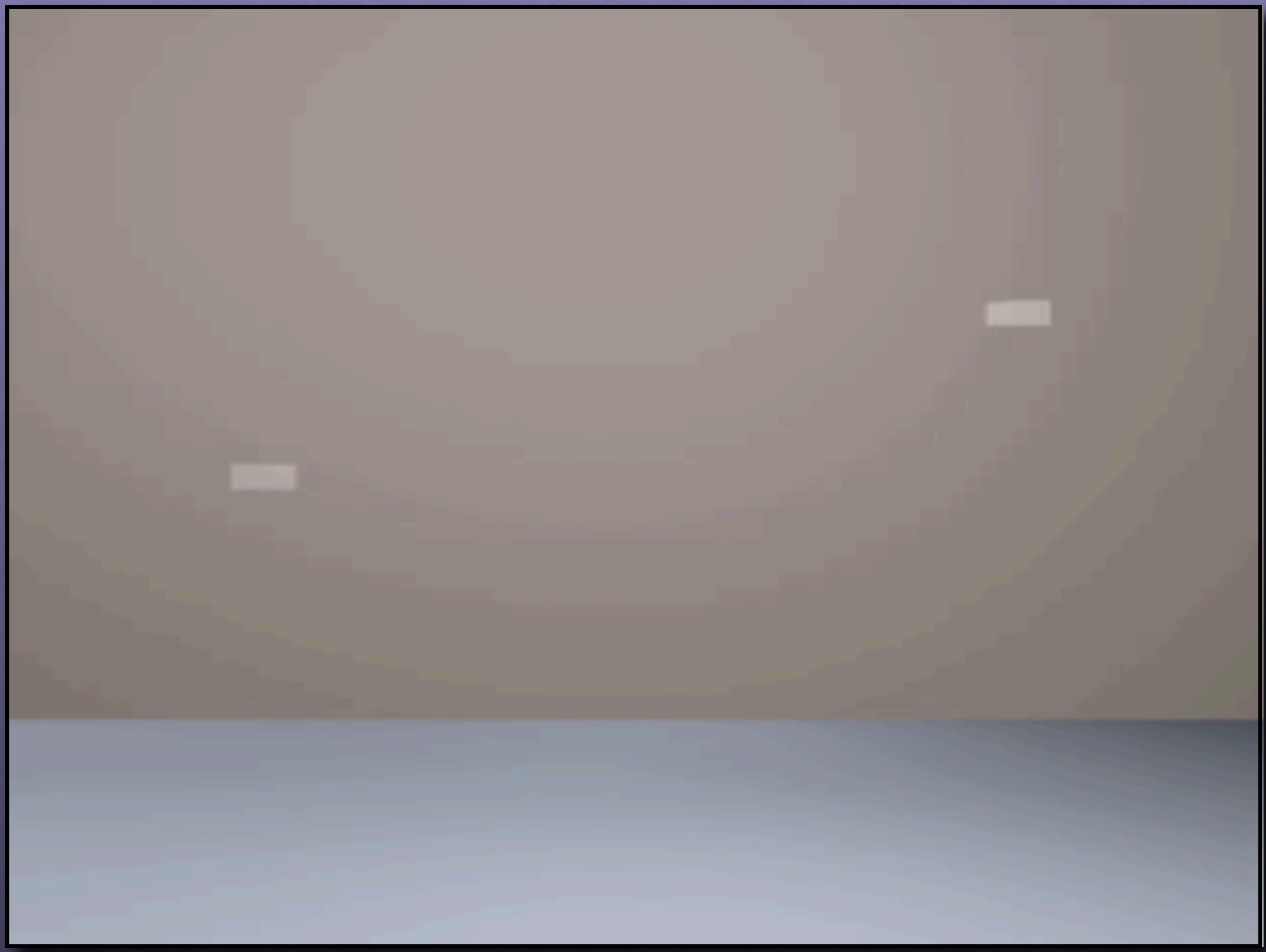
Results



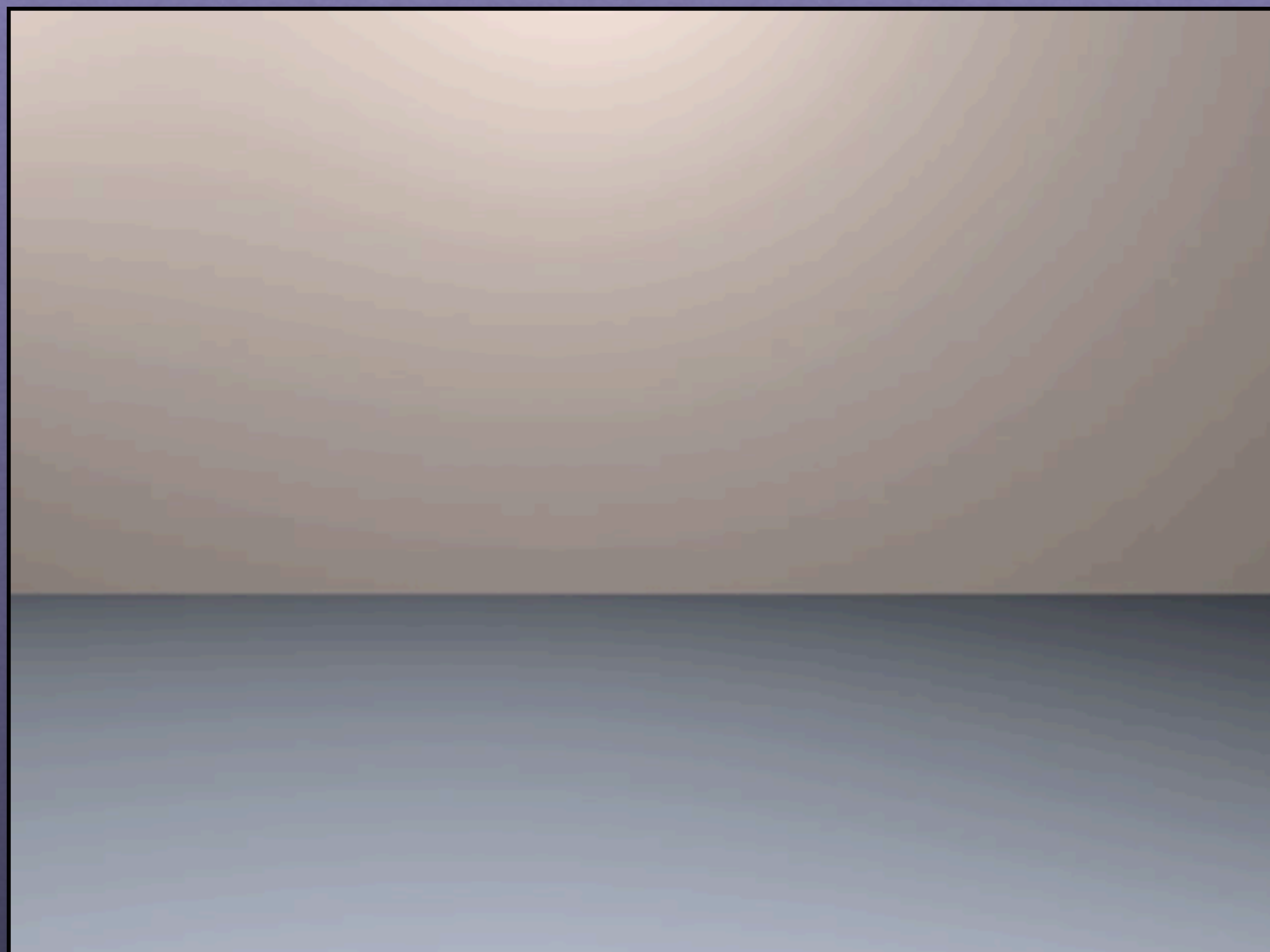
Results



Results

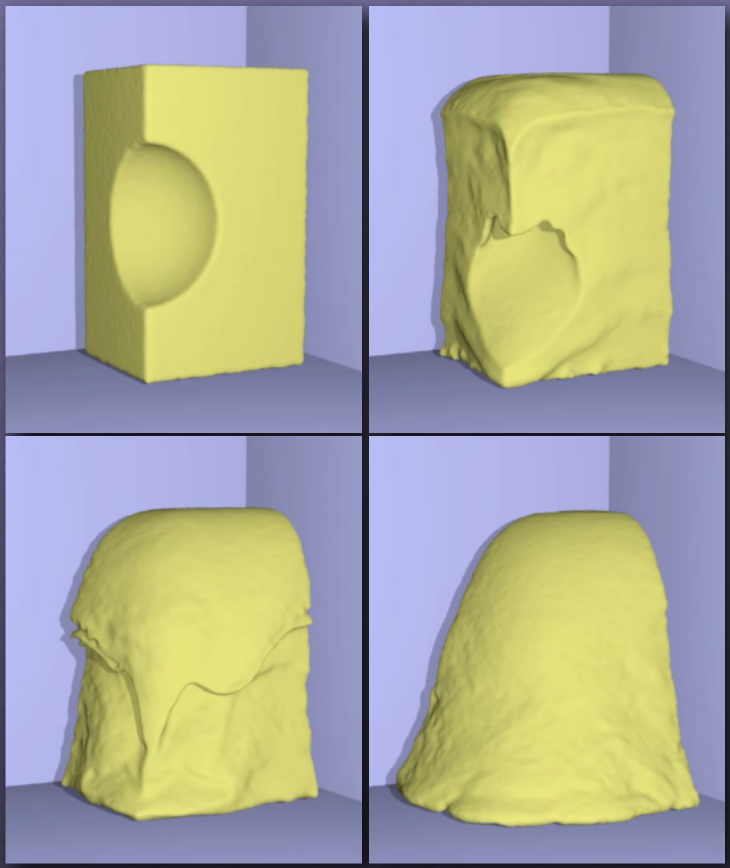


Results

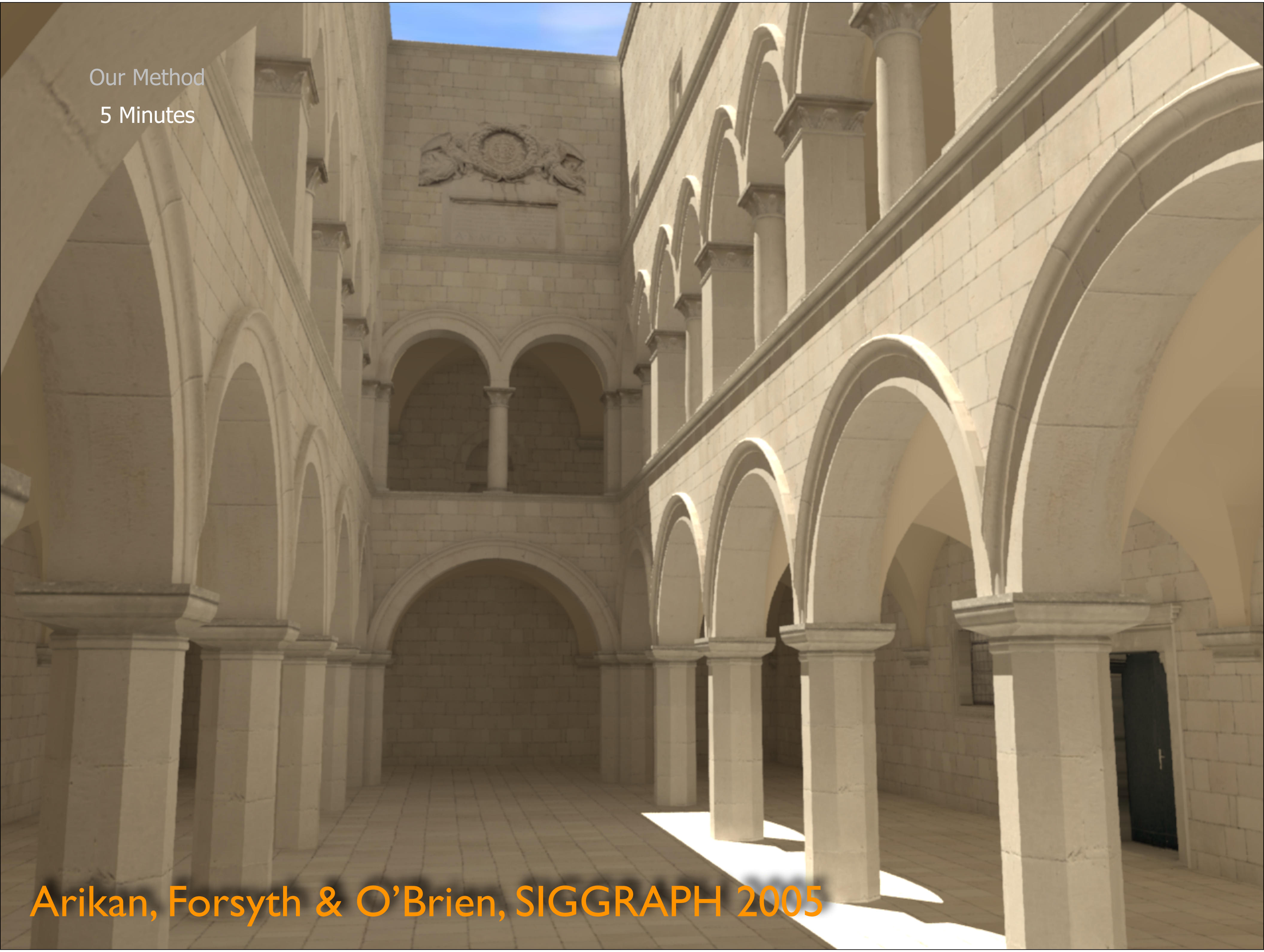


IMLS Surface Tracking?

- Zhu & Bridson SIGGRAPH 2005
  - Particles
  - MLS Blending
  - Cone point functions
  - Read their paper too!

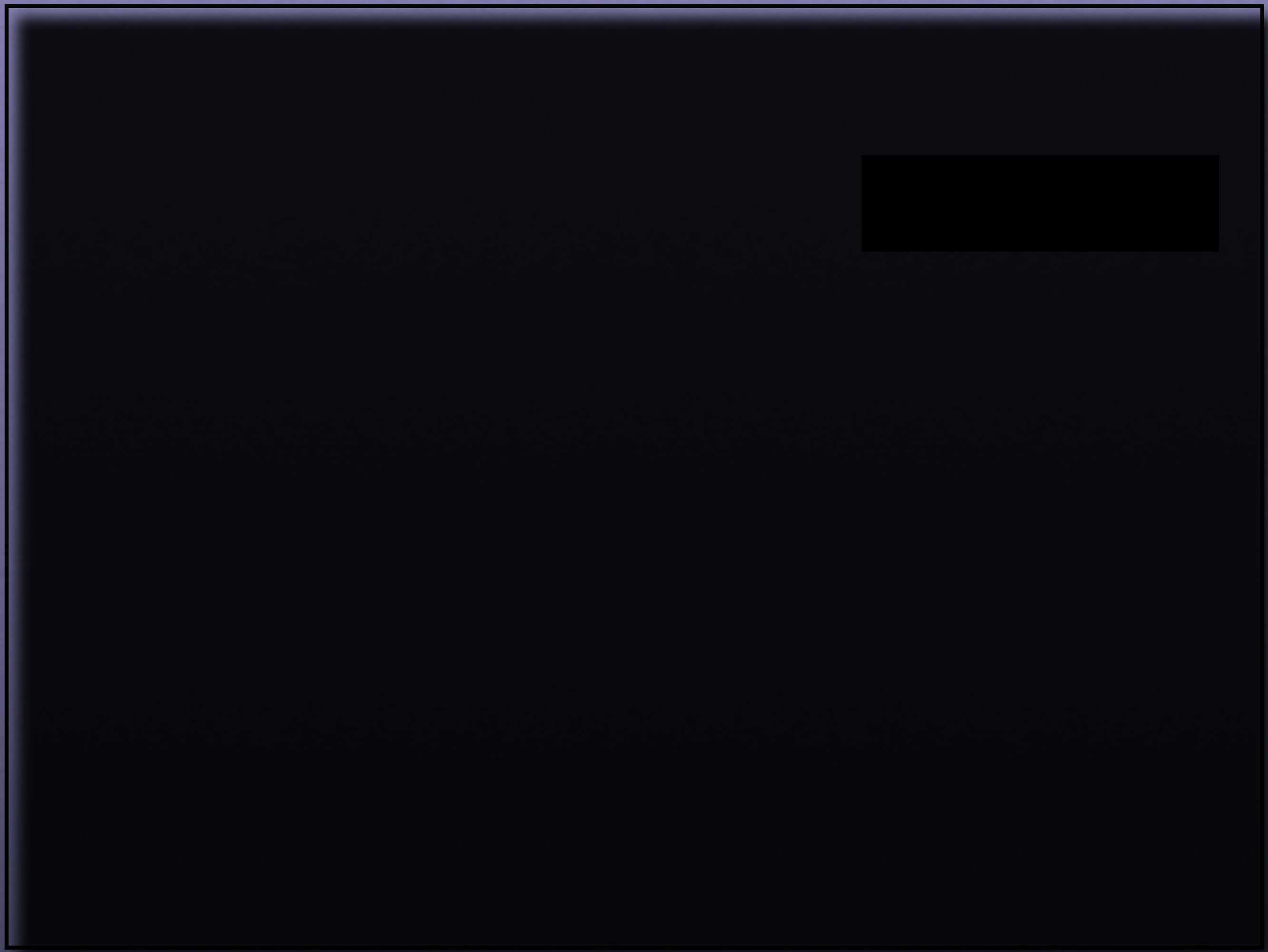



Ground Truth
(Monte Carlo)
24 Hrs.
Arikan, Forsyth & O'Brien, SIGGRAPH 2005


Our Method
5 Minutes
Arikan, Forsyth & O'Brien, SIGGRAPH 2005

Smoke on Hybrid Meshes

Feldman, O'Brien, Klingner, *SIGGRAPH 2005*



O’Brien, Hodgins, *SIGGRAPH 1999*

Sponsors

- Apple Computer
- Sony Computer Entertainment America
- Pixar Animation Studios
- Intel
- N.S.F.
- U.C. MICRO
- Okawa Foundation
- Hellman Family Faculty Fund
- Alfred P. Sloan Foundation