Session II: Contour Trees & Reeb Graphs

Overview

- Contour Tree Computation
- Reeb Graph Computation
- Contour Properties
- Simplification
- Contour Extraction
- Volume Rendering

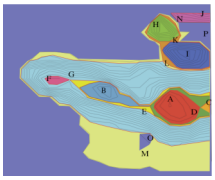
Equivalence Relation

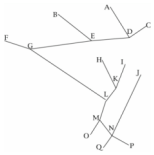
- A critical contour passes through a c.p.
- Let γ_1, γ_2 be two contours
  - Then $\gamma_1 \cong \gamma_2$ iff
    - there is a monotone path $p : \mathbb{R} \to \mathbb{R}^n$ s.t.

$$p \cap \gamma_1 \neq \varnothing \neq p \cap \gamma_2$$

$$p \cap \gamma = \varnothing \ \forall \text{ critical contours } \gamma$$

Equivalence Classes



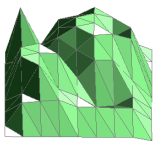


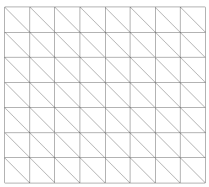
- Finite -> critical contours -> supernodes
- Infinite -> topological zones -> superarcs

Contour Tree Algorithms

- Monotone path extraction
  - Takahashi et al., 1995 (2D)
  - Takahashi et al., 2001 (3D, TR)
  - Takahashi et al., 2004 (3D)
- Contour sweep
  - van Kreveld et al., 1997 (2D, 3D, any dim.)
  - Tarasov & Vyalyi, 1998 (3D)
  - Pascucci, 2002 (3D)
- Sweep and merge
  - Carr et al., 2000 & 2003 (any dim.)
  - Pascucci & Cole-McLaughlin, 2002 (divide and conquer)
  - Chiang et al., 2003 (monotone path input)
  - Carr & Snoeyink, 2007 (arbitrary mesh)
- Reeb Graph
  - Shinagawa et al., 1991 (2-manifold)
  - Cole-McLaughlin, 2003 (2-manifold)
  - Pascucci et al., 2007 (streaming)

Contour Sweep





- Update (local) connectivity at each vertex

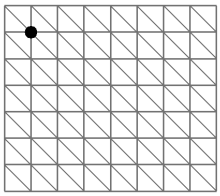
Sweep and Merge

- Three stages
  - Sweep downwards to compute *join tree*
  - Sweep upwards to compute *split tree*
  - Sweep inwards to merge join and split trees
- Simple treatment of multiple saddles / boundaries, higher dimensions
- Carr, Snoeyink & Axen, 2000/2004

Join Tree

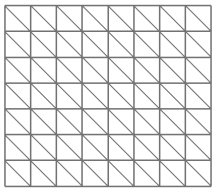
- Apply Union-Find to vertices in sorted order
  - Captures connectivity of set $\{x: f(x) \geq h\}$
- As we sweep, $\{x: f(x) \geq h\}$ gets bigger
  - And we can store the incremental connectivity
    - in the *join tree*

Join Sweep



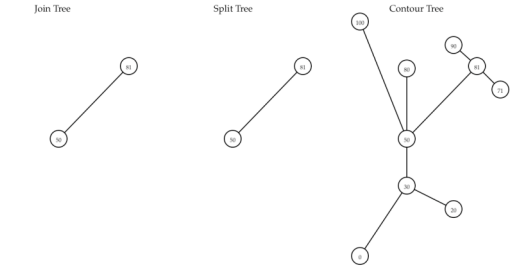
- Connectivity of $\{x: f(x) \geq h\}$

Split Sweep



- Connectivity of $\{x: f(x) \leq h\}$

Merge Phase



Non-Simplicial Meshes

- Algorithm is *graph*-based
  - processes simplex edges and vertices
- For other meshes, use any graph *G* s.t.
  - all critical points are in G
  - all monotone paths between critical points are in G
- Can even be extended to digital images

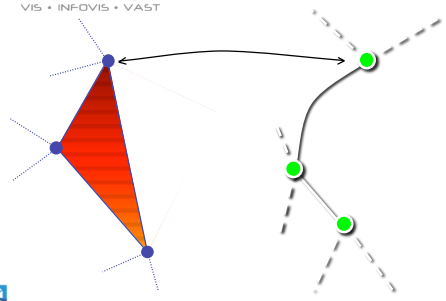
Suitable Graphs

- Contour Tree of cell (Pascucci & Cole-McLaughlin, 2003)
- Split / Join Trees of cell (Pascucci & Cole-McLaughlin, 2003)
- Fixed *widgets* in every cell (Carr & Snoeyink, 2007)
- *Dynamic* graph construction
  - using Finite State Machines (Carr & Snoeyink, 2007)

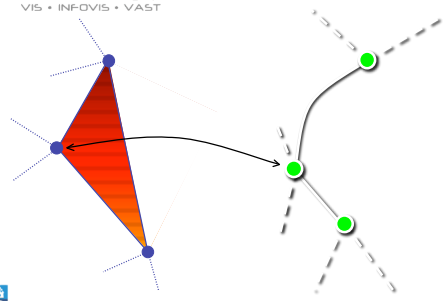
Streaming Reeb Graph Algorithm

- Pascucci et al., 2007
- *Zips* graph up one element at a time
  - i.e. vertices, edges & triangles
  - In any order desired
- Not asymptotically efficient
- But cache-friendly and *practically* efficient

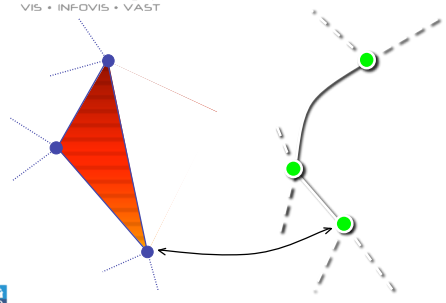
Reeb Graph Example



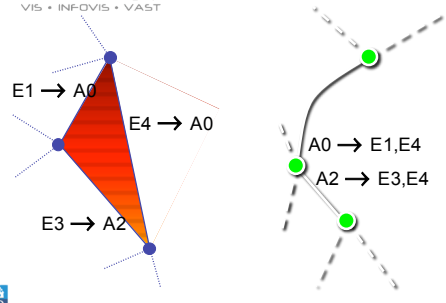
Reeb Graph Example

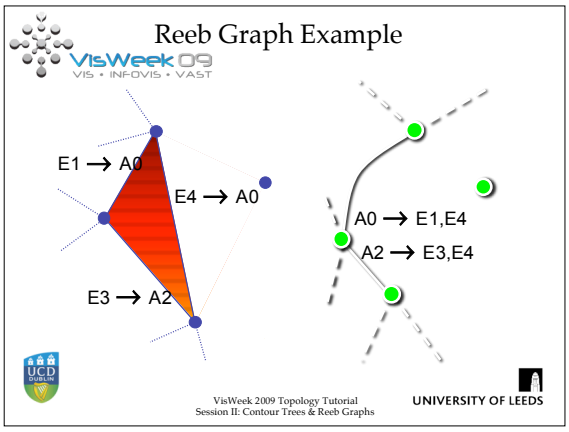


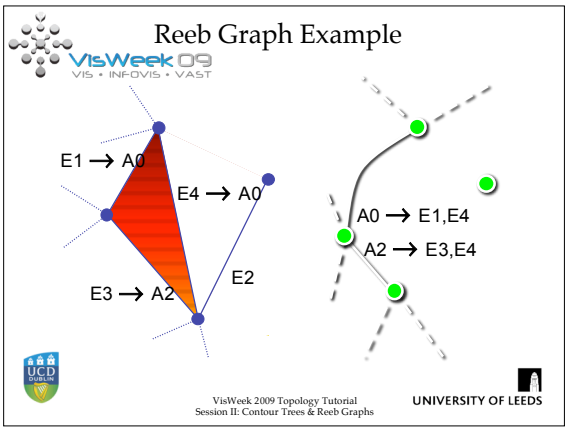
Reeb Graph Example

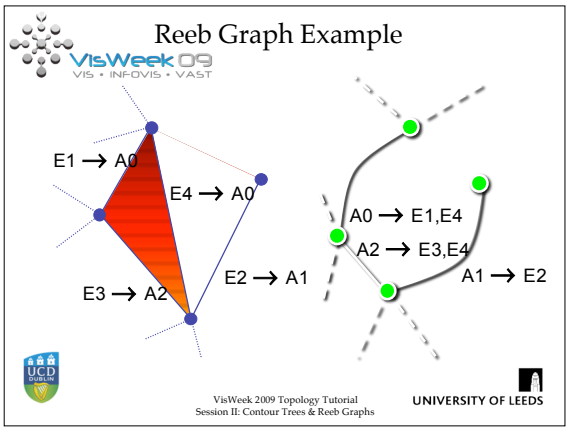


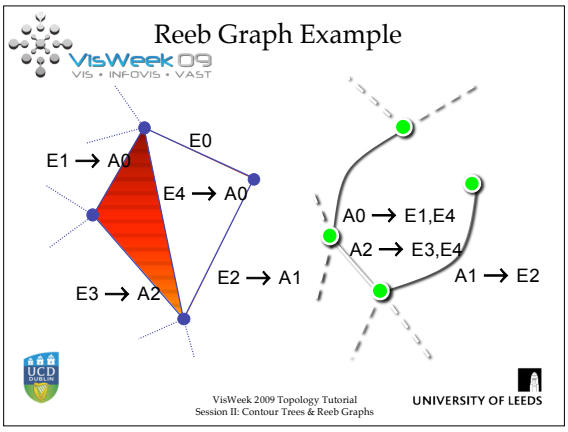
Reeb Graph Example

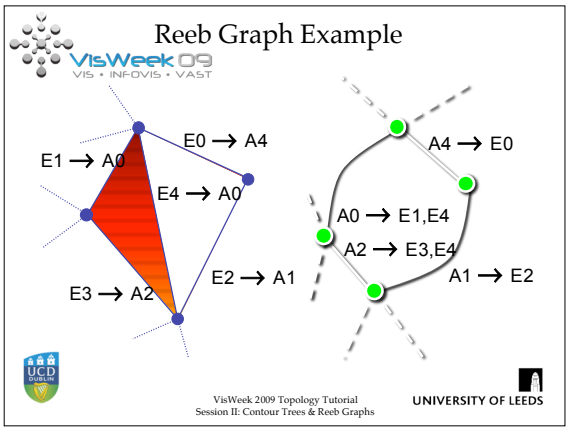


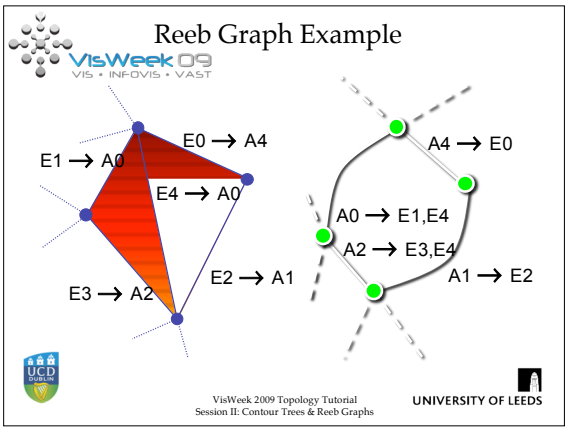

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E1 → A0
E4 → A0
E3 → A2
A0 → E1,E4
A2 → E3,E4
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

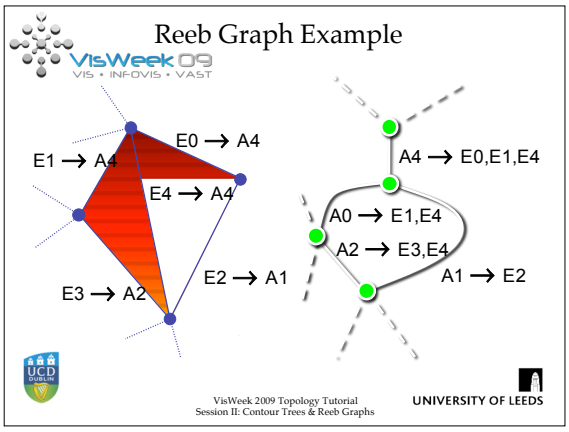

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E1 → A0
E4 → A0
E2
E3 → A2
A0 → E1,E4
A2 → E3,E4
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

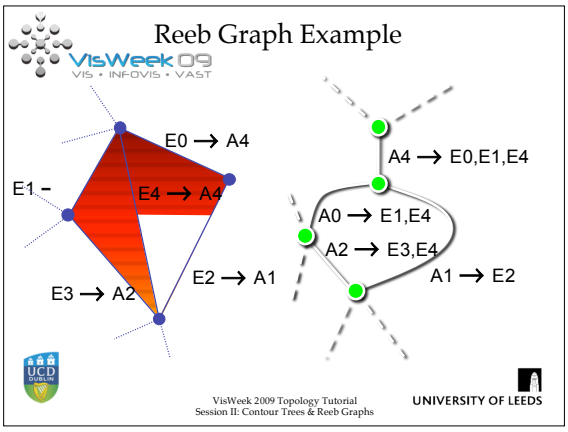

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E1 → A0
E4 → A0
E2 → A1
E3 → A2
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

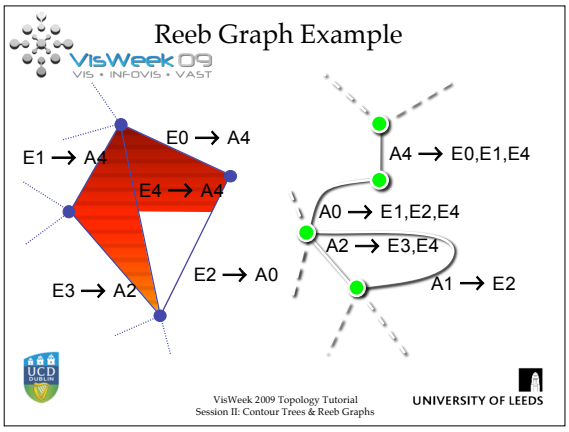

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0
E1 → A0
E4 → A0
E2 → A1
E3 → A2
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

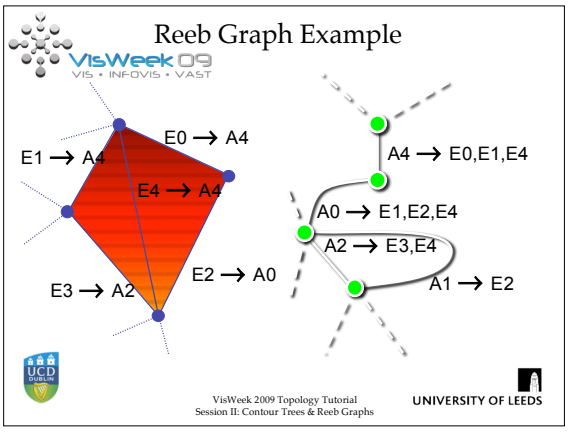

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A0
E4 → A0
E2 → A1
E3 → A2
A4 → E0
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

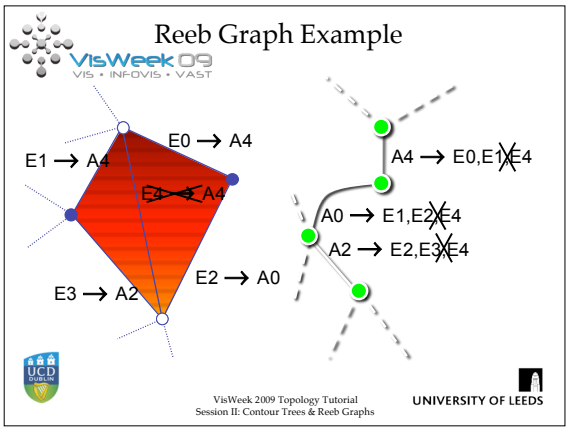

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A0
E4 → A0
E2 → A1
E3 → A2
A4 → E0
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

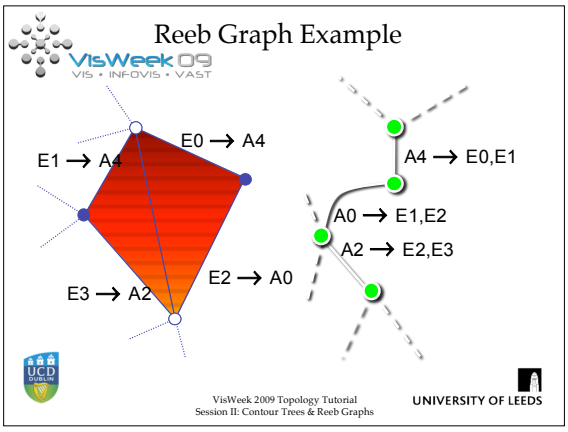

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A4
E4 → A4
E2 → A1
E3 → A2
A4 → E0,E1,E4
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS


Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 –
E4 → A4
E2 → A1
E3 → A2
A4 → E0,E1,E4
A0 → E1,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

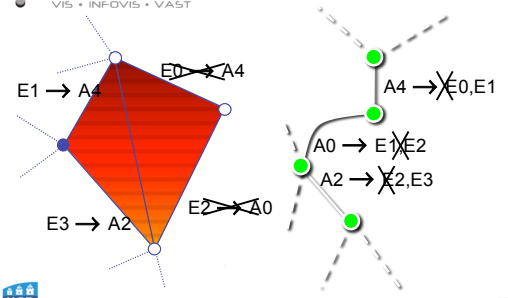

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A4
E4 → A4
E2 → A0
E3 → A2
A4 → E0,E1,E4
A0 → E1,E2,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS


Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A4
E4 → A4
E2 → A0
E3 → A2
A4 → E0,E1,E4
A0 → E1,E2,E4
A2 → E3,E4
A1 → E2
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

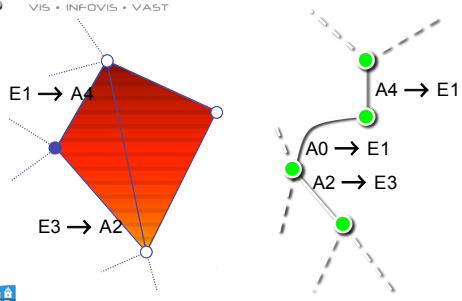

Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A4
E4 → A4
E2 → A0
E3 → A2
A4 → E0,E1,E4
A0 → E1,E2,E4
A2 → E2,E3,E4
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS


Reeb Graph Example
VisWeek 09
VIS • INFOVIS • VAST
E0 → A4
E1 → A4
E2 → A0
E3 → A2
A4 → E0,E1
A0 → E1,E2
A2 → E2,E3
VisWeek 2009 Topology Tutorial
Session II: Contour Trees & Reeb Graphs
UNIVERSITY OF LEEDS

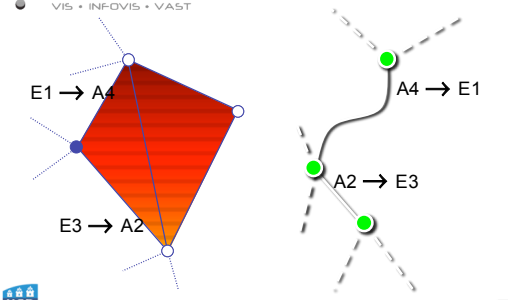
Reeb Graph Example



Reeb Graph Example



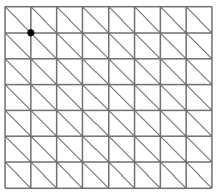
Reeb Graph Example



Contour Properties

- Length, area, volume, hypervolume
- Are all *functions* of the isovalue
  - Contour Spectrum, Bajaj, Pascucci & Schikore, 1997
- Genus, Betti #'s
  - Pascucci, 2001
- Can be computed for *each* contour
  - the merge phase is a series of sweeps
  - Carr, Snoeyink & van de Panne, 2004

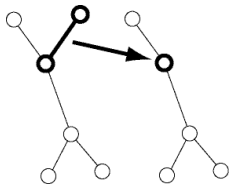
Illustration: Merge Sweep



- Merge phase uses isovalue sweeps

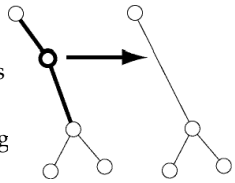
Tree Simplification

- Each peak is a leaf
  - so is each pit
- Choose smallest
- Repeat until reach
  - fixed size
  - fixed geometric property



Tree Pruning

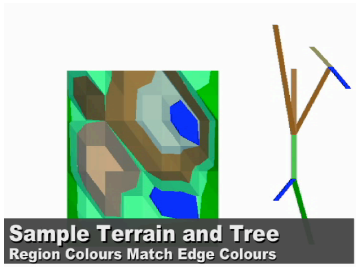
- May get regular nodes
  - reduce immediately
  - builds path from edg



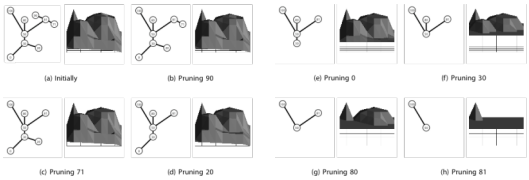
Effects of Pruning

- 1) Cut the region from the domain
- 2) Contract the region to the saddle
- 3) Flatten the region (cuts off the peak)
- 4) Flatten vertices inside the region
  - simplicial meshes only

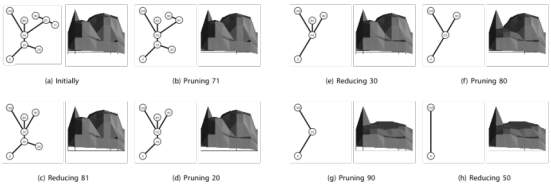
Pruning Example



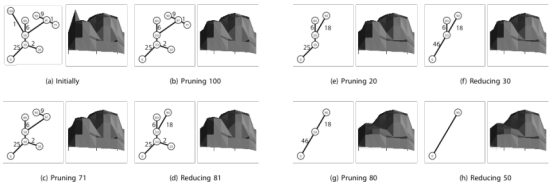
Simplify by Height



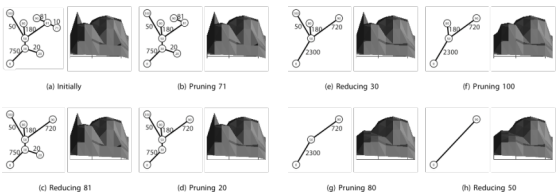
Simplify by Height, II



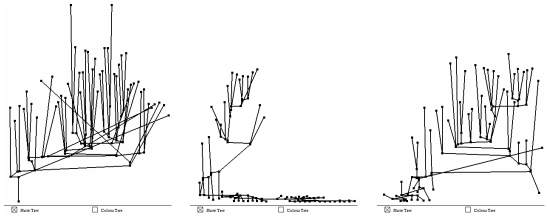
Simplify by Area



Simplify by Volume

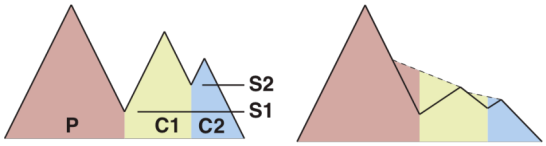


Comparison for CT-Head



Height (Persistence) | Volume (Sample Count) | Hypervolume (Sample Sum)

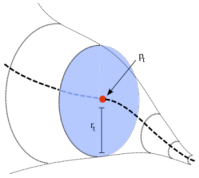
More Simplification



(a) Unsimplified (b) Simplified

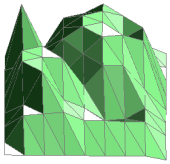
- *Reverse* the gradient
  - Weber & al., TVCG 2007

Tube Sweep

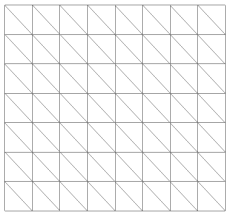


- Sweep a *ball* along gradient path
  - ball size is heuristic (so far)

Contour Extraction

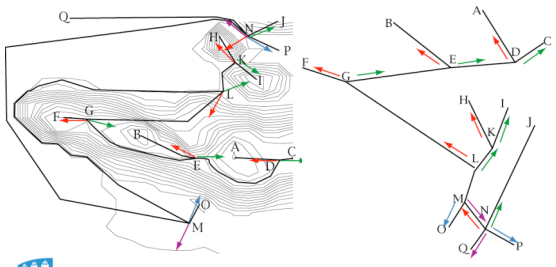


Continuation

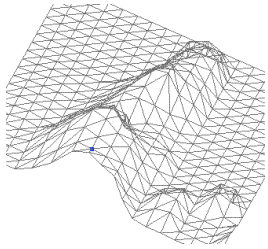


- Flood fill along contours, Wyvill & al., 1986

Path Seeds



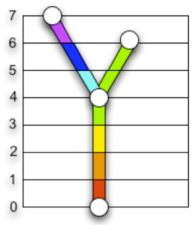
Path Seeds

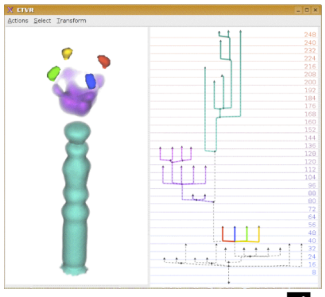


Path Seeds in 3D

- Of course, this also works in 3D

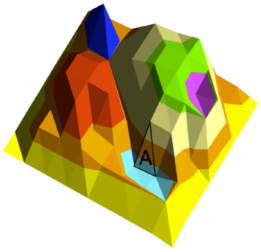
Zone-specific TF's





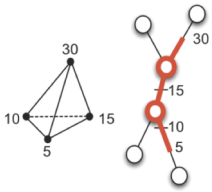
Which Zone

- Which Zone does tip of *A* belong to?

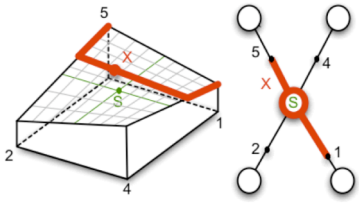


Mapping Paths

- Exploit monotone paths
- path in *f* <=> path in tree



Bi-/Tri-linear Cells



- Axis-aligned segments are monotone