VisWeek 08
VIS • INFOVIS • VAST

Edge Groups: An Approach to Understanding the Mesh Quality of Marching Methods

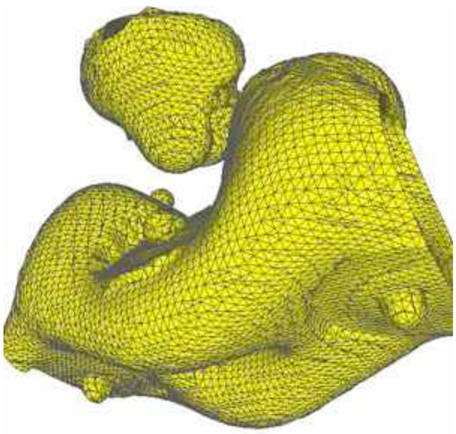
Carlos A. Dietrich, Carlos Scheidegger
João L. D. Comba, Luciana Porcher Nedel, Cláudio T. Silva

UFRGS

SCI Institute

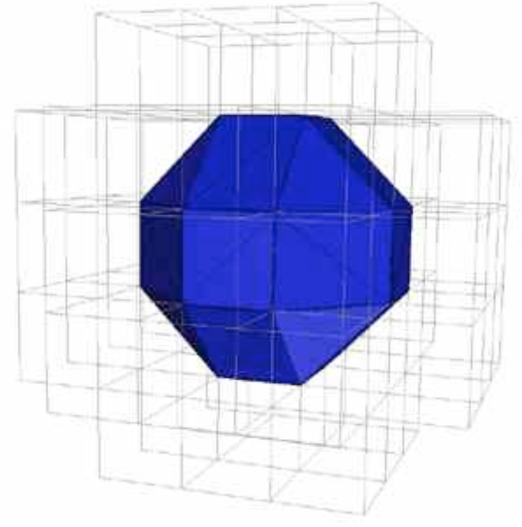
Polygonal Approximation of Isosurfaces (Meshes)

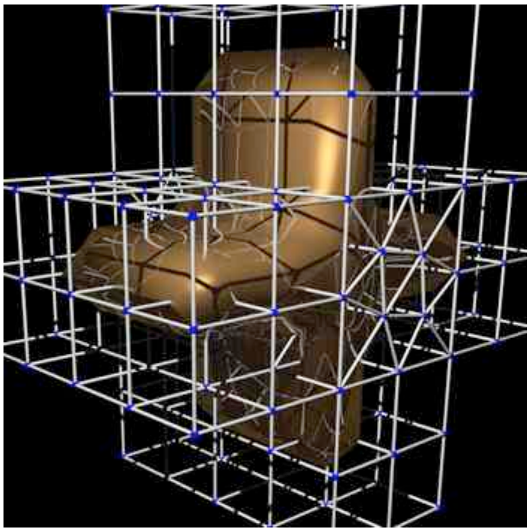
- Possible resulting meshes:
  - Continuous patches
  - Often closed
  - More than one connected component
  - Holes
  - Ambiguous Cases



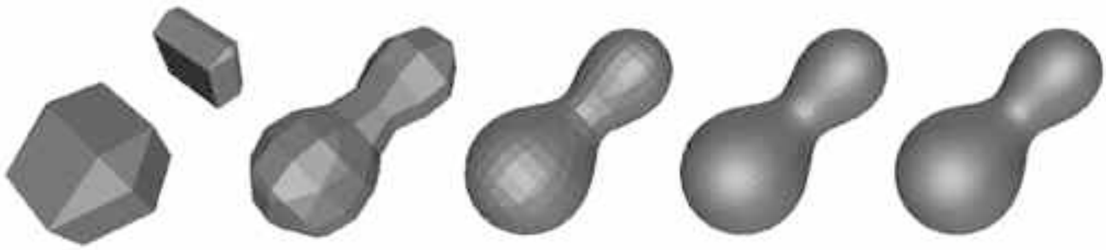
How to do it ?

Divide-and-Conquer





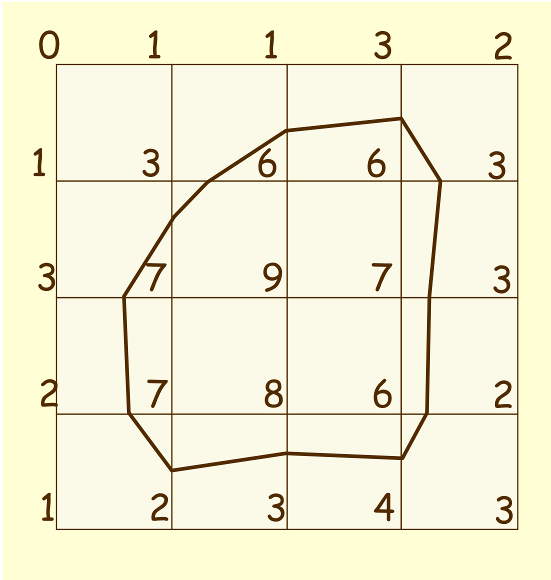
Divide-and-Conquer



Grid size=10 70 Facets | Grid size=5 220 Facets | Grid size=2 1700 Facets | Grid size=1 6800 Facets | Grid size=0.5 27000 Facets

2D Problem

- Consider 5x5 image, isovalue = 5



2D Problem

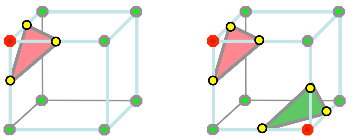
- Consider rectangular cells individually
- Classify each vertex as inside or outside the isosurface (polarity test)
  - How many cases can we have ? How about in 3D ?
- How to compute Intersections ?
- How to maintain continuity ?
- How to implement ? Easy and Fast way ?

Marching Cubes

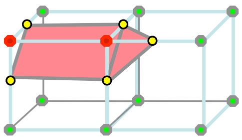
- For each cell:
  - Polarity Test (classify inside/outside each vertex)
  - Create a case index
  - Get list of edges with intersection from table
  - Compute intersections using interpolation

Marching Cubes

- Isosurface extraction from voxels [Lorensen 87]
  Assumes Linear Interpolation between data



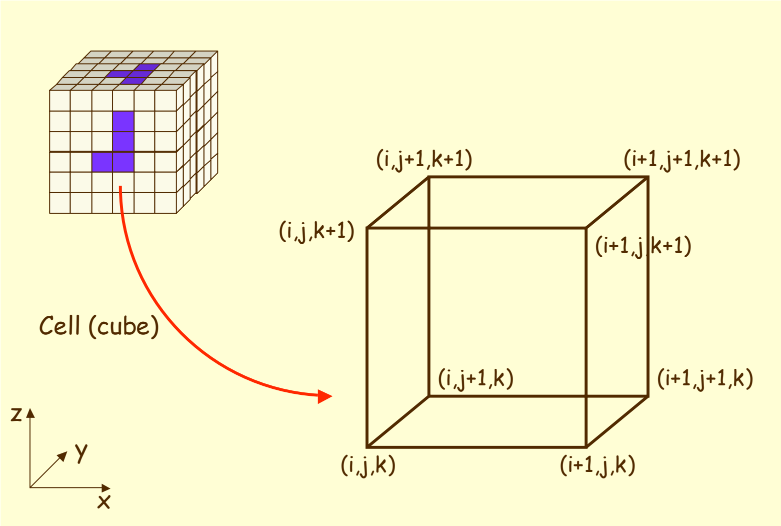
- Mesh extraction
  Connect surface intersections



- Render Mesh using traditional techniques

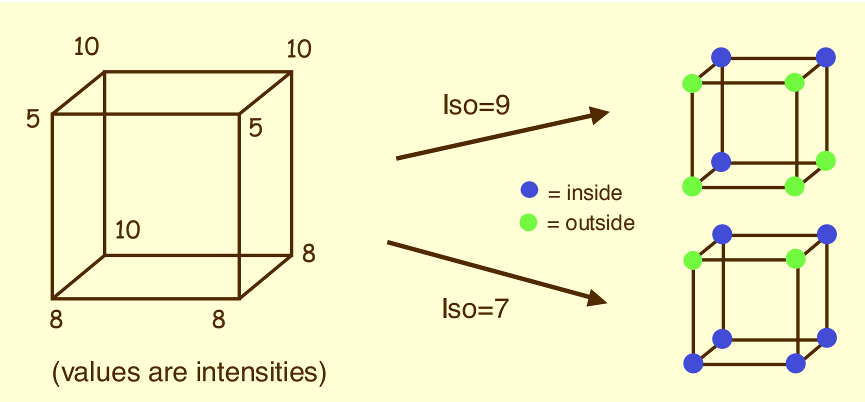
Marching Cubes

- For each voxel:



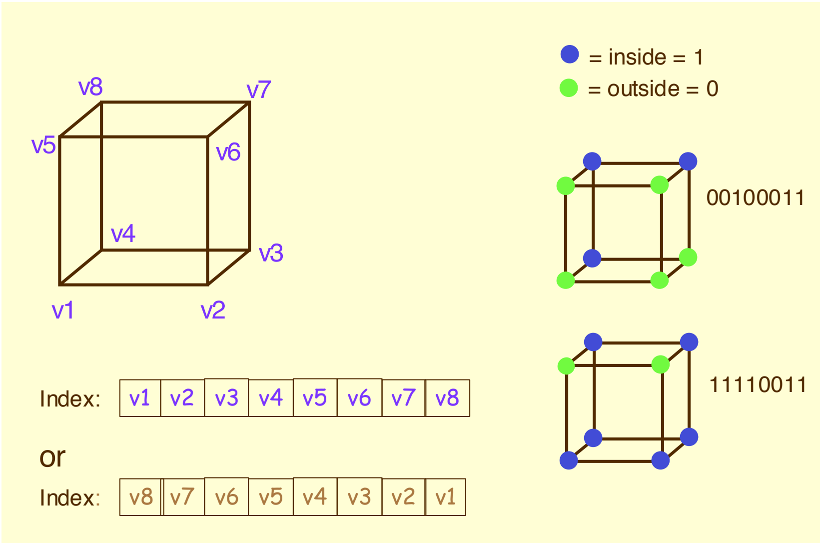
Marching Cubes

- Classify the state of vertices (vertex index)



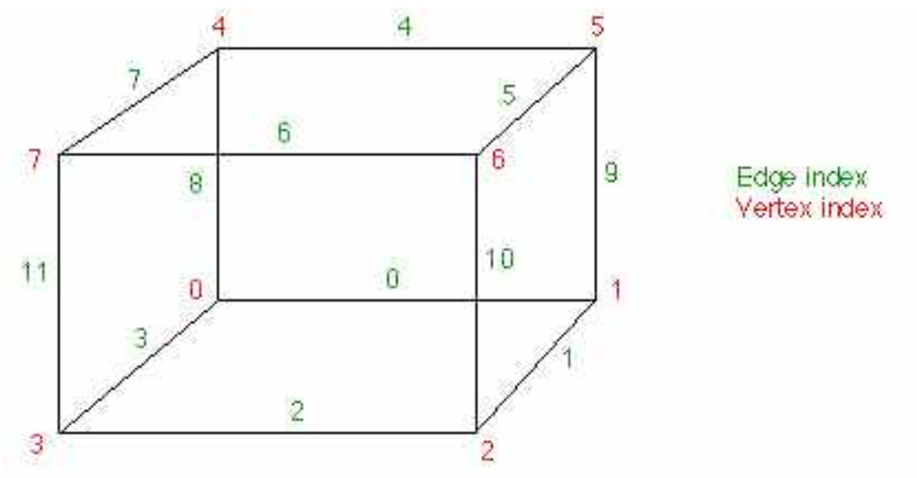
Marching Cubes

- Create Vertex Index



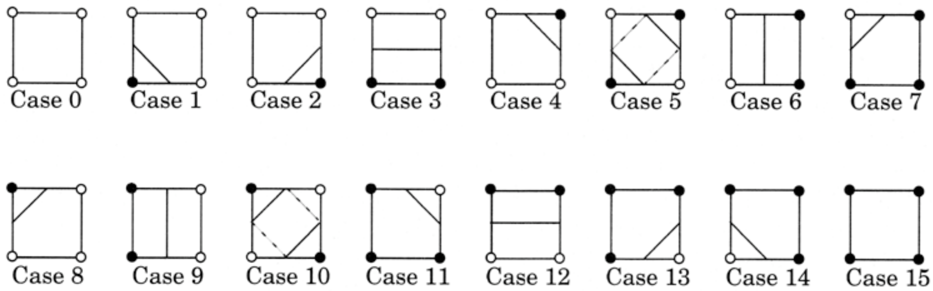
Marching Cubes

- Vertex Indexes and Edge Indexes

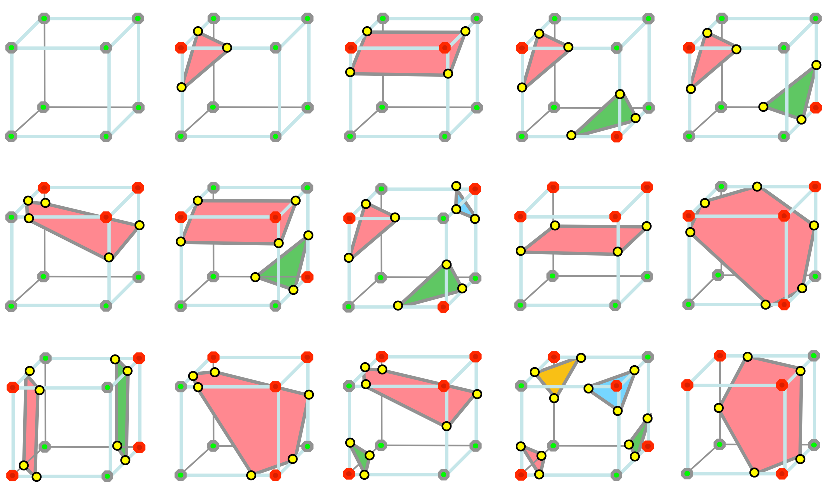


Marching Cubes

- Create Edge List
  - Use an edge index to encode the list of edges cut by the isosurface
  - 2D cell index: 4 bits (2^4 cases)



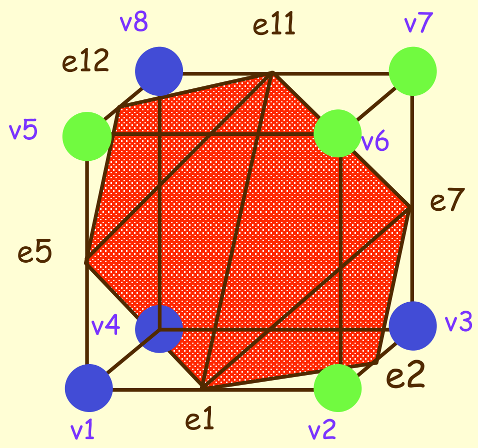
Marching Cubes Cases



Marching Cubes

- Edge list example:

- Index = 10110001
  - triangle 1 = e5, e11, e12
  - triangle 2 = e1, e11, e5
  - triangle 3 = e1, e7, e11
  - triangle 4 = e1, e2, e7



Marching Cubes

- Example:
  - Vertex 0 and 3 are below the isosurface
    - Vertex index =

Marching Cubes

- Example:
  - Vertex 0 and 3 are below the isosurface
    - Vertex index = 0000 1001 = 9

```
cubeindex = 0;
   if (grid.val[0] < isolevel) cubeindex |= 1;
   if (grid.val[1] < isolevel) cubeindex |= 2;
   if (grid.val[2] < isolevel) cubeindex |= 4;
   if (grid.val[3] < isolevel) cubeindex |= 8;
   if (grid.val[4] < isolevel) cubeindex |= 16;
   if (grid.val[5] < isolevel) cubeindex |= 32;
   if (grid.val[6] < isolevel) cubeindex |= 64;
   if (grid.val[7] < isolevel) cubeindex |= 128;
```

Marching Cubes

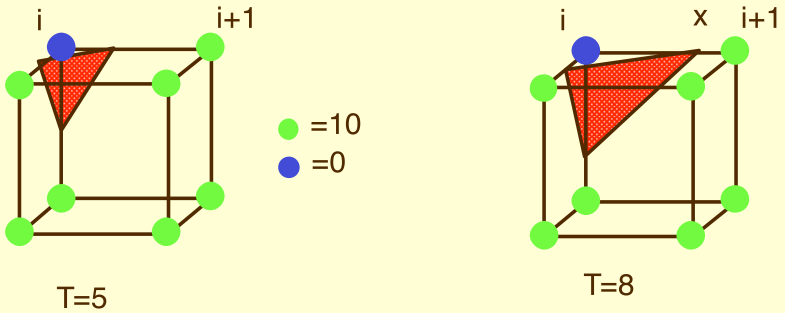
- Example:
  - Vertex 0 and 3 are below the isosurface
    - Vertex index = 0000 1001 = 9
    - EdgeTable[9] = 1001 0000 0101 (905 hex)
      - edges 11, 8, 2 and 0 are cut, compute intersection vertices
    - Triangle table (up to 5 triangles):
      - TriangleTable[9] = {0,11,2,8,11,0,-1,-1,...,-1}

Marching Cubes

- For each triangle, find an vertex location along the edge using linear interpolation of the values at the edge's two end points (*voxels*)

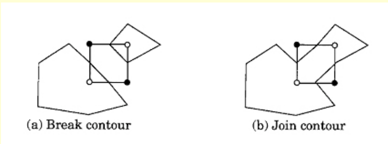
$$x = i + \left(\frac{T - v[i]}{v[i+1] - v[i]} \right)$$

(T is iso-value)

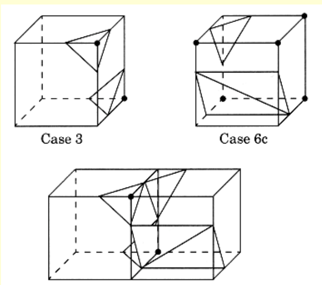


Marching Cubes

- Ambiguous Cases
  - 2D cases
    - Break contour
    - Join contour
  - Both are valid



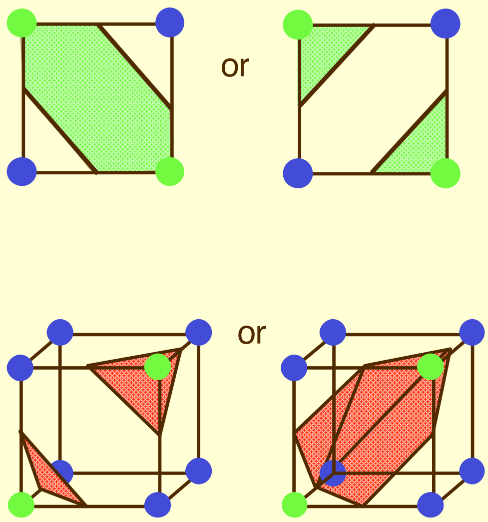
  - 3D cases: may generate a hole (must be eliminated)



Marching Cubes

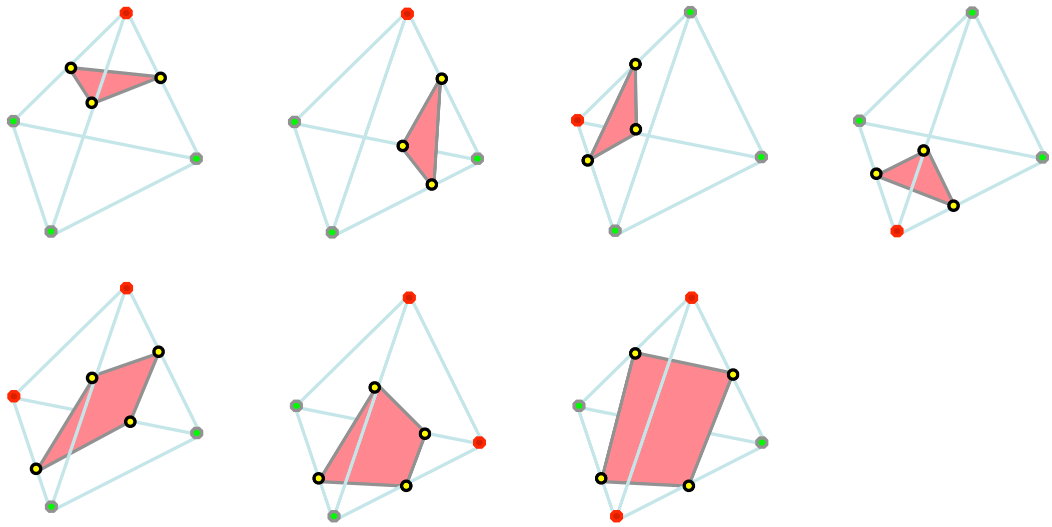
- Ambiguous Cases

  - Ambiguous cases: 3, 6, 7, 10, 12, 13
  - Adjacent vertices: different states
  - Diagonal vertices: same state
  - Need to decide which case is correct



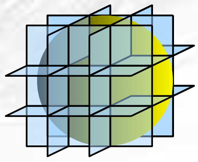
Marching Tetrahedra

- [Doi an Koide 91]



Marching Cubes - (Lorensen/Cline, 1987) 5178 citations 10/29/2008

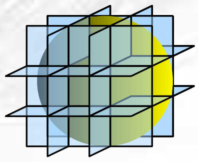
- Code is simple, efficient and produces stable results
- Table-driven implementation

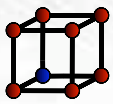


compute vertex polarity
↓
identify cell case → triangulation table
↓
identify active edges
↓
generate intersection vertices
↓
generate triangulation

Marching Cubes - (Lorensen/Cline, 1987) 5009 citations 9/10/2008

- Code is simple, efficient and produces stable results
- Table-driven implementation

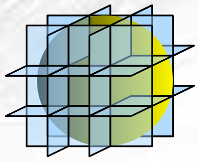


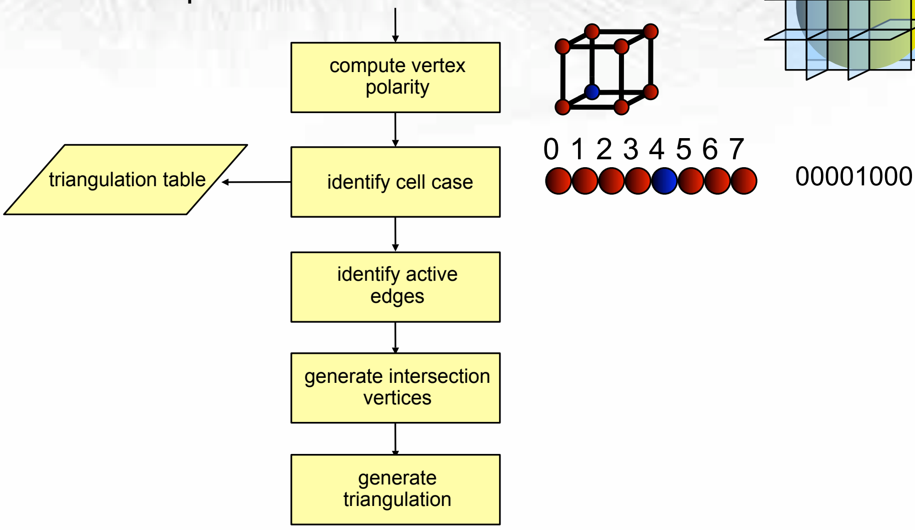


compute vertex polarity
↓
identify cell case → triangulation table
↓
identify active edges
↓
generate intersection vertices
↓
generate triangulation

Marching Cubes - (Lorensen/Cline, 1987) 5009 citations 9/10/2008

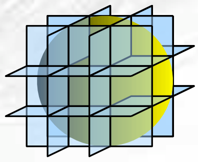
- Code is simple, efficient and produces stable results
- Table-driven implementation

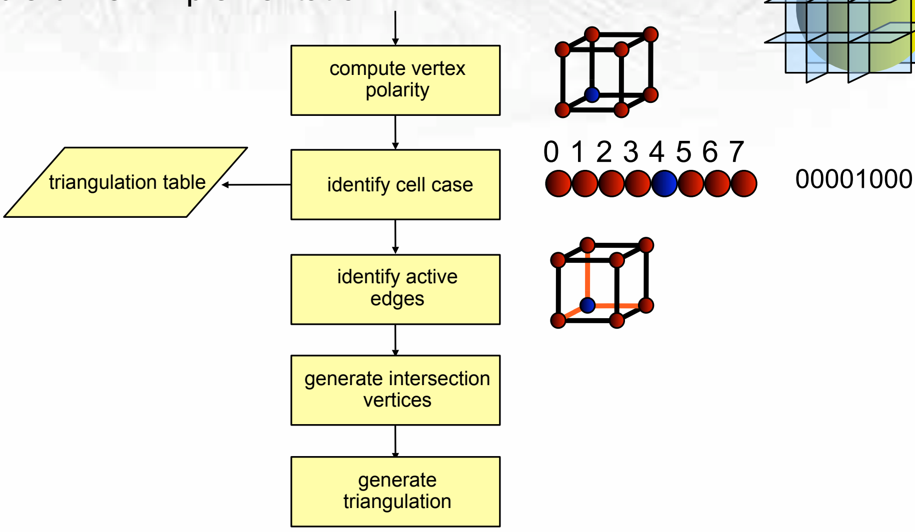




Marching Cubes - (Lorensen/Cline, 1987) 5009 citations 9/10/2008

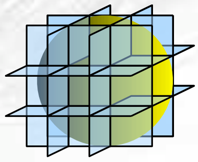
- Code is simple, efficient and produces stable results
- Table-driven implementation

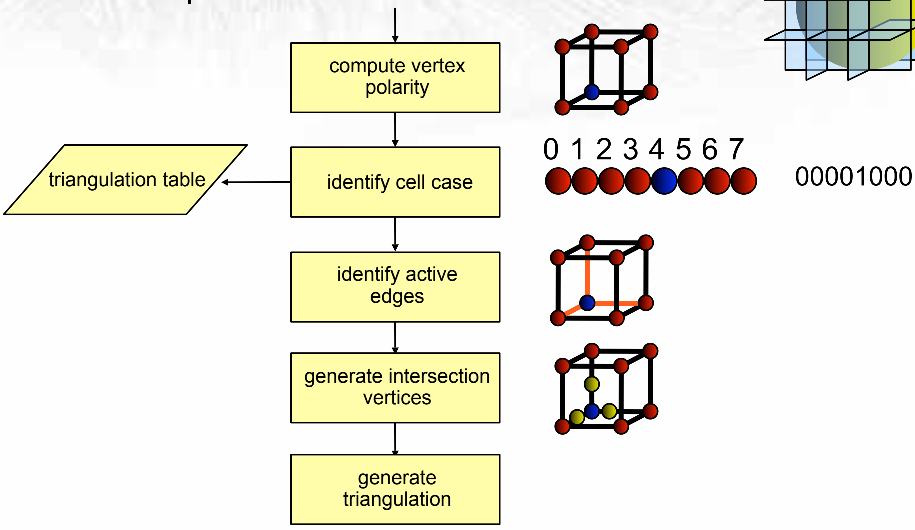




Marching Cubes - (Lorensen/Cline, 1987) 5009 citations 9/10/2008

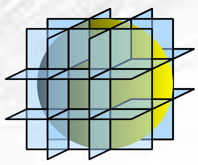
- Code is simple, efficient and produces stable results
- Table-driven implementation

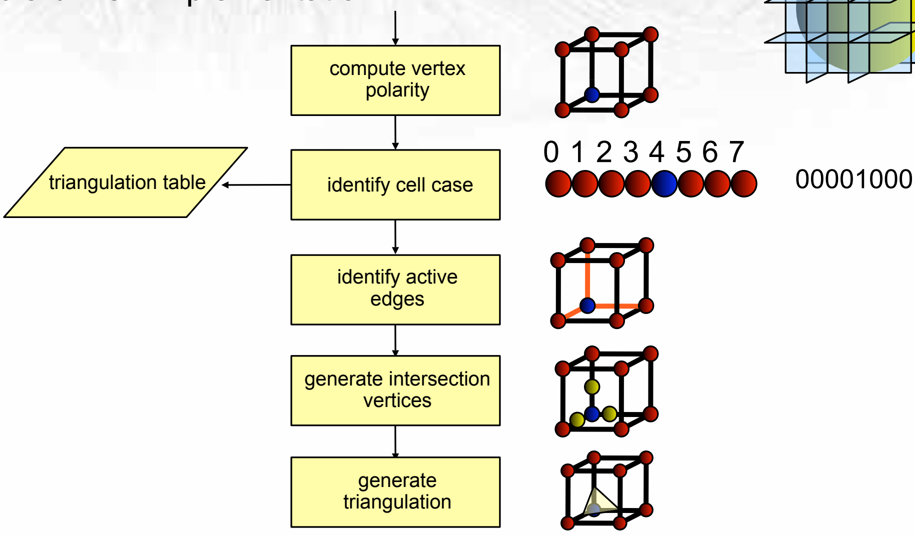




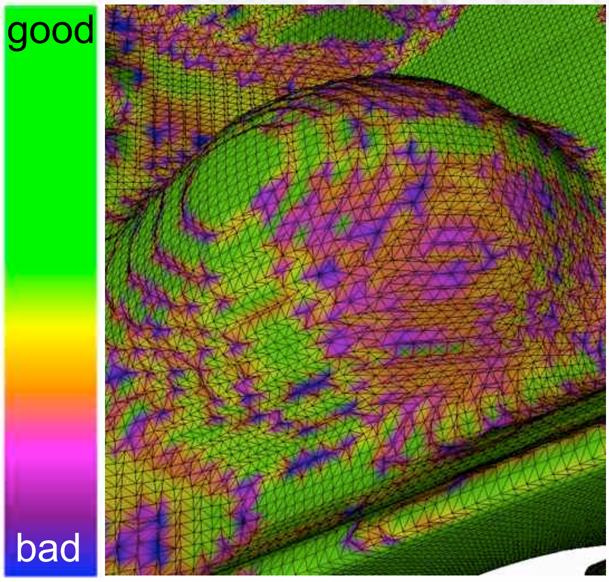
Marching Cubes - (Lorensen/Cline, 1987) 5009 citations 9/10/2008

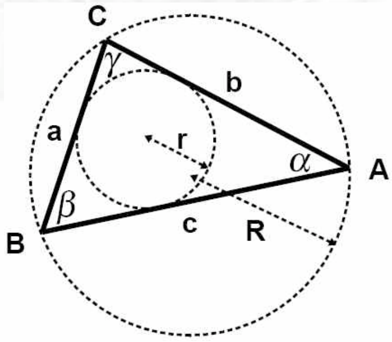
- Code is simple, efficient and produces stable results
- Table-driven implementation





Triangle Quality of Marching Cubes

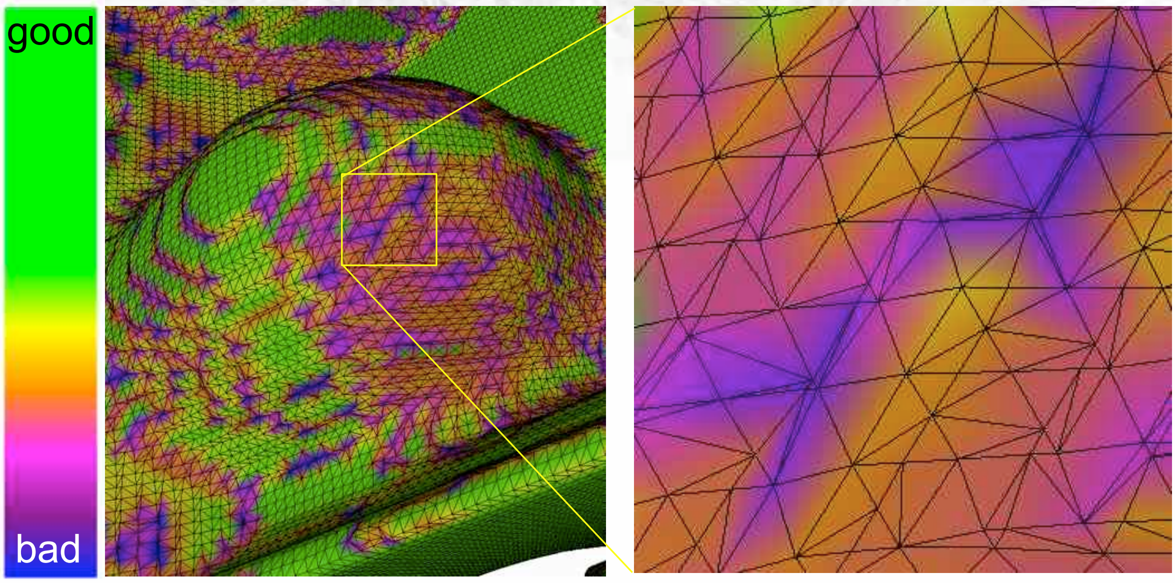




Radii ratio (2r/R)

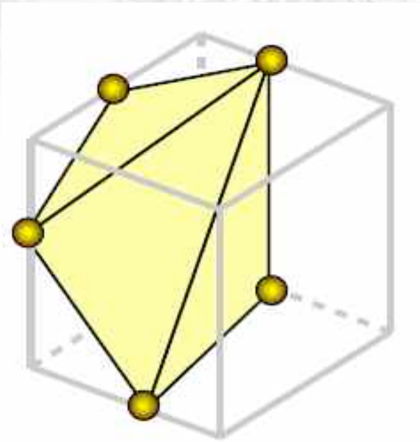
Triangle Quality of Marching Cubes

- Mostly good quality triangles. But some very poor quality triangles



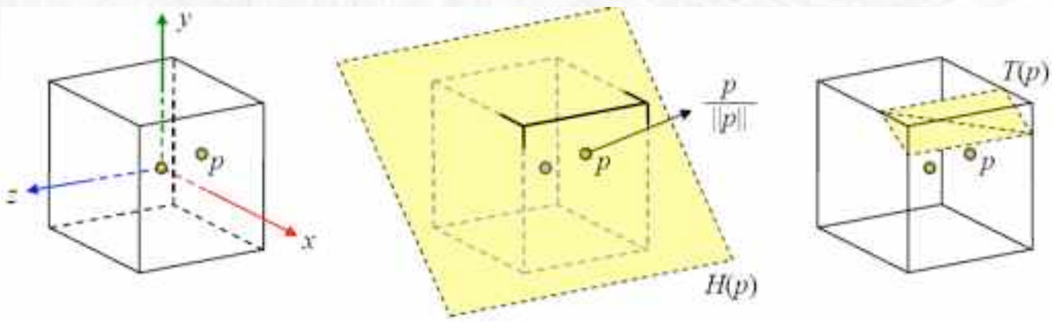
Triangle Quality of Marching Cubes

- In this work we propose a way to understand why this happens.

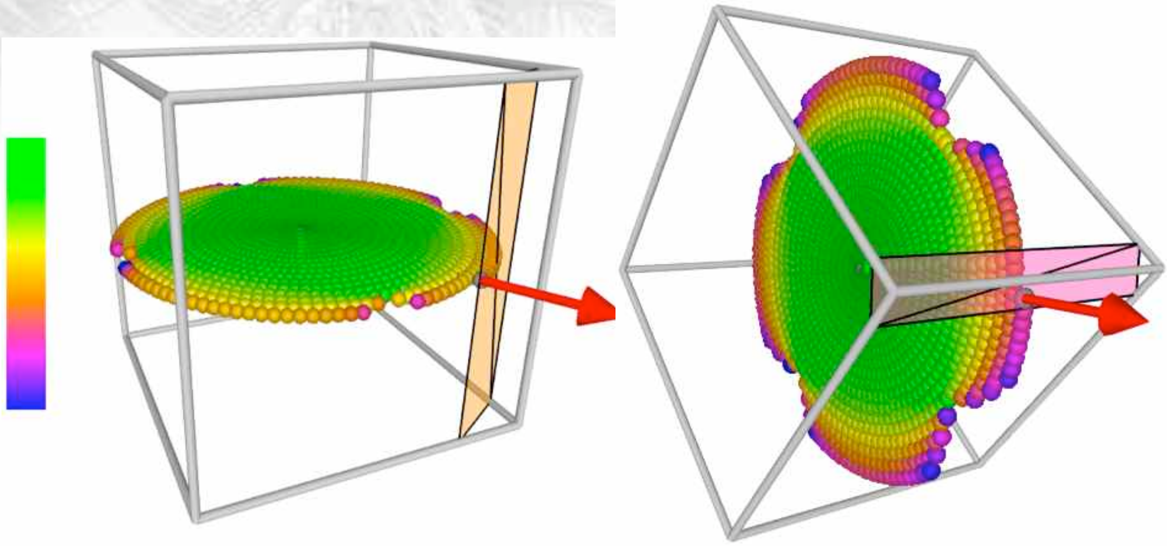


Mesh quality generated by MC

- Evaluate all possible planar intersections between the isosurface and the cell

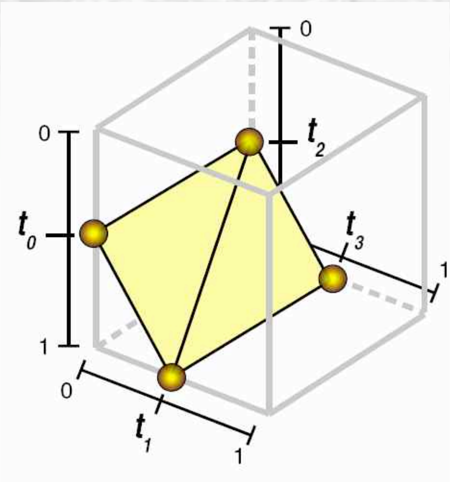


Mesh quality generated by MC

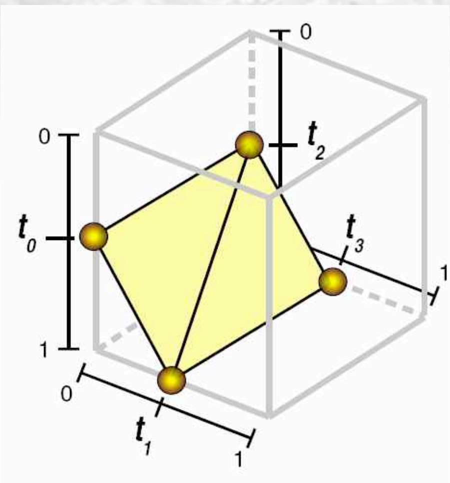


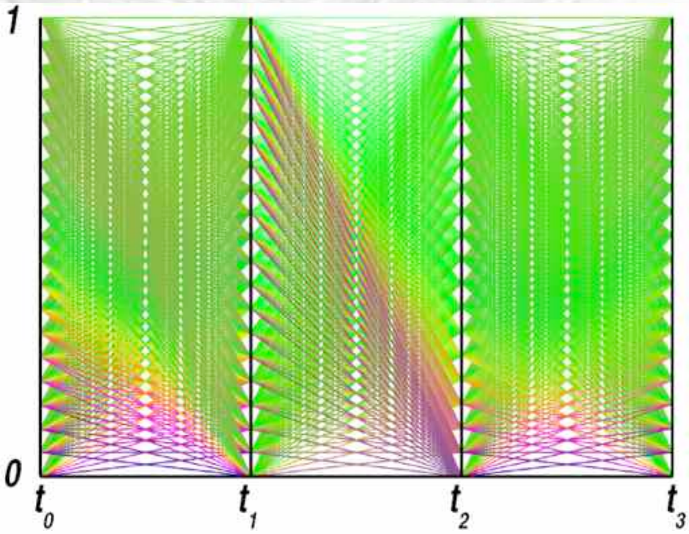
Edge Transformations for Improving Mesh Quality of Marching Cubes

Mesh quality generated by MC



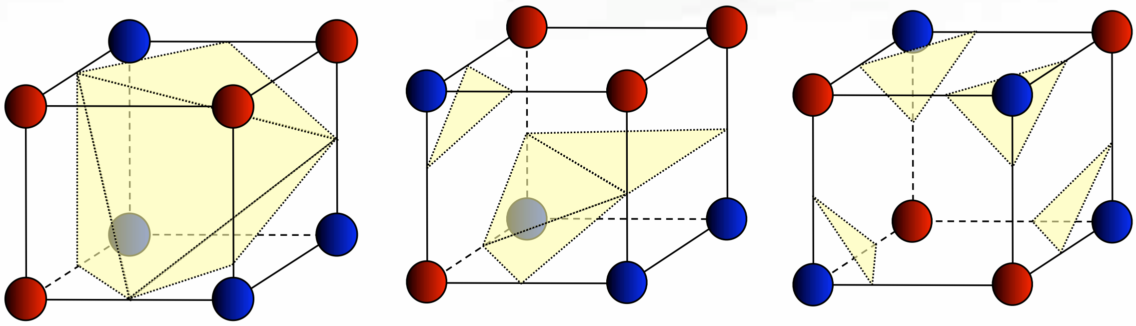
Mesh quality generated by MC





Mesh quality generated by MC

- Difficult analysis of parameters



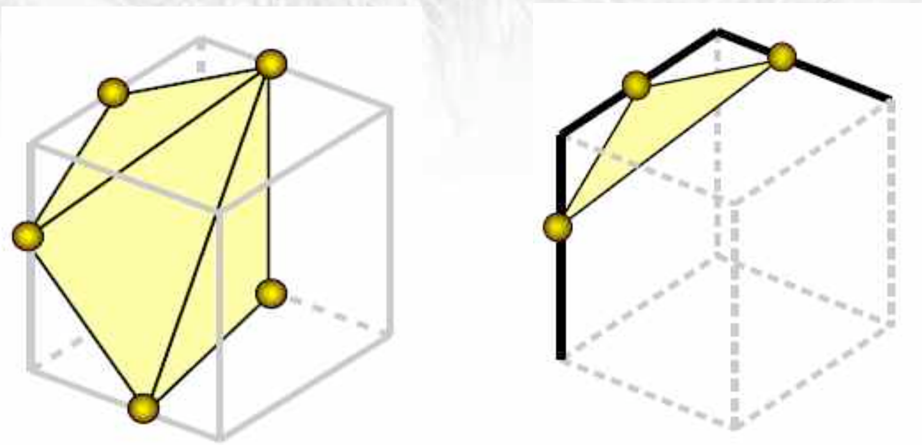
6 parameters

8 parameters

12 parameters

Triangle Quality of Marching Cubes

- Look at each triangle individually

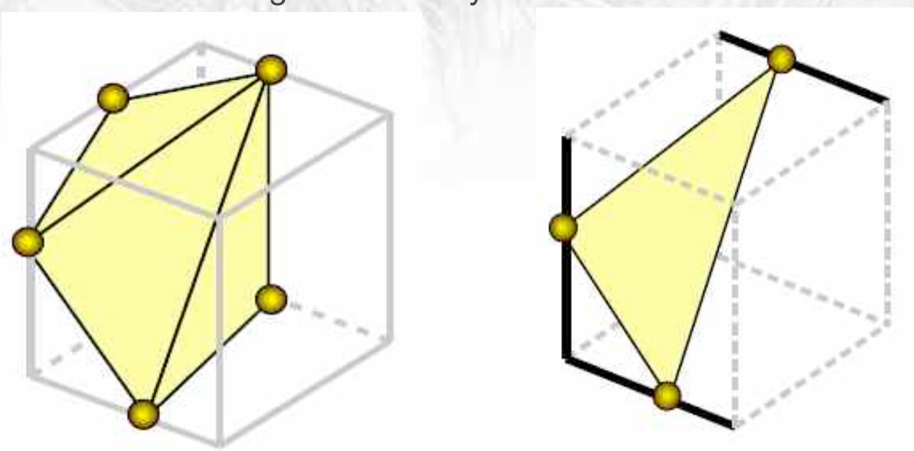


Important Observation:

Each triangle generated by MC has vertices only over active edges. Therefore, it does not generate arbitrary triangles

Triangle Quality of Marching Cubes

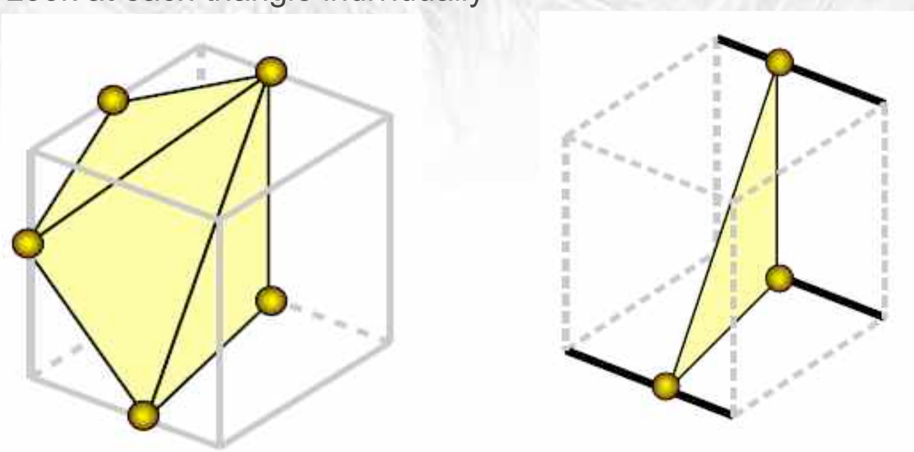
- Look at each triangle individually



Important Observation:
Each triangle generated by MC has vertices only over active edges. Therefore, it does not generate arbitrary triangles

Triangle Quality of Marching Cubes

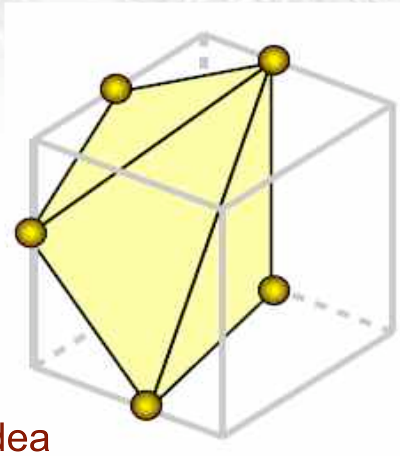
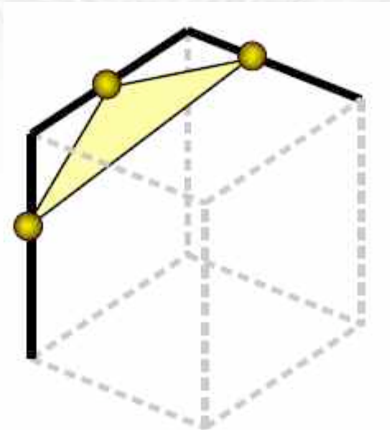
- Look at each triangle individually



Important Observation:
Each triangle generated by MC has vertices only over active edges. Therefore, it does not generate arbitrary triangles

Triangle Quality of Marching Cubes

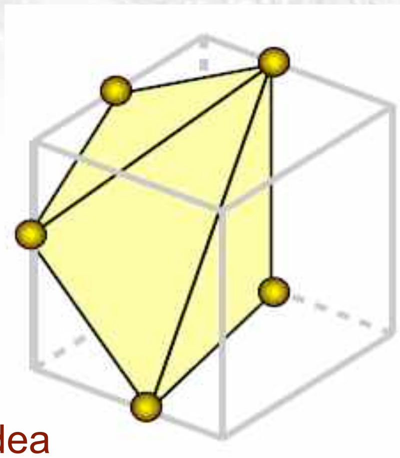
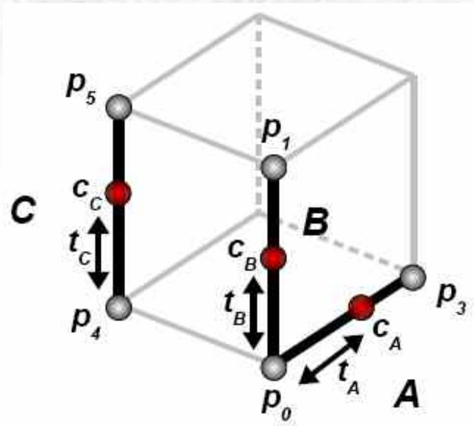
- Look at each triangle individually

Idea
Since three active edges define a triangle, we could evaluate all possible triangles generated by such configuration

Triangle Quality of Marching Cubes

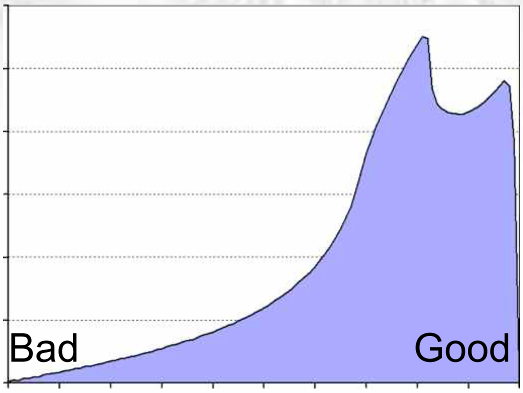
- Look at each triangle individually

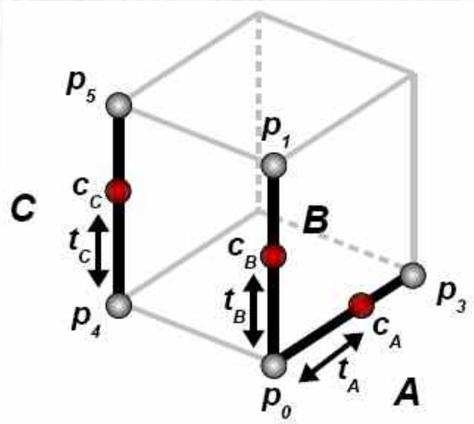
Idea
Since three active edges define a triangle, we could evaluate all possible triangles generated by such configuration

Triangle Quality of Marching Cubes

- Look at each triangle individually



histogram of triangle quality

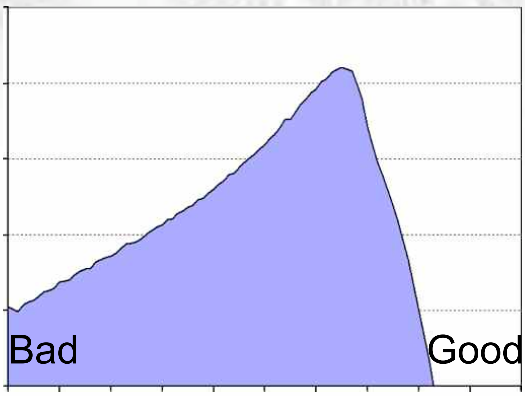


Idea

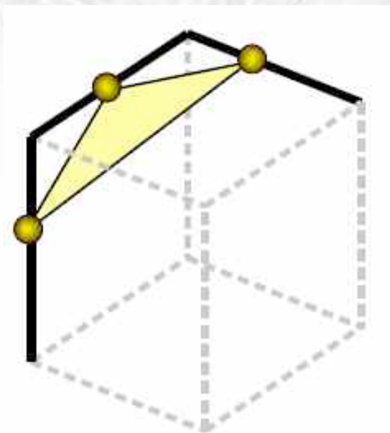
Since three active edges define a triangle, we could evaluate all possible triangles generated by such configuration

Triangle Quality of Marching Cubes

- Several conclusions can be derived



histogram of triangle quality



Interesting Observation

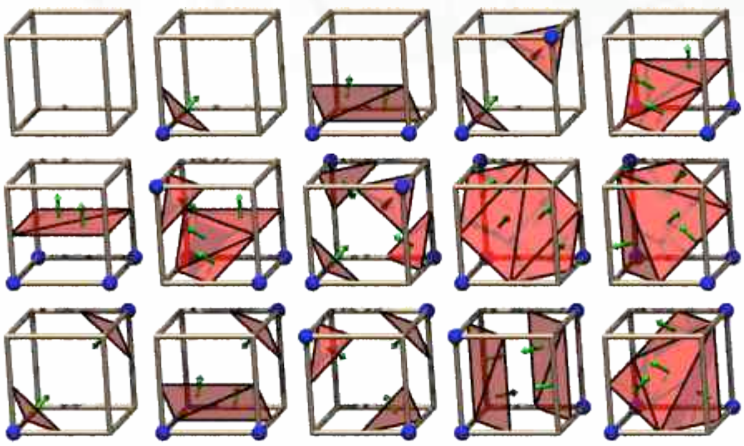
This configuration is responsible for the worst triangles of MC !!!

Triangle Quality of Marching Cubes

- Other Questions to answer:
  - How many cases of edge configurations ?

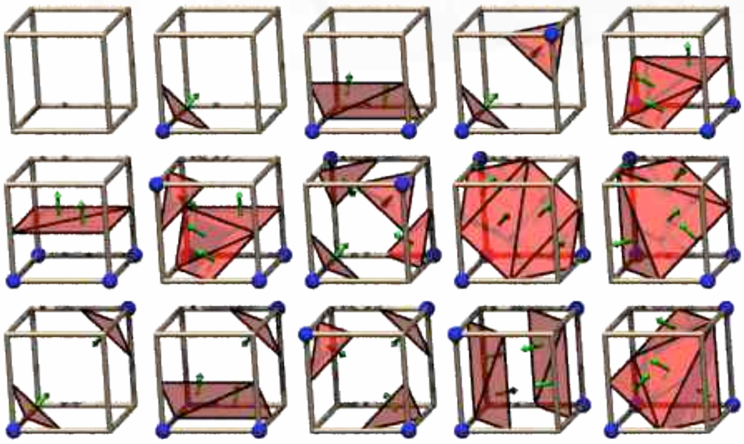
Triangle Quality of Marching Cubes

- Other Questions to answer:
  - How many cases of edge configurations ?
  - Is it similar to Marching Cubes Cases Enumeration ?



Triangle Quality of Marching Cubes

- Other Questions to answer:
  - How many cases of edge configurations ?
  - Is it similar to Marching Cubes Cases Enumeration ? YES, but at edge level



Outline

- Motivation
- Edge Groups in Cubic Cells
- Improving Triangle Quality Based on Edge Groups
- Results
- Conclusions

Edge Groups in Cubic Cells

- Counting Cases in MC
  - [Lorensen and Cline 87] 15 cases based on inspection
  - [Nielson et al 2002] 23 cases by rotation only
  - [Banks et al 2004] formal proof using group theory

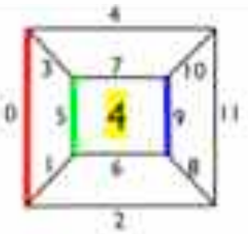
Edge Groups in Cubic Cells

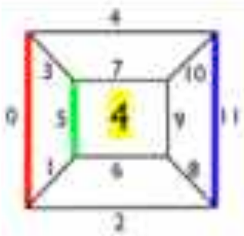
- Counting Cases in MC
  - [Lorensen and Cline 87] 15 cases based on inspection
  - [Nielson et al 2002] 23 cases by rotation only
  - [Banks et al 2004] formal proof using group theory

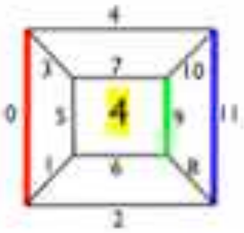
- Counting Cases in Edge Configurations:
  - Edge configuration (e_0, e_1, e_2): 3 different edges of the cube
  - Edge (Symmetry) Group: collection of edge configurations that can be transformed into another by symmetry transformations (e.g. rotations and improper reflections)

Edge Groups in Cubic Cells

- Counting Cases in MC
  - [Lorensen and Cline 87] 15 cases based on inspection
  - [Nielson et al 2002] 23 cases by rotation only
  - [Banks et al 2004] formal proof using group theory
- Counting Cases in Edge Configurations:
  - Edge configuration (e_0, e_1, e_2): 3 different edges of the cube
  - Edge (Symmetry) Group: collection of edge configurations that can be transformed into another by symmetry transformations (e.g. rotations and improper reflections)

(0,5,9)


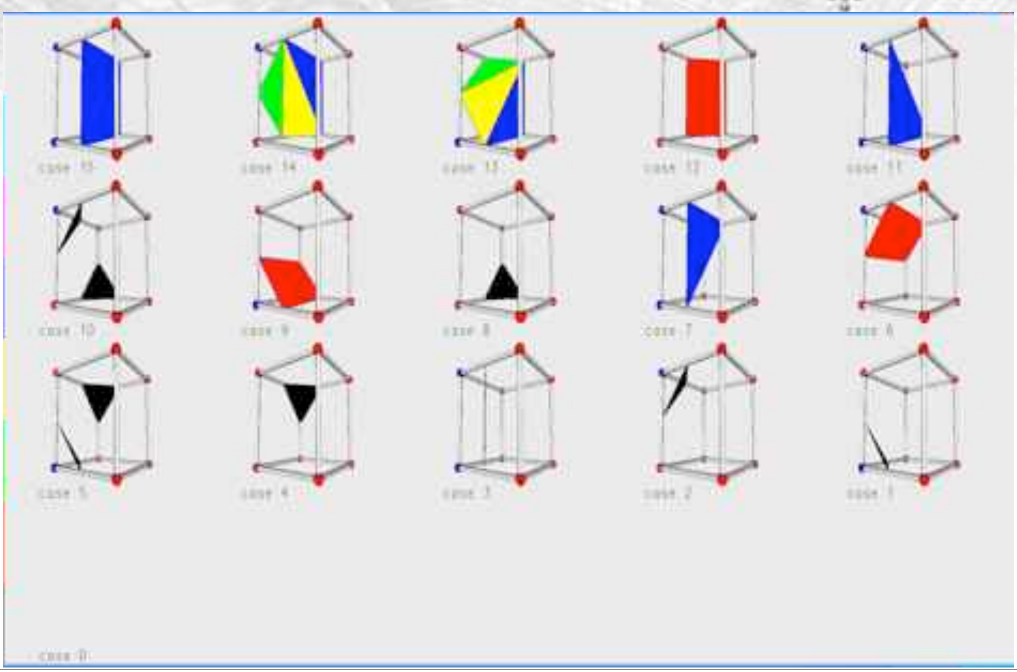
(0,5,11)


(0,9,11)


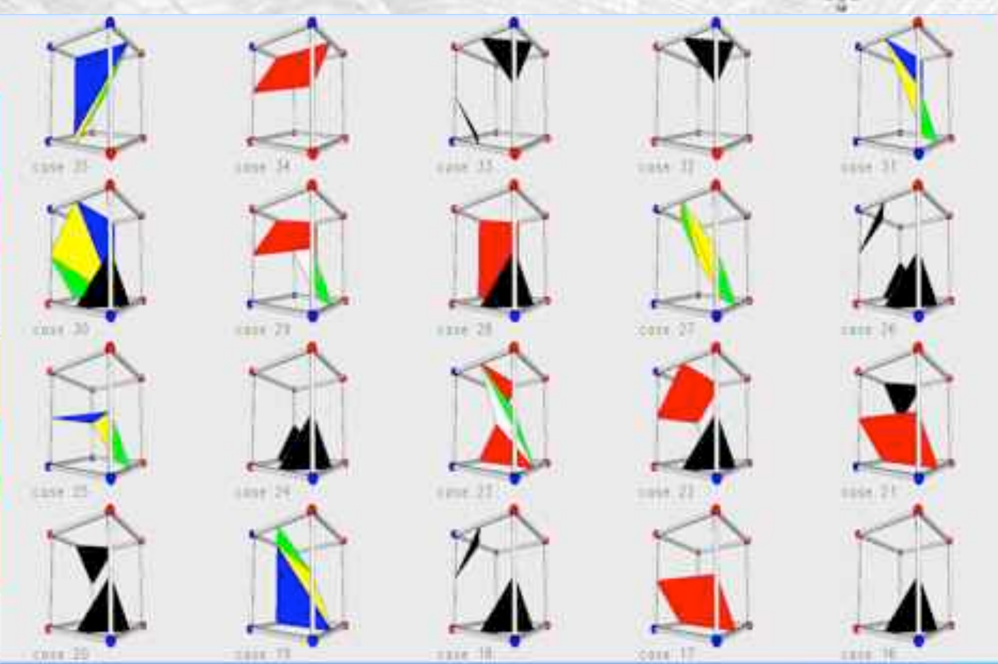
Counting Cases in Edge Groups

- True to the original Marching Cubes paper, we first did it by inspection too
- How ?

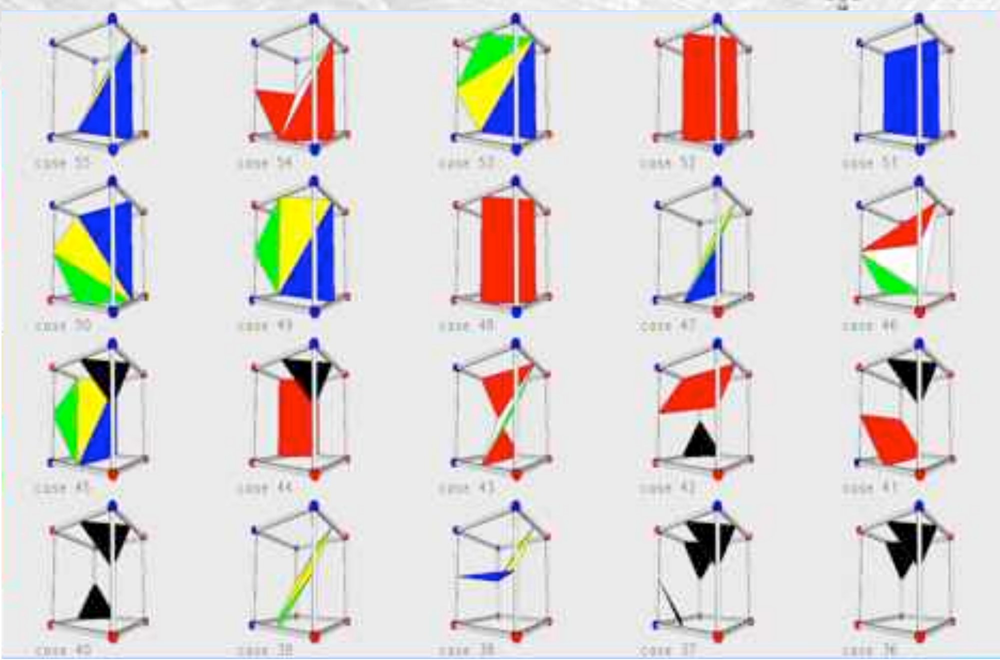
Edge Groups in the MC table



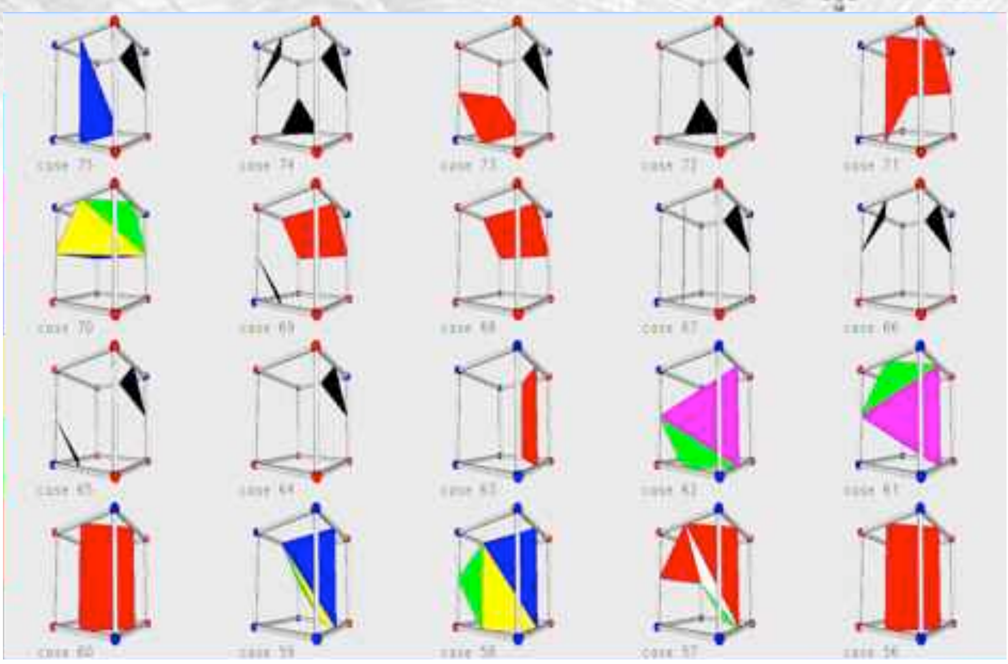
Edge Groups in the MC table



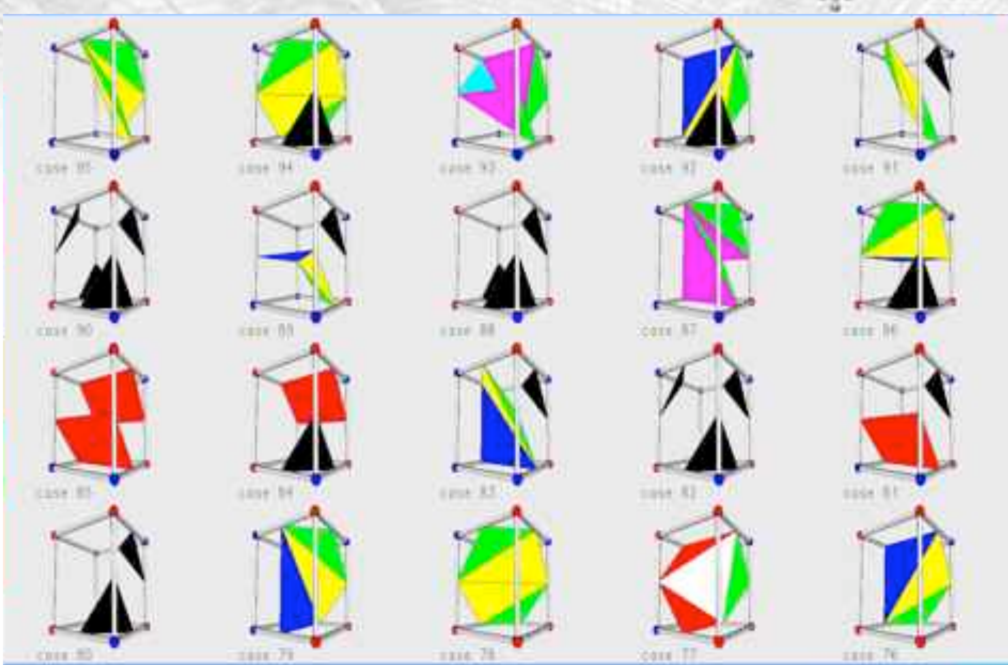
Edge Groups in the MC table



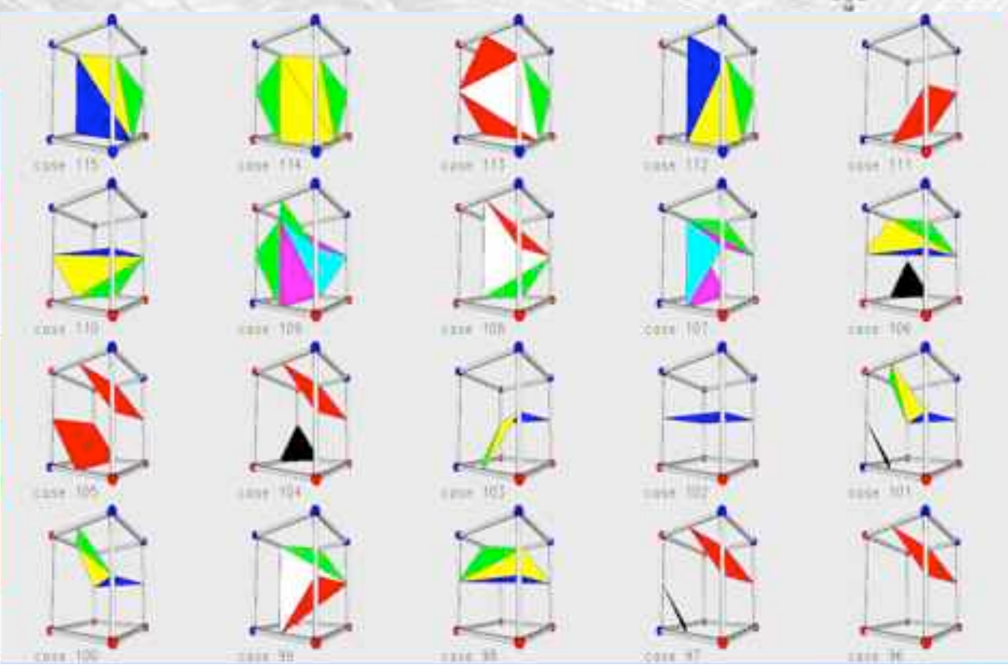
Edge Groups in the MC table



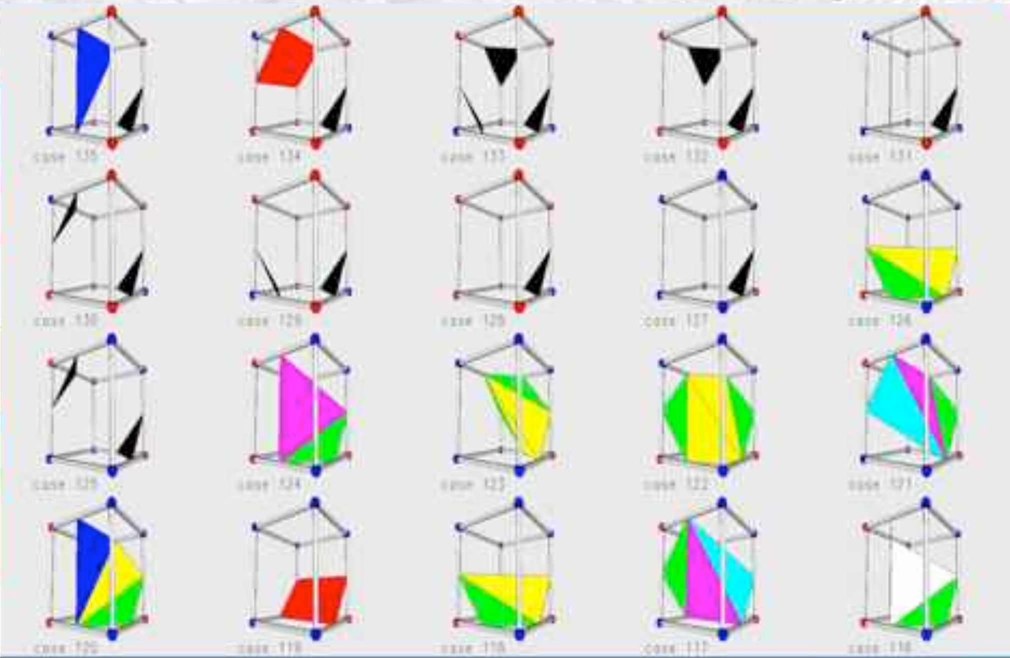
Edge Groups in the MC table



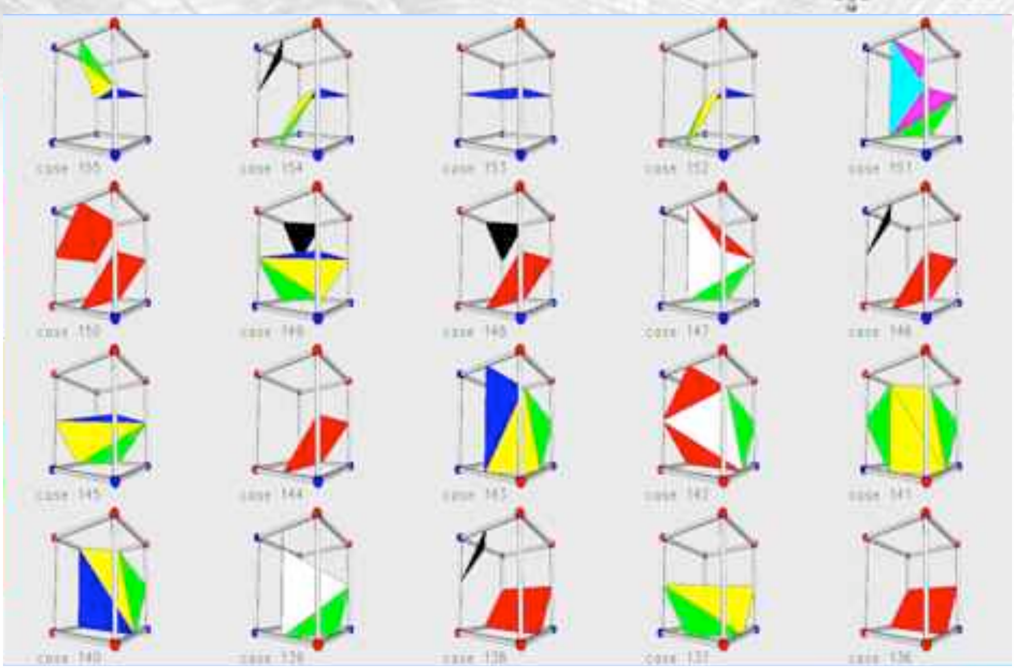
Edge Groups in the MC table



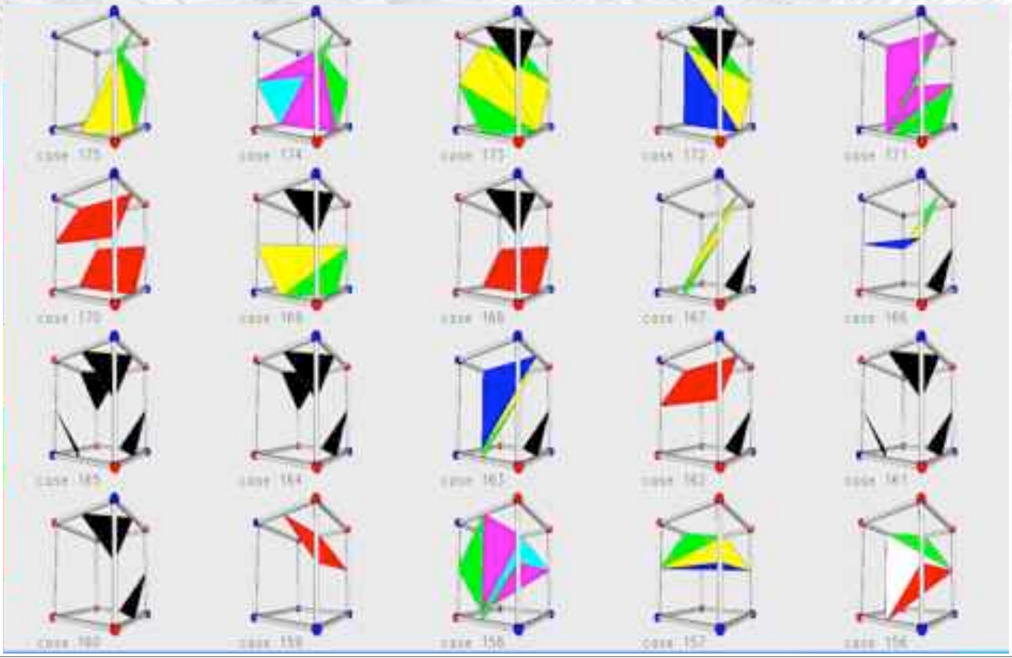
Edge Groups in the MC table



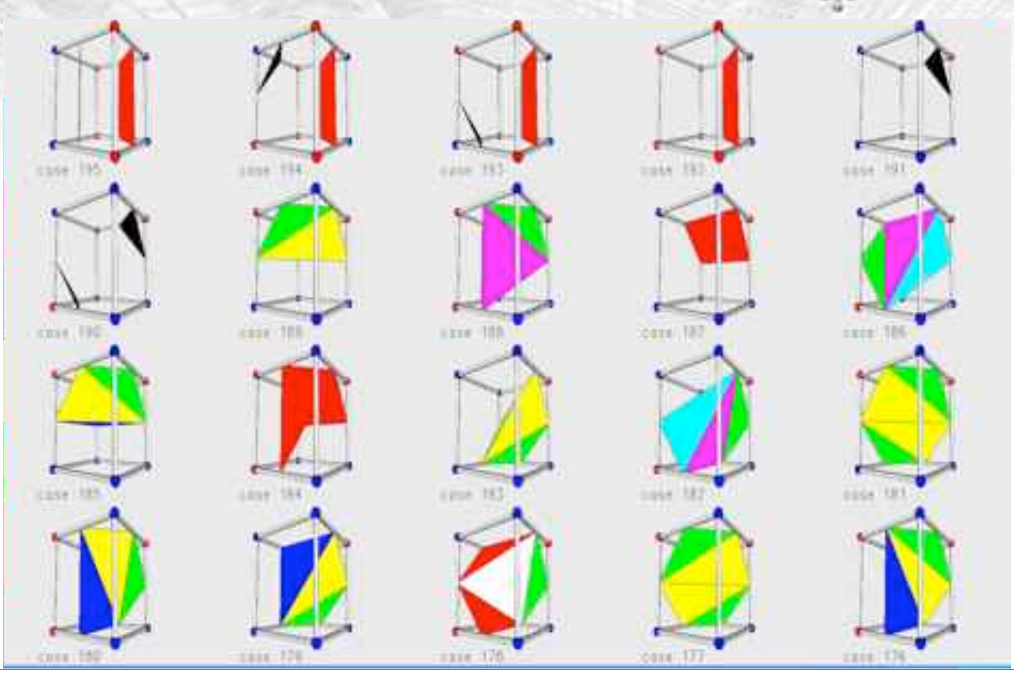
Edge Groups in the MC table



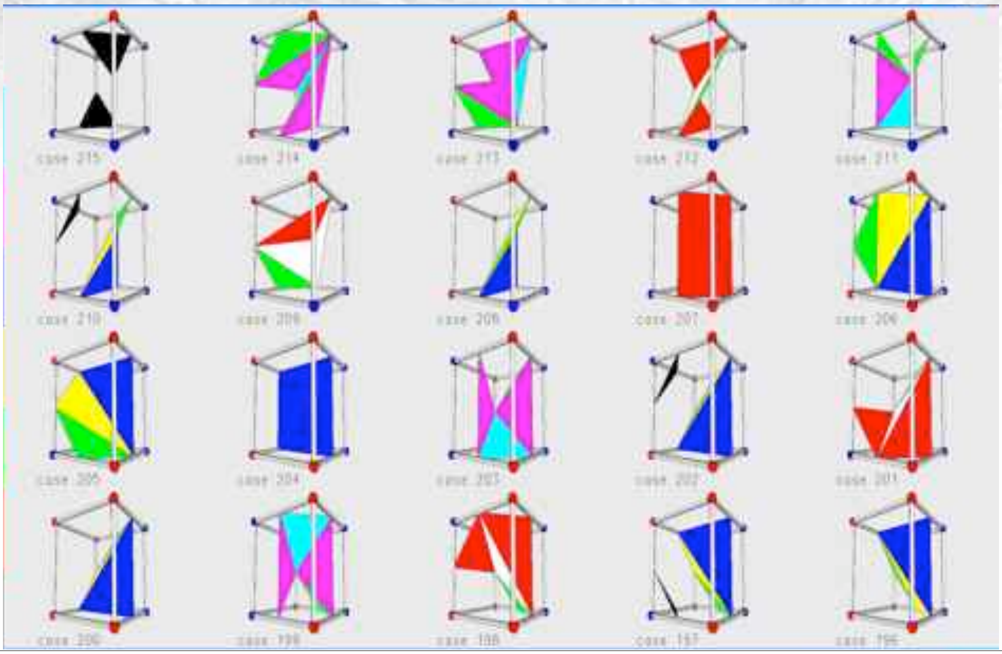
Edge Groups in the MC table



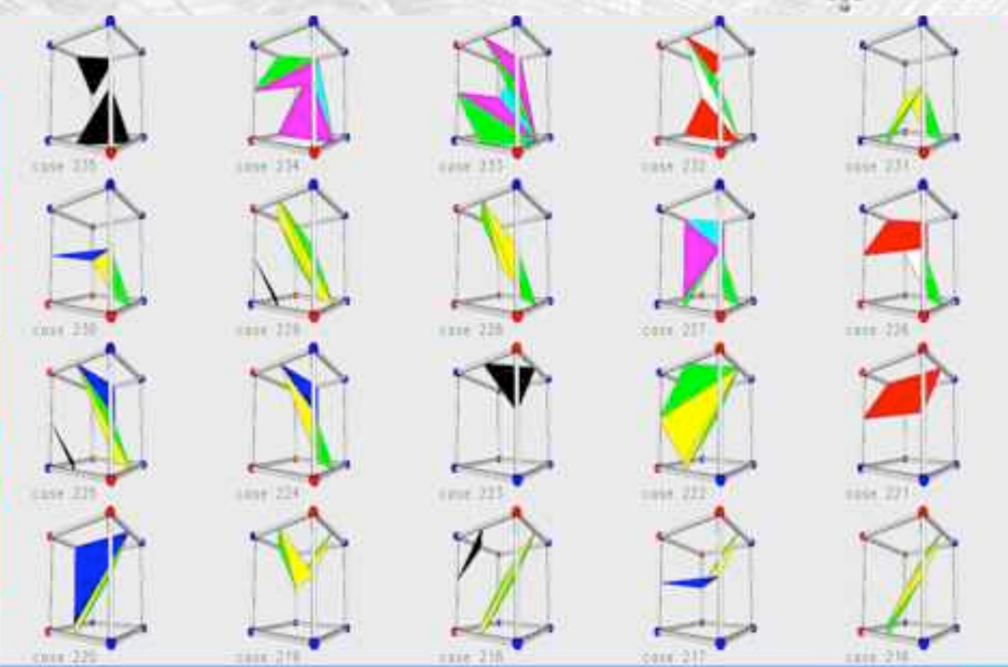
Edge Groups in the MC table



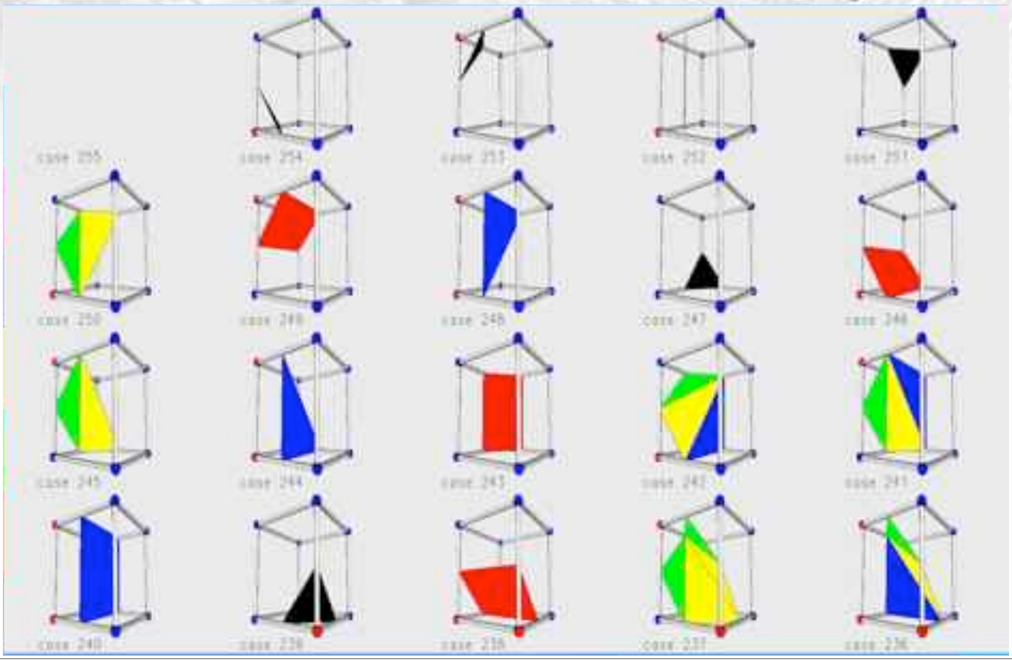
Edge Groups in the MC table



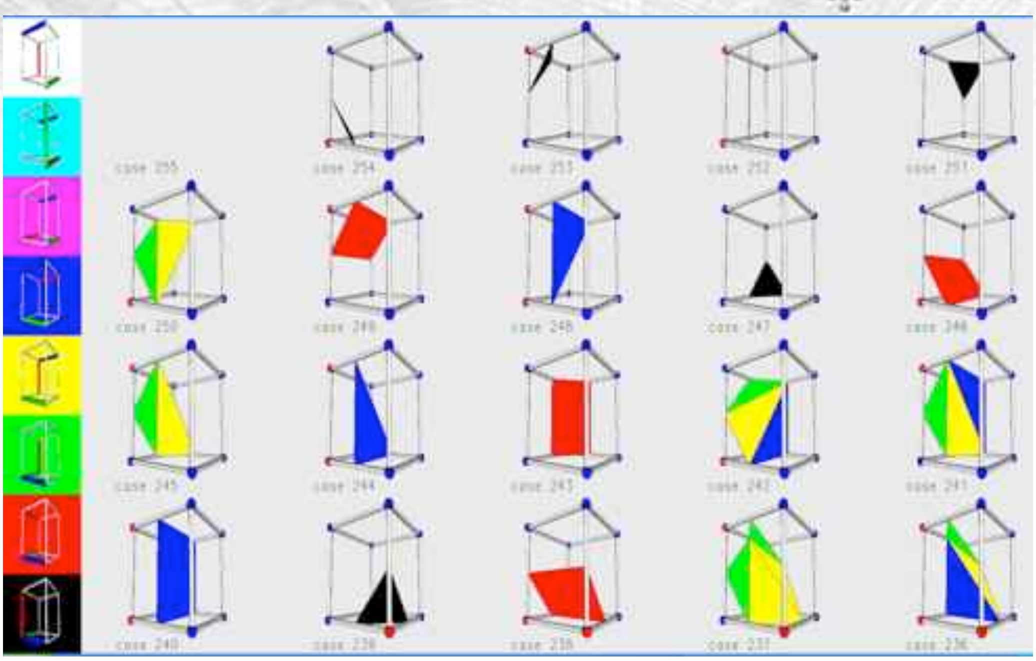
Edge Groups in the MC table



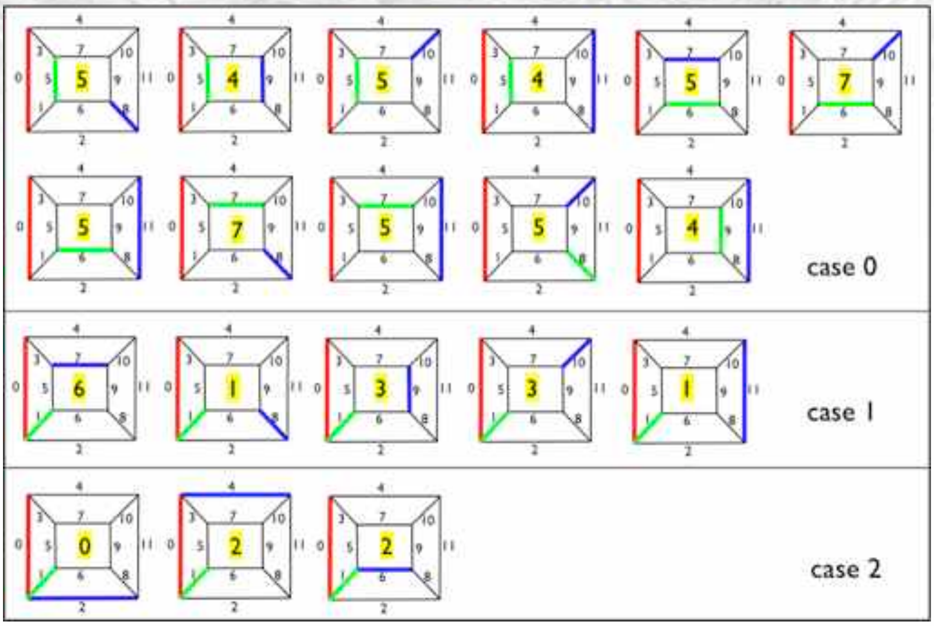
Edge Groups in the MC table



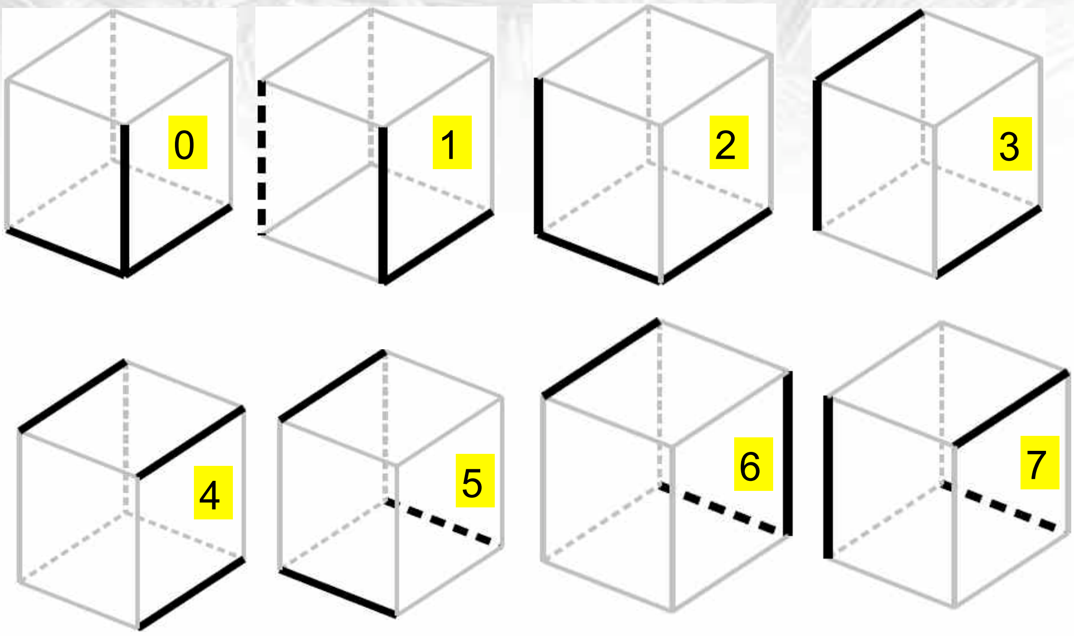
Edge Groups in the MC table



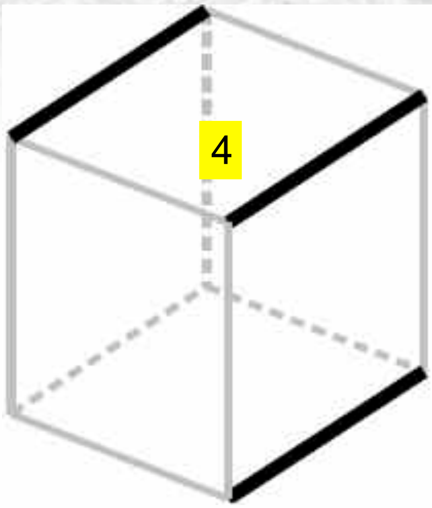
Edge Groups proof (paper)

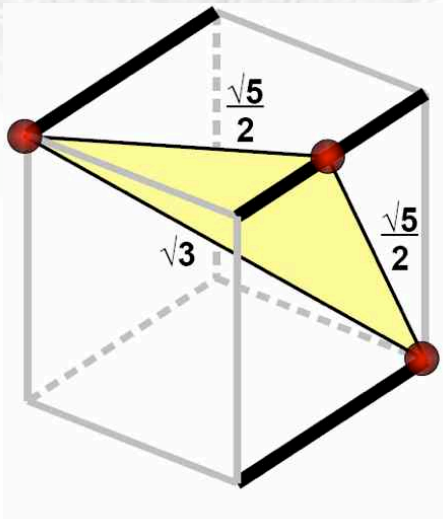


8 Edge Groups in Cubic Cells

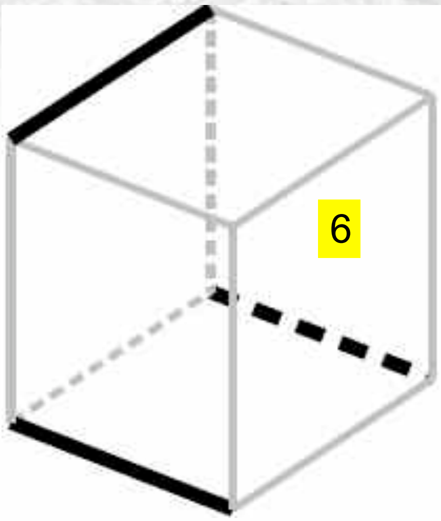


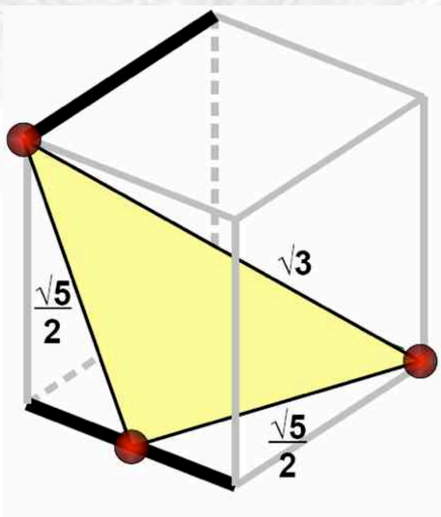
Triangle quality analysis: Worst triangle of case 4



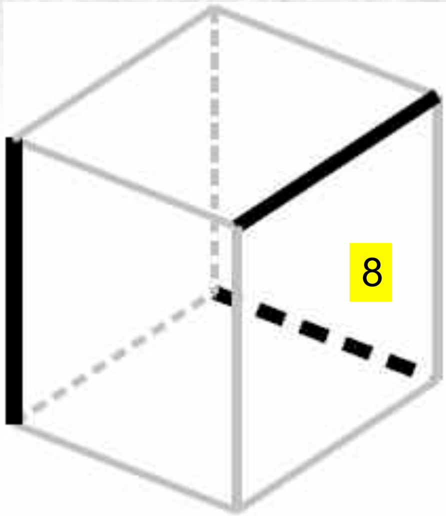


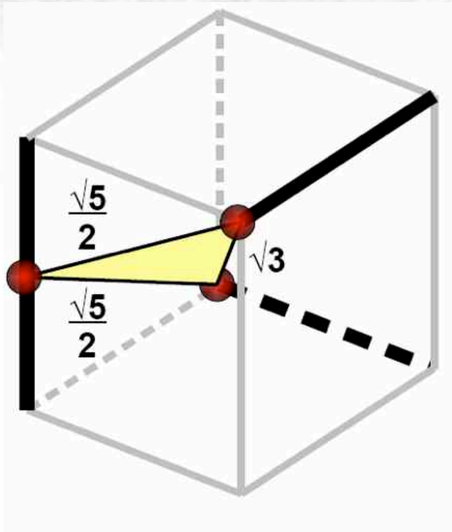
Triangle quality analysis: Worst triangle of case 6



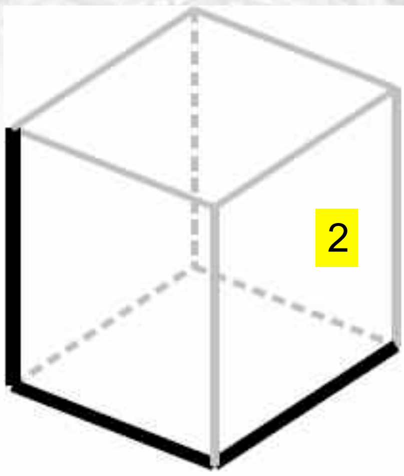


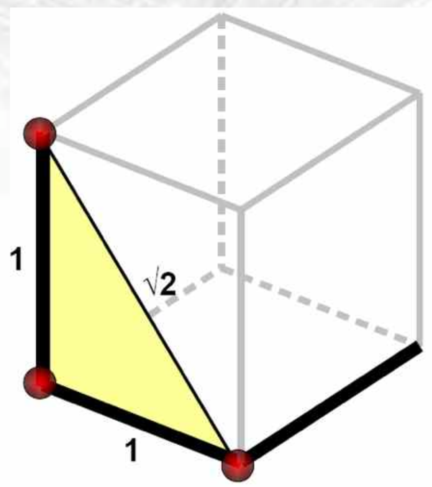
Triangle quality analysis: Worst triangle of case 8



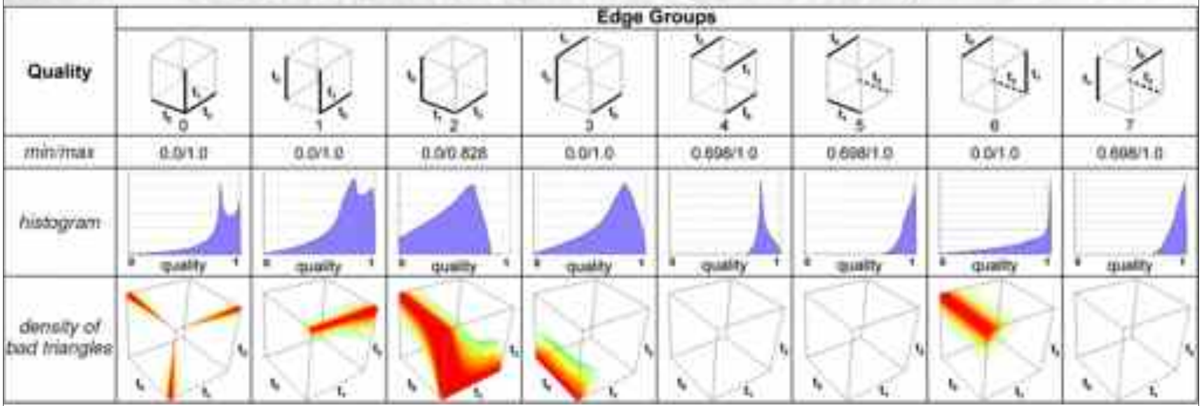


Triangle quality analysis: Best triangle of case 2

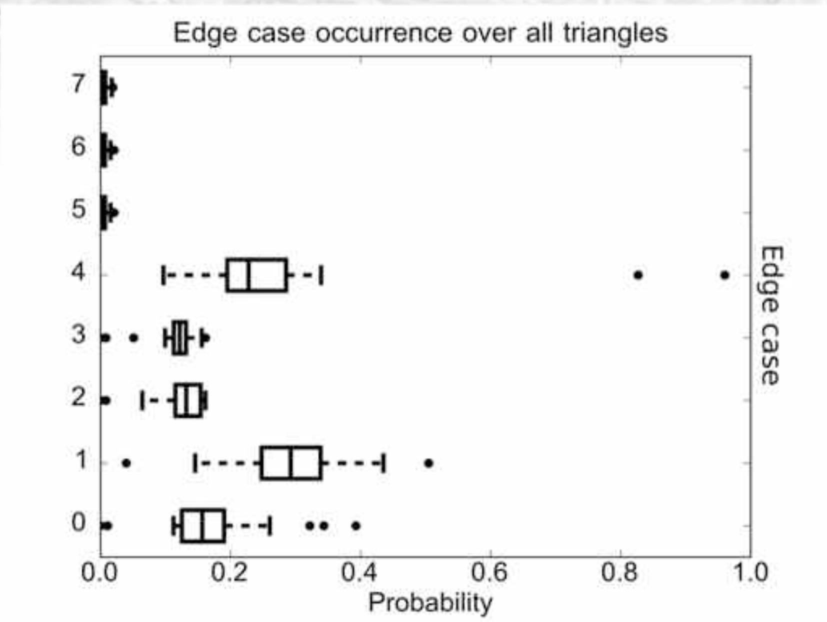




Triangle Quality Analysis: Edge Groups

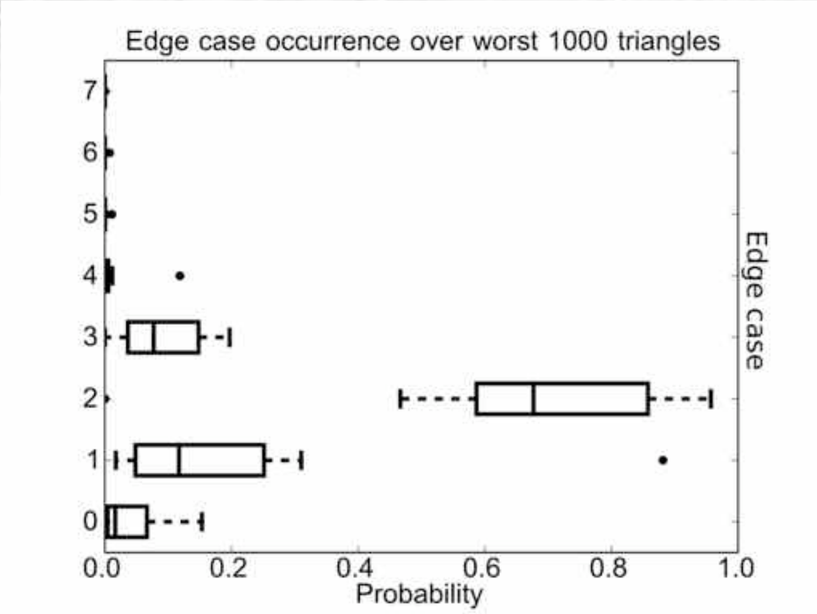


Edge Groups Occurrence



30 datasets

Interesting Observation: Edge Group 2 dominates



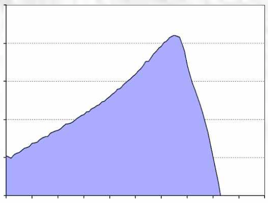
30 datasets

Outline

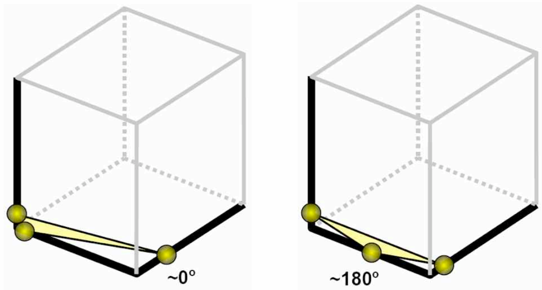
- Motivation
- Edge Groups in Cubic Cells
- Improving Triangle Quality Based on Edge Groups
- Results
- Conclusions

Improving Triangle Quality based on Edge Groups

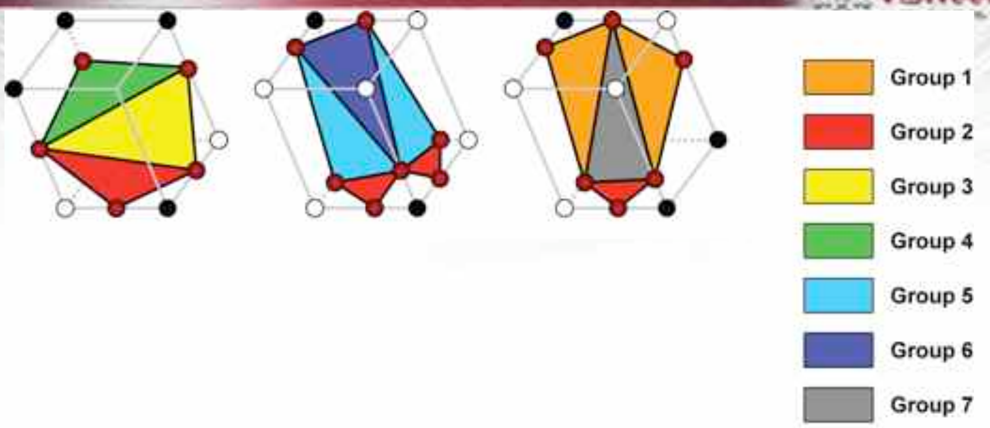
- Observation 1 : Edge Group 2 generates the worst triangles
  - Changing the MC table

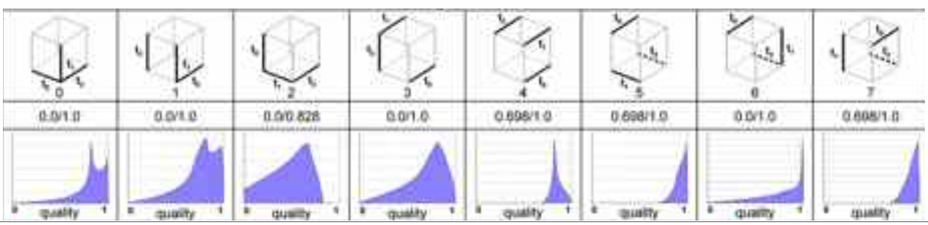


- Observation 2: Bad triangles when isosurface is parallel to the edge
  - Edge transformations using Macet [Dietrich et al 2009]

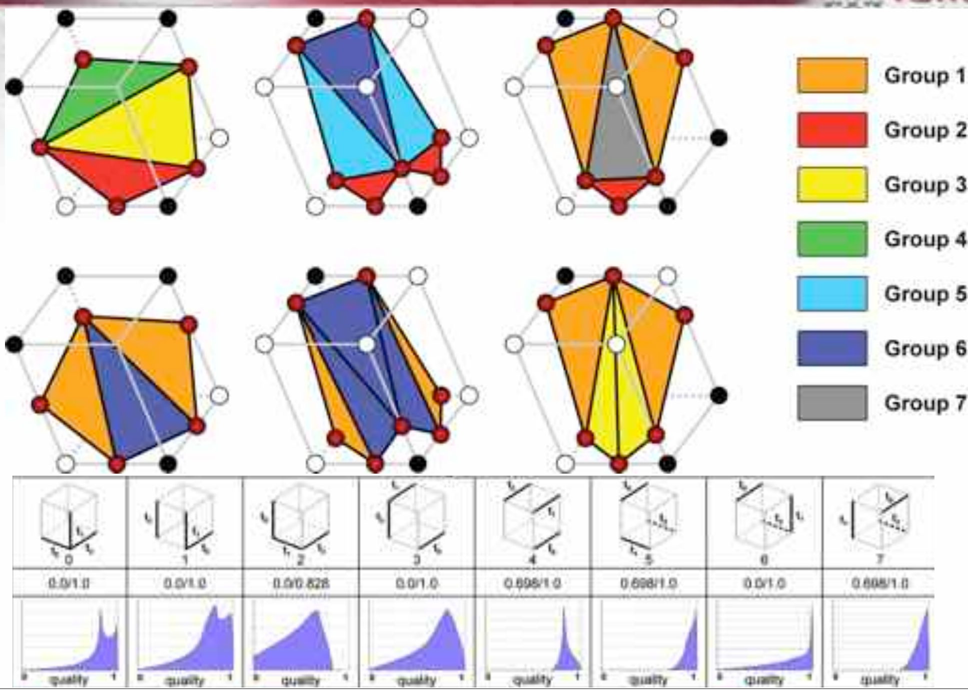


Improvement 1: Changing MC Table



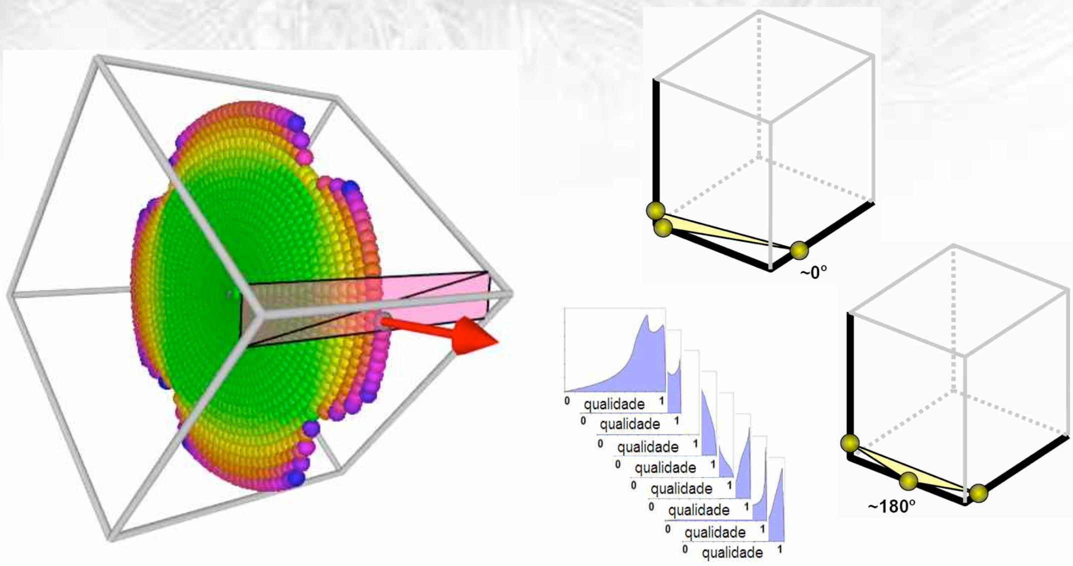


Improvement 1: Changing MC Table



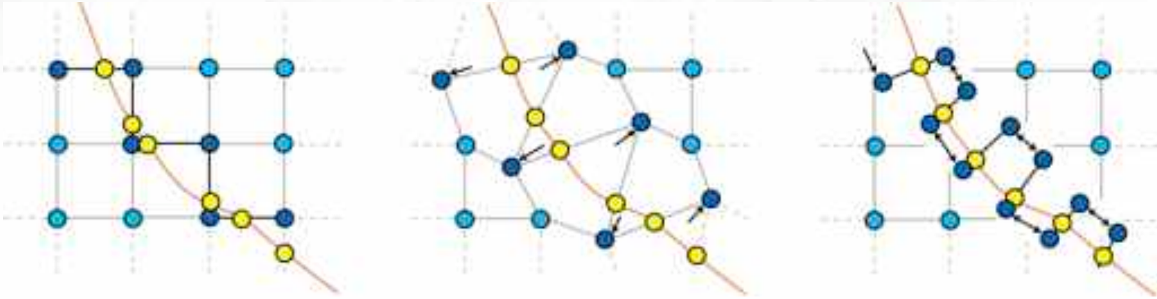
Evaluating Situations where bad triangles arise

- Bad cases happen when the isosurface is parallel to the edge



Improvement 2: Edge Transformations using Macet

- Macet [Dietrich et al 2009]
  - Edge transformations alter positions of grid points in MC

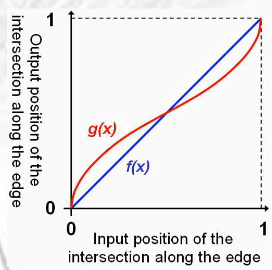


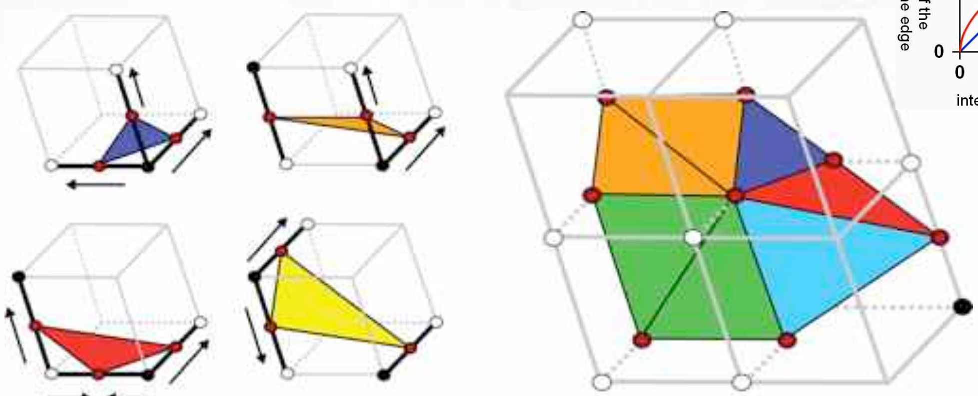
Gradient Transformation

Tangent Transformation

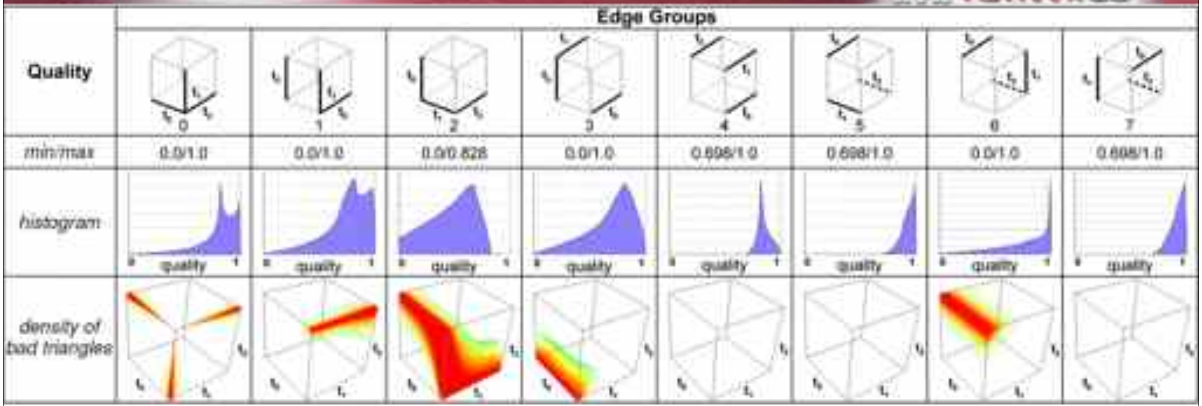
Improvement 2: Edge Transformations using Macet

- Displacement of Intersection Vertices

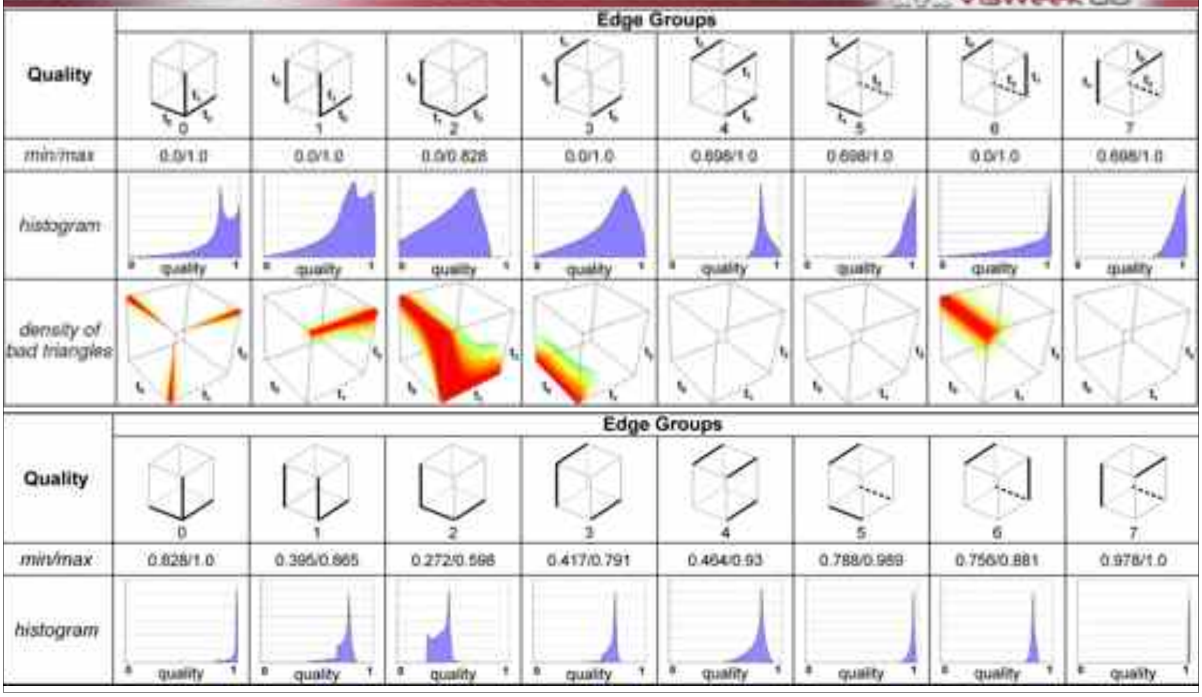




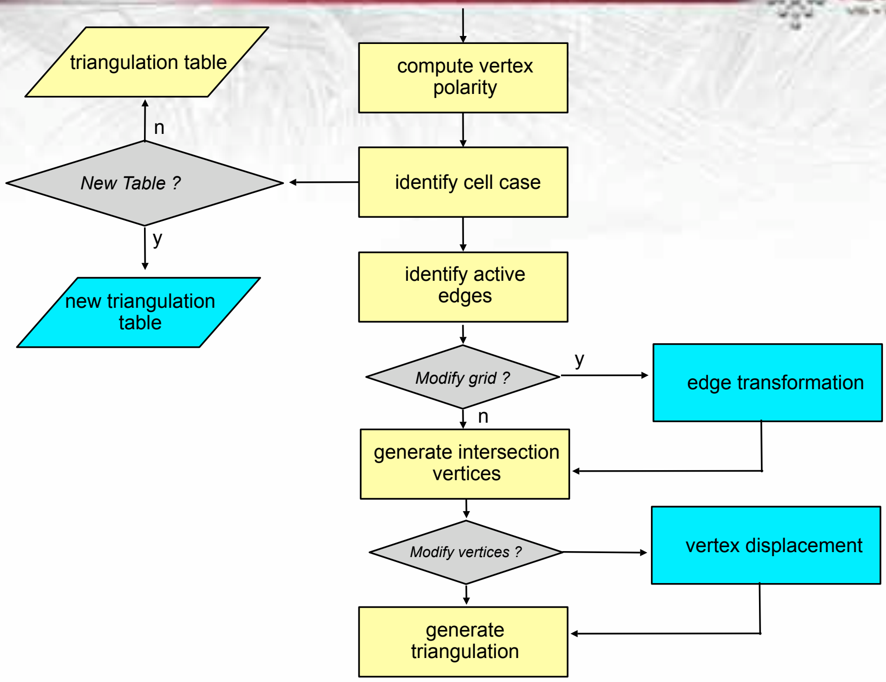
Improvement 2: Edge Transformations using Macet



Improvement 2: Edge Transformations using Macet



Summary of Changes to Marching Cubes Code



Outline

- Motivation
- Edge Groups in Cubic Cells
- Improving Triangle Quality Based on Edge Groups
- Results
- Conclusions

Results

	Original MC Old Table	Macet Old Table	Original MC New Table	Macet New Table Displacements
Silicium @ 140.5	0.00157			
Engine @ 49.5	0.00042			
Bonsai @ 49.5	0.00013			
Lobster @ 49.5	0.00088			

Results

	Original MC Old Table	Macet Old Table	Original MC New Table	Macet New Table Displacements
Silicium @ 140.5	0.00157		0.01792	
Engine @ 49.5	0.00042		0.00080	
Bonsai @ 49.5	0.00013		0.00152	
Lobster @ 49.5	0.00088		0.00153	

Results

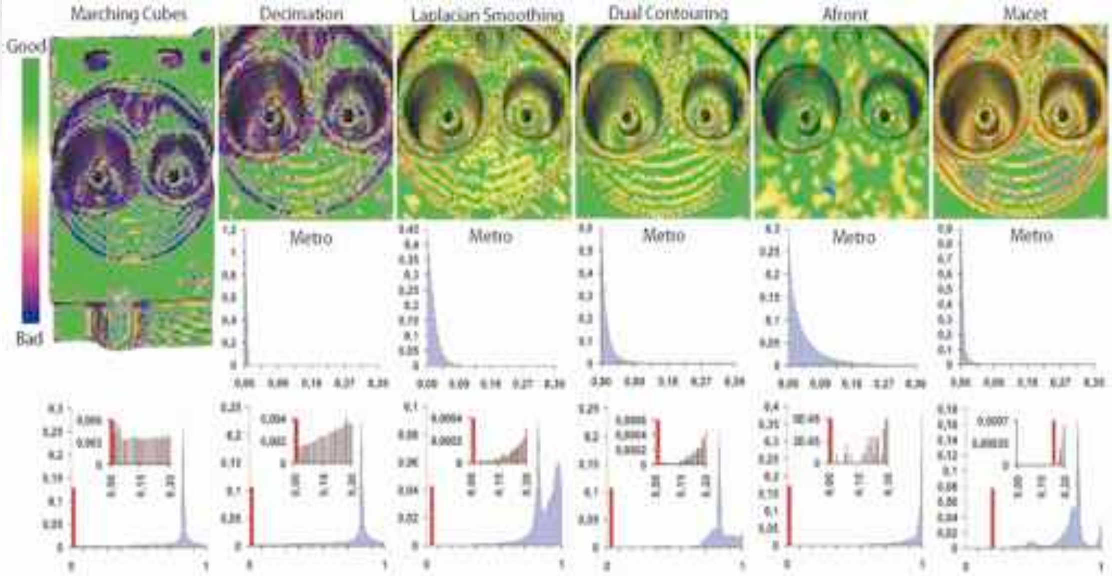
	Original MC Old Table	Macet Old Table	Original MC New Table	Macet New Table Displacements
Silicium @ 140.5	0.00157	0.199	0.01792	
Engine @ 49.5	0.00042	0.235	0.00080	
Bonsai @ 49.5	0.00013	0.102	0.00152	
Lobster @ 49.5	0.00088	0.116	0.00153	

Results

	Original MC Old Table	Macet Old Table	Original MC New Table	Macet New Table Displacements
Silicium @ 140.5	0.00157	0.199	0.01792	0.344
Engine @ 49.5	0.00042	0.235	0.00080	0.293
Bonsai @ 49.5	0.00013	0.102	0.00152	0.195
Lobster @ 49.5	0.00088	0.116	0.00153	0.247

Results (Previous Version of Macet)

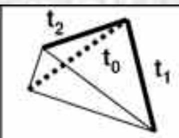
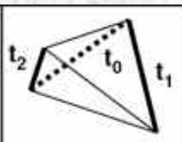
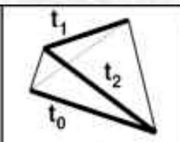
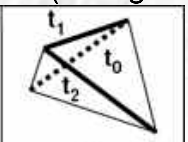
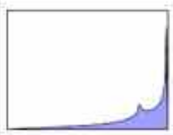
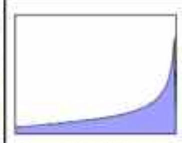
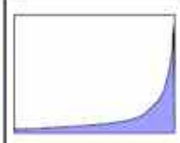
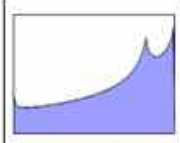
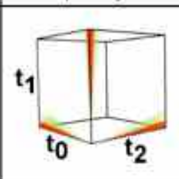
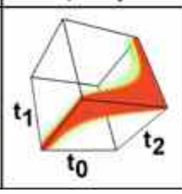
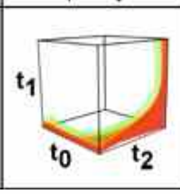
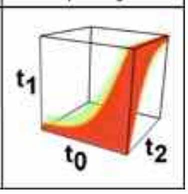
- Comparison Against Other Methods



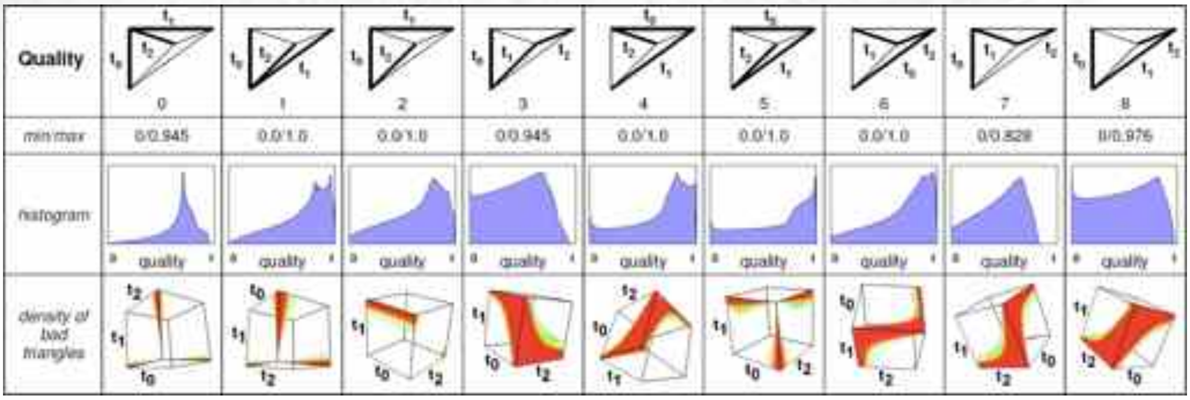
Edge Groups in Tetrahedra Cells

- Extensions to tetrahedra cells

BCC Tetrahedra (4 Edge Groups)

Quality	 0	 1	 2	 3
min/max	0.0/1.0	0.0/1.0	0.0/1.0	0.0/1.0
histogram	 0 quality 1	 0 quality 1	 0 quality 1	 0 quality 1
density of bad triangles				

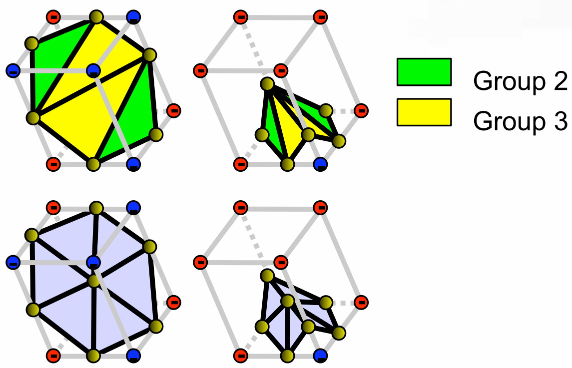
Edge Groups in Tetrahedra Cells

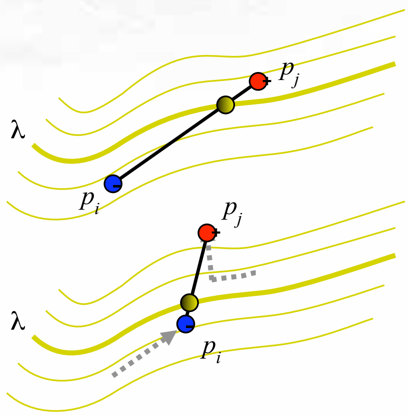


Kuhn Tetrahedra (9 Edge Groups)

Ongoing Work

- Unified Edge Transformations Macet (Technical Report coming soon)
  - Single Edge Transformation
  - Optional Vertex Displacement
  - More efficient, more quality





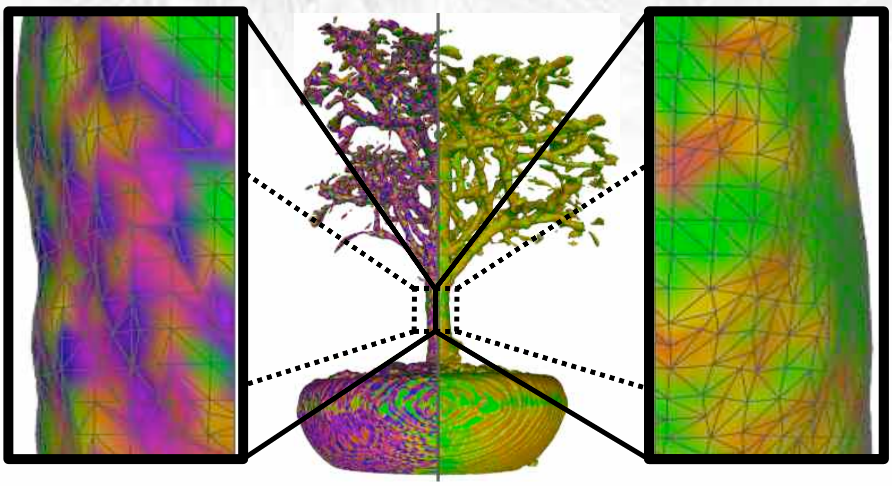
Latest Quality and Performance Results

- Unified Edge Transformations Macet

	MC		Unified Macet	
	time	*quality*	*time*	*quality*
Bonsai	1.4	0.0002	9.9	0.32
Engine	1.1	0.0002	8.6	0.35
Hydrogen	0.15	0.069	0.7	0.52
Laçador	0.7	4.8e-7	2	0.37
Neghip	0.07	0.0031	0.64	0.38
Siliicium	0.03	0.0009	0.38	0.41

Demo

- Side-by-side comparison: MC x Unified Macet



Marching Cubes

Macet

Conclusions

- Main contributions:
  - Edge groups proposal and analysis in MC:
    - Histogram analysis of each individual case
    - Case 2 generates the worst triangles
  - Modifications in MC
    - Change of MC table
    - Improvements and Validation of Macet
  - Reproducibility (open source):
    - http://www.sci.utah.edu/~csheid/vis2008/edge-groups (Vistrails)

Acknowledegments

- CNPq (Brazil) scholarship and grants 485853/2007-0, 453538/2008-8
- IBM Ph.D. Student Fellowship.
- Department of Energy SciDAC (VACET and SDM centers)
- National Science Foundation (grants CNS-0751152, CCF-0528201, OCE-0424602, CNS-0514485, IIS-0513692, CCF-0401498, OISE-0405402, CNS-0551724)
- IBM Faculty Awards (2005, 2006, and 2007)
- Reviewers for feedback

Questions ?