Scientific Visualization

Spring 2018
Center for Data Science
New York University

Claudio Silva Gustavo Nonato

Lecture 6: Isosurfacing

Isosurfacing

Volume Data: $f(x,y,z)$ known in each point (x,y,z) of a domain.

(in practice f is know in a set of sample points of the domains)

f: R^3 --> R (assigns a scalar to each point of the domain)

Isosurfacing

Volume Data: $f(x,y,z)$ known in each point (x,y,z) of a domain.

(in practice f is know in a set of sample points of the domains)

f: R^3 --> R (assigns a scalar to each point of the domain)

Problem: Find the set $Sc = \{(x,y,z) \mid f(x,y,z)=c\}$

Isosurfacing

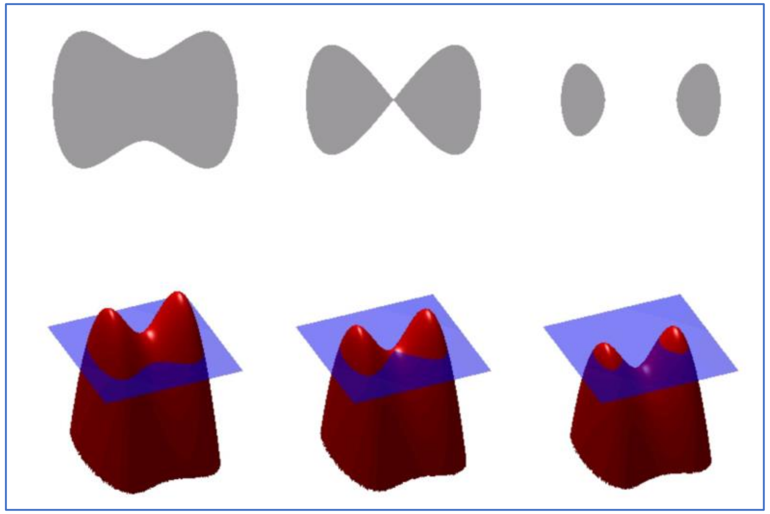
Volume Data: $f(x,y,z)$ known in each point (x,y,z) of a domain.

(in practice f is know in a set of sample points of the domains)

$f: R^3 \dashrightarrow R$ (assigns a scalar to each point of the domain)

Problem: Find the set $Sc = \{(x,y,z) \mid f(x,y,z)=c\}$

In 2D



Isosurfacing

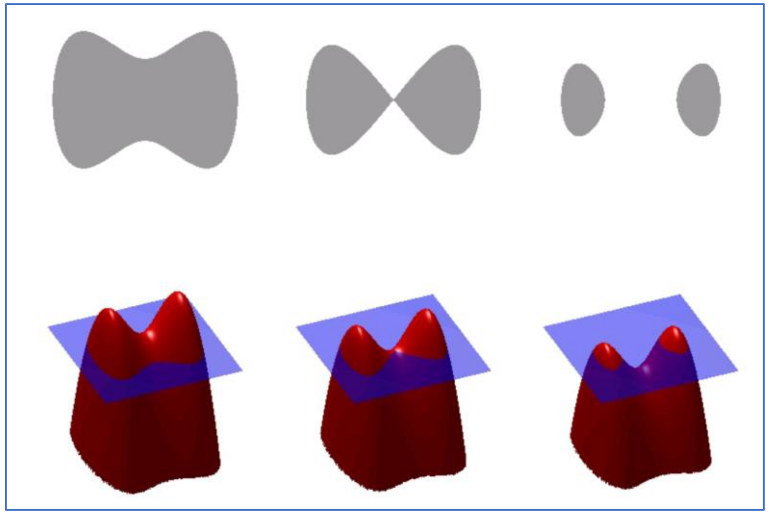
Volume Data: $f(x,y,z)$ known in each point (x,y,z) of a domain.

(in practice f is know in a set of sample points of the domains)

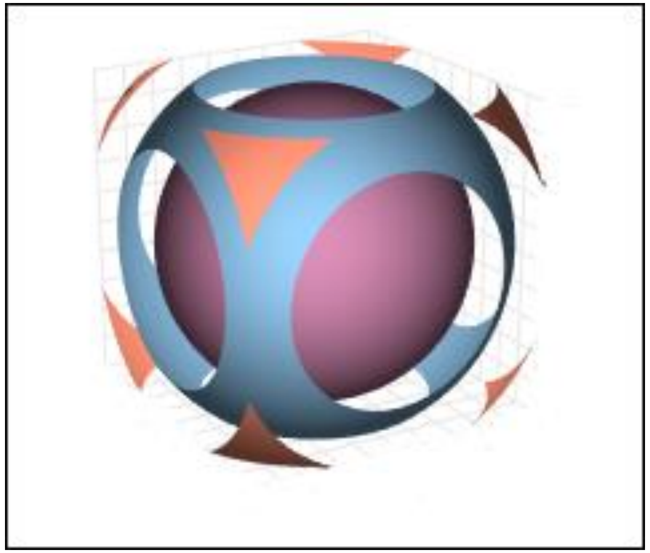
$f: R^3 \dashrightarrow R$ (assigns a scalar to each point of the domain)

Problem: Find the set $Sc = \{(x,y,z) \mid f(x,y,z)=c\}$

In 2D

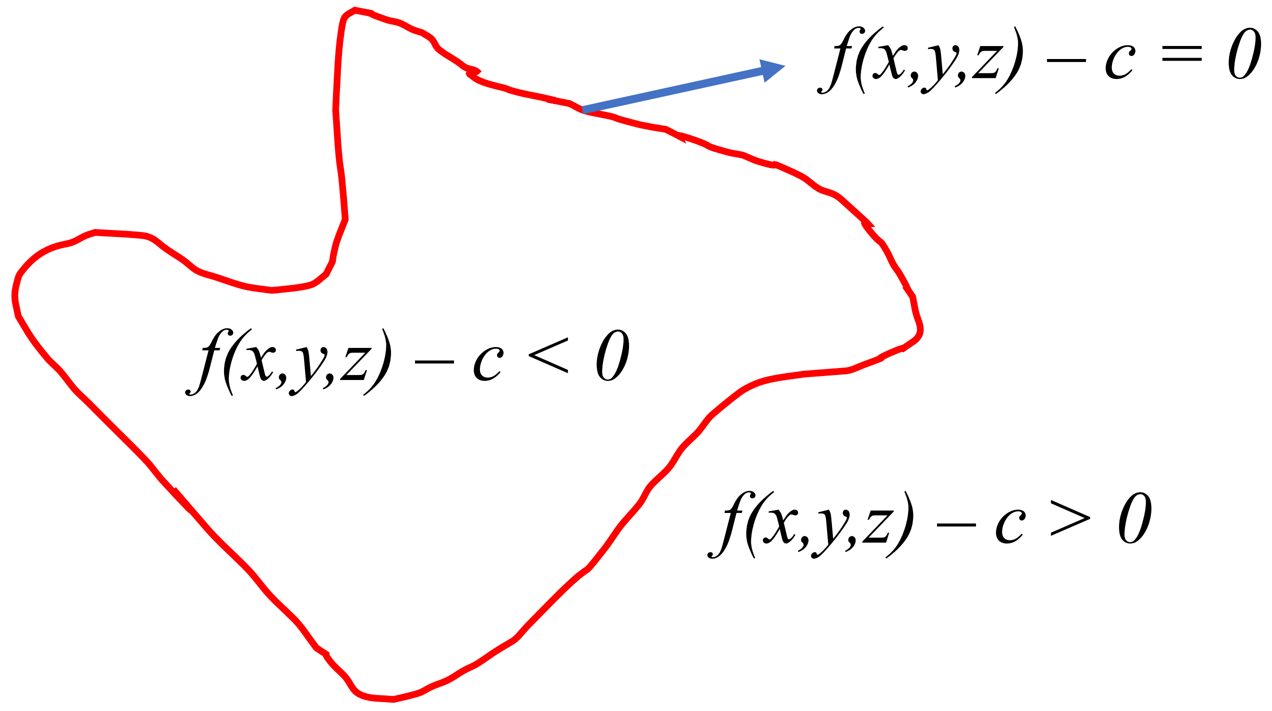


In 3D



Isosurfacing

Given the isovalue c, the sought solution is given by the points where $f(x,y,z) - c$ change its signal.



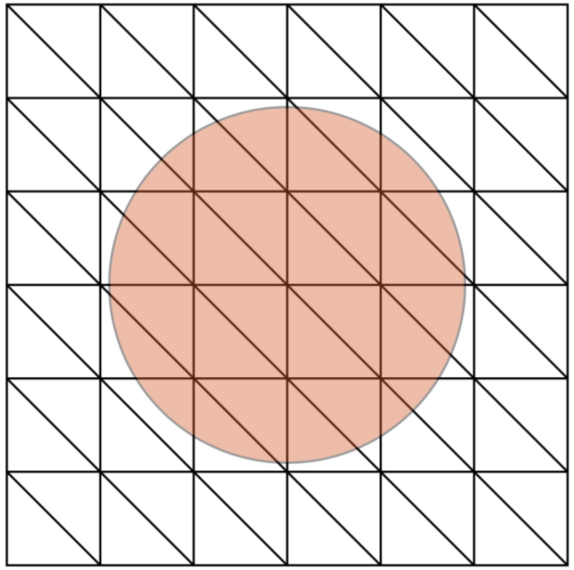
Marching Tetrahedra: Bloomenthal, 1998

Algorithm’s Steps

Marching Tetrahedra: Bloomenthal, 1998

Algorithm's Steps

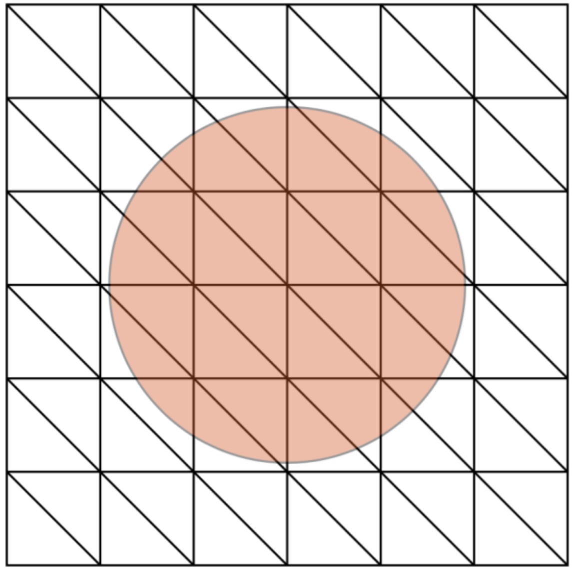
1. Decompose the domain in a tetrahedral mesh (triangles in 2D)



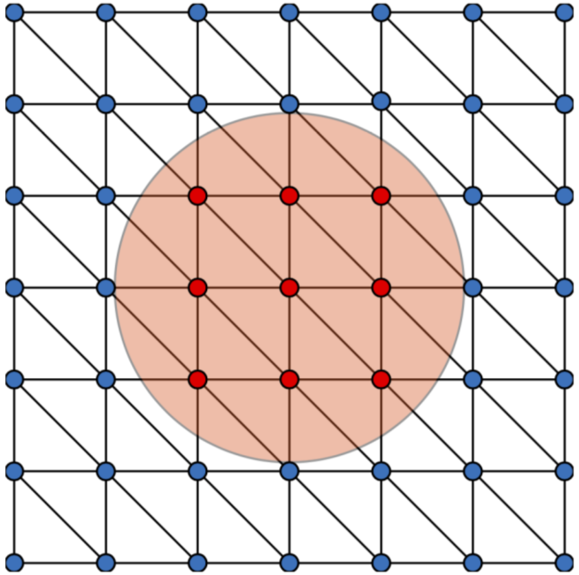
Marching Tetrahedra: Bloomenthal, 1998

Algorithm's Steps

1. Decompose the domain in a tetrahedral mesh (triangles in 2D)



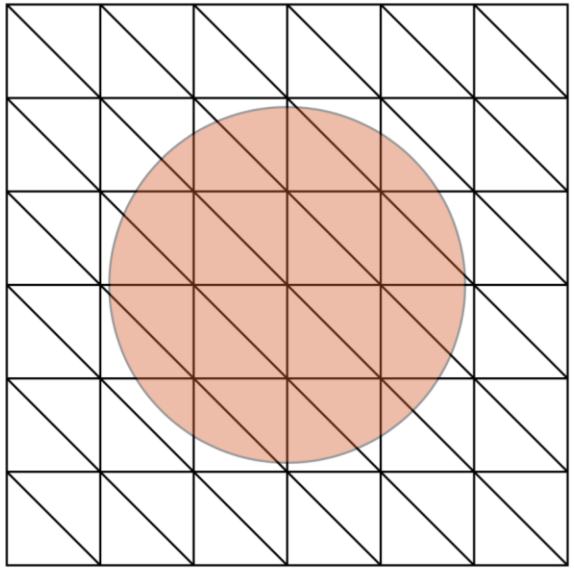
2. Evaluate f in the nodes of the mesh

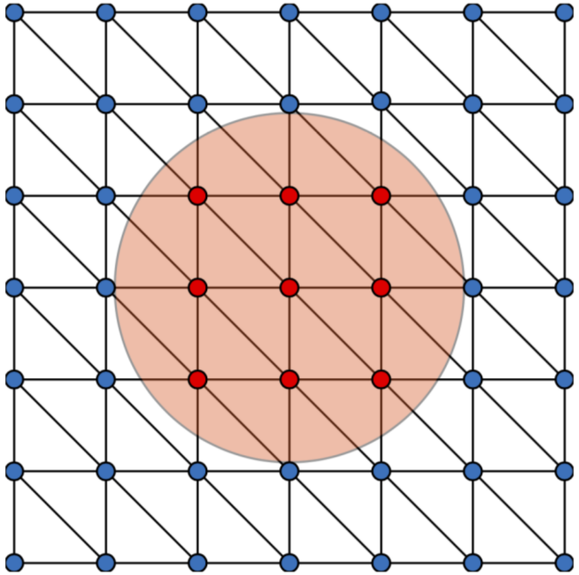


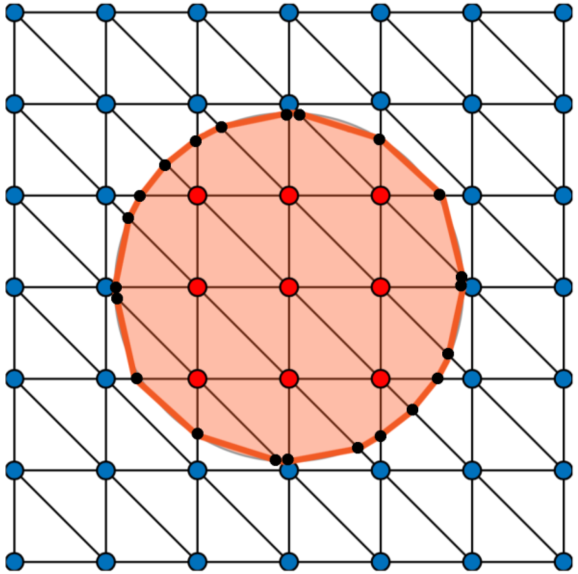
Marching Tetrahedra: Bloomenthal, 1998

Algorithm's Steps

1. Decompose the domain in a tetrahedral mesh (triangles in 2D)
2. Evaluate f in the nodes of the mesh
3. Find the points $f=0$ on edges where f changes its signal and generate the surface







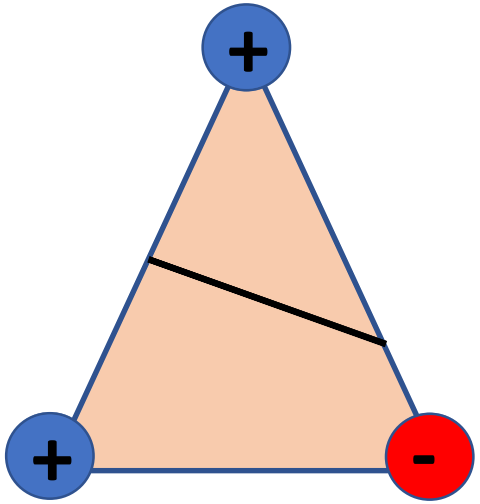
Marching Tetrahedra: Bloomenthal, 1998

Table of possible cases:

Marching Tetrahedra: Bloomenthal, 1998

Table of possible cases:

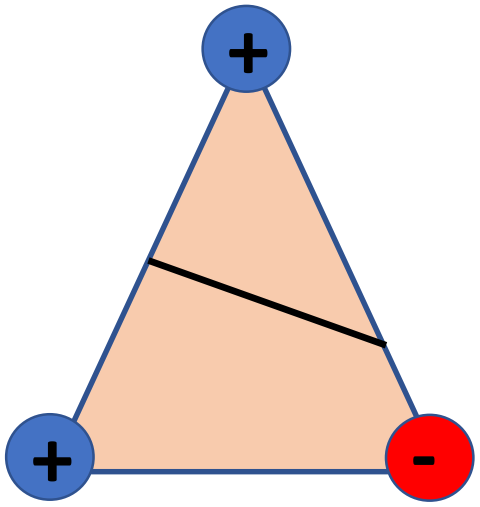
1 case for triangles



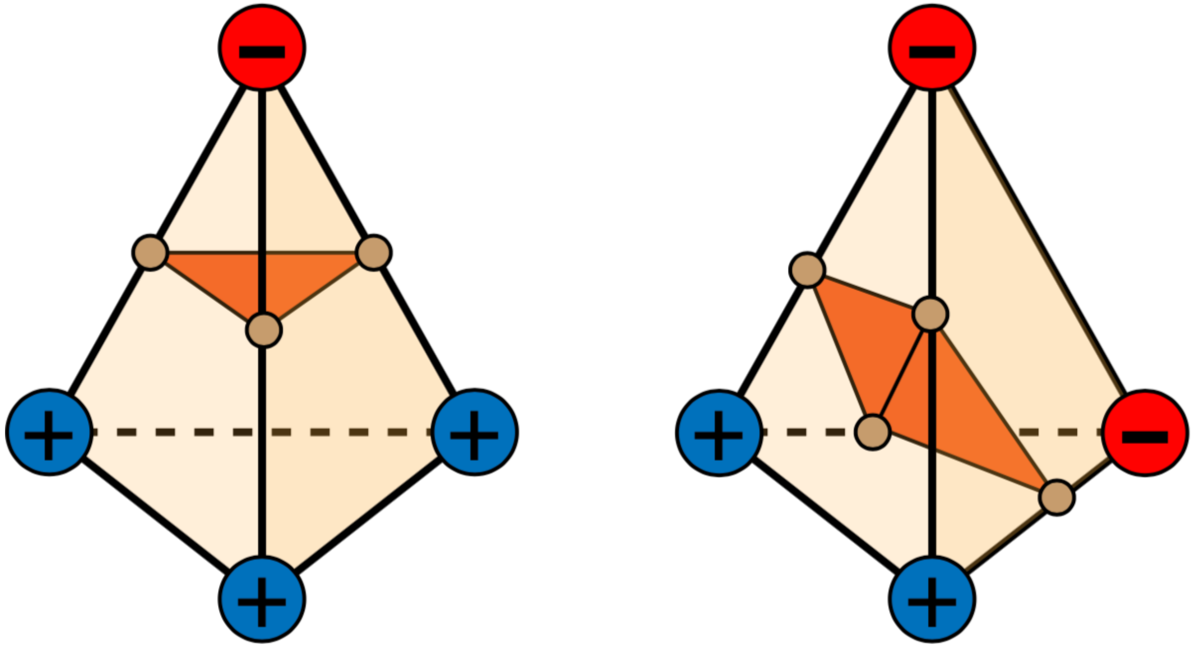
Marching Tetrahedra: Bloomenthal, 1998

Table of possible cases:

1 case for triangles

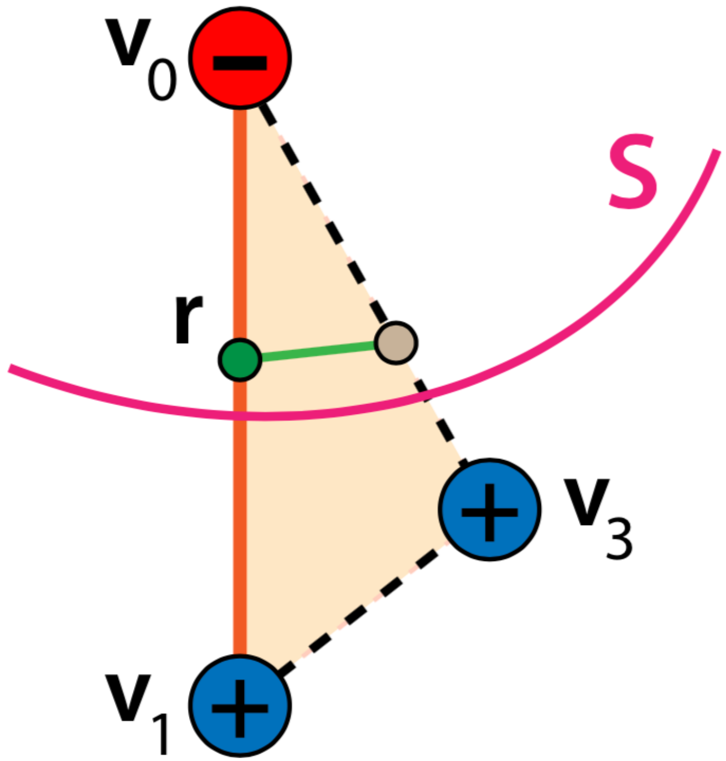


2 cases for tetrahedra



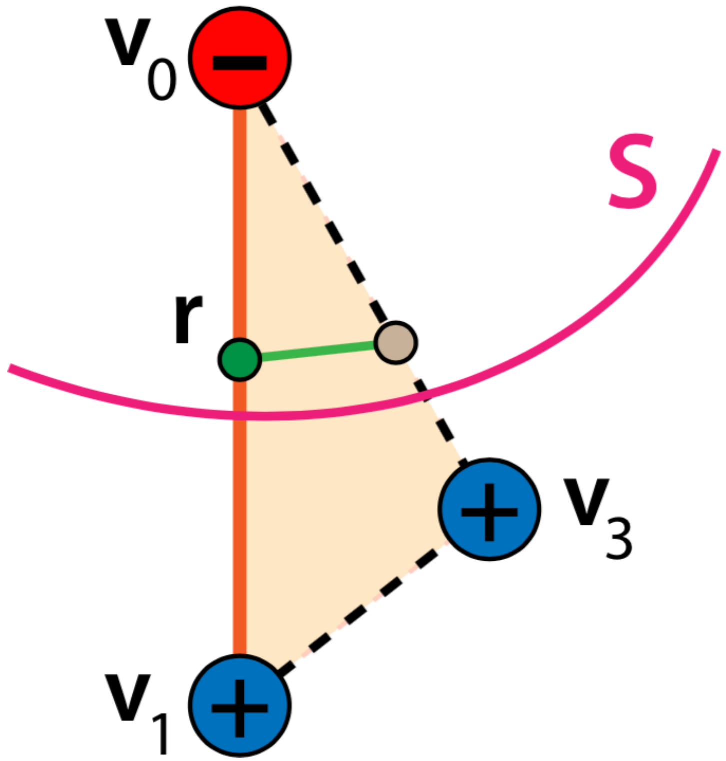
Marching Tetrahedra: Bloomenthal, 1998

Piecewise Linear Approximation:



Marching Tetrahedra: Bloomenthal, 1998

Piecewise Linear Approximation:

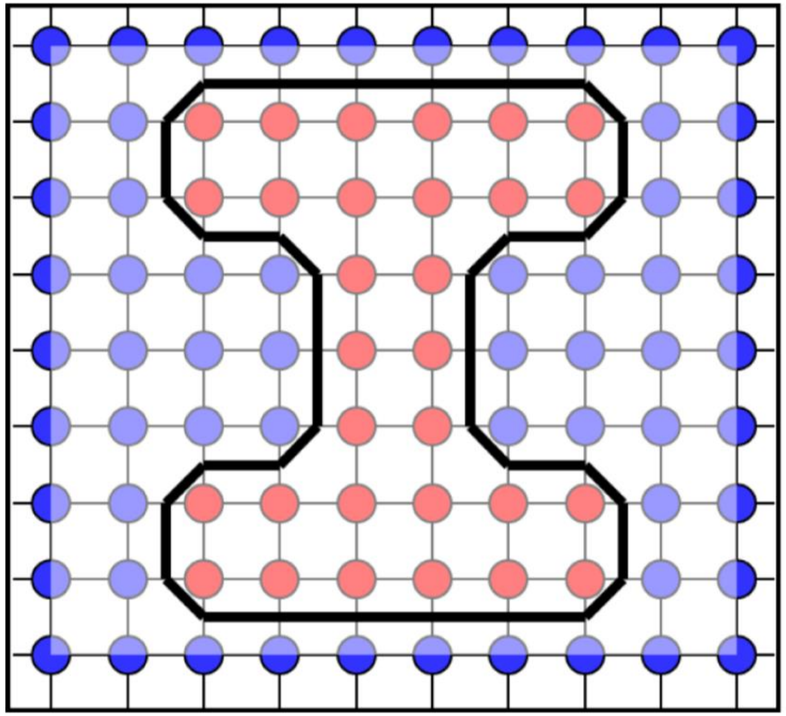


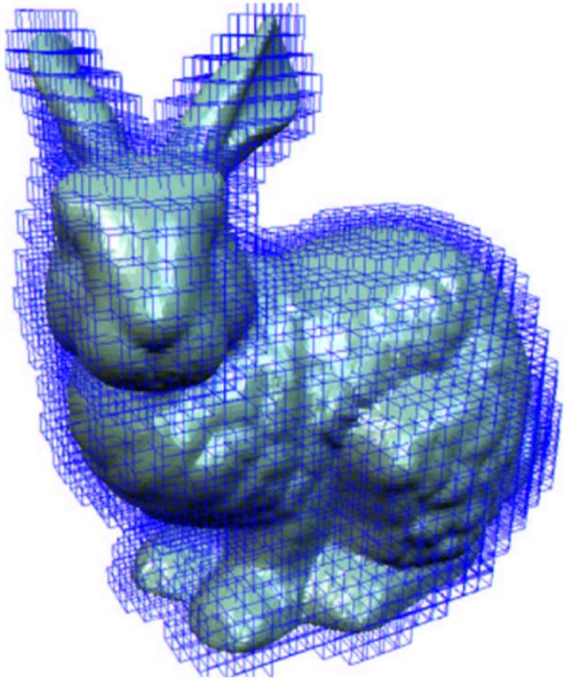
$$\mathbf{r} = (1-t)\,\mathbf{v}_0 + t\,\mathbf{v}_1$$

$$\begin{aligned} 0 \quad &= \quad f(\mathbf{r}) = f((1-t)\,\mathbf{v}_0 + t\,\mathbf{v}_1) \\ &\approx \quad (1-t)\,f(\mathbf{v}_0) + t\,f(\mathbf{v}_1) \end{aligned}$$

$$t = \frac{f(\mathbf{v}_0)}{f(\mathbf{v}_0) - f(\mathbf{v}_1)}$$

Marching Cubes: Lorensen & Cline, 1987





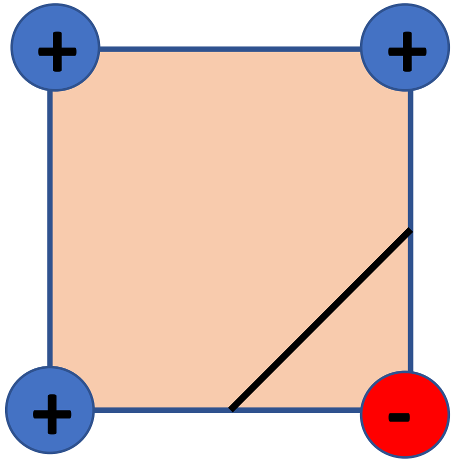
It follows the same idea as the marching tetrahedra, but relying in a Cartesian grid domain decomposition (cubic cells in 3D, squares in 2D)

Marching Cubes: Lorensen & Cline, 1987

3 cases in 2D

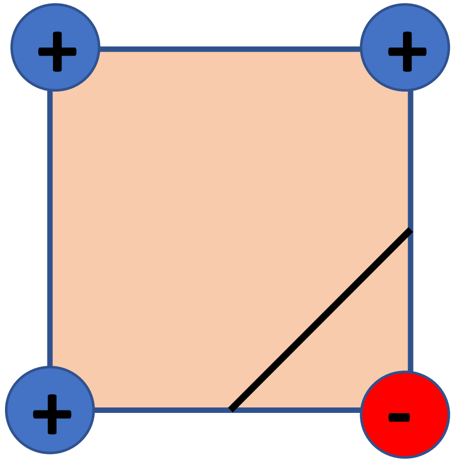
Marching Cubes: Lorensen & Cline, 1987

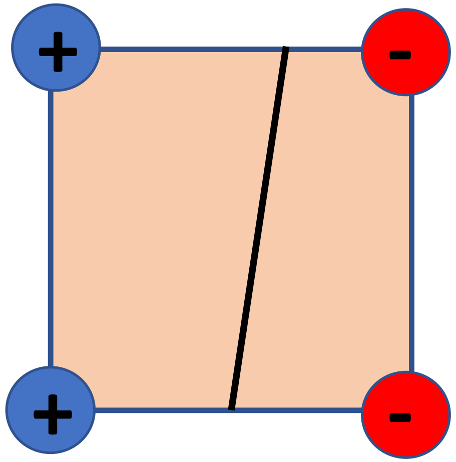
3 cases in 2D



Marching Cubes: Lorensen & Cline, 1987

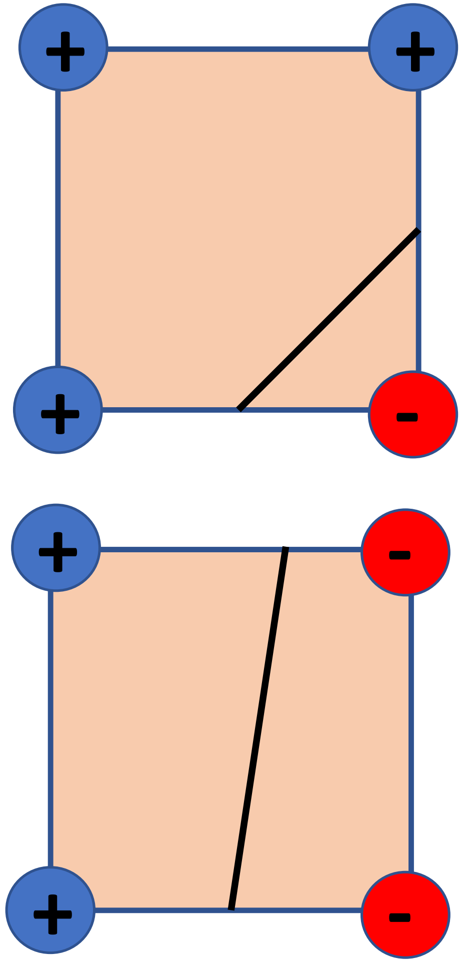
3 cases in 2D

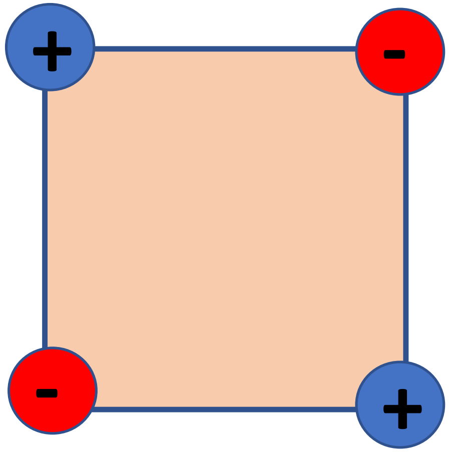




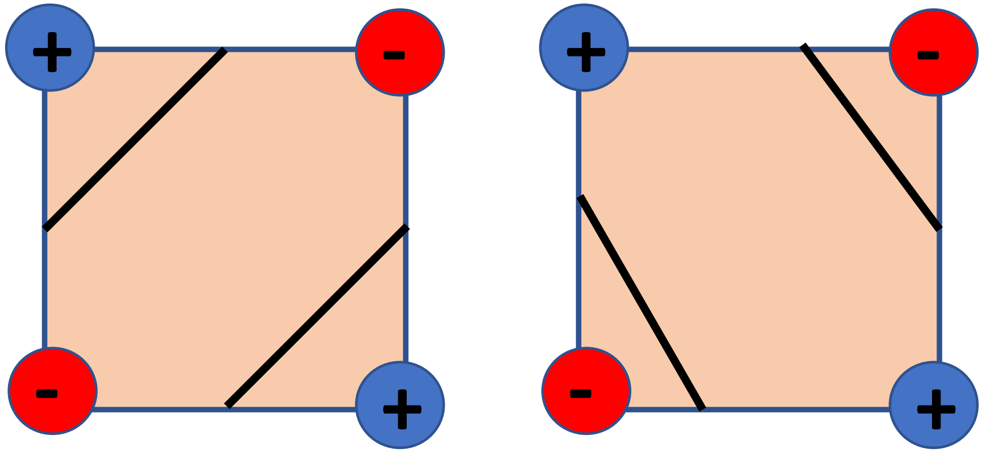
Marching Cubes: Lorensen & Cline, 1987

3 cases in 2D



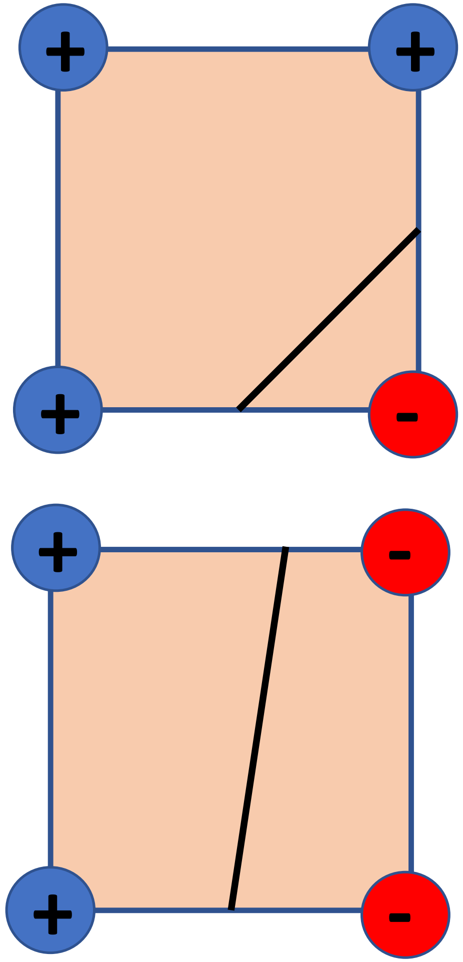


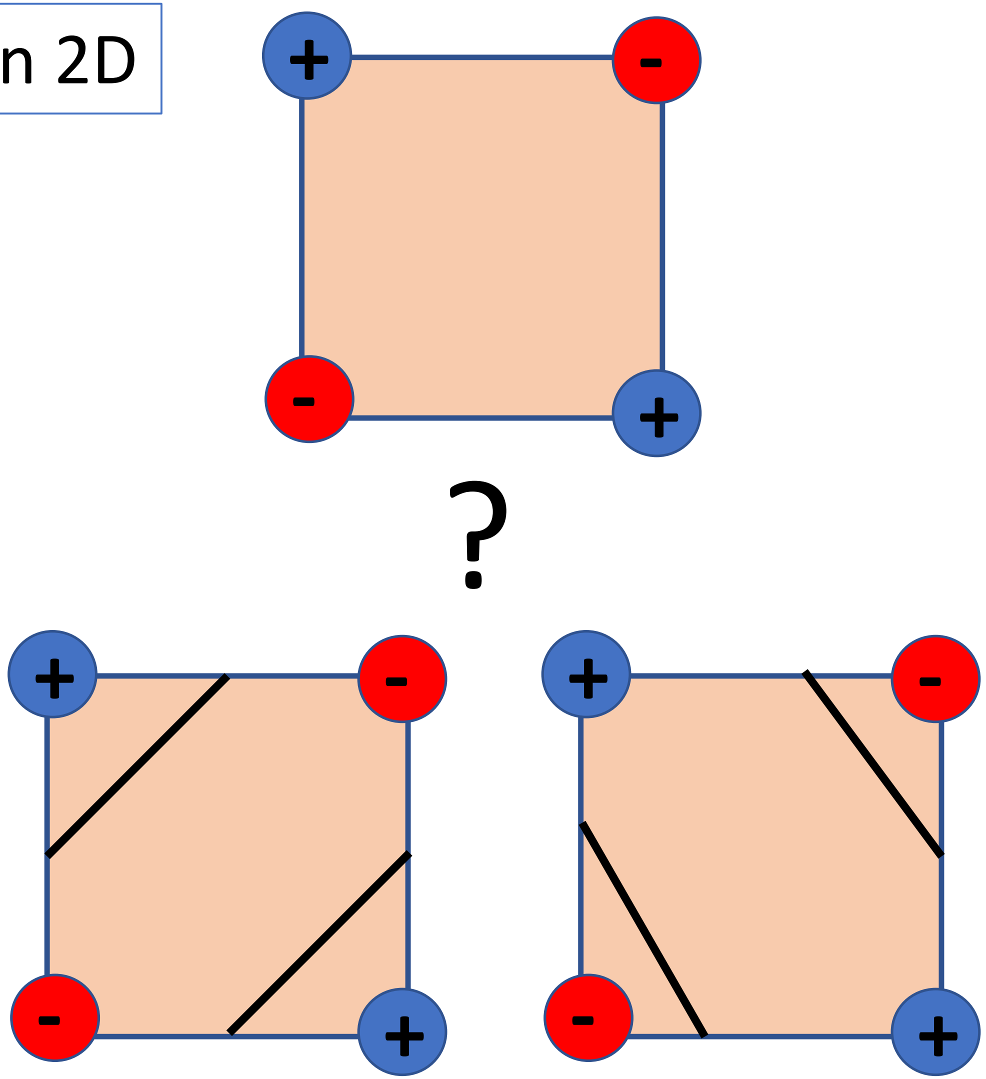
?

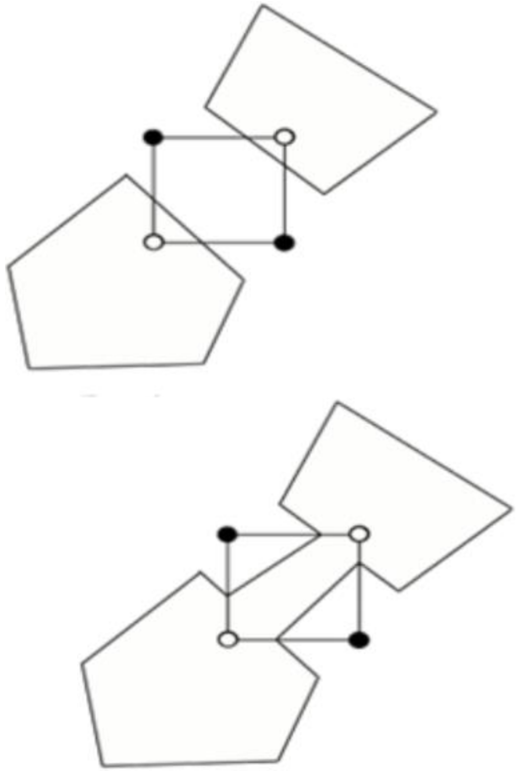


Marching Cubes: Lorensen & Cline, 1987

3 cases in 2D

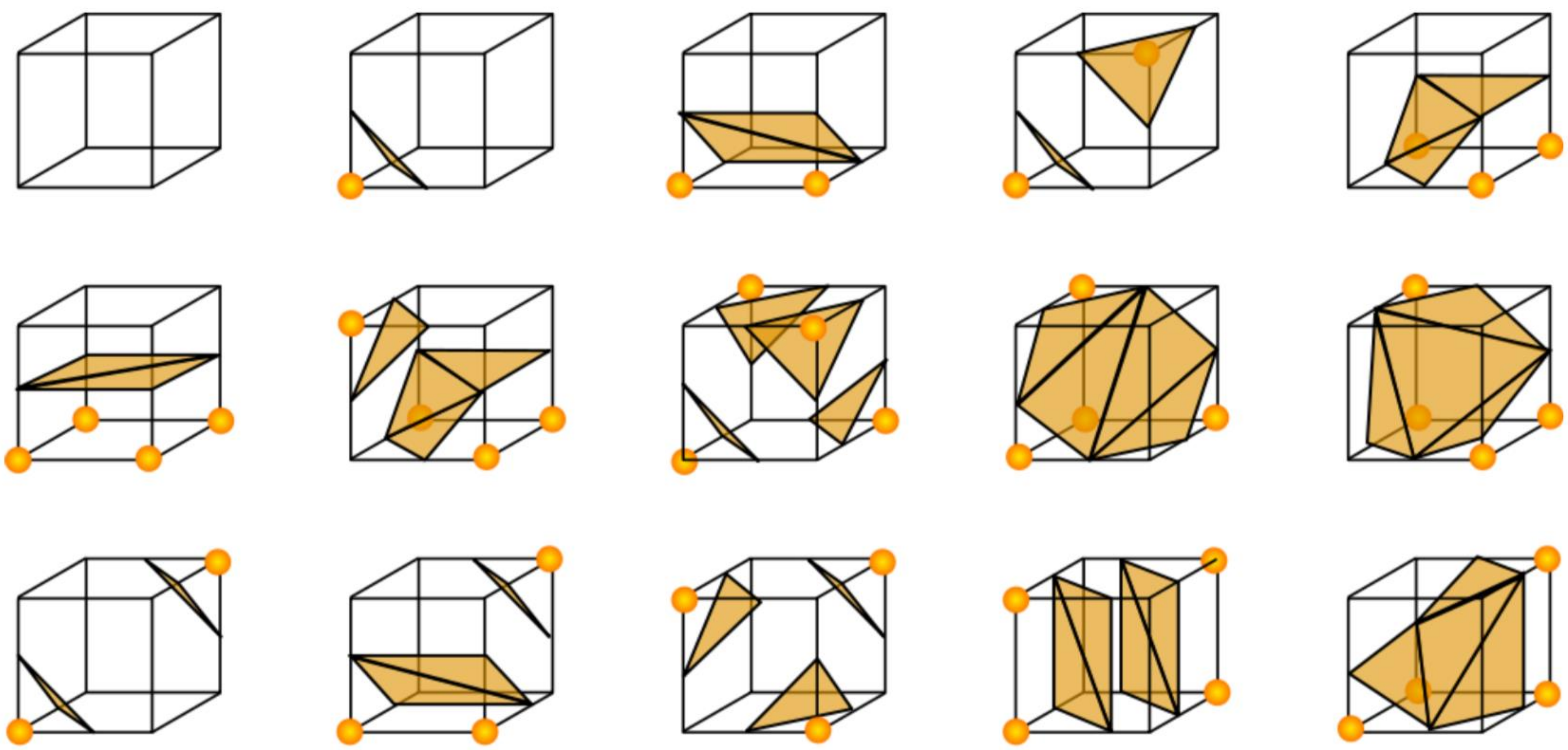






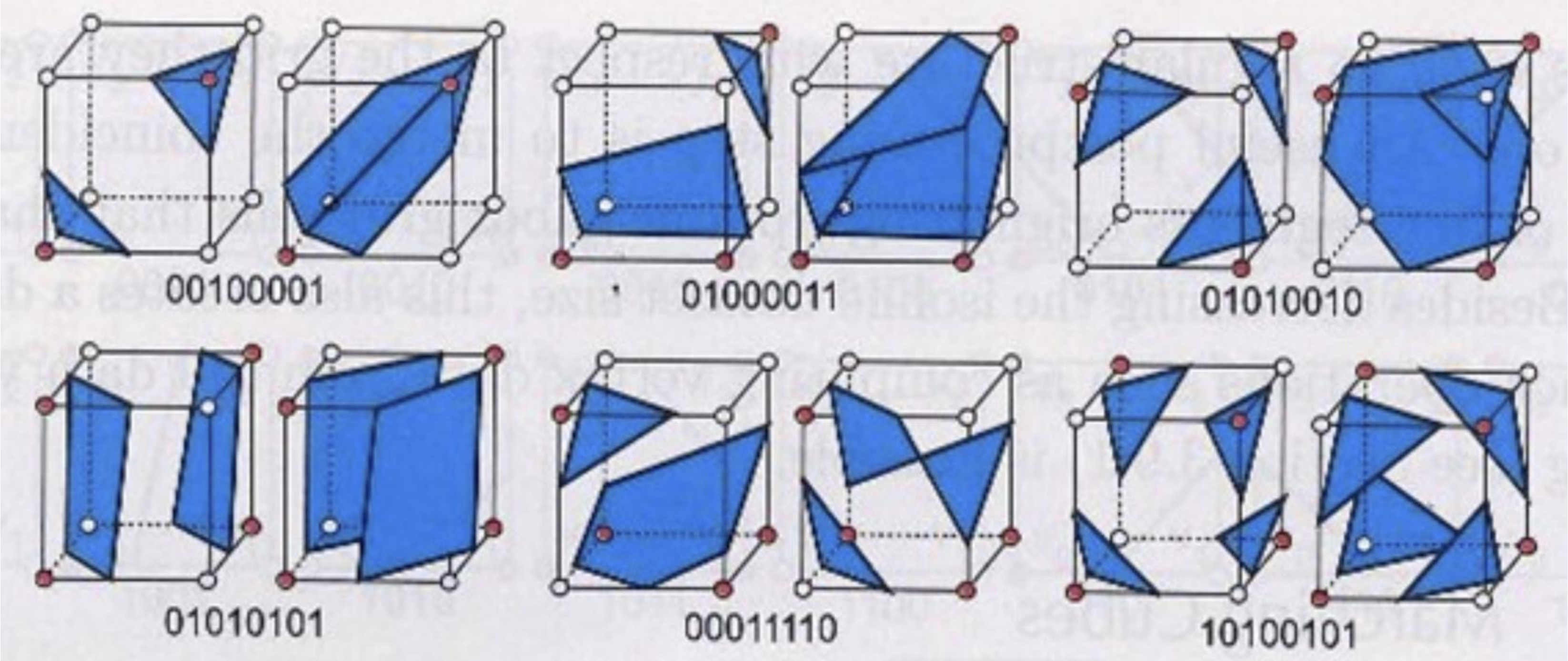
Marching Cubes: Lorensen & Cline, 1987

15 cases in 3D



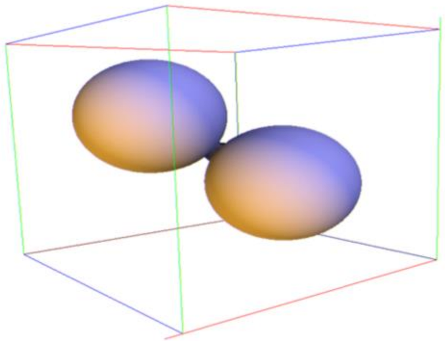
Marching Cubes: Lorensen & Cline, 1987

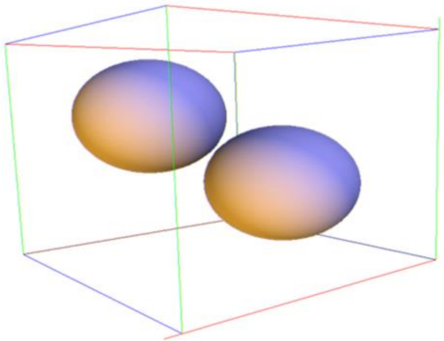
6 ambiguous cases

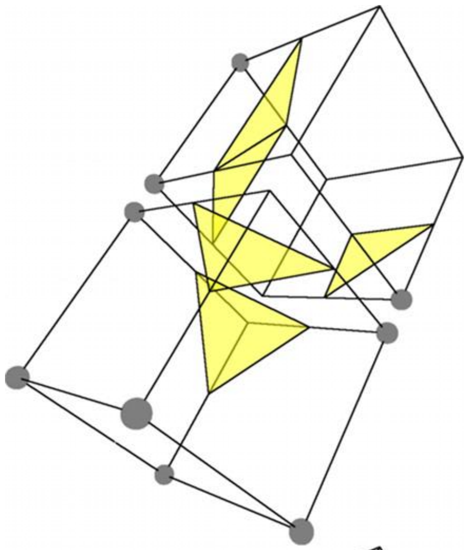


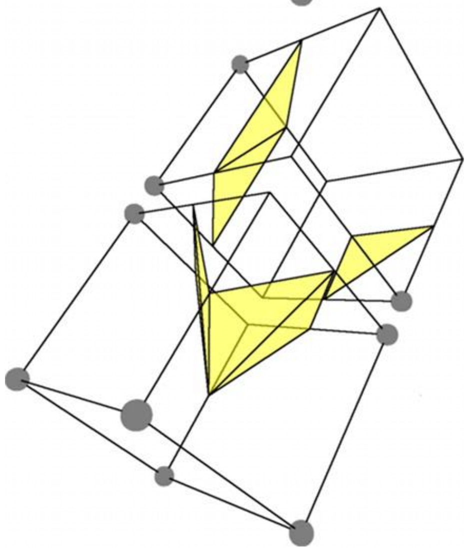
Marching Cubes: Lorensen & Cline, 1987

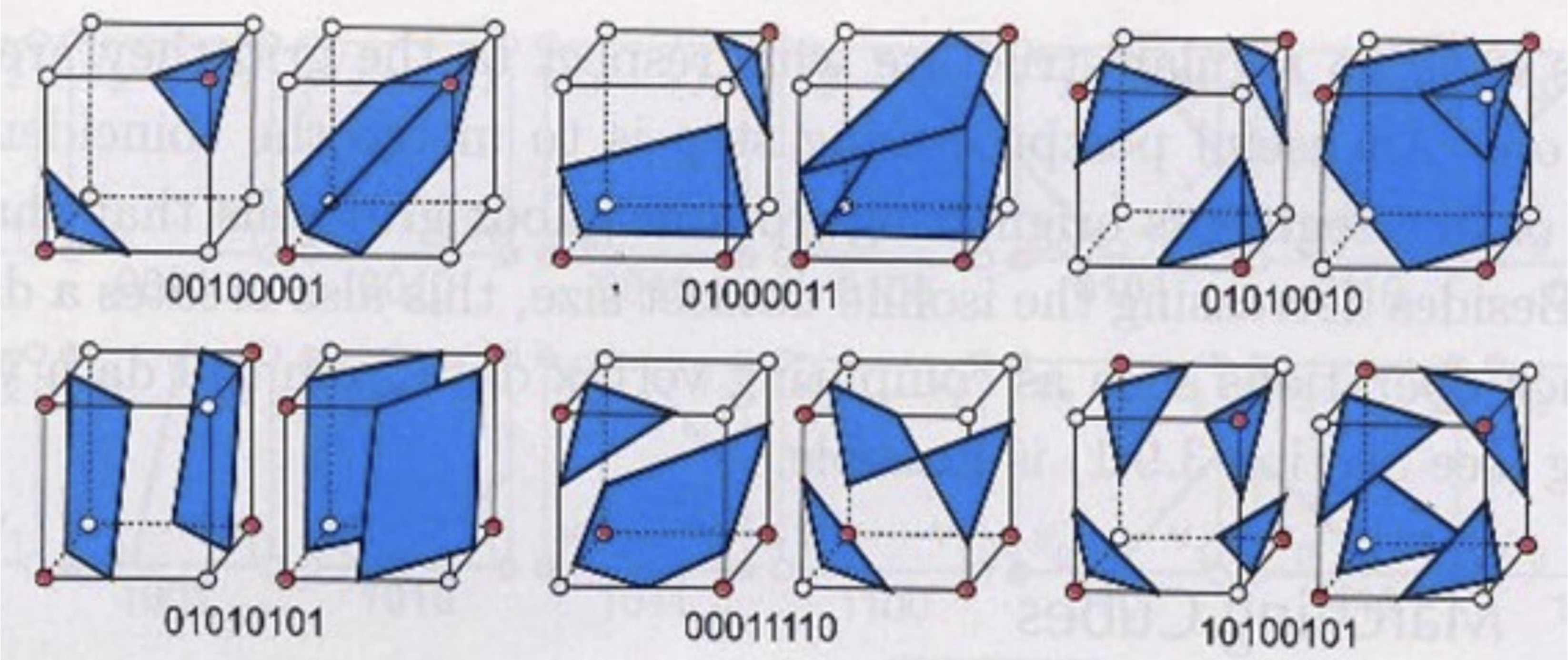
6 ambiguous cases







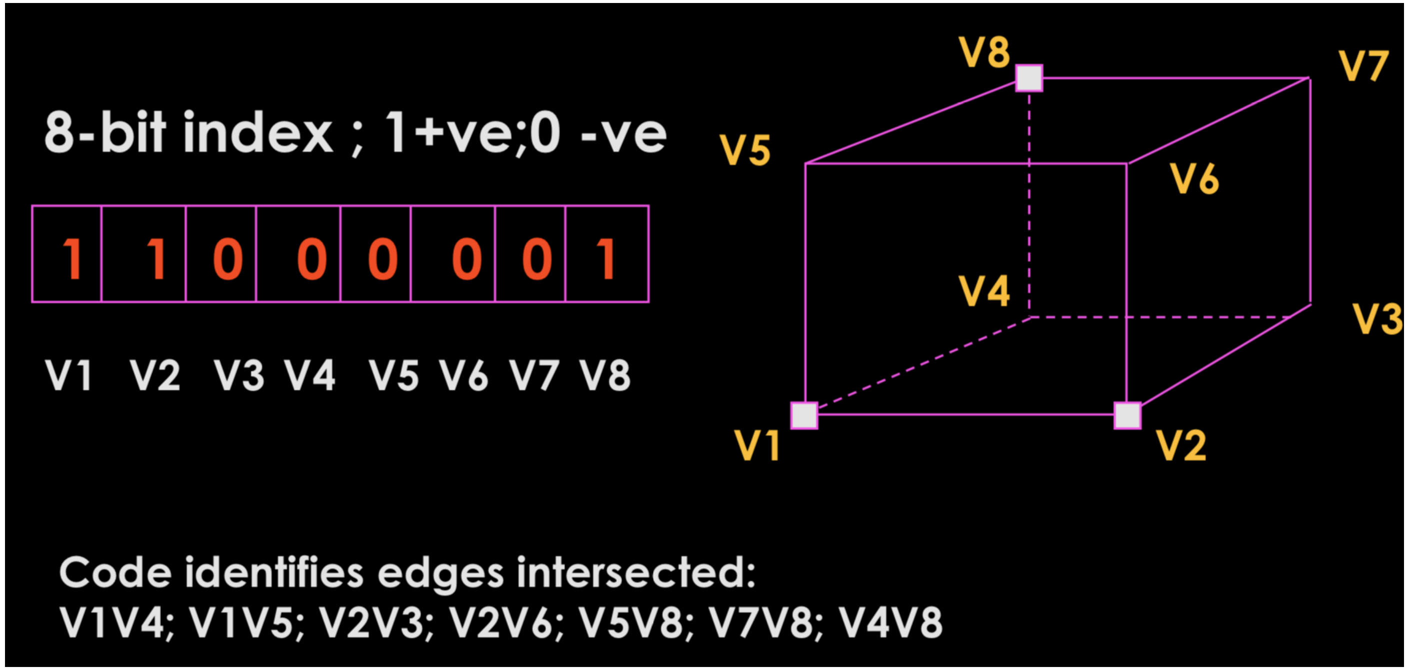




Marching Cubes: Lorensen & Cline, 1987

Algorithm’s Steps

1. Evaluate the function in the grid nodes
2. Classify the eight vertices relative to the isosurface value



Marching Cubes: Lorensen & Cline, 1987

Algorithm's Steps

3. Use a lookup table that identifies the cases

00000000	Configuration 0
10000000	Configuration 1
01000000	Configuration 1
...	
11000001	Configuration 6
...	
11111111	Configuration 0

256 configurations total

Marching Cubes: Lorensen & Cline, 1987

Algorithm's Steps

4. Linear interpolation along the identified edges to locate the intersection points
5. The lookup table determines how the pieces of the isosurface are created (0, 1, 2, 3 or 4 triangles)
6. Output the triangles

Algorithm marches from cube to cube produce a triangulated surface.

Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

- generates sub-cases for each of the 6 ambiguous configurations
- which sub-case is chosen depends on the value of the saddle-point on the face
- note that some configurations have several ambiguous faces so many subcases arise

Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

The solution relies on trilinear interpolation

$$f(x,y,z) \approx a_0 + a_1 x + a_2 y + a_3 z + a_4 xy + a_5 xz + a_6 yz + a_7 xyz$$

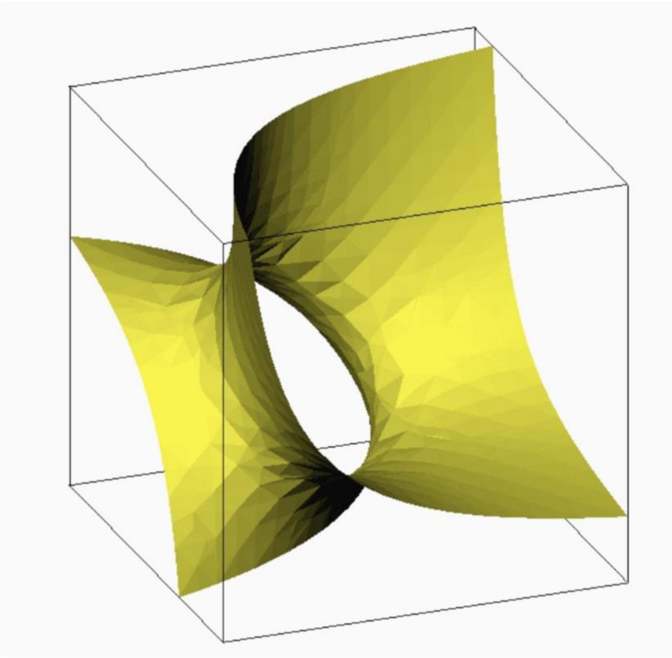
$$\begin{bmatrix} 1 & x_0 & y_0 & z_0 & x_0 y_0 & x_0 z_0 & y_0 z_0 & x_0 y_0 z_0 \\ 1 & x_1 & y_0 & z_0 & x_1 y_0 & x_1 z_0 & y_0 z_0 & x_1 y_0 z_0 \\ 1 & x_0 & y_1 & z_0 & x_0 y_1 & x_0 z_0 & y_1 z_0 & x_0 y_1 z_0 \\ 1 & x_1 & y_1 & z_0 & x_1 y_1 & x_1 z_0 & y_1 z_0 & x_1 y_1 z_0 \\ 1 & x_0 & y_0 & z_1 & x_0 y_0 & x_0 z_1 & y_0 z_1 & x_0 y_0 z_1 \\ 1 & x_1 & y_0 & z_1 & x_1 y_0 & x_1 z_1 & y_0 z_1 & x_1 y_0 z_1 \\ 1 & x_0 & y_1 & z_1 & x_0 y_1 & x_0 z_1 & y_1 z_1 & x_0 y_1 z_1 \\ 1 & x_1 & y_1 & z_1 & x_1 y_1 & x_1 z_1 & y_1 z_1 & x_1 y_1 z_1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} c_{000} \\ c_{100} \\ c_{010} \\ c_{110} \\ c_{001} \\ c_{101} \\ c_{011} \\ c_{111} \end{bmatrix}$$

Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

The solution relies on trilinear interpolation

$$f(x,y,z) \approx a_0 + a_1 x + a_2 y + a_3 z + a_4 xy + a_5 xz + a_6 yz + a_7 xyz$$



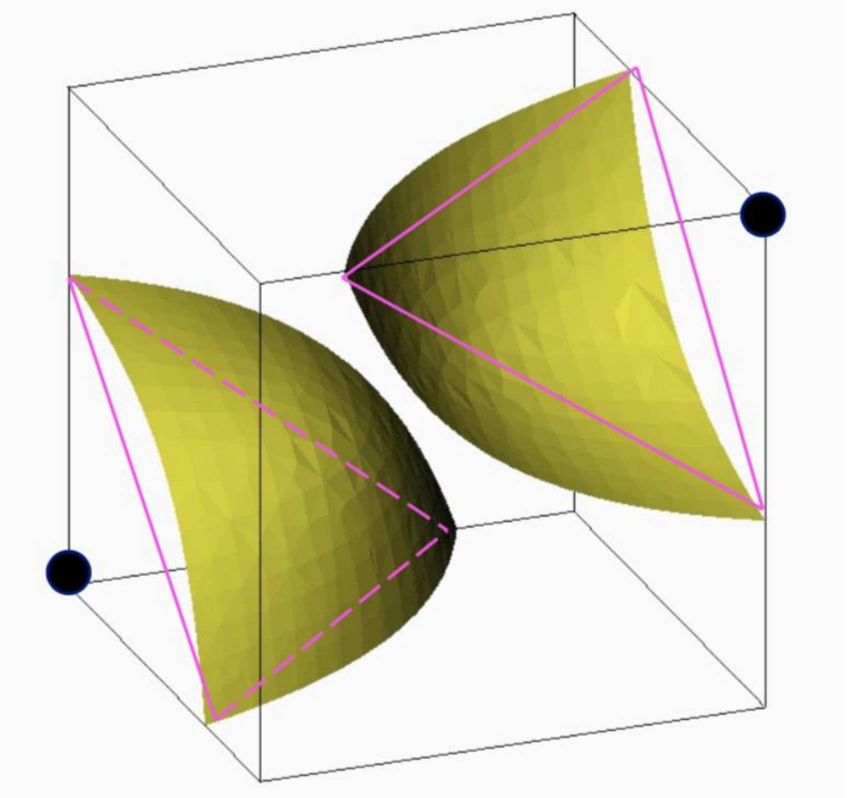
$$\begin{bmatrix} 1 & x_0 & y_0 & z_0 & x_0 y_0 & x_0 z_0 & y_0 z_0 & x_0 y_0 z_0 \\ 1 & x_1 & y_0 & z_0 & x_1 y_0 & x_1 z_0 & y_0 z_0 & x_1 y_0 z_0 \\ 1 & x_0 & y_1 & z_0 & x_0 y_1 & x_0 z_0 & y_1 z_0 & x_0 y_1 z_0 \\ 1 & x_1 & y_1 & z_0 & x_1 y_1 & x_1 z_0 & y_1 z_0 & x_1 y_1 z_0 \\ 1 & x_0 & y_0 & z_1 & x_0 y_0 & x_0 z_1 & y_0 z_1 & x_0 y_0 z_1 \\ 1 & x_1 & y_0 & z_1 & x_1 y_0 & x_1 z_1 & y_0 z_1 & x_1 y_0 z_1 \\ 1 & x_0 & y_1 & z_1 & x_0 y_1 & x_0 z_1 & y_1 z_1 & x_0 y_1 z_1 \\ 1 & x_1 & y_1 & z_1 & x_1 y_1 & x_1 z_1 & y_1 z_1 & x_1 y_1 z_1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} c_{000} \\ c_{100} \\ c_{010} \\ c_{110} \\ c_{001} \\ c_{101} \\ c_{011} \\ c_{111} \end{bmatrix}$$

$f(x,y,z)-c=0$ is now a cubic surface

Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

We are approximating a curved surface by triangles



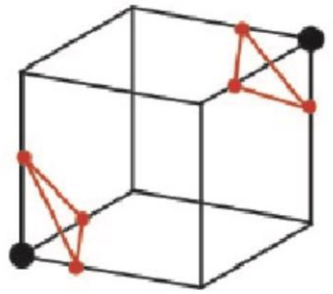
Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

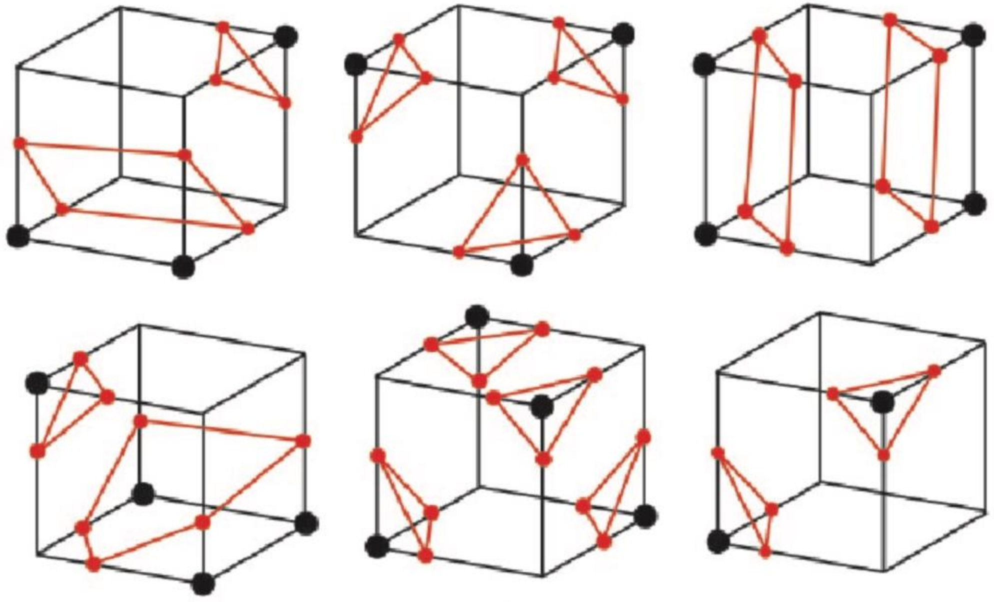
When constrained to the faces of the cube, the trilinear interpolation becomes bilinear.

We have two cases, ambiguity on the faces and in the interior of the cube.

Ambiguity on faces



Ambiguity on faces



Marching Cubes: Lorensen & Cline, 1987

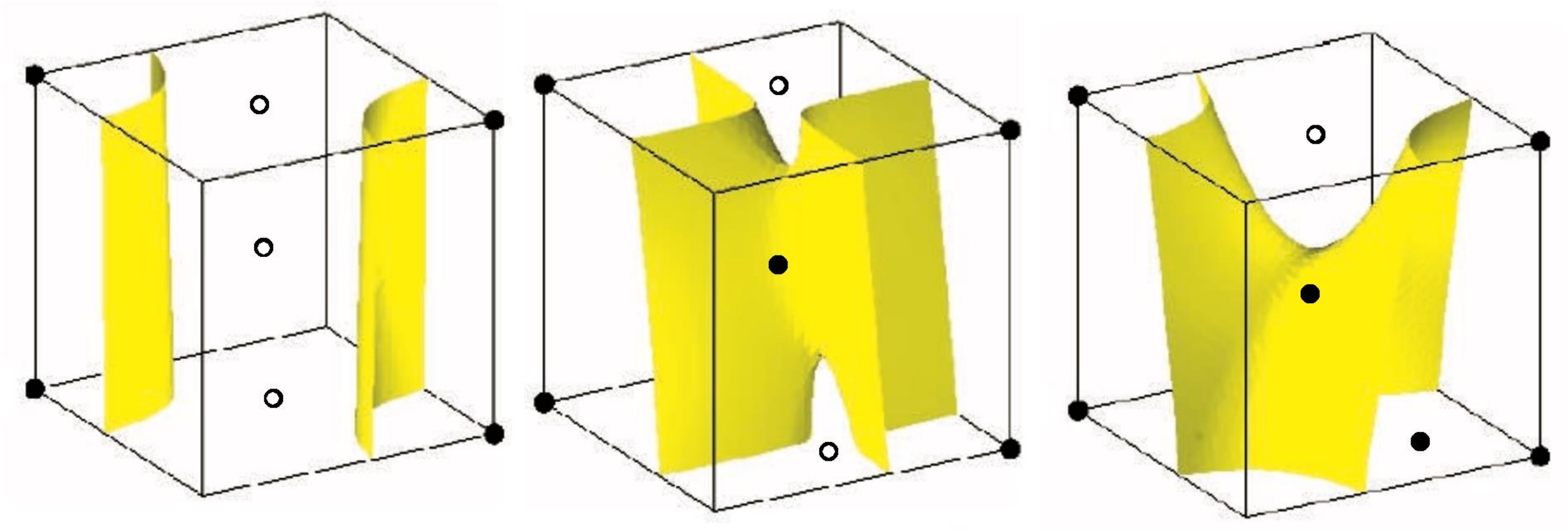
Dealing with ambiguity: The saddle point method

Solution:
- Compute the saddle point
- Connect the nodes sharing the same sign as the saddle point

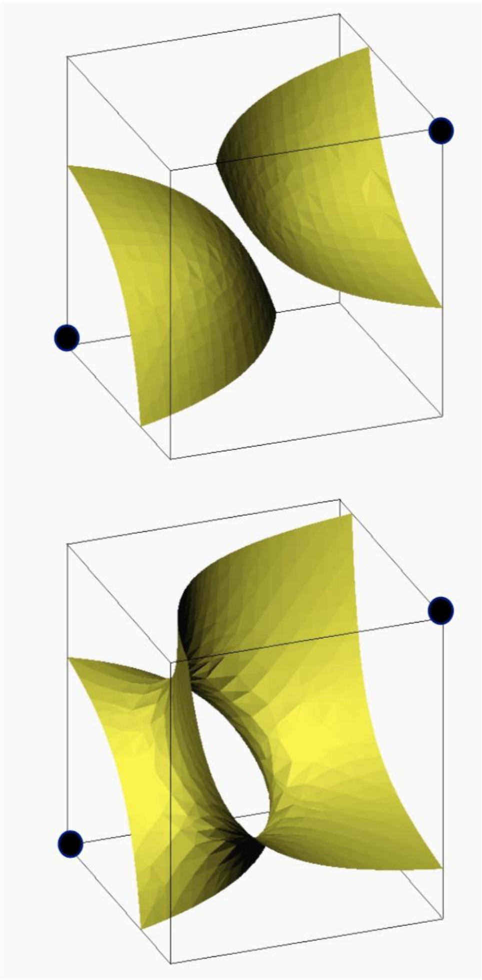
Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: The saddle point method

Solution:
- Compute the saddle point
- Connect the nodes sharing the same sign as the saddle point



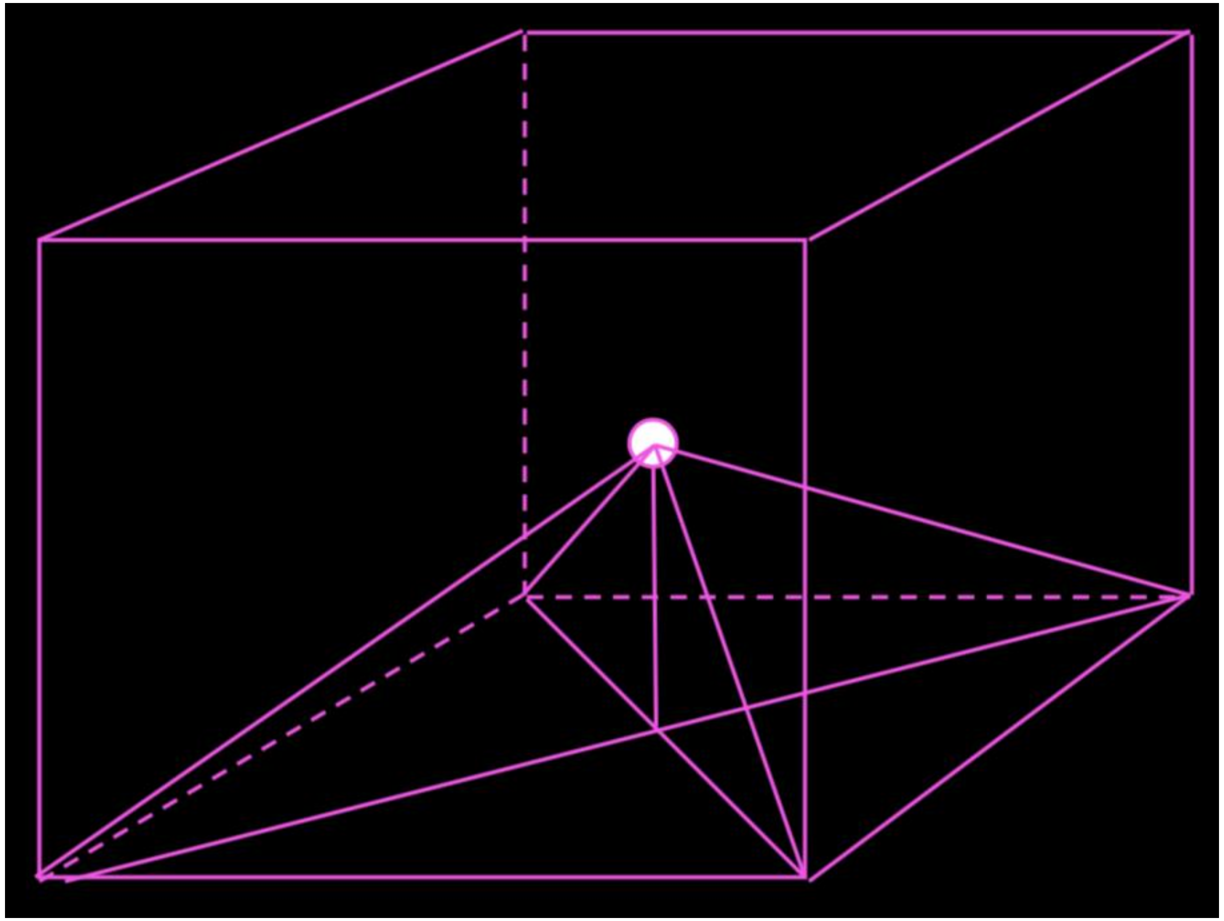
Saddle points used to solve ambiguities



Marching Cubes: Lorensen & Cline, 1987

Dealing with ambiguity: Tetrahedralization

Solution:
- Build a tetrahedral mesh in ambiguous cases
- A much larger lookup has to be used



Marching Cubes: Lorensen & Cline, 1987

Some references:

Original marching cubes algorithm
- – Lorensen and Cline (1987)

Handling Ambiguities
- – Nielson and Hamann (1992)
- – Natarajan (1994)
- – Lopes and Brodlie (2003)
- – Chernyaev (1995)