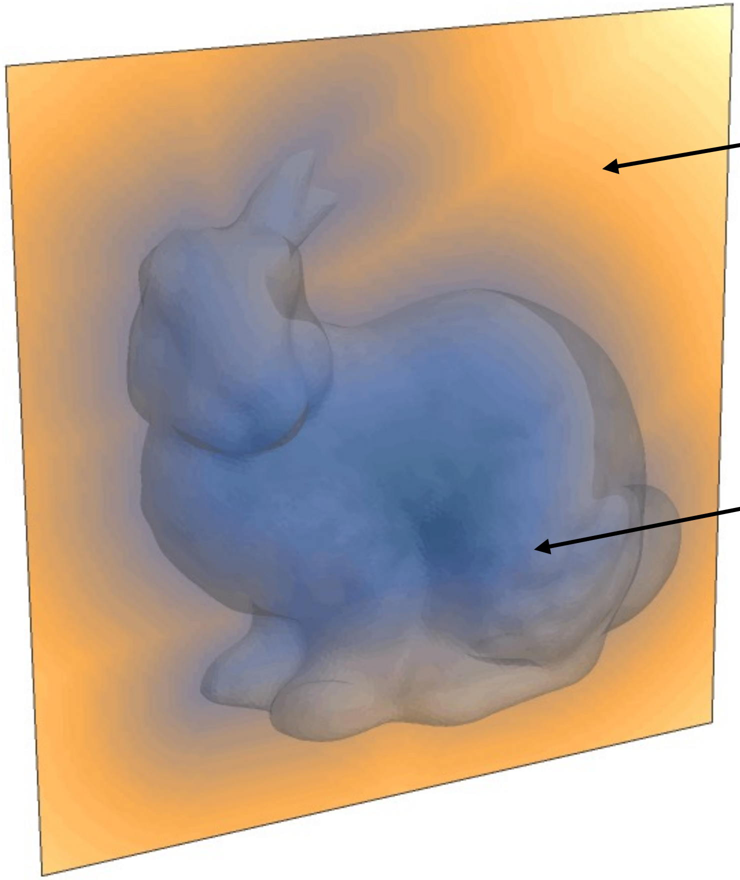
SIGGRAPH 2022
VANCOUVER+ 8-11 AUG

THE PREMIER CONFERENCE & EXHIBITION ON
COMPUTER GRAPHICS & INTERACTIVE TECHNIQUES

ROBUST COMPUTATION OF IMPLICIT SURFACE NETWORKS FOR PIECEWISE LINEAR FUNCTIONS

XINGYI DU, WASHINGTON UNIVERSITY IN ST. LOUIS
QINGNAN ZHOU, ADOBE RESEARCH
NATHAN CARR, ADOBE RESEARCH
TAO JU, WASHINGTON UNIVERSITY IN ST. LOUIS

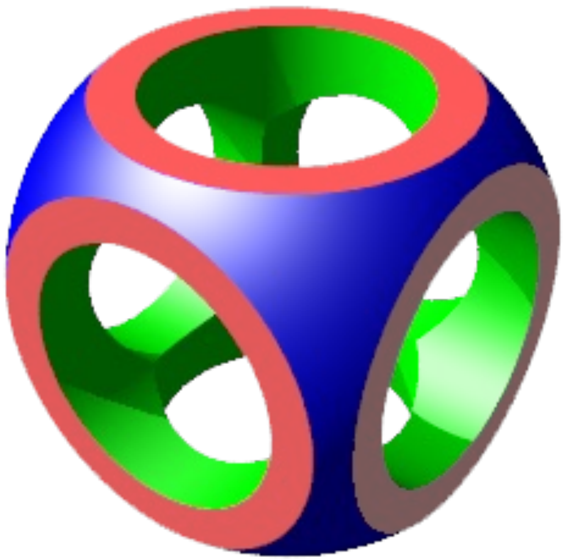
IMPLICIT SURFACES

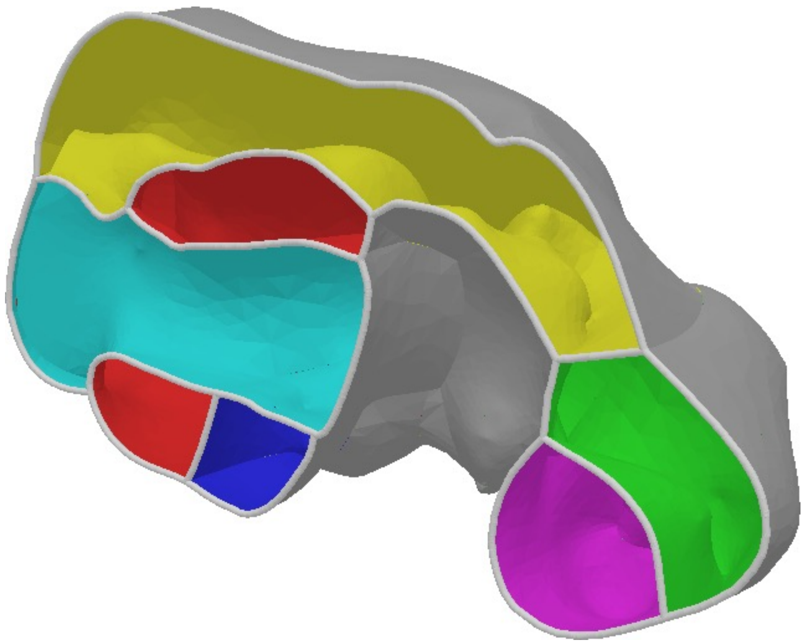


$f(x, y, z)$

$\{x, y, z\}\colon\ f(x, y, z) = 0$

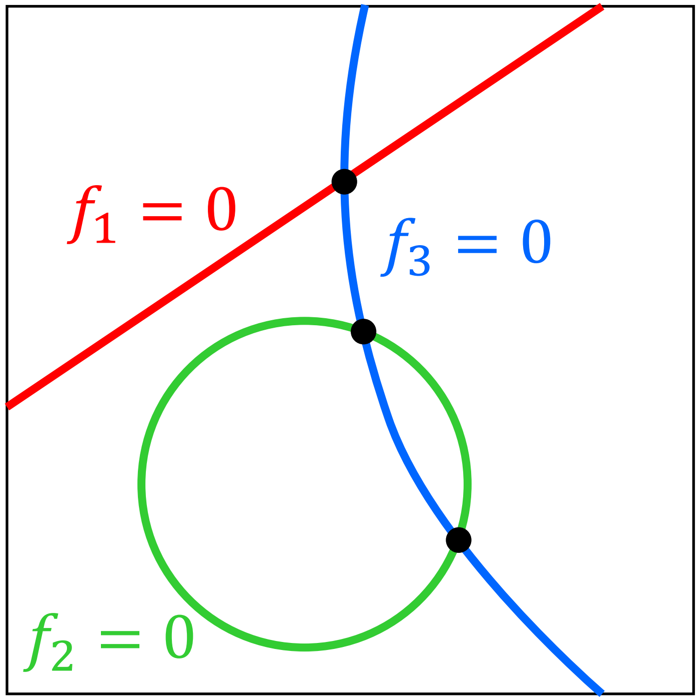
IMPLICIT SURFACE NETWORKS





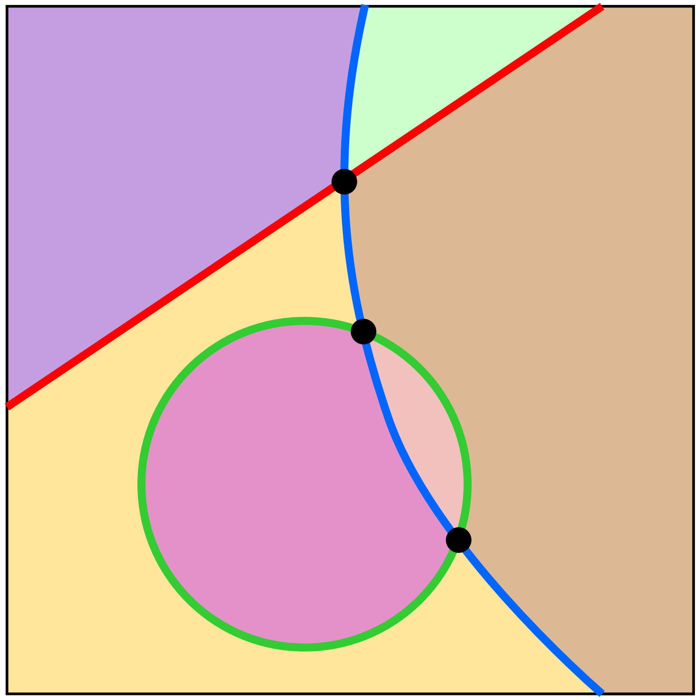
IMPLICIT SURFACE NETWORKS

- Implicit Arrangement (IA): Intersecting multiple implicit surfaces



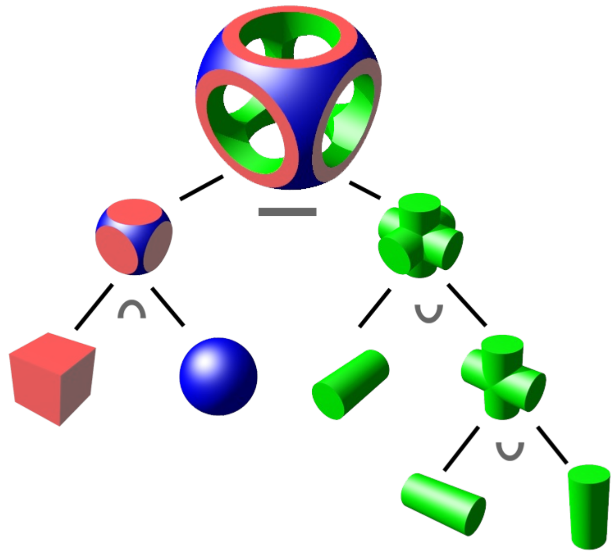
IMPLICIT SURFACE NETWORKS

- Implicit Arrangement (IA): Intersecting multiple implicit surfaces

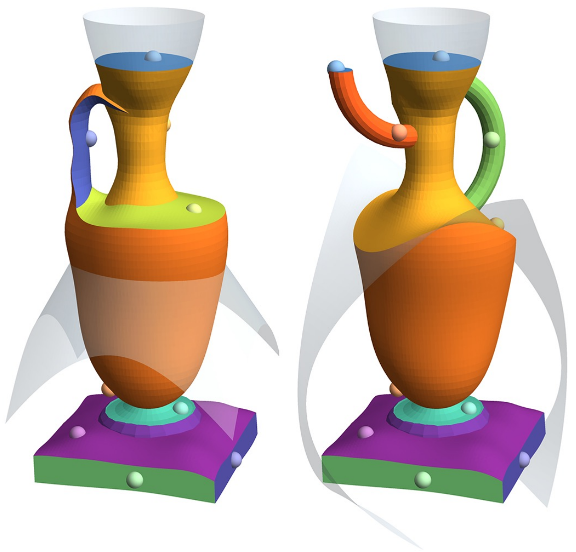


IMPLICIT SURFACE NETWORKS

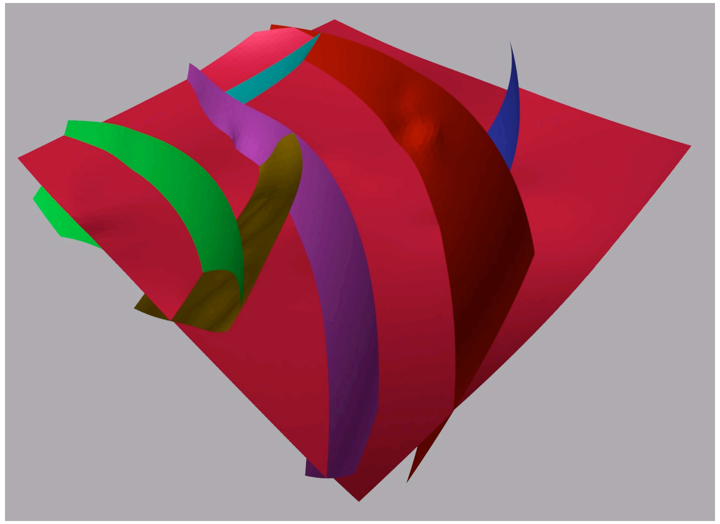
- Implicit Arrangement (IA): Intersecting multiple implicit surfaces



Constructive Solid Geometry (CSG)

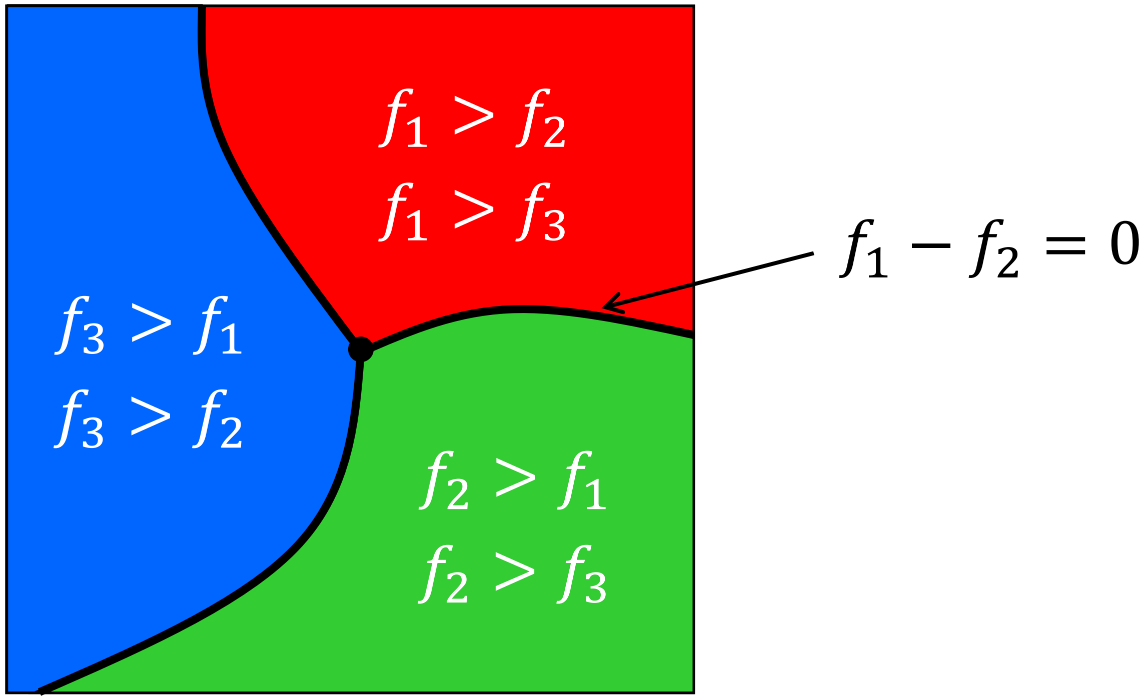


Boundary-sampled halfspaces [Du 21]



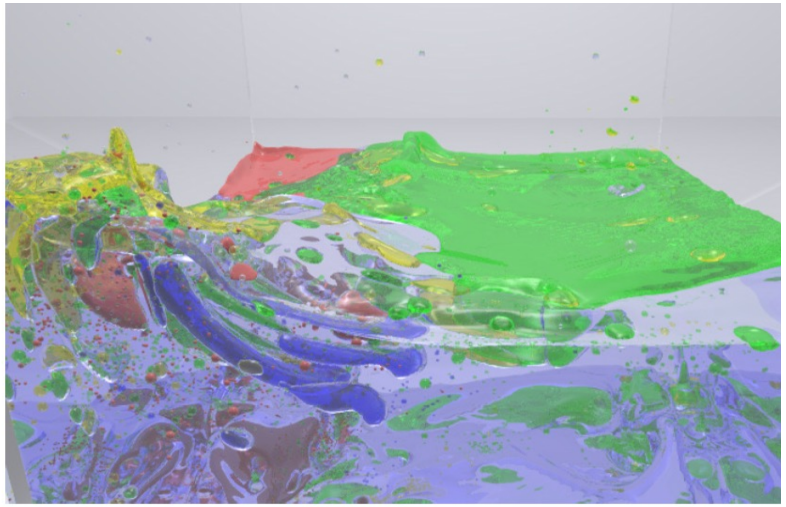
Geological structure [Bagley 16]

- Material Interfaces (MI): Boundary of regions dominated by one function

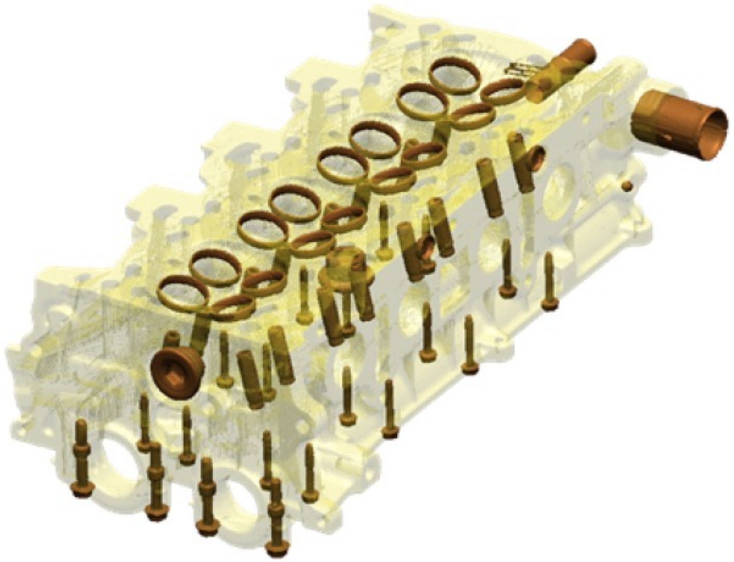


IMPLICIT SURFACE NETWORKS

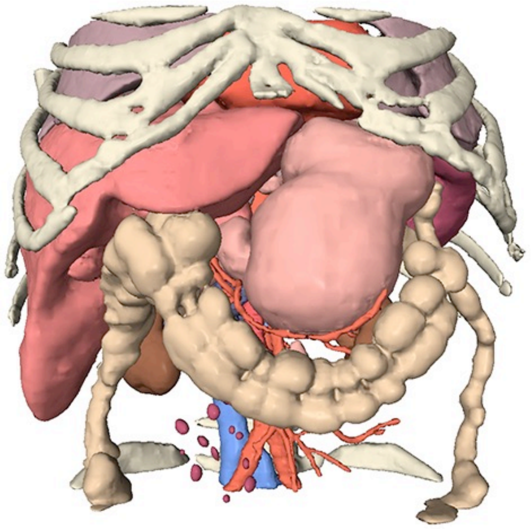
- Material Interfaces (MI): Boundary of regions dominated by one function



Multi-phase fluid
[Kim 10]



Mechanical parts
[Shammaa 08]

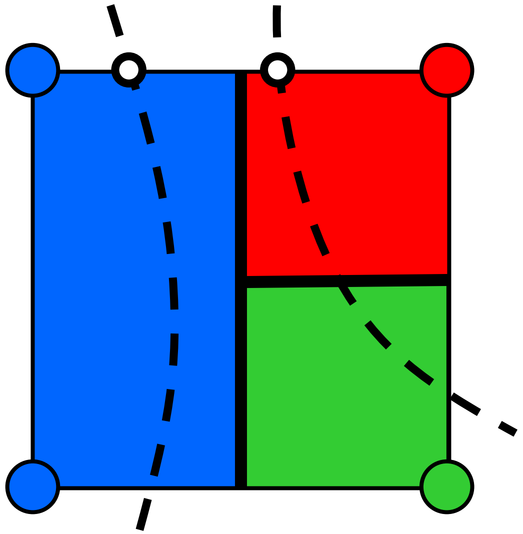


Anatomical structure
[Yuan 12]

DISCRETIZATION ON A GRID

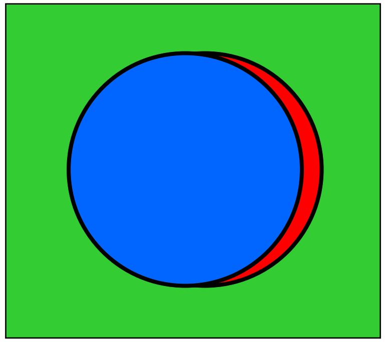
DISCRETIZATION ON A GRID

- Separating grid vertices of different labels
  - [Nielson 97; Hege 97; Ju 02; Wu 03; Bertram 05; Reitinger 05; Dillard 07; Shammaa 08,10; Zhang 07,12]
  - Geometric and topological artifacts near thin features

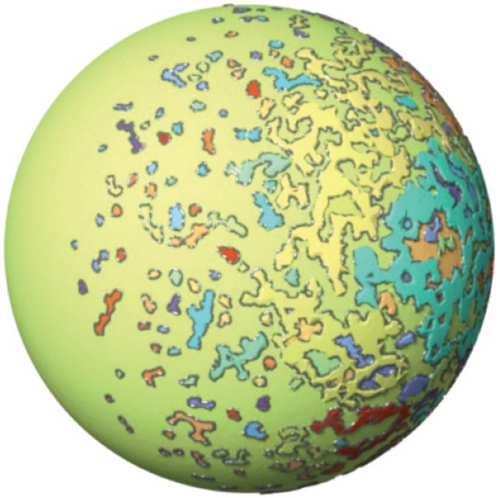


DISCRETIZATION ON A GRID

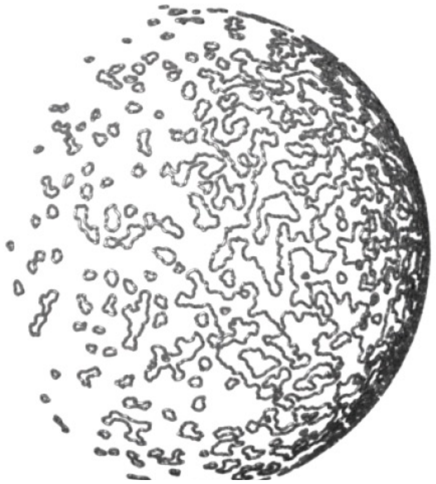
- Separating grid vertices of different labels
  - [Nielson 97; Hege 97; Ju 02; Wu 03; Bertram 05; Reitinger 05; Dillard 07; Shammaa 08,10; Zhang 07,12]
  - Geometric and topological artifacts near thin features



Material interface (cross-section)



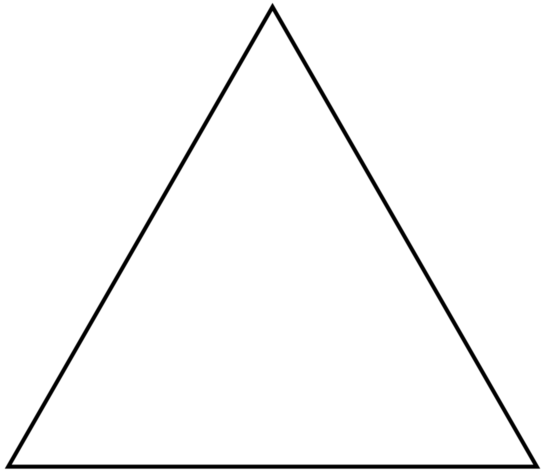
[Nielson 97]

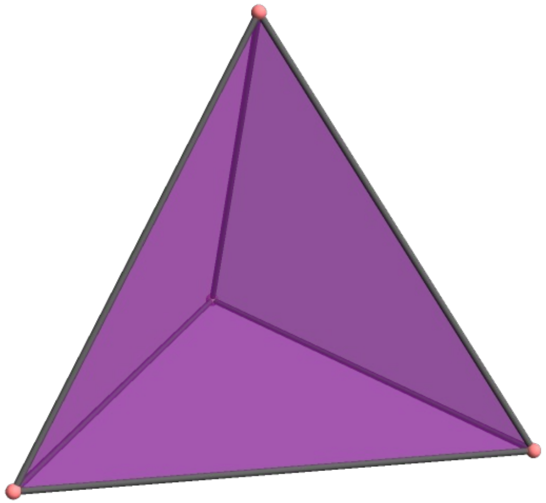


Non-manifold curves

DISCRETIZATION ON A GRID

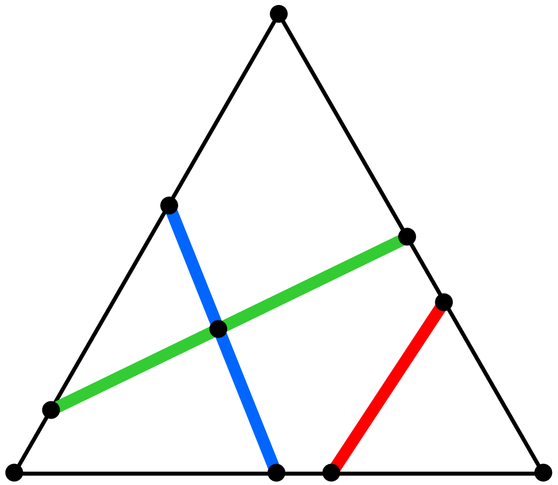
- Implicit surface network of a piecewise linear (PL) function
  - Implicit arrangement: [Kim 00; Bagley 16; Guo 21]
  - Material interface: [Bloomenthal 95; Bonnell 03; Saye 12,15]



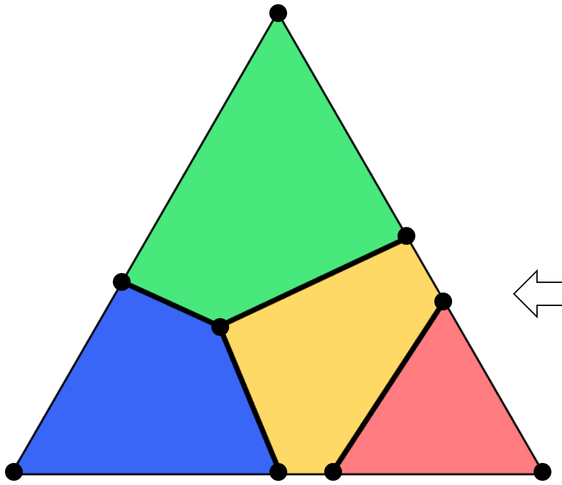


DISCRETIZATION ON A GRID

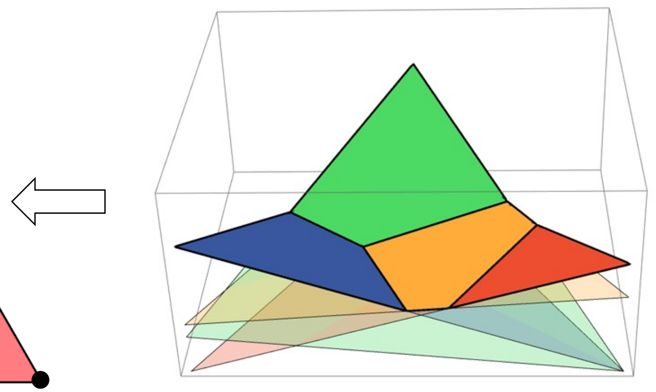
- Implicit surface network of a piecewise linear (PL) function
  - Implicit arrangement: [Kim 00; Bagley 16; Guo 21]
  - Material interface: [Bloomenthal 95; Bonnell 03; Saye 12,15]



Implicit arrangement

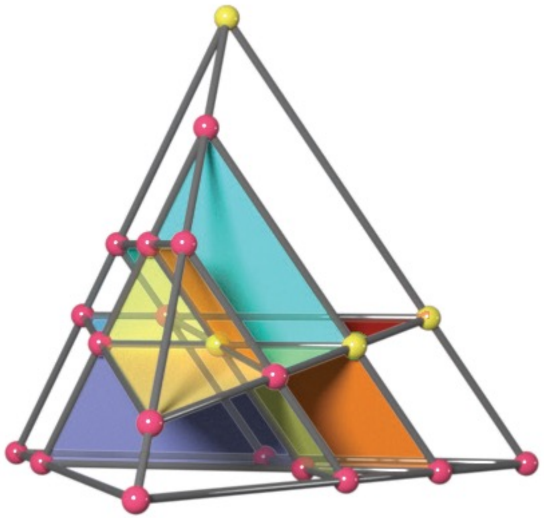


Material interface

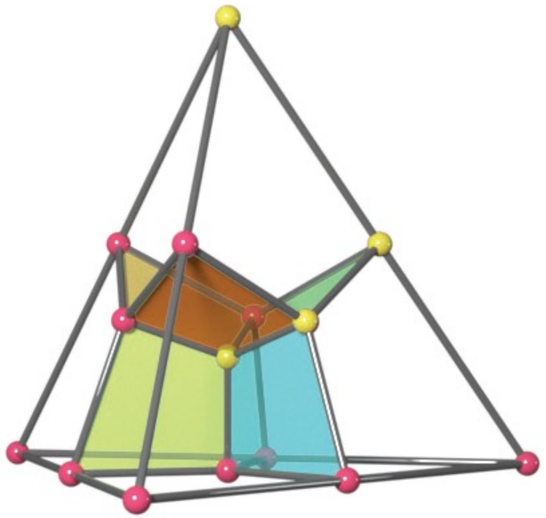


DISCRETIZATION ON A GRID

- Implicit surface network of a piecewise linear (PL) function
  - Implicit arrangement: [Kim 00; Bagley 16; Guo 21]
  - Material interface: [Bloomenthal 95; Bonnell 03; Saye 12,15]



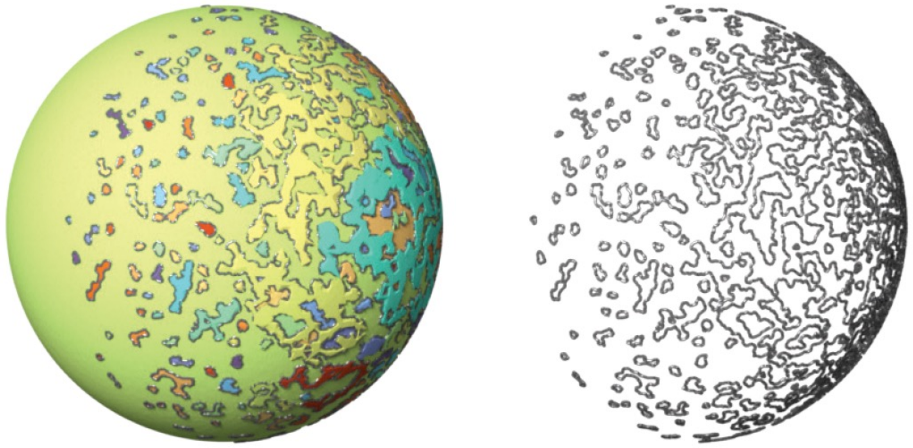
Implicit arrangement



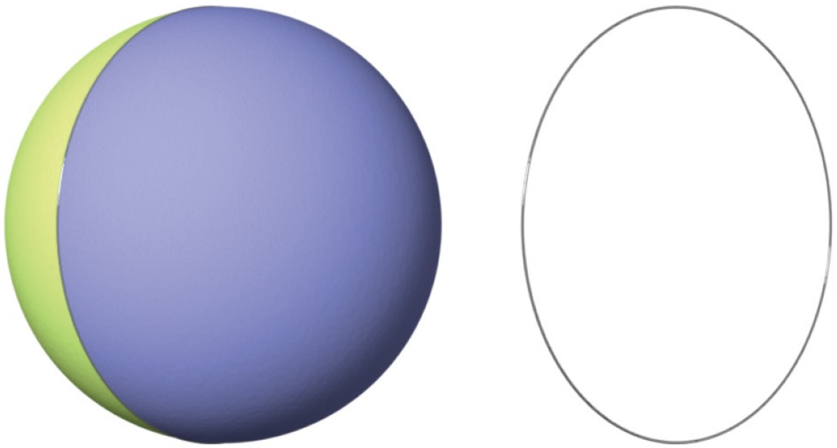
Material interface

DISCRETIZATION ON A GRID

- Implicit surface network of a piecewise linear (PL) function
  - Implicit arrangement: [Kim 00; Bagley 16; Guo 21]
  - Material interface: [Bloomenthal 95; Bonnell 03; Saye 12,15]



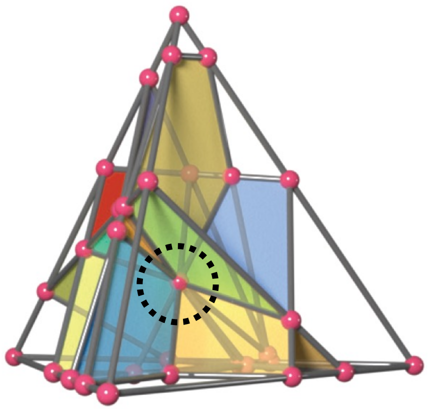
Label separation



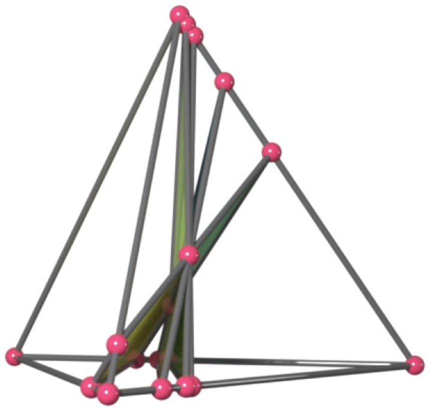
PL interpolation

DISCRETIZATION ON A GRID

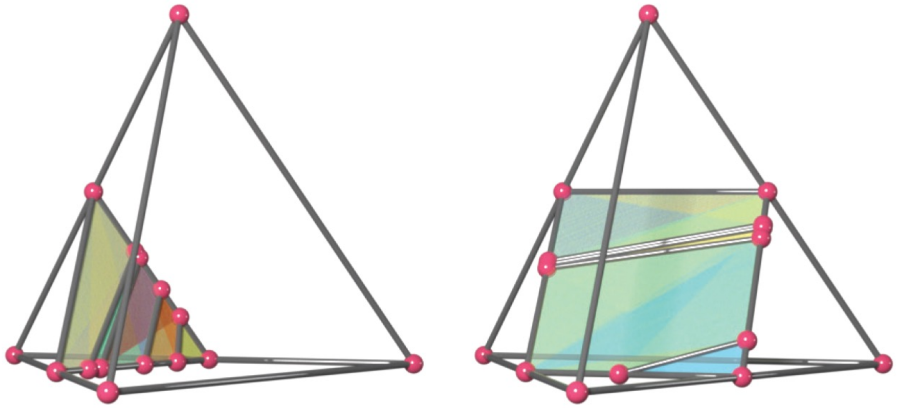
- Implicit surface network of a piecewise linear (PL) function
  - Implicit arrangement: [Kim 00; Bagley 16; Guo 21]
  - Material interface: [Bloomenthal 95; Bonnell 03; Saye 12,15]
  - Geometric intersections are prone to numerical errors



>3 planes at a point



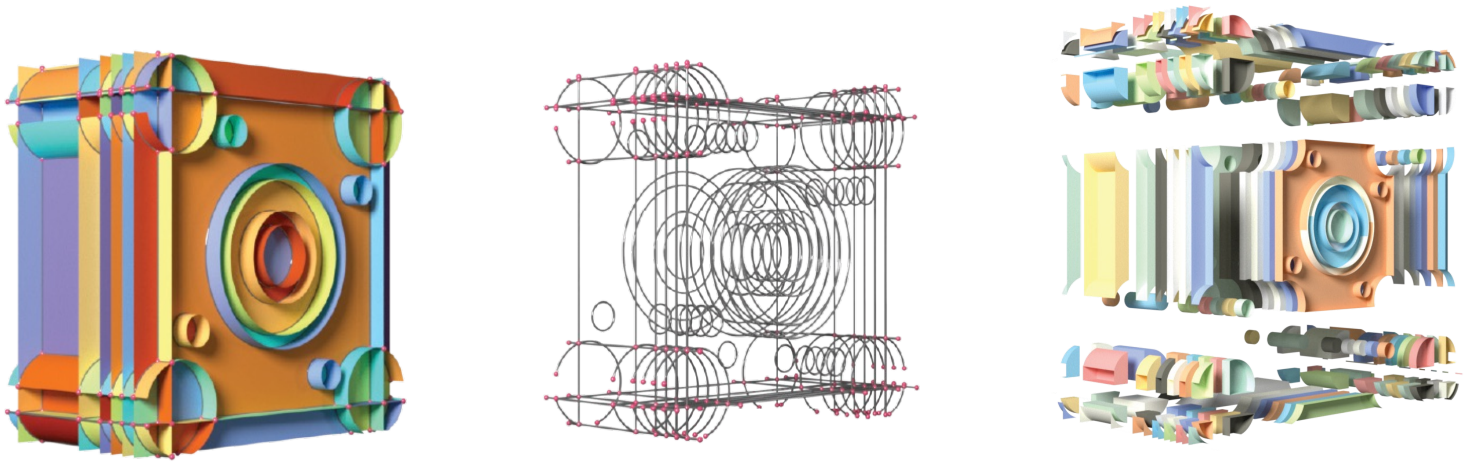
>2 planes at a line



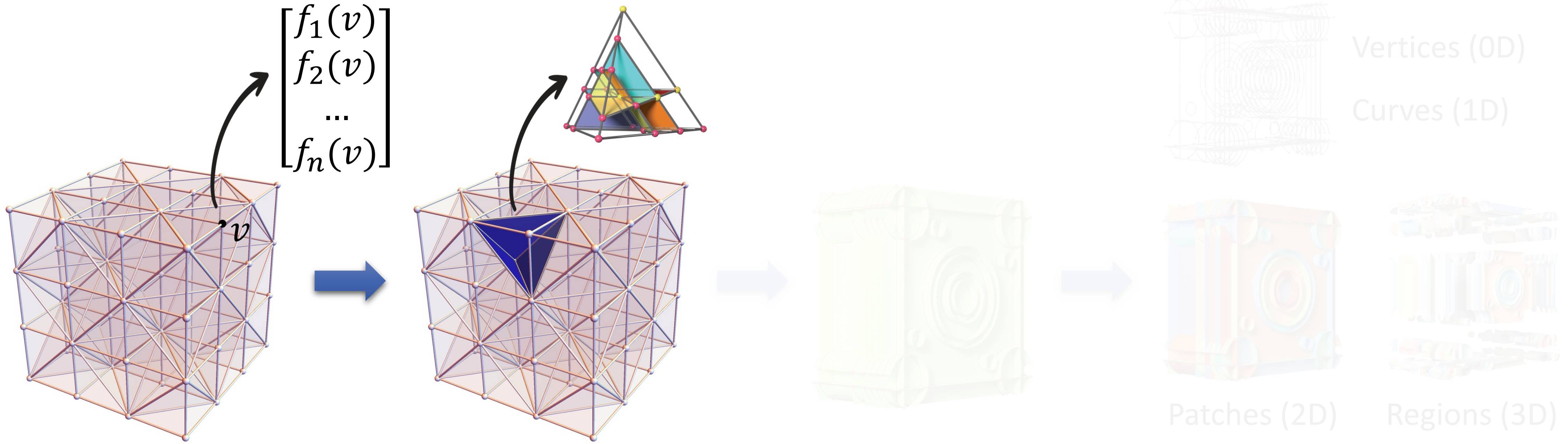
Almost co-planar

OUR METHOD

- Discretizing implicit surface networks using PL interpolation
  - Unified framework for both IA and MI
  - Guaranteed correct combinatorial structure
  - Scalable to complex inputs



PIPELINE



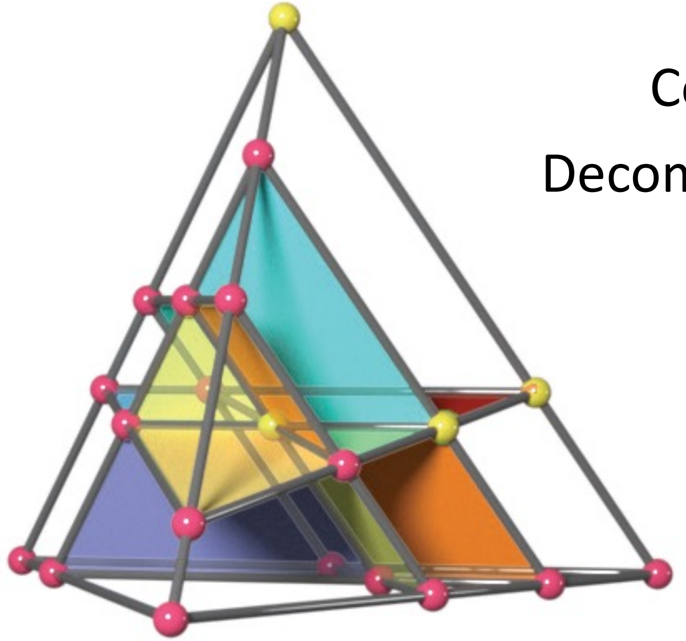
Input

Per-tet mesh

Complete mesh

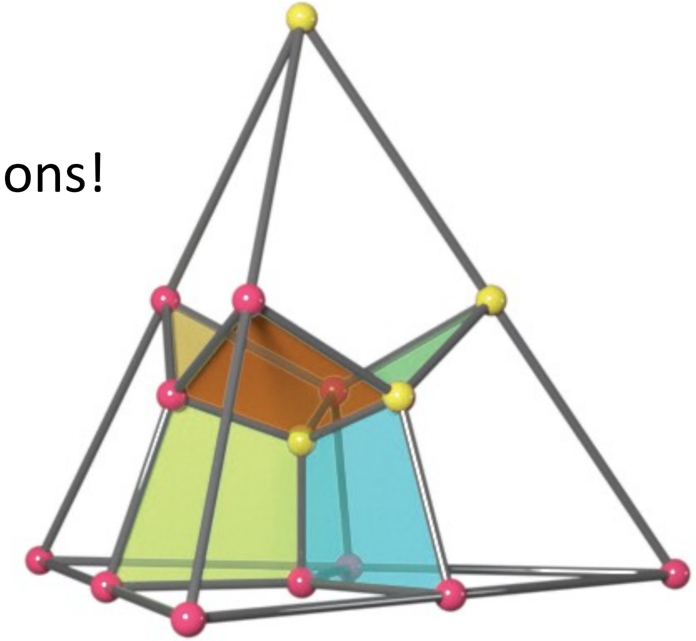
Space partitioning

TETRAHEDRON PROCESSING



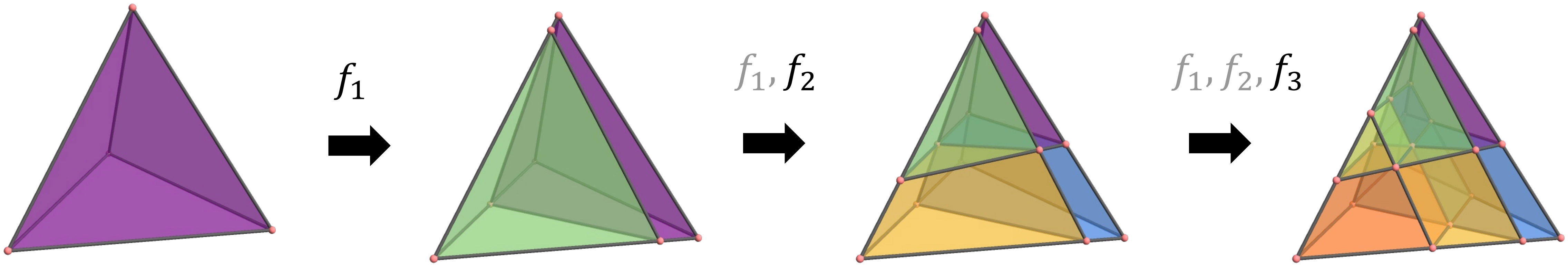
Implicit arrangement (IA)

Convex
Decompositions!



Material interface (MI)

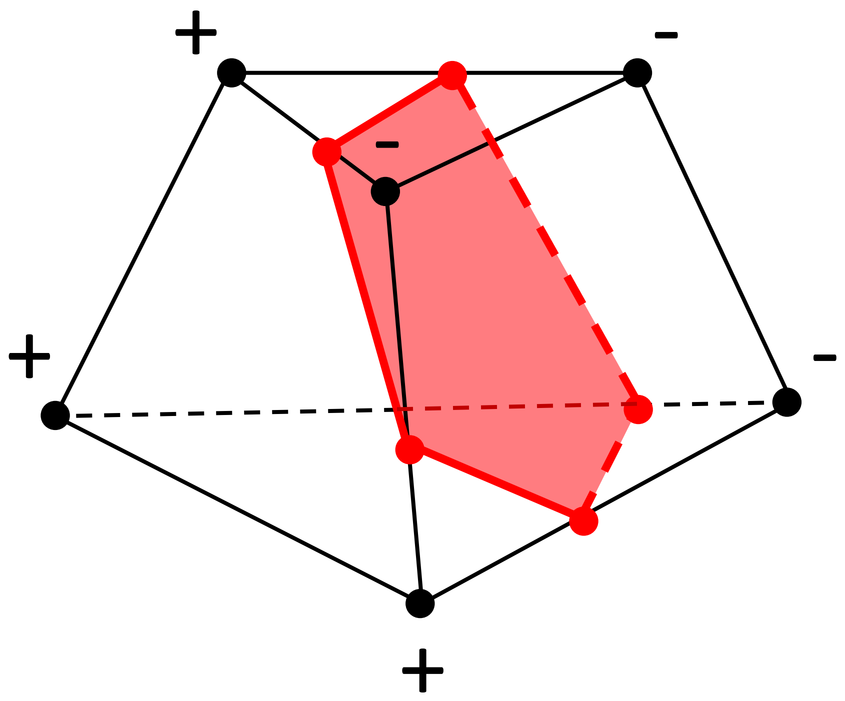
INCREMENTAL CONSTRUCTION



Implicit arrangement (IA)

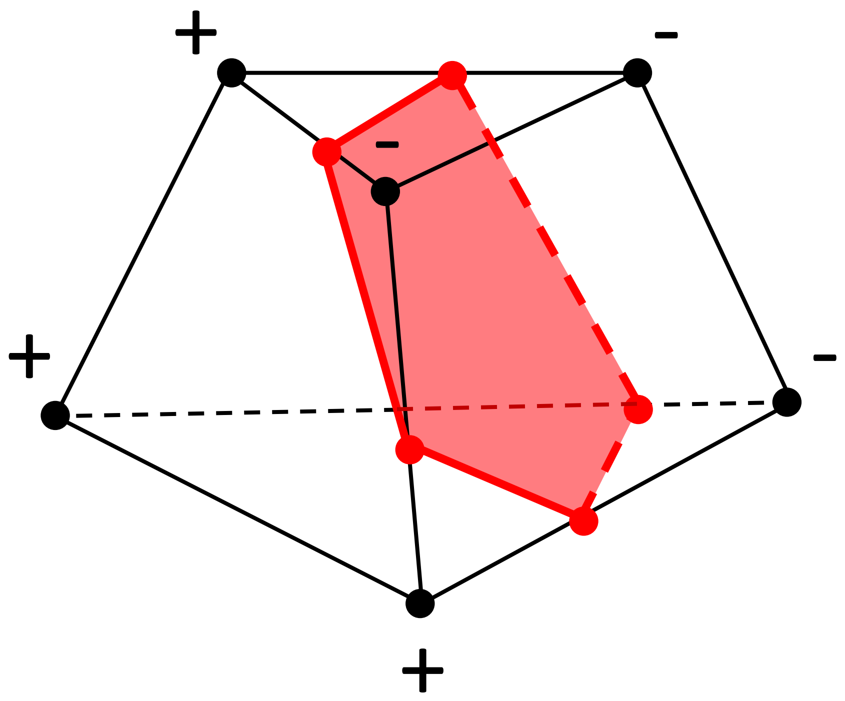
CELL SPLITTING

- Given a function f_i:
  - Compute sign of f_i (for IA) or $f_i - f_j$ (for MI) at vertices
  - Split edges by cut vertices
  - Split faces by cut edges
  - Split cells by cut faces



INCREMENTAL CONSTRUCTION

- Given a function f_i:
  - Compute sign of f_i (for IA) or $f_i - f_j$ (for MI) at vertices
  - Split edges by cut vertices
  - Split faces by cut edges
  - Split cells by cut faces

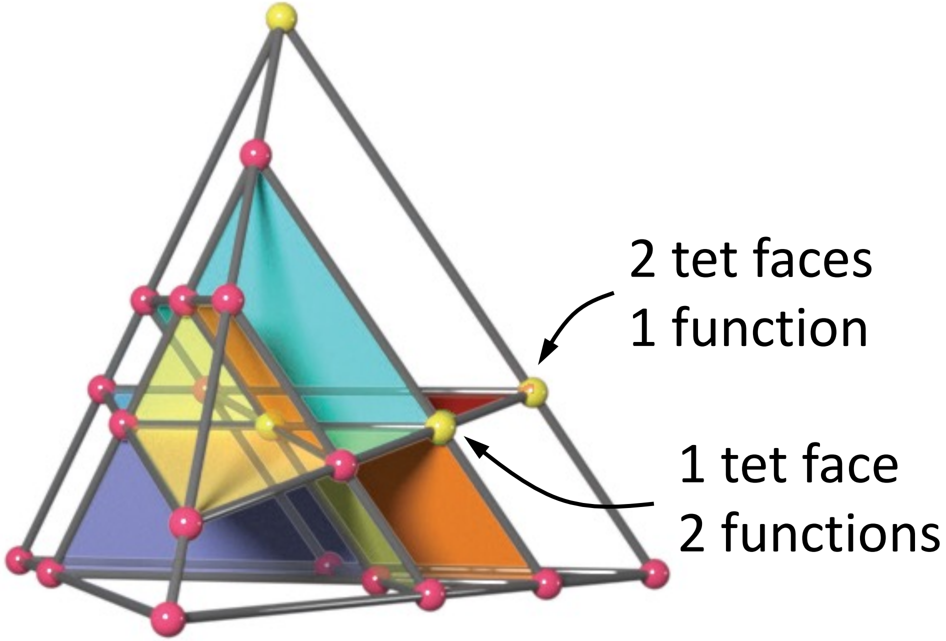


ROBUST SIGNING

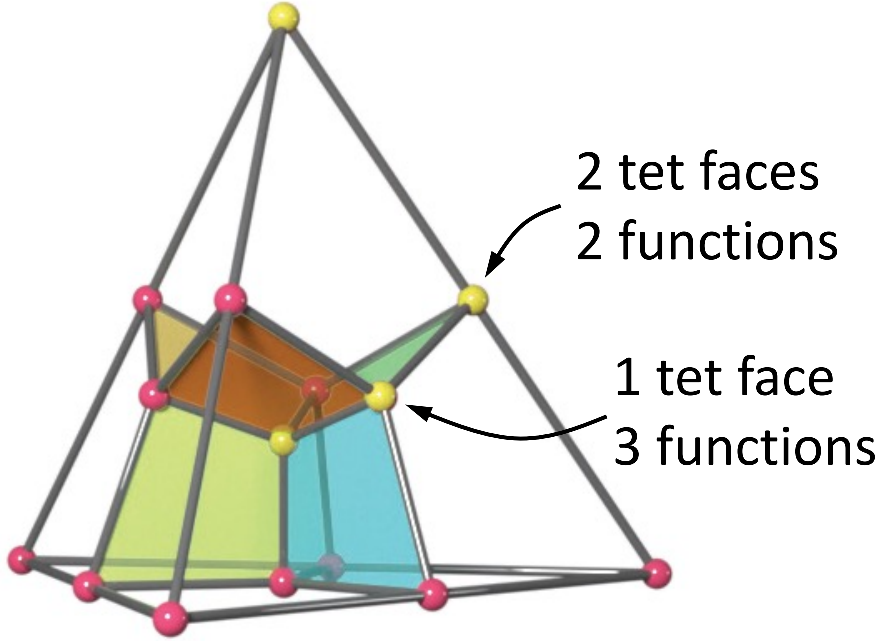
- Exact intersection and predicates
  - Requires rational representations of coordinates
- Our approach: coordinate-free exact signing
  - Encodes vertices using functions and tet faces
  - Simple predicate using only function values at tet vertices
  - Inspired by plane-based representations [Sugihara 89; Bernstein 09; Campen 10; Attene 20; Cherchi 20; Nehring-Wirxel 21; Diazzi 21]

VERTEX ENCODING

SIGGRAPH 2022
VANCOUVER+ 8-11 AUG

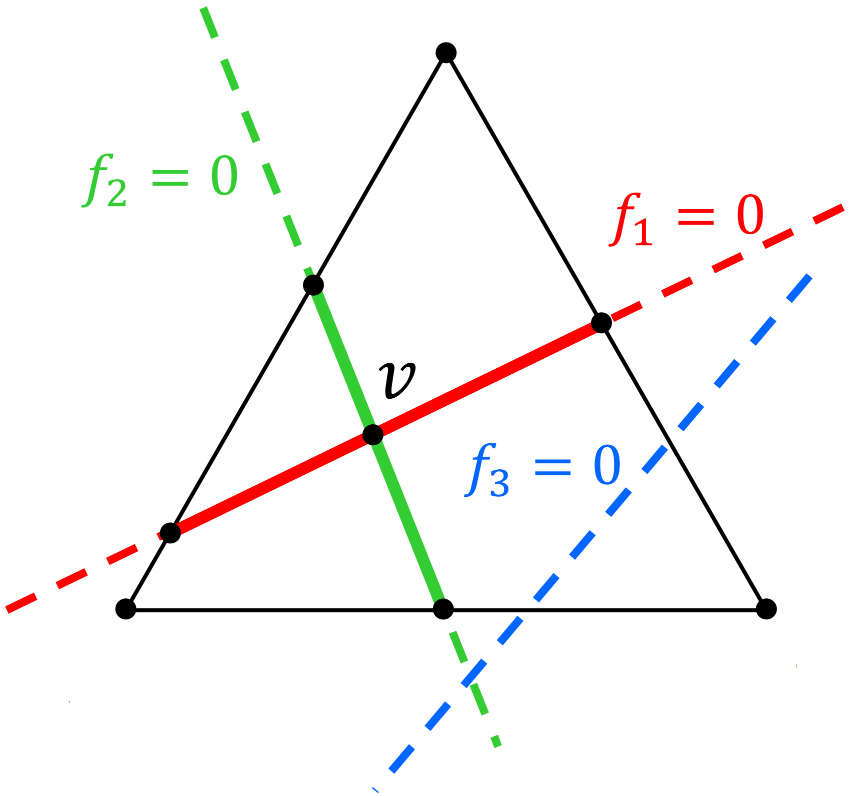


Implicit arrangement: **3** indices



Material interface: **4** indices

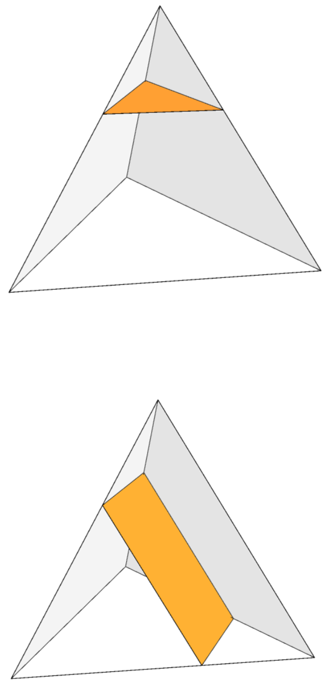
BARYCENTRIC PREDICATE



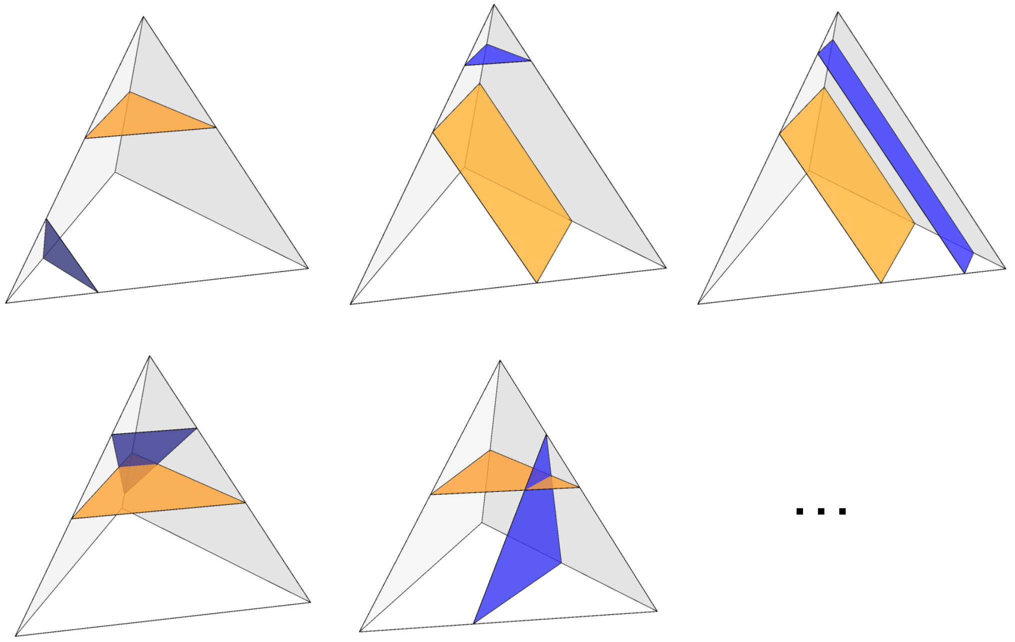
$$sign(f_3(v)) =$$

$$sign\left(det\begin{bmatrix} f_1^1 & f_1^2 & f_1^3 \\ f_2^1 & f_2^2 & f_2^3 \\ f_3^1 & f_3^2 & f_3^3 \end{bmatrix}\right) * sign\left(det\begin{bmatrix} f_1^1 & f_1^2 & f_1^3 \\ f_2^1 & f_2^2 & f_2^3 \\ 1 & 1 & 1 \end{bmatrix}\right)$$

LOOK-UP TABLES



1 function (MI)



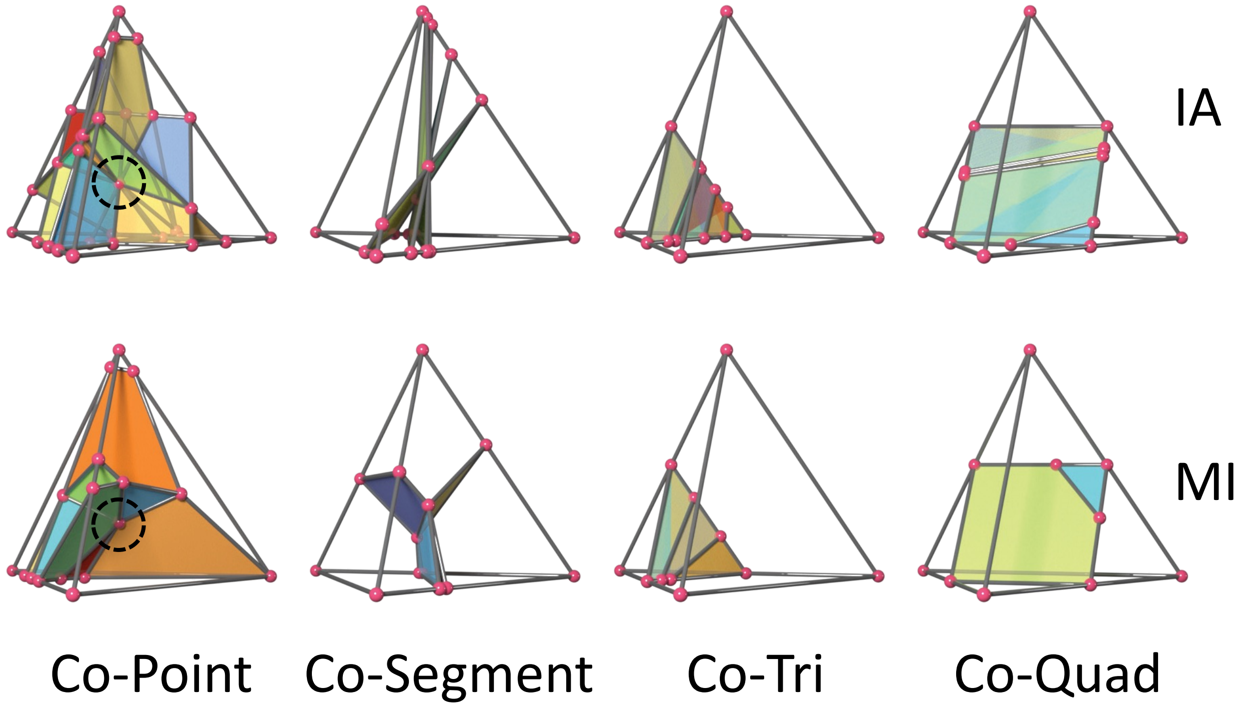
2 functions (MI)

RESULTS

RESULTS: ROBUSTNESS

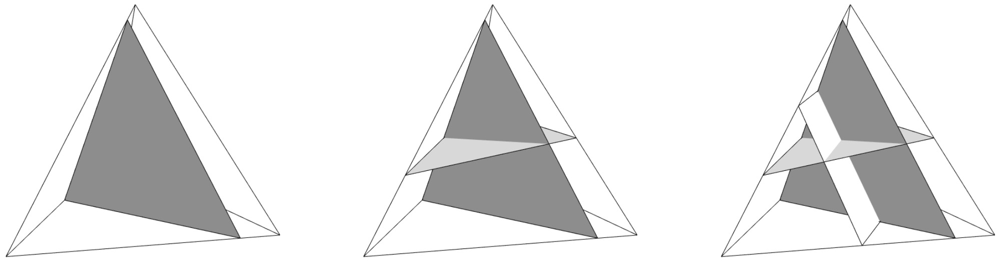
SIGGRAPH 2022 VANCOUVER+ 8-11 AUG

- Near-degeneracy test
  - 4 (IA) or 5 (MI) functions in a tetrahedron
  - 10 000 instances per degeneracy type
- 100% consistency
  - Same combinatorial structure after changing the order of functions



RESULTS: EFFICIENCY

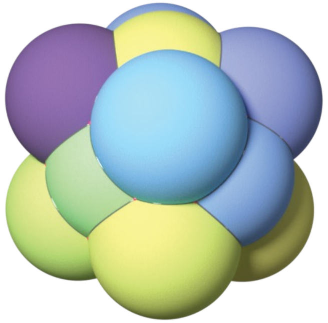
- Comparison with exact coordinate representations [CGAL]
  - IA of 1, 2, and 3 functions in a tetrahedron
  - 100 instances of each type
- Our method is ~100 faster



CGAL:	666.3	1754.8	3302.8
Ours:	6.6	13.7	26.3

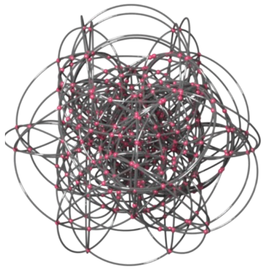
(average time in μs)

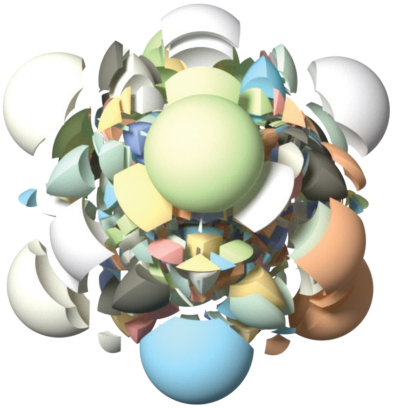
RESULTS: EFFICIENCY

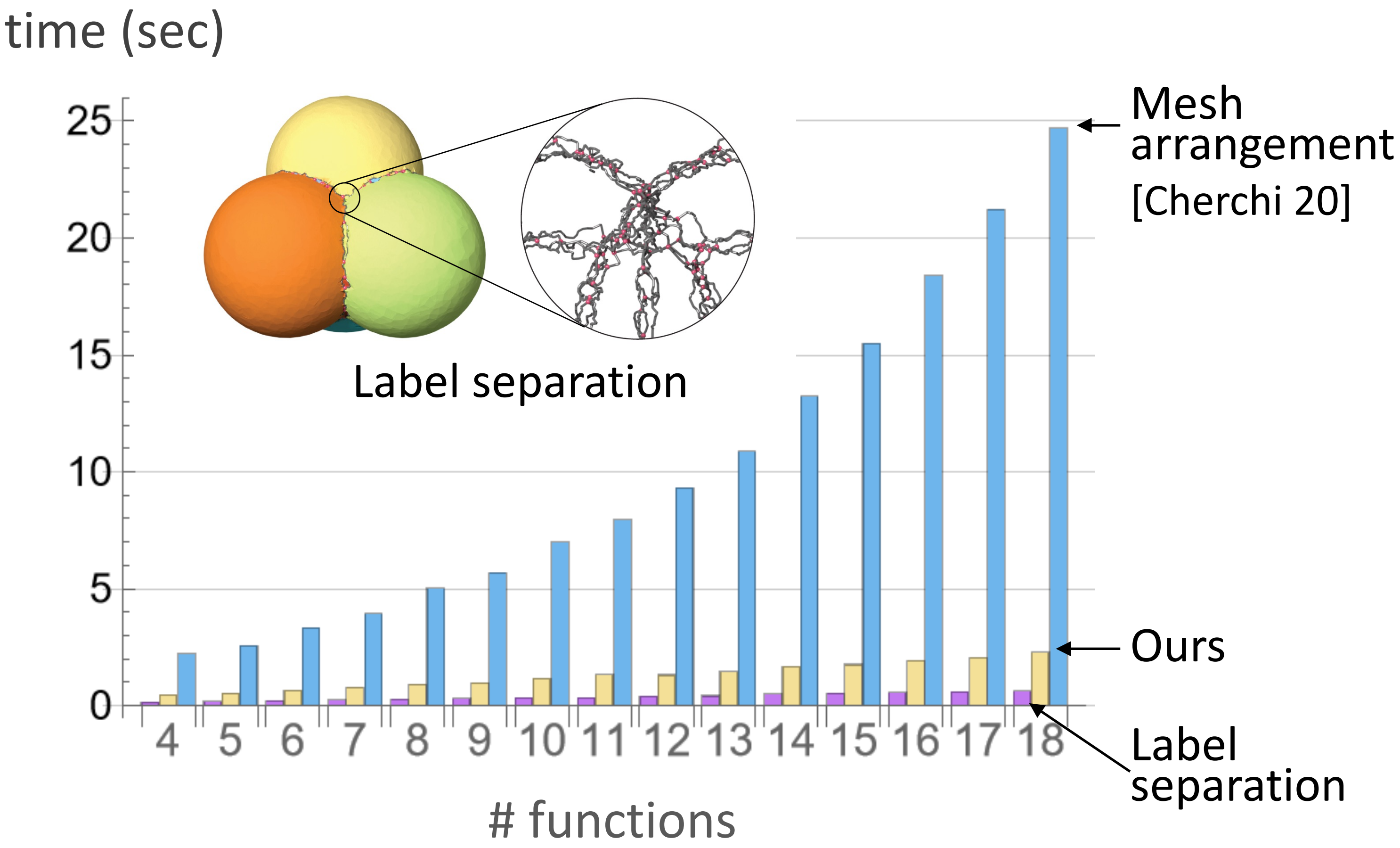


IA of spherical distance functions

(10^6 tet vertices)



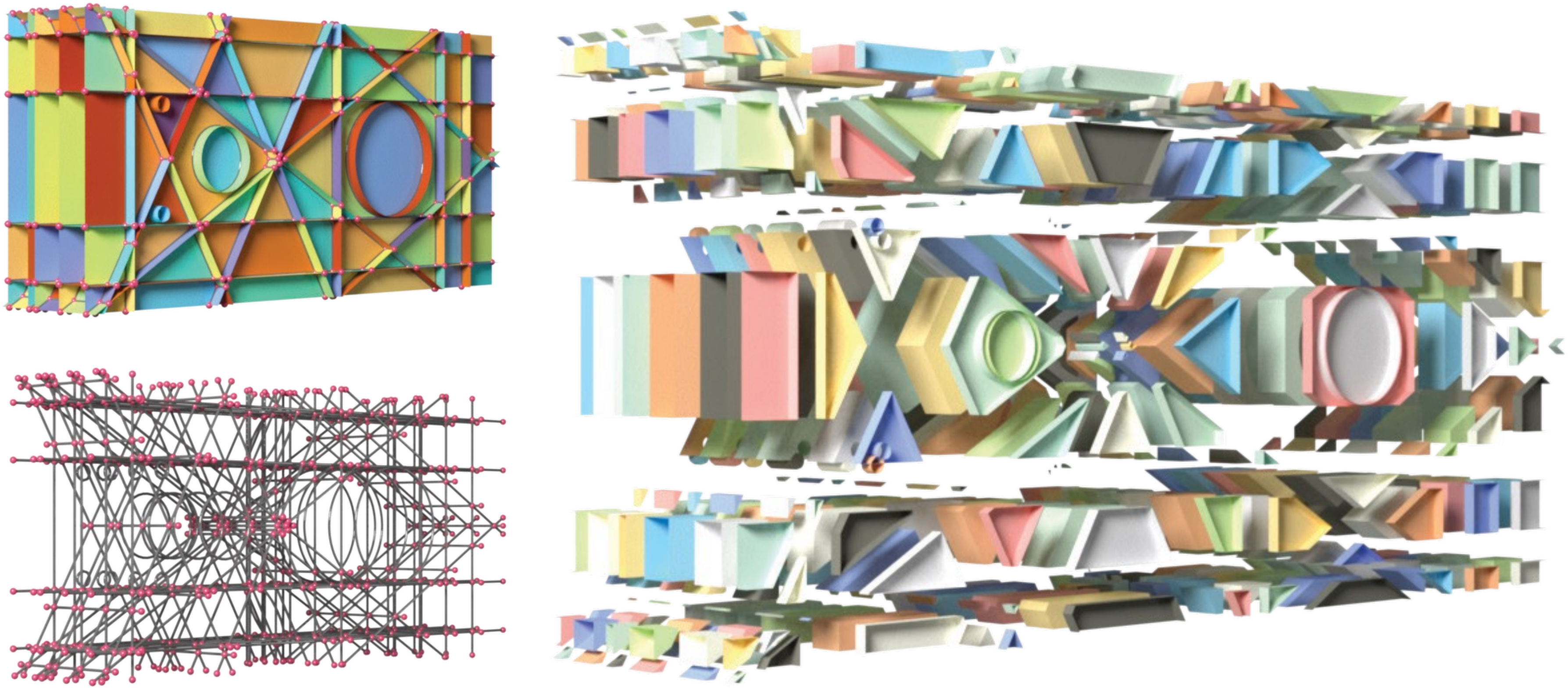




EXAMPLE: CSG

Arrangement of 26 CSG primitives

(10^6 tet vertices)

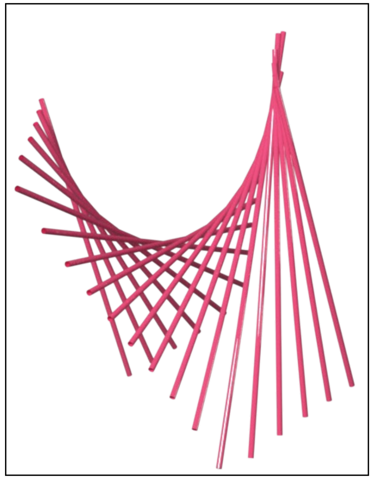


1543 patches, 590 cells

Time: 3.25 sec (ours), 153.4 sec (mesh arr.)

EXAMPLE: VORONOI DIAGRAM

Voronoi diagram of
20 rotating lines
(10^6 tet vertices)



345 patches, 20 cells
Time: 3.54 sec

CONCLUSION

- Robust and efficient algorithms for discretizing implicit arrangement and material interfaces
- Future directions:
  - Parallelization
  - Improving mesh quality
  - Grid generation

Code and data