A GENERAL ADVANCING FRONT TECHNIQUE FOR FILLING SPACE WITH ARBITRARY OBJECTS

Rainald Löhner[1] and Eugenio Oñate[2]

[1]GMU/SCS, George Mason University, Fairfax, VA
[2]CIMNE, UPC, Barcelona, Spain

REF: IJNME 61, 1977-1991 (2004)

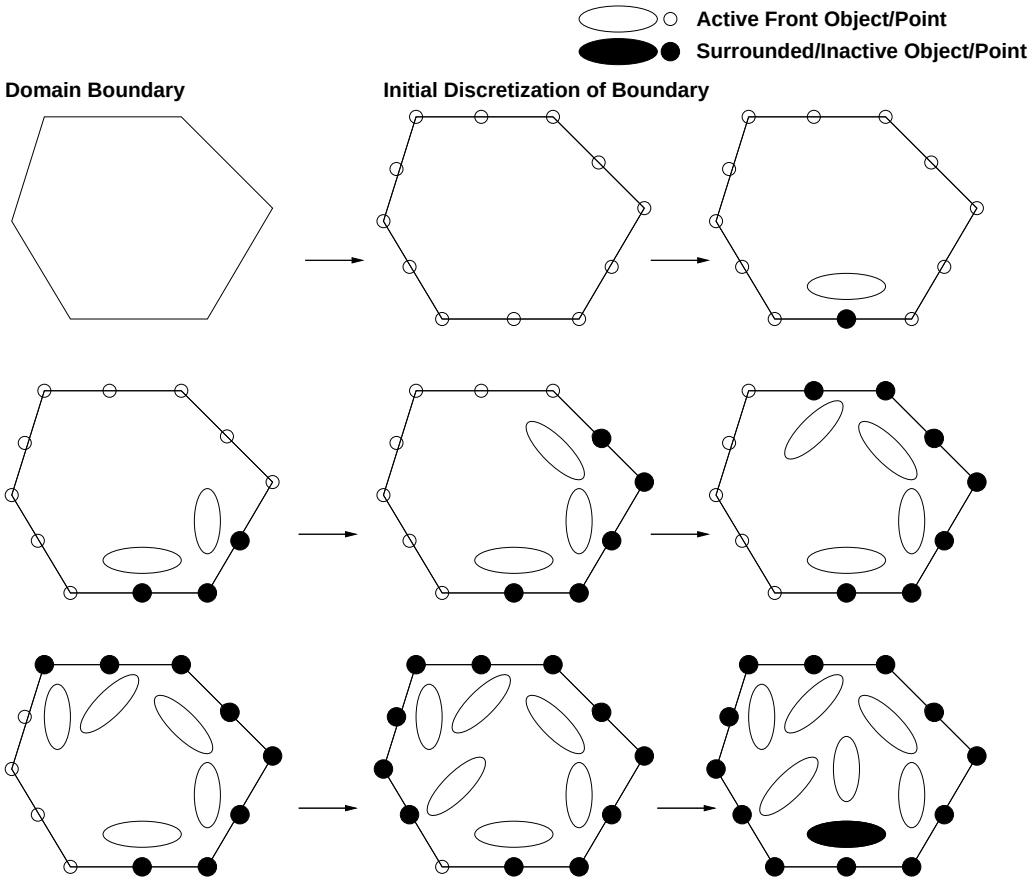
INTRODUCTION

Motivation:

- Perceived Difficulty of Generating 3-D Grids
- Desire for Optimal High-Order Schemes
  $\Rightarrow$ Large Effort in 'Gridless' or
  'Mesh Free' Schemes
- Discrete Element/Particle Methods (DEM/DPM)
  - Arbitrary Objects
  - Interesting for Treating Discontinua

But: Need to Generate Points/Arbitrary Objects

ALGORITHM



Advancing Front Space-Filling With Ellipses

ALGORITHM (1)

Assume Given:

- Desired Mean Distance Between Points δ_0
    - Background Grids
    - Sources
    - Attached to CAD-Data
- Surface Triangulation

ALGORITHM (2)

Then:

- Determine δ_0 for Points of Triangulation;
- `while`: Active Objects Left:
  - Remove Object `ioout` With Smallest δ_0;
  - Determine $\mathbf{x}$ for `nposs` Possible New Neighbours (Stencil);
  - Find Existing Objects Close to `ioout`;
  - Find Boundary Faces Close to `ioout`;
  - `do`: For each `ionew`:
    - If $|\mathbf{x}_{new} - \mathbf{x}_{old}| < \delta_{min}$: $\Rightarrow$ skip `ionew`;
    - If Line `ioout` - `ionew` Crosses Existing Faces: $\Rightarrow$ skip `ionew`;
    - Determine δ_0 for `ionew`;
    - `npoin=npoin+1`
    - Introduce `ionew` to the List of Coordinates;
    - Introduce `ionew` to the List of Active Front;

    `enddo`
- `endwhile`

DATA STRUCTURES

Main Search Operations:

- Finding Active Object With Smallest δ_0

  $\Rightarrow$ Heap-List

- Finding Objects Close to `ioout`

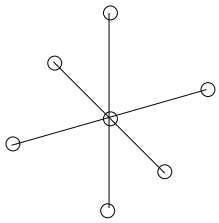
  $\Rightarrow$ Octree

- Finding Boundary Faces Close to `ioout`

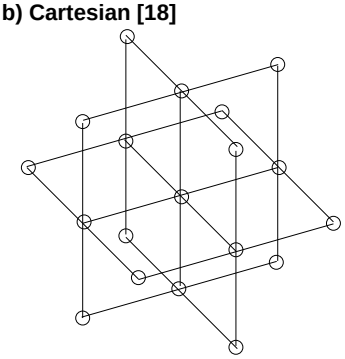
  $\Rightarrow$ Octree + Linked List

POINT STENCILS

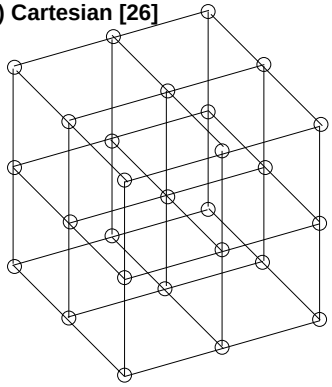
a) Cartesian [6]



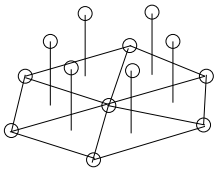
b) Cartesian [18]



c) Cartesian [26]

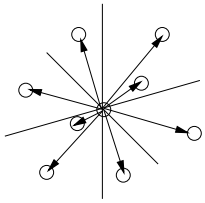


d) Tetrahedral [17]



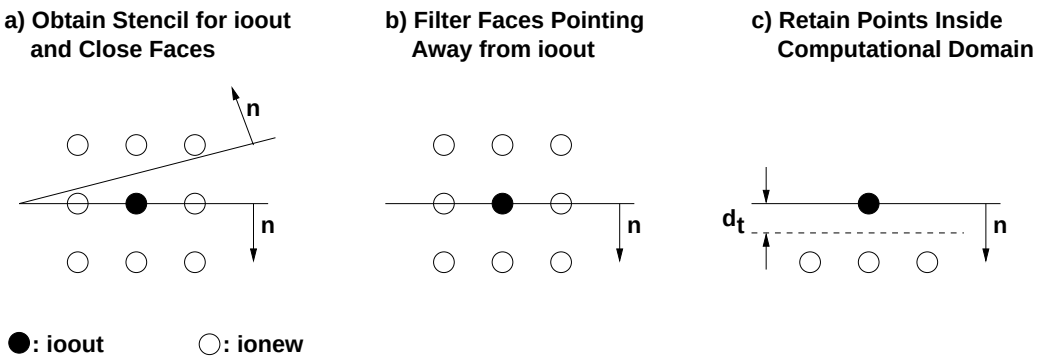
Symmetric

e) Random



Point Stencils

BOUNDARY CONSISTENCY CHECKS

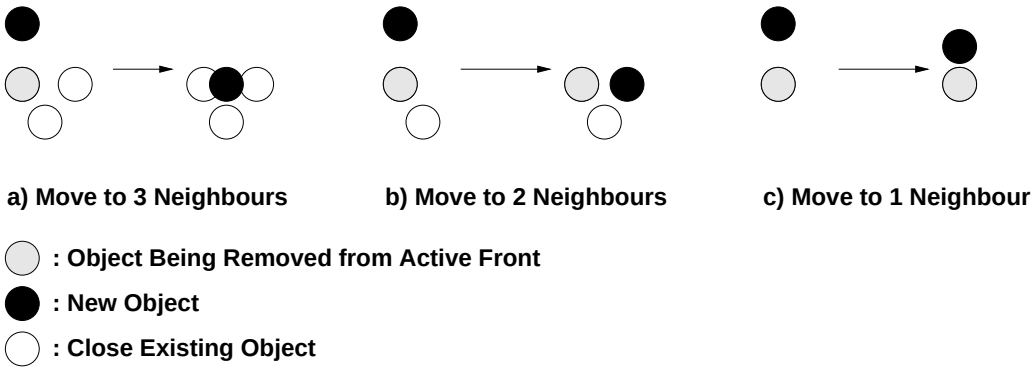


Boundary Consistecy Checks

MAXIMUM COMPACTION TECHNIQUES (1)

Usual Placement Leads to Fill-Ratios That Are Too Low
⇒ Compact Further

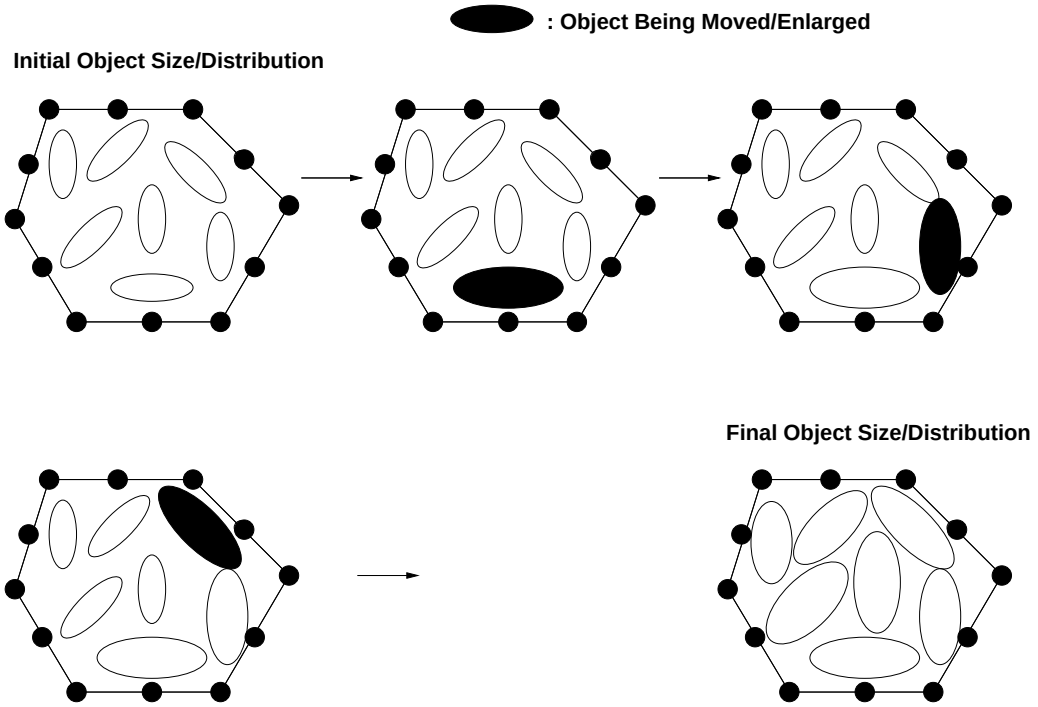
Option 1: Move to Existing Object



Closest Object Placement

MAXIMUM COMPACTION TECHNIQUES (2)

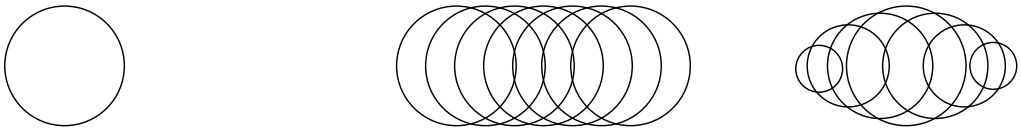
Option 2: Move + Enlarge Existing Objects



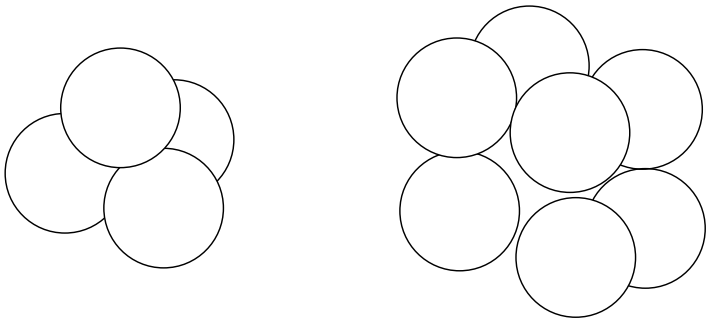
Movement and Enlargement of Objects

ARBITRARY OBJECTS

a) Sphere **b) Tube** **c) Ellipsoid**



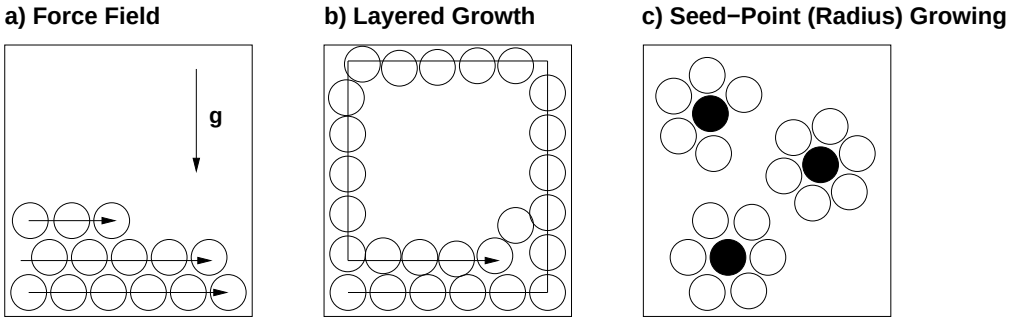
d) Tetrahedron **e) Cube** **etc.**



Arbitrary Objects As A Collection of Spheres

DEPOSITION PATTERNS

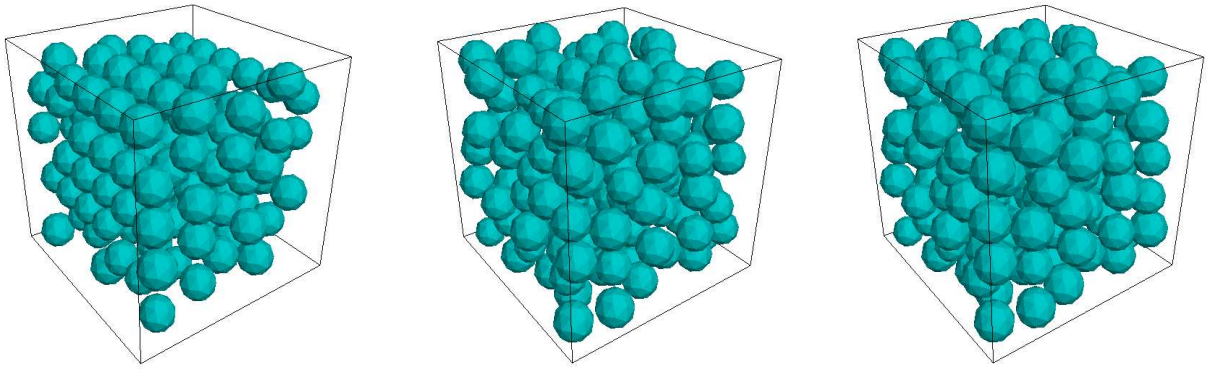
Different Deposition Required Depending on Application



Deposition Patterns

CUBE

- Unit Cube
- $\delta = 0.15 + / - 0.015$
- $n_s = 133, 169, 169$ $(v = 0.291, 0.410, 0.448)$

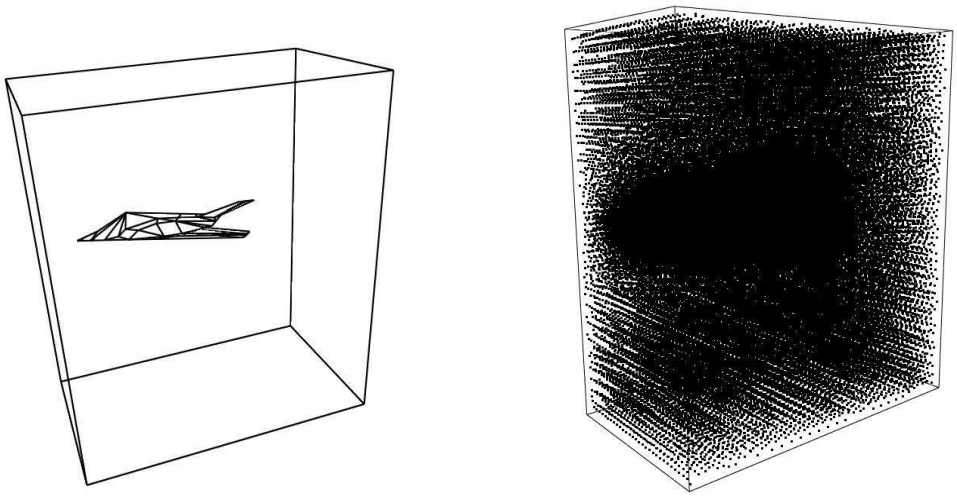


Cube Filled With Spheres

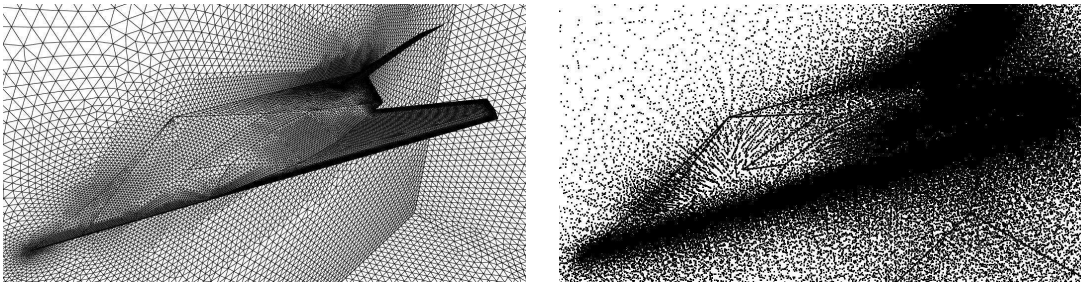
- CPU/100K Spheres: 33 sec, 120 sec

F-117 (1)

- `npoin=250K, ntria=50K`
- CPU: 44 sec (Simple Stencil Placement)

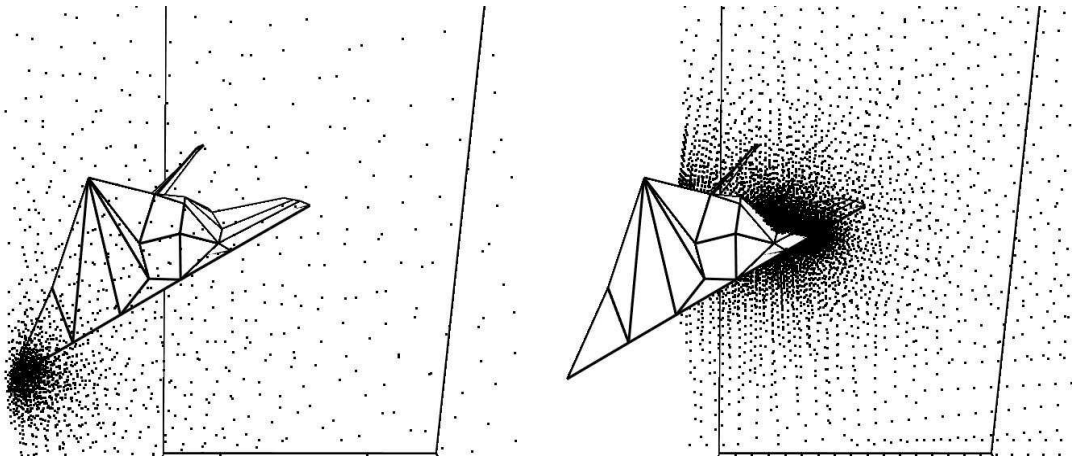


CAD Definition and Global Cloud of Points

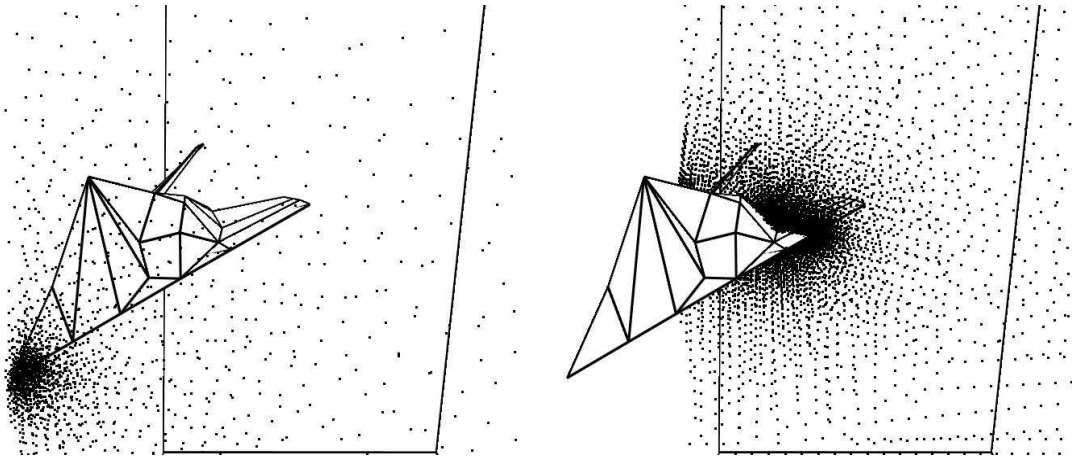


Close-up of Surface Mesh and Cloud of Points

F-117 (2)



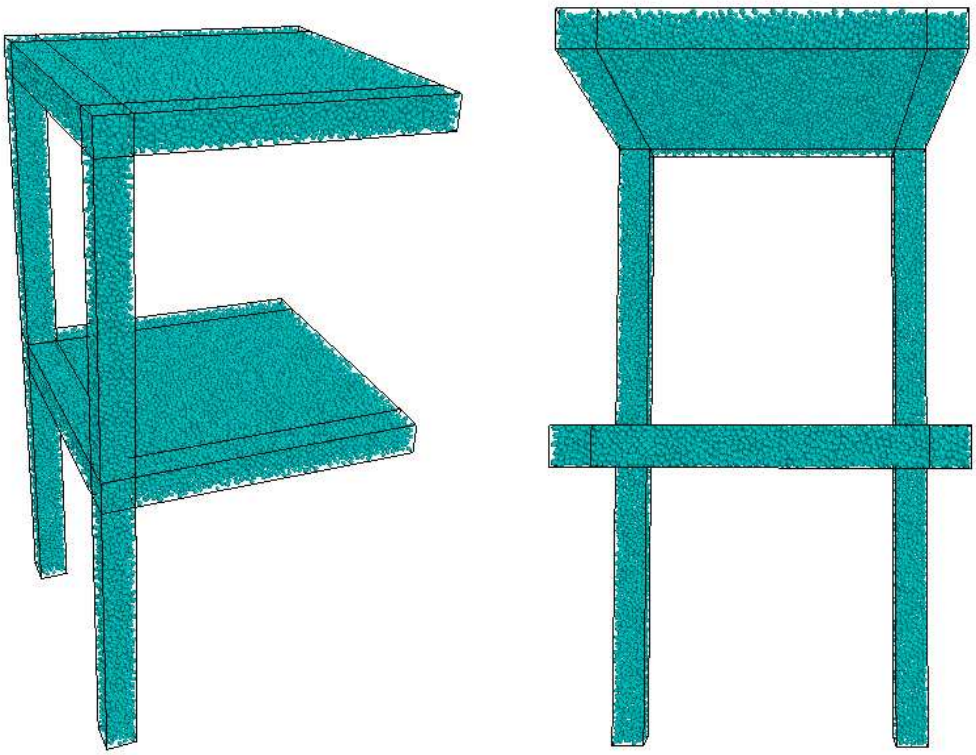
Cuts at $x = 0, 120$



Cut at $x = 120$ (Detail) and $x = 190$

CIVIL ENGINEERING STRUCTURE

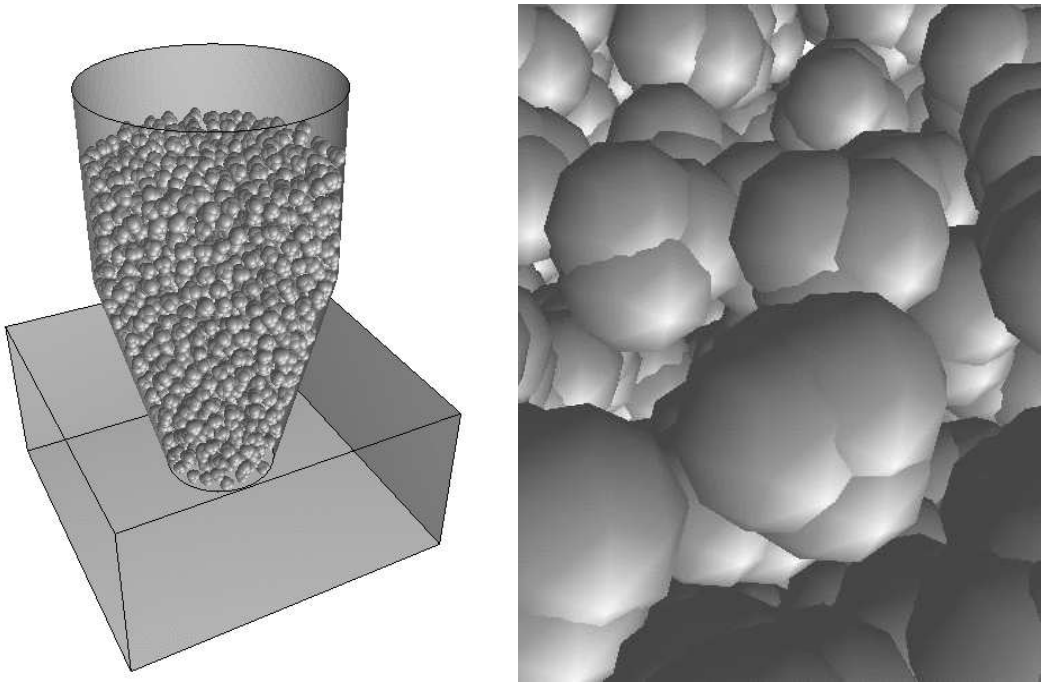
- 41,883 Spheres
- $v = 0.495$, $n_c = 5.71$
- CPU: 127 sec



DEM Discretization of Civil Engineering Structure

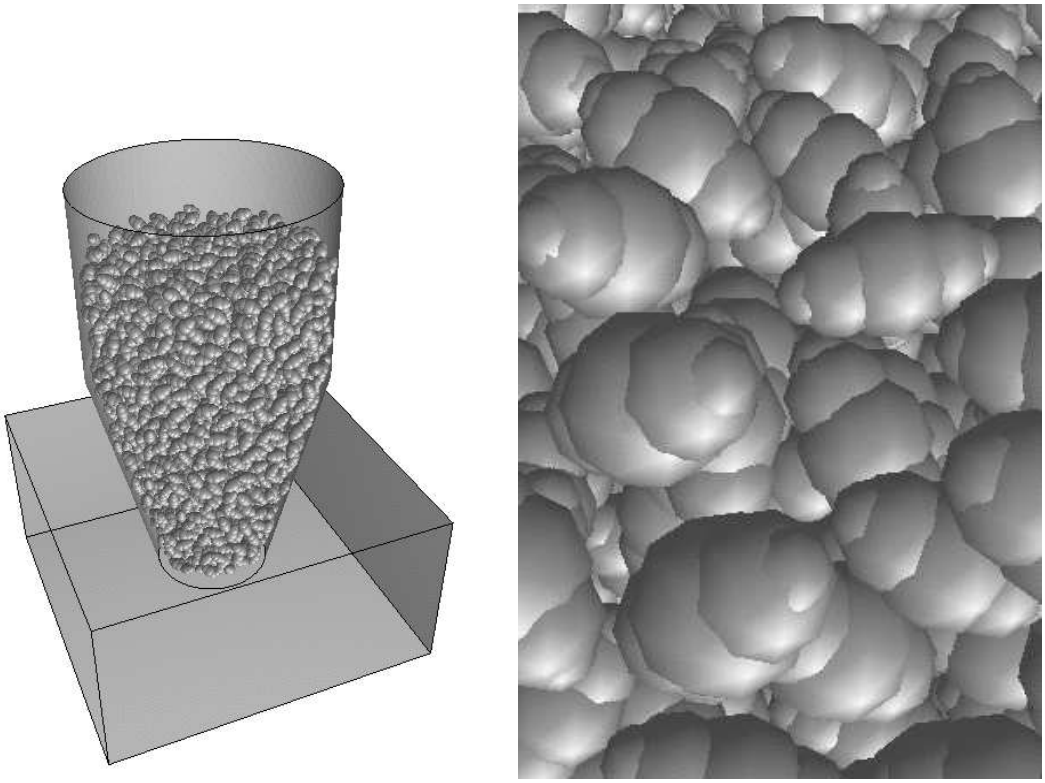
HOPPER (1)

O(2K) Objects, CPU: O(8 sec)



Hopper Filled With Beans

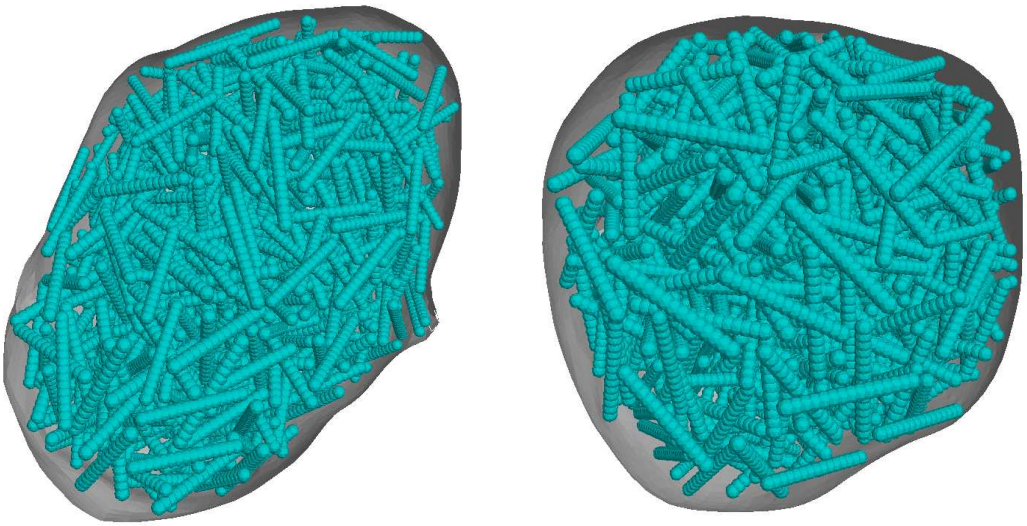
HOPPER (2)



Hopper Filled With Ellipsoids

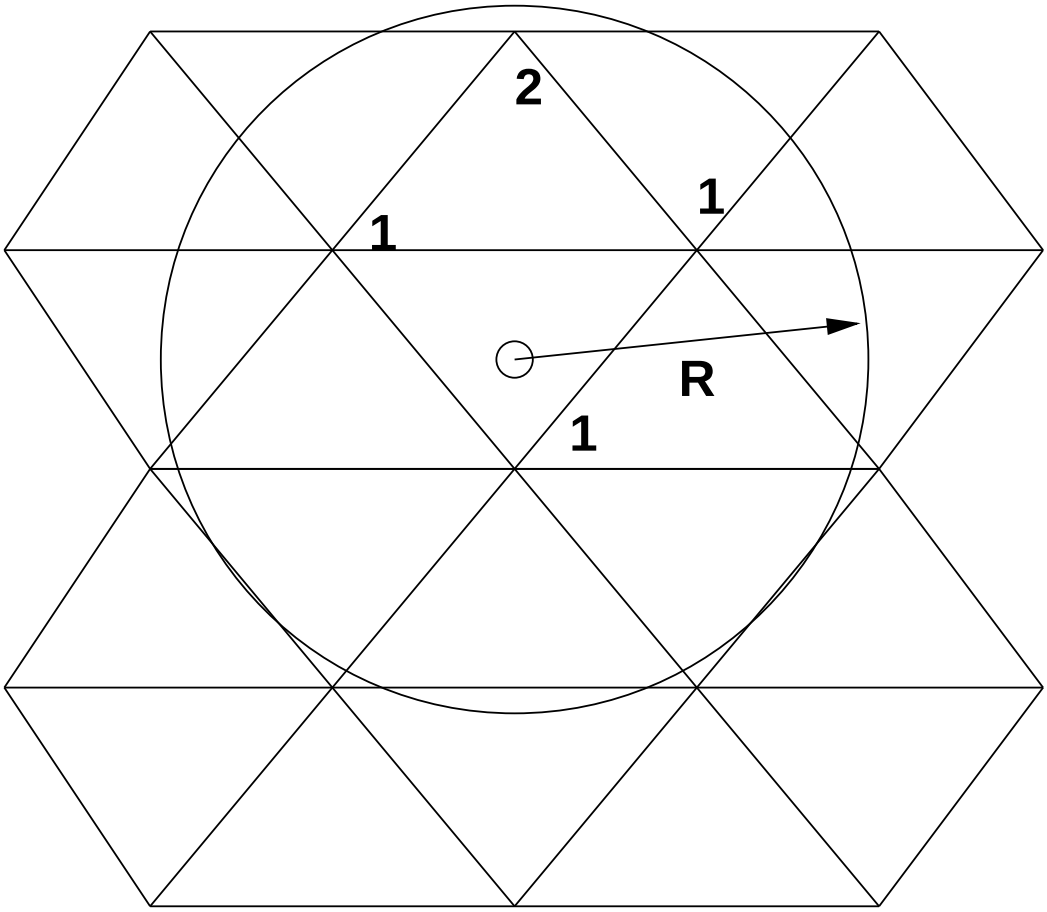
ANEURYSM WITH COILS

- $\Omega : 2 \times 2 \times 1.5cm$
- 750 Coils, 11,250 Spheres, 15 Spheres/Coil
- $v = 0.15$
- CPU: 8 sec



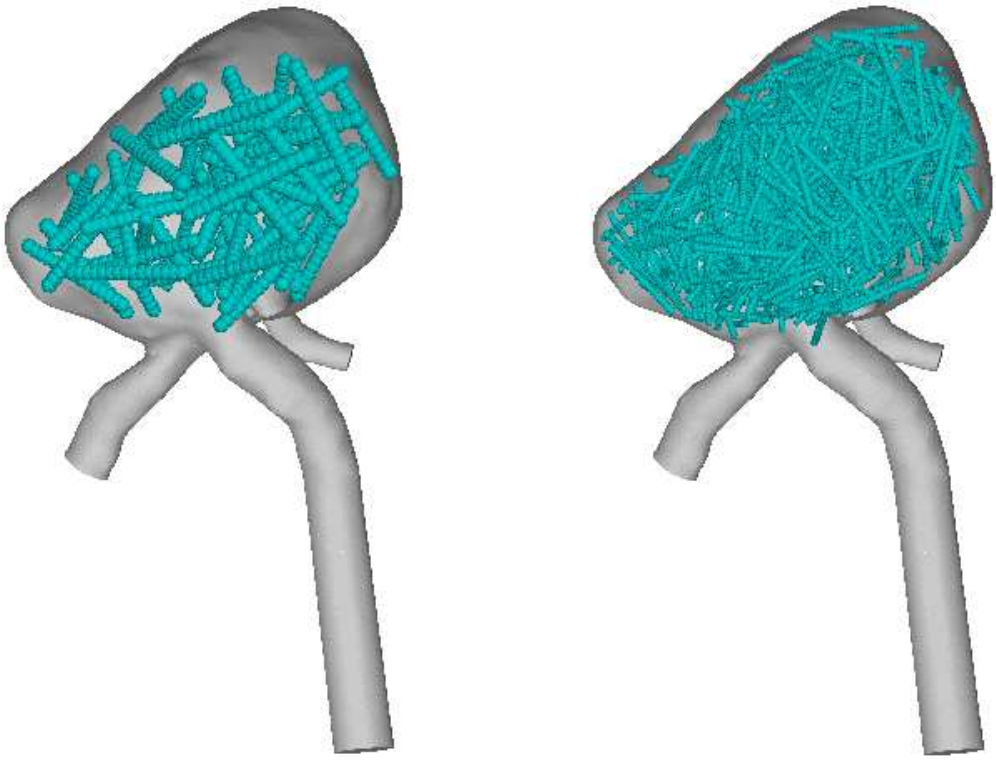
Aneurysm Filled With Coils

LINK TO DPM



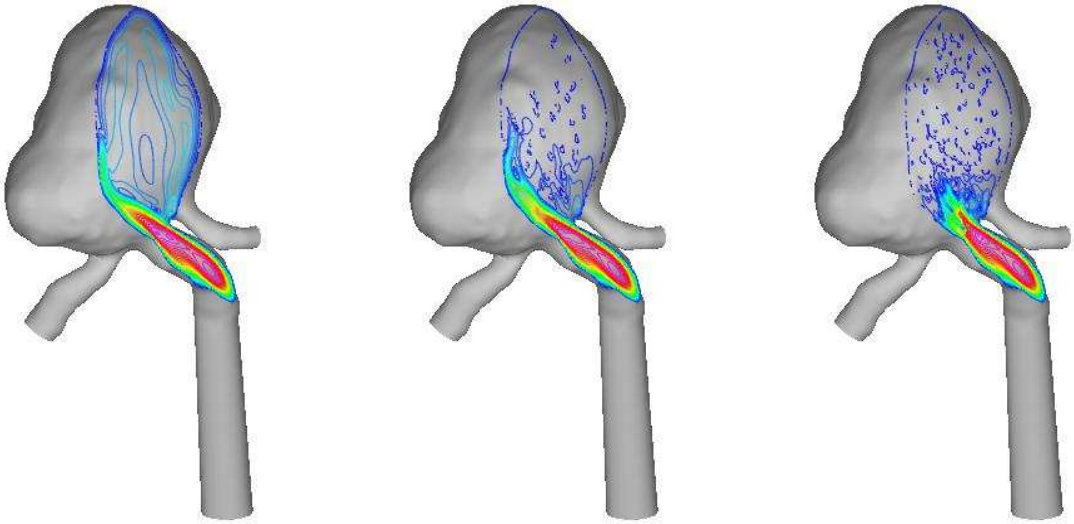
Link to Discrete Particle Method

ANEURYSM WITH COILS (1)



Medium and Fine Coil Discretization

ANEURYSM WITH COILS (2)



Velocity in Plane (No/Med/Fine Coils)