Current Status of Tool Development for Cut-cell-based Mesh Generation

Cut-Cell 기반 격자생성 기술 개발 현황

김종태, 홍종간
한국원자력연구원

김찬우, 이상봉
동아대학교

11th OpenFOAM Korea Users' Community Conference
2024. 9.26-27

❑ Research background

- ⭘ **Mesh Generation for CFD Analysis**
  - ⯈ For thermal hydraulic analysis using CFD, it is essential to create a grid for the flow field
- ⭘ **Complex geometry**
  - ⯈ Generating a grid can cause many difficulties
- ⭘ **Mesh generation technology**
  - ⯈ Various methods have been proposed and utilized to overcome the difficulties of generating grids for flow fields with complex geometries
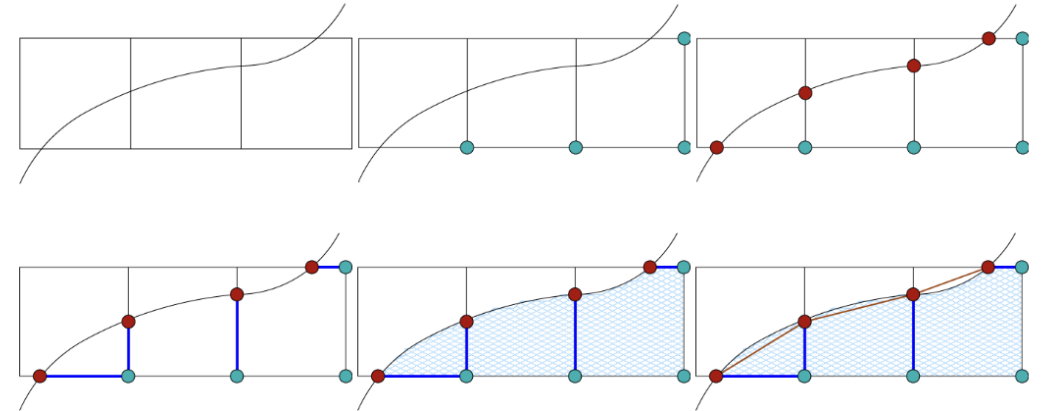
❑ Mesh types for complex domains

- ⭘ Single-block conformal mesh with immersed boundary
- ⭘ Multi-block conformal mesh
- ⭘ Multi-block overset mesh or hybrid mesh
- ⭘ Unstructured tetrahedral, hexahedral, and polyhedral mesh
- ⭘ Unstructured cut-cell mesh
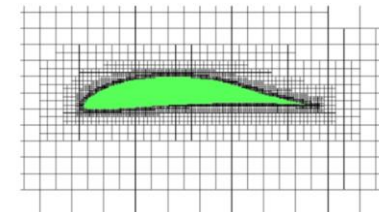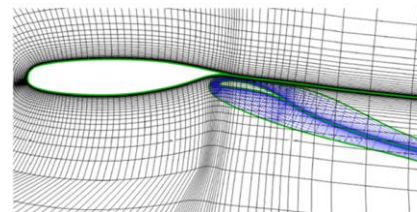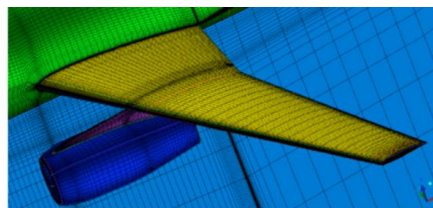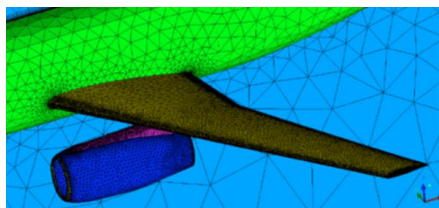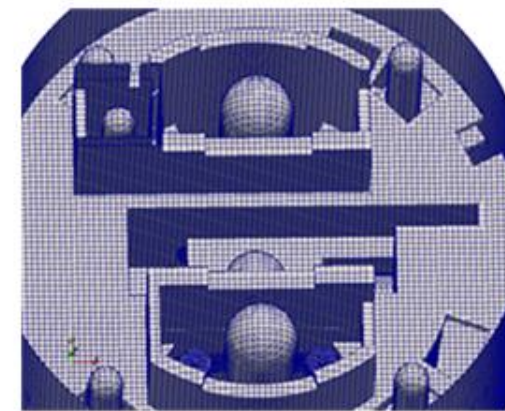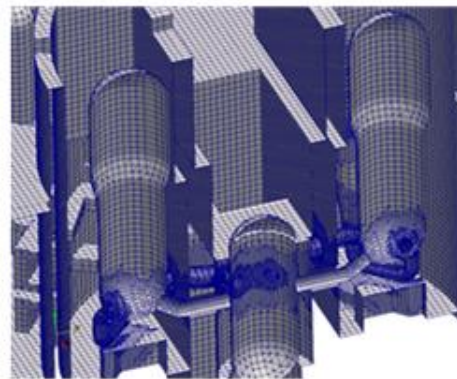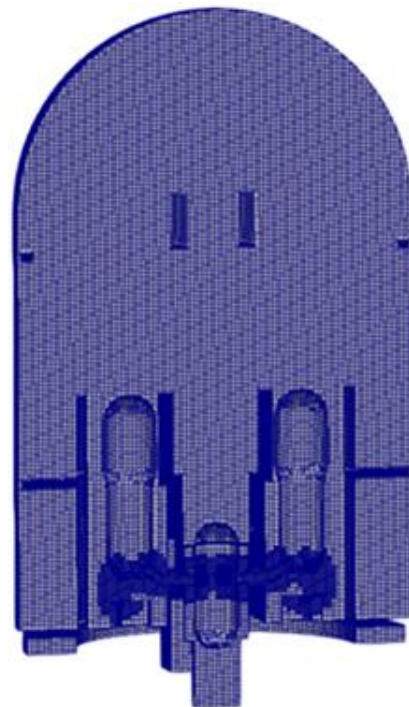
Cut-Cell Mesh Generation for Complex Domains

❑ **Cut-cell mesh ?**

   ❍ **정의: 배경격자와 형상을 오버랩하여 형상을 따라 셀을 제거하거나 잘라서 새로운 격자를 만드는 방식**

❑ **기존 cut-cell mesh generator**

   ❍ **ANSYS: 직각 격자를 배경격자로 사용**

Cut-cell mesh tool 개발

❑ **Cut-cell mesh 요건**

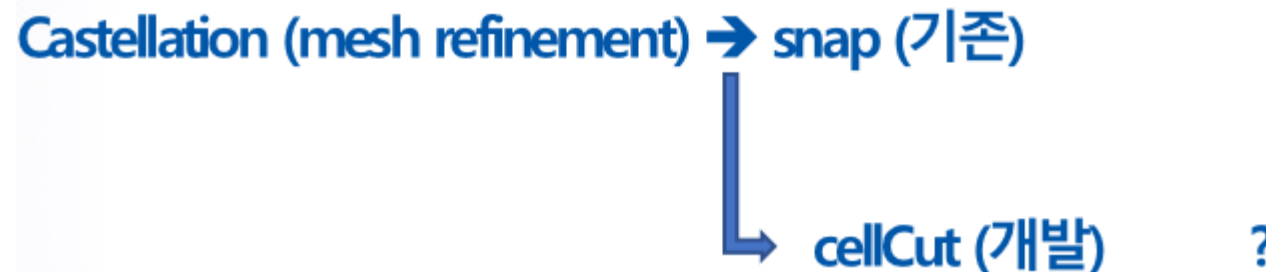 ❍ 배경격자: 최적 배경 격자 사용

    ‣ 원통형 격자, **multi-block** 격자, 비정렬 격자 등 부분적으로 **body-conformal** 격자

    ‣ 현재 상용 **cut-cell** 격자생성은 직각격자 기반 **refinement + cut-cell**

  ❍ 입력 파일과 **Cell cutting**

    ‣ 입력: **OpenFOAM mesh + STL geo** 파일

    ‣ **Cell-cut mesh driver: stand-alone, snappyHexMesh, cfmesh**

❑ **Cut-cell mesh 기능 도입**

  ❍ **SnappyHexMesh 기반 Cut-cell mesh 도구 개발**

    Castellation (mesh refinement) ➔ snap (기존)

                   ⌐→ cellCut (개발)   ?
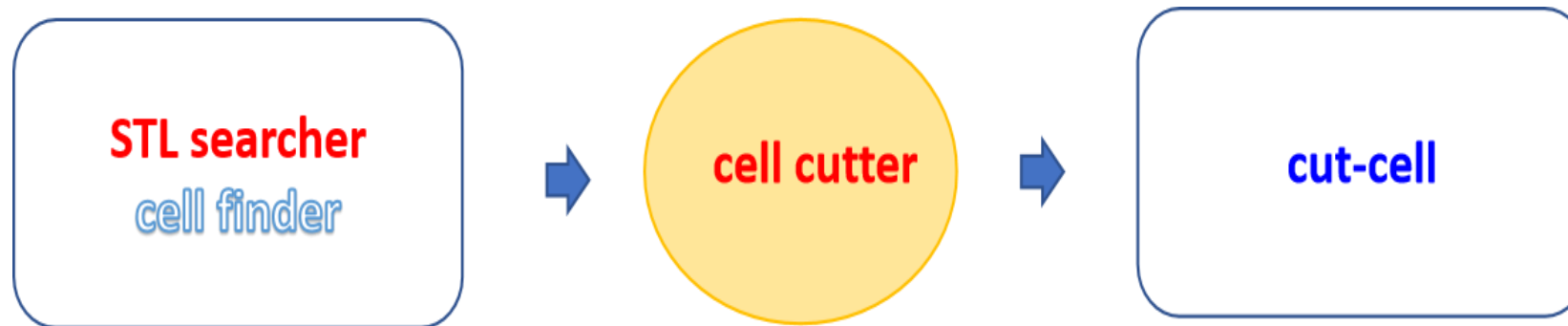
  ❍ **Stand-alone Cut-cell mesh 도구 개발**

   ‣ **topoSet + selectCells + cell-cutter**

❑ **Major Steps for cut-cell mesh generation**

  ○ The currently designed cut-cell tool consists of a core module called cell-cutter and a cell-finder

  ○ **Cell-finder:** STL-formatted surface geometry overlapped on the background grid to find and mark cells passing through the surface

  ○ **Cell-cutter:** cuts cells in the flow area based on the boundary

Partitioning of a cut-cell to fluid and solid cells
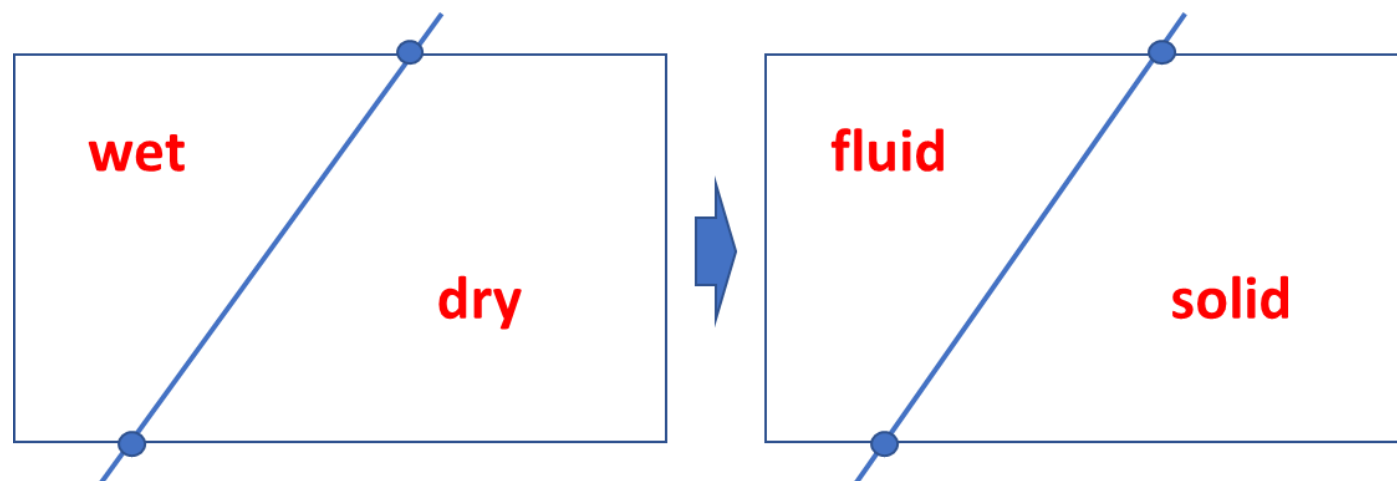
❑ cell cutter 개발 요건

- ❍ 오픈폼 내의 기존 클래스 라이브러리 재활용
  - ▸ **적합한 클래스 찾기 및 선정**
- ❍ cellCutter 클래스 개발
  - ▸ **기존 클래스 상속/복제**
- ❍ 문제점 파악 및 개선
  - ▸ **예상되는 문제점**
    - – **feature edge**

❑ cell cutter 기반 클래스 찾기

- ❍ **geometric VoF (cutCellIso, cutCellPLIC)**
  - ▸ **cut cells with surface defined by cutValue and et al.**
  - ▸ **Not split cells, only get sub-cells volume**
- ❍ **IBS (immersed boundary surface in foam-ext)**
  - ▸ **cut cells with surface in stl file**

❑ IBS (immersed boundary surface)

- ○ foam-ext
    - ▸ **IBS solver with immersed boundary treatment**
- ○ immersedBoundaryPolyPatch
    - ▸ **immersed boundary를 위한 new polyPatch**
- ○ immersedBoundaryPolyPatch.calcImmersedBoundary()
    - ▸ **key function for immersed boundary patch construction**
    - ▸ **Calculate the immersed boundary patch geometry**
    - ▸ **get access to polyMesh**
    - ▸ **contains triSurfaceMesh (ibMesh_)**
    - ▸ **public methods**
- ○ immersedBoundaryFvPatch
    - ▸ **contains immersedBoundaryPolyPatch**
    - ▸ **ibMesh(): Return immersed boundary surface mesh**

❑ Cut-cell modules

○ Cut-cell-finder module

- ▸ **function of finding and marking cells located on the surface**
- ▸ **The cut-cell-finder module receives**
  - − **background grid in OpenFOAM format**
  - − **Surface geometry information in STL format as input**
- ▸ **Marking the cells to be cut, and passes them to the next module as a cut-cell list**
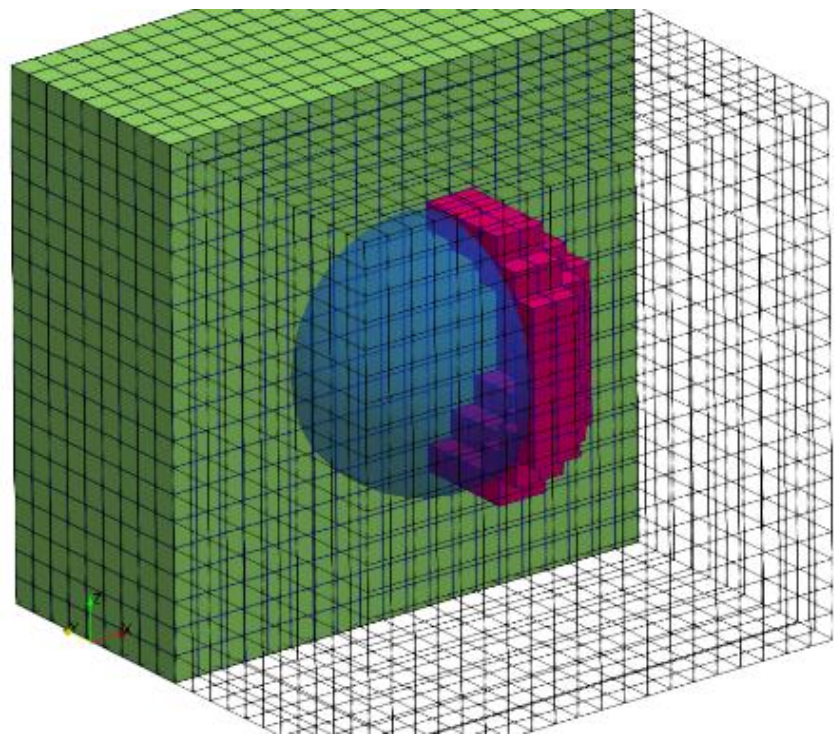
Marking cut-cells on a spherical surface in a cubic background mesh
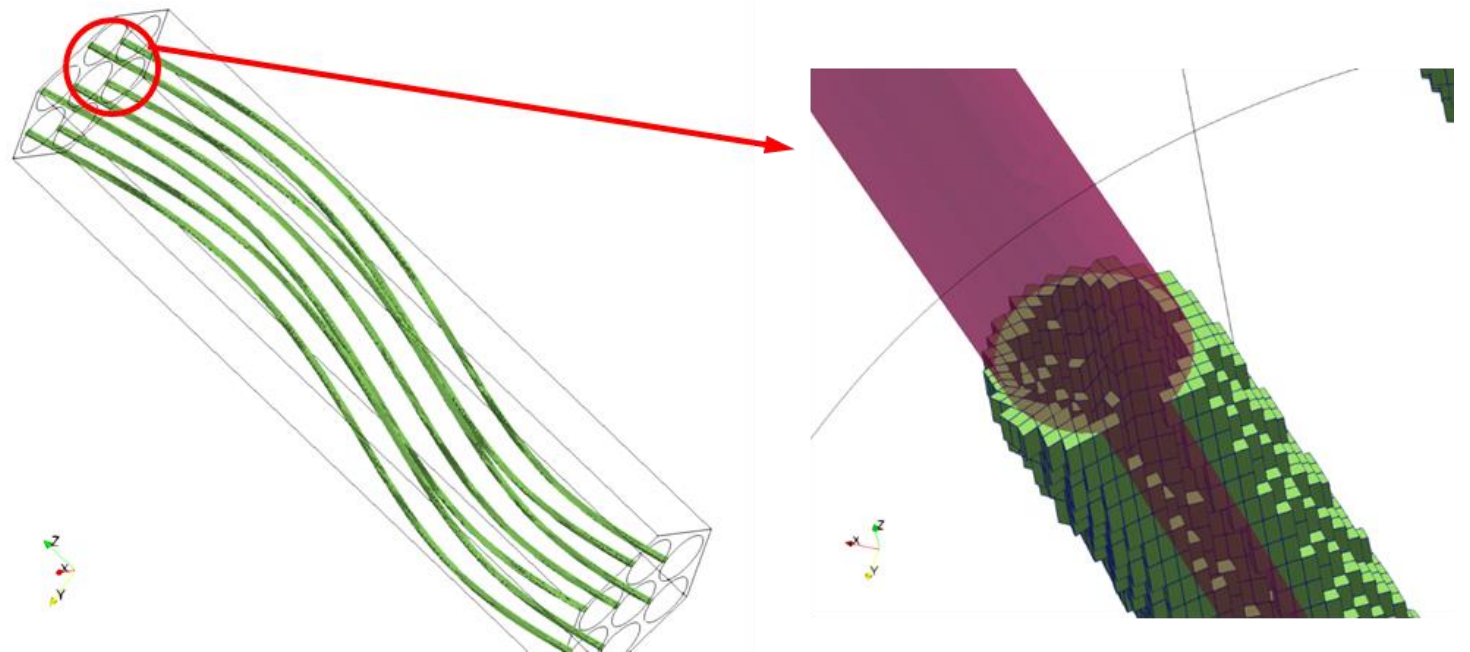
Marking cut-cells on a fuel-pin-wire surface in a hexagonal background mesh

❑ Cut-cell modules

○ Cell-cutter module

- ‣ When a cell in the cut-cell list is cut, its edges and faces are also cut.
- ‣ Neighbor cells of the cut faces must be searched and their connectivity data must be modified.
- ‣ Since the shape of the cut-cell is determined by the mutual position of the cell and the cutting-surface, there are many cases to consider.
- ‣ Therefore, in this study **divide-and-conquer method** was applied to find and correct cut-cell errors.

○ Divide-and-conquer method for debugging cell-cutter module

- ‣ Prepare a list of test cases based on the topology and complexity of the geometry
- ‣ Apply the cell-cutter
- ‣ Generate errors
- ‣ Correct errors
- ‣ Get the next test case and try again

❑ Cell-cutter for basic geometries
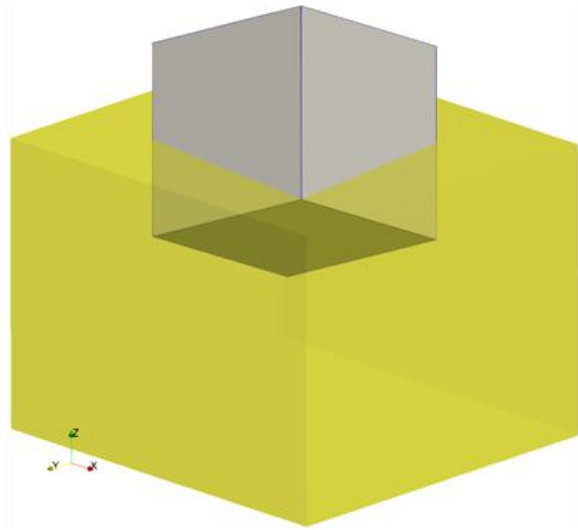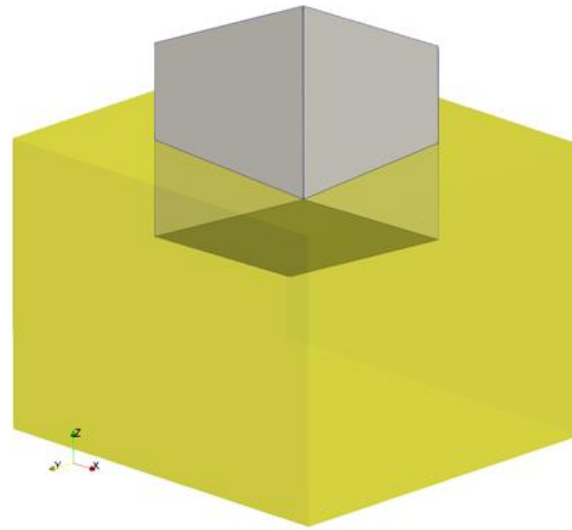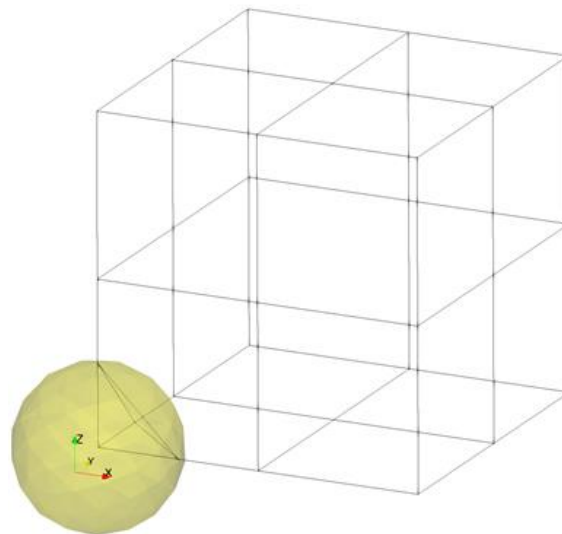
Initial mesh and cutting surface	Cell cutting	Cut mesh
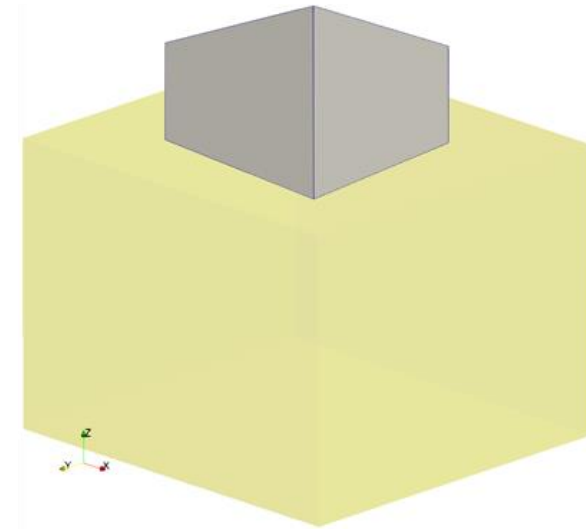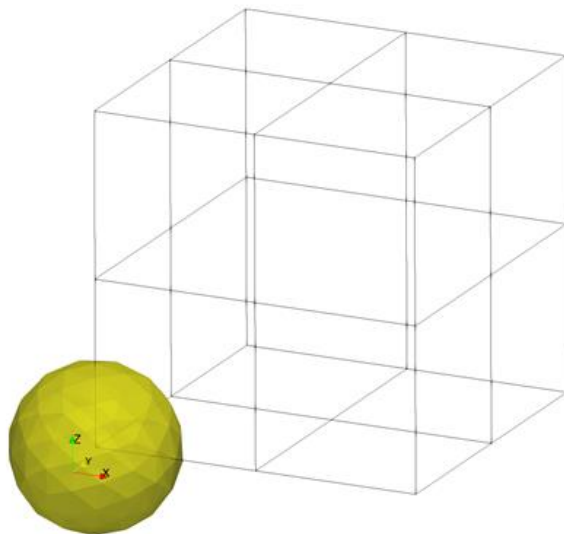
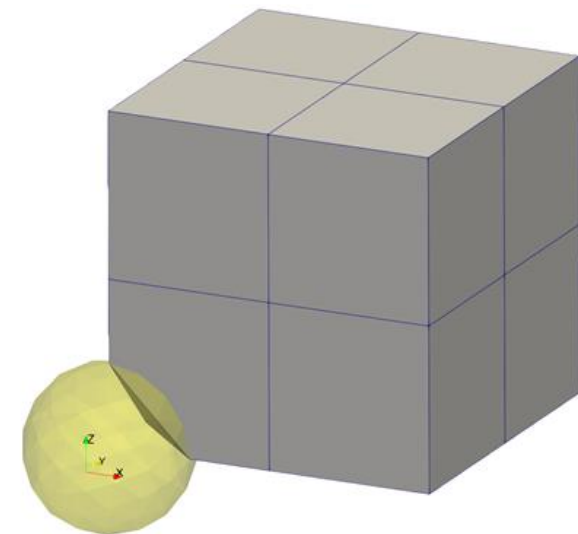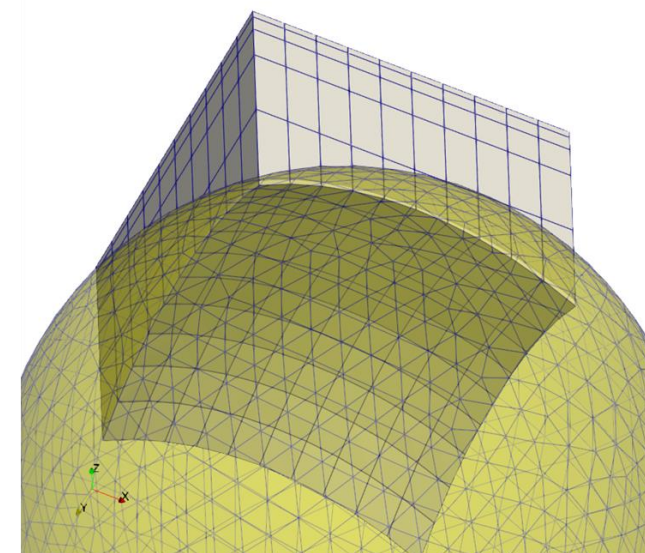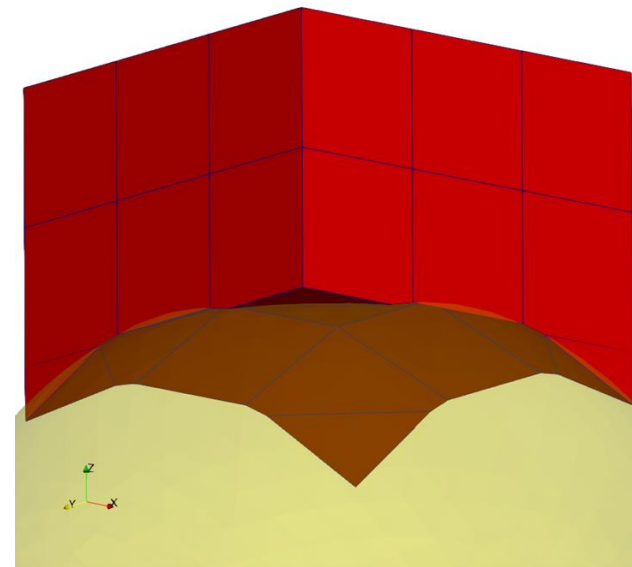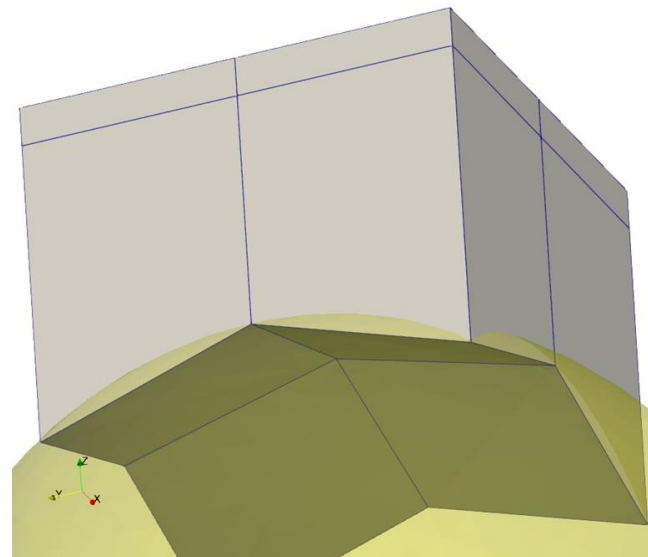| Initial mesh and cutting surface	Cell cutting	Cut mesh

❑ **Cell-cutter errors and corrections**
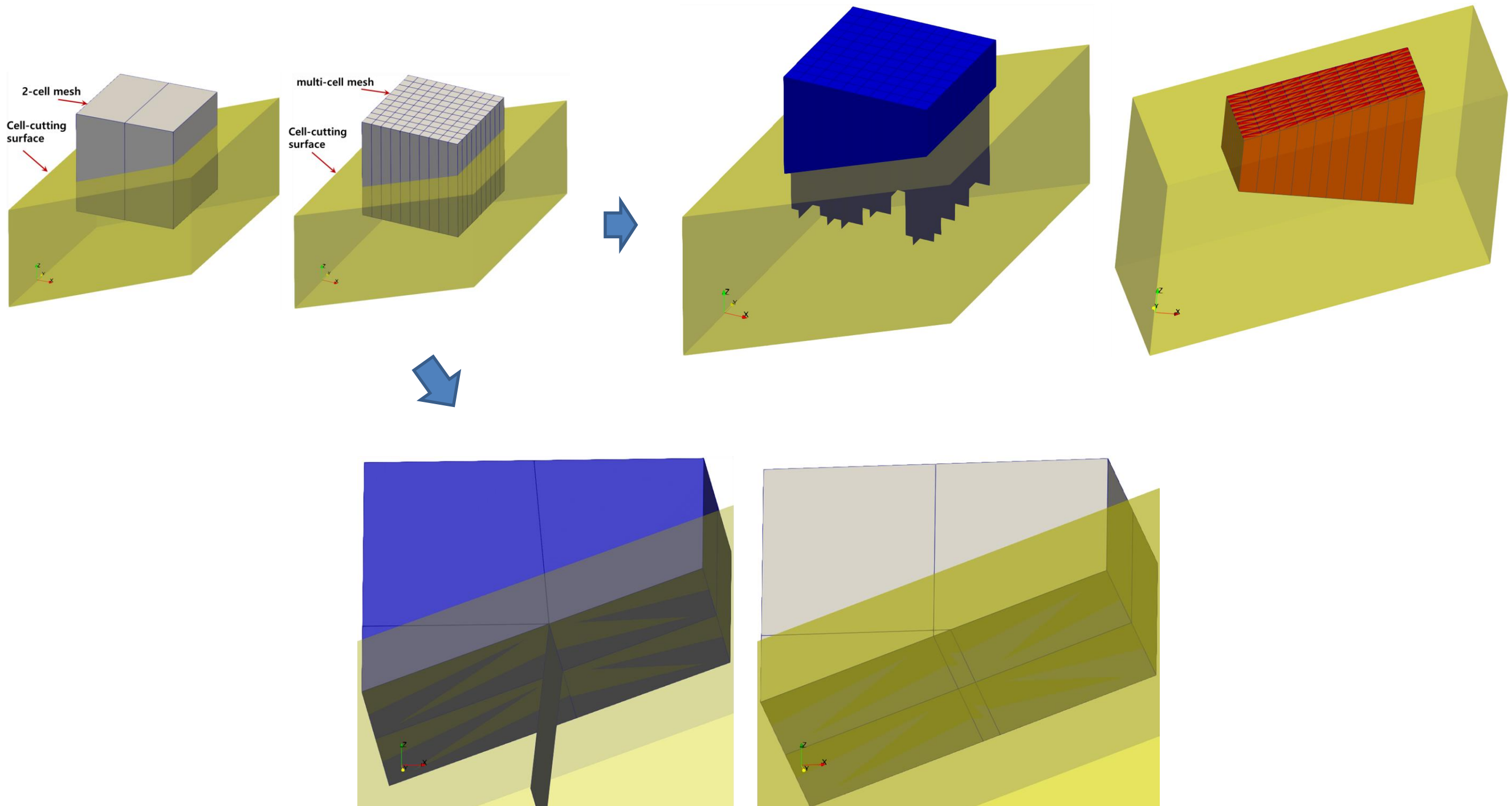
❑ **Cell-cutter errors and corrections**
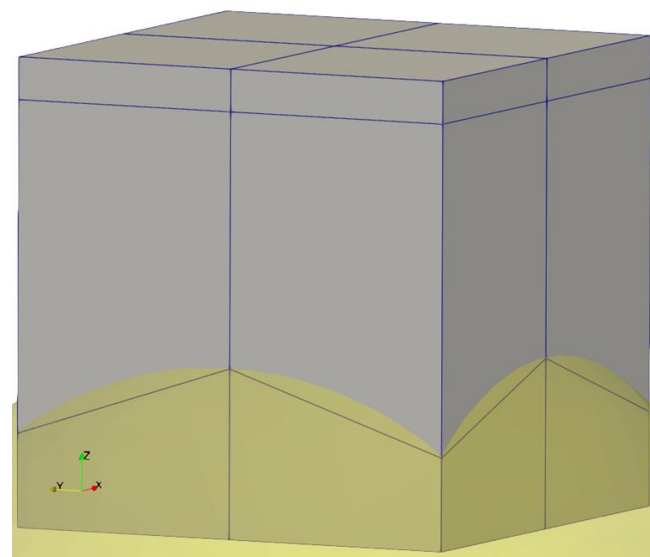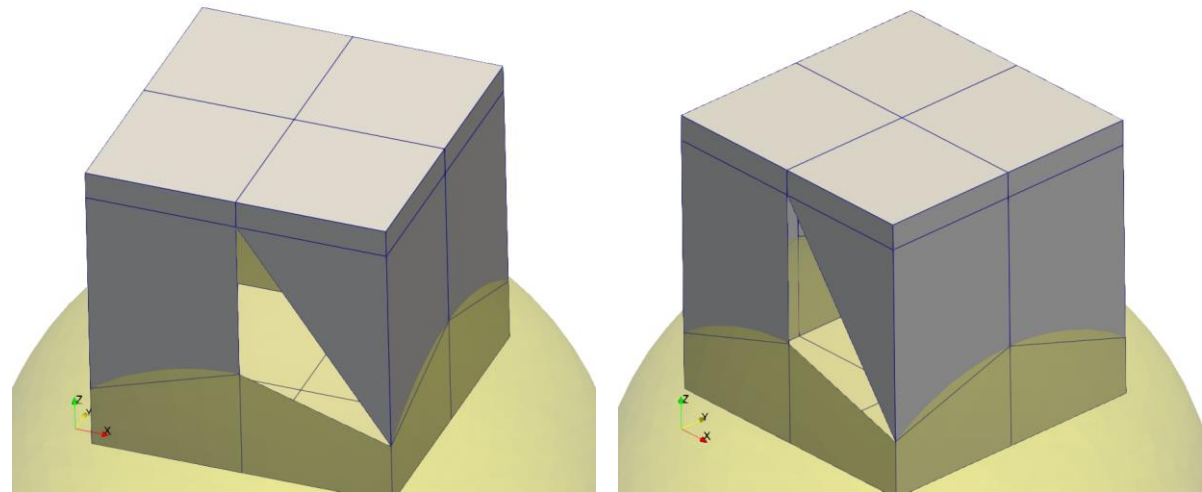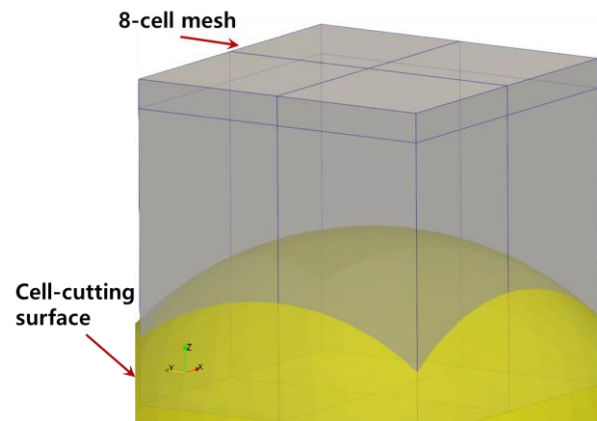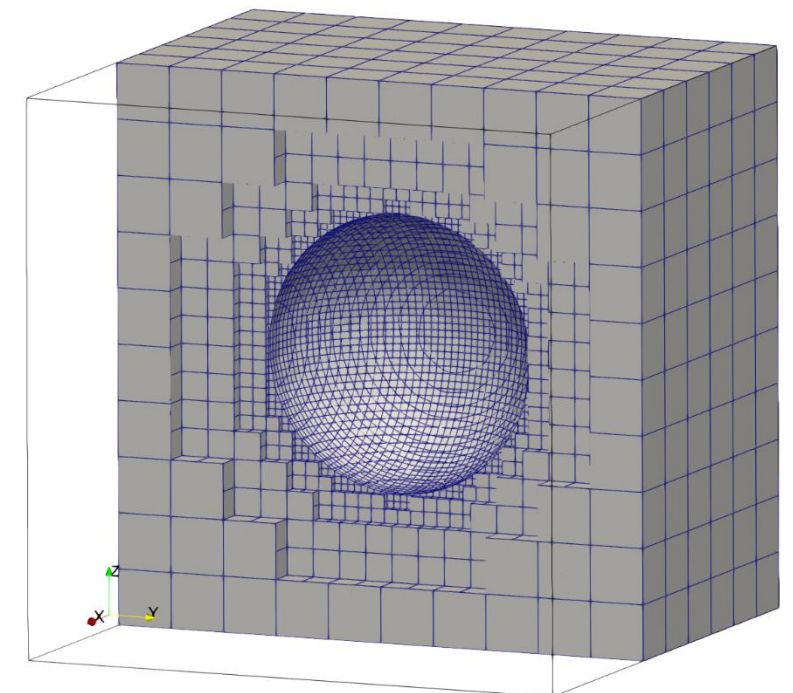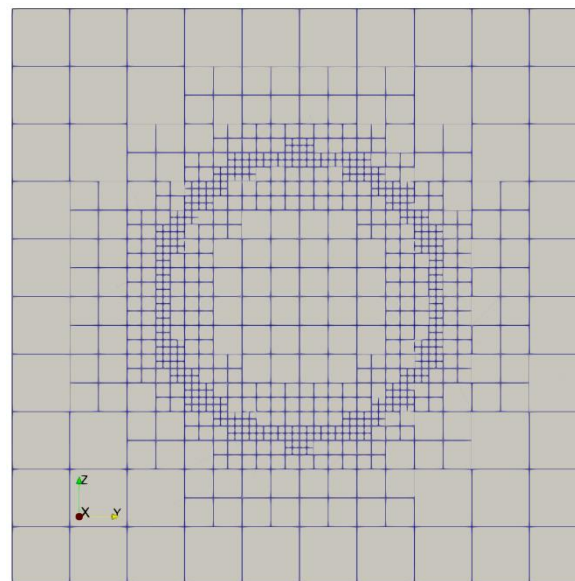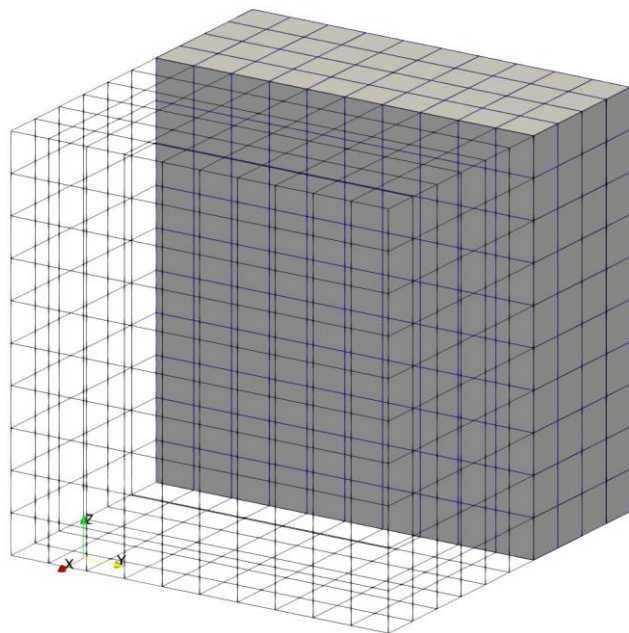
❑ Sphere in a cube

❍ Basic approach

⯮ **Background mesh ➜ cut-cell mesh**
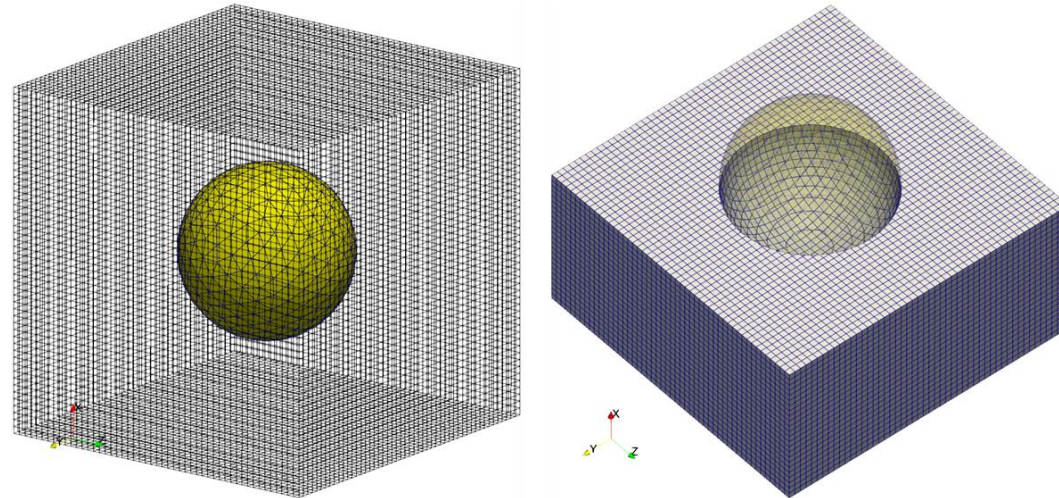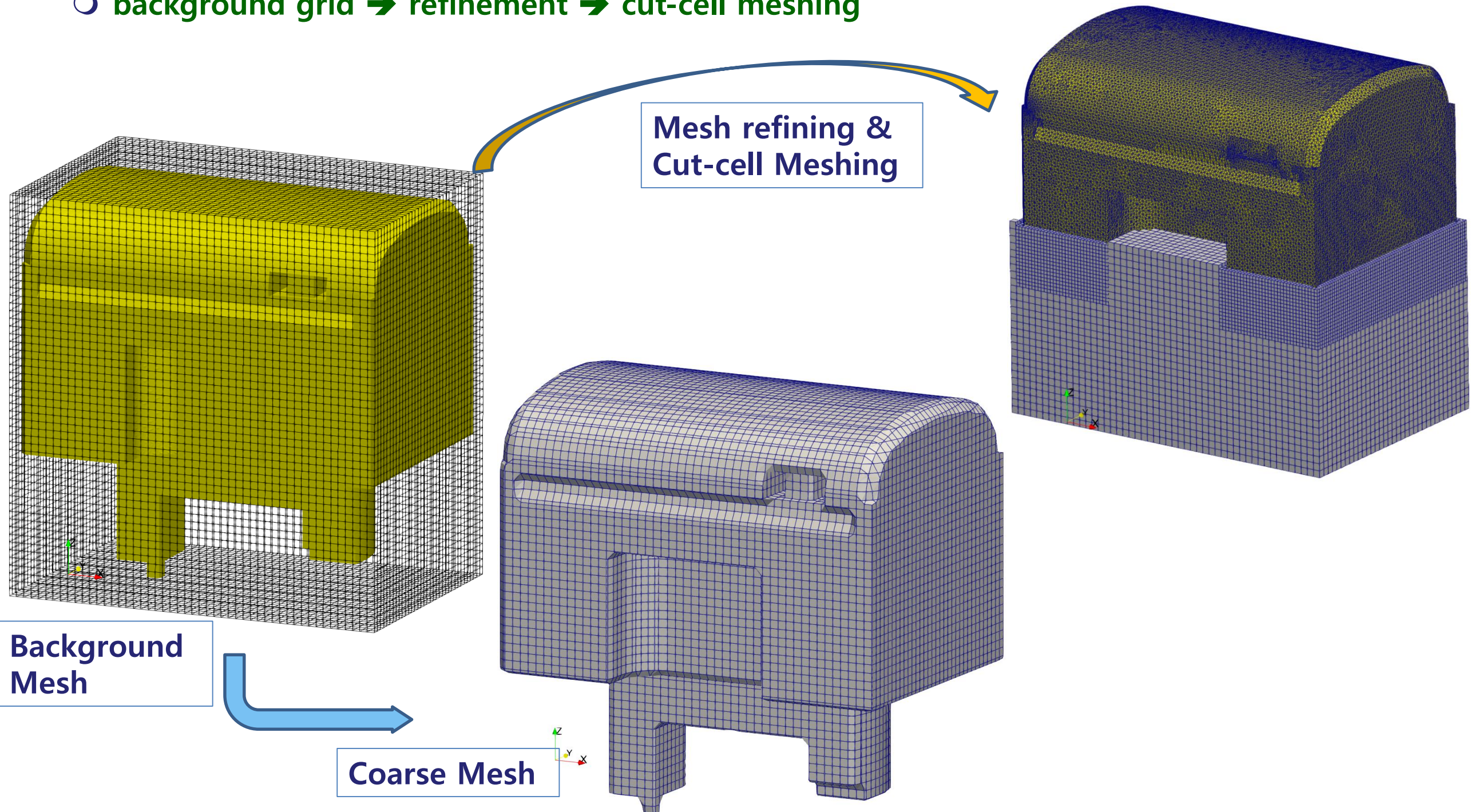

❍ Refinement approach

⯮ **Background mesh ➜ refined mesh ➜ cut-cell mesh**

Applying Cut-Cell Mesh for Complicated Geometry

❑ **Applying the cut-cell algorithm to the SMART100 containment building**

  ❍ **background grid ➔ refinement ➔ cut-cell meshing**

Mesh refining & Cut-cell Meshing

Background Mesh

Coarse Mesh

❑ 요약 및 개선 방향

- ⭕ **IBS 모듈을 재활용하는 방식으로 cut-cell 격자 생성 prototype 도구 개발**
- ⭕ **개발 방법 (divide-and-conquer method)**
  - ▸ **단순 형상의 다양한 구성을 가정하여 cut-cell 도구의 오류를 생성하고 수정하는 방식으로 알고리즘 수정**
- ⭕ **cut-cell 격자 생성 절차**
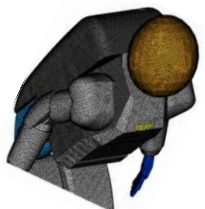 - ▸ **입력: STL-type 형상 모델**
  - ▸ **surface capture: mesh refinement 기능**
  - ▸ **출력: OpenFoam 형식 격자 (boundary, points, faces, owner-neighbour)**

❑ 개선 방향

- ⭕ **코드 안정성, 실행 속도 등 최적화 필요**
- ⭕ **실제 열유동 문제 해석을 통한 검증**

Thank you for your interest