Development of Cut-cell Mesh Generation for CFD Simulation of Complicated Geometries

Jongtae Kim [a*], Jonggan Hong
[a] *KAERI, Daeduk-daero 989-111, Daejeon, Korea*
[*] *Corresponding author: ex-kjt@kaeri.re.kr*

1. Introduction

For thermal hydraulic analysis using CFD, it is essential to create a grid for the flow field. In particular, nuclear reactors such as the SFR (sodium fast reactor) reactor have very complex geometries, and it is necessary to analyze multi-physics phenomena such as flow, structure, and nuclear physics in conjunction with each other. In cases like this, generating a grid can cause many difficulties.

Grid generation techniques that have been developed and commonly used to date include automated unstructured grids and multi-block grids, and a combination of these two methods is also used. Use of unstructured grids is a very efficient method that can be automatically generated, but the number of cells in the grid is very large, so an increase in the calculation amount of flow analysis is inevitable, and it is very disadvantageous in transient or unsteady analysis. Multi-block grids can very efficiently control the number of grid cells along the geometric surface and reflect flow characteristics well, but as the flow field boundaries become more complex, it is difficult to determine the topology of the grid.

Various methods have been proposed and utilized to overcome the difficulties of generating grids for flow fields with complex geometries. As a way to compensate for the disadvantages of the tetrahedral-based unstructured grid, such as the increase in the number of grid cells and the problem of solution accuracy due to the low degree of freedom of a tetrahedral cell (the number of cell faces is 4, which has the lowest interpolation degree of freedom), a method of converting to a dual polyhedral mesh is being used. In a multi-block grid, to simplify the difficult task of dividing the entire flow field into multiple blocks capable of conformal mapping, a body-fitted grid is created only for the local conformal area, and the space between blocks is filled with an unstructured grid.

However, there are still difficulties in grid generation: in poly-dual grids, concave shape cells (negatively affecting the convergence of numerical solutions) may occur, and in multi-block mixed grids, the grid is based on a directional non-uniform grid on surfaces, so additional tools are required to integrate non-uniform grids created from block faces.

Meanwhile, the cut-cell-based mesh generation [1, 2, 3, 4, 5], which constructs a grid by creating a background mesh that includes the entire flow field, then marking the grid cells lying on a surface of a complex geometry and cutting the cells along the surface, has been studied for a long time and is currently in use.

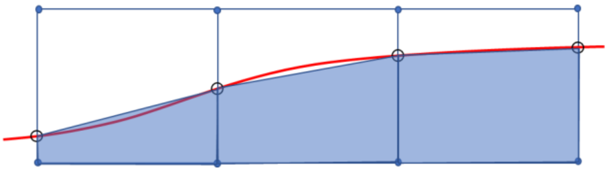

Fig. 1. Schematic to show the concept of the cut-cell mesh.

Fig. 1. schematically shows how to cut out cells lying on the surface shape by overlapping the surface with a background grid. The blue-colored area of the cells can be a flow region.

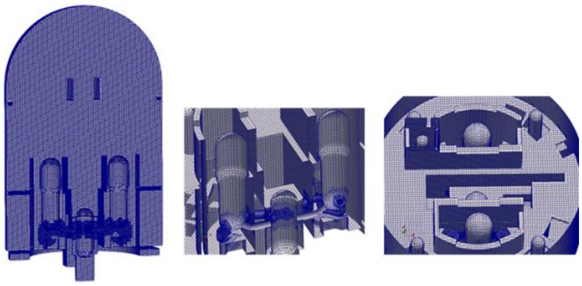

Fig. 2. Commercial cut-cell mesh generator applied for a reactor containment.

Currently, a commercial cut-cell based grid generation tool has been developed. Fig. 2 shows the grid generated using a commercial cut-cell tool for internal flow analysis of a reactor containment building. The commercial cut-cell grid generation tools can create grids for complex shapes very quickly, but as the surface geometry becomes more complex, the size of the grid increases very steeply. This is presumed to be because a shape-independent orthogonal grid is used as a background grid, grid cells are refined for all shape surfaces, and the cut-cell technique is finally applied. This method shows grid characteristics very similar to unstructured automatic grid generation.

In this study, we aim to increase cut-cell efficiency by creating the optimal background grid possible for complex shapes such as nuclear reactors. As a first step to the optimization of the mesh generation, a cut-cell mesh generation tool based on the open-source library OpenFOAM [6] is under development.

2. Methodology Development

2.1 Procedure for cut-cell mesh generation

The currently designed cut-cell tool consists of a core module called cell-cutter and a cell-finder that supports it, as shown in Fig. 3 [7]. Cell-finder uses the STL-formatted surface geometry overlapped on the background grid to find and mark cells passing through the surface and transfers them to the cell-cutter. The cell-cutter cuts cells in the flow area based on the boundary as shown in Fig. 4. It performs the function of configuring a cell by dividing it into a wet volume which becomes a fluid volume and a dry volume outside the wet volume and adding new points.

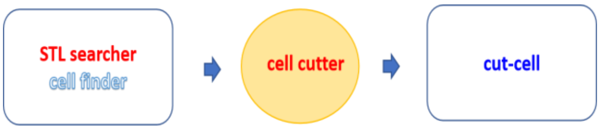

Fig. 3. Procedure for cut-cell mesh generation.

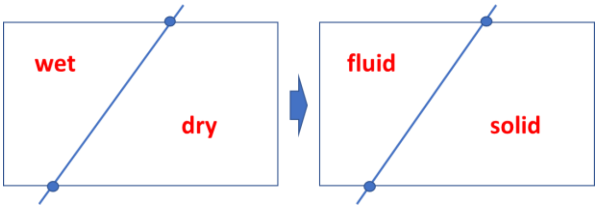

Fig. 4. Partitioning of a cut-cell to fluid and solid cells

We constructed a cut-cell-finder module that has the function of finding and marking cells located on the surface. The cut-cell-finder module receives background grid in OpenFOAM [6, 8, 9] format and surface geometry information in STL format as input, marks the cells to be cut, and passes them to the next module as a cut-cell list.

Fig. 5 shows the marking of cut-cell candidate cells for the spherical surface included in the hexahedral volume by the cut-cell-finder. This module marks cells on the sphere surface and delivers them to the cell-cutter.

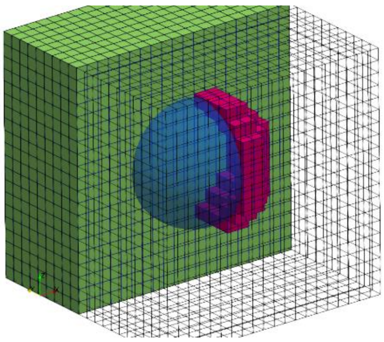

Fig. 5. Marking cut-cells on a spherical surface in a cubic background mesh.

2.2 Debugging cell-cutter by divide-and-conquer method

When a cell in the cut-cell list is cut, its edges and faces are also cut. And 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, the divide-and-conquer method was applied to find and correct cut-cell errors [10].

In heat and fluid flow analysis, the shape of a typical structure consists of a combination of curved surfaces and flat surfaces. Here, simple geometries such as spherical and inclined surfaces were used to evaluate the cut-cell algorithm, which finds grid cells that meet the surface and finds points where edges and the surface meet. Fig. 6 shows a background mesh and the overlapped spherical surface. In the first trial, the cell-cutting procedure failed because the cell-cutting was not propagated to the neighboring cells and boundary faces. Fig. 7 shows that the cut-cell grid is properly created after correcting the errors in the cut-cell tool. It is thought that cell cutting is performed appropriately and the influence of the cut cell is well reflected in neighboring cells.

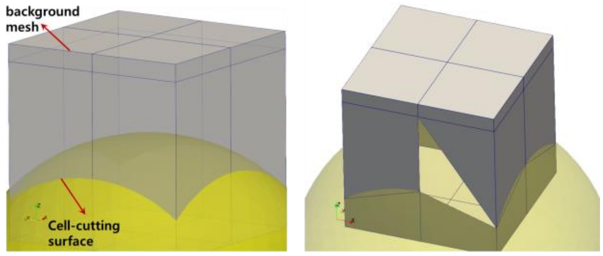

Fig. 6. Failure of a cell cutting using a spherical surface.

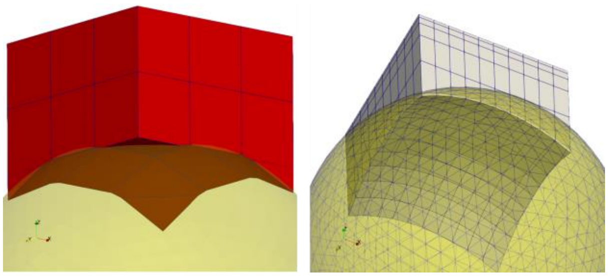

Fig. 7. Successful cutting of cells on a spherical surface.

Depending on the size of the cutting surface, there are several cells that meet this surface, and the cut-cells are formed continuously. In particular, when a cell cut is made to a cell, the cut edges and faces are shared with its neighboring cell, so the shape of the surrounding cells also changes. In cut-cell, the part that has the largest number of cases and requires a large amount of calculation is transmitting the influence of cut edges and faces to neighboring cells. Cell cutting with an inclined flat surface is the most basic problem for verifying the algorithm that transfers the influence of this cut surface to surrounding cells. It is used as an example to verify and modify the algorithm so that the influence of edges and faces can be accurately transmitted to neighboring cells.

Fig. 8 shows that when one grid point is located very close to the surface of the cut shape, the grid created by applying the cut-cell tool has a groove in the center. This results from the fact that the cutting surface is very close to the grid point, so the adjacent grid point is used as a cut-point rather than cutting the edge to create a new grid point. If the grooves at the mesh boundary cause problems in flow analysis, a solution is to move the mesh points close to the cutting surface before performing the cut-cell. In this study, the cut-cell tool was reapplied by adding a function to evaluate these grid points and move them to the surface. Fig. 8 shows another failed case where change of internal face shape by cell-cutting was not correctly shared by the neighboring cell.

If decision of edges with very close mesh points on the cutting surface is applied very strictly (tolerance of 10^{-3}), the mesh can be seen as being cut smoothly without grooves, as shown in Fig. 9.

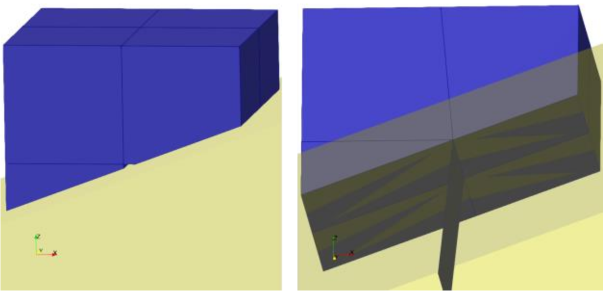

Fig. 8. Failure of a cell cutting using an inclined surface.

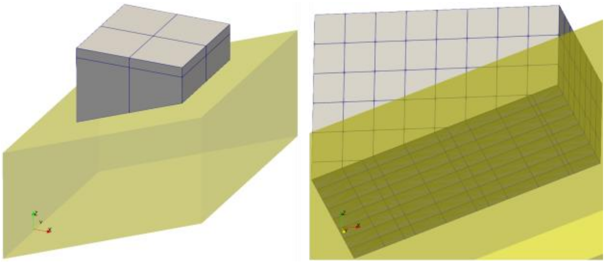

Fig. 9. Successful cutting of cells on an inclined surface.

3. Test Results of the Cut-cell tool

Fig. 10 is an example of verifying cell-cutting using a background grid composed of equally spaced cells and a spherical surface at the center. The right figure shows a spherical space located at the center of the grid by applying the cut-cell tool.

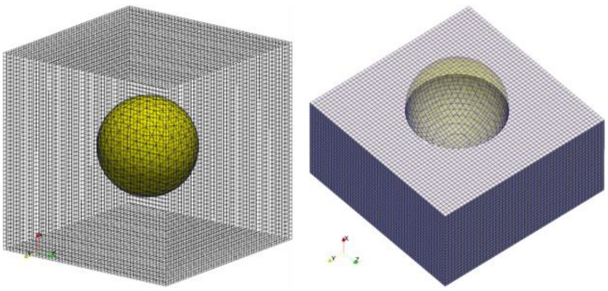

Fig.10. Cut-cell mesh around a sphere using a uniform background mesh

By applying refinement to cells located on the cut-cell surface, the cells can be cut more accurately. Fig. 11 shows the results of performing cut-cell after reducing the cell size by applying refinement to cells located on the surface of a sphere.

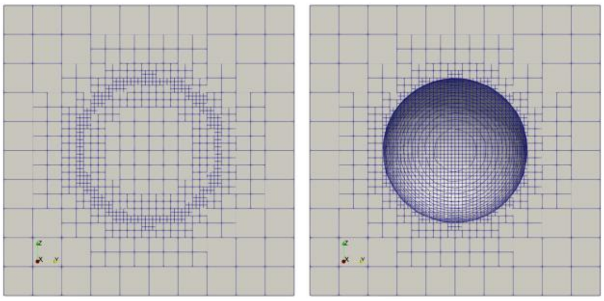

Fig.11. Cut-cell mesh around a sphere using a surface-refined background mesh.

Fig. 12 shows that when there is a U-shaped object inside the flow field, cells inside the object are removed based on the object surface and cells placed on the surface are cut. Through this test, it is thought that the cut-cell algorithm is well applied without errors.

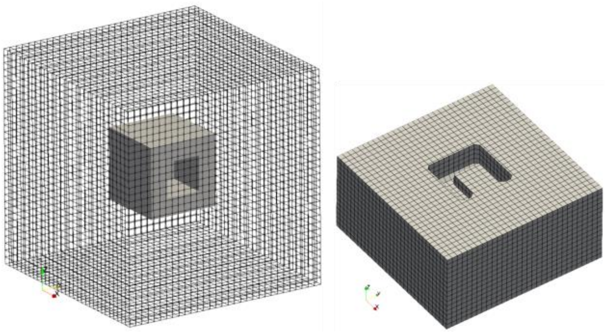

Fig.12. Cut-cell mesh for a flow field around U-shaped component.

A nuclear reactor containment building has a rather complex shape. Fig. 13 shows the results of applying the cut-cell algorithm to the SMART100 containment building. By overlapping the background grid in the form of a simple hexahedron and the surface geometry of the reactor building and applying cut-cell algorithm, the grid for the flow field of the reactor building was obtained.

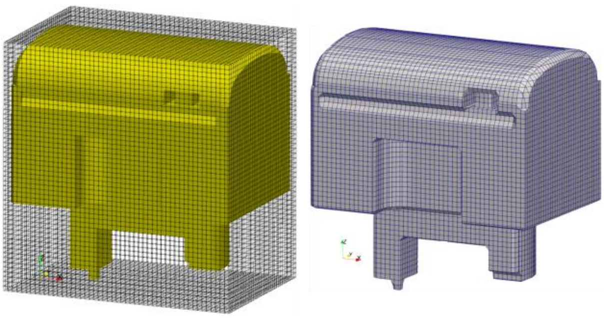

Fig.13. Cut-cell mesh for the containment of the nuclear reactor SMART100.

4. Conclusions

It is aimed to increase cut-cell efficiency by creating the optimal background grid possible for complex shapes such as nuclear reactors. As a first step to the optimization of the mesh generation, a cut-cell mesh generation tool is under development. In this study, a basic tool for cut-cell mesh generation was developed. The algorithmic errors in the tool was found and removed by applying the divide-and-conquer method. The surface refinement method was tested to accurately capture the geometry of cutting surfaces. In the future, the developed cut-cell tool will be improved and used for thermal hydraulic analysis of nuclear reactors with complex shapes.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant and National Research Council of Science & Technology (NST) grant funded by the Korean government (MSIT) [grant numbers 2021M2E2A2081061, CAP20034-100].

REFERENCES

[1] Yuki Takeda,Kazuyuki Ueno,Tatsuya Ishikawa and Yuta Takahashi, " Prediction Capability of Cartesian Cut-Cell Method with a Wall-Stress Model Applied to High Reynolds Number Flows", Appl. Sci. 2020, 10, 5050; doi:10.3390/app10155050

[2] D.M. Ingram, D.M. Causon, C.G. Mingham, "Developments in Cartesian cut cell methods", Mathematics and Computers in Simulation 61 (2003) 561–572.

[3] Ha-Duong Pham, Ph.D. thesis, Modeling A Scroll Compressor Using A Cartesian Cut-Cell Based CFD Methodology With Automatic Adaptive Meshing, 2018, Purdue University

[4] Marsha Berger, Michael J. Aftosmis, Steven R. Allmaras, "Progress Towards a Cartesian Cut-Cell Method for Viscous Compressible Flow", 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Jan. 9-12, 2012. Nashville, TN

[5] Y.G. Vrionis, K.D. Samouchos, K.C. Giannakoglus, “Implementation of a conservative cut-cell method for the simulation of two-phase cavitating flows”, htpps://api.semanticscholar.org

[6] Weller, H., et al., 2024. OpenFOAM: the Open Source CFD Toolbox User Guide. http://www.openfoam.org/docs/user/.

[7] J. Kim, et al., Development of cut-cell mesh generation technology for the analysis of thermal hydraulics in a nuclear reactor based on open-source CFD, SAL-200-E4-457-001, 2022.

[8] OpenFOAM, OpenCFD Ltd., http://www.openfoam.com, 2024.

[9] foam-extended, http://www.sourceforge.net/projects/foam-extended, 2024

[10] J. Kim, et al., Improvement of cut-cell mesh generator for automation of a nuclear reactor mesh generation, SAL-200-E4-457-009, 2023.