



## 기하학적 Octree 격자생성법을 이용한 자동차 헤드램프 내부의 열유동 계산

Computation of Thermal Flow for Automotive Lamp  
by Using Geometric Octree Method

\*사 종업<sup>1)</sup>\*, 박종렬<sup>2)</sup>, 강동민<sup>3)</sup>

Jong-Youb Sah, Jong-Ryul Park, Dong-Min Kang

Three dimensional orthogonal grid generation is able to control effectively the grid spacing near the boundaries, but there are some difficulty to meshing complex geometry. To mesh complex geometry by orthogonal grid generation method must divide block of geometry. It is required a careful skill, and long time. Its also difficulty to make unstructured mesh on complex geometry. Particularly, three dimensional geometry must have more time and effort. Recently, there have been growing interests in mesh generation of complex geometry, aslike an automobile headlamp, the heart. The method of easily meshing complex geometry is resarched to solve them. We suggest octree grid into one among these methods. As octree grid is automaticaly adapted at the boundaries by determine the level, operations to control the grid spacing near the boundaries are unnecessary. In this paper we showed three dimensional mesh generation, and heat-flow analysis on the octree mesh

### 1. 서론

수치해석을 이용한 유동 현상을 해석하는 전산유체역학이라는 학문의 발전은 산업전반에 중요한 기술로 적용되어 왔다. 복잡한 공업장치의 열전달 및 고성능 유체 기계의 유동을 실험적인 방법에 의해 측정한다는 것은 쉽지 않으며, 개발 단계에서 이러한 측정값을 이용하는 것은 막대한 시간과 비용과 고도의 실험기술이 요구된다. 이러한 실험이 수치해석으로 결과를 예측할 수 있다면 많은 비용과 시간을 절약할 수 있을 것이다. 이러한 기대와 컴퓨터의 급속한 발달에 힘입어 과거에는 이론적으로 해석할 수 없었던 복잡한 열, 유체 유동해석이 현재는 수치 해석적으로 가능하게 되었다.

수치해석의 발전으로 복잡한 형상에 대한 열, 유체 유동해석이 가능하게 되었으나 비전문인이 격자를 생성하고 유동해석을 하기에는 많은 시간과 전문적인 지식이 필요하다. 개발단계에서 복잡한 형상을 설계하고 유동해석을 하기 위한 격자

를 생성하는 데에는 격자를 형성하기 위한 전문적인 지식이 필요하게 되고 전문적인 지식이 있더라도 많은 시간과 노력이 요구된다. 유동해석을 하기 위해 소요되는 많은 시간과 노력은 개발 단계에서 많은 부담감을 줄 수 밖에 없다. 이런 부담감을 줄이기 위해서 본 연구에서는 Octree 격자를 사용하여 비전문인이라도 격자를 손쉽게 형성해서 유동해석을 할 수 있도록 하였다.

S.Chang [Ref 5.]은 유동을 해석하는 과정에서 Octree 형태의 적응격자를 사용하여 수렴속도를 비교하였고 그 외 선행 연구자들 또한 유동 영역에서 Octree 형태의 적응격자를 사용한 반면에 본 연구에서는 기하학적 형상이 복잡한 경우에 대한 격자 형성을 빠르게 하기 위해서 Octree 형태의 격자를 사용하였다. Octree 격자기반에서 열, 유체 유동해석을 하기 위해 유한체적법(Finite Volume Method, FVM)으로 이산화를 하였으며 SIMPLE 알고리즘을 사용하였다. Octree 격자의 특성상 cell의 중앙에서 해석을 하기 때문에 nonstaggered 격자를 사용하였다. Nonstaggered에서 압력의 decoupling을 방지하기 위해서 Rhie, Chow [Ref 1.]의 운동량 보간법을 사용하였다.

- 1) 영남대학교 기계공학과
- 2) 영남대학교
- 3) 영남대학교

Octree 격자는 경계부근에서 자동으로 밀집도가 일정하게 유지되며 밀집도를 조절하기 쉽고 직교 좌표계를 사용하기 때문에 일반 좌표계에 대한 좌표변환이 불필요하다.

본 연구에서는 객체지향형 프로그래밍기법을 이용하여 사용자가 사용하기 편한 윈도우의 GUI(Graphic User Interface)환경하에서 개발되었으며 IGES파일의 복잡한 3차원 형상Data를 입력 받아 Level에 따라 Octree격자를 자동으로 생성하고 3차원형상에 경계조건을 설정할 수 있는 전처리기와 열유체 유동해석을 하는 Solver및 해석된 결과를 가시화하는 후처리기로 분리 개발하였다.

## 2. 격자 형성

### 2.1 Level에 따른 격자

Level은 Initial cell을 level 0 설정하고 그 cell을 8등분으로 나누어진 cell은 이전 cell의 level보다 1단계 올라가게 된다. cell이 나누어지게 되면 기존의 cell은 Parent가 되고 나누어진 cell은 child가 된다. Fig 1.는 특정 지점에 level 3이 되도록 격자를 형성하는 과정을 나타낸 것이다.

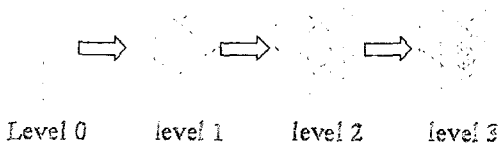


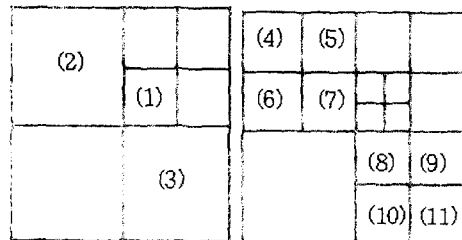
Fig. 1 Meshing at each level

### 2.2 격자형성 방법

Level을 올릴 때에는 주변cell의 level을 조사하여야 한다. cell간의 level차이가 1이상일 경우에는 cell과 cell간의 Interface에서 여러 경우에 대한 처리가 복잡해지고 계산이 어렵고 시간이 오래 걸리므로 본 연구에서는 cell간의 level 차이가 1이하를 유지하도록 하였다. cell간의 level 차이를 1이하로 유지할 경우 cell과 cell간의 Interface에 처리에 대한 경우가 2 가지의 경우로 나타난다. Level이 서로 같아 level의 차이가 0인 경우와 level의 차이가 1인 경우이다. 이 방법을 사용하였을 경우 level의 차이가 급격하게 변하

지 않고 선형적으로 변하기 때문에 유동계산에 무리가 가지 않고 생성된 격자가 자연스럽게 된다. Fig 2.에서 나타낸 것과 같이 cell (1)의 Level을 올리는 과정에서 주변cell의 level차이를 1로 유지하기 위해서 cell(2)와 cell(3)의 level를 올려야 한다. 이러한 과정에서 cell(4 ~11)들이 부수적으로 생성된다.

Level을 올릴 때에는 주변cell의 level을 조사하여야 한다. cell간의 level차이가 1이상일 경우에는 cell과 cell간의 Interface에서 여러 경우에 대한 처리가 복잡해지고 계산이 어렵고 시간이 오래 걸리므로 본 연구에서는 cell간의 level 차이가 1이하를 유지하도록 하였다. cell간의 level 차이를 1이하로 유지할 경우 cell과 cell간의 Interface에 처리에 대한 경우가 2 가지의 경우로 나타난다. Level이 서로 같아 level의 차이가 0인 경우와 level의 차이가 1인 경우이다. 이 방법을 사용하였을 경우 level의 차이가 급격하게 변하지 않고 선형적으로 변하기 때문에 유동계산에 무리가 가지 않고 생성된 격자가 자연스럽게 된다. Fig 6.에서 나타낸 것과 같이 cell (1)의 Level을 올리는 과정에서 주변cell의 level차이를 1로 유지하기 위해서 cell(2)와 cell(3)의 level를 올려야 한다. 이러한 과정에서 cell(4 ~11)들이 부수적으로 생성된다.



(a) before meshing (b) after meshing

Fig. 2 Traetment of neighbor cells in meshing

## 3. 계산방법

### 3.1 지배방정식

연속방정식

$$\iiint \frac{\partial}{\partial x}(\rho u) dV + \iiint \frac{\partial}{\partial y}(\rho v) dV + \iiint \frac{\partial}{\partial z}(\rho w) dV = 0$$

운동방정식

$$\iiint \nabla(\rho \phi v_i) dV = -\iiint \nabla p dV + \iiint \nabla^2(\mu \phi) dV + \iiint (\rho_0 - \rho) g dV$$



에너지방정식

$$\iiint \nabla(\rho c_p T v_i) dV = \iiint \nabla^2(\lambda T) dV$$

Laplace 방정식

$$\bullet \iiint \nabla^2(\lambda T) dV = 0$$

x,y,z	Coordinates
T	Temperature
S	Source
	Volume expansion coefficient
Pr	Prandtl number
Ra	Rayleigh number
u,v,w	Velocities
P	Pressure
	Density
	Thermal conductivity
Cp	Specific heat at constant pressure
	Dynamic viscosity
TH	High Temperature
TL	Low Temperature

Table. 1 Notation

수치 해석은 비압축성 정상 상태의 Navier-Stokes 방정식을 이용하여 FVM으로 이산화하였으며 SIMPLE 알고리즘을 사용하였다. Octree 격자의 특성상 cell의 중앙에서 해석을 하기 때문에 nonstaggered 격자를 사용하였다. Nonstaggered 에서 압력의 decoupling을 방지하기 위해서 Rhie, Chow [Ref 1.]의 운동량 보간법을 사용하였다.

### 3.2 Level이 다른 격자 간의 Flux 계산

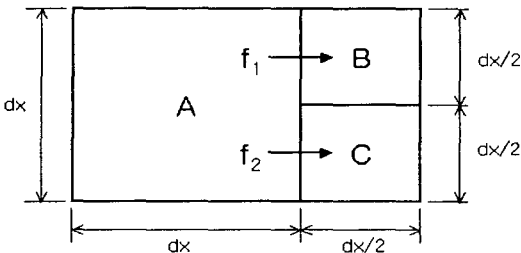


Fig. 3 Flux at interface of different levels

Fig 3.은 Level이 다른 Cell간의 flux를 2차원

으로 나타낸 것이다. A cell과 B,C cell사이의 Flux는 각각 f1, f2 이다. 이것을 다음과 같이 대표 flux F로 나타낼 수 있다.

$$F = \frac{f1 \times \frac{dx}{2} + f2 \times \frac{dx}{2}}{dx} = \frac{f1 + f2}{2}$$

윗식을 3차원으로 확장한다면 다음식으로 나타낼 수 있다.

$$F = \frac{f1 \times (\frac{dx}{2})^2 + f2 \times (\frac{dx}{2})^2 + f3 \times (\frac{dx}{2})^2 + f4 \times (\frac{dx}{2})^2}{(dx)^2} = \frac{f1 + f2 + f3 + f4}{4}$$

### 4. 계산 예

복잡한 형상에 대한 계산을 수행하기 위해서 상당히 많은 부품으로 이루어져 있는 Fig. 4의 자동차 헤드램프에 대해서 유동해석을 하였다. 초기 격자수는 10\*10\*10 에서 Level 6으로 격자를 형성하고 유동해석을 수행하였다.

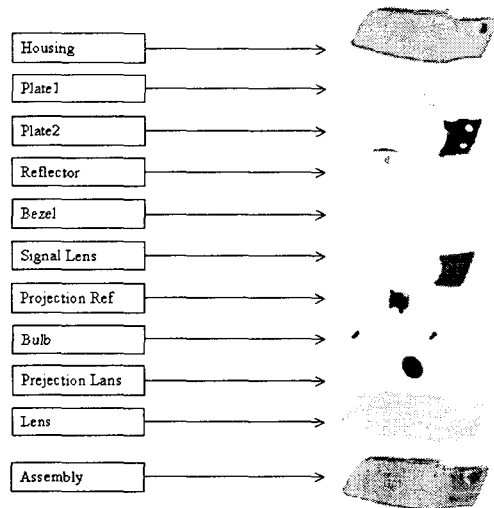
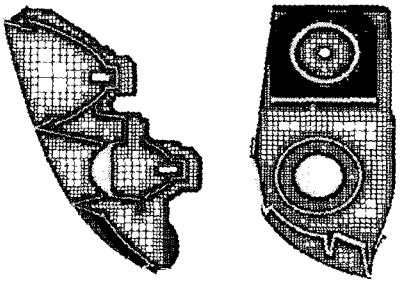
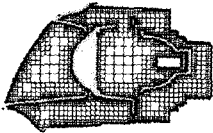


Fig. 4 Assembly of an automotive lamp

4.1 격자 생성



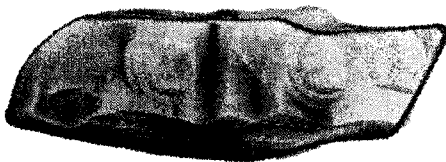
(a) XY section (b) YZ section



(c) XZ section

Fig. 5 Mesh generation of Automotive lamp

4.2 계산결과



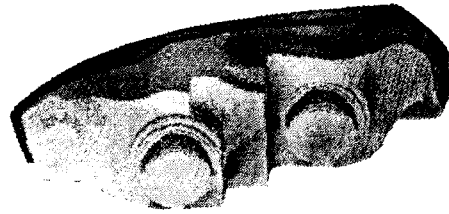
(a)



(b)



(c)



(d)

Fig. 6 Computational Result

5. 결론

수치해석을 이용한 유동 현상을 해석하는 전산유체역학이라는 학문의 발전과 더불어 컴퓨터의 용량과 연산 속도의 증가에 따라 실제 산업 현장에서 요구되는 복잡한 형상의 유동 문제에 적용되게 되었다. 그러나 복잡한 형상의 격자를 형성하는 것은 많은 시간과 기술을 필요로 한다. 이런 문제점을 해결하기 위해 Octree형태의 격자를 사용하여 복잡한 형상의 격자를 쉽게 형성하고 열,유동 해석을 하였다.

3차원 Octree격자를 형성할 경우 메모리의 사용이 적은 Index에 의한 방법과 메모리의 사용이 많은 Point에 의한 방법을 사용하였을 때 거의 같은 시간안에 격자가 형성되었으며 격자를 형성할 때에는 메모리의 사용이 적은 Index에 의한 방법이 효율적이었다. 그러나 유동해석을 할 경우, 주변격자에 대한 계산이 많이 이루어지기 때문에 Point에 의한 방법이 효율적이었다.

본 연구를 통하여 개발된 코드는 3차원 비압축성·점성 유동을 유한체적법을 이용하여 해석하는 통합 코드이다. 이 코드는 GUI(Graphic User Interface) 환경하에서 Iges 형태의 복잡한 형상을 입력하고, Level에 의해 Octree형태의 격자를 형성한다. 해석할 영역과 경계 조건, 물성치를 입력한 다음 해석프로그램을 실행하기 위한 파일을 만드는 전처리기와 전처리기에서 만들어진 해석용 파일을 입력받아 유동해석을 수행하는 유동 해석 코드, 해석된 결과를 모니터링 할 수 있는 후처리가 연계된 통합 코드를 개발하였다.

본 연구에서 개발된 코드는 실제 산업 현장에서 해석하고자 하는 복잡한 형상에 대해 격자를 빠른 시간안에 쉽게 형성할 수 있고 형성된 격자를 이용하여 열,유동 해석을 수행하여 실제 현상



의 해석에 기여도가 매우 클 것으로 기대된다.

### 참고문헌

- [1] C.M.Rhie, W.L.Chow "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation", AIAA Journal, Vol.21, No.11, 1983, pp1525-1532
- [2] Majumdar "Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids", Numerical Heat Transfer, Vol.13,1988,pp.125-132
- [3] Michael J. Laszlo "Computational Geometry and Computer Graphics in C++" (1996)
- [4] Suhas V.Patankar "Numerical Heat transfer and Fluid Flow" 1980
- [5] S.Chang, D.C.Haworth "Adaptive grid refinement using cell-level and global imbalances", Numerical methods in fluids, Vol.24, 1997,pp375-392
- [6] Berger, M.J. and LeVeque, R.J., "An Adaptive Cartesian Mesh Algorithm for the Euler Equations in Arbitrary Geometries", AIAA-89-1930, 9th Computational Fluid Dynamics Conference Buffalo, New York, June 1989.
- [7] Balasubramaniam Ramaswamy, Efficient Finite Element Method For Two-Dimensional Fluid Flow And Heat Transfer Problems, Numerical Heat Transfer, Part B, Vol.17, pp.123-154 (1990)
- [8] G.De Vahl Davis, I.P. Jones, Natural Convection in A Square Cavity : A Comparison Exercise, International Journal For Numerical Methods In Fluids, Vol.3, 227-248 (1983)
- [9] U.Ghia, K.N.Ghia, and C.T.Shin, High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method, Journal of Computational Physics 48, pp.387-411(1982)