Development of MUST (*M*ulti-group *U*nstructured geometry S_N *T*ransport) Code

[*]Ser Gi Hong , Jong Woon Kim, and Young Ouk Lee
[a]*Korea Atomic Energy Research Institute, Duckjin-dong, Yuseong-gu, Daejon, Korea*

[*]*Corresponding author: hongsg@kaeri.re.kr*

1. Introduction

In this paper, a computer code MUST (***M*ulti-group *U*nstructured geometry S_N *T*ransport**) for the neutral particle (e.g., neutron and photon) transport calculations in three-dimensional unstructured geometry is introduced. This code solves the discrete ordinates transport equation by using the discontinuous finite element method (DFEM)[1,2] where the problem domain is divided into tetrahedral meshes. The purpose of this paper is to report the status of the development and the preliminary test results of the code.

2. Theory and Method

This section describes the basic equations and the calculation schemes of the discontinuous finite element method for the discrete ordinates transport equation because MUST uses presently this method. The starting equation is the neutral particle transport equation given by

$$\hat{\Omega}_m \cdot \nabla \varphi_{m,g}(\vec{r}) + \sigma_g \varphi_{m,,g}(\vec{r}) = S_{m,g} + Q_{m,g}, \tag{1}$$

where *m* and *g* mean the directions of neutral particle and the energy group, respectively. The first and the second source terms represent the scattering source and the external source, respectively. In DFEM, the problem domain is divided into tetrahedral meshes. In each tetrahedral mesh, the flux is expanded in terms of the trial functions as follows :

$$\varphi_{m,g}(\vec{r}) = \vec{\Theta}^T(\vec{r})\vec{\psi}_{m,g} = \sum_{p=1}^{P} \psi_{m,g,p}\gamma_p(\vec{r}), \tag{2}$$

where $\gamma_p(\vec{r})$ are the trial functions and *P* is the number of the trial functions. The trial function $\gamma_p(\vec{r})$ should be unity at $\vec{r}_p$ but it should be zero at the other node points. The integration of Eq.(1) over a tetrahedral mesh (*k*) after substitution of Eq.(2) into Eq.(1) and multiplication of the basis function vector gives

$$\sum_{l=1}^{N_f} \int_{A_{k,l}} \hat{\Omega}_m \cdot \hat{n}_{k,l} \vec{\Theta}\vec{\Theta}^T \vec{\psi}_{m,g}^{s,k,l} dA - \hat{\Omega}_m \cdot \int_{V_k} [\nabla \vec{\Theta}] \vec{\Theta}^T \vec{\psi}_{k,m,g} dV + \int_{V_k} \sigma_g \vec{\Theta}\vec{\Theta}^T \vec{\psi}_{k,m,g} dV = \int_{V_k} \vec{\Theta}\vec{\Theta}^T [\vec{s}_{k,m,g} + \vec{q}_{k,m,g}] dV. \tag{3}$$

Next, Eq.(3) is rewritten as the matrix and vector form and the incoming faces are separated from the outgoing faces. The final equations are given by

$$[\sum_{l=1}^{N_{f,out}} U_{m,k,l}^{out} + K_{m,k} + \sigma_g M_k]\vec{\psi}_{k,m,g} = M_k(\vec{s}_{k,m,g} + \vec{q}_{k,m,g}) + \sum_{l=1}^{N_{f,in}} U_{m,k,l}^{in} \vec{\psi}_{\alpha\{l\},m,g} \tag{4}$$

In Eq.(4), it should be noted that the nodes fluxes on the incoming faces are given by the calculation for the upstream mesh neighboring through the incoming faces. The matrices of Eq.(4) are given by

$$\begin{aligned} [K_{m,k}]_{ij} &= -\int_{V_k} [\mu_m \frac{\partial \gamma_i}{\partial x} + \eta_m \frac{\partial \gamma_i}{\partial y} + \xi_m \frac{\partial \gamma_i}{\partial z}] \gamma_j dV, \\ [M_k]_{ij} &= \int_{V_k} \gamma_i \gamma_j dV, \\ [U_{m,k,l}]_{ij} &= \int_{A_{k,l}} |\hat{\Omega}_m \cdot \hat{n}_{k,l}| \gamma_i \gamma_j dA. \end{aligned} \tag{5}$$

In fact, the integrations and differentiations in Eq.(5) are performed in a local barycentric coordinates in order to simplify the calculations and then they are transformed into the ones of global coordinates. This transformation needs the volumetric and surface Jacobians.

The angular domain is divided into azimuthal and polar directions. For the polar direction, the Gauss-Legendre quadrature sets[3] are used while the $(0, 2\pi)$ azimuthal angle is uniformly divided into a specified number of angles. A uniform weight is used for all the azimuthal angles. The total weight is the product of the polar and azimuthal weights. The order of the angular quadrature set is given by the number of total directions for one octant (i.e., $N_\phi \times N_\theta$). We denote the order of angular quadrature set by $C(N_\phi, N_\theta)$.

3. Computational Procedures

At present, MUST requires a file which contains the information of nodes, elements, and faces. This mesh information file is prepared by using the TetGen program. TetGen[4] is a free software which generates tetrahedral meshes and Delaunay tetrahedralizations. On the other hand, we are pursuing the generation of this mesh file by using CAD software such as SolidWorks. MUST calculates the connectivities between tetrahedral meshes and then determines the order of transport sweeping. The next procedure is to calculate the matrices given in Eq.(5) and the inversions of the matrices in the left hand side of Eq.(4) are performed. The final matrices to be stored in a binary file are the ones which are the products of the inverse matrices

mentioned above and the right hand side matrices in Eq.(4). These final matrices should be calculated and stored in the sweeping order.

The final procedure is to perform the transport sweeping for all the directions and for all the energy groups. During the transport sweeping, the pre-stored matrices are read from the binary file.

4. Numerical Test

To test the MUST code, we considered a one-group fixed source problem which consists of inner and outer cubes. The inner cube has a uniform fixed source of 2.0 neutrons/cm^{3}sec. This region is defined as [-20cm<*x*,*y*,*z*<20cm]. The outer cube is defined as the intersection of the outer region of the inner cube and the interior region of the cube [-40cm<*x*,*y*,*z*<40cm]. This outer region has no source. The inner source region has its macroscopic cross section of $\sigma = 1.0cm^{-1}$ and $\sigma_s = 0.7cm^{-1}$ while the outer region has $\sigma = 0.7cm^{-1}$ and $\sigma_s = 0.6cm^{-1}$. For this problem, we used the C(2,2) angular quadrature set. Fig. 1 shows the meshes generated by TetGen.

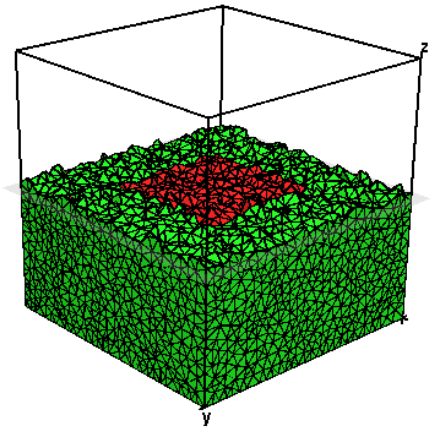

Fig. 1. Tetrahedral meshes for the test problem

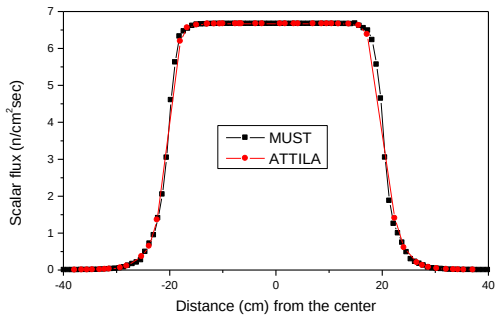

Fig. 2. Comparion of scalar flux along a line (y=0, z=0)

Fig. 2 compares the scalar flux distribution along a line (y=0, z=0) with that of the ATTILA code[5]. Basically, the MUST code uses the same DFEM method as the ATTILA code. However, their mesh divisions are different from each other for this case. As shown in Fig. 2, the results of MUST shows quite good agreements with ATTILA. Fig. 3 shows the 3-dimensional mapping of the scalar flux distribution obtained by MUST for the half of the problem domain.

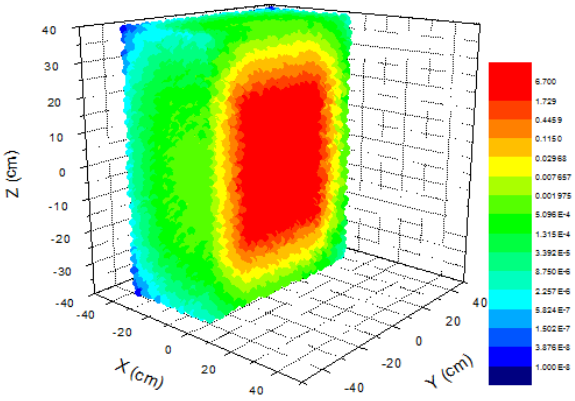

Fig. 3. 3D scalar flux mapping for half domain

Fig. 3 shows that the scalar flux distribution is physically reasonable.

5. Summary and Conclusion

In this paper, a transport theory code MUST for three-dimensional unstructured geometrical problems is introduced. A simple numerical test shows that MUST gives the solutions which have very good agreement with ATTILA. In the future, we will work to develop the new discretization method which may replace the DFEM method or may be used alternatively to DFEM. The other future works include the study of the acceleration schemes and the addition of time-dependent solver.

Acknowledgement

This work is supported by Agency for Defense Development (Contract No. UC080023GD).

REFERENCES

[1] T. A. Wareing, et al., "Discontinuous Finite Element SN Methods on Three-Dimensional Unstructured Grids," Nucl. Sci. Eng., Vol.138, p.256, 2001.
[2] M. L. Adams, "Discontinuous Finite Element Transport Solutions in Thick Diffusive Problems," Nucl. Sci. Eng., Vol.137, p.298, 2001.
[3] K. D. Lathrop and B. G. Carlson, "Discrete-Ordinates Angular Quadrature of the Neutron Transport Equation," LANL report LA-3186 (Feb. 1965).
[4] Hang Si, "TetGen : A Quality Tetrahedral Mesh Generator and Three-dimensional Delaunay Triangulator (Version 1.4 User's Manual)," Jan. 28, 2006.
[5] J. M. McGhee, T. A. Wareing, and D. A. Barnett, Jr., ATTILA User's Manual, Transpire, Inc., 15 January 2007.