Performance Evaluation of Moving Mesh Method with Higher-Order Numerical Scheme Applied to 1D Thermal-Hydraulic System Analysis Code

Won Woong Lee [a], Jeong Ik Lee [a*]
[a]Dept. of Nuclear and Quantum engineering, Korea Advanced Institute of Science and Technology
291 Daehak-ro, (373-1, Guseong-dong), Yuseong-gu, Daejeon 305-701, Republic of KOREA
Tel: +82-42-350-5869, Fax: +82-42-350-3810, Email: woong.aa@kaist.ac.kr, jeongiklee@kaist.ac.kr
[*]Corresponding author: jeongiklee@kaist.ac.kr

1. Introduction

In current, there are many nuclear system analysis code such as RELAP5, COBRA-TF, TRACE, TRAC, MARS and SPACE. These codes are using for regulation of nuclear systems by simulation of transient and steady state behavior of thermal-hydraulic systems. During the accident, the system pressure can fluctuate dramatically. So, there are many phenomena such as flashing, condensation and boiling. Therefore, two-phase model is implemented in these system codes. The one-dimensional conservation equations for mass, energy and momentum of the flow are solved by semi-implicit 1st order numerical scheme for space and time discretization. The 1st order numerical scheme is very robust and stable. But it is highly diffusive and less accurate. These characteristics are critical drawback in modeling the dramatically fluctuated situation like LOCA (Loss Of Coolant Accident).

There are two drawbacks in current nuclear system analysis code. First, the 1st order numerical scheme on the fixed grid can occur excessive numerical diffusion problem in simulation of accident condition due to the dramatic fluctuation. So, the prediction is less accurate and conservative than reality.

Second is very strict global requirement on the time step for the dramatic fluctuation. The time step is controlled by Courant number of explicit time integration schemes. Furthermore, the time step should be extremely small in order to reduce the error near the regions where the gradients should be high during the analysis. This results in inefficient computational cost. And even the code is dead.

Therefore, the 1st order numerical scheme on the fixed grid is not desirable during the analysis of accident conditions. So, the high predictive capability and efficient computational cost are required for the advanced nuclear system analysis code. For this study, the in-house code has been developed for application of the higher-order numerical schemes on 1D thermal-hydraulic system analysis code [11]. In this code, the moving mesh method is applied to compare the performance of the moving mesh method and the higher-order numerical schemes.

2. Methods and Results

For this study, MARS code will be used as the reference code. A single phase transient analysis code which is possible to calculate in the first-order and the higher-order scheme but mimics MARS solver is built in MATLAB environment. This code is called TWICE code (Transient Water system analysis code with ICE method) [11]. By using this code, this study will be conducted to evaluate performance of the moving mesh method in terms of the accuracy and computational cost. through a simple pipe flow simulation.

2.1 Governing Equations on Moving Mesh

The vast majority of numerical methods for solving hyperbolic problems like the governing equations of MARS code have been developed for fixed grids [1]. In the analysis of the nuclear system, the major challenge is to capture the sharp peak or the dramatic changes with sufficient accuracy while also keeping the efficient computational cost. Since these discontinuities are not stationary, it is attractive to allow the mesh points to move in time so that fine grid resolution can be maintained near discontinuities, thereby attaining a balance between accuracy and efficiency [1].

The governing equations of MARS and TWICE codes are typically indicated like eq. (1).

$$\frac{\partial f}{\partial t} + \frac{\partial (fu)}{\partial x} = S \tag{1}$$

where $f = \rho\psi$, ρ is density of fluid, $\psi = 1$ for the mass equation, $\psi = u$ (velocity) for the momentum equation, $\psi = e$ (internal energy) for the energy equation and S is source/sink terms of each equation. Let $0 \le \xi \le 1$ be the computational domain in which we have a fixed uniform grid with $\xi_i = (i-1)\Delta\xi$ for i=1,2, …, N+1, with $\Delta\xi = 1/N$. We also have a grid mapping $X(\xi, t)$ with the property that

$$x_i^n = X(\xi_i, t_n)$$

So, the differential equations (1) are transformed to an equation in (ξ, t) using a smooth mapping function $X(\xi, t)$ [2].

Let $\tilde{f}(\xi, t) \equiv f(X(\xi, t), t)$ and $\tilde{S}(\xi, \tilde{f}) = S(X(\xi, t), \tilde{f})$. Then we compute

$$(\tilde{f}u)_\xi = (fu)_x X_\xi \rightarrow (fu)_x = \frac{(\tilde{f}u)_\xi}{X_\xi}$$

$$\tilde{f}_t = f_t + X_t f_x = f_t + \frac{X_t \tilde{f}_\xi}{X_\xi} \rightarrow f_t = \tilde{f}_t - \frac{X_t \tilde{f}_\xi}{X_\xi}$$

Inserting these in eq. (1) and multiplying by X_ξ gives

$$X_\xi \tilde{f}_t + (\tilde{f}u)_\xi - X_t \tilde{f}_\xi = X_\xi \tilde{S} \quad (2)$$

We can put the left-hand side in conservation form by noting that

$$X_\xi \tilde{f}_t = (X_\xi \tilde{f})_t - X_{\xi t}\tilde{f}$$
$$X_t \tilde{f}_\xi = (X_t \tilde{f})_\xi - X_{t\xi}\tilde{f}$$

And so eq. (2) becomes

$$(X_\xi \tilde{f})_t + \left[(\tilde{f}u)_\xi - (X_t \tilde{f})_\xi\right] = X_\xi \tilde{S}$$
$$(X_\xi \tilde{f})_t + \left((u - X_t)\tilde{f}\right)_\xi = X_\xi \tilde{S} \quad (3)$$

So, eq. (3) can be discretized on the uniform computational grid in ξ (when S=0).

$$k_{i+1/2}^{n+1} f_{i+1/2}^{n+1} = k_{i+1/2}^{n} f_{i+1/2}^{n} - \frac{\Delta t}{\Delta \xi}\left[(u_{i+1}^{n+1} - \dot{x}_{i+1}^{n})\dot{f}_{i+1}^{n} - (u_i^{n+1} - \dot{x}_i^n)\dot{f}_i^n\right] \quad (4)$$

Where $k_{i+1/2}^n = X_\xi(\xi_i, t_n)$, $\dot{x}_i^n = X_t(\xi_i, t_n)$ and

$$\dot{f}_i^n = f_{i-1/2}^n + \phi(1 - \nu_i)\frac{f_{i+1/2}^n - f_{i-1/2}^n}{2} \quad \text{if } u_{i-\frac{1}{2}}^{n+1} \geq 0$$

$$= f_{i+1/2}^n - \phi(1 - \nu_i)\frac{f_{i+1/2}^n - f_{i-1/2}^n}{2} \quad \text{if } u_{i-1/2}^{n+1} \leq 0$$

where $\nu_{i+\frac{1}{2}} = \frac{u_{i+1/2}^{n+1}\Delta t}{\Delta x_{i+1/2}}$ is the Courant number. ϕ is determined by the numerical schemes as shown in Table I.

Table I. ϕ for the numerical schemes

Numerical scheme for the spatial	
1st order upwind scheme	ϕ=0
2nd order upwind scheme	ϕ=3, ν=0
Lax-Wendroff scheme	ϕ=1
Centered differencing scheme	ϕ=1, ν=0

We must rewrite eq. (4) using the observation.

$$\frac{\partial k}{\partial t} = \frac{\partial X_t}{\partial \xi}$$

This equation is discretized as below.

$$k_{i+1/2}^n f_{i+1/2}^n = k_{i+1/2}^{n+1} f_{i+1/2}^n - \frac{\Delta t}{\Delta \xi}(\dot{x}_{i+1}^n - \dot{x}_i^n) f_{i+\frac{1}{2}}^n$$

Inserting this in eq. (4) and rearranging gives

$$k_{i+1/2}^{n+1} f_{i+1/2}^{n+1} = k_{i+1/2}^{n+1} f_{i+1/2}^{n} - \frac{\Delta t}{\Delta \xi}\left[(u_{i+1}^{n+1} - \dot{x}_{i+1}^n)\dot{f}_{i+1}^n - (u_i^{n+1} - \dot{x}_i^n)\dot{f}_i^n + (\dot{x}_{i+1}^n - \dot{x}_i^n) f_{i+\frac{1}{2}}^n\right] \quad (5)$$

2.2 The Moving Mesh PDE

To determine the movement of mesh points, the moving mesh PDE approach by Huang et al. [1] is used.

$$\frac{\partial}{\partial \xi}\left(M \frac{\partial \dot{x}}{\partial \xi}\right) = -\frac{1}{\tau}\frac{\partial}{\partial \xi}\left(M \frac{\partial x}{\partial \xi}\right) \quad (6)$$

where M is the monitor function, τ is temporal smoothing parameter.

A commonly used form of the monitor function is the arclength monitor function [1].

$$M_{i+1/2} = \sqrt{1 + \frac{1}{\alpha}\left|\frac{\bar{Q}_{i+1} - \bar{Q}_i}{x_{i+1} - x_i}\right|^2}$$

Where α is the regularizing factor to avoid excessive concentration of the mesh points, $\bar{Q}_{i+1} = \frac{Q_{i+1/2}\Delta x_{i-1/2} + Q_{i-1/2}\Delta x_{i+1/2}}{\Delta x_{i+1/2} + \Delta x_{i-1/2}}$, $Q_{i+1/2}$ is a variable which determines the movement of mesh points.

The monitor function is largest where Q changes rapidly. So, eq. (6) serves to concentrate mesh points in regions where the gradient is high. However, to smooth the mesh, a regularized version $\widetilde{M}$ is used.

$$\widetilde{M}_{i+1/2} = \sqrt{\sum_{k=i-i_p}^{k=i+i_p} M_{k+\frac{1}{2}}^2 \left(\frac{\gamma}{1+\gamma}\right)^{|k-i|} \Big/ \sum_{k=i-i_p}^{k=i+i_p}\left(\frac{\gamma}{1+\gamma}\right)^{|k-i|}}$$

where γ and i_p are the spatial smooth factors. $\gamma = 2$ and $i_p = 4$ are recommended in [1].

So, eq. (6) is taken by Crank-Nicholson discretization like eq. (7).

$$\widetilde{M}_{i+\frac{1}{2}}^{n+1}(x_{i+1}^{n+1} - x_i^{n+1}) - \widetilde{M}_{i-\frac{1}{2}}^{n+1}(x_i^{n+1} - x_{i-1}^{n+1}) = \widetilde{M}_{i+\frac{1}{2}}^{n+1}(x_{i+1}^{n} - x_i^{n}) - \widetilde{M}_{i-\frac{1}{2}}^{n+1}(x_i^{n} - x_{i-1}^{n}) - \frac{\Delta t^n}{2\tau}(E_i^{n+1} + E_i^n) \quad (7)$$

Where E_i is a centered approximation to the term on the right hand side of eq. (6) given by

$$E_i = \widetilde{M}_{i+\frac{1}{2}}(x_{i+1} - x_i) - \widetilde{M}_{i-\frac{1}{2}}(x_i - x_{i-1})$$

So, the solutions are obtained by solving eq.(4) and eq.(7) in each time step. Fig. 1 shows the algorithm of TWICE code with the moving mesh method algorithm.

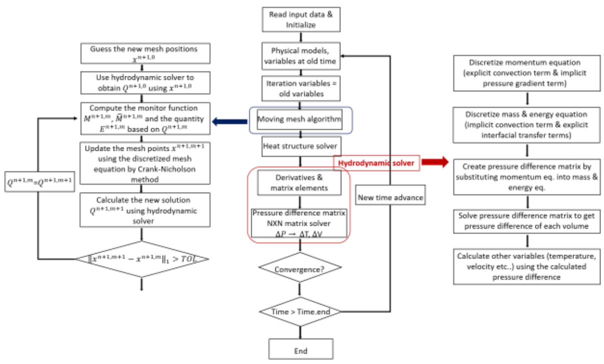

Fig. 1. Algorithm of TWICE code with the moving mesh method

2.3 Numerical Tests

A single phase pipe flow with a sine pulse of temperature is modeled by MARS and the TWICE codes separately and the results are compared to each other. Fig. 2 shows the configuration of single phase pipe flow with a sine pulse of temperature. In this test, the fluid flows at 1m/s through the pipe with cross sectional area of 0.5m^2 and 20m in length. The initial temperature and pressure of the fluid is 300K and 101,325Pa, respectively. The temperature of the injected fluid is changed with time as shown in Fig. 3. The pulse width is 5sec and the interval is 1.5 sec. This simulation is performed for several numbers of meshes to compare MARS with the TWICE code. A sensitivity test for other higher-order scheme is conducted. Table 1 shows the higher-order numerical schemes used for the sensitivity tests.

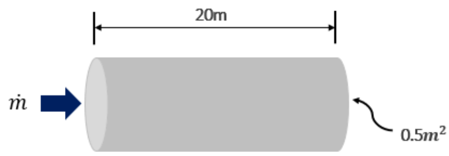

Fig. 2. Configuration of single phase pipe flow with sine pulse of temperature

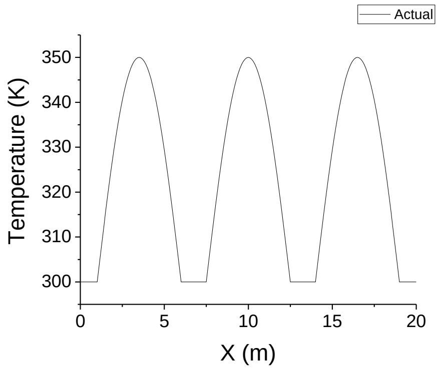

Fig. 3. Temperature profile of fluid injected at pipe inlet

Table I: Higher-order Numerical Schemes for Sensitivity Tests

Temporal scheme	Spatial scheme
1st order backward Euler scheme	1st order upwind scheme
	2nd order upwind scheme
2nd order backward Euler scheme	Centered differencing scheme
	Lax-Wendroff scheme

2.3 Results

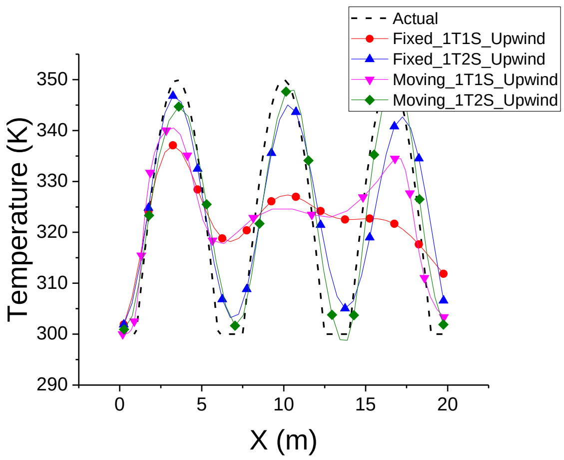

Fig. 4. Comparison of temperature profile for 1st order upwind & 2nd upwind scheme on the fixed mesh and moving mesh

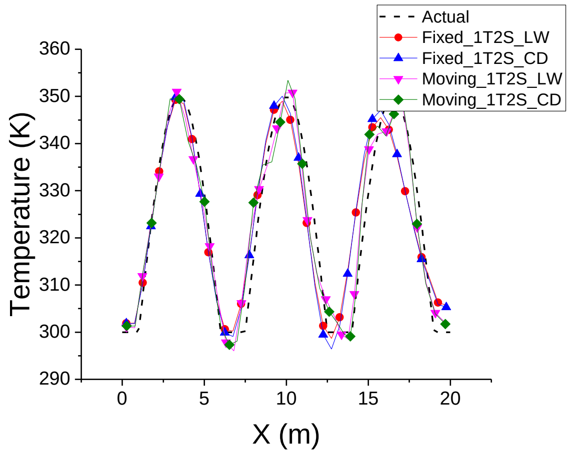

Fig. 5. Comparison of temperature profile for 2nd order Lax-Wendroff (LW) & 2nd centered differencing (CD) scheme on the fixed mesh and moving mesh

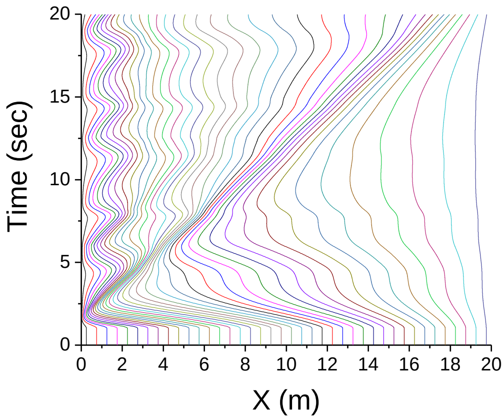

Fig. 6. Mesh movement along the time for 1st order upwind scheme on the moving mesh

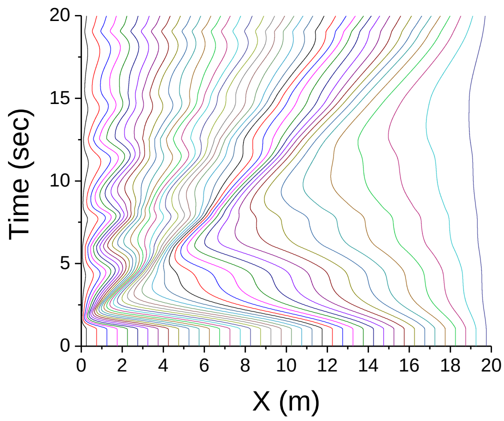

Fig. 7. Mesh movement along the time for 2nd order upwind scheme on the moving mesh

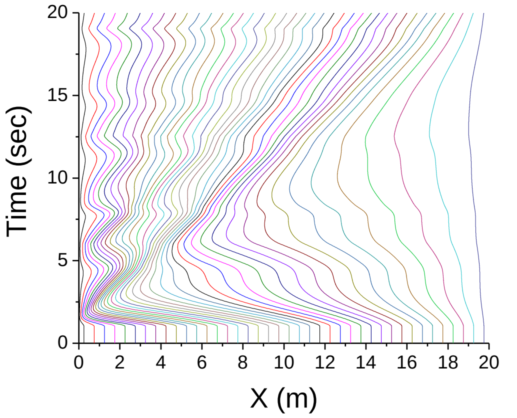

Fig. 8. Mesh movement along the time for 2nd order Lax-Wendroff scheme on the moving mesh

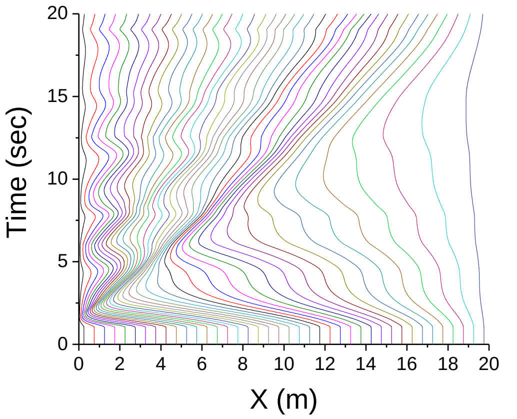

Fig. 9. Mesh movement along the time for 2nd order centered differencing scheme on the moving mesh

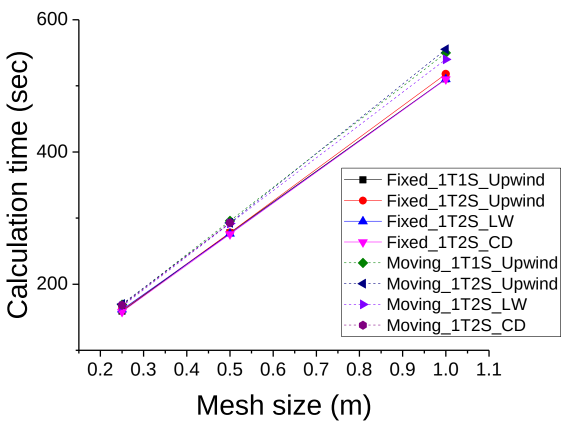

Fig. 10. Comparison of calculation time for the moving mesh and fixed mesh with the higher-order numerical schemes

3. Conclusions

This study evaluated the performance of the moving mesh method with the higher-order numerical schemes for the next generation nuclear system analysis code. The accuracy is slightly improved in the moving mesh than the fixed mesh. The results of the mesh movement show that the mesh points move depending on the propagation of the temperature pulse along time. The calculation time on the moving mesh is not much different with the fixed mesh. Also, there is no difference between the higher-order numerical schemes on the calculation time. Since the time step control on the fixed mesh is not carried out, the moving mesh method applied to the nuclear system analysis code has the possibility of improvement for the calculation efficiency. The more detail discussion will be presented during the conference.

REFERENCES

[1] John M. Stockie et al., "A moving mesh method for one-dimensional hyperbolic conservation laws", SIAM Journal on Scientific Computing, Vol.22, p. 1791-1813,

[2] R. Fazio et al., "Moving-mesh methods for one-dimensional hyperbolic problems using CLAWPACK", Computers and Mathematics with Applications, Vol.45, 273-298, 2003

[3] Randall J. Le Veque, " Wave Propagation Algorithms for Multidimensional Hyperbolic Systems", Journal of Computational Physics, Vol. 131, 327-353, 1997

[4] J.H. Mahaffy, "Numerics of codes: stability, diffusion, and convergence", Nuclear Engineering and Design, 145(131-145), 1993

[5] D.R. Liles, Wm. H. Reed, "A Semi-implicit Method for Two-Phase Fluid Dynamics", Journal of Compuational Physics, 26(390~407), 1978

[6] Jae-Jun Jeong, Kwi Seok Ha, Bub Dong Chung, and Won Jae Lee, "A Multi-Dimensional Thermal-Hydraulic System Analysis Code, MARS 1.3.1", Journal of the Korean Nuclear Society, Volume 31, Number 3, pp. 344~363, 1999

[7] H.K.Cho, H.D. Lee, I.K. Park, J.J. Jeong, "Implementation of a second-order upwind method in a semi-implicit two-phase flow code on unstructured meshes", 37(606~614), 2010

[8] H.K. Versteeg, W. Malalasekera, An introduction to computational fluid dynamics, Longman Scientific & Technical, pp. 103~133, 1995

[9] Ercilia Sousa et. al., "On the influence of numerical boundary conditions", Applied Numerical Methematics, 41(325-344), 2002

[10] Ercilia Sousa, "High-order methods and numerical boundary conditions", Computer Methods in Applied Mechanics and engineering, 196(4444-4457), 2007

[11] Won Woong Lee, Jeong Ik Lee, "Performance Comparison of Higher-Order Numerical Schemes Applied to 1D Thermal-Hydraulic System Analysis Code", 2017 Spring KNS