Applied Mathematical Sciences, Vol. 6, 2012, no. 126, 6287 - 6299

Meshless Methods for a Simple Atmospheric Model

A. Koomsubsiri [*] **and D. Sukawat** [†]

Department of Mathematics
Kingmongkut's University of Technology Thonburi (KMUTT)
Bangkok, 10140, Thailand

S. Tangmanee

Centre of Excellence in Mathematics (CEM)
Faculty of Science, Mahidol University
Bangkok, 10400, Thailand

Abstract

In this paper, two meshless methods, the "Smoothed Particle Hydrodynamics (SPH)", the "Meshless Local Petrov-Galerkin (MLPG)" and the classical upwind finite difference method (FDM) are applied to a linear one space dimension advection equation with specified periodic boundary condition. In SPH, a cubic spline weight function is used to determine the advection field at the nodes in the support domain. For MLPG method, the weight residual is confined to a very small local sub-domain. The numerical integrations are carried out over a local quadrature domain defined for the node, which can also be the local domain where the test (weight) function is defined. The results from three methods are compared with the corresponding exact solutions.

Keywords: Smoothed Particle Hydrodynamics, Meshless Local Petrov-Galerkin, Advection equation

1 INTRODUCTION

Numerical method is important for the successful simulation of physical processes as the underlying partial differential equation usually has no analytic solution and

* Corresponding author: koomsubsiri@gmail.com
† Corresponding author: dusadee.suk@kmutt.ac.th

has to be approximated. Conventional numerical methods need a priori definition of the connectivity of the nodes, i.e., they rely on a mesh. Finite Element Method (FEM), Finite Difference Method (FDM) and Finite Volume Method (FVM) may be the most well-known members of these thoroughly developed mesh-based methods. The large deformations in highly nonlinear problems that can deteriorate the accuracy because of mesh or element distortion may cause severe loss of accuracy or even complete failure of computations. The mesh-based methods are unsuitable for finding solutions of problems with changing domain shape. A new class of numerical methods has been developed which approximates partial differential equations based on only a set of nodes without the need for an additional mesh. This is called meshless methods. That is different from FEM because a mesh of element is not used in meshless method, the field variable u at any point $\mathbf{x}$ within the problem domain is interpolated using the displacements at its nodes within the support domain of the point at $\mathbf{x}$, i.e.

$$u(\mathbf{x}) = \sum_{j=1}^{n} \varphi_j(\mathbf{x}) u_j = \Phi(\mathbf{x}) U_s \tag{1}$$

for $\mathbf{x} = (x_1, x_2, x_3, ..., x_n)$, where n is the number of nodes include in a "small local domain" of the point at $\mathbf{x}$. The local domain means the interpolation area which is represented by point $\mathbf{x}$, u_j is the nodal field variable at the j th node in the small local domain, U_s is the vector that collects all the field variables at these nodes, and $\varphi_j(\mathbf{x})$ is the shape function of the j th node determined using the nodes that are included in the small domain of $\mathbf{x}$. This small local domain is called the support domain of $\mathbf{x}$. A number of meshless methods have been developed by different authors, such as Smooth Particle Hydrodynamics (SPH) [Lucy (1977)], Diffuse Element Method (DEM) [Naroles, Touzot, and Villon (1992)], Element Free Galerkin method (EFG) [Belytcshko, Lu, and Gu (1994)] and Meshless Local Petrov-Galerkin method (MLPG) [Atluri and Zhu (1998)].

In addition to the above methods, there are many methods that are meshless such as the Finite Point Method (FPM), Point Interpolation Method (PIM), Generalized Finite Difference Method (GFDM), Radial Basis Functions (RBF), Vortex Method (VM), and HP Clouds Method, etc.

2 MATERIALS AND METHODS

2.1 Smoothed Particle Hydrodynamics (SPH)

In the SPH method, the equation to be approximated is represented by a collection of nodes which are under the influence of hydrodynamic. Each node is a representation of surrounding particles in the support domain. The interpolation of support domain is obtained through the summation of the product of multiplying

the function with its weight function. The weight function is constructed using the information on all particles within the support domain, and the support domain size is defined by the smoothing length. The cubic spline is used to define the weight function in this research. The reason is that, with cubic spline weight function interpolation derivative of a function is transformed into a simple form of derivative of the weight function. That is, for a function $u(x)$[1][2],

$$u^h(x) \approx \sum_{j=1}^{N} \varphi_j(x) u_j \tag{2}$$

where $\varphi_j(x) = \widehat{W}(x - x_j, h)\Delta V_j$, $\widehat{W}(x - x_j, h)$ is a weight or smoothing function, h is the smoothing length and ΔV_j is the volume element carried by particle j.

The functions φ_j are the SPH shape functions of the approximation and the cubic spline weight function is used in the SPH method for this study. The derivative can be approximated from (1) as

$$\frac{\partial u^h(x)}{\partial x} \approx \sum_{j=1}^{N} u_j \frac{\partial \widehat{W}(x - x_j, h)}{\partial x} \Delta V_j \tag{3}$$

A summation representation is valid and converges when the weight function satisfies certain conditions[3].

1. $\widehat{W}(x - x_j, h) > 0$ over Ω (Positivity)
2. $\widehat{W}(x - x_j, h) = 0$ outside Ω (Compact)
3. $\int_{\Omega} \widehat{W}(x - x_j, h) dx_j = 1$ (Unity)
4. $\widehat{W}$ is monotonically decreasing function (Decay)
5. $\widehat{W}(x, h) \to \delta(x)$ *as* $h \to 0$ (Delta function)

2.2 Meshless Local Petrov-Galerkin (MLPG)

In the MLPG method, the problem domain is represented by a set of arbitrarily distributed nodes. The weighted residual method is used to create the discrete system equation. The weighted residual method is in integral form, and a background mesh of cells is still required for the integration. However, the weight residual in MLPG method is confined to a very small local sub-domain of a node. This means that the weak form is satisfied at each node in the problem domain in a local integral sense. Therefore, the weak form is integrated over a "local quadrature domain" that is independent of other domains of other nodes. The

moving least-square (MLS) is the basis for many meshless methods because it is generally considered to be one of the best schemes to interpolate data with a reasonable accuracy. MLS approximation is based on three components: a weight function of compact support associated with each node, polynomial basis functions and a set of coefficients that depend on the position **x** of the point. The approximation $u^h(\mathbf{x})$ of a function of the field variable $u(\mathbf{x})$ at any point **x** in the domain (Ω) of computation is expressed by[4]

$$u^h(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x}) a_j(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{a}(\mathbf{x}) \tag{4}$$

where *m* is the number of terms of monomials (polynomial basis), and a vector of coefficients $\mathbf{a}(\mathbf{x})$ is determined using the function values at a set of nodes that are included in the support domain of **x**. $\mathbf{p}(\mathbf{x})$ is a vector of basis function. In 1D space, a complete polynomial basis of order *m* is given by

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x}) = \{ p_0(x) \quad p_1(x) \quad \ldots \quad p_m(x) \} = \{ 1, \; x, \; x^2, \; \ldots, \; x^m \} \tag{4.1}$$

Note that the coefficient vector $\mathbf{a}(\mathbf{x})$ in Eq. (4) is a function of **x**. A functional of weighted residual is constructed using the approximated values of the field function and the nodal parameters, $u_i = u(\mathbf{x}_i)$

$$J = \sum_{i=1}^{n} \widehat{W}(\mathbf{x} - \mathbf{x}_i) \left[\mathbf{p}^{\mathrm{T}}(\mathbf{x}_i)\mathbf{a}(\mathbf{x}) - u_i \right]^2 \tag{4.2}$$

where *n* is the number of nodes in the support domain of **x** for which the weight function $\widehat{W}(\mathbf{x} - \mathbf{x}_i) \neq 0$. In MLS approximation, $\mathbf{a}(\mathbf{x})$ is chosen to minimize the weighted residual. The minimization condition requires

$$\frac{\partial J}{\partial \mathbf{a}} = 0 \tag{4.3}$$

which leads to the following set of linear relations

$$\mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\mathbf{U}_s \tag{4.4}$$

$\mathbf{A}(\mathbf{x})$ is called the weighted moment matrix defined by

$$\mathbf{A}(\mathbf{x}) = \sum_{i=1}^{n} \widehat{W}_i(\mathbf{x}) \mathbf{p}(\mathbf{x}_i) \mathbf{p}^{\mathrm{T}}(\mathbf{x}_i) \tag{4.5}$$

where $\widehat{W}_i(\mathbf{x}) = \widehat{W}(\mathbf{x} - \mathbf{x}_i)$.

The matrix **B** in Eq. (4.4) has the form of

$$\mathbf{B}(\mathbf{x}) = [\widehat{W}_1(\mathbf{x})\mathbf{p}(\mathbf{x}_1) \quad \widehat{W}_2(\mathbf{x})\mathbf{p}(\mathbf{x}_2) \quad \ldots \quad \widehat{W}_n(\mathbf{x})\mathbf{p}(\mathbf{x}_n)] \tag{4.6}$$

and $\mathbf{U}_s$ is the vector that collects the nodal parameters of the field variables for all the nodes in the support domain

$$\mathbf{U}_s = \{ u_1 \quad u_2 \quad \ldots \quad u_n \}^{\mathrm{T}} \tag{4.7}$$

Solving Eq. (4.4) for $\mathbf{a}(\mathbf{x})$,

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\mathbf{U}_s \tag{4.8}$$

Substituting the above equation back into Eq. (4) leads to

$$u^h(\mathbf{x}) = \sum_{i=1}^{n} \phi_i(\mathbf{x}) u_i \;=\; \mathbf{\Phi}^{\mathrm{T}}(\mathbf{x})\mathbf{U}_s \tag{4.9}$$

where $\mathbf{\Phi}(\mathbf{x})$ is the vector of MLS shape functions corresponding *n* nodes in the support domain of the point **x**. The shape function $\phi_i(\mathbf{x})$ for the *i*th node is defined by

$$\phi_i(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x})\left(\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\right)_{ji} \;=\; \mathbf{p}^{\mathrm{T}}(\mathbf{x})\left(\mathbf{A}^{-1}\mathbf{B}\right)_i \tag{4.10}$$

The weight functions which will be used for SPH and MLPG approximation are a cubic spline weight function [4]:

$$\widehat{W}(S) = \begin{cases} \dfrac{2}{3} - 4S^2 + 4S^3 & ,\ S \le \dfrac{1}{2} \\ \dfrac{4}{3} - 4S + 4S^2 - \dfrac{4}{3}S^3 & ,\ \dfrac{1}{2} < S \le 1 \\ 0 & ,\ S > 1 \end{cases} \tag{5}$$

where $S = \dfrac{d_c}{d_s}$

d_c is the nodal spacing near the point at **x** and d_s is the size of the support domain for the weight function (in SPH, d_s is the smoothing length *h*).

2.3 The Linear Advection Equation

Many physically realistic problems in atmospheric model actually involve some amount of diffusion. The advection-diffusion equation is an unsteady state flow problem. In general, one-dimensional linear advection with constant diffusion takes the form [5]

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} - D\frac{\partial^2 u}{\partial x^2} = 0 \tag{6}$$

where $u = u(x, t)$ is the advection quantity, c is the advection speed and and D is the diffusion coefficient. Eq. (6) can be simplified to be a linear advection equation by ignore the diffusion term. The advection equation describes the motion of some conserved quantity along a constant velocity field. This is the simplest conservation law, the one-dimensional linear advection equation is given by the expressions

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \tag{6a}$$

In this test problem, the initial condition of Eq. (6a) is

$$u(x, 0) = \sin(kx), \tag{6b}$$

which gives the exact solution,

$$u(x,t) = \sin\left(k\left(x - ct\right)\right) \tag{6c}$$

2.4 The Linear Advection Equation in SPH form

In the SPH method, Eq. (3), the derivative of u can be written as

$$\frac{\partial u_i}{\partial x} = \sum_{j=1}^{N} u_j \frac{\partial \widehat{W}_{ij}(S)}{\partial x} \Delta V_j \tag{7}$$

Such that $\dfrac{\partial \widehat{W}_{ij}(S)}{\partial x} = \dfrac{x_j - x_i}{\left|x_j - x_i\right|} \dfrac{\partial \widehat{W}_{ij}(S)}{\partial x_i}$.

In one dimensional space, ΔV_j for interior nodes is $\Delta V_j = \left(x_{j+1} - x_{j-1}\right)/2$. For node j on the left boundary, $\Delta V_j = \left(x_{j+1} - x_j\right)/2$, with a similar expression on the right.

The forward time difference is used for time derivative[6],

$$\frac{\partial u}{\partial t} \approx \frac{u^{n+1} - u^n}{\Delta t} \tag{7.1}$$

The time step is controlled in the computation to satisfy the following Courant-Friedrichs-Lewy (CFL) condition: $\Delta t \le \dfrac{\Delta x}{c}$.

From Eq. (7) and (7.1), the advection equation (6a) can be written in the form of SPH method as,

$$u_i^{n+1} = u_i^n - c\Delta t \left[\sum_{j=1}^{N} u_j^n \frac{\partial \widehat{W}_{ij}(S)}{\partial x} \Delta V_j \right] \tag{7.2}$$

This straightforward approximation is usually not accurate, and often destroys the conservation property of the associated continuous system. Nonetheless, when the approximation is combined with an additional term which contains a null expression, it may produce better results [1]. A null expression term is the term that approximate the derivative at the node by using weight function,

$$\sum_{j=1}^{N} u_i \frac{\partial \widehat{W}_{ij}(S)}{\partial x} \Delta V_j = u_i \frac{\partial}{\partial x}\left(\sum_{j=1}^{N} \widehat{W}_{ij}(S) \Delta V_j \right) = u_i \frac{\partial (1)}{\partial x} = 0$$

Since $\widehat{W}_{ij}$ is a partition of unity, the better results are obtained. Thus, approximation of the derivative function using function as in Eq. (7) with a null expression term is then,

$$\frac{\partial u_i}{\partial x} = \sum_{j=1}^{N} u_j \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j - \sum_{j=1}^{N} u_i \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j$$
$$= \sum_{j=1}^{N} u_j \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j - u_i \sum_{j=1}^{N} \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j$$
$$= \sum_{j=1}^{N} \left(u_j - u_i\right) \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j$$

Hence, Eq. (7.2) can be written as

$$u_i^{n+1} = u_i^n - c\Delta t \sum_{j=1}^{N} \left(u_j^n - u_i^n\right) \frac{\partial \widehat{W}_{ij}(S)}{\partial x}\Delta V_j \tag{7.3}$$

2.5 The Linear Advection Equation in MLPG form

The partial derivatives of displacement u in Eq. (6a) can be transformed to the local weak form using the weight residual with the integral by part and the divergence theorem as[7]

$$\int_{\Omega_s} w\frac{\partial u}{\partial t} d\Omega + \int_{L_s} wcu\mathbf{n}d\Gamma + \int_{\Gamma_{su}} wcu\mathbf{n}d\Gamma - \int_{\Omega_s} \frac{\partial w}{\partial x} cud\Omega = 0 \tag{8}$$

where u is the trial function, w is the test function, Γ_{su} is a part of the boundary $\partial\Omega_s$ of local sub-domain $\Omega_s\left(\in\Omega\right)$ over which the essential boundary conditions are specified and L_s being the other part of the local boundary over which no boundary condition is specified. In Eq. (8), the continuity requirements on trial and test function are not the same, the formulation is called "local unsymmetric weak formulation" and denote as LUSWF1. In order to simplify the above equation by choosing the cubic spline weight function in Eq. (5) as also the test function, with the radius of the support of the weight function being replaced by the radius of the local sub-domain Ω_s, such that the test function vanishes over the local boundary L_s. Then, Eq. (8) can be generated the discretized system equations in the matrix form as

$$\int_{\Omega_s} \mathbf{W}_I \mathbf{\Phi}\mathbf{u}^t d\Omega + \int_{\Gamma_{su}} \mathbf{W}_I \mathbf{cn}\mathbf{\Phi}\mathbf{u}d\Gamma - \int_{\Omega_s} \bar{\mathbf{W}}_I \mathbf{c}\mathbf{\Phi}\mathbf{u}d\Omega = 0 \tag{8.1}$$

Such that

$$\frac{\partial \mathbf{u}^h}{\partial t} = \begin{bmatrix}\phi_1 & \phi_2 & \cdots & \phi_n\end{bmatrix} \begin{Bmatrix} u_1' \\ u_2' \\ \vdots \\ u_n' \end{Bmatrix} = \mathbf{\Phi}_{1\times n}\mathbf{u}'_{n\times 1} \tag{8.2}$$

Rearranging and rewriting Eq. (8.1) into the new form as

$$\mathbf{C}\mathbf{u}' = \mathbf{K}\mathbf{u} \tag{8.3}$$

where the matrix $\mathbf{C}$ contains the time derivative terms

$$\mathbf{C} = \int_{\Omega_s} \mathbf{W}_I \mathbf{\Phi} d\Omega \tag{8.4}$$

the stiffness matrix $\mathbf{K}$ contains the terms with spatial derivatives

$$\mathbf{K} = \int_{\Omega_s} \mathbf{c}\bar{\mathbf{W}}_I \mathbf{\Phi} d\Omega - \int_{\Gamma_{su}} \mathbf{c}\mathbf{W}_I \mathbf{n}\mathbf{\Phi} d\Gamma \tag{8.5}$$

Time is best discretized using the Crank-Nicolson scheme, which replaces the time derivative at half-step $u'\left(t+\frac{\Delta t}{2}\right)$ with the central difference approximation[8]

$$u'\left(t+\frac{\Delta t}{2}\right) = \frac{u\left(t+\Delta t\right)-u\left(t\right)}{\Delta t} \tag{8.6}$$

The term $u\left(t+\frac{\Delta t}{2}\right)$ will be approximated by using the average of $u(t)$ and $u\left(t+\Delta t\right)$.

This scheme is unconditionally stable. Therefore, the time-discretized in the matrix form of the linear system equation, Eq. (8.3), is obtained as

$$\mathbf{C}\left(\frac{\mathbf{u}^{t+\Delta t}-\mathbf{u}^{t}}{\Delta t}\right) = \mathbf{K}\left(\frac{\mathbf{u}^{t+\Delta t}+\mathbf{u}^{t}}{2}\right) \tag{8.7}$$

where $\mathbf{u}^{t+\Delta t}$ represent the matrix of $u\left(t+\Delta t\right)$ and $\mathbf{u}^{t}$ is the matrix of $u\left(t\right)$.

The final system that has to be solved at every time-step has the form

$$(2\mathbf{C}-\Delta t\mathbf{K})\mathbf{u}^{t+\Delta t} = (2\mathbf{C}+\Delta t\mathbf{K})\mathbf{u}^{t} \tag{8.8}$$

If the advection equation in Eq. (6a) is transformed to a local weak form without using the weight residual and the divergence theorem, then another local unsymmetric weak formulation can be written as

$$\int_{\Omega_s} w\frac{\partial u}{\partial t} d\Omega + \int_{\Omega_s} w\left(c\frac{\partial u}{\partial x}\right) d\Omega = 0 \tag{8.9}$$

Eq. (8.9), is denoted as LUSWF2, can be generated the discretized system equations in a matrix form as

$$\int_{\Omega_s} \mathbf{W}_I \mathbf{\Phi}\mathbf{u}' d\Omega + \int_{\Omega_s} \mathbf{c}\mathbf{W}_I \bar{\mathbf{\Phi}}\mathbf{u} d\Omega = 0 \tag{8.10}$$

Rearranging Eq. (8.10) into the matrix form of Eq. (8.3), the new matrix $\mathbf{K}$ is

$$\mathbf{K} = -\int_{\Omega_s} \mathbf{c}\mathbf{W}_I \bar{\mathbf{\Phi}} d\Omega \tag{8.11}$$

The solution from LUSWF2 can be obtained by repeating the procedure in Eq. (8.8) with **K** from Eq. (8.11).

2.6 The Linear Advection Equation in FDM form

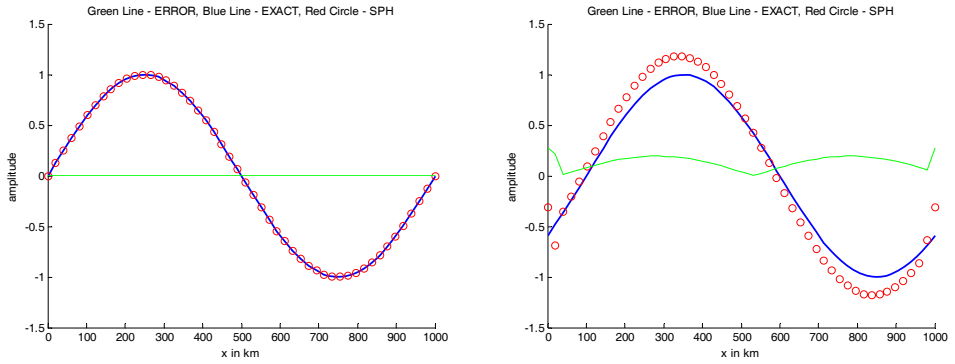
From using the forward upwind scheme (FU), the spatial upwind discretization will be applied in Eq. (6a). In order to obtain a stable explicit method as follow[6]

$$\frac{u_m^{n+1} - u_m^n}{\Delta t} = -c\frac{u_m^n - u_{m-1}^n}{\Delta x} \tag{9}$$

This method is explicit and conditionally stable. Δt must be chosen sufficiently small in order to guarantee stability that is CFL condition.

3 NUMERICAL RESULTS

The linear advection equation, Eq.(6a), is solved by SPH formulation with the initial conditions as : $k = 2\pi/Lx$, $Lx = 1000$ ($x \in [0, 1000]$), and the advection speed is $c = 10$. Eq. (6a) is approximated from $t = 0$ to $t = 10\,\Delta t$ with the time step of $\Delta t = 0.01$. The boundary condition is periodic. From Figure 1, the advection of sine wave moves from left hand side to right hand side. The result from SPH method is compared with the exact solution where the blue line is the exact solution, the red circle is the SPH result, and the green line represents the approximation error. In the figure, when $t = 10\,\Delta t$, the large error occurred near the boundary due to asymmetry of weight function. The main cause of error in this approximation is the accumulate error on the boundary in every time step.

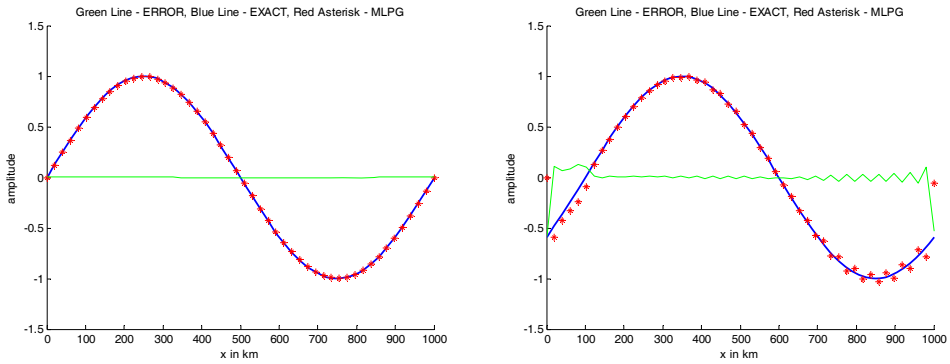


$t = 0$ $t = 10\,\Delta t$

Figure 1 Numerical simulations from SPH method at $t = 0$ and $10\,\Delta t$

Next, the MLPG formations are used to solve the linear advection equation with the same initial conditions. First formation, LUSWF1, is used to approximate the

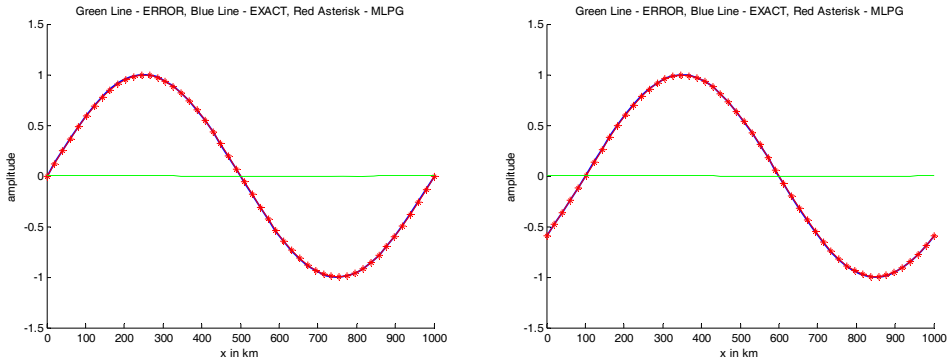
moving of sine wave and the simulation is shown in Figure 2. The red asterisk represents the MLPG solution. From the figure, at the initial time ($t = 0$), the result is similar to the exact solution. At the later time, when $t = 10\,\Delta t$, the interior domain, there is no significant difference in amplitude between the MLPG solution and the exact solution. The boundary of sine wave had a complex pattern. The problem may come from the boundary integral term in Eq. (8.5), which is not satisfied with the periodic boundary condition.



$t = 0$ $t = 10\,\Delta t$

Figure 2 Numerical simulations from MLPG with LUSWF1 at $t = 0$ and $10\,\Delta t$

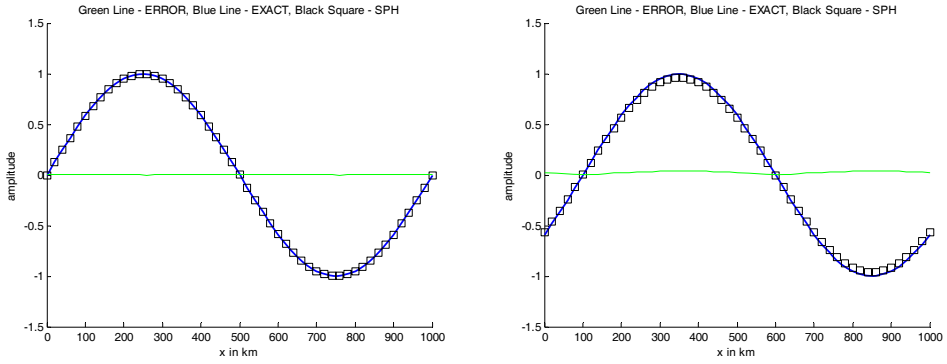
Repeating the procedure in Eq. (8.8) using second formation, LUSWF2, Figure 3 shows the numerical simulations from the MLPG approximation with LUSWF2. The efficiency and the accuracy are very good. When $t = 10\,\Delta t$, the sine wave on boundary are smooth curve fits. Thus the periodic boundary condition in this problem is suitable to the LUSWF without the boundary integral term.



$t = 0$ $t = 10\,\Delta t$

Figure 3 Numerical simulations from MLPG with LUSWF2 at $t = 0$ and $10\,\Delta t$

The numerical simulation from FDM is shown in Figure 4. The black square is the FDM approximation. That gives good results in every time steps.



$t = 0$ $t = 10\,\Delta t$

Figure 4 Numerical simulations from FDM at $t = 0$ and $10\,\Delta t$

There have been very few formal studies of the accuracy of SPH carried out. The example, this may involve measuring the error in the final result of an SPH calculation at time $t = 10\,\Delta t$ in terms of a mean absolute error can be defined as[9]

$$\text{Mean error} = \frac{1}{N}\sum_i \left|u^h - u\right| \tag{10}$$

where N is the number of nodes, u^h is the numerical solution and u is the analytic solution. Therefore, Eq. (10) is used to study the effect of increasing number of nodes in mesh and meshless approximations, an experiment is performed using 10, 80 and 150 nodes. It is found that increasing number of nodes results in better approximation, as shown in Figure 5. The red asterisk, blue circle and the black square represent the MLPG with LUSWF2, SPH and the FDM approximations, respectively. The results from MLPG with LUSWF2 give the better solution of approximation in this problem. Considering the slope of SPH curve, that is less than the slope of MLPG and FDM curve. This means the rate of decreasing in the error is slow. The curve of MLPG and FDM are show the similar rate. Table 1 shows the mean absolute error of approximation in different node quantities. For clearly, the MLPG approximation provided the least error.

Table 1 Mean absolute error of mesh and meshless approximations at $t = 10\,\Delta t$

Number of nodes	**Mean absolute error**		
	SPH	**MLPG**	**FDM**
10	0.1611	0.0810	0.1269
80	0.1332	0.0041	0.0156
150	0.1292	0.0040	0.0083

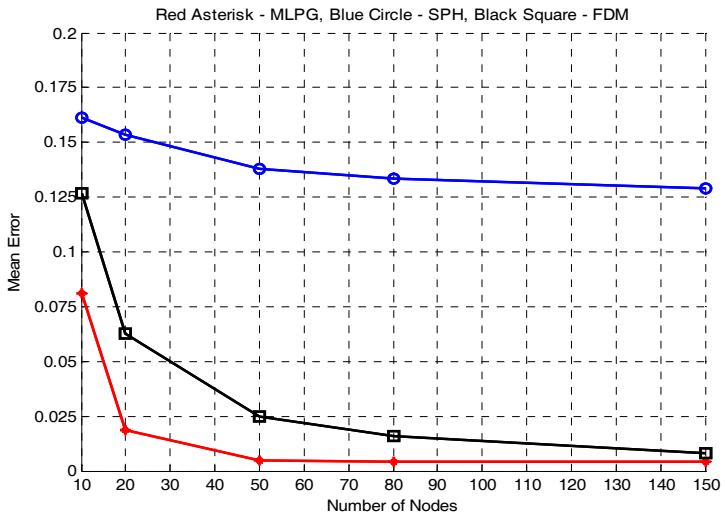


Figure 5 A comparison of the rate of decreasing in the error.

The accuracy of the meshless approximation of the advection equation is also depends on the number of node used in computation. If the number of node is increased, the accuracy will increase but the time for computing will also increase.

4 SUMMARY

Smooth particle hydrodynamics (SPH) is a meshfree method, based on the strong form. With SPH method, the equation to be approximated is represented by a collection of nodes which are under the influence of hydrodynamic. Application of the SPH method to the linear advection equation with the cubic spline weight function shows that the error of approximation is mainly due to accumulate error of boundary approximation in each time step. Thus, the weak point of approximation is the asymmetric weight function on the boundary. Meshless Local Petrov-Galerkin (MLPG) is a meshless method based on the idea of the local weak form which eliminates the need of the background cell and, consequently, performs the numerical integration in a meshless sense. In this approach, the global set of equations is derived by writing the weak forms over the small local sub-domains defined around the nodes used for the discretization. Consequently, the use of the global mesh of background cells or elements is not necessary throughout the solution procedure. Application of the MLPG method to the linear advection equation, the cubic spline test function shows that the error of approximation occur on the boundary. However, the local unsymmetric weak formulation without a boundary integral term showed a good approximation for the problems with periodic boundary conditions.

Nonetheless, in the comparative study of numerical error, the accuracy of the approximations are also depend on the number of nodes used in computation. It is apparent that both meshless methods and FDM potentially offer similar results. Therefore, the proposed meshless scheme can be realized effectively as one of alternative methods for solving physical problems on the complex domains.

ACKNOWLEDGEMENTS

This research was supported by the Research Fund from Thai-Nichi Institute of Technology (TNI) and the Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT).

References

[1] S. Li, and W.K. Liu, Meshfree Particle Methods, Springer, New York, 2004.

[2] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming and P. Krysl, MeshlessMethods: An Overview and Recent Developments, Computer Methods in Applied Mechanics and Engineering, **139** (1996), 3-47.

[3] G.R. Liu, Meshfree Methods Moving beyond the Finite Element Method, CRC Press, Boca Raton, 2003.

[4] G.R. Liu and Y.T. Gu, An Introduction to Meshfree Methods and Their Programming, Springer-Verlag, New York, 2005.

[5] R.H. James and R.H. Thomas, An Introduction to Dynamic Meteorology, Elsevier Inc, USA, 2004.

[6] G.D. Smith, Numerical Solution of Partial Differential Equations, Clarendon Press, United States, 1985.

[7] S.N. Atluri and T. Zhu, A new Meshless Local Petrov-Galerkin (MLPG) approach in computational mechanics, Computational Mechanics, **22** (1998), 117-127.

[8] M. Sterk, B. Robic and R. Trobec, Mesh Free Method Applied to the Diffusion Equation, Parallel numerics'05: Theory and Applications, (2005), 57-66

[9] V. Springel, Smoothed Particle Hydrodynamics in Astrophysics, Annual Review of Astronomy and Astrophysics, **48** (2010), 391-430.

Received: August, 2012