Journal of the Serbian Society for Computational Mechanics / Vol. 12 / No. 2, 2018 / pp 128-152
(DOI: 10.24874/jsscm.2018.12.02.09)

Development of the Meshless Local Petrov-Galerkin Method to Analyze Three-Dimensional Transient Incompressible Laminar Fluid Flow

M. J. Mahmoodabadi*, F. Mahmoodabadi, M. Atashafrooz

Department of Mechanical Engineering, Sirjan University of Technology, Sirjan, Iran
E-mails: mahmoodabadi@sirjantech.ac.ir

**corresponding author*

Abstract

In this paper, a numerical algorithm is presented to simulate the three-dimensional transient incompressible flow using a meshless local Petrov-Galerkin (MLPG) method. In the proposed algorithm, the forward finite difference (FFD) and MLPG methods are employed for discretization of time derivatives and solving the Poisson equation of the pressure, respectively. The moving least-square (MLS) approximation is considered for interpolation, while the Gaussian weight function is used as a test function. Furthermore, the penalty approach is applied to satisfy the boundary conditions. Moreover, in two examples, the accuracy and efficiency of this approach is compared with the exact solutions.

Keywords: Meshless Local Petrov-Galerkin; Forward finite difference; Moving least Squares; Gaussian weight function; Three-dimensional laminar flow; Transient incompressible flow.

1. Introduction

In the last few decades, a number of numerical techniques have been presented to solve the Navier-Stokes equations and for simulation of fluid flow. The computational fluid dynamics (CFD) techniques consisting of the finite difference, the finite element and the finite volume methods have achieved a lot of success in computer modeling of fluid flows. However, there are some drawbacks in these methods. For example, the mesh generation stage for these techniques is a far more time-consuming and expensive task, especially in three-dimensional problems or for those with moving boundaries. Therefore, it is necessary to find appropriate methods to overcome the difficulties associated with the mesh generation.

In recent years, a set of new methods known as meshfree or meshless methods has been developed to solve these problems. Among the meshfree methods, the Meshless local Petrov-Galerkin (MLPG) method introduced by Atluri and Zhu in 1998 has been well-known and one of the most successful of them (Atluri and Zhu 1998). The MLPG method does not need any mesh for either field interpolation or background integration. So far, considerable efforts have been made to use the MLPG method to solve various problems in the solid and fluid mechanics fields such as elasticity (Atluri and Zhu 2000; Baradaran et al. 2011; Bagheri et al. 2011; Mahmoodabadi et al. 2014), elastodynamics (Batra and Ching 2002; Soares et al. 2012; Heaney et al. 2010), fracture (Feng et al. 2009), crack analysis (Sladek et al. 2010; Sladek et al. 2012), heat transfer (Sladek et al. 2004; Baradaran and Mahmoodabadi 2009; Baradaran and Mahmoodabadi 2010; Mahmoodabadi et al. 2011; Karagiannakis et al. 2016; Li et al. 2018), vibrations (Andreaus et al. 2005; Gu and Liu 2001; Gu and Liu 2001a; Rashidi Moghaddam and Baradaran 2017), plate and shell (Sladek et al. 2007; Li et al. 2008; Sladek et al. 2013; Vaghefi et al. 2016), convection-diffusion (Chen et al. 2018), Klein–Gordon equation (Darani 2017), electric-field integral equation (Honarbakhsh 2017), metal removal in laser drilling (Abidouab et al. 2018) and fluid flow (Arefmanesh et al. 2010; Najafi et al. 2012).

On the other hand, some researchers have tried to analyze and solve the fluid flow problems. For instance, Lin and Atluri used the MLPG method to solve incompressible Navier-Stokes equations for stokes flow and lid driven cavity flow problems (Lin and Atluri 2001). The application of the MLPG approach to simulation of incompressible flow was presented in Wu et al (2005). In that study, the MLPG formulation with some modifications was used for an irregular domain with scattered nodal distribution. Arefmanesh et al. employed the MLPG method with the unity weighting function to solve the Navier–Stokes and energy equations for non-isothermal fluid flow based on the stream function and vorticity formulation (Arefmanesh et al. 2008). A stabilized MLPG method for steady state incompressible flow was studied in (Wu et al. 2010). In that work, the streamline upwind Petrov–Galerkin method was used to overcome oscillations in convection-dominated problems. Mohammadi (2008) presented a new type of upwind scheme to stable the convection operator in the streamline. Avila et al. (2011) developed a novel MLPG-mixed finite volume method to analyze the steady state stokesian equations (Avila et al. 2011). Simulation of two-dimensional unsteady incompressible flow is done by the MLPG method and with a forward difference scheme for the time derivatives in Sataprahm and Luadsong (2014). In one of the latest researches, Enjilela and Arefmanesh proposed a stabilized two-step Taylor-characteristic-based MLPG method to solve laminar fluid flow and heat transfer problems (Enjilela and Arefmanesh 2015). The results of that work shown that very good agreements exist between the results obtained using the proposed meshless method with those obtained using the conventional methods.

Although there are some studies about the simulation of laminar fluid flow using the meshless methods, to the best of the author’s knowledge, the analysis of three-dimensional transient incompressible fluid flow problems using the MLPG approach is still not studied. This motivates the present study for extension of the MLPG formulation to solve the three-dimensional transient incompressible Navier-Stokes equations in the primitive variables form. More precisely, in order to solve the Poisson equation of the pressure in the three-dimensional domain, the MLPG method is employed and combined with the FFD technique.

The rest of the paper is organized as follows. Section 2 presents the problem formulation and Discretization of time derivatives and the algorithm. In Section 3, details of MLPG formulation and local weak forms, moving least squares (MLS) approximation and the considered weight function are described. Section 4 illustrates the numerical examples and the related validations. Finally, Section 5 concludes the paper.

2. Problem and analysis algorithm

2.1 Problem formulation

The governing equation for unsteady incompressible viscous fluid flow as the three-dimensional Navier-Stokes equation together with the continuity equation in the convection term, i.e., non-conservative form can be written as:

$$\frac{\partial u}{\partial t}=\frac{1}{\mathrm{Re}}(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2})-\mathrm{u}\frac{\partial u}{\partial x}-v\frac{\partial u}{\partial y}-w\frac{\partial u}{\partial z}-\frac{\partial p}{\partial x}+f_x \tag{1}$$

$$\frac{\partial v}{\partial t}=\frac{1}{\mathrm{Re}}(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2})-\mathrm{u}\frac{\partial v}{\partial x}-v\frac{\partial v}{\partial y}-w\frac{\partial v}{\partial z}-\frac{\partial p}{\partial y}+f_y \tag{2}$$

$$\frac{\partial w}{\partial t}=\frac{1}{\mathrm{Re}}(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2})-\mathrm{u}\frac{\partial w}{\partial x}-v\frac{\partial w}{\partial y}-w\frac{\partial w}{\partial z}-\frac{\partial p}{\partial z}+f_z \tag{3}$$

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{4}$$

where u , v and w are the velocities in x , y and z direction, respectively. p shows the pressure, f_x , f_y and f_z are the body force. Re represents the Reynolds number. Equations (1) to (3) are the momentum equation and Equation (4) is the continuity equation. The boundary conditions can be assumed to be:

$$u=\bar{u} \quad v=\bar{v} \quad w=\bar{w} \quad p=\bar{p} \quad on \quad \Gamma_u \tag{5}$$

where Γ_u and Γ_q are subset of Γ which satisfy $\Gamma_u \cup \Gamma_q = \Gamma$.

2.2 Discretization of time derivatives and the algorithm

To deal with the time derivatives, a time stepping method based on the following approximations is employed.

$$\frac{\partial u}{\partial t}(\mathrm{x},\mathrm{t}_n)\simeq\frac{u^{n+1}(\mathrm{x})-u^n(\mathrm{x})}{\Delta t} \tag{6}$$

$$\frac{\partial v}{\partial t}(\mathrm{x},\mathrm{t}_n)\simeq\frac{v^{n+1}(\mathrm{x})-v^n(\mathrm{x})}{\Delta t} \tag{7}$$

$$\frac{\partial w}{\partial t}(\mathrm{x},\mathrm{t}_n)\simeq\frac{w^{n+1}(\mathrm{x})-w^n(\mathrm{x})}{\Delta t} \tag{8}$$

where $x=(\mathrm{x,y,z})^T, u^n=u(\mathrm{x},t_n),\ v^n=v(\mathrm{x},t_n),\ w^n=w(\mathrm{x},t_n)$ and Δt is the time step. For discretizing Equations (1) to (3) at times level n, the following substitutions are regarded.

$$\frac{u^{n+1}-u^n}{\Delta t}=\frac{1}{\mathrm{Re}}(\frac{\partial^2 \mathrm{u}^n}{\partial x^2}+\frac{\partial^2 \mathrm{u}^n}{\partial y^2}+\frac{\partial^2 \mathrm{u}^n}{\partial z^2})-\mathrm{u}\frac{\partial u^n}{\partial x}-v\frac{\partial u^n}{\partial y}-w\frac{\partial u^n}{\partial z}-\frac{\partial p^n}{\partial x}+f_x^n \tag{9}$$

$$u^{n+1}=u^n-\Delta t\frac{\partial p^n}{\partial x}+\Delta t\ [\frac{1}{\mathrm{Re}}(\frac{\partial^2 \mathrm{u}^n}{\partial x^2}+\frac{\partial^2 \mathrm{u}^n}{\partial y^2}+\frac{\partial^2 \mathrm{u}^n}{\partial z^2})-\mathrm{u}\frac{\partial u^n}{\partial x}-v\frac{\partial u^n}{\partial y}-w\frac{\partial u^n}{\partial z}+f_x^n] \tag{10}$$

let,

$$F^n=u^n+\Delta t[\frac{1}{\mathrm{Re}}(\frac{\partial^2 \mathrm{u}^n}{\partial x^2}+\frac{\partial^2 \mathrm{u}^n}{\partial y^2}+\frac{\partial^2 \mathrm{u}^n}{\partial z^2})-\mathrm{u}\frac{\partial u^n}{\partial x}-v\frac{\partial u^n}{\partial y}-w\frac{\partial u^n}{\partial z}+f_x^n] \tag{11}$$

therefore,

$$u^{n+1} = F^n - \Delta t \frac{\partial p^n}{\partial x} \tag{12}$$

and

$$\frac{v^{n+1} - v^n}{\Delta t} = \frac{1}{\mathrm{Re}}(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2} + \frac{\partial^2 v^n}{\partial z^2}) - \mathrm{u}\frac{\partial v^n}{\partial x} - v\frac{\partial v^n}{\partial y} - w\frac{\partial v^n}{\partial z} - \frac{\partial p^n}{\partial y} + f_y^n \tag{13}$$

$$v^{n+1} = v^n - \Delta t \frac{\partial p^n}{\partial y} + \Delta t[\frac{1}{\mathrm{Re}}(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2} + \frac{\partial^2 v^n}{\partial z^2}) - \mathrm{u}\frac{\partial v^n}{\partial x} - v\frac{\partial v^n}{\partial y} - w\frac{\partial v^n}{\partial z} + f_y^n] \tag{14}$$

let

$$G^n = v^n + \Delta t\ [\frac{1}{\mathrm{Re}}(\frac{\partial^2 v^n}{\partial x^2} + \frac{\partial^2 v^n}{\partial y^2} + \frac{\partial^2 v^n}{\partial z^2}) - \mathrm{u}\frac{\partial v^n}{\partial x} - v\frac{\partial v^n}{\partial y} - w\frac{\partial v^n}{\partial z} + f_y^n] \tag{15}$$

therefore

$$v^{n+1} = G^n - \Delta t \frac{\partial p^n}{\partial y} \tag{16}$$

and

$$\frac{w^{n+1} - w^n}{\Delta t} = \frac{1}{\mathrm{Re}}(\frac{\partial^2 w^n}{\partial x^2} + \frac{\partial^2 w^n}{\partial y^2} + \frac{\partial^2 w^n}{\partial z^2}) - \mathrm{u}\frac{\partial w^n}{\partial x} - v\frac{\partial w^n}{\partial y} - w\frac{\partial w^n}{\partial z} - \frac{\partial p^n}{\partial z} + f_z^n \tag{17}$$

$$w^{n+1} = w^n - \Delta t \frac{\partial p^n}{\partial z} + \Delta t[\frac{1}{\mathrm{Re}}(\frac{\partial^2 w^n}{\partial x^2} + \frac{\partial^2 w^n}{\partial y^2} + \frac{\partial^2 w^n}{\partial z^2}) - \mathrm{u}\frac{\partial w^n}{\partial x} - v\frac{\partial w^n}{\partial y} - w\frac{\partial w^n}{\partial z} + f_z^n] \tag{18}$$

let

$$L^n = w^n + \Delta t\ [\frac{1}{\mathrm{Re}}(\frac{\partial^2 w^n}{\partial x^2} + \frac{\partial^2 w^n}{\partial y^2} + \frac{\partial^2 w^n}{\partial z^2}) - \mathrm{u}\frac{\partial w^n}{\partial x} - v\frac{\partial w^n}{\partial y} - w\frac{\partial w^n}{\partial z} + f_z^n] \tag{19}$$

therefore

$$w^{n+1} = L^n - \Delta t \frac{\partial p^n}{\partial z} \tag{20}$$

Using Equations (12), (16) and (20), the following relations are obtained.

$$\frac{\partial u^{n+1}}{\partial x} = \frac{\partial F^n}{\partial x} - \Delta t \frac{\partial^2 p^n}{\partial x^2} \tag{21}$$

$$\frac{\partial v^{n+1}}{\partial y} = \frac{\partial G^n}{\partial y} - \Delta t \frac{\partial^2 p^n}{\partial y^2} \tag{22}$$

$$\frac{\partial w^{n+1}}{\partial z}=\frac{\partial L^{n}}{\partial z}-\Delta t\frac{\partial^{2}p^{n}}{\partial z^{2}} \tag{23}$$

Substitution Equations (21 to (23) into Equation (4), yields Equations (24) and (25).

$$\frac{\partial u^{n+1}}{\partial x}+\frac{\partial v^{n+1}}{\partial y}+\frac{\partial w^{n+1}}{\partial z}=0 \tag{24}$$

and

$$\frac{\partial^{2}p^{n}}{\partial x^{2}}+\frac{\partial^{2}p^{n}}{\partial y^{2}}+\frac{\partial^{2}p^{n}}{\partial z^{2}}=\frac{1}{\Delta t}(\frac{\partial F^{n}}{\partial x}+\frac{\partial G^{n}}{\partial y}+\frac{\partial L^{n}}{\partial z}) \tag{25}$$

Equation (25) is the poisons equation with the non-zero source term. The numerical implementation of the procedure described from Equations (12) through (25) can summarized as follows:

1. Compute the intermediate velocity field using

$$F^{n}=u^{n}+\Delta t[\frac{1}{\mathrm{Re}}(\frac{\partial^{2}\mathrm{u}^{n}}{\partial x^{2}}+\frac{\partial^{2}\mathrm{u}^{n}}{\partial y^{2}}+\frac{\partial^{2}\mathrm{u}^{n}}{\partial z^{2}})-\mathrm{u}\frac{\partial u^{n}}{\partial x}-v\frac{\partial u^{n}}{\partial y}-w\frac{\partial u^{n}}{\partial z}+f_{x}^{n}] \tag{26}$$

$$G^{n}=v^{n}+\Delta t\ [\frac{1}{\mathrm{Re}}(\frac{\partial^{2}v^{n}}{\partial x^{2}}+\frac{\partial^{2}v^{n}}{\partial y^{2}}+\frac{\partial^{2}v^{n}}{\partial z^{2}})-\mathrm{u}\frac{\partial v^{n}}{\partial x}-v\frac{\partial v^{n}}{\partial y}-w\frac{\partial v^{n}}{\partial z}+f_{y}^{n}] \tag{27}$$

$$L^{n}=w^{n}+\Delta t\ [\frac{1}{\mathrm{Re}}(\frac{\partial^{2}w^{n}}{\partial x^{2}}+\frac{\partial^{2}w^{n}}{\partial y^{2}}+\frac{\partial^{2}w^{n}}{\partial z^{2}})-\mathrm{u}\frac{\partial w^{n}}{\partial x}-v\frac{\partial w^{n}}{\partial y}-w\frac{\partial w^{n}}{\partial z}+f_{z}^{n}] \tag{28}$$

2. Solve the pressure Poisson’s equation

$$\frac{\partial^{2}p^{n}}{\partial x^{2}}+\frac{\partial^{2}p^{n}}{\partial y^{2}}+\frac{\partial^{2}p^{n}}{\partial z^{2}}=\frac{1}{\Delta t}(\frac{\partial F^{n}}{\partial x}+\frac{\partial G^{n}}{\partial y}+\frac{\partial L^{n}}{\partial z}) \tag{29}$$

3. Update the velocity field to the time level $n+1$ using

$$\frac{\partial u^{n+1}}{\partial x}=\frac{\partial F^{n}}{\partial x}-\Delta t\frac{\partial^{2}p^{n}}{\partial x^{2}} \tag{30}$$

$$\frac{\partial v^{n+1}}{\partial y}=\frac{\partial G^{n}}{\partial y}-\Delta t\frac{\partial^{2}p^{n}}{\partial y^{2}} \tag{31}$$

$$\frac{\partial w^{n+1}}{\partial z}=\frac{\partial L^{n}}{\partial z}-\Delta t\frac{\partial^{2}p^{n}}{\partial z^{2}} \tag{32}$$

3. Theory and numerical solution

3.1 MLPG formulation and local weak forms

The meshless local Petrov-Galerkin approach is constructed over a local sub-domain which is located inside the global domain. Here, the local sub-domain Ω_s is either a sphere or a part of a sphere. A generalized local weak form of Equation (25) over a local sub-domain Ω_s can be written as

$$\int_{\Omega s} [(\frac{\partial^2 p^n}{\partial x^2}+\frac{\partial^2 p^n}{\partial y^2}+\frac{\partial^2 p^n}{\partial z^2})-\frac{1}{\Delta t}(\frac{\partial F^n}{\partial x}+\frac{\partial G^n}{\partial y}+\frac{\partial L^n}{\partial z})]\,\Psi d\Omega-\alpha\int_{\Gamma_{su}}(p^n-\bar{p}^n)\,\Psi d\Gamma=0 \qquad (33)$$

where Ψ is the test function and Γ_{su} is a part of the boundary $\partial\Omega_s$ of Ω_s which the essential boundary conditions are specified on it. In general, $\partial\Omega_s=\Gamma_s\bigcup L_s$ where Γ_s is a part of the local boundary located on the global boundary and L_s .is the other part of the local boundary located in the global domain (Figure 3). In equation (33) a penalty parameter α is used to impose the essential boundary conditions. By using

$$(\nabla^2\,\mathrm{p})\,\Psi=[\frac{\partial}{\partial x}(\frac{\partial p}{\partial x}\Psi)+\frac{\partial}{\partial y}(\frac{\partial p}{\partial y}\Psi)+\frac{\partial}{\partial z}(\frac{\partial p}{\partial z}\Psi)]-(\frac{\partial p}{\partial x}\frac{\partial \Psi}{\partial x}+\frac{\partial p}{\partial y}\frac{\partial \Psi}{\partial y}+\frac{\partial p}{\partial z}\frac{\partial \Psi}{\partial z}) \qquad (34)$$

and the divergence theorem, Equation (32) leads to

$$\begin{aligned}&\int_{\Omega s}(\frac{\partial p^n}{\partial x}\frac{\partial \Psi}{\partial x}+\frac{\partial p^n}{\partial y}\frac{\partial \Psi}{\partial y}+\frac{\partial p^n}{\partial z}\frac{\partial \Psi}{\partial z})\,d\Omega-\int_{\Gamma_s}(\frac{\partial p^n}{\partial x}n_1+\frac{\partial p^n}{\partial y}\mathrm{n}_2+\frac{\partial p^n}{\partial z}\mathrm{n}_3)\Psi\,d\Gamma+\\&\alpha\int_{\Gamma_{su}}p^n\,\Psi\,d\Gamma-\int_{\Gamma_{su}}(\frac{\partial p^n}{\partial x}n_1+\frac{\partial p^n}{\partial y}\mathrm{n}_2+\frac{\partial p^n}{\partial z}\mathrm{n}_3)\,\Psi\,d\Gamma=+\alpha\int_{\Gamma_{su}}\bar{p}^n\,\Psi d\Gamma\\&+\frac{1}{\Delta t}\int_{\Omega s}(\mathrm{F}^n\frac{\partial \Psi}{\partial x}+G^n\frac{\partial \Psi}{\partial y}+L^n\frac{\partial \Psi}{\partial z})\,d\Omega-\frac{1}{\Delta t}\int_{\Gamma_s}(\mathrm{F}^n\mathrm{n}_1\,\Psi+G^n\,\mathrm{n}_2\,\Psi+L^n\,\mathrm{n}_3\,\Psi)\,d\Gamma\\&-\frac{1}{\Delta t}\int_{\Gamma_{su}}(\mathrm{F}^n\mathrm{n}_1\,\Psi+G^n\,\mathrm{n}_2\,\Psi+L^n\,\mathrm{n}_3\,\Psi)\,d\Gamma\end{aligned} \qquad (35)$$

where n_i $(i=1,2,3)$ are the components of outward unit normal vector to the sub-domain boundary $\partial\Omega_s$ usually composed of three parts; the internal boundary L_s, the boundary for the essential (Dirichlet) conditions Γ_{su} and the boundary for the natural (Neuman) conditions Γ_{sq}.

$$\sum_{j=1}^{M}K_{ij}\hat{p}=f_i \qquad i=1,2,3,\ldots,M \qquad (36)$$

where $\hat{p}=[\hat{\mathrm{p}}_1,\hat{\mathrm{p}}_2,\hat{\mathrm{p}}_3,\ldots\hat{\mathrm{p}}_N]^T$ is the pressure vector. The stiffness matrix K and the load vector f are defined as follows.

$$K_{ij} = \sum_{J=1}^{M} \int_{\Omega s} (\frac{\partial \varphi^J}{\partial x}\frac{\partial \Psi_I}{\partial x} + \frac{\partial \varphi^J}{\partial y}\frac{\partial \Psi_I}{\partial y} + \frac{\partial \varphi^J}{\partial z}\frac{\partial \Psi_I}{\partial z})\, d\Omega + \alpha \sum_{J=1}^{M} \int_{\Gamma_{su}} \varphi^J\ \Psi\ d\Gamma$$

$$-\sum_{J=1}^{M} \int_{\Gamma_s} (\frac{\partial \varphi^J}{\partial x} n_1 + \frac{\partial \varphi^J}{\partial y} \mathrm{n}_2 + \frac{\partial \varphi^J}{\partial z} \mathrm{n}_3)\ \Psi_I\ d\Gamma - \sum_{J=1}^{M} \int_{\Gamma_{su}} (\frac{\partial \varphi^J}{\partial x} n_1 + \frac{\partial \varphi^J}{\partial y} \mathrm{n}_2 + \frac{\partial \varphi^J}{\partial z} \mathrm{n}_3)\Psi\ d\Gamma \quad (37)$$

$$f_i = +\frac{1}{\Delta t} \int_{\Omega s} (\mathrm{F}^n \frac{\partial \Psi}{\partial x} + G^n \frac{\partial \Psi}{\partial y} + L^n \frac{\partial \Psi}{\partial z})\, d\Omega - \frac{1}{\Delta t} \int_{\Gamma_s} (\mathrm{F}^n \mathrm{n}_1\, \Psi + G^n\, \mathrm{n}_2\, \Psi + L^n\, \mathrm{n}_3\, \Psi)\ d\Gamma$$

$$-\frac{1}{\Delta t} \int_{\Gamma_{su}} (\mathrm{F}^n \mathrm{n}_1\, \Psi + G^n\, \mathrm{n}_2\, \Psi + L^n\, \mathrm{n}_3\, \Psi)\ d\Gamma + \alpha \int_{\Gamma_{su}} \bar{p}^n\ \Psi d\Gamma \quad (38)$$

3.2 Moving least squares (MLS) approximation

In the MLPG method, the field variable $u(\mathrm{x})$ could be approximated by the MLS technique. This approximation is based on three components: (1) a weight function of compact support associated with each node, (2) polynomial basis functions, and, (3) a set of coefficients that depend on the position x of the point. First, a sub-domain Ω_x located in the problem domain Ω and called the domain of definition of the MLS approximation for the trial function at the point x is considered.

Then, the unknown trial approximation $u_h(\mathrm{x})$ of the function $u(\mathrm{x})$ is defined using the following equation (Atluri and Zhu 1998).

$$u_h(\mathrm{x}) = \sum_{j=1}^{m} p_j(\mathrm{x})\ \mathrm{a}_j(\mathrm{x}) = \mathrm{p}^T(\mathrm{x})\mathrm{a}(\mathrm{x}), \quad \forall \mathrm{x} \in \Omega_x \quad (39)$$

In the above equation, $p^T(\mathrm{x}) = [\mathrm{p}_1(\mathrm{x}), \mathrm{p}_2(\mathrm{x}), ..., \mathrm{p}_m(\mathrm{x})]$ is a vector of the complete monomial basis of order m, and $a(\mathrm{x})$ is a vector containing unknown coefficients $a_j(\mathrm{x}), \quad \mathrm{j}=1,2,...\mathrm{m}$.

Two examples of $\mathrm{p}^T(\mathrm{x})$ in the three-dimensional problems are as follow (Atluri and Zhu 1998).

$$\mathrm{p}^T(\mathrm{x}) = [1, \mathrm{x}, \mathrm{y}, \mathrm{z}] \ \ \mathrm{for\, linear\, basis} \quad m = 4 \quad (40)$$

$$\mathrm{p}^T(\mathrm{x}) = [1, \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{x}^2, \mathrm{y}^2, \mathrm{z}^2, \mathrm{xy}, \mathrm{xz}, \mathrm{yz}]\, \mathit{for\, quadratic\, basis}, \quad m = 10 \quad (41)$$

The m unknown parameters $a_j(\mathrm{x})$ can be determined by minimizing the weighted discrete L_2 norm, defined as (Atluri and Zhu 1998):

$$J(\mathrm{x}) = \sum_{i=1}^{n} \Lambda_i(\mathrm{x})[\mathrm{p}^T(\mathrm{x}_i)\ \mathrm{a}(\mathrm{x})\text{-}\hat{\mathrm{u}}_i]^2 \quad (42)$$

where $w_i(\mathrm{x})$ is the weight function associated with the node i with $\Lambda_i(\mathrm{x}) > 0$ for all x located at the domain of influence of x_i, x_i denotes the regarded node i, n is the number of points in the domain of influence of x, Ω_x for which the weight functions $\Lambda_i(\mathrm{x}) > 0$ (Figure1), and $\hat{u}_i$ refers the fictitious nodal value and not the nodal value of the function u_h at the point x_i. Minimization of $J(\mathrm{x})$ in Equation (42) with respect to $a(\mathrm{x})$ leads to the following linear relation (Atluri and Zhu 1998).

$$A(\mathrm{x})\mathrm{a}(\mathrm{x}) = \mathrm{B}(\mathrm{x})\hat{\mathrm{u}} \tag{43}$$

where

$$A(\mathrm{x}) = \sum_{i=1}^{n} \Lambda_i(\mathrm{x})\mathrm{p}(\mathrm{x}_i)\mathrm{p}^T(\mathrm{x}_i) \tag{44}$$

$$B(\mathrm{x}) = [\Lambda_1(\mathrm{x})\mathrm{p}(\mathrm{x}_1), \Lambda_2(\mathrm{x})\mathrm{p}(\mathrm{x}_2), \ldots, \Lambda_n(\mathrm{x})\mathrm{p}(\mathrm{x}_n)] \tag{45}$$

$$\hat{u}^T = [\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_n] \tag{46}$$

If Equation (43) is solved for $a(\mathrm{x})$, it yields (Atluri and Zhu 1998):

$$a(x) = A^{-1}(x)B(x)\hat{u} \tag{47}$$

As it can be seen from Equation (47), the unknown coefficients $a(\mathrm{x})$ can be obtained only if $A(\mathrm{x})$ defined by Equation (43) is non-singular. Hence, a necessary condition for a well-defined MLS approximation is that at least m weight functions are non-zero (i.e. $n \geq m$) for each sample point $x \in \Omega$ (Atluri and Zhu 1998).

$$u^h(\mathrm{x}) = \sum_{i=1}^{n} \varphi_i(\mathrm{x})\hat{u}_i \quad , \quad u^h(\mathrm{x}) \equiv u_i \neq \hat{\mathrm{u}}_i \quad , \mathrm{x} \in \Omega_x \tag{48}$$

where

$$\varphi_i(\mathrm{x}) = \sum_{j=1}^{m} p_j(\mathrm{x})[\mathrm{A}^{-1}(\mathrm{x})\mathrm{B}(\mathrm{x})]_{ji} \tag{49}$$

$\varphi_i(\mathrm{x})$ is usually called the shape function of the MLS approximation corresponding to the node i.

The spatial derivatives of the shape function $\varphi_i(\mathrm{x})$ are (Atluri and Zhu 1998):

$$\varphi^I_{,K} = \sum_{j=1}^{m} \left[P_{j,k}\left(A^{-1}B\right)_{jI} + P_j\left(A^{-1}B_K + A^{-1}_K B\right)_{jI} \right] \tag{50}$$

where $()_{,i} \equiv \partial() / \partial x_i$. The derivative of A^{-1} in Equation (50) can be computed by taking the derivative of $A^{-1}A = I$. Thus,

$$A_{,K}^{-1} = -A^{-1} A_{,K} A^{-1} \tag{51}$$

The second partial derivatives of $\varphi_i(\mathrm{x})$ are obtained as:

$$\varphi_{,KL}^{I} = \sum_{j=1}^{m} \left[\begin{array}{r} P_{j,kl} \left(A^{-1} B \right)_{jI} + P_{j,k} \left(A^{-1} B_{,L} + A_{,L}^{-1} B \right)_{jI} + P_{j,L} \left(A^{-1} B,_{K} + A_{,K}^{-1} B \right)_{jI} \\ + P_j \left(A^{-1} B_{,KL} + A_{,KL}^{-1} B + A_{,L}^{-1} B_{,K} + A_{,K}^{-1} B_{,L} \right)_{jI} \\ \\ \end{array} \right] \tag{52}$$

With

$$A_{,KL}^{-1} = A^{-1} A_{,L} A^{-1} A_{,K} A^{-1} - A^{1} A_{,KL} A^{-1} + A^{-1} A_{,K} A^{-1} A,_{L} A^{-1} \tag{53}$$

It should be noted that the shape functions $\varphi_i(\mathrm{x}_j)$ derived from the MLS approximation do not satisfy the Kronecker delta criterion $\varphi_i(\mathrm{x}_j) \neq \delta_{ij}$. Therefore, they are not nodal interpolants and the name “approximation” is used, i.e. $u_h(\mathrm{x}_i) \neq \hat{\mathrm{u}}_i$ (Figure 2 illustrates the distinction between u_i and $\hat{u}_i$ for a simple one dimensional case).

This property makes the satisfaction of the essential boundary conditions more difficult than that in the finite element method. Several techniques have been developed to enforce the essential boundary conditions, such as: Lagrange multipliers (Belytschko et al. 1994), modified variational principles (Dolbow and Belytschko 1998), and the penalty method (Atluri and Zhu 1998). In the current research, the penalty method is implemented to enforce the essential boundary conditions.

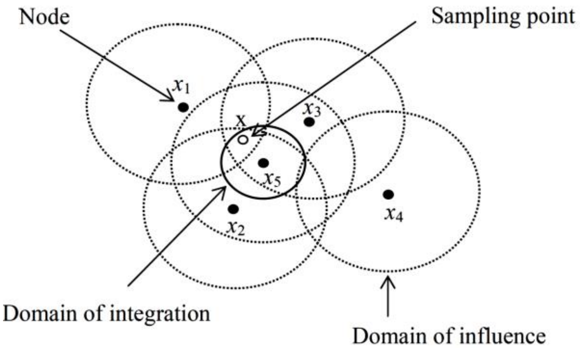

Figure 1. The illustration for the domain of influence and the domain of integration

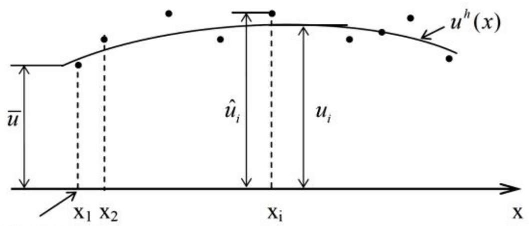

Figure 2. Distinction between u_i and $\hat{u}_i$ (Atluri and Zhu 1998)

3.3 Weight Function

An important ingredient in the meshless method is the weight function introduced in Equation (42). It should be non-zero only over a small neighborhood of a node (the domain of influence) in order to generate a set of sparse discrete equations.

More precisely, on a compact support of a node, it should be positive and increasing as $|x - x_i|$ is decreasing. Furthermore, it is desirable that $\Lambda_i(\mathrm{x})$ be smooth, if $\Lambda_i(\mathrm{x})$ is C^1 continuous, then for the polynomial basis, the shape functions $\varphi_i(\mathrm{x})$ are also C^1 continuous (Lu et al. 1994). Here, the Gaussian weight function commonly used for the meshless methods is implemented.

$$\Lambda_I\left(x\right)=\left\{\begin{array}{ll} \dfrac{\exp[-(\mathrm{d}_i/\mathrm{c}_i)^{2k}]-\exp[-(\mathrm{r}_i/\mathrm{c}_i)^{2k}])}{1-\exp[-(\mathrm{r}_i/\mathrm{c}_i)^{2k}]} & 0\le \mathrm{d}_i \le r_i \\ 0 & \mathrm{d}_i > r_i \end{array}\right\} \tag{54}$$

where $d_i = |x - x_i|$ is the distance between the sampling point x and the node x_i, and r_i is the radius of the domain of influence of the weight function $\Lambda_i(\mathrm{x})$. The parameters c_i and k in Equation (54) regulate the shape of the Gaussian weight function $\Lambda_i(\mathrm{x})$. In fact, there is not any accurate theory to determine a proper value for the parameter c_i. Lu et al. recommended a method to choose this constant as the distance from the node x_i to the third nearest neighboring node (Lu et al. 1994). The domain of influence r_i can be chosen as $r_i \ge 3.5c_i$ so that the weight function $\Lambda_i(\mathrm{x})$ covers a sufficient number of nodes to ensure the non-singularity of matrix A in Equation (49). The parameter k in Equation (54) is usually set at 1.

The first derivatives of a Gaussian weight function $\Lambda_i(\mathrm{x})$ can be calculated as:

$$\Lambda_{i,x}\left(x\right)=\left\{\begin{array}{ll} \dfrac{-2k\ (\mathrm{x}-\mathrm{x}_i)\ \mathrm{d}_i^{2k-2}\exp[-(\mathrm{d}_i/\mathrm{c}_i)^{2k}]}{c_i^{2k}\ (1-\exp[-(\mathrm{r}_i/\mathrm{c}_i)^{2k}])} & 0\le \mathrm{d}_i \le r_i \\ 0 & \mathrm{d}_i > r_i \end{array}\right\} \tag{55}$$

$$\Lambda_{i,y}\left(x\right)=\left\{\begin{array}{ll} \dfrac{-2k\ (y-y_i)\ \mathrm{d}_i^{2k-2}\exp[-(\mathrm{d}_i/\mathrm{c}_i)^{2k}]}{c_i^{2k}\ (1-\exp[-(\mathrm{r}_i/\mathrm{c}_i)^{2k}])} & 0\le \mathrm{d}_i \le r_i \\ 0 & \mathrm{d}_i > r_i \end{array}\right\} \tag{56}$$

$$\Lambda_{i,z}\left(x\right)=\left\{\begin{array}{ll} \dfrac{-2k\ (z-z_i)\ \mathrm{d}_i^{2k-2}\exp[-(\mathrm{d}_i/\mathrm{c}_i)^{2k}]}{c_i^{2k}(1-\exp[-(\mathrm{r}_i/\mathrm{c}_i)^{2k}])} & 0\le \mathrm{d}_i \le r_i \\ 0 & \mathrm{d}_i > r_i \end{array}\right\} \tag{57}$$

In the case that d_i is equal to zero, the first derivatives of $\Lambda_i(\mathrm{x})$ may be written as:

$$\Lambda_{i,x} = \Lambda_{i,y} = \Lambda_{i,z} = 0 \tag{58}$$

Similarly, the second derivatives of the weight function can be computed as:

$$\Lambda_{i,xx}\left(x\right) = \begin{cases} \dfrac{-2k * d_i^{2k-4} \exp\left[-\left(\dfrac{d_i}{C_i}\right)^{2K}\right]}{C_i^{2k}(1-\exp\left[-\left(\dfrac{r_i}{C_i}\right)^{2K}\right])} \left[\left(\dfrac{-2k(x-x_i)^2 d_i^{2k}}{C_i^{2k}}\right) + (2k-2)(x-x_i)^2 + d_i^2\right] & d_I \le r_i \\ 0 & d_I > r_i \end{cases} \tag{59}$$

$$\Lambda_{i,yy}\left(x\right) = \begin{cases} \dfrac{-2k * d_i^{2k-4} \exp\left[-\left(\dfrac{d_i}{C_i}\right)^{2K}\right]}{C_i^{2k}(1-\exp\left[-\left(\dfrac{r_i}{C_i}\right)^{2K}\right])} \left[\left(\dfrac{-2k(y-y_i)^2 d_i^{2k}}{C_i^{2k}}\right) +(2k-2)(y-y_i)^2 + d_i^2\right] & d_I \le r_i \\ 0 & d_I > r_i \end{cases} \tag{60}$$

$$\Lambda_{i,zz}\left(x\right) = \begin{cases} \dfrac{-2k * d_i^{2k-4} \exp\left[-\left(\dfrac{d_i}{C_i}\right)^{2K}\right]}{C_i^{2k}(1-\exp\left[-\left(\dfrac{r_i}{C_i}\right)^{2K}\right])} \left[\left(\dfrac{-2k(z-z_i)^2 d_i^{2k}}{C_i^{2k}}\right) +(2k-2)(z-z_i)^2 + d_i^2\right] & d_I \le r_i \\ 0 & d_I > r_i \end{cases} \tag{61}$$

Furthermore, for the case that d_i is equal to zero, the second derivatives of $\Lambda_i(\mathrm{x})$ may be written as:

$$\Lambda_{i,xx}(x)=\Lambda_{i,yy}(x)=\Lambda_{i,zz}(x)=\frac{-2k*d_i^{2k-4}\exp\left[-\left(\frac{d_i}{C_i}\right)^{2K}\right]}{C_i^{2k}(1-\exp\left[-\left(\frac{r_i}{C_i}\right)^{2K}\right])} \tag{62}$$

More details about these derivatives have been illustrated in Tanojo (2007).

4. Results of numerical examples

In this section, the introduced strategy based on the meshless local Petrov–Galerkin method is applied to analyze three-dimensional transient incompressible Navier–Stokes equations in two examples. The global domain is considered as a cubic geometry (Figure 4) with $a = b = 1$ and the node distribution with 64 regular nodes is regarded (Figure 5) for the both numerical simulations. Furthermore, other effective parameters are set as follows. The final time solution $t = 0.1$, the time step $\Delta t = 0.0001$, the weight function parameters $c_i = 0.4$ and $r_i = 1.7$, the radius of the sub-domain $r_o = 0.01$, the penalty parameter $\alpha = 10^{17}$ and finally $Re = 100$.

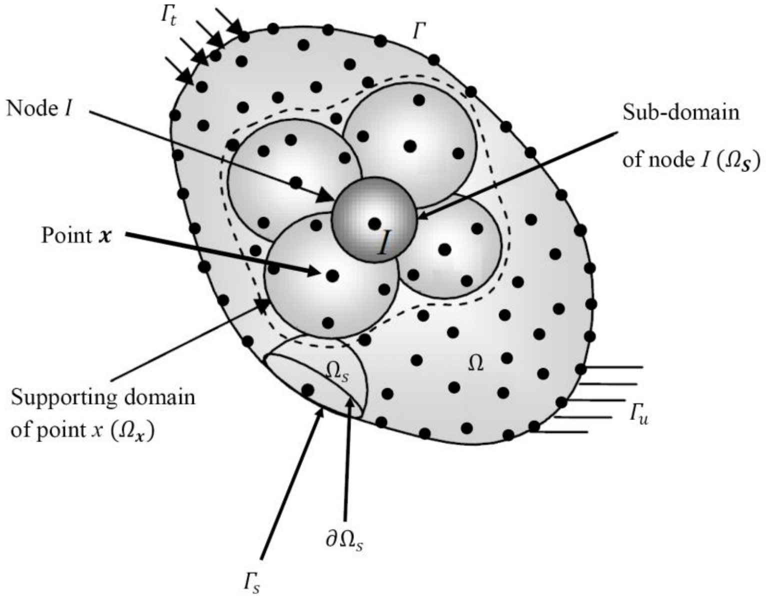

Figure 3. An overview of the MLPG method definitions (Mahmoodabadi et al. 2014)

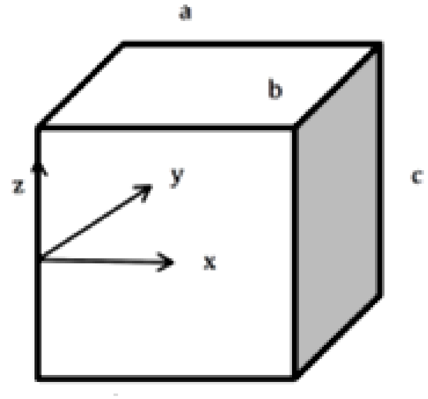

Figure 4. The cubic geometry for the numerical simulations

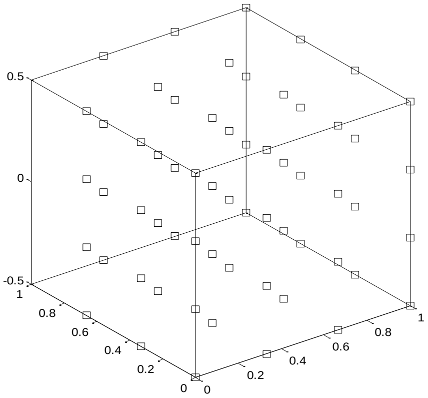

Figure 5. The regular node distribution with 64 nodes

4.1 Example 1

Boundary conditions and initial conditions of this example are as follows:

$$p(0,y,z,t) = (-0.25 - 0.25\cos 2y - z^2)e^{-2t/\mathrm{Re}} \tag{63}$$

$$p(1,y,z,t) = (-0.25\cos(2) - 0.25\cos 2y - z^2)e^{-2t/\mathrm{Re}} \tag{64}$$

$$p(x,0,z,t) = (-0.25\cos 2x - 0.25 - z^2)e^{-2t/\mathrm{Re}} \tag{65}$$

$$p(x,1,z,t) = (-0.25\cos 2x - 0.25\cos(2) - z^2)e^{-2t/\mathrm{Re}} \tag{66}$$

$$p(x,y,0,t) = (-0.25\cos 2x - 0.25\cos 2y)e^{-2t/\mathrm{Re}} \tag{67}$$

$$p(x,y,1,t) = (-0.25\cos 2x - 0.25\cos 2y - 1)e^{-2t/\mathrm{Re}} \tag{68}$$

$$p(x,y,z,0) = (-0.25\cos 2x - 0.25\cos 2y - z^2) \tag{69}$$

The analytical solutions for this problem are:

$$u(\mathrm{x,y,z,t}) = -\sin x\,\mathrm{e}^{-t/\mathrm{Re}} \tag{70}$$

$$v(\mathrm{x,y,z,t}) = -\sin y\,\mathrm{e}^{-t/\mathrm{Re}} \tag{71}$$

$$w(\mathrm{x,y,z,t}) = z(\mathrm{cosx+cosy})\mathrm{e}^{-t/\mathrm{Re}} \tag{72}$$

$$p(\mathrm{x,y,z,t}) = (-\frac{1}{4}\mathrm{cos2x}-\frac{1}{4}\mathrm{cos2y}-\mathrm{z}^2)\mathrm{e}^{-2t/\mathrm{Re}} \tag{73}$$

$$f_x = 0 \tag{74}$$

$$f_y = 0 \tag{75}$$

$$f_z = 2z\cos x\cos y\; e^{-2t/\mathrm{Re}} \tag{76}$$

The obtained numerical solutions for the pressure and velocities are shown in Figures 6 through 9 and compared with those of the analytical approach. These comparisons obviously depict the ability and effectiveness of the proposed strategy to solve the three-dimensional fluid flow problems. Moreover, Figures 10 and 13 illustrate the surface plots of the numerical and analytical solutions for the pressure distribution and the velocities. As shown in these figures, the results of MLPG method agree with the values obtained by analytical solution. The convergence of the MLPG approach has been demonstrated in these figures.

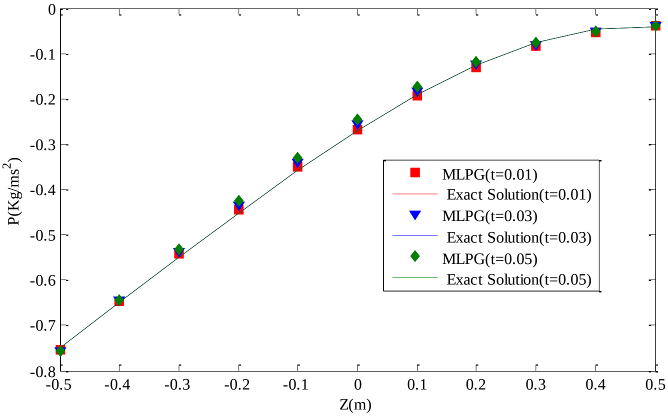

Figure 6. The pressure distribution (P (Kg/m^2)) for Example 1

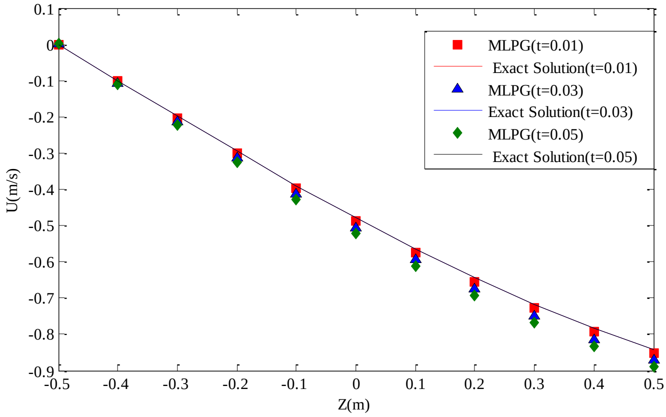

Figure 7. The velocity of X direction (U (m/s)) for Example 1

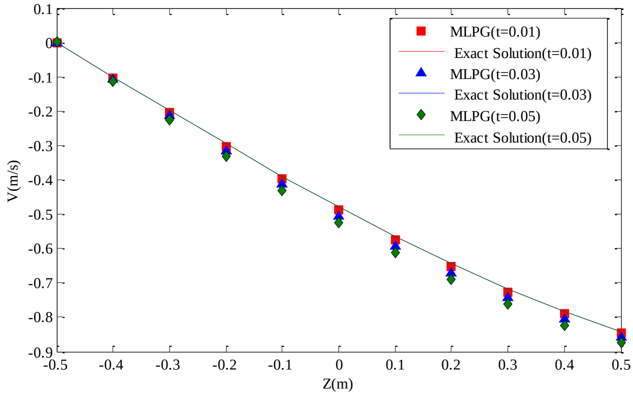

Figure 8. The velocity of Y direction (V (m/s)) for Example 1

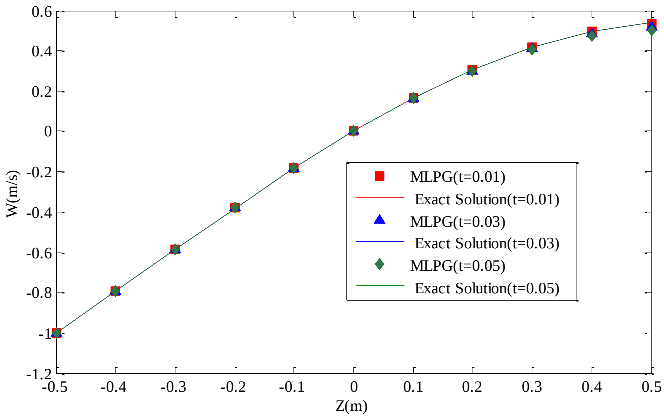

Figure 9. The velocity of Z direction (W (m/s)) for Example 1

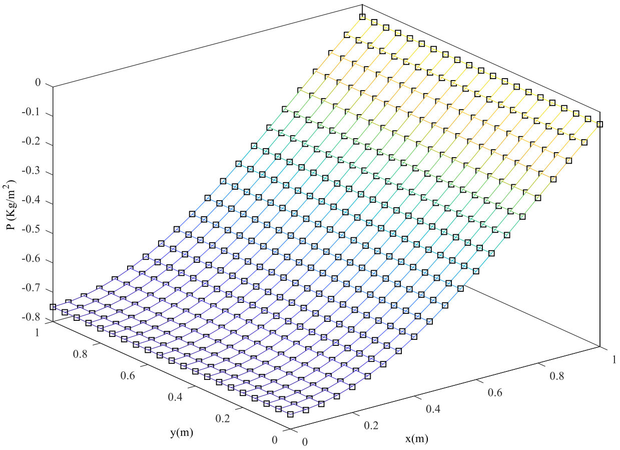

Figure 10. Surface plots of the numerical and analytical solutions for pressure distribution (P (Kg/m^2)) for Example 1

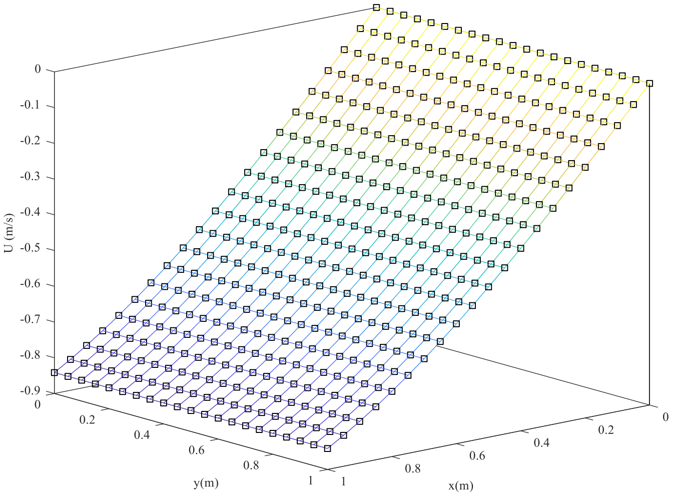

Figure 11. Surface plots of the numerical and analytical solutions for X direction velocity (U (m/s)) for Example 1

r_0	*U* (m/s)	*V* (m/s)	*W* (m/s)	*P* (Kg/ms^2)
0.01	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	5.6371×10^{-3}
0.05	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	3.2735×10^{-3}
0.1	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	5.3504×10^{-3}
0.2	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	4.2712×10^{-3}

Table 1. Obtained errors for different values of the radius of the sub-domain for Example 1.

r_i	*U* (m/s)	*V* (m/s)	*W* (m/s)	*P* (Kg/ms^2)
1.7	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	5.6371×10^{-3}
3	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	3.1005×10^{-4}
5	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	1.5224×10^{-4}

Table 2. Obtained errors for different values of radius of the domain of influence of the weight function for Example 1.

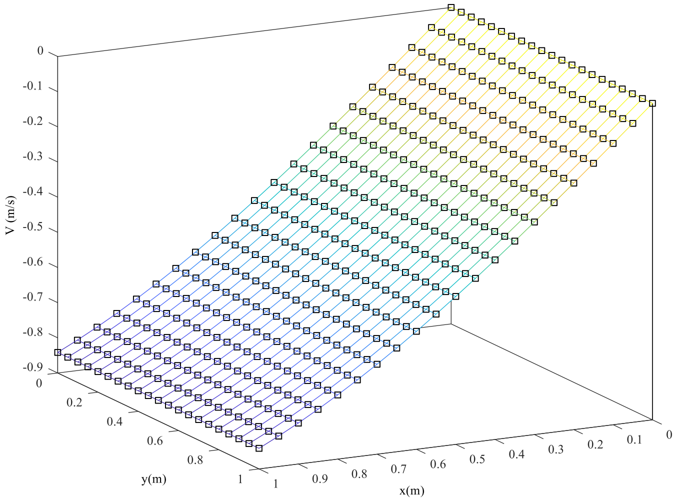

Figure 12. Surface plots of the numerical and analytical solutions for Y direction velocity (V (m/s)) for Example 1

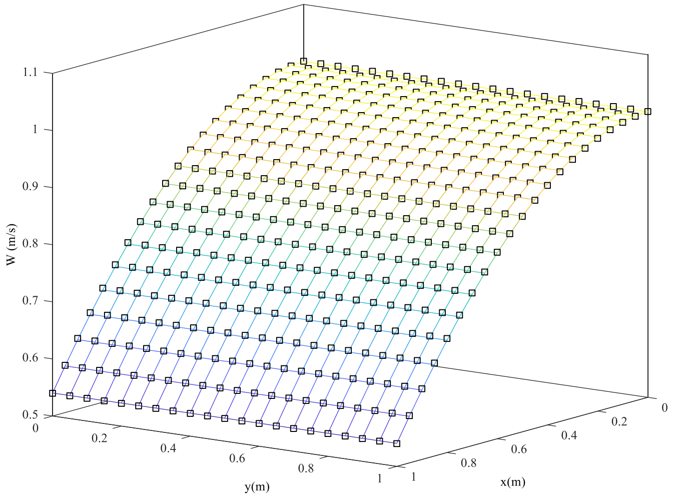

Figure 13. Surface plots of the numerical and analytical solutions for Z direction velocity (W (m/s)) for Example 1

M	*U* (m/s)	*V* (m/s)	*W* (m/s)	*P* (Kg/ms^2)
64	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	5.637×10^{-3}

125	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	8.815×10^{-5}
216	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	1.129×10^{-9}

Table 3. Obtained errors for different number of nodes for Example 1.

Δt	U (m/s)	V (m/s)	W (m/s)	P (Kg/ms^2)
0.005	6.8971×10^{-3}	4.7887×10^{-3}	1.9386×10^{-3}	4.5802×10^{-2}
0.001	6.9632×10^{-3}	4.0436×10^{-3}	2.1486×10^{-3}	4.1782×10^{-2}
0.0005	6.9733×10^{-3}	4.1354×10^{-3}	2.1040×10^{-3}	3.6819×10^{-2}
0.0001	7.0194×10^{-3}	6.3572×10^{-3}	1.5506×10^{-3}	5.637×10^{-3}

Table 4. Obtained errors for different values of the time steps for Example 1.

In Table 1, the effect of the value of the radius of the sub-domain has been investigated on the accuracy of the results. For these simulations, N=64, t=0.01, Δt=0.0001 and r_i=1.7 have been utilized. It can be observed from this table that this parameter doesn't affect the algorithm accuracy, and the solution process is not dependent on the radius of the sub-domain.

In Table 2, the effect of the value of the radius of the domain of influence of the weight function on the accuracy of the results has been examined with N=64, t=0.01, Δt=0.0001 and r_0=0.01. It can be seen that increasing this parameter would decreased the error and increased the solution accuracy of the pressure distribution.

The results of changing the value of the number of nodes on the velocities and pressure distributions have been studied in Table 3. For these simulations, r_0=0.01, t=0.01, Δt=0.0001 and r_i=1.7 have been utilized. It can be noted from this table that increasing the numbers of nodes would significantly improve the solutions accuracy.

In Table 4, the effect of the value of the time steps has been considered on the accuracy of the results with N=64, t=0.01, Δt=0.0001, r_0=0.01 and r_i=1.7 have been utilized. It can be observed from this table that for small values of this parameter, the algorithm has an acceptable performance.

4.2 Example 2

Boundary conditions and initial conditions of this example is as follows:

$$\mathrm{p}(0,\mathrm{y},\mathrm{z},\mathrm{t}) = 0.5(-2\mathrm{y}^2 + z^2)\mathrm{e}^{-t} \tag{77}$$

$$\mathrm{p}(1,\mathrm{y},\mathrm{z},\mathrm{t}) = 0.5(1 - 2\mathrm{y}^2 + z^2)\mathrm{e}^{-t} \tag{78}$$

$$\mathrm{p}(\mathrm{x},0,\mathrm{z},\mathrm{t}) = 0.5(x^2 + z^2)\mathrm{e}^{-t} \tag{79}$$

$$\mathrm{p}(\mathrm{x},1,\mathrm{z},\mathrm{t}) = 0.5(x^2 - 2 + z^2)\mathrm{e}^{-t} \tag{80}$$

$$\mathrm{p}(\mathrm{x},\mathrm{y},0,\mathrm{t}) = 0.5(x^2 - 2\mathrm{y}^2)\mathrm{e}^{-t} \tag{81}$$

$$\mathrm{p}(\mathrm{x},\mathrm{y},1,\mathrm{t}) = 0.5(x^2 - 2\mathrm{y}^2 + 1)\mathrm{e}^{-t} \tag{82}$$

$$p(\mathrm{x},\mathrm{y},\mathrm{z},0) = 0.5(x^2 - 2\mathrm{y}^2 + z^2) \tag{83}$$

The analytical solutions for this problem are:

$$u(\mathrm{x,y,z,t}) = (\mathrm{x} - 2\mathrm{y} + z)\mathrm{e}^{-t} \tag{84}$$

$$v(\mathrm{x,y,z,t}) = (\mathrm{x} - 2\mathrm{y} + z)\mathrm{e}^{-t} \tag{85}$$

$$w(\mathrm{x,y,z,t}) = (\mathrm{x} - 2\mathrm{y} + z)\mathrm{e}^{-t} \tag{86}$$

$$p(\mathrm{x,y,z,t}) = 0.5(x^2 - 2\mathrm{y}^2 + z^2)\mathrm{e} \tag{87}$$

$$f_y = -(\mathrm{x} + \mathrm{z})e^{-t} \tag{88}$$

$$f_x = (2y - z)e^{-t} \tag{89}$$

$$f_z = (2\mathrm{y} - \mathrm{x})e^{-t} \tag{90}$$

In order to challenge the feasibility and efficiency of the introduced scenario, the pressure distribution and the velocities of three different directions are presented in Figures 14 to 17 for *t*=1, 2 and 3 *s*. As evidenced by the overall assessment based on these figures, the extended MLPG method could successfully implemented to analyze three-dimensional transient incompressible fluid flow. Moreover, for more validations, Figures 18 and 19 respectively illustrate the contours of the pressure and X direction velocity distributions. It can be mentioned that according to Equations (84) to (86), the contours of Y and Z directions are same as the presented results in Figure 19. This results are also validated by the obtained results using the MLPG method.

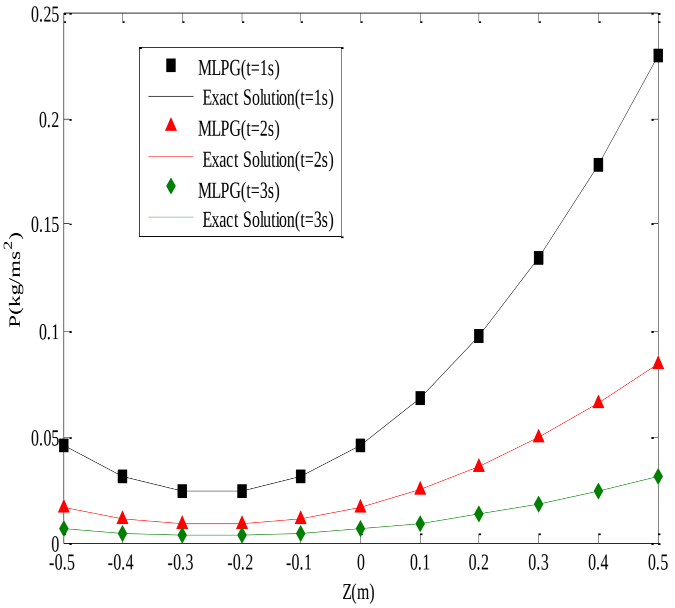

Figure 14. The pressure distribution (P (Kg/m^2)) for Example 2

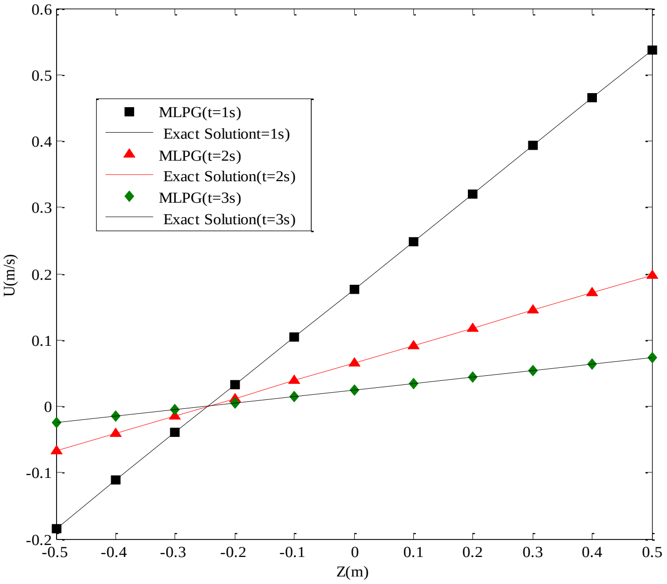

Figure 15. The velocity of X direction (U (m/s)) for Example 2

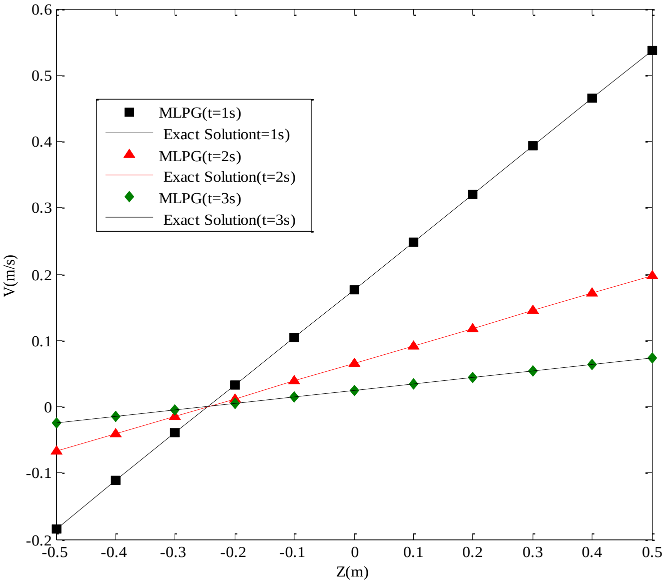

Figure 16. The velocity of Y direction (V (m/s)) for Example 2

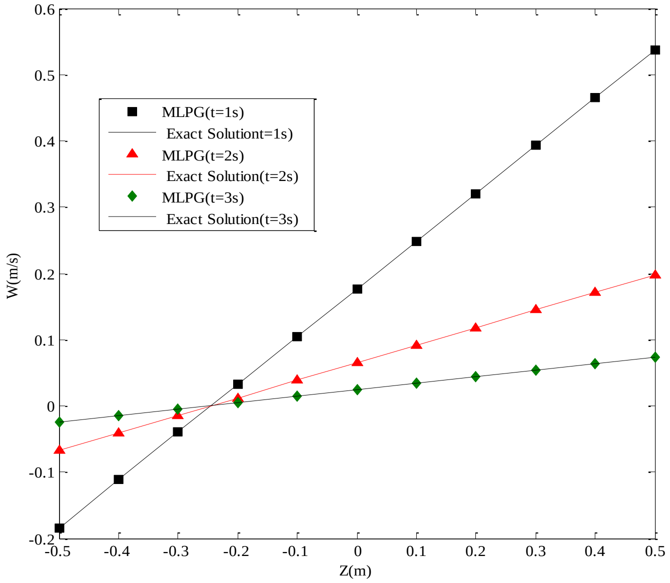

Figure 17. The velocity of Z direction (W (m/s)) for Example 2

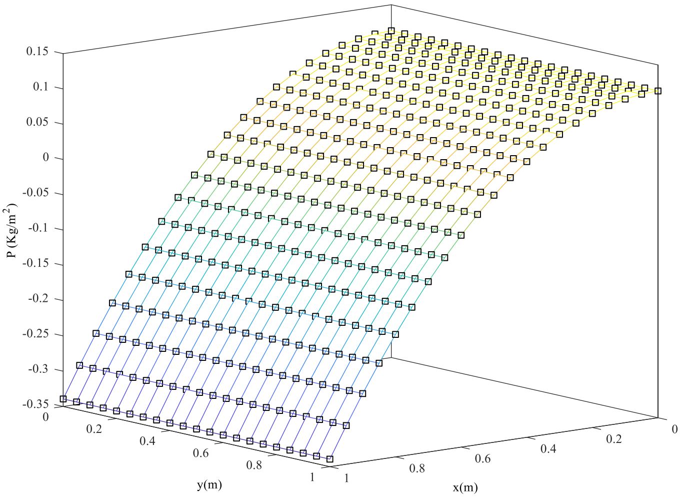

Figure 18. Surface plots of the numerical and analytical solutions for pressure distribution (P (Kg/m^2)) for Example 2

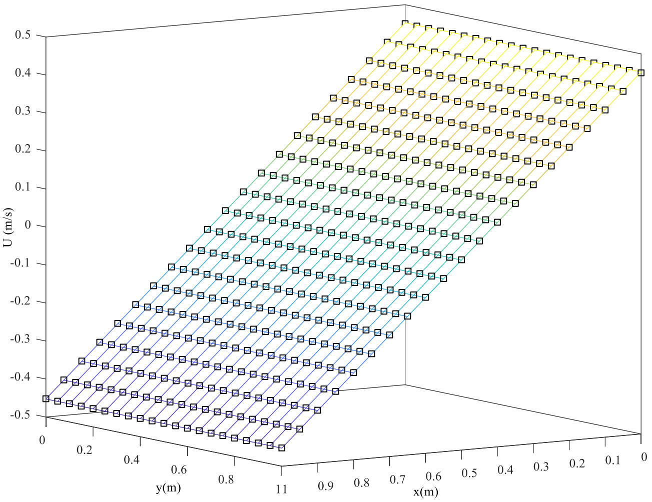

Figure 19. Surface plots of the numerical and analytical solutions for velocities distribution for Example 2

r_0	U (m/s)	V (m/s)	W (m/s)	P (Kg/ms^2)
0.01	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	2.3343×10^{-8}
0.05	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	2.2735×10^{-8}
0.1	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	1.7504×10^{-8}
0.2	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	3.2952×10^{-8}

Table 5. Obtained errors for different values of the radius of the sub-domain for Example 2.

r_i	U (m/s)	V (m/s)	W (m/s)	P (Kg/ms^2)
1.1	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	1.3024×10^{-4}
1.3	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	2.3454×10^{-8}
1.7	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	1.7504×10^{-8}
2	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	9.6643×10^{-9}

Table 6. Obtained errors for different values of radius of the domain of influence of the weight function for Example 2.

M	U (m/s)	V (m/s)	W (m/s)	P (Kg/ms^2)
64	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	1.7504×10^{-8}
125	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	4.2523×10^{-9}
216	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	2.7484×10^{-11}

Table 7. Obtained errors for different numbers of nodes for Example 2.

Δt	U (m/s)	V (m/s)	W (m/s)	P (Kg/ms^2)
0.05	1.5485×10^{-3}	1.5485×10^{-3}	1.5485×10^{-3}	2.3454×10^{-8}
0.01	3.0792×10^{-4}	3.0790×10^{-4}	3.0789×10^{-4}	1.8313×10^{-8}
0.001	3.0865×10^{-5}	3.0838×10^{-5}	3.0827×10^{-5}	2.0401×10^{-8}

Table 8. Obtained errors for different values of the time steps for Example 2.

In Table 5, the effect of the value of the radius of the sub-domain has been investigated on the accuracy of the results. For these simulations, N=64, t=0.1, Δt=0.05 and r_i=1.3 have been utilized. It can be observed from this table that this parameter doesn't affect the algorithm accuracy, and the solution process is not dependent on the radius of the sub-domain.

In Table 6, the effect of the value of the radius of the domain of influence of the weight function on the accuracy of the results has been examined with N=64, t=0.1, Δt=0.05 and r_0=0.1. It can be seen that increasing this parameter would decreased the error and increased the solution accuracy of the pressure distribution.

The results of changing the value of the number of nodes on the velocities and pressure distributions have been studied in Table 7. For these simulations, r_0=0.1, t=0.1, Δt=0.05 and r_i=1.3 have been utilized. It can be noted from this table that increasing the numbers of nodes would significantly improve the solutions accuracy.

In Table 8, the effect of the value of the time steps has been considered on the accuracy of the results with N=64, t=0.1, Δt=0.0001, r_0=0.1 and r_i=1.3 have been utilized. It can be observed from this table that for small values of this parameter, the algorithm has an acceptable performance.

5. Conclusion

This study deals with the development of a meshless local Petrov-Galerkin method for solving the three-dimensional transient incompressible Navier–Stokes equations based on the local weak forms. The MLS approximation is used for constructing the trial functions, while the imposition of the essential boundary conditions is carried out by using the penalty approach. Comparisons of the obtained numerical results with the exact solutions reveal that the introduced scenario operates very well in terms of solution accuracy, convergence and reliability.

References

Abidouab D, Sarhand AAD, Yusoffa N, Ghazalia NNN, Awangc MAO, Hassane MA (2018). Numerical simulation of metal removal in laser drilling using meshless local Petrov–Galerkin collocation method, Applied Mathematical Modelling, 56, 239-253.

Andreaus U, Batra RC, Porfiri M (2005). Vibrations of cracked Euler-Bernoulli beams using meshless local Petrov-Galerkin (MLPG) method, Computer Modeling in Engineering & Sciences, 9(2), 111-131.

Arefmanesh A, Najafi M, Abdi H (2008). Meshless local Petrov-Galerkin method with unity test function for non-isothermal fluid flow, Computer Modeling in Engineering & Sciences, 25(1), 9-22.

Arefmanesh A, Najafi M, Nikfar M (2010). MLPG application of nanofluid flow mixed convection heat transfer in a wavy wall cavity, Computer Modeling in Engineering & Sciences, 69(2), 91-118.

Atluri SN and Zhu T (1998). A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, Computational Mechanics, 22(2), 117-127.

Atluri SN and Zhu T (2000). The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics, Computational Mechanics, 25(2-3), 169-179.

Avila R, Han Z, Atluri SN (2011). A novel MLPG-finite-volume mixed method for analyzing stokesian flows & study of a new vortex mixing flow, Computer Modeling in Engineering & Sciences, 71(4), 363-396.

Bagheri A, Ehsany R, Mahmoodabadi MJ, Baradaran GH (2011). Optimization of meshless local Petrov-Galerkin parameters using genetic algorithm for 3D elasto-static problems (technical note), International Journal of Engineering, Transactions A, 24(2), 143-152.

Baradaran GH and Mahmoodabadi MJ (2009). Optimal Pareto parametric analysis of two dimensional steady-state heat conduction problems by MLPG method, International Journal of Engineering, Transactions B, 22(4), 387-406.

Baradaran GH and Mahmoodabadi MJ (2010). Parametric study of the MLPG method for the analysis of three dimensional steady state heat conduction problems, Journal of Mechanical Engineering, 61(1), 31-61.

Baradaran GH, Mahmoudabadi MJ and Sarfarazi MM (2011). Analyze of 3D elasto-static problems by meshless local Petrov-Galerkin method, International Journal of Advanced Design and Manufacturing Technology, 3(2), 37-44.

Batra RC and Ching HK (2002). Analysis of elastodynamic deformations near a Crack/Notch tip by the meshless local Petrov-Galerkin (MLPG) method, Computer Modeling in Engineering & Sciences, 3(6), 717-730.

Belytschko T, Lu YY, Gu L (1994). Element-free Galerkin methods, International Journal for Numerical methods in Engineering, 37(2), 229-256.

Chen ZJ, Li ZY, Xie WL, Wu XH (2018). A two-level variational multiscale meshless local Petrov–Galerkin (VMS-MLPG) method for convection-diffusion problems with large Peclet number, Computers & Fluids, 164, 73-82.

Darani MA (2017). Direct meshless local Petrov–Galerkin method for the two-dimensional Klein–Gordon equation, Engineering Analysis with Boundary Elements, 74, 1-13.

Dolbow J and Belytschko T (1998). An introduction to programming the meshless Element Free Galerkin method, Archives of Computational Methods in Engineering, 5(3), 207–241.

Enjilela V and Arefmanesh A (2015). Two-step Taylor-characteristic-based MLPG method for fluid flow and heat transfer application, Engineering Analysis with Boundary Elements, 51, 174-190.

Feng WJ, Han X, Li YS (2009). Fracture analysis for two-dimensional plane problems of non-homogeneous magneto-electro-thermo-elastic plates subjected to thermal shock by using the meshless local Petrov-Galerkin method, Computer Modeling in Engineering & Sciences, 48(1), 1-26.

Gu YT and Liu GR (2001). A meshless local Petrov-Galerkin (MLPG) method for free and forced vibration analyses for solids, Computational Mechanics, 27, 188-198.

Gu YT and Liu GR (2001). A meshless local Petrov-Galerkin (MLPG) formulation for static and free vibration analyses of thin plates, Computer Modeling in Engineering & Sciences, 2, 463-476.

Heaney C, Augarde C, Deeks A (2010). Modelling elasto-plasticity using the hybrid MLPG method, Computer Modeling in Engineering & Sciences, 56(2), 153-177.

Honarbakhsh B (2017). Numerical solution of EFIE using MLPG methods, Engineering Analysis with Boundary Elements, 80, 199-217.

Karagiannakis NP, Bourantas GC, Kalarakis AN, Skouras ED, Burganos VN (2016). Transient thermal conduction with variable conductivity using the Meshless Local Petrov–Galerkin method, Applied Mathematics and Computation, 272, 676-686.

Li D, Lin Z, Li S (2008). Numerical analysis of Mindlin shell by meshless local Petrov-Galerkin method, Acta Mechanica Solida Sinica, 21(2), 160-169.

Li ZY, Chen ZJ, Wu XH, Tao WQ (2018). Coupled MLPG–FVM simulation of steady state heat conduction in irregular geometry, Engineering Analysis with Boundary Elements, Corrected proof, available online.

Lin H and Atluri SN (2001). The meshless local Petrov-Galerkin (MLPG) method for solving incompressible Navier-Stokes equation, Computer Modeling in Engineering & Sciences, 2, 117-142.

Lu YY, Belytschko T, Gu L (1994). A new implementation of the element free Galerkin method, Computer Methods in Applied Mechanics and Engineering, 113, 397-414.

Mahmoodabadi MJ, Abedzadeh Maafi R, Bagheri A, Baradaran GH (2011). Meshless local Petrov-Galerkin method for 3D steady-state heat conduction problems, Advances in Mechanical Engineering, doi:10.1155/2011/251546.

Mahmoodabadi MJ, Sarfarazi MM, Bagheri A, Baradaran GH (2014). Meshless local Petrov-Galerkin method for elasto-static analysis of thick-walled isotropic laminated cylinders, International Journal of Engineering, Transaction B, 27(11), 1731-1740, 2014.

Mohammadi MH (2008). Stabilized meshless local Petrov-Galerkin (MLPG) method for incompressible viscous fluid flows, Computer Modeling in Engineering & Sciences, 29(2), 75-94.

Najafi M, Arefmanesh A, Enjilela V (2012). Meshless local Petrov-Galerkin method-higher Reynolds numbers fluid flow applications, Engineering Analysis with Boundary Elements, 36(11), 1671-1685.

Rashidi Moghaddam M and Baradaran GH (2017). Three-dimensional free vibrations analysis of functionally graded rectangular plates by the meshless local Petrov–Galerkin (MLPG) method, Applied Mathematics and Computation, 304, 153-163.

Sataprahm C and Luadsong A (2014). The meshless local pertov-Galerkin method for simulating unsteady incompressible fluid flow, Journal Egyption Mathematical Society, 22(3), 501-510.

Sladek J, Sladek V, Atluri SN (2004). Meshless local Petrov-Galerkin method for heat conduction problem in an anisotropic medium, Computer Modeling in Engineering & Sciences, 6(3), 309-318.

Sladek J, Sladek V, Krivacek J, Wen PH, Zhang C (2007). Meshless local Petrov-Galerkin (MLPG) method for Reissner-Mindlin plates under dynamic load, Computer Methods in Applied Mechanics and Engineering, 196, 2681-2691.

Sladek J, Sladek V, Stanak P, Zhang C, Wunsche M (2013). Analysis of the bending of circular piezoelectric plates with functionally graded material properties of a MLPG method, Engineering Structures, 47, 81-89.

Sladek J, Sladek V, Zhang C, Wunsche M (2010). Crack analysis in piezoelectric solids with energetically consistent boundary conditions by the MLPG, Computer Modeling in Engineering & Sciences, 68(2), 185-219.

Sladek J, Sladek V, Zhang C, Wunsche M (2012). Semi-permeable crack analysis in magneto electro elastic solids, Smart Materials & Structures, 21(2), 025003.

Soares DJ, Sladek V, Sladek J (2012). Modified meshless local Petrov-Galerkin formulations for elastodynamics, International Journal for Numerical Methods in Engineering, 90(12), 1508-1528.

Tanojo E (2007). Derivative of moving least squares approximation shape functions and its derivatives using the exponential weight function, Civil Engineering Dimension, 9(1), 19–24.

Vaghefi R, Hematiyan MR, Nayebi A (2016). Three-dimensional thermo-elastoplastic analysis of thick functionally graded plates using the meshless local Petrov–Galerkin method, Engineering Analysis with Boundary Elements, 71, 34-49.

Wu XH, Tao WQ, Shen SP, Zhu XW (2010). A stabilized MLPG method for steady state incompressible fluid flow simulation, Journal of Computational Physics, 229 (22), 8564-8577.

Wu YL, Liu GR, Gu YT (2005). Application of meshless local Petrov-Galerkin (MLPG) approach to simulation of incompressible flow, Numerical Heat Transfer, part B, 48, 459-475.