 ICCES, vol.5, no.4, pp.211-238

Accuracy and Computational Efficiency of the Finite Volume Method Combined with the Meshless Local Petrov-Galerkin in Comparison with the Finite Element Method in Elasto-static Problem

M.R. Moosavi[1] and A. Khelil[1]

Summary

In this paper, a combined formulation of the Finite Volume Method (FVM) and the Meshless Local Petrov-Galerkin (MLPG) is investigated to solve elasto-static problem. Accuracy and computational efficiency study between the combined formulation and the Finite Element Method (FEM) is presented. Some problems of beam under various loading and boundary conditions are analyzed by the proposed method, and the numerical results are compared with analytical solution and result of other numerical method which is obtained by well-known FEM software ABAQUS. The advantages of the FVM combined MLPG (FVMLPG) with respect to the FEM are illustrated. Higher accuracies and computational efficiencies of the FVMLPG in comparison with the FEM are the most important its benefits.

keywords: Meshless Local Petrov-Galerkin, Finite volume method, Finite element method

Introduction

In computational mechanics many considerable research has been devoted to the development of meshless methods recently. In these methods, the domain of interest is discretized by a scattered set of points. The main objective of meshless or meshfree methods is to get rid of or, at least, alleviate the difficulty of meshing and remeshing the entire structure by only adding or deleting nodes in the entire structure. One of the earliest developments in meshless methods was the SPH method. The foundation of the SPH method is the kernel estimate introduced by Monaghan (1982, 1988) [1, 2]. In this method, partial differential equations, such as conservation laws, are transformed into integral equations, and the kernel estimate then provides the approximation to estimate field variables at discrete points. Other path in the evolution of meshless methods has been the development of generalized finite difference (GFD) method, also called meshless finite difference method. One of the early contributors to the former was Perrone and Kao (1975) [3], but the most robust of these methods was developed by Liszka and Orkisz (1980) [4], using moving least squares (MLS) interpolation. The diffuse element method developed by (Nayroles et al., 1992 [5]) was the first meshless method developed for structural analysis. They proposed a Diffuse Element Method that employs moving

[1]Nancy Université – LEMTA CNRS UMR 7563, IUT Nancy Brabois – Génie Civil, 54601 Villers Les Nancy

least-squares approximation in conjunction with the Galerkin method to provide a mesh-free computational formulation. Belytschko et al. (1994) [6] developed an alternative implementation using moving least squares approximation as defined by Lancaster and Salkauskas (1981) [7]. They called their approach the Element Free Galerkin (EFG) method. In their work, Belytschko and his co-workers have introduced a background cell structure in order to carry out integration by numerical quadrature and Lagrange multipliers to enforce essential boundary conditions. Liu et al. (1995) [8] has recently proposed a different kind of "grid-less" multiple scale methods based on reproducing kernel and wavelet analysis (RPKM method), to improve the accuracy of the SPH method for finite domain problems. In this method, the kernel function is modified by introducing a correction function to meet the reproducing conditions.

On the one hand, Duarte and Oden (1996) [9] and, on the other, Babuska and Melenk (1997) [10] have shown how meshless methods can be based on the partition of unity. In this line, the first authors have developed a new method that they denominate h–p clouds. Oñate et al. (1996) [11] focused on the application to fluid flow problems with a standard point collocation technique. Recently a method based on local sub-domains, rather than a global problem domain, was introduced by Atluri and his colleagues. A truly meshless method, called the Meshless Local Petrov–Galerkin (MLPG) method, has been developed (Atluri and Zhu, 1998, 2000a, 2000b; Atluri et al.) [12, 13, 14]. The MLPG method is based on a local weak form and a moving least square (MLS) approximation. In the MLPG method, an integration method in regularly shaped local domains (such as spheres, rectangles, and ellipsoids) is used. The MLPG method does not need any 'element' or 'mesh' for either field interpolation or background integration. The flexibility in choosing the size and the shape of the local sub-domain leads to a more convenient formulation in dealing with non-linear problems. All these methods can be considered as Finite Point or Meshless Methods.

One of the most important advances in the field of numerical methods was the development of the FEM in the 1950s. In the FEM, a continuum with a complicated shape is divided into elements, finite elements. The individual elements are connected together by a topological map called a mesh. The FEM is a robust and thoroughly developed method, and hence it is widely used in engineering fields due to its versatility for complex geometry and flexibility for many types of linear and non-linear problems. Most practical engineering problems related to solids and structures are currently solved using well developed FEM packages that are commercially available.

However, the FEM has the inherent shortcomings of numerical methods that rely on meshes or elements that are connected together by nodes in a properly

predefined manner. The following limitations of FEM are becoming increasingly evident [15]:

1.1 High cost in creating an FEM mesh
The creating of a mesh for a problem domain is a prerequisite in using any FEM code and package. Usually the analyst has to spend most of the time in such a mesh creation, and it becomes the major component of the cost of a computer aided design (CAD) project. Since operator costs now outweigh the cost of CPU (central processing unit) time of the computer, it is desirable that the meshing process can be fully performed by the computer without human intervention. This is not always possible without compromising the quality of the mesh for the FEM analysis, especially for problems of complex three-dimensional domains.

1.2 Low accuracy of stress
Many FEM packages do not accurately predict stress. The stresses obtained in FEM are often discontinuous at the interfaces of the elements due to the piecewise (or element-wise) continuous nature of the displacement field assumed in the FEM formulation. Special techniques (such as the use of the so-called super-convergence points or patches) are required in the post-processing stage to recover accurate stresses.

1.3 Difficulty in adaptive analysis
One of the current new demands on FEM analysis is to ensure the accuracy of the solution; we require a solution with a desired accuracy. To achieve this purpose, a so-called adaptive analysis must be performed.
In an adaptive analysis using FEM, remeshing (rezoning) is required to ensure proper connectivity. For this remeshing purpose, complex, robust and adaptive mesh generation processors are limited to two-dimensional problems. Technical difficulties have precluded the automatic creation of hexahedron meshes for arbitrary three-dimensional domains. In addition, for three-dimensional problems, the computational cost of remeshing at each step is very expensive analysis requires "mappings" of field variables between meshes in successive stage of the analysis. This mapping process can often lead to additional computation as well as a degradation of accuracy in the solution.

1.4 Limitation in the analyses of some problems

- Under large deformations, considerable loss in accuracy in FEM results can arise from the element distortions.
- It is difficult to simulate crack growth with arbitrary and complex paths which do not coincide with the original element interfaces.

- It is very difficult to simulate the breakage of material with large number of fragments; the FEM is based on continuum mechanics, in which the elements cannot be broken; an element must either stay as a whole, or disappear completely. This usually leads to a misrepresentation of the breakage path. Serious error can occur because the problem is nonlinear and the results and the results path-dependent.

The root of these problems is the use of elements or mesh in the formulation stage. The idea of getting rid of the elements and meshes in the process of numerical treatments has naturally evolved, and the concepts of meshfree or meshless methods have been shaped up.

MLPG approximation

If there is a body Ω with boundary Γ in 2-D coordinates $\mathbf{x}_I = \{x_I^1, x_I^2\}$ with essential and natural boundary conditions in addition a body force as shown in Fig. 1, for a linear elastic body undergoing infinitesimal deformations the governing differential equation can be obtained by the linear momentum balance

$$\sigma_{ij,j} + f_i = 0 \tag{1}$$

with the boundary conditions

$$u_i = \bar{u}_i \qquad \text{on } \Gamma_u \tag{2a}$$

$$t_i = \sigma_{ij} n_j = \bar{t}_i \qquad \text{on } \Gamma_t \tag{2b}$$

where σ_{ij} is the stress tensor, f_i is the body force, $\bar{u}_i$ is the prescribed displacements on the displacement boundary Γ_u and $\bar{t}_i$ is the prescribed tractions on the traction boundary Γ_t, n_j is the unit vector outward normal to the boundary Γ.

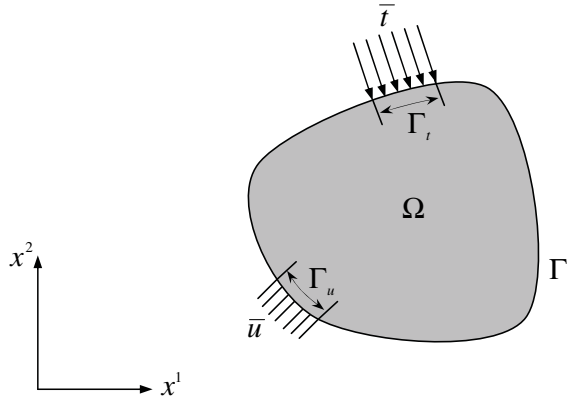


Figure 1: Arbitrary body with displacement and traction boundary conditions

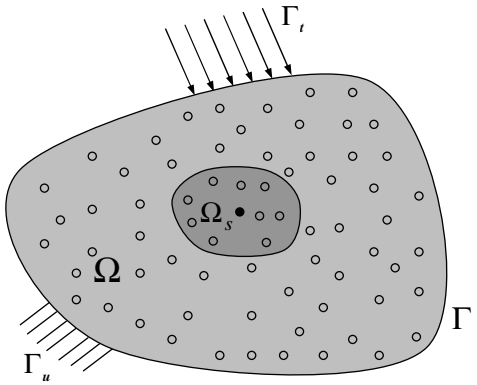


Figure 2: Arbitrary sub-domain around a point

In each numerical method such as the finite element method (FEM), the boundary element method (BEM), the finite difference method (FDM), and the finite volume method (FVM) it is usually considered some sub-domains which in the FEM and the BEM these sub-domains are elements in the FDM and the FVM these are meshes. Solving the problem in sub-domains is a way to approach to the solution in the whole of the domain. In the meshless local Petrov-Galerkin method (MLPG), sub-domain Ω_s is contained only some points which are surrounded in an area or a volume with arbitrary shape, Fig. 2. In general form there is a distribution of field variable in the domain; in other words, each point has a magnitude of field variable which the aim is to determine it. It is better to state here field variable is target of the solution of problem. For example in an elastic body undergoing a traction field variable can be displacement, or reaction force, or etc. To approximate field variable distribution there are some methods; one of the best approximation methods is moving least squares method (MLS). In cause of existence of completeness, continuity and robustness in the MLS, it is possible to interpolate random data with a reasonable accuracy [16]. If field variable be a function such as $u(\mathrm{x})$ with the MLS the distribution of u in Ω_s can be approximated over a number of scattered local points x_I, $(I = 1, 2, 3, \ldots, n)$ as

$$u(\mathrm{x}) = \mathrm{p}^{\mathrm{T}}(\mathrm{x})\mathrm{a}(\mathrm{x}) \quad \forall \mathrm{x} \in \Omega_s \tag{3}$$

where $\mathrm{p}^{\mathrm{T}}(\mathrm{x}) = [p_1(\mathrm{x}), p_2(\mathrm{x}), \ldots, p_m(\mathrm{x})]_{1\times m}$ is a monomial basis of order m, for 2-D problem

$$\mathrm{p}^{\mathrm{T}}(\mathrm{x}) = \left[1, x^1, x^2\right] \qquad \text{linear basis } m = 3 \tag{4a}$$

$$\mathrm{p}^{\mathrm{T}}(\mathrm{x}) = \left[1, x^1, x^2, \left(x^1\right)^2, x^1x^2, \left(x^2\right)^2\right] \qquad \text{quadratic basis } m = 6 \tag{4b}$$

and $\mathrm{a}(\mathrm{x}) = [a_1(\mathrm{x}), a_2(\mathrm{x}), \ldots, a_m(\mathrm{x})]^T_{1\times m}$ is a vector containing coefficients which are functions of the global Cartesian coordinates $[x^1, x^2]^T$, depending on the mono-

mial basis. These coefficients are determined by minimizing a weighted discrete L_2 norm with respect to nodal points, defined as

$$\begin{aligned} J(\mathrm{x}) &= \sum_{I=1}^{m} w_I(\mathrm{x}) \left[\mathrm{p}^T(\mathrm{x}_I)\,\mathrm{a}(\mathrm{x}) - \widehat{\mathrm{u}}^I\right]^2 \\ &= \left[\mathrm{P}\cdot\mathrm{a}(\mathrm{x}) - \widehat{\mathrm{u}}\right]^T \mathrm{W}\left[\mathrm{P}\cdot\mathrm{a}(\mathrm{x}) - \widehat{\mathrm{u}}\right] \end{aligned} \tag{5}$$

where $w_I(\mathrm{x})$ is the weight function associated with the node I, with $w_I(\mathrm{x}) > 0$ for all x in the support of $w_I(\mathrm{x})$, x_I denote the value of x at node I, n is the number of nodes in Ω_s for which the weight functions $w_I(\mathrm{x}) > 0$, the matrices P and W are defined

$$\mathrm{P} = \begin{bmatrix} \mathrm{p}^T(\mathrm{x}_1) \\ \mathrm{p}^T(\mathrm{x}_2) \\ \vdots \\ \mathrm{p}^T(\mathrm{x}_n) \end{bmatrix}_{n\times m} \tag{6}$$

$$W = \begin{bmatrix} w_1(\mathrm{x}) & \cdots & 0 \\ \vdots & \vdots & \vdots \\ 0 & \cdots & w_n(\mathrm{x}) \end{bmatrix}_{n\times n} \tag{7}$$

and

$$\widehat{\mathrm{u}}^T = \begin{bmatrix} \widehat{\mathrm{u}}_1 & \widehat{\mathrm{u}}_2 & \cdots & \widehat{\mathrm{u}}_n \end{bmatrix} \tag{8}$$

Here it should be noted that $\widehat{\mathrm{u}}_i$, $i = 1, 2, \ldots, n$ in Eqs. (5) and (8) are the fictitious nodal values. The stationarity of J in Eq. (5) with respect to $\mathrm{a}(\mathrm{x})$ leads to the following linear relation between $\mathrm{a}(\mathrm{x})$ and $\widehat{\mathrm{u}}_i$.

$$\mathrm{A}(\mathrm{x})\mathrm{a}(\mathrm{x}) = \mathrm{B}(\mathrm{x})\widehat{\mathrm{u}} \tag{9}$$

where the matrices $\mathrm{A}(\mathrm{x})$ and $\mathrm{B}(\mathrm{x})$ are defined by

$$\mathrm{A}(\mathrm{x}) = \mathrm{P}^T\,\mathrm{W}\,\mathrm{P} = \mathrm{B}(\mathrm{x})\mathrm{P} = \sum_{i=1}^{n} w_i(\mathrm{x})\mathrm{p}(\mathrm{x}_i)\,\mathrm{p}^T(\mathrm{x}_i) \quad \forall\, \mathrm{x} \in \Omega_s \tag{10}$$

$$\mathrm{B}(\mathrm{x}) = \mathrm{P}^T\mathrm{W} = \left[w_1(\mathrm{x})\mathrm{p}(\mathrm{x}_1), w_2(\mathrm{x})\mathrm{p}(\mathrm{x}_2), \ldots, w_n(\mathrm{x})\mathrm{p}(\mathrm{x}_n)\right] \quad \forall \mathrm{x} \in \Omega_s \tag{11}$$

The MLS approximation is well defined only when the matrix A in Eq. (9) is non-singular.

The shape function may be found as

$$u(\mathrm{x}) = \mathrm{p}^{\mathrm{T}}(\mathrm{x})\mathrm{A}^{-1}(\mathrm{x})\mathrm{B}(\mathrm{x})\,\widehat{\mathrm{u}} \equiv \Phi^T(\mathrm{x})\widehat{\mathrm{u}} \quad \forall \mathrm{x} \in \Omega_s \tag{12}$$

The weight function in Eq. (5) has a compact circular support, and it defines influence range of node I. In this work weight function is a fourth order spline as below

$$w_I(\mathrm{x}) \begin{cases} 1-6\left(\frac{d_I}{r_I}\right)^2+8\left(\frac{d_I}{r_I}\right)^3-3\left(\frac{d_I}{r_I}\right)^4 & 0 \le d_I \le r_I \\ 0 & d_I \ge r_I \end{cases} \tag{13}$$

where $d_I = |\mathrm{x}-\mathrm{x}_I|$ is the Euclidean distance from node x_I to the point x; and r_I is the size of the support for the weight function w_I which determines the support of node x_I. Since the spline weight function Eq. (13) is C^1 continuous over the entire domain Ω then the shape functions $\Phi(\mathrm{x})$ and the trial function are also C^1 continuous over the entire domain.

Finite volume MLPG (FVMLPG) approach

The finite volume (FV) discretization is based on the integral form of the equation over the control volume or sub-domain Ω_s. In other words, the FV discretization uses the integral form of Eq. (1) over the sub-domain Ω_s around node I as [17, 18, 19]

$$\int_{\Omega_s} (\sigma_{ij,j}+f_i)\, d\Omega = 0 \tag{14}$$

and if the divergence theorem is applied to the first integral term then

$$\int_{\partial\Omega_s} \sigma_{ij} n_j d\Gamma - \int_{\Omega_s} f_i d\Omega = 0. \tag{15}$$

where n_j is the outward normal to the local boundary $\partial\Omega_s$. At this point the conservative nature of the FVM is established as the flux, stress σ_{ij}, is integrated over the local boundary $\partial\Omega_s$.

By considering the traction boundary conditions from Eq. (2) and imposing in Eq. (15) it changes to

$$\int_{L_s} t_i d\Gamma + \int_{\Gamma_{su}} t_i d\Gamma + \int_{\Gamma_{st}} \bar{t}_i d\Gamma - \int_{\Omega_s} f_i d\Omega = 0 \tag{16}$$

In Eq. (16), as shown in Fig. 4, there are two sets local boundaries; one is the boundary which is completely inside of global domain, it is shown by L_s; the other is the boundary which has common side with boundary of global domain, this shared part is indicated by Γ_s. If the common part be on the displacement boundary Γ_u it is calledΓ_{su}, in other words, $\Gamma_{su} = \Gamma_s \cap \Gamma_u$; and if the shared part be on the traction boundary Γ_t it is named Γ_{st} or in other words, $\Gamma_{st} = \Gamma_s \cap \Gamma_t$.

The equation (16) represents a physical meaning in the balance law of the local sub-domain Ω_s as conventional FVM with the traction boundary conditions being

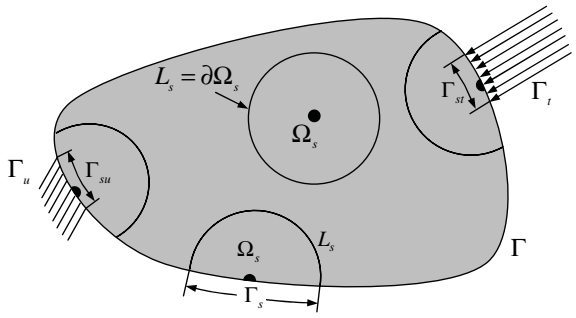


Figure 4: Different types of boundary of local domain

enforced. Hence it is called the finite volume MLPG (FVMLPG) formulation of the equilibrium equation in two-dimensional problem of linear elasticity, Eq. (1).

The constitutive relations of isotropic linear elastic homogeneous solid is used in the tractions of Eq. (2b)

$$t_i = \sigma_{ij} n_j = E_{ijkl} \varepsilon_{kl} n_j \tag{17}$$

the strain-displacement relations for linear elasto-statics problem are

$$\varepsilon_{kl} = \frac{1}{2}\left(u_{k,l} + u_{l,k}\right) \tag{18}$$

The strain can be interpolated with the same shape function in Eq. (12)

$$\varepsilon_{kl}(\mathrm{x}) = \sum_{K=1}^{n} \Phi^{(K)}(\mathrm{x}) \varepsilon_{kl}^{(k)} \tag{19}$$

At now Eq. (16) is discretized by substituting Eq. (17) and (19) in it

$$-\sum_{K=1}^{n}\left[\int_{L_s} \Phi^{(K)}(x) E_{ijkl} n_j d\Gamma\right] \varepsilon_{kl}^{(K)} - \sum_{K=1}^{n}\left[\int_{\Gamma_{su}} \Phi^{(K)}(x) E_{ijkl} n_j d\Gamma\right] \varepsilon_{kl}^{(K)} = \int_{\Gamma_{st}} \bar{t}_i d\Gamma + \int_{\Omega_s} f_i d\Omega \tag{20}$$

The advantage of Eq. (20) is that there is not any shape function derivative in it; because the meshless approximation is not efficient in calculating such derivative everywhere in the domain, especially when the MLS approximation is used. Hence it is the benefit resulting from this work in comparison with the traditional MLPG [primal] displacement method; in other words, in the primal MLPG the displacement is approximated directly therefore the derivative of the shape function will be appeared in the discretized local form. In addition since in the FVMLPG the strain, which is the secondary field variable, is approximated independently of

the displacement the requirement of the completeness and continuity of the shape function is reduced by one-order. In fact with this method requires lower-order polynomial terms in the meshless approximation, and a smaller nodal influence size of local domain, to speed up the shape function calculation. On the other hand, Eq. (20) shows the number of equations is less than the number of the independent strain variables, because the nodal strain variables are more than the displacement ones for example in 2D problem there are three nodal-strain variables, but only two displacement nodal variables. It is possible to reduce number of variables by transforming the strain variables back to the displacements without any changes to Eq. (20). Hence by applying Eq. (19) only at each nodal point x_I instead of the entire solution domain the nodal strain variables are expressed in terms of the nodal displacement variables

$$\varepsilon_{kl}(\mathrm{x}_I) = \varepsilon_{kl}^I = \frac{1}{2}(u_{k,l} + u_{l,k}) \quad I = 1, 2, \ldots, n \tag{21}$$

Now with the displacement approximation

$$u(\mathrm{x}) = \sum_{J=1}^{n} \phi^J(\mathrm{x})\hat{u}^J \tag{22}$$

the two sets of nodal variables can be transformed through a linear algebraic matrix

$$\widehat{\varepsilon}_{kl}^I = \widehat{H}_{klm}^{IJ}\widehat{u}_m^J \tag{23}$$

where the transformation matrix H is banded. After substituting Eq. (23) in (20) the relation between displacement and force is obtained as

$$K\,\mathrm{u} = \mathrm{f} \tag{24}$$

where

$$K_{IJ} = \left[\int_{L_s} \Phi^{(I)}(x) E_{ijkl} n_j d\Gamma - \int_{\Gamma_{su}} \Phi^{(I)}(x) E_{ijkl} n_j d\Gamma\right] H_{klm}^{(I)(J)} \tag{25}$$

is the stiffness matrix, u is the displacement vector, and

$$\mathrm{f}_I = \int_{\Gamma_{st}} \bar{t}_i d\Gamma + \int_{\Omega_s} f_i d\Omega \tag{26}$$

is the force vector.

Imposition of boundary conditions

In this section it is discussed how the boundary conditions, given by Eqs. (2a) and (2b), can be incorporated efficiently. With the meshless approximation, another problem is that the essential boundary conditions can not be imposed directly. In the FVMLPG neither penalty parameters nor Lagrange multipliers appear in the equations because essential boundary conditions can be imposed by the collocation method. For a nodal point $\mathrm{x}^{(I)}$, if its ith displacement DOF belongs to the displacement boundary, i.e., $u_i^{(I)} \in \Gamma_{su}$, the corresponding system equation can be replaced by the one generated from the collocation for this particular DOF, as

$$\alpha u_i\left(\mathrm{x}^{(I)}\right) = \alpha \bar{u}_i\left(\mathrm{x}^{(I)}\right) \tag{27}$$

This standard collocation still keeps the system equations sparse and banded.

Numerical integration

Numerical integration is an important ingredient of a meshless method. In order to integrate it needs the integrals to be evaluated over the boundary of local sub-domains. By definition the tractions, $\bar{t}_i = \sigma_{ij} n_j$, contain the normal n to the boundary it involves the trigonometric functions. Polynomial expressions can not be expected over the entire local boundary, which may not be covered by all local nodes. Therefore it is not possible to apply the conventional numerical quadrature schemes are designed for polynomials in trigonometric functions, because however a 2-point Gauss quadrature evaluates $\int_0^1 x^3 dx$ accurately the numerical experiments show that, it gives an error of about 0.07% when 4-point Gauss quadrature is used to evaluate one dimensional integral $\int_0^1 x^3 \sin 2\pi x dx$ over a 2D circle, in which only one trigonometric function is involved.

There are two ways for controlling the numerical errors; one is to increase the order of Gauss quadrature scheme, and another is to subdivide the domain of the integration into small segments for better accuracy. The numerical experiments show the second way is much more efficient than the first one. For those nodes on the global boundary consider a local circular sub-domain centered at node I, $\mathrm{x}^{(I)}$, with a radius denoted by $r_0^{(I)}$. By drawing a line from node I to its neighbor node J $\left\{\mathrm{x}^{(J)}\right\}$, $(J = 1, 2, \ldots, m)$, a point can be obtained at the intersection between the line and the local circle, denoted by $\left\{\mathrm{y}^{(J)}\right\}$, $(J = 1, 2, \ldots, m)$. A subset of these intersecting points is used to divide integration domain, i.e. the local circle. It should be pointed out that the intersection points between the local and the global boundaries are automatically included in $\mathrm{y}^{(J)}$.

The shape of sub-domain is circle and its size is related the nodal distance. Normally, it should be so great to make sure that there are enough points to support the nodes on the global boundary. The size of the sub-domain affects the accuracy of the solution and the efficiency of the method.

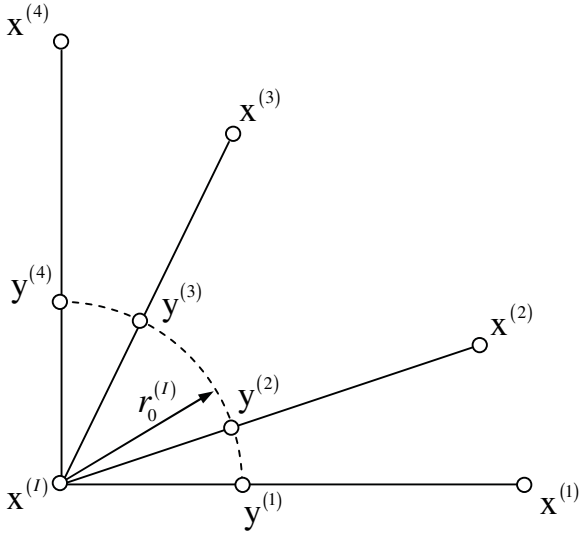


Figure 5: Numerical integration over local boundary

Implementation routine

The implementation of the method can be carried out according to the following routine [12]:

1. Choose a finite number of nodes in the domain Ω and on the boundary Γ of the given physical domain; decide the basis functions and weight functions such that the MLS approximation is well defined.
2. Determine the local sub-domain Ω_s and its corresponding local boundary $\partial\Omega_s$ for each node.
3. Loop over all node located inside the global domain and at the global boundary Γ.

- Determine Gaussian quadrature points x_Q in Ω_s and on $\partial\Omega_s$.
  - Loop over quadrature points x_Q in the sub-domain Ω_s and on the local boundary $\partial\Omega_s$
    (a) determine the nodes x_i located in the domain of definition of the MLS approximation for the trial function at point x_Q, i.e., those nodes with $w_i(\mathrm{x}_Q) > 0$;
    (b) for those nodes in the domain of the definition of the MLS approximation of trial function at point x_Q; calculate $\phi_i(\mathrm{x}_Q)$;
    (c) evaluate numerical integrals
    (d) assemble contributions to the linear system for all nodes in K, f;
    (e) End quadrature point loop

4. End node loop.
5. Solve the linear system for the fictitious nodal values.
6. Calculate the value of the unknown variable at those sample points under consideration.

Numerical examples

This method is evaluated by the following three examples. Those are presented to illustrate the implementation, accuracy and efficiency of the present FVMLPG approach.

Example 1 Cantilever beam subjected to end shear

A cantilever beam with a transverse load is considered as shown in Fig. 6. The characters of beam are L length, c half width, E module of elasticity; υ Poisson's ratio; P transverse load.

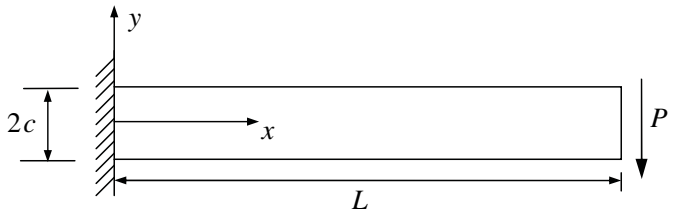


Figure 6: A cantilever beam under a transverse load

This example is solved in state of plane stress with the following properties:

$$L=24, \quad c=2, \quad E=1, \quad \upsilon=0.25, \quad P=1.$$

For displacement boundary conditions

$$u_x(0,0)=u_y(0,0)=0 \text{ and } u_x(0,c)=u_x(0,-c)=0$$

The exact solution for this problem is given in Timoshenko and Goodier [20] as:

$$u_x=-\frac{Py}{6EI}\left[3x(2L-x)+(2+\upsilon)\left(y^2-c^2\right)\right] \tag{28a}$$

$$u_y=\frac{P}{6EI}\left[x^2(3L-x)+3\upsilon(L-x)y^2+(4+5\upsilon)c^2x\right] \tag{28b}$$

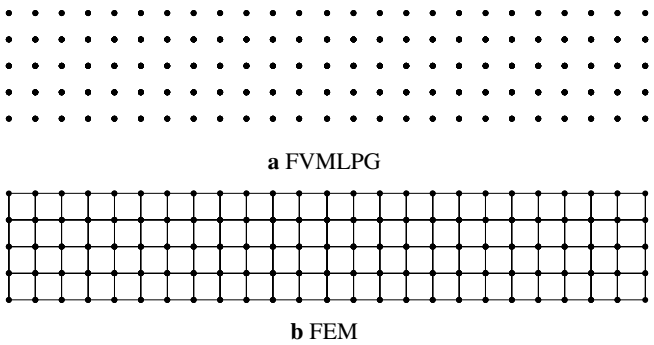
where I is the area moment of inertia of the beam and it is given as $I=2tc^3/3$ where t is a constant beam thickness where in this example $t=1$. The corresponding stresses are

$$\sigma_x=-\frac{P}{I}(L-x)y \tag{29}$$

$$\sigma_y=0 \tag{30}$$

$$\sigma_{xy}=-\frac{P}{2I}\left(y^2-c^2\right) \tag{31}$$

In order to perform a FVMLPG solution to the problem it needs to consider some nodes which are selected by numerical experiment. Here regular uniform nodal configurations with nodal distances, d, of 2.0, 1.0 and 0.5 are used. The non-dimensional number d is the distance between two nodes neighbourhood in x or y direction which determined by division the beam length by the number of nodes in the direction of the beam length. The number of nodes are 39, 125 and 441 respectively. For comparison purpose, FE meshes are also constructed from the nodal configurations by using the 4-node bilinear plane stress quadrilateral, incompatible modes element of the well-known FE software ABAQUS. In other words, the FE meshes are 12×2, 24×4 and 48×8 in x and y direction respectively. This kind of element gives the best results for this problem with respect to other kind of elements of ABAQUS. The FVMLPG nodal configuration and the FE mesh with nodal distance 1.0 are shown in Fig. 7.



a FVMLPG

b FEM

Figure 7: **a** The FVMLPG nodal configuration with nodal distance $d = 1.0$. **b** the FE mesh with nodal distance $d = 1.0$

The normalized vertical displacement of the beam with nodal distance 0.5 is shown in Fig. 8. The essential boundary conditions are imposed only in three points; one point in the origin of the coordinate system, middle of the left side of the beam which is fixed in two directions of x and y while two other points in the top edge and the bottom edge of the left side of the beam which are fixed only in $x-$direction.

A shear load is applied to the free end of the cantilever beam and the problem is solved by using the FVMLPG. The numerical results of the shear deformation are shown in Figure 9, which agree with the analytical solution very well. The maximum relative vertical displacement error of the FVMLPG is 0.0001389% but the error of the FEM is 0.07661% which the both of them occur in nodal distance of 0.5. It is clear from Fig. 9 the accuracy of the FVMLPG in vertical displacement is very better than the one of the FEM which is obtained by ABAQUS in this example.

The time of run of computer code or software versus the accuracy indicates

computational efficiency. Here the computational efficiency in the maximum vertical displacement of two methods is indicated in Fig. 10. It is clear CPU time or time consumption of the FVMLPG in nodal distances $d = 2.0$ (39 node) and $d = 1.0$ (125 node) is lower than the one of the FEM and also the accuracy of the

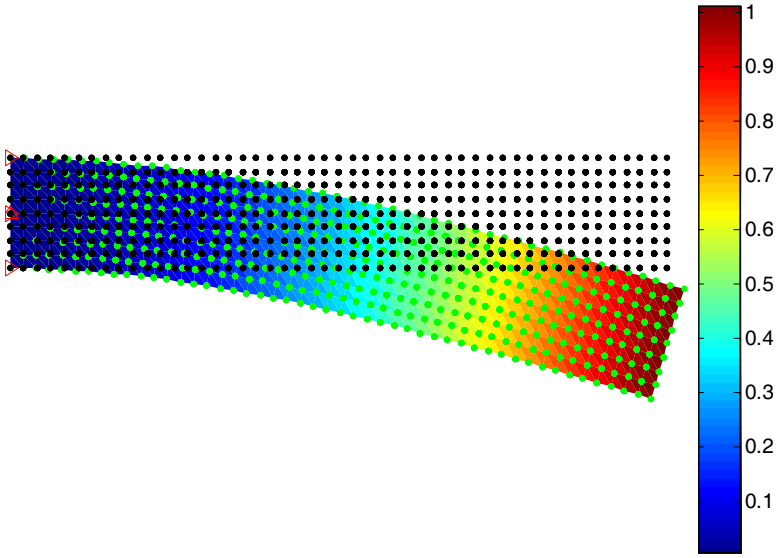


Figure 8: Normalized vertical displacement of a cantilever beam under a shear load (441 nodes)

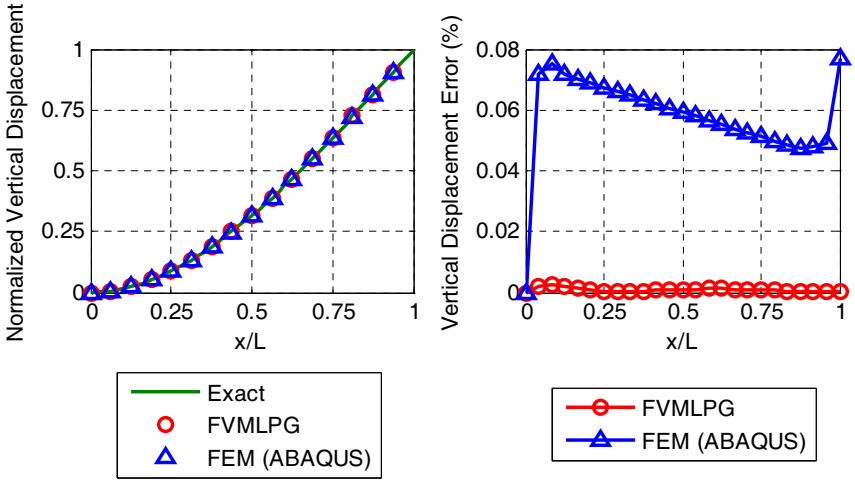


Figure 9: Normalized vertical displacement and its error of the beam by the FVMLPG & FEM

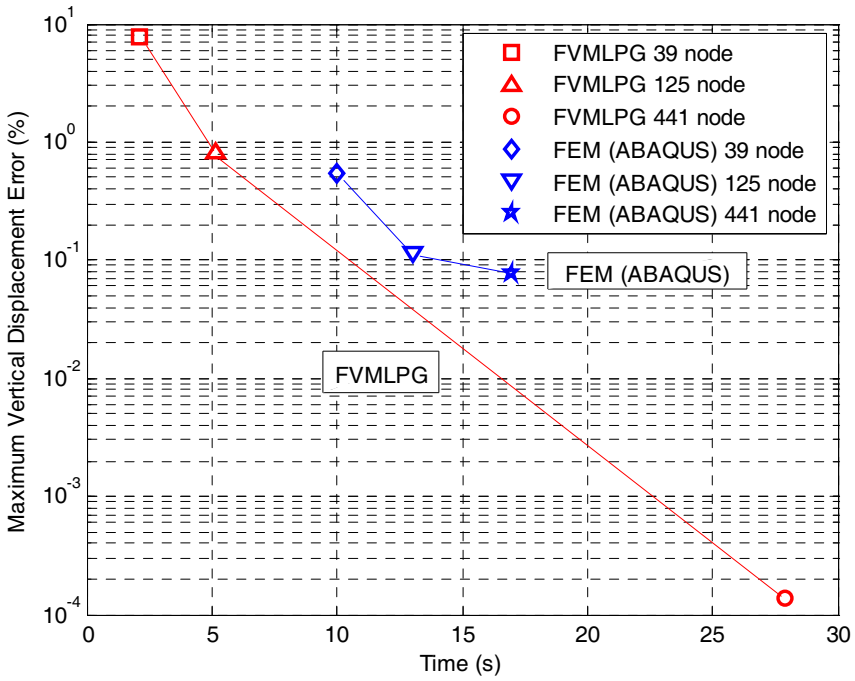


Figure 10: Computational efficiency of the FVMLPG and the FEM in the maximum vertical displacement

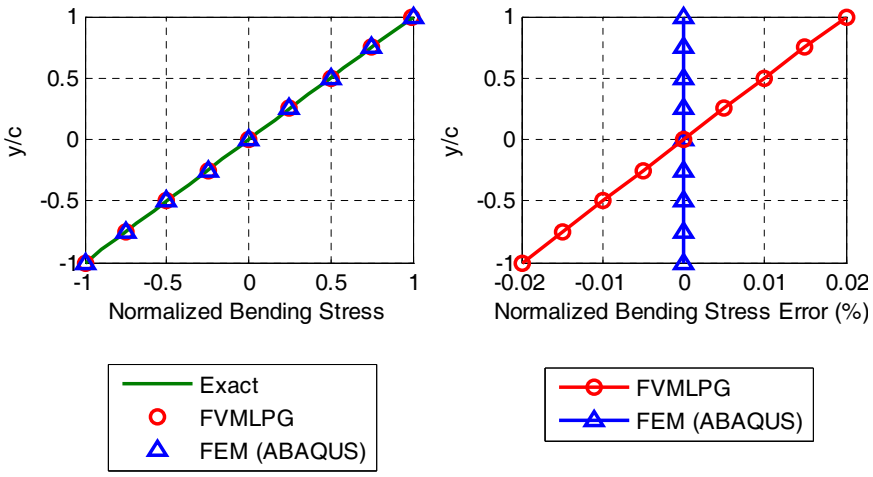


Figure 11: Normalized bending stress and its error in the middle section of the beam, $x = L/2$ by the FVMLPG & FEM

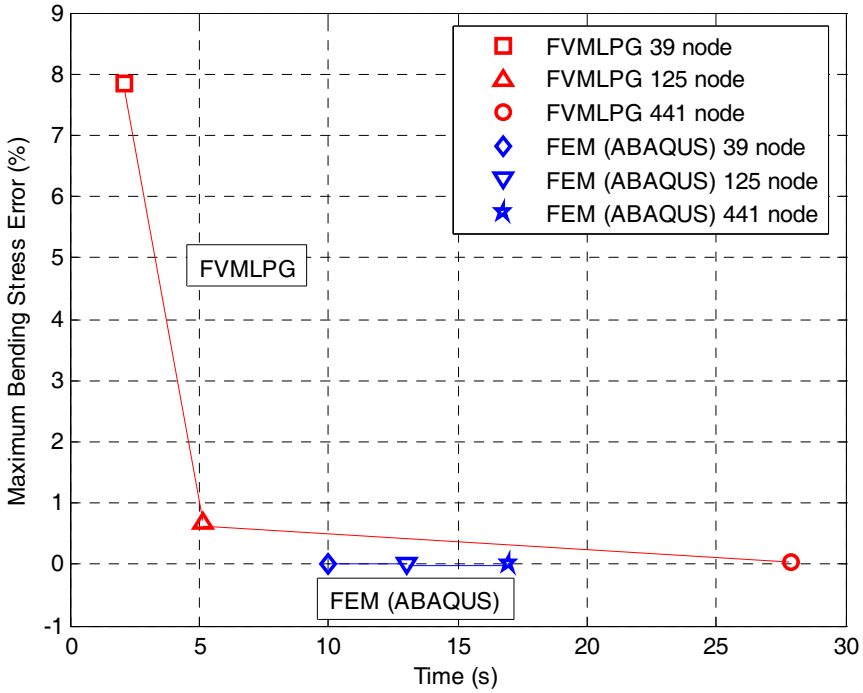


Figure 12: Computational efficiency of the FVMLPG and the FEM in the maximum bending stress in the middle section of the beam, $x = L/2$

FVMLPG in nodal distances 2.0 and 1.0 is lower than the corresponding accuracy of the FEM, but the accuracy of the FVMLPG in nodal distance $d = 0.5$ (441 node) is very higher than the one of the FEM; however its CPU time is high.

In addition the results of the normalized bending stress in the section of the middle of the beam, $x = L/2$, by the FVMLPG and the FE are lied on the exact solution [21], as shown in Fig. 11. The maximum error in the FVMLPG is 0.02%, and in the FEM that is 0%; both of these errors are in nodal distance 0.5.

In bending stress the computational efficiency of this method, as shown in Fig. 12 is lower than the efficiency of the FEM because the accuracy of the FVMLPG in nodal distance 2.0 and 1.0 is lower than the ones of the FEM. The FEM gives exact solution in bending stress in three nodal distances.

Finally in Fig. 13 the normalized shear stress distribution in the middle section of the beam, $x = L/2$, in the FVMLPG is well but in the FEM there is some deviation near the upper and lower boundaries of the beam. The minimum shear stress error of the FVMLPG is 0.01251%and the maximum error is 0.7086% but in the FE the minimum shear stress error is 2.083% and the maximum error is 22.92% on the upper and lower boundaries nodes. There are two advantages for the FVMLPG

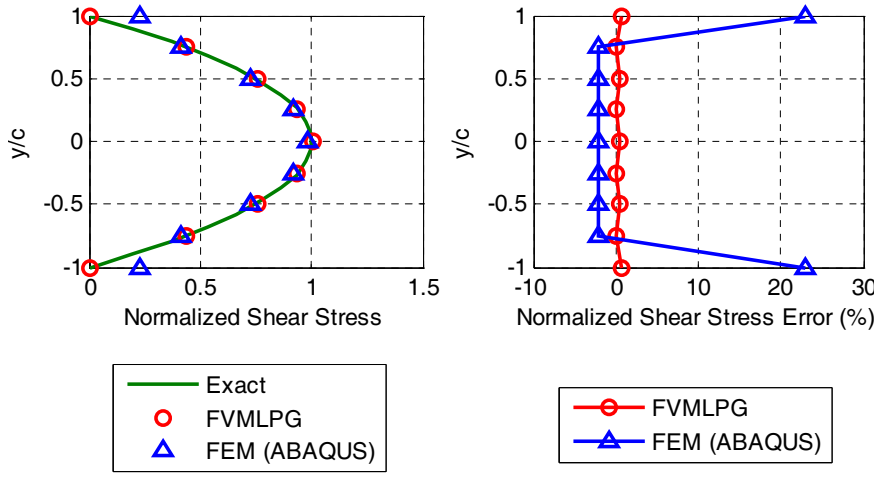


Figure 13: Normalized shear stress and its error in the middle section of the beam, $x = L/2$ by the MPLGFV & FEM

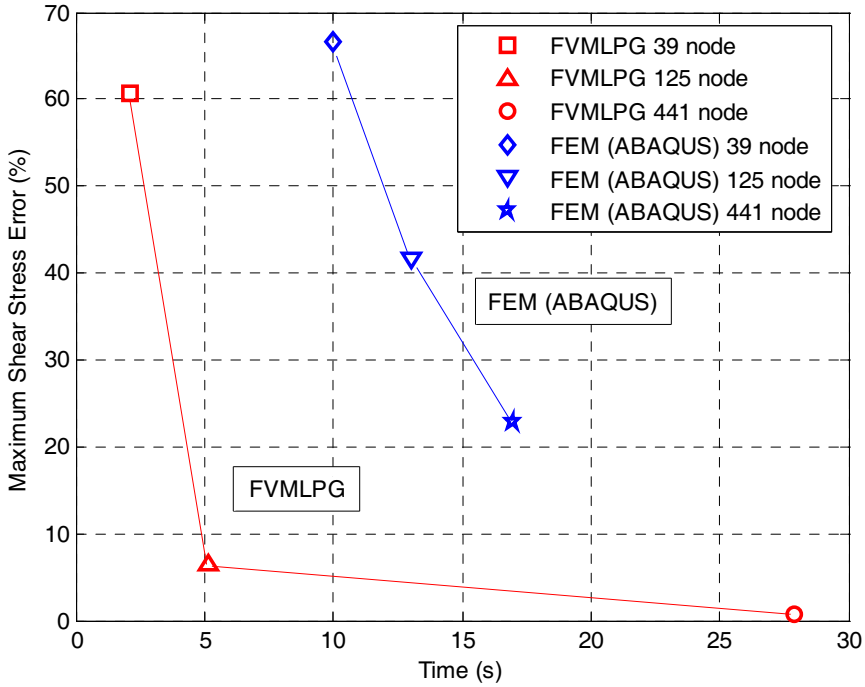


Figure 14: Computational efficiency of the FVMLPG and the FEM in the maximum shear stress in the middle section of the beam, $x = L/2$

results, the first is lower error and the second is no high deviation in the results near the boundaries which are clear in Fig. 13.

The study of computational efficiency of shear stress shows with increasing nodal distance shear stress error of the FVMLPG decreases better than the one of the FE. In Fig. 14, the minimum error of the maximum shear stress in the nodal distance 0.5 is 0.7086% however this error in the FE is 22.92%. At all the shear stress computational efficiency of the FVMLPG is higher than the one of the FEM.

Example 2 Cantilever beam subjected to compression load

The second example is a cantilever beam under compression load as Fig. 15.

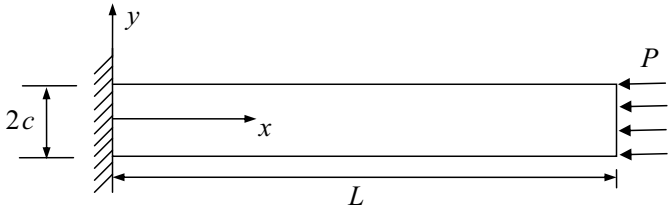


Figure 15: A cantilever beam under a compression load

This example is solved for plane stress case with following magnitudes:

$$L = 24, \quad c = 2, \quad E = 1, \quad \upsilon = 0.25, \quad P = 1$$

which by the two methods, the FEM and the FVMLPG method the results are obtained in non-dimensional forms. The normalized axial displacement obtained from the FVMLPG is very better than the one of the FEM. Primary geometry of the beam and its normalized axial displacement are shown in Fig. 16. The nodal distance of the two methods is 0.5.

As shown in Fig. 17 the maximum error in the axial displacement by the FVMLPG is $6.14 \times 10^{-10}\%$ however, this value of the FEM is 0.1546%. The accuracy of the FVMLPG in axial displacement of this example is very well in comparison with the one of the FEM which is obtained by ABAQUS.

Computational efficiency of two methods in the axial displacement is shown in Fig. 18. The computational efficiency of the FVMLPG in the axial displacement is very higher than the one of the FEM because of the higher accuracy of this method. However, the time of running of the FVMLPG is higher than the one of the FEM but high accuracy of FVMLPG makes up this time delaying in return.

The normal stress in normalized form in the middle section of the beam, $x = L/2$ and its error with respect to exact solution are shown in Fig. 19. Both methods give very well results; there is only a very small deviation in the FEM results in the

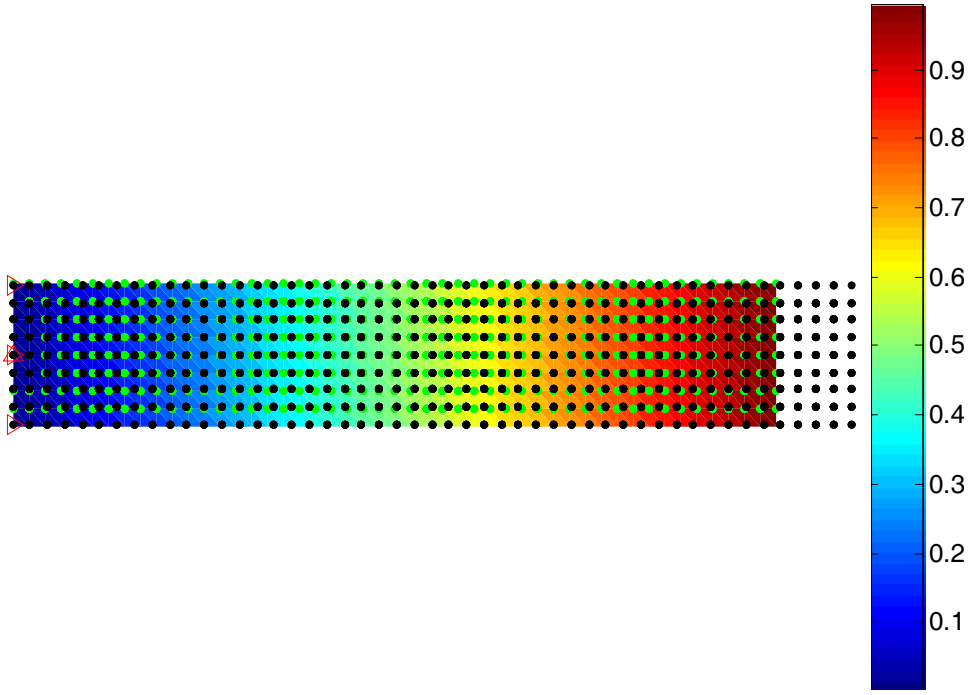


Figure 16: Normalized axial displacement of a cantilever beam under a compression load (441 nodes)

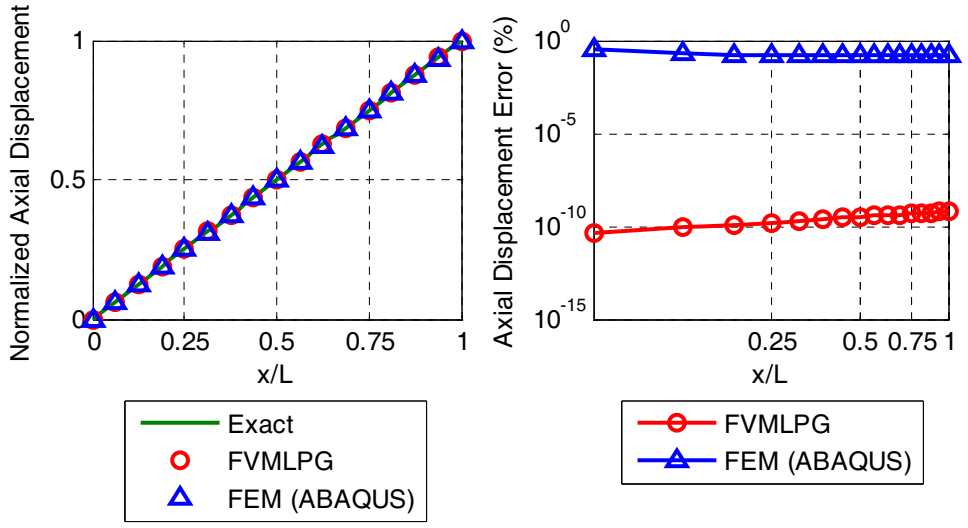


Figure 17: Normalized axial displacement and its error of the beam by the FVMLPG & FEM

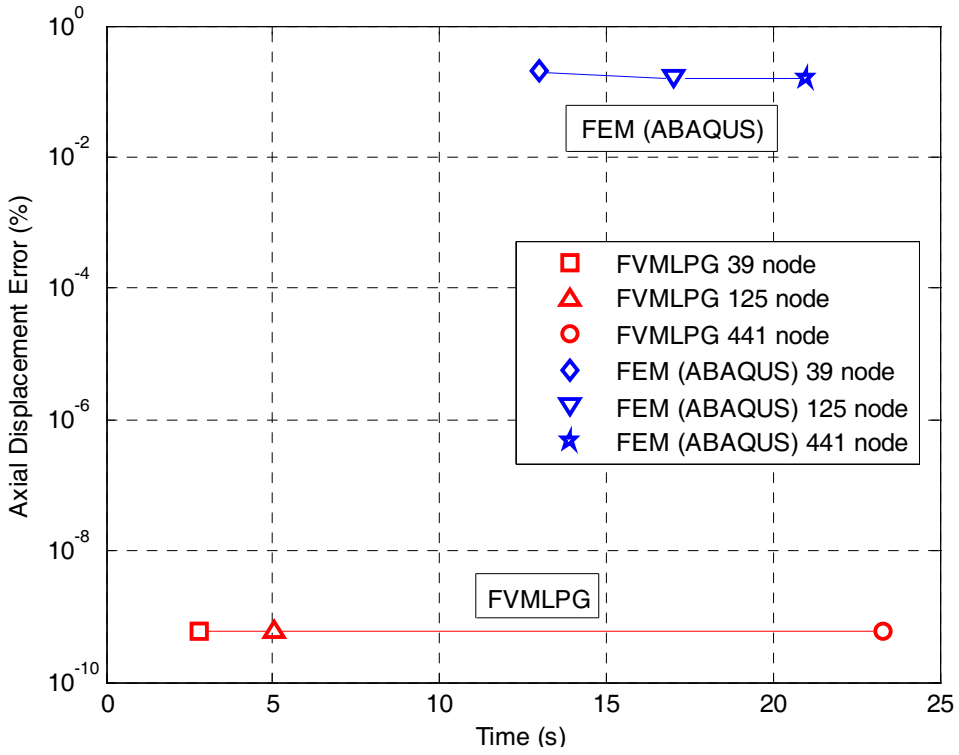


Figure 18: Computational efficiency of the FVMLPG and the FEM in the axial displacement

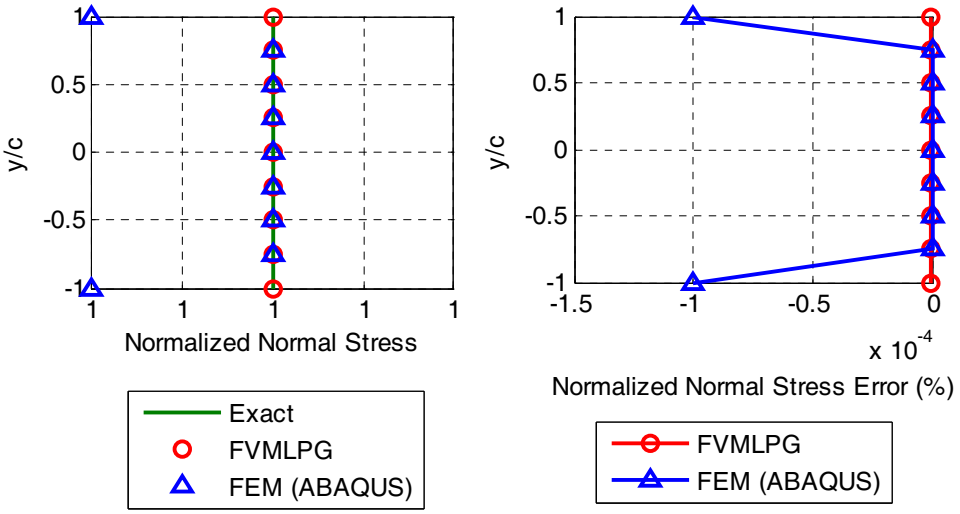


Figure 19: Normalized normal stress and its error of the beam by the FVMLPG & FEM in the middle section of the beam, $x = L/2$

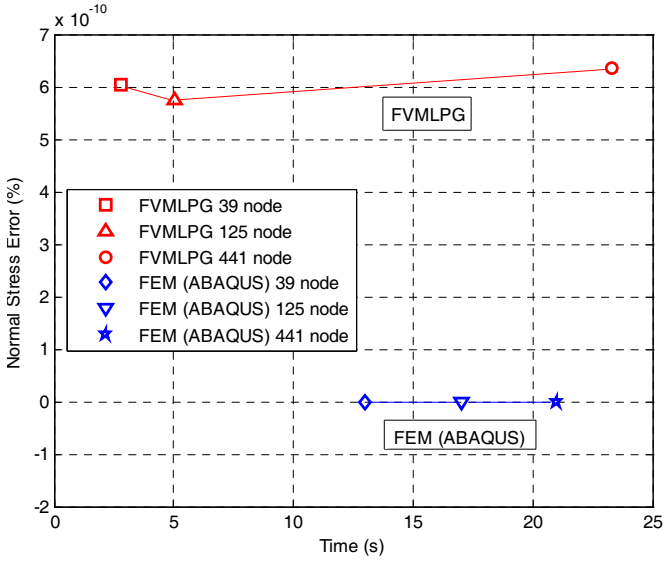


Figure 20: Computational efficiency of the FVMLPG and the FEM in the normal stress

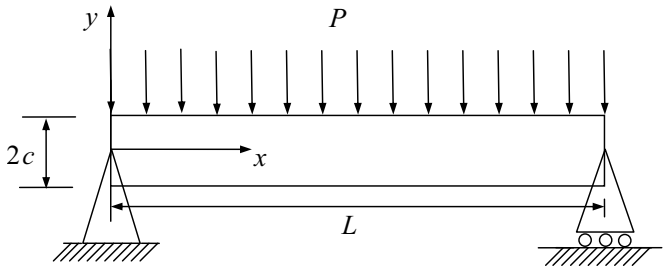


Figure 21: A simply supported beam with uniform distributed lateral load

upper and lower surface of the beam which it is not important because its value is 0.0001%. This deviation is not appeared in the FVMLPG result which is benefit of this method.

Computational efficiency of two methods in normal stress is shown in Fig. 20. Since accuracy of two methods is very close to each other then there is not a very important difference between computational efficiency of these methods.

Example 3 Simply supported beam with uniform distributed lateral load

The third example is a simply supported beam with uniform distributed lateral load, as shown in Fig. 21.

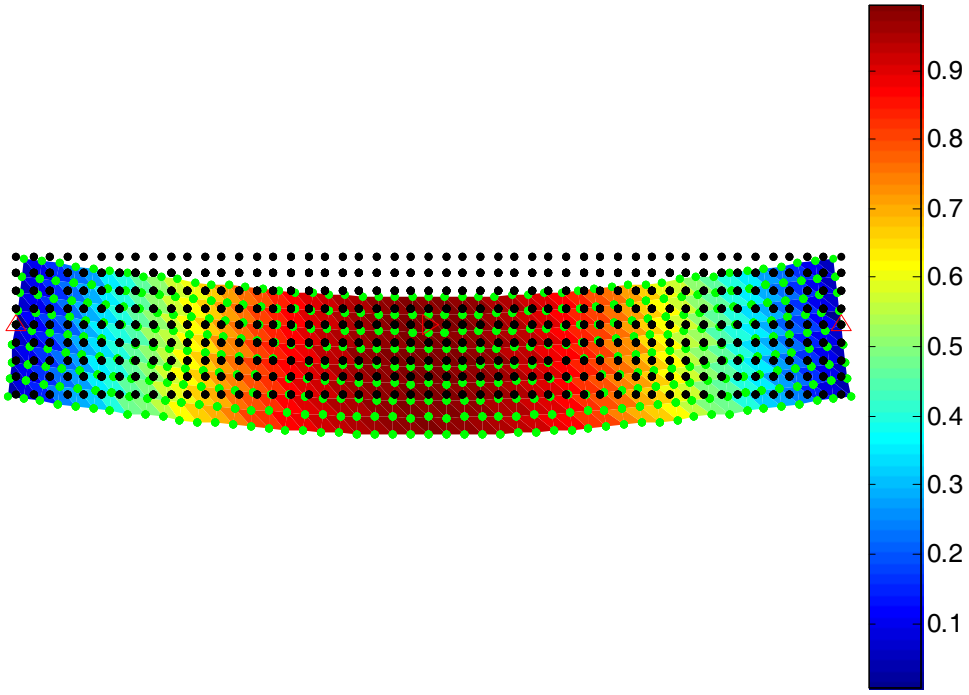


Figure 22: Normalized lateral displacement of a simply supported beam under a uniform distributed lateral load (441 nodes)

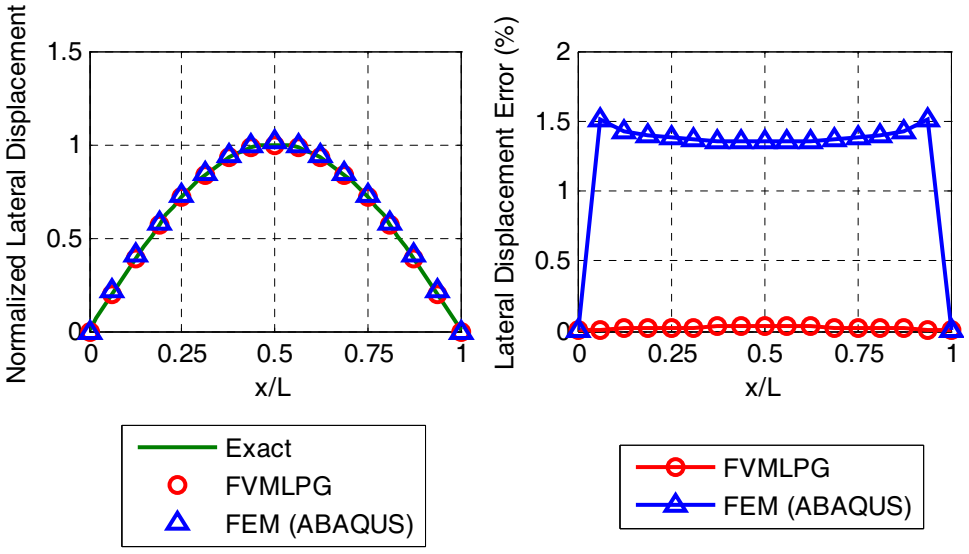


Figure 23: Normalized lateral displacement and its error of a simply supported beam by the FVMLPG & FEM

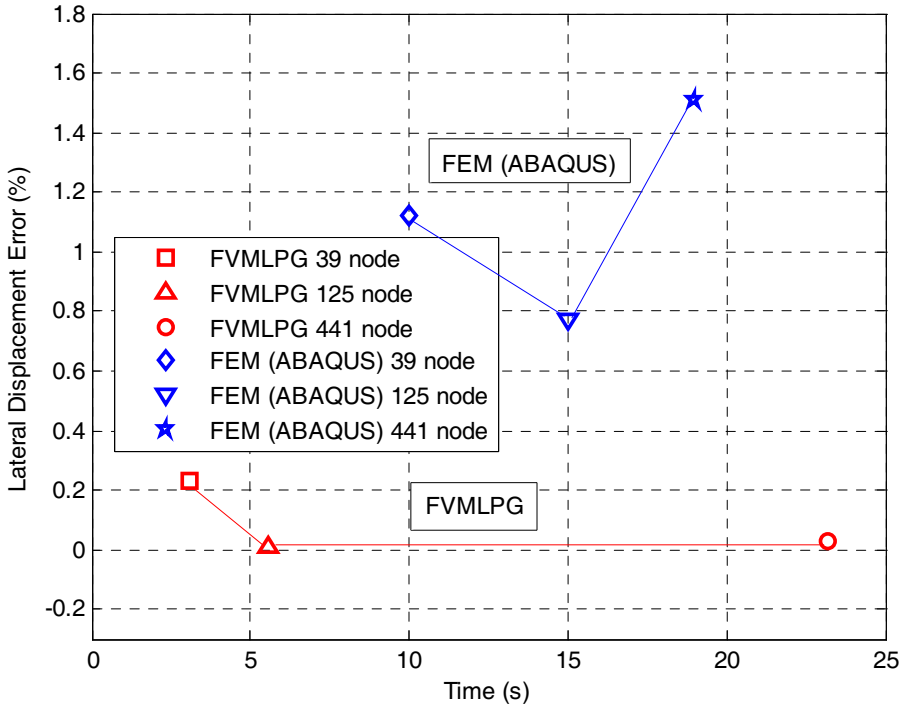


Figure 24: Computational efficiency of the FVMLPG and the FEM in the lateral displacement

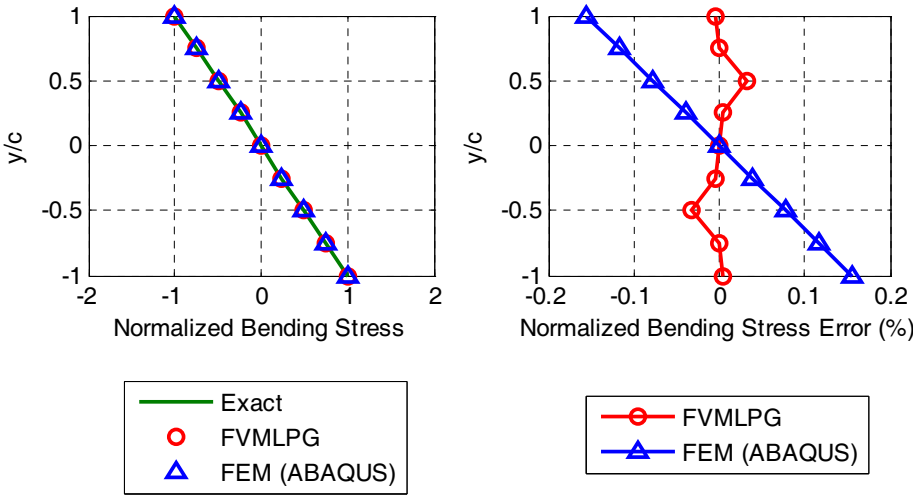


Figure 25: Normalized bending stress and its error of the simply supported beam by the FVMLPG & FEM

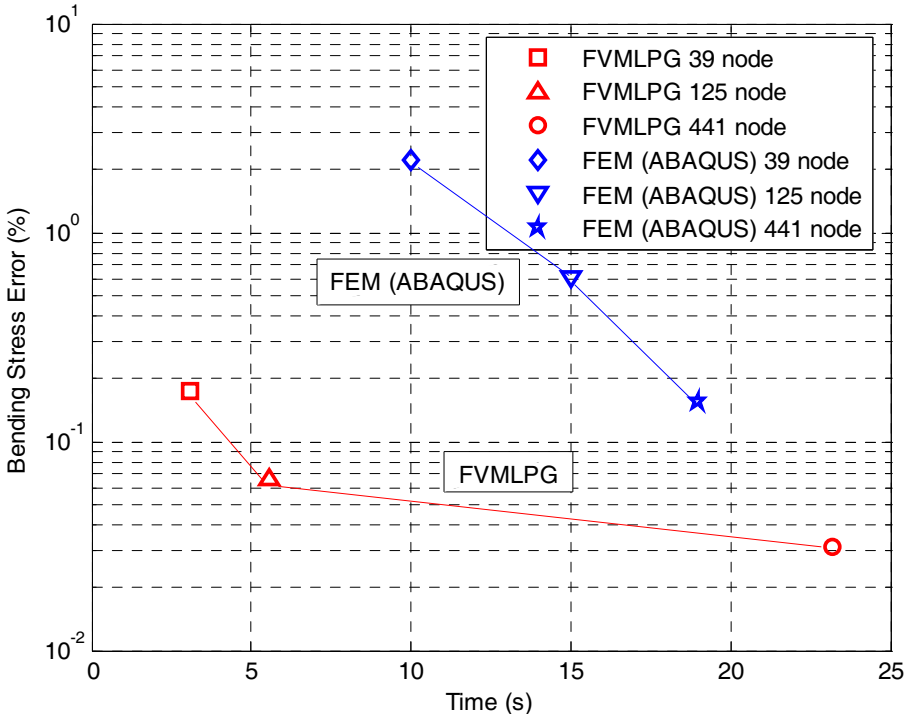


Figure 26: Computational efficiency of the FVMLPG and the FEM in the bending stress in the middle section of the beam, $x = L/2$

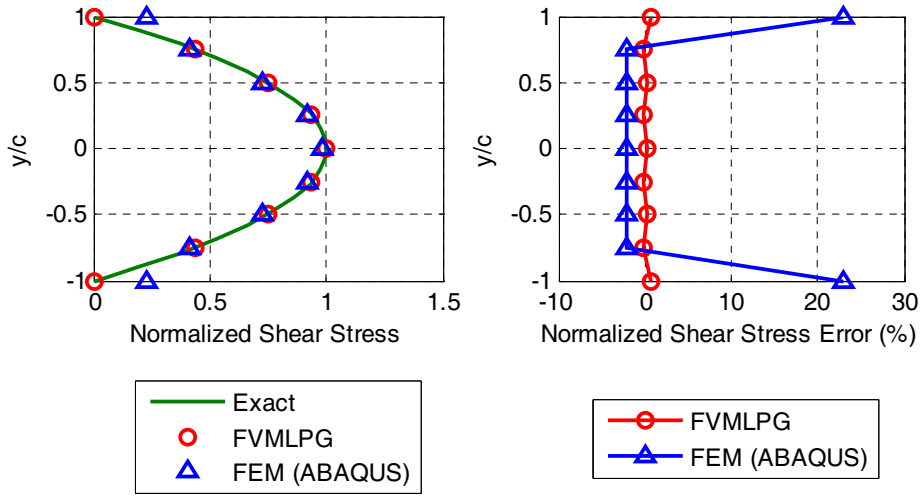


Figure 27: Normalized shear stress and its error of the simply supported beam by the FVMLPG & FEM

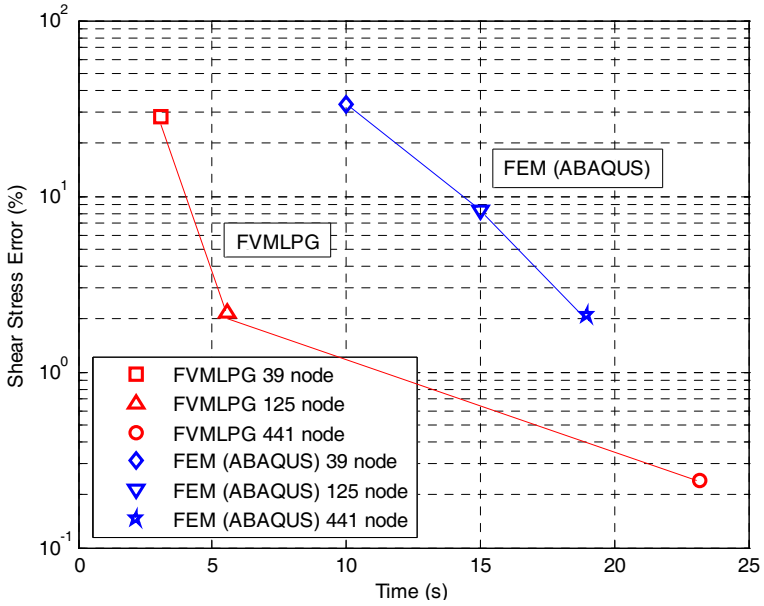


Figure 28: Computational efficiency of the FVMLPG and the FEM in the shear stress in the middle section of the beam, $x = L/2$

This example is solved for plane stress case with following magnitudes:

$$L = 24, \quad c = 2, \quad E = 1, \quad \upsilon = 0.25, \quad P = 1$$

The lateral displacement of beam obtained by the FVMLPG is shown in Fig. 22.

The maximum lateral displacement error obtained by the FVMLPG is 0.03078% with respect to the maximum lateral displacement of exact solution, as shown in Fig. 23.

After running this example in the FEM the maximum lateral displacement error is 1.507%, as shown in Fig. 23. It is clear the difference between errors of two methods is high. This shows the accuracy of the FVMLPG with respect to the FE in lateral displacement of this example. The computational efficiency of these methods in lateral displacement is shown in Fig. 24. It is clear the computational efficiency of the FVMLPG is better than the one of the FEM.

In bending stress results of two methods lay in the analytical solution but the result of the FVMLPG has minimum error with respect to the result of the FEM. Maximum error of bending stress which is obtained by the FVMLPG is 0.03153% and its minimum error is 0.0009071% however maximum error of bending stress which is obtained by the FEM is 0.1555%; this error is error in the upper and lower plane of the beam while this error in the FVMLPG is only 0.003505%. In the error of the FVMLPG there are two nodes which have a jumping in their bending stress

errors.

Computational efficiency of the FVMLPG and the FEM in the bending stress in the middle section of the beam, $x = L/2$ is shown in Fig. 26. Accuracy of two methods increases by decreasing in nodal distance but accuracy of the FVMLPG is higher than the one of the FEM however the CPU time of the FVMLPG in nodal distance 0.5 is greater than the corresponding CPU time of the FEM.

Behavior of two methods in shear stress is indicated in Fig. 27. It shows a good agreement between the result of the FVMLPG and the exact solution with respect to the result of the FEM. The maximum error of the FVMLPG is 0.6127% however, the error of the FEM is 2.085%. In addition there is a great deviation in shear stress of the element beside the upper and lower surface of the beam while this deviation of the FVMLPG is very small.

The accuracy of the FVMLPG in shear stress is very better than the one of the FEM. These good results in shear stress are obtained in nodal distance 0.5. Also accuracy of the FVMLPG in nodal distance 1.0 is higher than the accuracy of the FEM; at all the computational efficiency of the FVMLPG in shear stress is better than the one of the FEM, as shown in Fig. 28.

Conclusion

In this paper, a combined formulation (FVMLPG) of the Finite Volume Method (FVM) and the Meshless Local Petrov-Galerkin (MLPG) is developed to solve elastostatic problems by using the FVM directly in these problems. For the meshless interpolation scheme in this method, the conventional moving least squares interpolation is applied. With the FVMLPG, by choosing the strains as independent variables in the local weak form it is not necessary to differentiate the shape function because of interpolating the strains directly. In addition the continuity requirement on the trial function reduces by one-order then it is possible to use a smaller support size in the meshless approximations with a lower-order polynomial basis. This procedure is implemented in a computer code in MATLAB. Accuracy and computational efficiency of the FVMLPG is investigated with analytical solution of some examples in elasto-static. In addition performance of this method is compared with the performance of the traditional FEM by using well-known FE software ABAQUS in the same examples. In the first example, a cantilever beam under a tip concentrated load is analyzed by the present method. The comparisons between two methods shows the FVMLPG is very more accurate than the FEM in vertical displacement so that its result in vertical displacement is very close to the analytical solution of this example. The result of the bending stress of the FVMLPG has a small error with the exact solution but by the FEM there is not error in bending stress. In investigation of shear stress in this example, the result of the FVMLPG has very good agreement with the exact solution with respect to the one of the

FEM. In part of the computational efficiency the FVMLPG has a higher efficiency with respect to the FEM, except in the bending stress. In the second example, a cantilever beam with a compression force, the results of the FVMLPG in axial displacement is more accurate than the one of the FEM and in normal stress two methods have the same good results with respect to the analytical solution. Computational efficiency of the FVMLPG in axial displacement is very higher than the efficiency of the FEM but the normal stress efficiency of the FVMLPG is almost in the same order of the one of the FEM. In the third example, a simply supported beam under uniform lateral load, the FVMLPG lateral displacement results is very more accurate than the FEM results; also in bending and shear stress the FVMLPG gives very good results in comparison to the one of the FEM. At all the complexity and high computational time are the main barriers for the meshless approaches to fully fulfil their application potentials. In the FVMLPG method is tried to keep a balance the accuracy and efficiency. Only the run time in the FVMLPG is more than the FEM.

References

1. Monaghan, J.J. (1982): Why particle methods work. *SIAM J. Sci. Statist. Comput.* 3, 422–433.
2. Monaghan, J.J. (1988): An introduction to SPH. *Comput. Phys. Comm.* 48, 89–96.
3. Perrone, N.; Kao, R. (1975): A general finite difference method for arbitrary meshes. *Comp. Structures* 5, 45–58.
4. Liszka, T.; Orkisz, J. (1980): The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Computer and Structures* 11, 83–95.
5. Nayroles, B.; Touzot, G.; Villon, P. (1992): Generalizing the finite element method: diffuse aproximation and diffuse elements. *Comput. Mech.* 10, 307–318.
6. Belytschko, T.; Lu, Y.Y.; Gu, L. (1994): Element-free Galerkin method. *Int. J. Numer. Methods Engrg.* 37, 229–256.
7. Lancaster, P.; Salkauskas, K. (1981): Surfaces generated by moving least squares methods. *Math. Comput.* 37, 141–158.
8. Liu, W.K.; Jun, S.; Li, S.; Adee, J.; Belytschko, T. (1995): Reproducing kernel particle methods for structural dynamics. *Int. J. Numer. Methods Engrg.* 38, 1655–1679.
9. Duarte, A.; Oden, J.T. (1996a): H-P cloud – An h-p meshless method. *Numer. Methods Partial Differential Equations* 12, 673–705.

10. Babuska, I.; Melenk, J.M. (1997): The partition of unity method. *Int. J. Numer. Methods Engrg.* 40, 727–758.

11. Oñate, E.; Idelsohn, S.; Zienkiewicz, O.C.; Taylor, R.L. (1996): A finite point method in computational mechanics. Aplications to convective transport and fluid flow. *Int. J. Numer. Methods Engrg.* 39, 3839–3866.

12. Atluri, S.N.; Zhu, T. (1998): A new meshless local Petrov–Galerkin (MLPG) approach in computational mechanics. *Comput. Mech.* 22, 117–127.

13. Atluri, S.N.; Zhu, T. (2000a): New concepts in meshless methods. Int. J. Numer. *Methods Engrg.* 47, 537–556.

14. Atluri, S.N.; Zhu, T. (2000b): The meshless local Petrov–Galerkin (MLPG) approach for solving problems in elasto-statics. *Comput. Mech.* 25, 169–179.

15. Liu, G.R.; Gu, Y.T. (2005): *An introduction to meshfree methods and their programming*. Springer, 479 pages.

16. Atluri, S.N. (2005): *Methods of computer modeling in engineering & the sciences*. Vol. I, 600 pages.

17. Oñate, E.; Cervera, M.; Zienkiewicz, O.C. (1994): A finite volume format for structural mechanics. *Int. J. Numer. Methods Engrg.* 37, 181–201.

18. Demirdzic I.; Muzaferija S. (1994): Finite volume method for stress analysis in complex domains. *Int. J. Numer. Methods Engrg.* 37, 3751–3766.

19. Jasak, H.; Weller, H.G. (2000): Application of the finite volume method and unstructured meshes to linear elasticity. *Int. J. Numer. Methods Engrg.* 48, 267–287.

20. Atluri, S.N.; Zhu, T. (1998): A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics* 22, pp. 117–127.

21. Timoshenko, S.P.; Goodier, J. N. (1976): *Theory of elasticity*. 3^{rd} edition, McGraw Hill.