Vietnam Journal of Mechanics, NCST of Vietnam Vol. 21, 1999, No 4 (239 – 250)

ON THE ELEMENT FREE GALERKIN METHOD IN THE STATIC-ANALYSIS FOR THE 2-D ELASTIC-LINEAR PROBLEM

NGUYEN HOAI SON - NGUYEN THE QUANG
Institute of Engineering Education, Ho Chi Minh City

ABSTRACT. The Element Free Galerkin (EFG) method is a meshless method for solving partial differential equations in which the trial and test functions employed in the discretization process result from moving least square interpolations (weak form of the variational principle). In this paper, the EFG method for solving problems in elastostatics (1-D, 2-D) is developed and numerically implemented. The present method is a meshless method, as it does not need a "finite element mesh" and it is only composed by the particles with theirs "compact support" (the influence domain) in the whole domain. Specially, the shape functions are not satisfying the Kronecker delta property, therefore, in this paper, we must enforce the essential boundary conditions by the Lagrangian multipliers method. Finally, several numerical examples are presented to illustrate the performance of the EFG method. The results are compared with the other method (EFM) and also with the analytic solutions. It shows that the EFG method gives the good effectiveness of the proposed error estimator in the global energy norm and the high rates of convergence with the size of the "compact support".

1. Introduction

The meshless methods are very attractive for the development of adaptive methods (h-adaptive, p-adaptive and $h-p$ adaptive) for solving boundary value problems. The meshless approach is based on the local symmetric weak form and moving least squares approximations. The main advantage of this method over the widely used finite element method and other so called meshless methods, EFG method [Belytschko et al 1994]. [Lu et al 1994], [Zhu and Atluri 1998], reproducing kernel particle method (RKPM) [Liu et all 1996], H-P cloud method [Duart et al 1998], is that it does not need a "finite element mesh" and requires only nodal data for the construction of the approximation. The approximating functions in EFG are moving least square approximates (MLS). They are not interpolate because the approximation does not pass through the data, this is often referred to as failure to satisfy the Kronecker delta property. Consequence, the essential boundary conditions can't be specified directly. However, we need enforcing the

essential boundary conditions by the Lagrangian multipliers method, modified the form of variational principle, the penalty method.... In this paper, we use the Lagrangian multipliers method to implement the essential boundary conditions in the EFG method [Belytschko et al 1994], [Lu et al 1994].

The distribution of the node is regular and the domain of influence of node is the straight line (1-D) and the rectangular (2-D). The local Petrov-Galerkin formulation for elasto-statics and its numerical implementation are developed. The numerical results is shown by the comparison table with the analytic solution and the diagrams (displacements, shear stress, rate of convergence...).

2. Moving Least-Square Formula in EFG Method

Let $\mathbf{u}(\mathbf{x})$ be a sufficiently smooth function ($\mathbf{u}(\mathbf{x}) \in \mathbf{C}^0(\overline{\Omega})$ at least) that is defined on a simply connected open set $\Omega \in \mathbf{R}^n$. Then for a fixed point $\overline{\mathbf{x}} \in \overline{\Omega}$, one should always be able to approximate $\mathbf{u}(\mathbf{x})$ by a polynomial series locally according to the Stone-Weierstrass theorem. Thus we can define a local function:

$$\mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}}) = \begin{cases} \mathbf{u}(\mathbf{x}) & \forall \mathbf{x} \in \mathbf{B}(\overline{\mathbf{x}}), \\ 0 & \forall \mathbf{x} \notin \mathbf{B}(\overline{\mathbf{x}}), \end{cases} \tag{2.1}$$

where the open sphere $\mathbf{B}(\overline{\mathbf{x}})$ is defined as

$$\mathbf{B}(\overline{\mathbf{x}}) = \left\{ \mathbf{x} \middle| |\mathbf{x} - \overline{\mathbf{x}}| \prec \mathbf{r}, \mathbf{x} \in \overline{\Omega} \right\}. \tag{2.2}$$

The local approximation $\mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}})$ of the function $\mathbf{u}(\mathbf{x})$ is expressed as the inner product of a vector of the polynomial basis $\mathbf{p}(\overline{\mathbf{x}})$ and a vector of the coefficients $\mathbf{a}(\mathbf{x})$:

$$\mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}}) = \mathbf{p}(\overline{\mathbf{x}})\mathbf{a}(\mathbf{x}) \qquad \mathbf{x} \in \overline{\Omega}, \tag{2.3}$$

where $\mathbf{p}(\mathbf{x})$, $\mathbf{a}(\mathbf{x}) \in \mathbf{R}^m$, m is the number of monomials in the polynomial basis. If we suppose that $\mathbf{u}_i$ is the value at a node i, then with the difference between the local approximation and the nodal value at a node i there exists a residual error: $\epsilon_i(\mathbf{x},\mathbf{x}_i) = [\mathbf{u}_i - \mathbf{u}_i^h(\mathbf{x},\mathbf{x}_i)]$. In the meshless method, each a node i is associated a compact support with the dilation a and weight function $\mathbf{W}_i(\mathbf{x},\mathbf{a})$. By minimizing the difference between the local approximation and the nodal value at a node i associated with the weight function $\mathbf{W}_i(\mathbf{x},\mathbf{a})$ we have

$$\mathbf{J}(\mathbf{a}(\mathbf{x})) = \sum_{i=1}^{n} \mathbf{W}_i(\mathbf{x},\mathbf{a})[\mathbf{u}_i - \mathbf{u}_i^h(\mathbf{x},\mathbf{x}_i)]^2 = \sum_{i=1}^{n} \mathbf{W}_i(\mathbf{x},\mathbf{a})[\mathbf{u}_i - \mathbf{p}(\mathbf{x}_i)\mathbf{a}(\mathbf{x})]^2, \ \mathbf{x}_i \in \mathbf{B}_i \tag{2.4}$$

where n is the number of nodes in compact support $\mathbf{B}$, $\mathbf{W}_i(\mathbf{x},\mathbf{a}) = \mathbf{W}_i(\mathbf{x}-\mathbf{x}_i,\mathbf{a})$ is a weight function with compact support B_i of measure $\mathbf{a}$, compact support corresponds to the domain influence of the weight function.

The stationary of eq. (2.4) with respect to $\mathbf{a}(\mathbf{x})$ leads to:

$$\frac{\partial \mathbf{J}(\mathbf{a}(\mathbf{x}))}{\partial \mathbf{a}(\mathbf{x})} = \frac{\partial}{\partial \mathbf{a}(\mathbf{x})}\Big(\sum_{i=1}^{n} \mathbf{W}_i(\mathbf{x},\mathbf{a})[\mathbf{u}_i - \mathbf{p}(\mathbf{x}_i)\mathbf{a}(\mathbf{x})]^2\Big) = 0 \tag{2.5}$$

or

$$\sum_{i=1}^{n} -\mathbf{p}^T(\mathbf{x}_i)\mathbf{W}_i(\mathbf{x},\mathbf{a})[\mathbf{u}_i - \mathbf{p}(\mathbf{x}_i)\mathbf{a}(\mathbf{x})] = 0,$$

so that

$$\sum_{i=1}^{n} \mathbf{p}^T(\mathbf{x}_i)\mathbf{W}_i(\mathbf{x},\mathbf{a})\mathbf{p}(\mathbf{x}_i)\mathbf{a}(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{p}^T(\mathbf{x}_i)\mathbf{W}_i(\mathbf{x},\mathbf{a})\mathbf{u}_i. \tag{2.6}$$

Put

$$\begin{aligned}
\mathbf{A}(\mathbf{x},\mathbf{a}) &= \sum_{i=1}^{n} \mathbf{p}^T(\mathbf{x}_i)\mathbf{W}_i(\mathbf{x},\mathbf{a})\mathbf{p}(\mathbf{x}_i),\\
\mathbf{C}(\mathbf{x},\mathbf{a}) &= \left[\mathbf{W}_1(\mathbf{x},\mathbf{a})\mathbf{p}(\mathbf{x}_1)\ \mathbf{W}_2(\mathbf{x},\mathbf{a})\mathbf{p}(\mathbf{x}_2)\cdots \mathbf{W}_n(\mathbf{x},\mathbf{a})\mathbf{p}(\mathbf{x}_n)\right],\\
\mathbf{u}^T &= [\mathbf{u}_1\ \mathbf{u}_2\ \cdots\ \mathbf{u}_n]
\end{aligned} \tag{2.7}$$

and note that: $\mathbf{W}_i(\mathbf{x},\mathbf{a}) = \mathbf{W}(\mathbf{x}-\mathbf{x}_i,\mathbf{a})$, the equation (2.6) becomes as following

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x},\mathbf{a})\mathbf{C}(\mathbf{x},\mathbf{a})\mathbf{u}. \tag{2.8}$$

The local approximation $\mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}}) = \mathbf{p}(\overline{\mathbf{x}})\mathbf{a}(\mathbf{x})$ is rewritten as:

$$\mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}}) = \mathbf{p}(\overline{\mathbf{x}})\mathbf{a}(\mathbf{x}) = \mathbf{p}(\overline{\mathbf{x}})\mathbf{A}^{-1}(\mathbf{x},\mathbf{a})\mathbf{C}(\mathbf{x},\mathbf{a})\mathbf{u}. \tag{2.9}$$

The fixed point $\overline{\mathbf{x}}$ is arbitrary, it can be any point $\mathbf{x} \in \overline{\Omega}$, therefore, one can let "move" over the whole domain, $\overline{\mathbf{x}} \to \mathbf{x}$, which leads to a global approximation of $\mathbf{u}(\mathbf{x})$, i.e.

$$\mathbf{u}_g^h(\mathbf{x}) = \lim_{\overline{\mathbf{x}}\to\mathbf{x}} \mathbf{u}_i^h(\mathbf{x},\overline{\mathbf{x}}) \qquad \forall \mathbf{x} \in \overline{\Omega} \tag{2.10}$$

$$\mathbf{u}_g^h(\mathbf{x}) = \mathbf{p}(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{C}(\mathbf{x})\mathbf{u} = \sum_{i=1}^{n} \Psi_i(\mathbf{x})\mathbf{u}_i \tag{2.11}$$

3. Application: Timoshenko Beam Problem 2-D

Consider a beam of length L subjected to a parabolic traction at the free end as Fig. 1. The beam has characteristic height D and is considered to be of unit depth and is assumed to be in a state of plane stress.

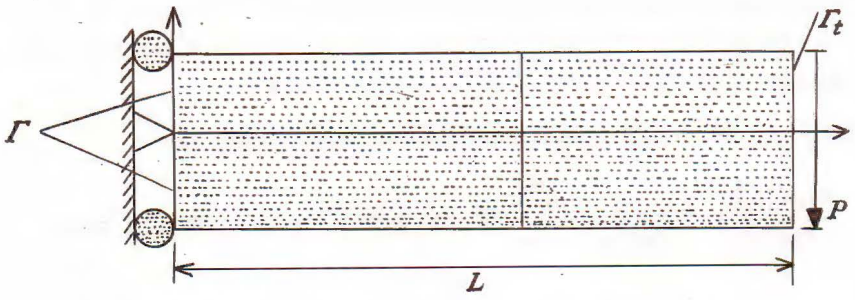


Fig. 1. A cantilever beam 2-D for comparison with the analytic solution from Timoshenko and Goodier

The exact solution to this problem is given by Timoshenko and Goodier [5]:

$$\begin{aligned} u_x &= \frac{-Py}{6EI}\left[(6L-3x)x+(2+\nu)\left(y^2-\frac{D^2}{4}\right)\right], \\ u_y &= \frac{P}{6EI}\left[3\nu y^2(L-x)+(4+5\nu)\frac{D^2x}{4}+(3L-x)x^2\right], \end{aligned} \tag{3.1}$$

where $I = \dfrac{D^2}{12}$ - moment of inertia of the beam.

The stresses corresponding to the displacements (3.1) are

$$\begin{aligned} \sigma_{xx}(x,y) &= \frac{P(L-x)y}{I}, \\ \sigma_{yy}(x,y) &= 0, \\ \sigma_{xy}(x,y) &= \frac{P}{2I}\left(\frac{D^2}{4}-y^2\right). \end{aligned} \tag{3.2}$$

The problem was solved for the plane stress case the regular mesh of nodes and the foundation mesh used to integrate the Galerkin weak form. In each integration cell, 4×4 Gauss quadrature was used to evaluate the stiffness matrix. The solution were obtained using a linear basis function with the cubic spline weight function.

3.1. Discretization of the equation

The equilibrium equation on the domain Ω with boundary Γ

$$\nabla\cdot\sigma+\mathbf{b}=0 \quad \text{in } \Omega \tag{3.3}$$

σ: stress Cauchy tensor with the displacement $\mathbf{u}$ and the body force vector $\mathbf{b}$, the boundary conditions

$$\sigma\cdot\mathbf{n}=\bar{\mathbf{t}} \quad \text{on} \quad \Gamma_t \tag{3.4}$$

$$\mathbf{u}=\bar{\mathbf{u}} \quad \text{on} \quad \Gamma_u \tag{3.5}$$

$\mathbf{n}$ is the unit outward normal on the boundary surface, $\Gamma = \Gamma_u \cup \Gamma_t$, with $\Gamma = \Gamma_u \cap \Gamma_t = \emptyset$, $\bar{\mathbf{u}}$, $\bar{\mathbf{t}}$ the prescribed values on the bound respectively. For the linear elastostatic problem $\varepsilon = \nabla_s \mathbf{u}$ and $\sigma = \mathbf{D}\varepsilon$.

The variational form (weak form) of the equilibrium equation is formulated as follows. Consider the trial function $\mathbf{u}(x) \in H^1$ and the Lagrange multipliers $\lambda \in H^0$ for all test functions $\delta\mathbf{v}(x) \in H^1$, $\delta\lambda \in H^0$, they satisfy the following equations

$$\int_\Omega \delta(\nabla_s \mathbf{v}^T)\sigma d\Omega - \int_\Omega \delta\mathbf{v}^T \cdot \mathbf{b} d\Omega - \int_{\Gamma_t} \delta\mathbf{v}^T \cdot \bar{\mathbf{t}} d\Gamma - \int_{\Gamma_u} \sigma^T \cdot (\mathbf{u} - \bar{\mathbf{u}}) d\Gamma - \int_{\Gamma_u} \delta\mathbf{v}^T \cdot \lambda d\Gamma = 0 \tag{3.6}$$

$\forall \delta\mathbf{v} \in H^1$, $\delta\lambda \in H^0$,

where $\nabla_s \mathbf{v}^T$ is the symmetric part of $\nabla \mathbf{v}^T$, H^1, H^0 are Hilbert spaces of degree one and zero, respectively. The equation (3.6) satisfy (3.1), (3.2), (3.3), but the trial functions do not satisfy the essential boundary condition. The Lagrange multipliers are written:

$$\begin{aligned} \lambda(x) &= N_i(s)\lambda_i, \quad x \in \Gamma_u \\ \delta\lambda(x) &= N_i(s)\delta\lambda_i, \quad x \in \Gamma_u \end{aligned} \tag{3.7}$$

$N_i(s)$ is Lagrange interpolation and s is the arclength of the bound. Finally, we have the system of equations under matrix form as following:

$$\begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{q} \end{Bmatrix} \tag{3.8}$$

with

$$\begin{aligned} K_{ij} &= \int_\Omega B_i^T D B_j d\Omega, \\ G_{ik} &= -\int_{\Gamma_u} \Phi_i N_k d\Gamma, \\ f_i &= \int_{\Gamma_t} \Phi_i \bar{t}_i d\Gamma + \int_\Omega \Phi_i b_i d\Omega, \\ q_k &= -\int_{\Gamma_u} N_k \bar{u}_k d\Gamma, \end{aligned} \tag{3.9}$$

where

$$B_i = \begin{bmatrix} \Phi_{i,x} & 0 \\ 0 & \Phi_{i,y} \\ \Phi_{i,y} & \Phi_{i,x} \end{bmatrix}$$

$$N_k = \begin{bmatrix} N_k & 0 \\ 0 & N_k \end{bmatrix}; \quad N_k = [B_k]^T[D]^T[n]^T[S]$$

$$\mathbf{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}; \quad [S] = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}; \quad [n] = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix}$$

$$S_x = \begin{cases} 1 & \text{if prescribed } u_x \text{ on } \Gamma_u \\ 0 & \text{if non-prescribed } u_x \text{ on } \Gamma_u \end{cases}$$

$$S_y = \begin{cases} 1 & \text{if prescribed } u_y \text{ on } \Gamma_u \\ 0 & \text{if non-prescribed } u_y \text{ on } \Gamma_u. \end{cases} \tag{3.10}$$

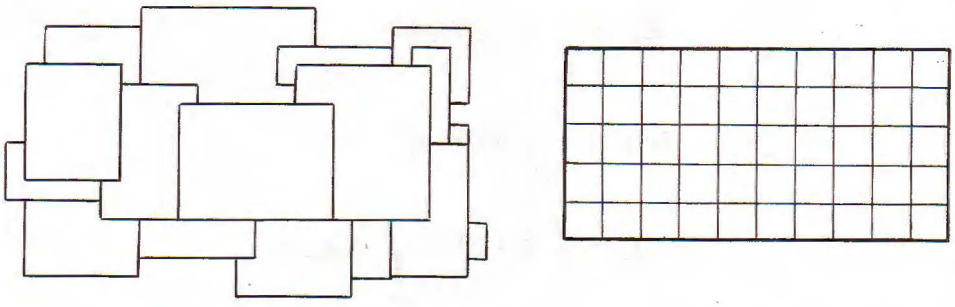
The rate of convergence in energy is calculated as

$$\|E\|_{L2} = \left\{\frac{1}{2}\int_{\Omega} \left(\varepsilon^{app} - \varepsilon^{an\ell}\right)^T \mathbf{D}\left(\varepsilon^{app} - \varepsilon^{an\ell}\right) d\Omega\right\}^{1/2} \tag{3.11}$$

or

$$\|E\|_{L2} = \left\{\frac{1}{2}\int_{\Omega} \left(\sigma^{app} - \sigma^{an\ell}\right)^T \mathbf{D}^{-1}\left(\sigma^{app} - \sigma^{an\ell}\right) d\Omega\right\}^{1/2} \tag{3.11'}$$

ε is defined as the symmetric gradient of the displacement $\mathbf{u}$. The value h was chosen to be the horizontal distance between the nodes in the model, and in each case a $d_{\max}$ of 2.0 with cubic spline weight function was used. In addition, 4×4 Gauss quadrature was used to integrate the Galerkin weak form, 4 points Gauss quadrature was used to integrate the $\mathbf{q}$ vector and $\mathbf{G}$ matrix along the essential boundary.



a) — support node i b) cells 10×5

Fig. 2. Cells and distribution of the nodes

3.2. Implementation

Get $\mathbf{x}^T = [x \quad y]$.

The linear basis function 2-D is chosen:

$$\mathbf{p}^T = [1 \quad x \quad y] = \mathbf{p}^T(\mathbf{x}) \tag{3.12}$$

$$\mathbf{A}(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{W}(\mathbf{x}-\mathbf{x}_i)\mathbf{p}(\mathbf{x}_i)\mathbf{p}^T(\mathbf{x}_i) = \mathbf{W}(\mathbf{x}-\mathbf{x}_1)\begin{bmatrix} 1 & x_1 & y_1 \\ x_1 & x_1^2 & x_1y_1 \\ y_1 & x_1y_1 & y_1^2 \end{bmatrix} + \tag{3.13}$$

$$+ \mathbf{W}(\mathbf{x}-\mathbf{x}_2)\begin{bmatrix} 1 & x_2 & y_2 \\ x_2 & x_2^2 & x_2y_2 \\ y_2 & x_2y_2 & y_2^2 \end{bmatrix} + \cdots + \mathbf{W}(\mathbf{x}-\mathbf{x}_n)\begin{bmatrix} 1 & x_n & y_n \\ x_n & x_n^2 & x_ny_n \\ y_n & x_ny_n & y_n^2 \end{bmatrix}$$

The shape functions:

$$\Phi_i(\mathbf{x}) = \sum_{j=1}^{m} \mathbf{p}_j(\mathbf{x})(\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}))_{ji} = \mathbf{p}^T\mathbf{A}^{-1}B_i. \tag{3.14}$$

The derivative shape function:

$$\Phi_{i,x} = \gamma_{,x}^T(\mathbf{x})B_i(\mathbf{x}) + \gamma(\mathbf{x})B_{i,x}(\mathbf{x}) \tag{3.15}$$

with

$$\gamma(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{p}(\mathbf{x})$$

The weight function

$$\mathbf{W}(\mathbf{x}-\mathbf{x}_i) = \mathbf{W}(\mathbf{r}_x)\cdot\mathbf{W}(\mathbf{r}_y) = \mathbf{W}_x\mathbf{W}_y \tag{3.16}$$

with

$$\mathbf{W}_x = \begin{cases} \frac{2}{3} - 4r_x^2 + 4r_x^3 & r_x \le \frac{1}{2} \\ \frac{4}{3} - 4r_x + 4r_x^2 - \frac{4}{3}r_x^3 & \frac{1}{2} < r_x \le 1 \\ 0 & r_x > 1 \end{cases}$$

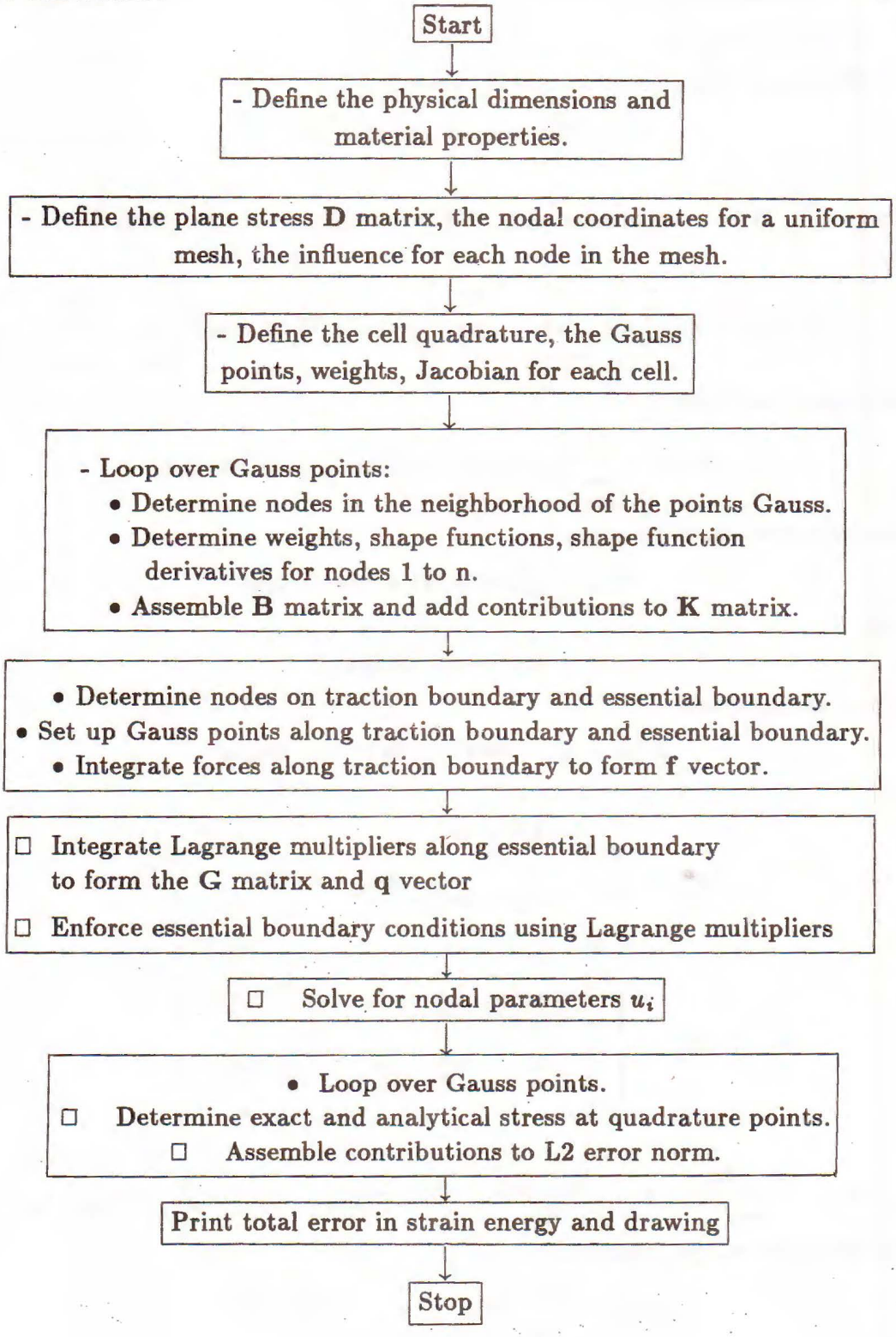
$$\mathbf{W}_y = \begin{cases} \frac{2}{3} - 4r_y^2 + 4r_y^3 & r_y \le \frac{1}{2} \\ \frac{4}{3} - 4r_y + 4r_y^2 - \frac{4}{3}r_y^3 & \frac{1}{2} < r_y \le 1 \\ 0 & r_y > 1 \end{cases}$$

$$r_x = \frac{\|x-x_i\|}{d_{mx}}, \quad r_y = \frac{\|y-y_i\|}{d_{my}}, \quad d_{mx} = d_{\max}\cdot c_{xi}, \quad d_{my} = d_{\max}\cdot c_{yi}.$$

The derivative weight function:

$$w_{,x} = \frac{dW_x}{dx}\cdot W_y \qquad w_{,y} = \frac{dW_y}{dy}\cdot W_x$$

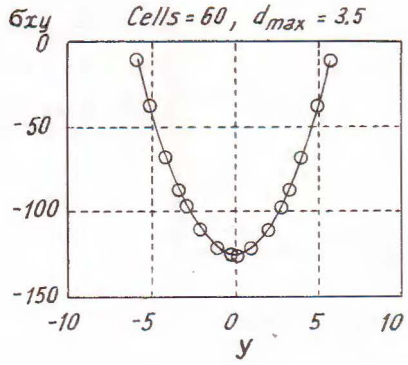
3.3. Flowchart

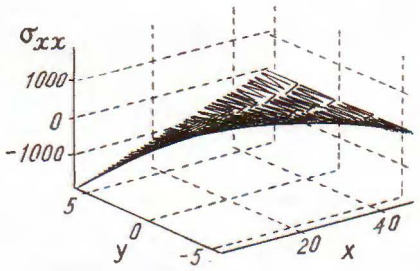


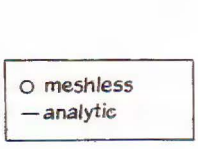
3.4. Results

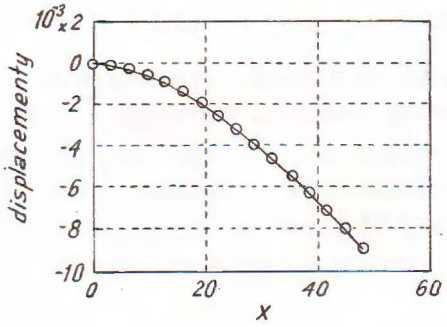
Table. Comparision of vertical displacement at end of beam the solution exact: $u_y^{\text{exact}} = -0.0089$, uniform space, $P = 1000$

Number of cells	Size of support $d_{\max}$	Vertical displacement u_y	Relative of errors (%) $\frac{(u_y^{(\text{exact})} - u_y)}{u_y^{(\text{exact})}}$	Energy (10^{-4})	Errors norm of energy
60	1.5	-0.00884	0.6	0.7814	0.0045
60	2.0	-0.00888	0.2	0.7885	0.0034
60	3.5	-0.00891	0.01	0.7921	0.0014
48	1.5	-0.00876	1.57	0.7673	0.0052
48	2.0	-0.008876	0.34	0.7878	0.0038
48	3.5	-0.008898	0.025	0.7917	0.0008
40	1.5	-0.00870	2.2	0.7569	0.0058
40	2.0	-0.00887	0.3	0.7868	0.0043
40	3.5	-0.008897	0.03	0.7915	0.0016
28	1.5	-0.0085	4.5	0.7225	0.0073
28	2.0	-0.00883	0.79	0.7796	0.0052
28	3.5	-0.008897	0.033	0.7914	0.0017



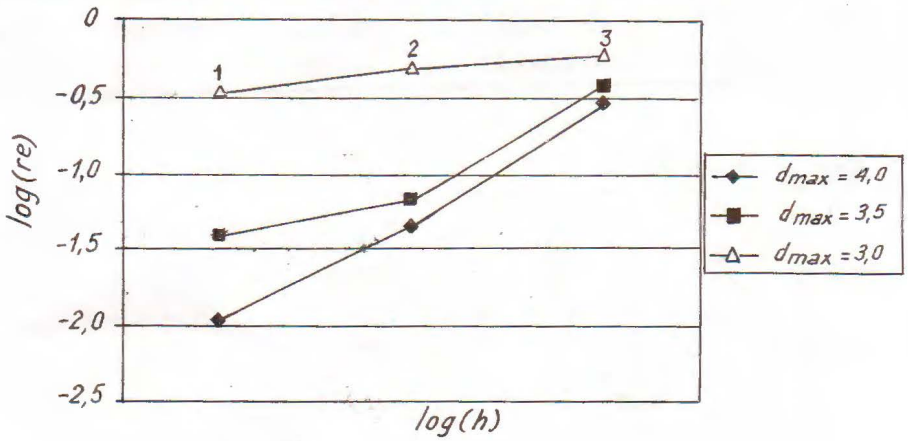






- Energy norm $\|\varepsilon\| = \left(\frac{1}{2}\int \sigma^T \mathbf{D}^{-1}\sigma d\Omega\right)^{1/2}$
- The relative error for $\|\varepsilon\|$ is defined as: $r_e = \frac{\|\varepsilon^{num} - \varepsilon^{exact}\|}{\|\varepsilon^{exact}\|}$

size of cell (h)	$\log_{10}(h)$	r_e	$\log_{10}(r_e)$
	$d_{\max} = 4$		
3.2	1.1631	0.1402	-1.9647
4.8	1.5686	0.2428	-1.4155
9.6	2.2617	0.6322	-0.4585
	$d_{\max} = 3.5$		
3.2	1.1631	0.2585	-1.3529
4.8	1.5686	0.3081	-1.1773
9.6	2.2617	0.7355	-0.3072
	$d_{\max} = 3$		
3.2	1.1631	0.5861	-0.5343
4.8	1.5686	0.6556	-0.4222
9.6	2.2617	0.7961	-0.2280



Relative errors and convergence rate

4. Conclusions

The Moving Least-Square approximation do not satisfy Kronecker delta criterion. For enforcing essential boundary conditions, generally, one has proposed three methods: Langrangian multipliers - Modified Variational Principle - Transformation of the Incremental Equation from Generalized Coordinates to nodal coordinate - Penalty Method. In this paper, we use the first method. Consequently, our given numerical solution is very good.

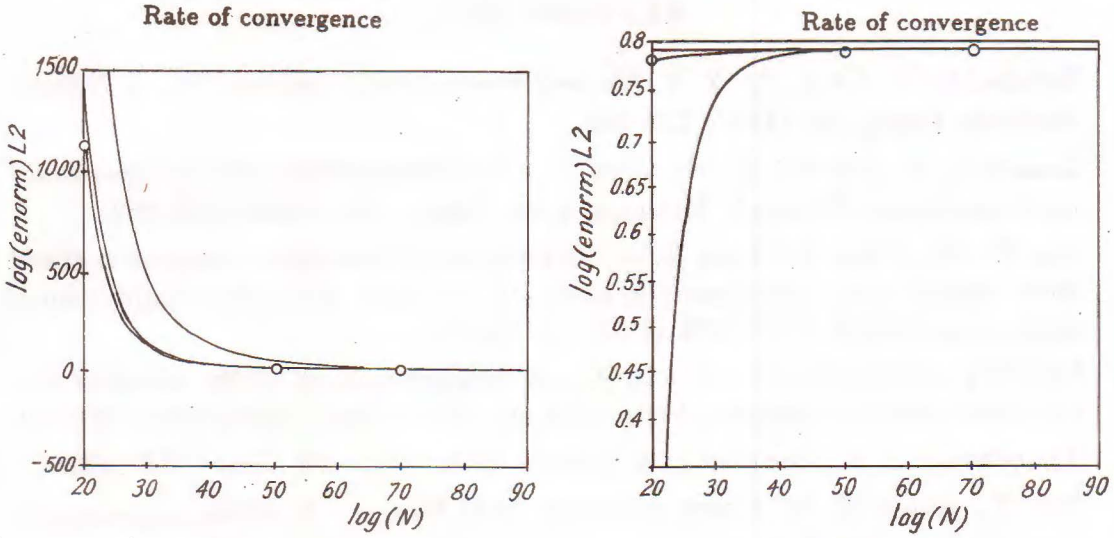


Fig. (a) *Fig. (b)*

Model Static admissible (a) and Kinematic admissible (b)

In the Meshless method, the construction of the window function (RKPM), the Weight function (EFG), the Cloud function (HP-cloud) play a very important role and define the size of compact support. The cubic spline and the adaptive parameter are chosen. They are able to allow to choose shape functions near to the geometry of the boundary thus reducing interpolation error. They also influence on the stability, the error and the rate of convergence of the method.

The present method is considerably more accurate for computing the displacements and stresses than the finite element method. For example, the relative error of vertical displacement for meshless: 0.010% and FEM: 0.016%.

No smoothing mesh technique is required to compute the stresses and strains, as the original result is smooth enough.

No element connectivity is need and only uniform space distributed nodal points are constructed.

No using HP adaptivity technique is required to augmente order precise. But it also give the results very precise for the displacements, the shear stresses and the strains.

Convergence studies in the numerical examples show that the present method possesses an excellent rate of convergence for displacements as well as for the strain energy at $d_{\max} = 4.0$ and $d_{\max} = 3.5$.

The large supports gives more accurate results than smaller supports. *A* very small of the size compact support may result in a relatively large numerical error in using Gauss numerical quadrature to calculate the entries in the system matrix. If we choose very large of the size compact support which is not to maintain the local character of the MLS approximation.

REFERENCES

1. Belytschko T., Gu L., Lu Y. Y. Element-free Galerkin method, Int. J. Numer. Methods Engrg. **37** (1994) 229-256.
2. Duarte C. A., Oden J. T. H_p Clouds - a meshless method to solve boundary value problems. Comput. Methods Appl. Mech. **139** (1996), 237-262.
3. Liu W. K., Chen Y., Uras R. A. Enrichment of the finite element method with reproducing kernel particle methods. Current topics in computational mechanics, ASME PVP, **305** (1995), 4015-4037.
4. Lu Y. Y., Belytschko T., Gu L. A new implementation of the element free Galerkin method. Comput. Methods Appl. Mech. Eng. **113** (1994), 397-414.
5. Timoshenko S. P., Goodier J. N. Theory of elasticity Mc Graw-Hill 1970.
6. Zhu T., Atluri S. N. A new meshless local Petrov - Galerkin approach in computational mechanics. Comput. Mech. **22** (1998), 117-127.

Received July 3, 1999

VỀ PHƯƠNG PHÁP PHẦN TỬ TỰ DO GALERKIN GIẢI BÀI TOÁN TĨNH ĐÀN HỒI TUYẾN TÍNH HAI CHIỀU

Phương pháp phần tử tự do Galerkin (EFG) là phương pháp không lưới giải các phương trình đạo hàm riêng, mà các hàm thử và hàm kiểm tra dùng trong quá trình rời rạc được xác định từ các nội suy theo kiểu di truyền bình phương tối thiểu (MLS - dạng yếu của nguyên lý biến phân). Trong bài báo này sử dụng phương pháp EFG để giải bài toán đàn hồi tĩnh. Phương pháp này không cần lưới như "lưới phần tử hữu hạn", mà chỉ là tập các chất điểm với các miền ảnh hưởng của chúng bao phủ toàn bộ miền bài toán. Đặc biệt các hàm dạng của nó không thỏa mãn tính chất delta Kronecker. Chính vì vậy, trong bài báo này chúng tôi hiệu lực hóa các điều kiện trên chính bằng phương pháp các nhân tử Lagrange. Sau cùng, nhiều ví dụ bằng số được giới thiệu để minh họa việc thực hiện phương pháp phần tử hữu hạn và với lời giải giải tích.

Kết quả cho ta thấy phương pháp không lưới kiểu EFG đạt kết quả tốt, tốc độ hội tụ cao.