 CMES, vol.7, no.3, pp.241-268, 2005

Simulation of a 4^{th} Order ODE: Illustration of Various Primal & Mixed MLPG Methods

S. N. Atluri[1] **and Shengping Shen**[1]

Abstract: Various MLPG methods, with the MLS approximation for the trial function, in the solution of a 4^{th} order ordinary differential equation are illustrated. Both the primal MLPG methods and the mixed MLPG methods are used. All the possible local weak forms for a 4^{th} order ordinary differential equation are presented. In the first kind of mixed MLPG methods, both the displacement and its second derivative are interpolated independently through the MLS interpolation scheme. In the second kind of mixed MLPG methods, the displacement, its first derivative, second derivative and third derivative are interpolated independently through the MLS interpolation scheme. The nodal values of the independently interpolated derivatives are expressed in terms of nodal values of the independently interpolated displacements, by simply enforcing the strain-displacement relationships directly by collocation at the nodal points. The mixed MLPG methods avoid the need for a direct evaluation of high order derivatives of the primary variables in the local weak forms, and thus reduce the continuity-requirement on the trial function. Numerical results are presented to illustrate the effectiveness of the primal, as well as two kind of mixed MLPG methods. It is concluded that the mixed MLPG methods are very cost-effective.

keyword: MLPG, Mixed Methods, Moving Least Squares, Local Weak Forms.

1 Introduction

Most problems in mechanics are characterized by partial differential equations, in space and time. The development of approximate methods for the solution of these PDEs has attracted the attention of engineers, physicists and mathematicians for several decades. In the beginning, the finite difference methods were extensively developed to solve these equations. As being derived from the variational principles, or their equivalent weak-forms, the finite element methods have emerged as the most successful methods to solve these partial differential equations, over the past three decades. Recently, the so-called meshless methods of discretization have become very attractive, as they are efficient for solving PDEs by avoiding the tedium of mesh-generation, especially for those problems having complicated geometries, as well as those involving large strains. As a systematic framework for developing various truly-meshless methods, the Meshless Local Petrov-Galerkin (MLPG) approach has been proposed as a fundamentally new concept [Atluri and Zhu (1998); Atluri and Shen (2002a, b); Atluri (2004)]. The generality of the MLPG approach, based on the symmetric or unsymmetric weak-forms of the PDEs, and using a variety of interpolation methods (trial functions), test functions, and integration schemes without background cells, has been widely investigated [Atluri and Shen (2002a, b); Atluri(2004)].

Remarkable successes of the MLPG method have been reported in solving the convection-diffusion problems [Lin and Atluri (2000)]; fracture mechanics [Batra and Ching (2002)]; Navier-Stokes flows [Lin and Atluri (2001)]; and plate bending problems [Long and Atluri (2002)]. Recently, the MLPG method has made some strides, and it is applied successfully in studying strain gradient materials [Tang, Shen and Atluri, (2003)), three dimensional elasticity problems [Li, Shen, Han and Atluri (2003), Han and Atluri (2004a)], elstodynamics [Han and Atluri (2004b)], and multiscale structure and nanomechanics [Shen and Atluri (2004, 2005)]. These research successes demonstrate that the MLPG method is one of the most promising alternative methods for computational mechanics. The interrelation of the various meshless approaches, and the recent developments and

[1] Center for Aerospace Research & Education
University of California at Irvine
5251 California Avenue, #140
Irvine, CA 92617, USA

applications of the MLPG methods can be founded in Atluri (2004).

Very recently, Atluri, Han and Rajendran (2004) developed a mixed MLPG method for the 2^{nd} order partial differential equation system of elasticity, in which both strains and displacements were independently interpolated through meshless interpolation schemes. The nodal values of strains were expressed in terms of the independently interpolated nodal values of displacements, by simply enforcing the strain-displacement relationships directly by collocation at the nodal points. The mixed MLPG method avoids the direct differentiation of the trial function (and the evaluation of these directly derived derivatives at each Gauss point in the integration of the weak form), and reduces the continuity-requirement on the trial function by one-order. A smaller interpolation domain can be used in the meshless approximation with a lower-order polynomial basis. The mixed MLPG method improves very much the efficiency of the primal MLPG method. In this paper, we will employ this idea to solve the 4^{th} order ordinary differential equation.

The 4^{th} order ordinary differential equation is more complicated. For the 4^{th} order ordinary differential equation, in the conventional displacement-based approaches in FEM, the interpolation of displacement requires C^1 continuity (in order to ensure convergence of the finite element procedure for $4th$ order theories), which inevitably involves very complicated shape functions. These shape functions involve large numbers of degrees of freedom in every element, including nodal displacements, nodal rotations (i.e. first order gradients of displacement), and even higher order derivatives. C^1 continuous methods are mostly feasible only for one-dimensional problems. The standard approach for solving Bernoulli-Euler beam problems is by employing C^1 continuous Hermite cubic shape functions, interpolating both displacements and rotations (i.e., slopes, the 1^{st} derivative). For two dimensional problems, such as involving plate and shell analysis, C^1 continuous methods are very complicated, and formulations for three-dimensional problems become more or less intractable. The high computational cost and large number of degrees of freedom soon place such formulations beyond the realm of practicality. Atluri, Cho and Kim (1999) presented an analysis of thin beam problems using the MLPG method with a generalized moving least squares (GMLS) approximation. Then, Cho & Atluri (2001) extended it to the shear flexible beams based on a locking-free formulation. Raju and Phillips (2003) applied MLPG with the GMLS approximation to a continuous beam problem to evaluate their effectiveness, and discussed the effects of various parameters on the numerical results clearly and systematically. Long & Atluri (2002) analyzed the bending problem of a thin plate by means of MLPG with the MLS approximation.

In this paper, we will illustrate various MLPG methods with the MLS approximation for the simulation of the 4^{th} order ordinary differential equation (of a beam on an elastic foundation), including both the primal MLPG methods and the mixed MLPG methods. All the possible local weak forms for a 4^{th} order ordinary differential equation are given in this paper.

The outline of the paper is as follows. In Section 2, the meshless interpolation-the moving least square method is described briefly for the sake of completeness. The local weak forms and the corresponding primal MLPG methods are discussed in Section 3, and their numerical results are also presented. In Section 4, the first kind of mixed MLPG methods and their numerical results are given. In the first kind of mixed MLPG methods, both the displacement and its second derivative are interpolated independently through meshless interpolation schemes. The second kind of mixed MLPG methods and their numerical results are presented in Section 5. In the second kind of mixed MLPG methods, the displacement, its first derivative, second derivative and third derivative are all interpolated independently through meshless interpolation schemes. The conclusions and discussions are given in Section 6.

2 Meshless Interpolation: the Moving Least Square Method

In general, meshless methods use a local interpolation, or an approximation, to represent the trial function, using the values (or the fictitious values) of the unknown variable at some randomly located nodes in the local vicinity. A variety of local interpolation schemes that interpolate the data at randomly scattered points (without the need for a mesh) are currently available [Atluri and Shen (2002a, b), Atluri (2004)].

The moving least-square method is generally considered to be one of the best schemes to interpolate data with a reasonable accuracy. Basically the MLS interpolation

does not pass through the nodal data. Consider a domain in question with control points for boundaries (i.e. nodes on boundaries) and some scattered nodes inside, where every node has its undetermined nodal coefficient (fictitious nodal value) and an influence radius (radius for local weight function). Now for the distribution of trial function at any point x and its neighborhood Ω_x located in the problem domain Ω, $u^h(x)$ may be defined by

$$u^h(x) = \mathbf{p}^T(x)\,\mathbf{a}(x) \;\; \forall x \in \Omega_x \tag{1}$$

where $\mathbf{p}^T(x)$=[$p_1(x)$, $p_2(x)$, …,$p_m(x)$] is a complete monomial basis of order m, and $\mathbf{a}(x)$ is a vector containing coefficients $a_j(x)$, j=1, 2, …,m which are functions of the space co-ordinates x. The commonly used bases in 1-D are the linear basis (m=2), due to their simplicity. In the present 4th order problem, we will also employ the quadratic basis (m=3)

$$\mathbf{p}^T(x) = \begin{bmatrix} 1 & x & x^2 \end{bmatrix} \tag{2}$$

and the cubic basis (m=4)

$$\mathbf{p}^T(x) = \begin{bmatrix} 1 & x & x^2 & x^3 \end{bmatrix} \tag{3}$$

The coefficient vector $\mathbf{a}(x)$ is determined by minimizing a weighted discrete L_2 norm, which can be defined as

$$J(x) = \sum_{I=1}^{N} w_I(x)\left[\mathbf{p}^T(x_I)\,\mathbf{a}(x) - \hat{u}^I\right]^2 \tag{4}$$

where $w_I(x)$, is a weight function associated with the node I, with $w_I(x) > 0$ for all x in the support of $w_I(x)$, x_I denotes the value of x at node I, N is the number of nodes in Ω_x for which the weight functions $w_I(x) > 0$. Here it should be noted that $\hat{u}^I$, I=1, 2,…, N, in equation (4), are the fictitious nodal values (undetermined nodal coefficients), and not the exact nodal values of the unknown trial function $u^h(\mathbf{x})$, in general.

Solving for $\mathbf{a}(x)$ by minimizing J in equation (4), and substituting it into equation (1), give a relation which may be written in the form of an interpolation function similar to that used in the FEM, as

$$u^h(x) = \sum_{I=1}^{N} \phi^I(x)\,\hat{u}^I,\; u^h(x_I) \equiv u^I \neq \hat{u}^I,\; x \in \Omega_x \tag{5}$$

where

$$\phi^I(x) = \sum_{j=1}^{m} p_j(x)\left[\mathbf{A}^{-1}(x)\,\mathbf{B}(x)\right]_{jI} \tag{6}$$

with the matrix $\mathbf{A}(x)$ and $\mathbf{B}(x)$ being defined by:

$$\mathbf{A}(x) = \sum_{I=1}^{N} w_I(x)\,\mathbf{p}(x_I)\,\mathbf{p}^T(x_I) \tag{7}$$

$$\mathbf{B}(x) = \left[w_1(x)\,\mathbf{p}(x_1),\; w_2(x)\,\mathbf{p}(x_2),\; \cdots,\; w_N(x)\,\mathbf{p}(x_N)\right] \tag{8}$$

Equation (6) can be rewritten as

$$\mathbf{\Phi}(x) = \boldsymbol{\alpha}^T(x)\,\mathbf{B}(x) \tag{9}$$

with

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{p} \tag{10}$$

The partial derivatives of $\boldsymbol{\alpha}$ are obtained through the relations:

$$\mathbf{A}\boldsymbol{\alpha}_{,x} = \mathbf{p}_{,x} - \mathbf{A}_{,x}\boldsymbol{\alpha} \tag{11}$$

$$\mathbf{A}\boldsymbol{\alpha}_{,xx} = \mathbf{p}_{,xx} - \mathbf{A}_{,xx}\boldsymbol{\alpha} - 2\mathbf{A}_{,x}\boldsymbol{\alpha}_{,x} \tag{12}$$

$$\boldsymbol{\alpha}_{,xxx} = \mathbf{p}_{,xxx} - \mathbf{A}_{,xxx}\boldsymbol{\alpha} - 3\mathbf{A}_{,xx}\boldsymbol{\alpha}_{,x} - 3\mathbf{A}_{,x}\boldsymbol{\alpha}_{,xx} \tag{13}$$

Now, the partial derivatives of $\phi^I(\mathbf{x})$ are obtained as

$$\phi^I_{,x} = \boldsymbol{\alpha}^T_{,x}\mathbf{B}_I + \boldsymbol{\alpha}^T\mathbf{B}_{I,x} \tag{14}$$

$$\phi^I_{,xx} = \boldsymbol{\alpha}^T_{,xx}\mathbf{B}_I + 2\boldsymbol{\alpha}^T_{,x}\mathbf{B}_{I,x} + \boldsymbol{\alpha}^T\mathbf{B}_{I,xx} \tag{15}$$

$$\phi^I_{,xxx} = \boldsymbol{\alpha}^T_{,xxx}\mathbf{B}_I + 3\boldsymbol{\alpha}^T_{,xx}\mathbf{B}_{I,x} + 3\boldsymbol{\alpha}^T_{,x}\mathbf{B}_{I,xx} + \boldsymbol{\alpha}^T\mathbf{B}_{I,xxx} \tag{16}$$

in which $\mathbf{B}_I$ is the Ith column of matrix $\mathbf{B}$. Thus, the derivatives of $\phi^I(\mathbf{x})$ become rather complicated.

The nodal shape function is complete up to the order of the basis. The smoothness of the nodal shape function $\phi^I(x)$ is determined by that of the basis and of the weight function. The choice of the weight function is more or less arbitrary as long as the weight function is positive and continuous. The following weight function is considered in the present work:

$$w_I(x) = \begin{cases} 1 - \sum\limits_{k=1}^{p} a_k \left(\frac{d_I}{r_I}\right)^k & 0 \le d_I \le r_I = \rho_I h_I \\ 0 & d_I > r_I = \rho_I h_I \end{cases} \tag{17}$$

where $d_I = |x - x_I|$ is the distance from node x_I to point x, h_I in the nodal distance, ρ_I is the scaling parameter for the size of the sub-domain Ω_{Ir}^I, and p is the order of spline function. The coefficients a_k are obtained by taking the following boundary conditions:

$$\begin{cases} w_I(d_I/r_I = 0) = 1, & m_0 = 0 \\ \frac{\partial^{m_0} w_I(d_I/r_I=0)}{\partial x^{m_0}} = 0, & m_0 \geq 1 \end{cases} \tag{18}$$

and

$$\begin{cases} w_I(d_I/r_I = 1) = 0, & m_1 = 0 \\ \frac{\partial^{m_1} w_I(d_I/r_I=1)}{\partial x^{m_1}} = 0, & m_1 \geq 1 \end{cases} \tag{19}$$

where $p = m_0 + m_1 + 1$. The form of the weight functions may be changed by the geometry of the sub-domain Ω_{Ir}^I over which the weight function is non-zero. Since the weight function is a type of a polynomial, the "nodal shape function" given in equation (6) has the characteristics of a rational function. One can easily obtain a global C^l continuity up to a desired order l if the order of spline is changed. Therefore, the C^l continuity depends upon value of m_0 and m_1 in equations (18) and (19), i.e. $\phi(\mathbf{x}) \in C^{\min(m_0,m_1)}$ if m_0 is even; $\phi(\mathbf{x}) \in C^{\min(m_0+1,m_1)}$ if m_0 is odd. It is very important to preserve the smoothness of the derivatives of shape functions, because discontinuities and vertices in the derivatives of the shape functions make numerical integration difficult. In this paper, we choose a 4^{th} order spline function (C^2)

$$w_I(x) = \begin{cases} 1 - 6\left(\frac{d_I}{r_I}\right)^2 + 8\left(\frac{d_I}{r_I}\right)^3 - 3\left(\frac{d_I}{r_I}\right)^4, & 0 \leq d_I \leq r_I \\ 0, & d_I \geq r_I \end{cases} \tag{20}$$

as well as a 7^{th} order spline function (C^3),

$$w_I(x) = 1 - 35\left(\frac{d_I}{r_I}\right)^4 + 84\left(\frac{d_I}{r_I}\right)^5 - 70\left(\frac{d_I}{r_I}\right)^6 + 20\left(\frac{d_I}{r_I}\right)^7 \tag{21}$$

for $0 \leq d_I \leq r_I$, as weight functions.

Now, in order to depict its performance, we will employ the MLS to simulate the following function, which is the known exact solution of the 4^{th} order differential equation

$$\begin{aligned} &\frac{d^4u}{dx^4} + u - 1 = 0,\ 0 < x < 1 \\ &u = u'' = 0,\ x = 0, \text{ and } 1 \end{aligned} \tag{22}$$

i.e.,

$$\begin{aligned} u_{exact} = 1 &+ \left(\frac{-e^{-\frac{1}{\sqrt{2}}}\left(\cos\frac{1}{\sqrt{2}} + e^{-\frac{1}{\sqrt{2}}}\right)}{2e^{-\frac{1}{\sqrt{2}}}\cos\frac{1}{\sqrt{2}} + e^{-\frac{2}{\sqrt{2}}} + 1}\right) e^{\frac{x}{\sqrt{2}}}\cos\frac{x}{\sqrt{2}} \\ &+ \left(\frac{-e^{-\frac{1}{\sqrt{2}}}\sin\frac{1}{\sqrt{2}}}{2e^{-\frac{1}{\sqrt{2}}}\cos\frac{1}{\sqrt{2}} + e^{-\frac{2}{\sqrt{2}}} + 1}\right) e^{\frac{x}{\sqrt{2}}}\sin\frac{x}{\sqrt{2}} \\ &+ \left(\frac{-\left(e^{-\frac{1}{\sqrt{2}}}\cos\frac{1}{\sqrt{2}} + 1\right)}{2e^{-\frac{1}{\sqrt{2}}}\cos\frac{1}{\sqrt{2}} + e^{-\frac{2}{\sqrt{2}}} + 1}\right) e^{-\frac{x}{\sqrt{2}}}\cos\frac{x}{\sqrt{2}} \\ &+ \left(\frac{-e^{-\frac{1}{\sqrt{2}}}\sin\frac{1}{\sqrt{2}}}{2e^{-\frac{1}{\sqrt{2}}}\cos\frac{1}{\sqrt{2}} + e^{-\frac{2}{\sqrt{2}}} + 1}\right) e^{-\frac{x}{\sqrt{2}}}\sin\frac{x}{\sqrt{2}} \end{aligned} \tag{23}$$

In this simulation, the linear basis, and the 7^{th} order spline function (21) are employed; 20 nodes are distributed evenly from 0 to 1, the radius of the support domain of the trial function is taken to be $5.5h_I$. Thus, setting:

$$u_{exact}(x_J) = \sum_{I=1}^{20} \phi^I(x_J)\,\hat{u}^I;\ J = 1, \cdots, 20 \tag{24}$$

and using (6), we obtain the values of $\hat{u}^I$. Then we obtain the interpolation $u^h(x)$ for u_{exact}, as:

$$u^h(x) = \sum_{I=1}^{20} \phi^I(x)\,\hat{u}^I \tag{25}$$

Fig. 1 shows the distinction between the exact nodal values u^I of the trial function $u^h(x)$, and the fictitious nodal values $\hat{u}^I$. Figs. 2-5 compare the values of the function $u^h(x)$ and its 1^{st}-3^{rd} order derivatives calculated from the MLS interpolation with the corresponding exact values obtained from (23). Form these figures, we can see that the MLS can interpolate the function u_{exact} and its first

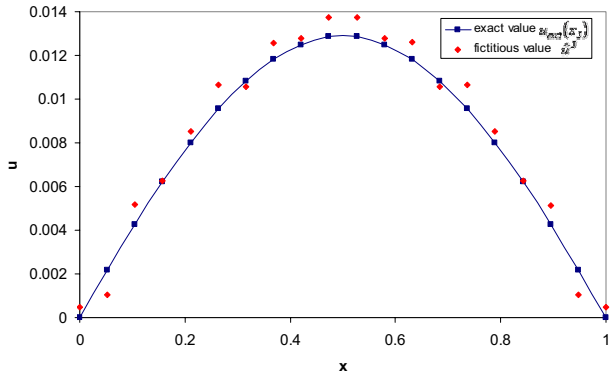

Figure 1 : The distinction between the exact nodal values u^I of the trial function $u^h(x)$, and the fictitious nodal values $\hat{u}^I$.

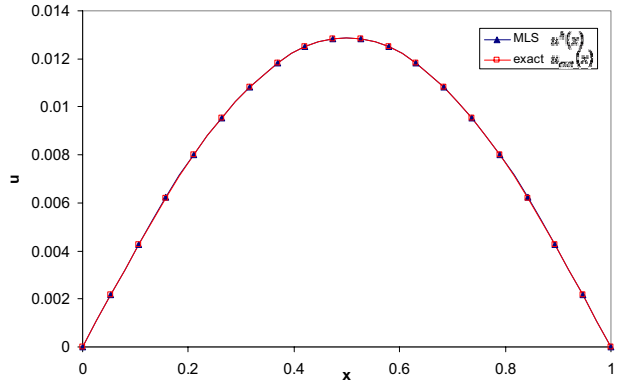

Figure 2 : The function u.

derivative very well, but it has poor accuracy to interpolate the higher order derivatives. Hence, in the weak forms, we should avoid the appearance of the higher order derivatives of $u^h(x)$.

3 Local Weak Forms and Primal MLPG Methods

Consider a 4^{th} order ODE

$$\frac{d^4u}{dx^4} + u - 1 = 0 \tag{26}$$

in domain Ω $(0 \le x \le 1)$. Equation (26) is actually the governing equation for a thin beam on an elastic foundation undergoing small deformations, in which u is the normalized transverse displacement. The boundary conditions at x=0 and x=1 can have several combinations.

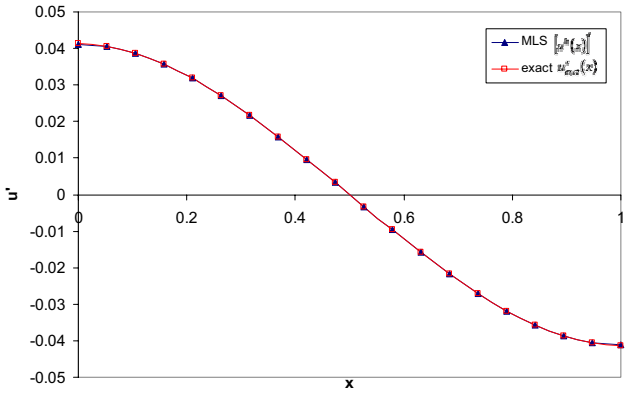

Figure 3 : The first derivative of the function u.

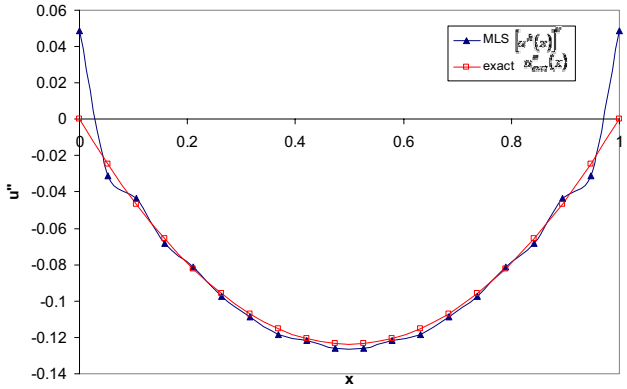

Figure 4 : The second derivative of the function u.

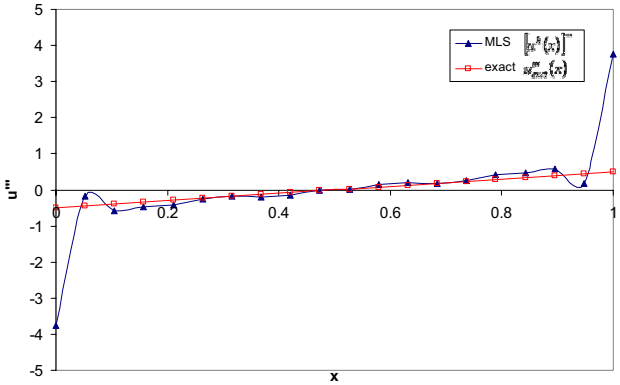

Figure 5 : The third derivative of the function u.

The essential boundary conditions are of the form

$$u = \overline{u} \text{ on } \Gamma_u \tag{27}$$

$$u' = \overline{u}' \text{ on } \Gamma'_u \tag{28}$$

and the natural boundary conditions are of the form

$$u''' = \overline{u}''' \text{ on } \Gamma'''_u \tag{29}$$

$$u'' = \overline{u}'' \text{ on } \Gamma''_u \tag{30}$$

where Γ_u, Γ'_u, Γ''_u, and Γ'''_u denote the boundary points where u (deflection), u' (slope), u''' (shear), and u'' (moment) are prescribed, respectively. Note that the prescriptions of $\overline{u}$ & $\overline{u}'''$, and $\overline{u}'$ & $\overline{u}''$ are mutually disjoint, i.e., when $u = \overline{u}$ is prescribed, u''' becomes the corresponding reaction, and when $u' = \overline{u}'$ is prescribed, u'' becomes the corresponding reaction.

Now, we will give all the local weak forms for equation (26), which are the basis of the MLPG methods. A generalized local weak form (LWF) of the 4^{th} order differential equation (26) over a local subdomain Ω_s, can be written as:

$$\int_{\Omega_s} \left(\frac{d^4 u}{dx^4} + u - 1 \right) v dx = 0 \tag{31}$$

The local weak form (31) includes the fourth derivative of the trial function. If the test function v is taken to be the Delta function, the collocation method will be derived from equation (31). In section 2, it is shown that it is too difficult to obtain the accurate higher order derivatives of the trial function; moreover, they are not very accurate. Hence, this weak form is not appropriate for the numerical implementation. In fact, we did implement the collocation in our numerical experiments; However, we could not obtain any meaningful results, as was expected (these numerical results are omitted in this paper).

To obtain an accurate and efficient meshless method, one should decrease the order of the derivatives of the trial function in the local weak forms. There are two ways to reach this goal. One is by means of integrating by parts, through which the differentiation can be transferred from the trial function, u, to the test function, v. Then, the higher order derivatives in the domain integration will disappear. However, the higher order derivatives of the trial function still appear in the local boundary integral (local boundary for 1D problem). This is the primal MLPG method. Another promising approach is the use of the "mixed" MLPG approach [Atluri, Han, Rajendran (2004)], wherein, independent meshless interpolations are also used for the derivatives, as well as the function per se. The mixed methods are described in detail in the next section. In this section, we limit ourselves to the primal MLPG methods.

3.1 Unsymmetric local weak form 1

By integrating (31) by part once, one obtain:

$$n_x \left[u''' v \right]_{\Gamma_s} - \int_{\Omega_s} u''' v' dx + \int_{\Omega_s} (u-1) v dx = 0 \tag{32}$$

where $n_x [\cdot]_{\Gamma_s}$ denotes the boundary term and n_x is the outward normal. For one-dimensional problems, the local boundary Γ_s has two points, and n_x has the values of ± 1. Equation (32) is an unsymmetric weak form, and the trial function should be C^2 continuious. There is a third derivative of function u in LWF (32). Imposing the boundary conditions (29), one obtains

$$\begin{aligned} &n_x \left[\overline{u}''' v \right]_{\Gamma_{su'''}} + n_x \left[u''' v \right]_{\Gamma_{su}} + n_x \left[u''' v \right]_{L_s} \\ &- \int_{\Omega_s} u''' v' dx + \int_{\Omega_s} (u-1) v dx = 0 \end{aligned} \tag{33}$$

where $\Gamma_{su'''}$ is the boundary where u''' is prescribed on the local boundary ($\Gamma_s \cap \Gamma'''_u$). In general, when a local boundary, Γ_s, intersects a global boundary, Γ, four boundary condition possibilities exist. These possibilities are $\Gamma_s \cap \Gamma_u$, $\Gamma_s \cap \Gamma'_u$, $\Gamma_s \cap \Gamma''_u$, and $\Gamma_s \cap \Gamma'''_u$, and are denoted as Γ_{su}, $\Gamma_{su'}$, $\Gamma_{su''}$, and $\Gamma_{su'''}$, respectively; and L_s is the other part of the local boundary which is inside the solution domain. Since the boundary Γ_u and Γ'''_u are mutually disjoint, and are related by $\Gamma = \Gamma_u \cup \Gamma'''_u$, the local boundary Γ_s can be decomposed into disjoint subsets of L_s, Γ_{su} and $\Gamma_{su'''}$. By the same reason, it can also be decomposed into disjoint sunsetsL_s, $\Gamma_{su'}$ and $\Gamma_{su''}$. By using these decompositions, along with the boundary conditions (27)-(30), equation (33) is obtained.

In order to simplify the above equation, we can select a test function v such that it vanishes over L_s, then equation (33) can be rewritten as

$$\begin{aligned} &\int_{\Omega_s} uv dx - \int_{\Omega_s} u''' v' dx + n_x \left[u''' v \right]_{\Gamma_{su}} \\ &= \int_{\Omega_s} v dx - n_x \left[\overline{u}''' v \right]_{\Gamma_{su'''}} \end{aligned} \tag{34}$$

With equation (5), one may discretize the local unsymmetric weak form (34) as

$$\sum_{I=1}^{N}\int_{\Omega_s}\left[\phi^I(x)\,v-\frac{d^3\phi^I(x)}{dx^3}v'\right]dx\hat{u}_I + \sum_{I=1}^{N}n_x\left[\frac{d^3\phi^I(x)}{dx^3}v\right]_{\Gamma_{su}}\hat{u}_I=\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}} \quad (35)$$

As discussed in Atluri (2004), one may choose the Heaviside function as the test function, and obtains

$$n_x\left[u'''v\right]_{\Gamma_{su}}+n_x\left[u'''v\right]_{L_s}+\int_{\Omega_s}uvdx =\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}} \quad (36)$$

Equation (36) can be discretized as

$$\sum_{I=1}^{N}n_x\left[\frac{d^3\phi^I(x)}{dx^3}v\right]_{\Gamma_{su}}\hat{u}_I+\sum_{I=1}^{N}n_x\left[\frac{d^3\phi^I(x)}{dx^3}v\right]_{L_s}\hat{u}_I + \sum_{I=1}^{N}\int_{\Omega_s}\phi^I(x)vdx\hat{u}_I=\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}} \quad (37)$$

In this study, the collocation method is used to impose the boundary conditions. For boundary node x_I, one has

$$\beta_1 u(x_I)+\beta_2 u'(x_I)+\beta_3 u''(x_I)=\beta_1\overline{u}+\beta_2\overline{u}'+\beta_3\overline{u}'' \quad (38)$$

It is noted that actually there exist only 1 or 2 terms in both sides of equation (38), depending on the combination of the boundary conditions.

However, as discussed in Section 2, we know that the MLS interpolation is hardly capable of approximating the third derivative of function u. Our numerical experiments based on LWF (33), cannot obtain any stable and convergent results (using some special parameter-combinations, one may happen to obtain some good results), using either weight function (20) or Heaviside function as test function. Hence, the unsymmetric local weak form (33) is not appropriate for the numerical implementation either. However, the mixed MLPG method based on this LWF (33) can still generate stable and convergent results, as will be described in next section.

3.2 Symmetric local weak form

Integrating (31) by parts twice yields the following symmetric local weak form,

$$n_x\left[u'''v\right]_{\Gamma_s}-n_x\left[u''v'\right]_{\Gamma_s} +\int_{\Omega_s}u''v''dx+\int_{\Omega_s}(u-1)vdx=0 \quad (39)$$

This is a symmetric local weak form, and both the trial and test function should be C^1 continuious. Imposing the boundary conditions (29) and (30), one obtains

$$n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[u'''v\right]_{\Gamma_{su}}+n_x\left[u'''v\right]_{L_s} -n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}}-n_x\left[u''v'\right]_{\Gamma_{su'}}-n_x\left[u''v'\right]_{L_s} +\int_{\Omega_s}u''v''dx+\int_{\Omega_s}(u-1)vdx=0 \quad (40)$$

In order to simplify the above equation, one can select a test function v such that it, and its derivative, vanish over L_s, then equation (40) can be rewritten as

$$n_x\left[u'''v\right]_{\Gamma_{su}}-n_x\left[u''v'\right]_{\Gamma_{su'}}+\int_{\Omega_s}u''v''dx+\int_{\Omega_s}uvdx =\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}} \quad (41)$$

By means of the MLS interpolation (5), the local symmetric weak form (41) can be discretized as

$$\sum_{I=1}^{N}n_x\left[\frac{d^3\phi^I(x)}{dx^3}v\right]_{\Gamma_{su}}\hat{u}_I-\sum_{I=1}^{N}n_x\left[\frac{d^2\phi^I(x)}{dx^2}v'\right]_{\Gamma_{su'}}\hat{u}_I + \sum_{I=1}^{N}\int_{\Omega_s}\left[\phi^I(x)\,v+\frac{d^2\phi^I(x)}{dx^2}v''\right]dx\hat{u}_I =\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}} \quad (42)$$

The collocation method is used to impose the boundary conditions. For boundary node x_I, one has

$$\beta_1 u(x_I)+\beta_2 u'(x_I)=\beta_1\overline{u}+\beta_2\overline{u}' \quad (43)$$

It is noted that actually there may exist 0 to 2 terms in each sides of equation (43), depending on the combination of the boundary conditions.

To illustrate their effectiveness, the MLPG methods based on the LWF (41) are used to solve equation (26) with boundary conditions:

$$u = u'' = 0 \text{ at } x = 0, 1 \tag{44}$$

The exact solution is equation (22).

For the purpose of error estimation and convergence studies, the following norm is used:

$$\|u\| = \left(\int_\Omega u^2 dx \right)^{\frac{1}{2}} \tag{45}$$

The relative errors for every order derivative are defined as

$$e_k = \frac{\left\| \left(\frac{d^{(k)}u}{dx^k} \right)^{\text{num}} - \left(\frac{d^{(k)}u}{dx^k} \right)^{\text{exact}} \right\|}{\left\| \left(\frac{d^{(k)}u}{dx^k} \right)^{\text{exact}} \right\|}, \; k = 0, 1, 2, 3 \tag{46}$$

For the present problem of a beam on elastic foundation, the normalized elastic energy stored in the system, at equilibrium, may be written as

$$W = \frac{1}{2} \int_0^1 \left(u'' + u^2 \right) dx$$

The relative error for the energy is defined as

$$e_p = \frac{\left| W^{\text{num}} - W^{\text{exact}} \right|}{\left| W^{\text{exact}} \right|} \tag{47}$$

At first, we use the cubic basis, i.e. m=4. Both the weight function in the MLS interpolation and the test function are taken to be equation (20). 41 nodes are used (h=0.025, with h being the distance between nodes). Fig. 6 shows the influence of the radius of the test domain on the relative errors e_0 and e_1, where the radius of the trial function domain is taken to be $4.5h$. From this figure, it can be found that the relative errors of the function u and its first derivative are less than 1% when the trial function domain is big enough ($> 2.5h$). It is noticed that the accuracy is not sensitive to the radius of the test domain from 3-5h. Fig. 7 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be $3.5h$. The results for the function u and its first derivative are highly accurate. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain from 3.5-5.5h, and less than 1%. For the linear (m=2) basis, the same trends can be observed. However, for m=2, a larger radius of the trial function domain should be chosen to obtain an accurate and stable result. In fact, in MLS, to increase the radius of the trial function domain is equivalent to increase m of the basis function.

The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be $3.5h$, and the radius of the trial domain is taken to be $4.5h$. For linear (m=2) basis, the radius of the test domain is taken to be $3.5h$, and the radius of the trial domain is taken to be $8h$. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 8, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 9 only for m=4. It can be seen that the present MLPG method has high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate, and of higher convergent rate than those from the linear (m=2) basis. However, the results for the third derivative are not very accurate, and the convergence rate for the relative error e_3 is not high.

3.3 Unsymmetric local weak form 2

Integrating (31) by parts three times yields the following local unsymmetric weak form,

$$\begin{aligned} &n_x \left[u''' v \right]_{\Gamma_s} - n_x \left[u'' v' \right]_{\Gamma_s} + n_x \left[u' v'' \right]_{\Gamma_s} \\ &- \int_{\Omega_s} u' v''' dx + \int_{\Omega_s} (u - 1) v dx = 0 \end{aligned} \tag{48}$$

This is an unsymmetric weak form, and the trial function can be C^0 continuious. Imposing the boundary conditions (29) and (30), one obtains

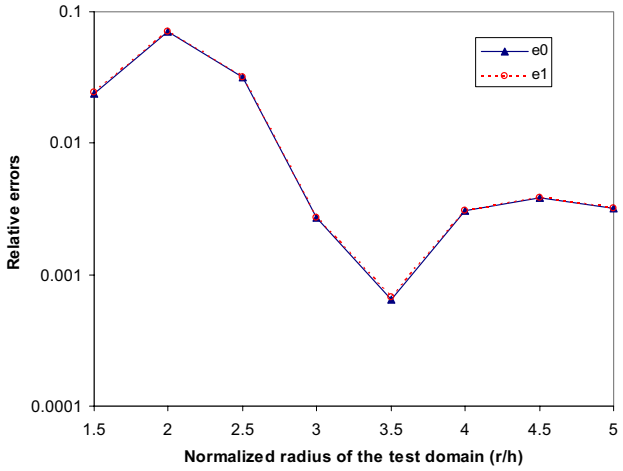

Figure 6 : The influence of the test domain size (41 nodes).

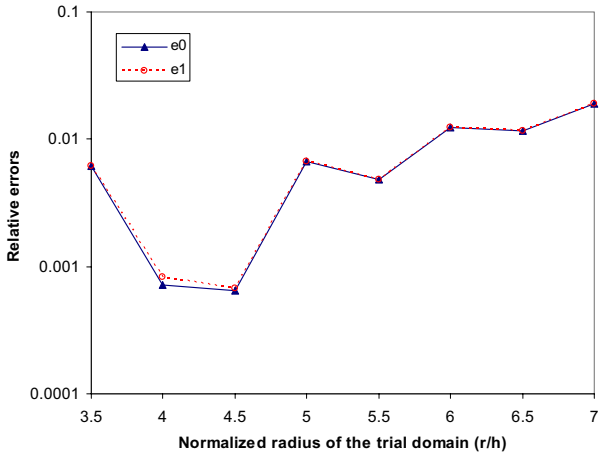

Figure 7 : The influence of the trial domain size (41 nodes).

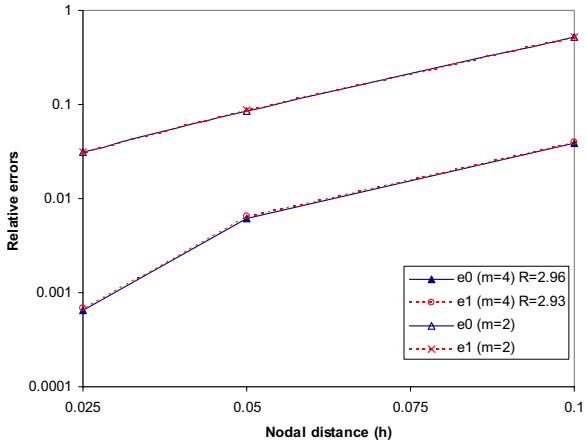

Figure 8 : Convergence rate in relative errors e_0 and e_1.

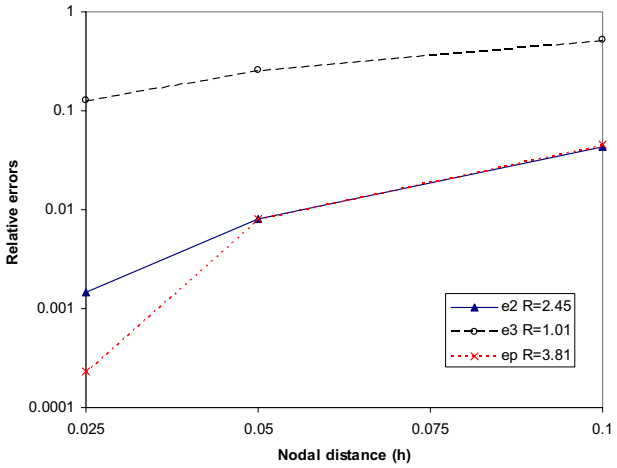

Figure 9 : Convergence rate in relative errors e_2, e_3, and e_p.

$$
\begin{aligned}
&n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[u'''v\right]_{\Gamma_{su}}+n_x\left[u'''v\right]_{L_s}\\
&-n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}}-n_x\left[u''v'\right]_{\Gamma_{su'}}-n_x\left[u''v'\right]_{L_s}\\
&+n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}}+n_x\left[u'v''\right]_{\Gamma_{su''}}+n_x\left[u'v''\right]_{L_s}\\
&-\int_{\Omega_s}u'v'''dx+\int_{\Omega_s}(u-1)vdx=0
\end{aligned}
\tag{49}
$$

One can select a test function v such that itself, and its first and second derivatives vanish over L_s. Such a test function is given in (20) and (21). Then equation (49) can be simplified as

$$
\begin{aligned}
&n_x\left[u'''v\right]_{\Gamma_{su}}-n_x\left[u''v'\right]_{\Gamma_{su'}}+n_x\left[u'v''\right]_{\Gamma_{su''}}\\
&+\int_{\Omega_s}uvdx-\int_{\Omega_s}u'v'''dx\\
&=\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}\\
&+n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}}-n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}}
\end{aligned}
\tag{50}
$$

With equation (5), one may discretize the local symmet-

ric weak form (50) as

$$\sum_{I=1}^{N} n_x \left[\frac{d^3\phi^I(x)}{dx^3} v\right]_{\Gamma_{su}} \hat{u}_I - \sum_{I=1}^{N} n_x \left[\frac{d^2\phi^I(x)}{dx^2} v'\right]_{\Gamma_{su'}} \hat{u}_I + \sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v''\right]_{\Gamma_{su''}} \hat{u}_I + \sum_{I=1}^{N} \int_{\Omega_s} \left[\phi^I(x) v - \frac{d\phi^I(x)}{dx} v'''\right] dx \hat{u}_I = \int_{\Omega_s} v dx - n_x \left[\overline{u}''' v\right]_{\Gamma_{su'''}} + n_x \left[\overline{u}'' v'\right]_{\Gamma_{su''}} - n_x \left[\overline{u}' v''\right]_{\Gamma_{su'}} \quad (51)$$

The collocation method is used to impose the only boundary condition, which does not appear in the left side of equation (51). For boundary node x_I, one has

$$\beta_1 u(x_I) = \beta_1 \overline{u} \quad (52)$$

In the numerical example, we also use the cubic basis, i.e. m=4. The weight function in MLS is taken to be as in equation (20), while the test function in LWF (49) is chosen to be as in equation (21). 41 nodes are used (h=0.025) to consider the influences of the radius of both test and trial domains. Fig. 10 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be 4.5h. From this figure, it can be found that the accuracy of the function, u as well as its first derivative, is high, when the trial function domain is big enough ($> 2.5h$). Similarly, it is noticed that the accuracy is not sensitive to the radius of the test domain from 3-5h. Fig. 11 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be 3.5h. The results for the function u, as well as its first derivative, are of high accuracy. The relative errors e_0 and e_1 are less than 1%, and are not sensitive to the radius of the trial function domain as the radius of the trial domain is greater than 4h. For the linear (m=2) basis, the same trends can be observed. Again, for m=2, a larger radius of the trial function domain should be chosen to obtain an accurate and stable result.

Similarly, the convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. The effects of the basis functions: linear (m=2) and cubic (m=4) bases are investigated. For cubic (m=4) basis, the radius

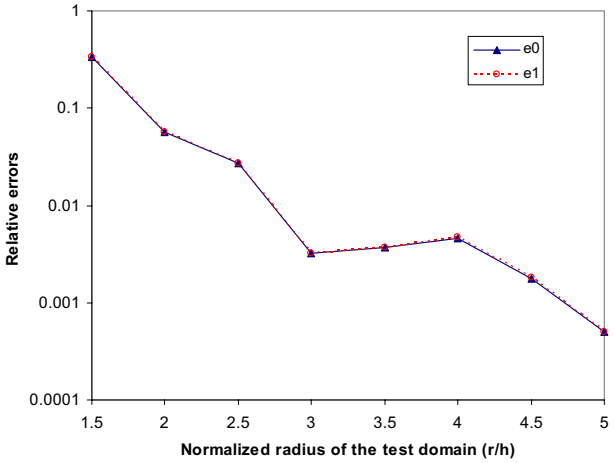

Figure 10 : The influence of the test domain size (41 nodes).

of the test domain is taken to be 3.5h, and the radius of the trial domain is taken to be 4.5h. For linear (m=2) basis, the radius of the test domain is taken to be 3h, and the radius of the trial domain is taken to be 6.5h. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 12, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 13 only for m=4. It can be seen that the present MLPG method has stable convergence rates for norms e_0, e_1, e_2, and e_p, and gives reasonably accurate results for the unknown variable, its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate, and are of a higher convergent rate than those from the linear (m=2) basis. However, the results for the third derivative are not very accurate, and the convergence rate for the relative error e_3 is not high.

3.4 *Unsymmetric local weak form 3*

Integrating (31) by parts four times yields the following unsymmetric local weak form,

$$n_x \left[u''' v\right]_{\Gamma_s} - n_x \left[u'' v'\right]_{\Gamma_s} + n_x \left[u' v''\right]_{\Gamma_s} - n_x \left[u v'''\right]_{\Gamma_s} + \int_{\Omega_s} u \frac{d^4 v}{dx^4} dx + \int_{\Omega_s} (u-1) v dx = 0 \quad (53)$$

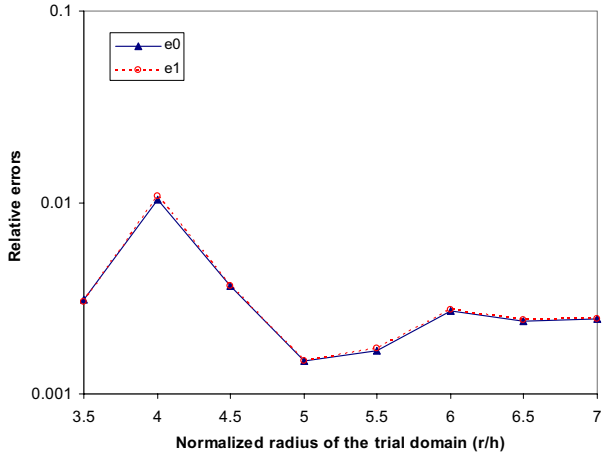

Figure 11 : The influence of the trial domain size (41 nodes).

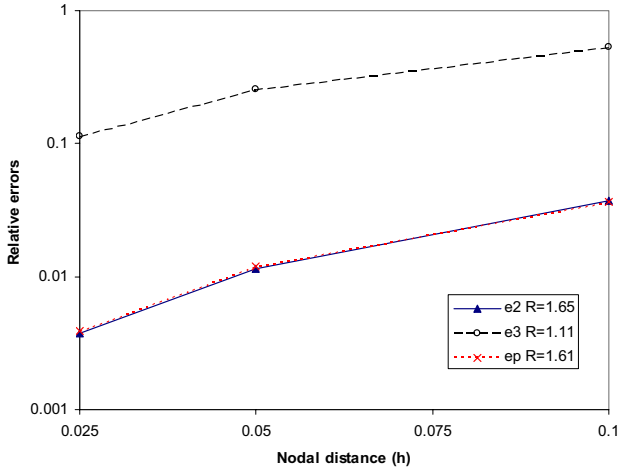

Figure 13 : Convergence rate in relative errors e_2, e_3, and e_p.

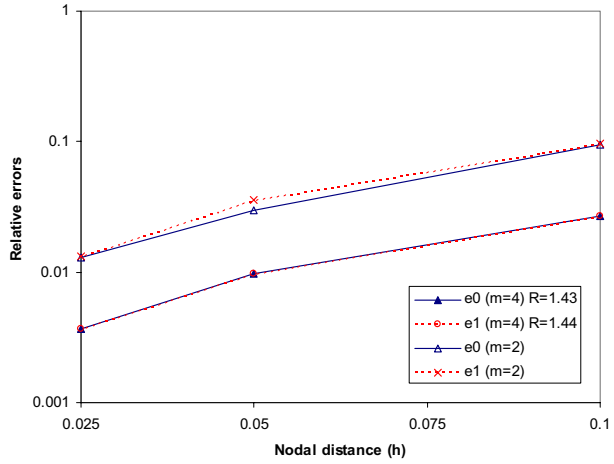

Figure 12 : Convergence rate in relative errors e_0 and e_1.

In this weak form, no derivatives of the trial function appear in the domain integration. Moreover, all the boundary conditions become "natural" boundary conditions. Then, there is no difficulty in the implementation of the boundary conditions.

Imposing the boundary conditions (29) and (30), one obtains

$$\begin{aligned}
&n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[u'''v\right]_{\Gamma_{su}}+n_x\left[u'''v\right]_{L_s}\\
&-n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}}-n_x\left[u''v'\right]_{\Gamma_{su'}}-n_x\left[u''v'\right]_{L_s}\\
&+n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}}+n_x\left[u'v''\right]_{\Gamma_{su''}}+n_x\left[u'v''\right]_{L_s}\\
&-n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}-n_x\left[uv'''\right]_{\Gamma_{su'''}}-n_x\left[uv'''\right]_{L_s}\\
&+\int_{\Omega_s}u\frac{d^4v}{dx^4}dx+\int_{\Omega_s}(u-1)vdx=0
\end{aligned}\tag{54}$$

In order to simplify the above equation, we can select a test function v, such that itself and its first, second and third derivatives vanish over L_s. Such a test function is given in (21). Then equation (54) can be rewritten as

$$\begin{aligned}
&n_x\left[u'''v\right]_{\Gamma_{su}}-n_x\left[u''v'\right]_{\Gamma_{su'}}+n_x\left[u'v''\right]_{\Gamma_{su''}}\\
&-n_x\left[uv'''\right]_{\Gamma_{su'''}}+\int_{\Omega_s}u\frac{d^4v}{dx^4}dx+\int_{\Omega_s}uvdx\\
&=\int_{\Omega_s}vdx-n_x\left[\overline{u}'''v\right]_{\Gamma_{su'''}}+n_x\left[\overline{u}''v'\right]_{\Gamma_{su''}}\\
&-n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}}+n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}
\end{aligned}\tag{55}$$

With equation (5), one may discretize the local unsymmetric weak form (55) as

$$\sum_{I=1}^{N} n_x \left[\frac{d^3\phi^I(x)}{dx^3} v\right]_{\Gamma_{su}} \hat{u}_I - \sum_{I=1}^{N} n_x \left[\frac{d^2\phi^I(x)}{dx^2} v'\right]_{\Gamma_{su'}} \hat{u}_I +$$
$$\sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v''\right]_{\Gamma_{su''}} \hat{u}_I - \sum_{I=1}^{N} n_x \left[\phi^I(x) v'''\right]_{\Gamma_{su'''}} \hat{u}_I$$
$$+ \sum_{I=1}^{N} \int_{\Omega_s} \phi^I(x) \left[v + \frac{d^4 v}{dx^4}\right] dx \hat{u}_I$$
$$= \int_{\Omega_s} v dx - n_x \left[\overline{u}''' v\right]_{\Gamma_{su'''}} + n_x \left[\overline{u}'' v'\right]_{\Gamma_{su''}}$$
$$- n_x \left[\overline{u}' v''\right]_{\Gamma_{su'}} + n_x \left[\overline{u} v'''\right]_{\Gamma_{su}} \tag{56}$$

To illustrate the effectiveness of the MLPG method base on the local weak form (54). The same problem as that in the Subsection 3.2 is solved here. We also start with the cubic basis, i.e. m=4, in the MLS interpolation. The weight function in MLS is taken to be equation (20), while the test function in LWF (55) is chosen to be equation (21), because of the C^3 continuity-requirement for the test function. 41 nodes are used (h=0.025). Fig. 14 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be 4.5h. From this figure, it can be found that the accuracy of the function u, and its first derivative, is very high when the trial function domain is big enough ($> 2.5h$). Similarly, it is noticed that the accuracy is not sensitive to the radius of the test domain from 3-6h, the relative errors e_0 and e_1 are less than 1%. Fig. 15 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be 4h. The results for the function u and its first derivative are of high accuracy. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain. For the linear (m=2) basis, the same trends can be observed. Again, for m=2, a larger radius of the trial function domain should be chosen to obtain an accurate and stable result. It can be found that the results in this subsection are more accurate that those in the previous subsections, since no derivative of the trial function appears in the domain integration in the local weak form.

Similarly, the convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions, and linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken

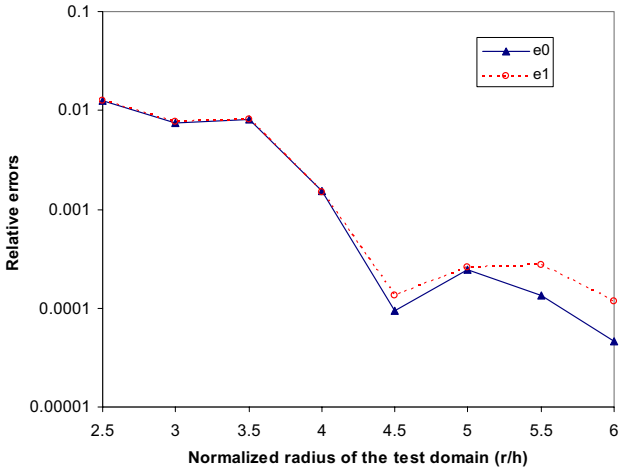

Figure 14 : The influence of the test domain size (41 nodes).

to be 4h, and the radius of the trial domain is taken to be 5h. For linear (m=2) basis, the radius of the test domain is taken to be 4h, and the radius of the trial domain is taken to be 6.5h. The results clearly illustrate stable convergence rates for both e_0 and e_1 for the present MLPG methods. Obviously, the results from the cubic (m=4) basis are more accurate than those from the linear (m=2) basis. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 16, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 17 only for m=4. It can be seen that the present MLPG method has high convergence rates for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, as well as its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate and of higher convergent rate than those from the linear (m=2) basis. However, the results for the third derivative are not very accurate, and the convergent rate for the relative error e_3 is low.

4 The First Kind of Mixed MLPG Methods

For a 4^{th} order ODE, there are two approaches to develop the MLPG mixed methods. In this section, the first kind of mixed MLPG methods will be developed to solve the 4^{th} order ODE. In the first kind of mixed MLPG methods, both the displacement as well as its second derivative are

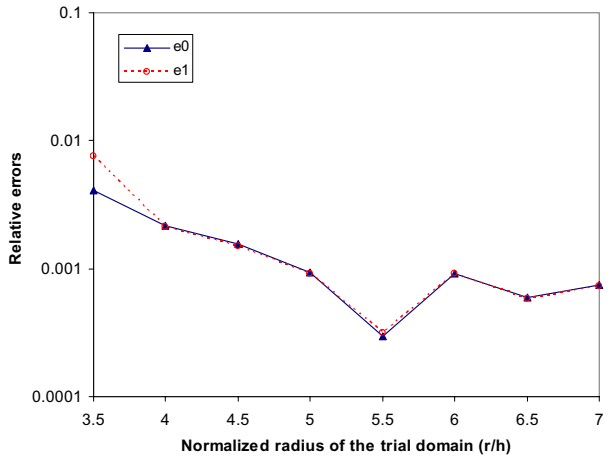

Figure 15 : The influence of the trial domain size (41 nodes).

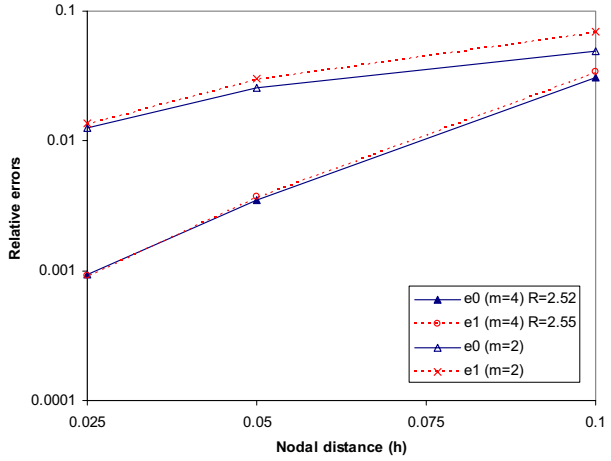

Figure 16 : Convergence rate in relative errors e_0 and e_1.

interpolated independently through meshless interpolation schemes, in order to eliminate the direct differentiation of the trial function to obtain its second and higher order derivatives of the shape function in the local weak forms.

First, we introduce a function z as

$$u'' = z \tag{57}$$

Then, equation (26) becomes

$$z'' + u - 1 = 0 \tag{58}$$

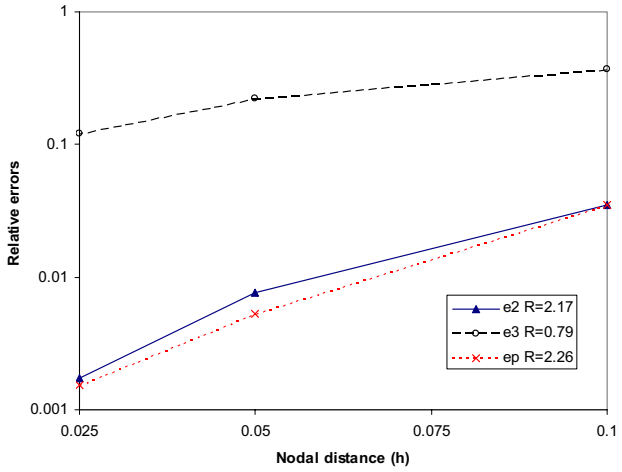

Figure 17 : Convergence rate in relative errors e_2, e_3, and e_p.

With the use of the MLS approximation, the function z can be independently interpolated as

$$z^h(x) = \sum_{I=1}^{N} \phi^I(x)\hat{z}^I,\ z^h(x_I) \equiv z^I \neq \hat{z}^I,\ x \in \Omega_x \tag{59}$$

4.1 Local weak form 1

Analogously, a local weak form can be derived as

$$n_x\left[z'v\right]_{\Gamma_s} - \int_{\Omega_s} z'v'dx + \int_{\Omega_s} (u-1)vdx = 0 \tag{60}$$

This weak form corresponds to the local unsymmetric weak form (32), but, now there exist only the first derivatives of z and v in the local weak form, and the requirement of the continuity for the trial function z is only C^0. As discussed in Section 2, the direct differentiation of u is not efficient in calculating its higher order derivatives everywhere. Thus, compared to the primal MLPG method based on LWF (32), the present mixed MLPG method should be more efficient.

In order to simplify the above equation, a test function v can be selected such that it vanishes over L_s, then equation (60) can be rewritten as

$$\begin{aligned} &n_x\left[z'v\right]_{\Gamma_{su}} + \int_{\Omega_s} uvdx - \int_{\Omega_s} z'v'dx \\ &= \int_{\Omega_s} vdx - n_x\left[\overline{z}'v\right]_{\Gamma_{su'''}} \end{aligned} \tag{61}$$

With the interpolations (5) and (59), one may discretize the local weak form (61) as

$$\sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v\right]_{\Gamma_{su}} \hat{z}_I + \sum_{I=1}^{N} \int_{\Omega_s} \left[\phi^I(x) v \hat{u}_I - \frac{d\phi^I(x)}{dx} v' \hat{z}_I\right] dx = \int_{\Omega_s} v dx - n_x \left[\overline{z}' v\right]_{\Gamma_{su'''}} \quad (62)$$

One may choose the Heaviside function as the test function, and obtains

$$n_x \left[z' v\right]_{\Gamma_{su}} + n_x \left[z' v\right]_{L_s} + \int_{\Omega_s} uv dx = \int_{\Omega_s} v dx - n_x \left[\overline{z}' v\right]_{\Gamma_{su'''}} \quad (63)$$

Equation (63) can be discretized as

$$\sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v\right]_{\Gamma_{su}} \hat{z}_I + \sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v\right]_{L_s} \hat{z}_I + \sum_{I=1}^{N} \int_{\Omega_s} \phi^I(x) v dx \hat{u}_I = \int_{\Omega_s} v dx - n_x \left[\overline{z}' v\right]_{\Gamma_{su'''}} \quad (64)$$

In equation (62) or (64), there are $2N$ independent unknowns (N second derivative variables $\hat{z}_I$ and N displacement variables $\hat{u}_I$), but the number of the equation is only N. However, one can reduce the number of the variables by relating z to u'' via the collocation methods, without any changes to equation (62) or (64). The collocation method is employed to enforce equation (57) only at each node x_I, instead of the entire solution domain. Thus, the function z at node x_I is expressed in terms of the function u at node x_I, as

$$z(x_I) = u''(x_I) \quad (65)$$

With the interpolations (5) and (59), the two sets of nodal variables can be transformed through a linear algebraic matrix,

$$\hat{z}_I = H_{IJ} \hat{u}_J \quad (66)$$

where the transformation matrix $\mathbf{H}$ is banded. Substituting equation (66) into equation (62) or (64), one can obtain a linear algebraic equation system of $\hat{u}_I$.

The collocation method is used to impose the boundary conditions. For boundary node x_I, one has

$$\beta_1 u(x_I) + \beta_2 u'(x_I) + \beta_3 z(x_I) = \beta_1 \overline{u} + \beta_2 \overline{u}' + \beta_3 \overline{z} \quad (67)$$

It is noted that actually there exist only 1 or 2 terms on both the sides of equation (67), depending on the combination of the boundary conditions.

To demonstrate the effectiveness of this method, the same numerical example as in Section 3 is considered in this section.

In this subsection, only the linear basis, i.e. m=2, is employed in the MLS interpolation, and the weight function in MLS is taken to be equation (20).

4.1.1 The test function is taken to be the weight function

At first, we take equation (20) as the test function, i.e. use the discretized equation (62). 41 nodes are used (h=0.025). Fig. 18 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be $5.1h$. It is noticed that the relative errors e_0 and e_1 (around 2%) are almost independent of the radius of the test domain from 1-3.5h. Fig. 19 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be $2.5h$. The results for the function u and its first derivative are acceptable (less than 10%). The relative errors e_0 and e_1 are not very sensitive to the radius of the trial function domain from 2.5-5.5h. The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes for linear (m=2) basis. The radius of the test domain is taken to be $2.5h$, and the radius of the trial domain is taken to be $2.5h$. The results clearly demonstrate stable convergence rates for both e_0 and e_1 for the present mixed MLPG methods. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 20. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 21. It can be seen that the present mixed MLPG method has stable convergence rate for norms e_0, e_1, e_2, and e_p, and gives reasonably accurate results for the unknown variable, its first and second derivatives, and the energy. However,

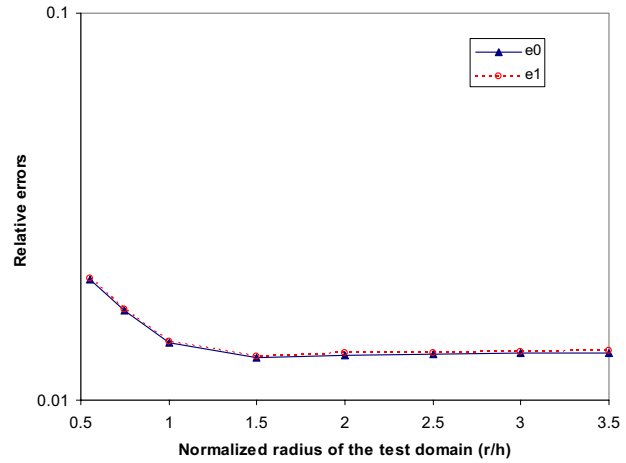

Figure 18 : The influence of the test domain size (41 nodes).

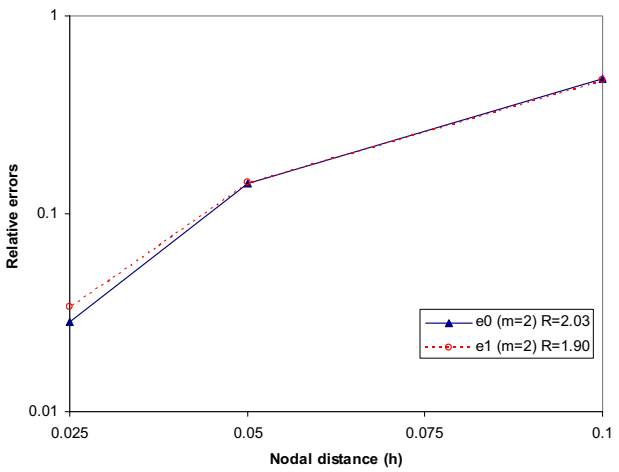

Figure 20 : Convergence rate in relative errors e_0 and e_1.

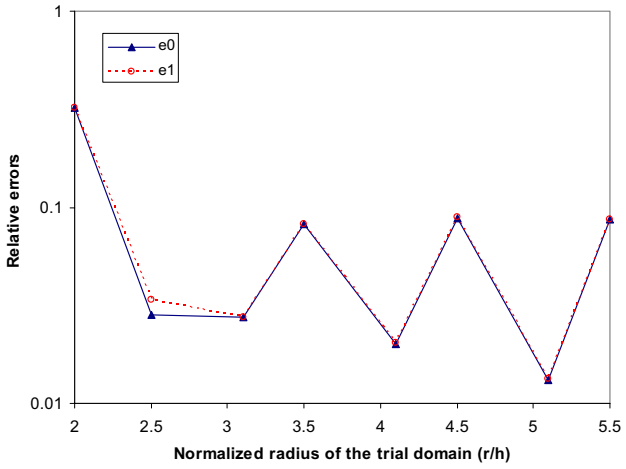

Figure 19 : The influence of the trial domain size (41 nodes).

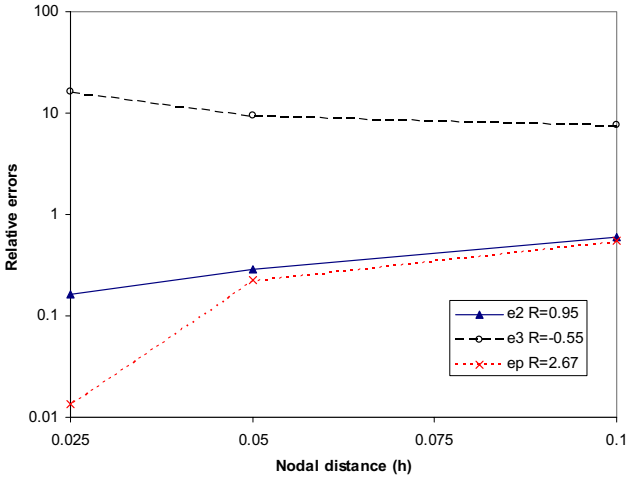

Figure 21 : Convergence rate in relative errors e_2, e_3, and e_p.

the results for the third derivative are not accurate, and the convergence rate for the relative error e_3 is not stable.

These figures show that, based on the same LWF (32), the mixed MLPG method can obtain stable results, although the accuracy of the results is not very high, while the primal MLPG method cannot work.

4.1.2 The test function is taken to be Heaviside function

Now, we choose Heaviside function as the test function, i.e. use the discretized equation (64). 41 nodes are used (h=0.025). Fig. 22 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be $5.1h$. It is noticed that the relative errors e_0 and e_1 (around 1%) are almost independent of the radius of the test domain from 0.5-2.5h. Fig. 23 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be $2.5h$. The results for the function u and its first derivative are acceptable. The relative errors e_0 and e_1 are not very sensitive to the radius of the trial function domain from 2.5-5.5h. The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes for linear (m=2) basis. The radius of

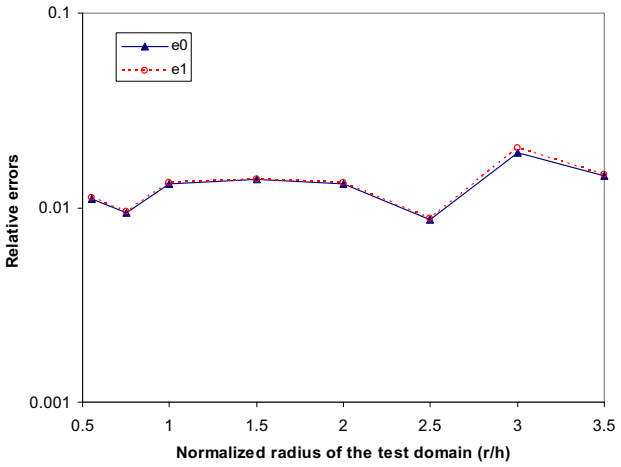

Figure 22 : The influence of the test domain size (41 nodes).

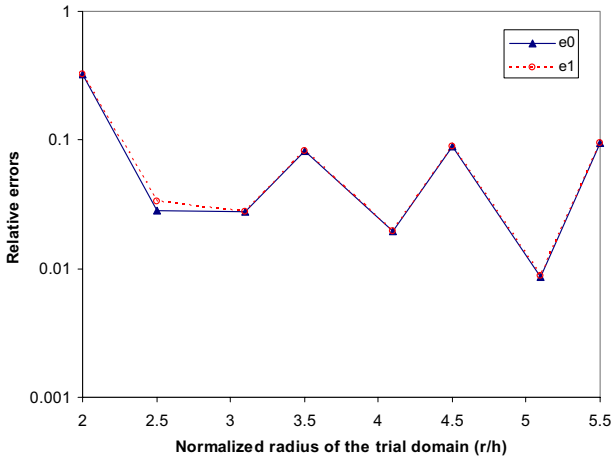

Figure 23 : The influence of the trial domain size (41 nodes).

the trial domain is taken to be $5.1h$, and the radius of the test domain is taken to be $2.5h$. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 24. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 25. It can be seen that the present mixed MLPG method has stable convergence rates for norms e_0, e_1, e_2, and e_p, and gives reasonably accurate results for the unknown variable, its first and second derivatives, and the energy. However, the results for the third derivative are not accurate, and the convergence rate for the relative error e_3 is not stable. The mixed MLPG method with the Heaviside function being the test function is more accurate than the mixed MLPG method with equation (20) being the test function.

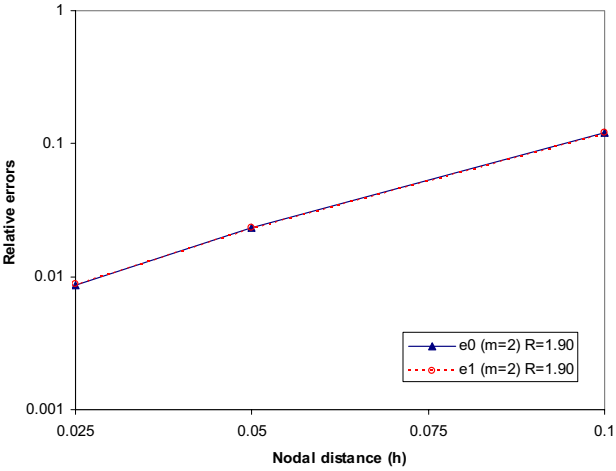

Figure 24 : Convergence rate in relative errors e_0 and e_1.

4.2 *Local weak form 2*

In LWF (60), there still exists the first derivative of z in the domain integration, which can be eliminated by integrating (60) by part once. Thus, another local weak form can be obtained as

$$n_x \left[z'v\right]_{\Gamma_s} - n_x \left[zv'\right]_{\Gamma_s} + \int_{\Omega_s} zv''dx + \int_{\Omega_s} (u-1)vdx = 0 \tag{68}$$

which corresponds to the local symmetric weak form (39), but no derivatives of either z or u appear in the domain integration. In LWF (39), there the second derivative of u occurs in the domain integration.

In order to simplify the above equation, we can select a test function v such that, itself and its first derivative, vanish over L_s, then equation (68) can be rewritten as

$$n_x \left[z'v\right]_{\Gamma_{su}} - n_x \left[zv'\right]_{\Gamma_{su'}} + \int_{\Omega_s} uvdx + \int_{\Omega_s} zv''dx = \int_{\Omega_s} vdx - n_x \left[\overline{z}'v\right]_{\Gamma_{su'''}} + n_x \left[\overline{z}v'\right]_{\Gamma_{su''}} \tag{69}$$

With equations (5) and (59), one may discretize the local

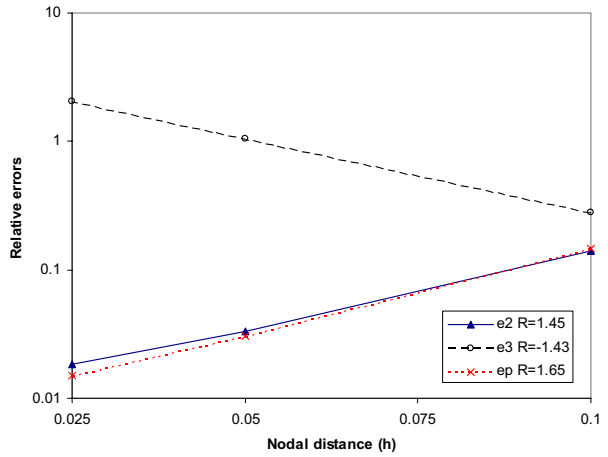

Figure 25 : Convergence rate in relative errors e_2, e_3, and e_p.

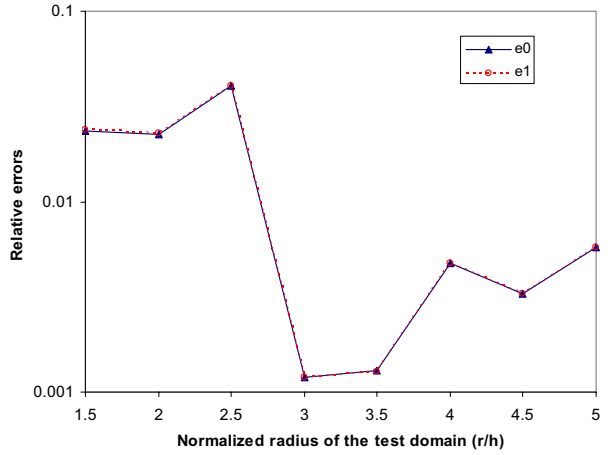

Figure 26 : The influence of the test domain size (41 nodes).

weak form (69) as

$$\sum_{I=1}^{N} n_x \left[\frac{d\phi^I(x)}{dx} v\right]_{\Gamma_{su}} \hat{z}_I - \sum_{I=1}^{N} n_x \left[\phi^I(x) v'\right]_{\Gamma_{su'}} \hat{z}_I + \sum_{I=1}^{N} \int_{\Omega_s} \left[\phi^I(x) v \hat{u}_I - \phi^I(x) v'' \hat{z}_I\right] dx = \int_{\Omega_s} v dx - n_x \left[\bar{z}' v\right]_{\Gamma_{su''}} + n_x \left[\bar{z} v'\right]_{\Gamma_{su''}} \quad (70)$$

Substituting equation (66) into equation (70), one can obtain a linear algebraic equation system of $\hat{u}_I$.

The collocation method is employed to impose the boundary conditions, as in equation (43).

To illustrate the effectiveness of this method, the same numerical example as in the previous section is considered here again. At first, the cubic basis (m=4) is used in the MLS interpolation. Both the weight function in MLS and the test function are taken to be equation (20). 41 nodes are used (h=0.025). Fig. 26 shows the influence of the radius of the test domain on the relative errors e_0 and e_1, where the radius of the trial function domain is taken to be $4.5h$. From this figure, it can be found that the accuracy of the function u and its first derivative is high when the test function domain is big enough ($> 2.5h$). It is noticed that the relative errors e_0 and e_1 are not sensitive to the radius of the test domain from 3-5h (less than 1%). Fig. 27 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be $3.5h$. The results for the function u and its first derivative are accurate. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain from 3.5-7h. For the linear (m=2) basis, the same trends can be observed. However, for m=2, a larger radius of the trial function domain should be chosen to obtain an accurate and stable result.

The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be $4h$, and the radius of the trial domain is taken to be $4.5h$. For linear (m=2) basis, the radius of the test domain is taken to be $3.5h$, and the radius of the trial domain is taken to be $8h$. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 28, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 29 only for m=4. It can be seen that the present mixed MLPG method has high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives reasonably accurate results for the unknown variable, its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate and of higher convergent rate than those from the linear (m=2) basis. However, the results for the third derivative are not very accurate, while the convergence rate for the relative error

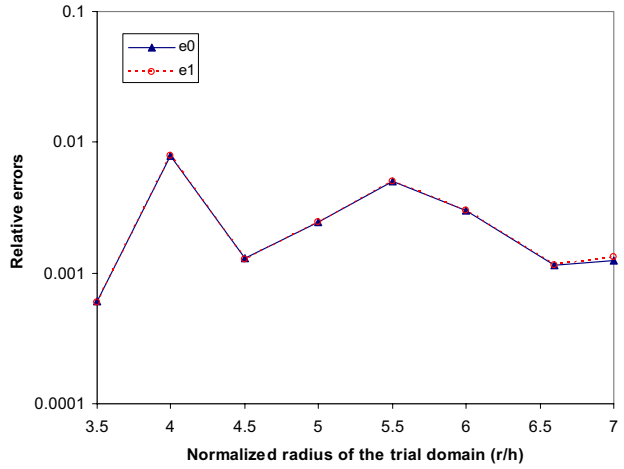

Figure 27 : The influence of the trial domain size (41 nodes).

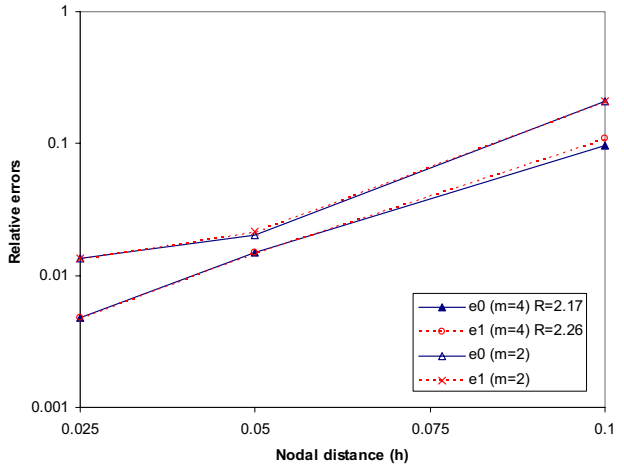

Figure 28 : Convergence rate in relative errors e_0 and e_1.

e_3 is high.

Compared to the corresponding primal MLPG method based on the local weak form (39), this mixed MLPG method requires less Gaussian points, and is more stable and accurate. The mixed MLPG method is cheaper and faster.

4.3 Local weak form 3

By means of the idea of the mixed MLPG method, the local weak form (53) can be rewritten as,

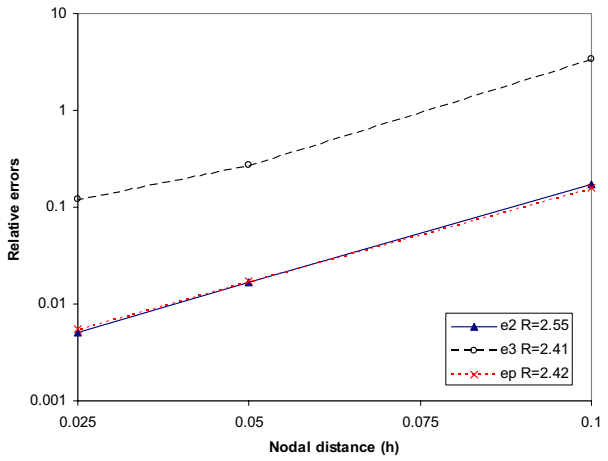

Figure 29 : Convergence rate in relative errors e_2, e_3, and e_p.

$$n_x\left[z'v\right]_{\Gamma_s}-n_x\left[zv'\right]_{\Gamma_s}+n_x\left[u'v''\right]_{\Gamma_s}$$
$$-n_x\left[uv'''\right]_{\Gamma_s}+\int_{\Omega_s}u\frac{d^4v}{dx^4}dx+\int_{\Omega_s}(u-1)vdx=0 \qquad (71)$$

Compared to the LWF (53), the LWF (71) does not have higher derivatives in the local boundary.

Imposing the boundary conditions (29) and (30), one obtains

$$n_x\left[\overline{z}'v\right]_{\Gamma_{su'''}}+n_x\left[z'v\right]_{\Gamma_{su}}+n_x\left[z'v\right]_{L_s}$$
$$-n_x\left[\overline{z}v'\right]_{\Gamma_{su''}}-n_x\left[zv'\right]_{\Gamma_{su'}}-n_x\left[zv'\right]_{L_s}$$
$$+n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}}+n_x\left[u'v''\right]_{\Gamma_{su''}}+n_x\left[u'v''\right]_{L_s}$$
$$-n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}-n_x\left[uv'''\right]_{\Gamma_{su'''}}-n_x\left[uv'''\right]_{L_s}$$
$$+\int_{\Omega_s}u\frac{d^4v}{dx^4}dx+\int_{\Omega_s}(u-1)vdx=0 \qquad (72)$$

In order to simplify the above equation, we can select a test function v such that, itself and its first, second and third derivatives vanish over L_s. Such a test function is given in (21). Then equation (72) can be rewritten as

$$
\begin{aligned}
&n_x\left[z'v\right]_{\Gamma_{su}} - n_x\left[zv'\right]_{\Gamma_{su'}} + n_x\left[u'v''\right]_{\Gamma_{su''}} \\
&- n_x\left[uv'''\right]_{\Gamma_{su'''}} + \int_{\Omega_s} u\frac{d^4v}{dx^4}dx + \int_{\Omega_s} uvdx \\
&= \int_{\Omega_s} vdx - n_x\left[\overline{z}'v\right]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} \\
&- n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}} + n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}
\end{aligned} \tag{73}
$$

With the interpolations (5) and (59), one may discretize the local symmetric weak form (73) as

$$
\begin{aligned}
&\sum_{I=1}^{N} n_x\left[\frac{d\phi^I(x)}{dx}v\right]_{\Gamma_{su}}\hat{z}_I - \sum_{I=1}^{N} n_x\left[\phi^I(x)v'\right]_{\Gamma_{su'}}\hat{z}_I + \\
&\sum_{I=1}^{N} n_x\left[\frac{d\phi^I(x)}{dx}v''\right]_{\Gamma_{su''}}\hat{u}_I - \sum_{I=1}^{N} n_x\left[\phi^I(x)v'''\right]_{\Gamma_{su'''}}\hat{u}_I \\
&+ \sum_{I=1}^{N}\int_{\Omega_s}\phi^I(x)\left[v + \frac{d^4v}{dx^4}\right]dx\hat{u}_I \\
&= \int_{\Omega_s} vdx - n_x\left[\overline{z}'v\right]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} \\
&- n_x\left[\overline{u}'v''\right]_{\Gamma_{su'}} + n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}
\end{aligned} \tag{74}
$$

Substituting equation (66) into equation (74), one can obtain a linear algebraic equation system of $\hat{u}_I$.

The numerical results still start with the cubic basis, i.e. m=4, in the MLS approximation. The weight function in MLS is taken to be equation (20), while the test function in LWF (73) is chosen to be equation (21). 41 nodes are used (h=0.025). Fig. 30 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be $4.5h$. From this figure, it can be found that the accuracy of the function u and its first derivative is very high when the trial function domain is big enough ($> 2.5h$). Similarly, it is noticed that the accuracy is not sensitive to the radius of the test domain from 3-6h. Fig. 31 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be $4h$. The results for the function u, as well as its first derivative, are of high accuracy. The relative errors e_0 and e_1 are less than 1% for the radius of the trial function domain between 3.5-7h. For the linear (m=2) basis, the same trends can be observed. However, in this mixed MLPG method, for m=2, a larger radius of the trial function domain is needed, in order to obtain an accurate and stable result.

Similarly, the convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be $4h$, and the radius of the trial domain is taken to be $4.5h$. For linear (m=2) basis, the radius of the test domain is taken to be $4h$, and the radius of the trial domain is taken to be $6.5h$. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 32, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 33 only for m=4. It can be seen that the present MLPG method has high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, its first and second derivatives, and the energy. This mixed MLPG method can obtain the almost the same accurate results for both m=2 and m=4. However, the results for the third derivative are not very accurate, and the convergence rate for the relative error e_3 is low.

Compared to the corresponding primal MLPG method based on the local weak form (53), this mixed MLPG method requires less Gaussian points, and is more stable and accurate, especially for m=2. The mixed MLPG method is cheaper and faster.

5 The Second Kind of Mixed MLPG Methods

In the first kind of mixed MLPG methods, there still exists the first derivative of z or u in the local weak forms. To avoid fully the appearance of any derivatives in the local weak forms, in this section, the second kind of the mixed MLPG methods are introduced. In the second kind of mixed MLPG methods, the displacement, its first derivative, second derivative, as well as the third derivative are all interpolated independently through meshless interpolation schemes. The second kind of mixed MLPG method is developed by introducing the following 3 functions

$$
\begin{aligned}
u' &= g \\
g' &= z \\
z' &= q
\end{aligned} \tag{75}
$$

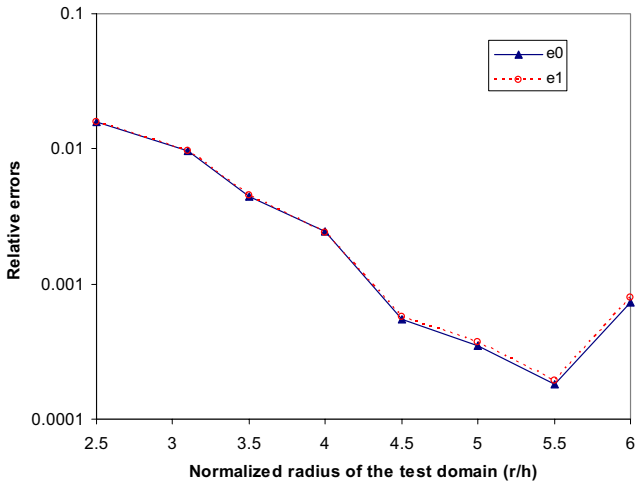

Figure 30 : The influence of the test domain size (41 nodes).

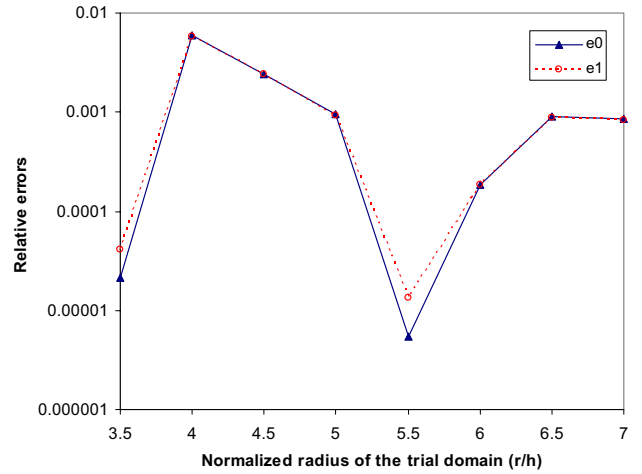

Figure 31 : The influence of the trial domain size (41 nodes).

Then, equation (26) becomes

$$q' + u - 1 = 0 \tag{76}$$

With the use of the MLS approximation, the functions g, z, and q can be independently interpolated as

$$g^h(x) = \sum_{I=1}^{N} \phi^I(x)\,\hat{g}^I,\ g^h(x_I) \equiv g^I \neq \hat{g}^I,\ x \in \Omega_x \tag{77}$$

$$z^h(x) = \sum_{I=1}^{N} \phi^I(x)\,\hat{z}^I,\ z^h(x_I) \equiv z^I \neq \hat{z}^I,\ x \in \Omega_x \tag{78}$$

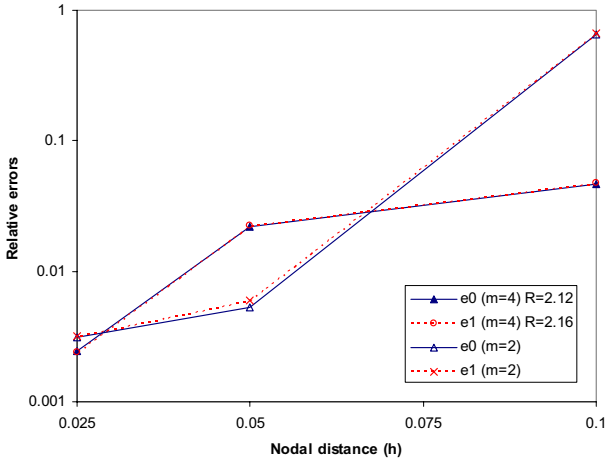

Figure 32 : Convergence rate in relative errors e_0 and e_1.

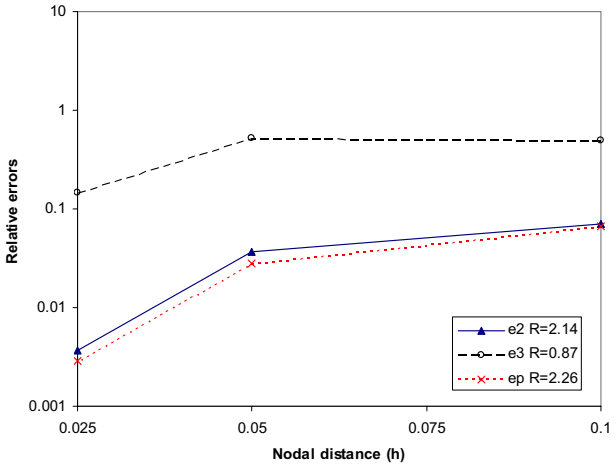

Figure 33 : Convergence rate in relative errors e_2, e_3, and e_p.

$$q^h(x) = \sum_{I=1}^{N} \phi^I(x)\,\hat{q}^I,\ q^h(x_I) \equiv q^I \neq \hat{q}^I,\ x \in \Omega_x \tag{79}$$

5.1 The local weak form 1

Analogously, the following local weak forms can be derived, as

$$n_x [qv]_{\Gamma_s} - \int_{\Omega_s} qv'dx + \int_{\Omega_s} (u-1)vdx = 0 \tag{80}$$

This weak form corresponds to the unsymmetric weak form (32) or (60). However, no derivative of the trial functions appear in this local weak form.

In order to simplify the above equation, one can select a test function v such that it vanishes over L_s; then equation (80) can be rewritten as

$$n_x [qv]_{\Gamma_{su}} + \int_{\Omega_s} uvdx - \int_{\Omega_s} qv'dx = \int_{\Omega_s} vdx - n_x [\overline{q}v]_{\Gamma_{su'''}} \tag{81}$$

With equations (5) and (79), one may discretize the local weak form (81) as

$$\sum_{I=1}^{N} n_x \left[\phi^I(x)\, v\right]_{\Gamma_{su}} \hat{q}_I + \sum_{I=1}^{N} \int_{\Omega_s} \phi^I(x) \left[v\hat{u}_I - v'\hat{q}_I\right] dx = \int_{\Omega_s} vdx - n_x [\overline{q}v]_{\Gamma_{su'''}} \tag{82}$$

In equation (82), there are $2N$ independent unknowns (N second derivative variables $\hat{q}_I$ and N displacement variables $\hat{u}_I$), but the number of the equation is only N. However, one can reduce the number of the variables by transforming the variables g, z, and q back to the displacement variable via the collocation methods, without any changes to equation (82). The collocation method is employed to enforce equation (75) only at each node x_I, instead of the entire solution domain. Thus, the functions g, z, and q at node x_I are expressed in terms of the derivatives of the related function at node x_I, as

$$\begin{aligned} g(x_I) &= u'(x_I) \\ z(x_I) &= g'(x_I) \\ q(x_I) &= z'(x_I) \end{aligned} \tag{83}$$

With the interpolations (5) and (77)-(79), the two related sets of nodal variables can be transformed through a linear algebraic matrix,

$$\begin{aligned} \hat{g}_I &= G_{IJ}\hat{u}_J \\ \hat{z}_I &= G_{IJ}\hat{g}_I \\ \hat{q}_I &= G_{IJ}\hat{z}_I \end{aligned} \tag{84}$$

where the transformation matrix $\mathbf{G}$ is banded. Through (84), the nodal variables of the function g, z, and q can be related to the nodal variable of function u, as

$$\begin{aligned} \hat{g}_I &= G_{IJ}\hat{u}_J \\ \hat{z}_I &= R_{IJ}\hat{u}_J \\ \hat{q}_I &= T_{IJ}\hat{u}_J \end{aligned} \tag{85}$$

where $\mathbf{R} = \mathbf{G}^2$, and $\mathbf{T} = \mathbf{G}^3$.

Substituting equation (85) into equation (82), one can obtain a linear algebraic equation system of $\hat{u}_I$.

However, our numerical experiments based on LWF (80), cannot obtain any stable and convergent results (using some special parameter-combinations, one may happen to obtain some good results), using either weight function (20) or Heaviside function as test function. Hence, this local weak form is not appropriate for the numerical implementation either. This may be since the errors introduced by the collocation method are enlarged by $\mathbf{T} = \mathbf{G}^3$.

5.2 The local weak form 2

By using the auxiliary functions (75), the local weak form (39) can be rewritten as

$$n_x [qv]_{\Gamma_s} - n_x \left[zv'\right]_{\Gamma_s} + \int_{\Omega_s} zv''dx + \int_{\Omega_s} (u-1)vdx = 0 \tag{86}$$

Compared to the local symmetric weak form (39), the LWF (86) has no derivative of the trial function in either domain integration or local boundary integration.

In order to simplify the above equation, we can select a test function v such that, itself and its first derivative, vanishes over L_s. then equation (86) can be rewritten as

$$n_x [qv]_{\Gamma_{su}} - n_x \left[\overline{z}v'\right]_{\Gamma_{su'}} + \int_{\Omega_s} uvdx + \int_{\Omega_s} zv''dx = \int_{\Omega_s} vdx - n_x [\overline{q}v]_{\Gamma_{su'''}} + n_x \left[\overline{z}v'\right]_{\Gamma_{su''}} \tag{87}$$

With the interpolations (5) and (78), one may discretize the local weak form (87) as

$$\sum_{I=1}^{N} n_x \left[\phi^I(x)\, v\right]_{\Gamma_{su}} \hat{q}_I - \sum_{I=1}^{N} n_x \left[\phi^I(x)\, v'\right]_{\Gamma_{su'}} \hat{z}_I + \sum_{I=1}^{N} \int_{\Omega_s} \left[\phi^I(x)\, v\hat{u}_I - \phi^I(x)\, v''\hat{z}_I\right] dx = \int_{\Omega_s} vdx - n_x [\overline{q}v]_{\Gamma_{su'''}} + n_x \left[\overline{z}v'\right]_{\Gamma_{su''}} \tag{88}$$

Substituting equation (85) into equation (88), one can obtain a linear algebraic equation system of $\hat{u}_I$.

The collocation method is employed to impose the boundary conditions, as in equation (43).

To illustrate the effectiveness of this method, we consider the same numerical example as in previous section again. At first, we use the cubic basis, i.e. m=4 in the MLS interpolation. Both the weight function in MLS, and the test function, are taken to be as in equation (20). 41 nodes are used (h=0.025). Fig. 34 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be 4.5h. From this figure, it can be found that the accuracies of the function u and its first derivative are very high, when the test function domain is big enough $(> h)$. The relative errors e_0 and e_1 are less than 0.01%. It is noticed that the accuracy is almost independent of the radius of the test domain from 1.5-5h. Fig. 35 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be 3.5h. The results for the function u and its first derivative are highly accurate. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain from 3.5-7h. For the linear (m=2) basis, the same trends can be observed. In this method, for m=2, the requirement of a larger radius of the trial function domain is not needed.

The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be 4h, and the radius of the trial domain is taken to be 5h. For linear (m=2) basis, the radius of the test domain is taken to be 3.5h, and the radius of the trial domain is taken to be 4.5h. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 36, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 37 only for m=4. It can be seen that the present mixed MLPG method has very high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate, and are of higher convergent rate than those from the linear (m=2) basis. However, the results for the third derivative are not very accurate, while the convergence rate for the relative error e_3 is very high.

Compared to the corresponding primal MLPG method based on the local weak form (39), this mixed MLPG method requires less Gaussian points, is more stable, and the results is of two-orders higher accuracy. This mixed MLPG method is also more accurate than the corresponding first kind of mixed MLPG method.

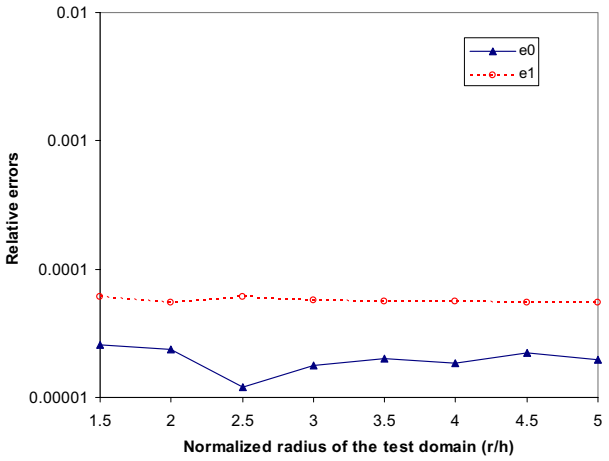

Figure 34 : The influence of the test domain size (41 nodes).

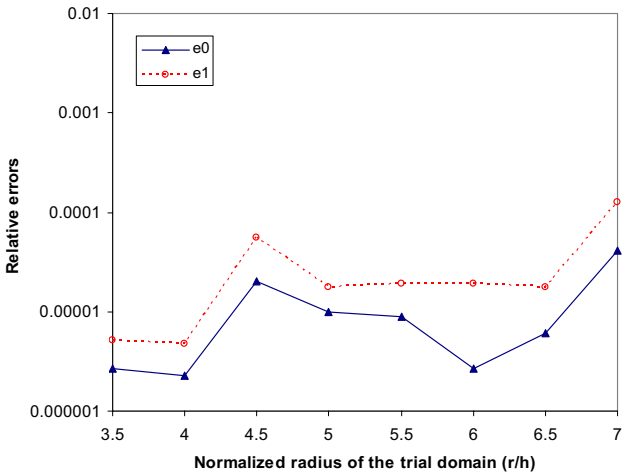

Figure 35 : The influence of the trial domain size (41 nodes).

5.3 *The local weak form 3*

By using the auxiliary functions (75), the local weak form (48) can be rewritten as

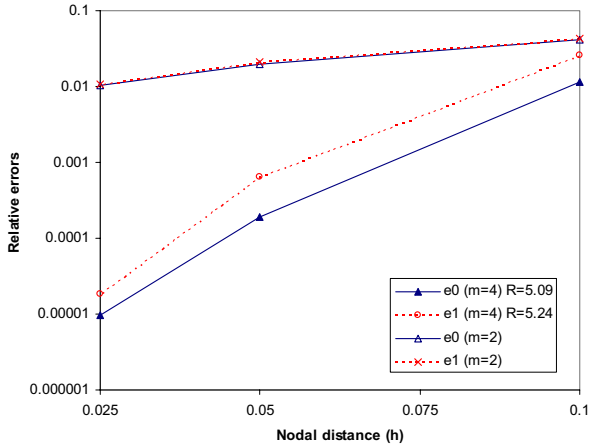

Figure 36 : Convergence rate in relative errors e_0 and e_1.

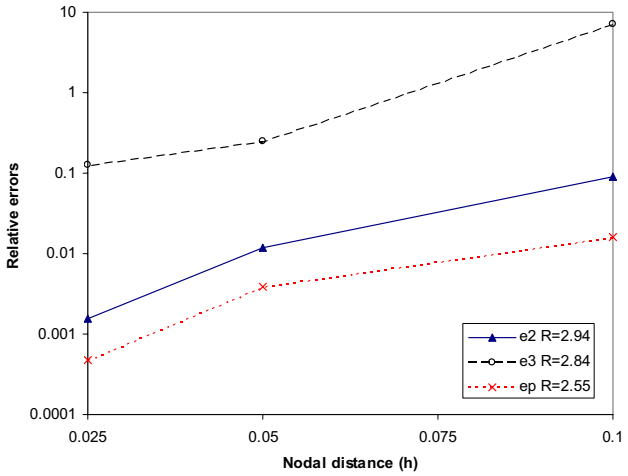

Figure 37 : Convergence rate in relative errors e_2, e_3, and e_p.

$$n_x[qv]_{\Gamma_s} - n_x\left[zv'\right]_{\Gamma_s} + n_x\left[gv''\right]_{\Gamma_s} - \int_{\Omega_s} gv'''dx + \int_{\Omega_s}(u-1)vdx = 0 \tag{89}$$

Compared to the local symmetric weak form (48), the LWF (89) has no derivatives of the trial function in either domain integration or local boundary integration.

Imposing the boundary conditions (29) and (30), one obtains

$$\begin{aligned} &n_x[\overline{q}v]_{\Gamma_{su'''}} + n_x[qv]_{\Gamma_{su}} + n_x[qv]_{L_s} \\ &- n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} - n_x\left[zv'\right]_{\Gamma_{su'}} - n_x\left[zv'\right]_{L_s} \\ &+ n_x\left[\overline{g}v''\right]_{\Gamma_{su'}} + n_x\left[gv''\right]_{\Gamma_{su''}} + n_x\left[gv''\right]_{L_s} \\ &- \int_{\Omega_s} gv'''dx + \int_{\Omega_s}(u-1)vdx = 0 \end{aligned} \tag{90}$$

In order to simplify the above equation, one can select a test function v such that, itself and its first and second derivatives, vanish over L_s. Such test functions are given in (20) and (21). Then equation (90) can be rewritten as

$$\begin{aligned} &n_x[qv]_{\Gamma_{su}} - n_x\left[zv'\right]_{\Gamma_{su'}} + n_x\left[gv''\right]_{\Gamma_{su''}} \\ &+ \int_{\Omega_s} uvdx - \int_{\Omega_s} gv'''dx \\ &= \int_{\Omega_s} vdx - n_x[\overline{q}v]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} - n_x\left[\overline{g}v''\right]_{\Gamma_{su'}} \end{aligned} \tag{91}$$

With the interpolations (5) and (78), one may discretize the local weak form (91) as

$$\begin{aligned} &\sum_{I=1}^{N} n_x\left[\phi^I(x)v\right]_{\Gamma_{su}}\hat{q}_I - \sum_{I=1}^{N} n_x\left[\phi^I(x)v'\right]_{\Gamma_{su'}} \\ &+ \sum_{I=1}^{N} n_x\left[\phi^I(x)v''\right]_{\Gamma_{su''}}\hat{g}_I \\ &+ \sum_{I=1}^{N}\int_{\Omega_s}\phi^I(x)\left[\hat{u}_I v - \hat{g}_I v'''\right]dx \\ &= \int_{\Omega_s} vdx - n_x[\overline{q}v]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} - n_x\left[\overline{g}v''\right]_{\Gamma_{su'}} \end{aligned} \tag{92}$$

The collocation method is used to impose the boundary condition as in equation (52). Substituting equation (85) into equation (92), one can obtain a linear algebraic equation system of $\hat{u}_I$.

The collocation method is used to impose the boundary conditions, as in equation (43).

To illustrate the effectiveness of this method, we consider the same numerical example again. Similarly, we start with the cubic basis, i.e. m=4, in the MLS interpolation. The weight function in MLS is taken to be equation (20),

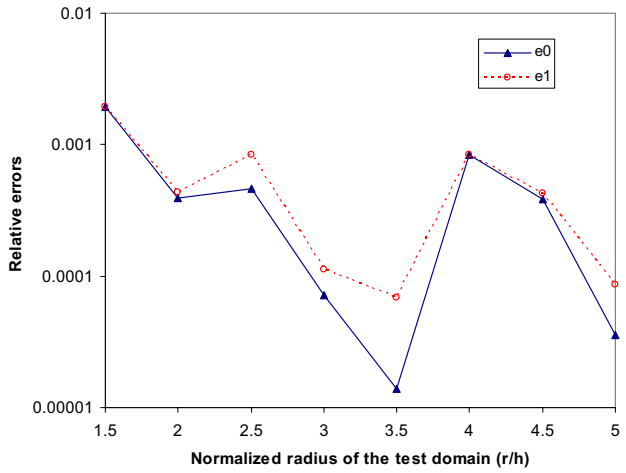

Figure 38 : The influence of the test domain size (41 nodes).

while the test function in LWF (49) is chosen to be equation (21). 41 nodes are used (h=0.025). Fig. 38 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be 4.5h. From this figure, it can be found that the accuracy of the function u, as well as its first derivative, is very high when the test function domain is big enough ($> h$). The relative errors e_0 and e_1 are less than 0.1%, when the radius of the test domain is greater than 1.5h. It is noticed that the accuracy is not sensitive to the radius of the test domain from 1.5-5h. Fig. 39 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be 3.5h. The results for the function u and its first derivative are high accurate. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain from 3.5-7h. For the linear (m=2) basis, the same trends can be observed. In this method, for m=2, the requirement of a larger radius of the trial function domain is not needed either.

The convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be 4h, and the radius of the trial domain is taken to be 5h. For linear (m=2) basis, the radius of the test domain is taken to be 3.5h, and the radius of the trial domain is taken to be 4.5h. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 40, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 41 only for m=4. It can be seen that the present mixed MLPG method has very high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, its first and second derivatives, and the energy. The results from the cubic (m=4) basis are more accurate, and are of higher convergent rate than those from the linear (m=2) basis, although in this method the results from m=2 are already very accurate. However, the results for the third derivative are not very accurate, while the convergence rate for the relative error e_3 is very high.

Compared to the corresponding primal MLPG method based on the local weak form (48), this mixed MLPG method requires less Gaussian points, is more stable, and the results is of two-orders higher accuracy. This mixed MLPG method is more accurate, cheaper and faster.

5.4 *The local weak form 4*

By using the auxiliary functions (75), the local weak form (53) can be rewritten as

$$n_x [qv]_{\Gamma_s} - n_x \left[zv'\right]_{\Gamma_s} + n_x \left[gv''\right]_{\Gamma_s} - n_x \left[uv'''\right]_{\Gamma_s} + \int_{\Omega_s} u\frac{d^4 v}{dx^4}dx + \int_{\Omega_s} (u-1)vdx = 0 \tag{93}$$

Compared to the local symmetric weak form (53), the LWF (93) has no derivative of the trial function in either domain integration or local boundary integration.

Imposing the boundary conditions (29) and (30), one obtains

$$\begin{aligned} &n_x [\overline{q}v]_{\Gamma_{su'''}} + n_x [qv]_{\Gamma_{su}} + n_x [qv]_{L_s} \\ &- n_x \left[\overline{z}v'\right]_{\Gamma_{su''}} - n_x \left[zv'\right]_{\Gamma_{su'}} - n_x \left[zv'\right]_{L_s} \\ &+ n_x \left[\overline{g}v''\right]_{\Gamma_{su'}} + n_x \left[gv''\right]_{\Gamma_{su''}} + n_x \left[gv''\right]_{L_s} \\ &- n_x \left[\overline{u}v'''\right]_{\Gamma_{su}} - n_x \left[uv'''\right]_{\Gamma_{su'''}} - n_x \left[uv'''\right]_{L_s} \\ &+ \int_{\Omega_s} u\frac{d^4 v}{dx^4}dx + \int_{\Omega_s} (u-1)vdx = 0 \end{aligned} \tag{94}$$

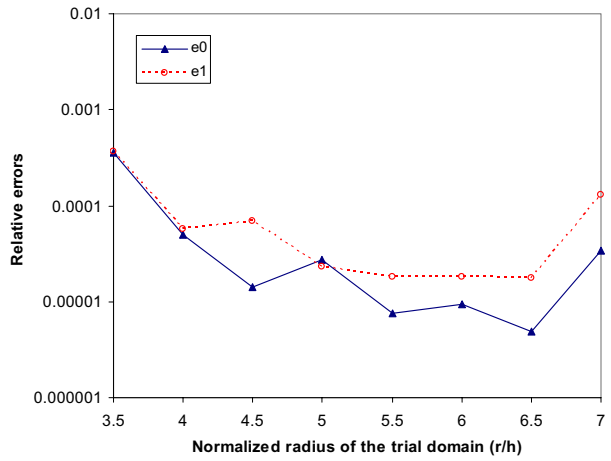

Figure 39 : The influence of the trial domain size (41 nodes).

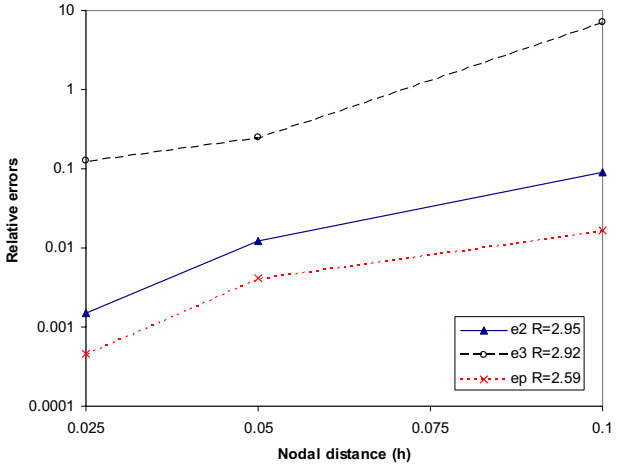

Figure 41 : Convergence rate in relative errors e_2, e_3, and e_p.

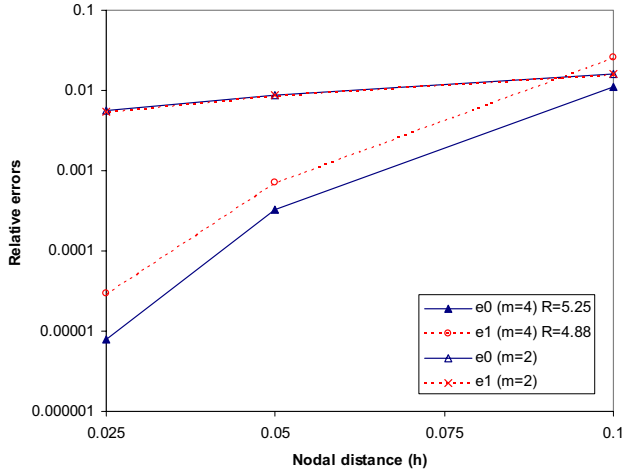

Figure 40 : Convergence rate in relative errors e_0 and e_1.

$$
\begin{aligned}
&n_x\left[qv\right]_{\Gamma_{su}} - n_x\left[zv'\right]_{\Gamma_{su'}} + n_x\left[gv''\right]_{\Gamma_{su''}} \\
&- n_x\left[uv'''\right]_{\Gamma_{su'''}} + \int_{\Omega_s} u\frac{d^4v}{dx^4}dx + \int_{\Omega_s} uvdx \\
&= \int_{\Omega_s} vdx - n_x\left[\overline{q}v\right]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} \\
&- n_x\left[\overline{g}v''\right]_{\Gamma_{su'}} + n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}
\end{aligned} \tag{95}
$$

With the interpolations (5) and (78), one may discretize the local symmetric weak form (95) as

$$
\begin{aligned}
&\sum_{I=1}^{N} n_x\left[\phi^I(x)\,v\right]_{\Gamma_{su}}\hat{q}_I - \sum_{I=1}^{N} n_x\left[\phi^I(x)\,v'\right]_{\Gamma_{su'}}\hat{z}_I + \\
&\sum_{I=1}^{N} n_x\left[\phi^I(x)\,v''\right]_{\Gamma_{su''}}\hat{g}_I - \sum_{I=1}^{N} n_x\left[\phi^I(x)\,v'''\right]_{\Gamma_{su'''}}\hat{u}_I \\
&+ \sum_{I=1}^{N}\int_{\Omega_s}\phi^I(x)\left[v + \frac{d^4v}{dx^4}\right]dx\hat{u}_I \\
&= \int_{\Omega_s} vdx - n_x\left[\overline{q}v\right]_{\Gamma_{su'''}} + n_x\left[\overline{z}v'\right]_{\Gamma_{su''}} \\
&- n_x\left[\overline{g}v''\right]_{\Gamma_{su'}} + n_x\left[\overline{u}v'''\right]_{\Gamma_{su}}
\end{aligned} \tag{96}
$$

In order to simplify the above equation, we can select a test function v such that it and its first, second and third derivatives vanish over L_s. Such a test function is given in (21). Then, equation (94) can be rewritten as

Substituting equation (85) into equation (96), one can obtain a linear algebraic equation system of $\hat{u}_I$.

Again, the same numerical example is considered to illustrate the effectiveness of this method. We also start

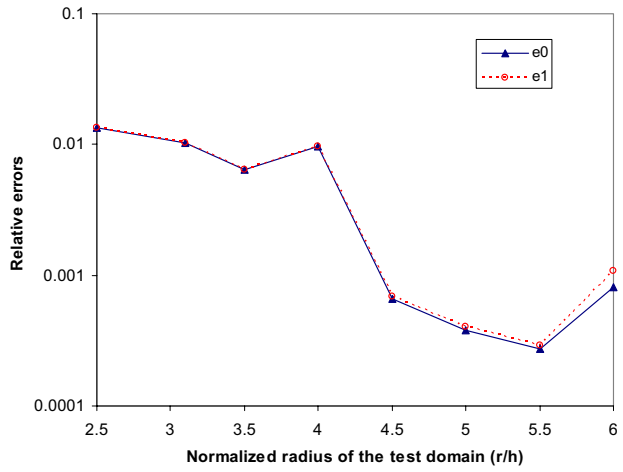

Figure 42 : The influence of the test domain size (41 nodes).

with the cubic basis, i.e. m=4, in the MLS interpolation. The weight function in MLS is taken to be equation (20), while the test function in LWF (95) is chosen to be equation (21). 41 nodes are used (h=0.025). Fig. 42 shows the influence of the radius of the test domain on the errors e_0 and e_1, where the radius of the trial function domain is taken to be 4.5h. From this figure, it can be found that the accuracy of the function u and its first derivative is high when the trial function domain is big enough ($\geq 2.5h$). Similarly, it is noticed that the accuracy is not sensitive to the radius of the test domain from 3-6h, and the relative errors e_0 and e_1 are less than 1%. Fig. 43 shows the influence of the radius of the trial domain on the errors e_0 and e_1, where the radius of the test domain is taken to be 4h. The results for the function u and its first derivative are of high accuracy. The relative errors e_0 and e_1 are not sensitive to the radius of the trial function domain. For the linear (m=2) basis, the same trends can be observed. In this method, for m=2, the requirement of a larger radius of the trial function domain is no longer needed.

Similarly, the convergence rate is investigated with three nodal configurations: 11, 21, and 41 nodes. We also consider the effects of the basis functions: linear (m=2) and cubic (m=4) bases are used in this investigation. For cubic (m=4) basis, the radius of the test domain is taken to be 4h, and the radius of the trial domain is taken to be 5h. For linear (m=2) basis, the radius of the test domain is taken to be 4h, and the radius of the trial domain is taken to be 6.5h. The relative errors e_0 and e_1 and the convergence rates R of the displacement and first derivative are depicted in Fig. 44, for both m=4 and m=2. The convergence rates R of the relative errors e_2, e_3 and e_p for the second, third derivatives and the energy, are plotted in Fig. 45 only for m=4. It can be seen that the present mixed MLPG method has very high rates of convergence for norms e_0, e_1, e_2, and e_p, and gives very accurate results for the unknown variable, its first and second derivatives, and the energy. However, the results for the third derivative are not very accurate, while the convergence rate for the relative error e_3 is very high. In this method the results from m=2 are also very accurate.

Compared to the corresponding primal MLPG method based on the local weak form (39), this mixed MLPG method requires less Gaussian points, is more stable and accurate. This mixed MLPG method is also more accurate than the corresponding first kind of mixed MLPG method. This mixed MLPG method possesses very high convergence rates for the displacement and its first to third derivatives.

6 Conclusions

Both the primal and mixed MLPG methods are presented fro the 4^{th} order ordinary differential equations. Various local weak forms are developed. In the first kind of mixed MLPG methods, both the displacement and its second derivative are interpolated independently through the moving least squares interpolation scheme. In the second kind of mixed MLPG methods, the displacement, its first derivative, the second derivative and the third derivative are all interpolated independently through the moving least squares interpolation scheme. The mixed MLPG methods avoid the occurrence of high order derivatives of the primary trial function, in the local weak forms, and thus reduce the continuity-requirement on the trial function. The mixed MLPG methods are far more efficient than the primal MLPG methods. The numerical examples demonstrate that both the primal and mixed MLPG methods obtain accurate results and possess excellent rate of convergence for the displacement, its first and second derivatives, and the energy. However, among them, the second kind of mixed MLPG methods give more stable and accurate results, and possess very high convergence rates, even for the third derivative.

Acknowledgement: This work was supported by the U. S. Army Research Office, and the U. S. Army Re-

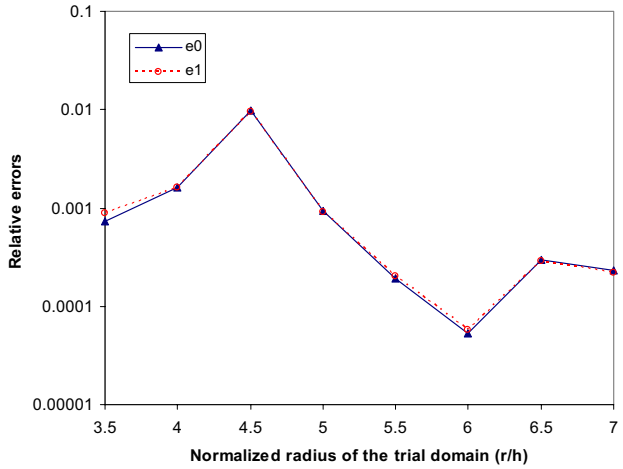

Figure 43 : The influence of the trial domain size (41 nodes).

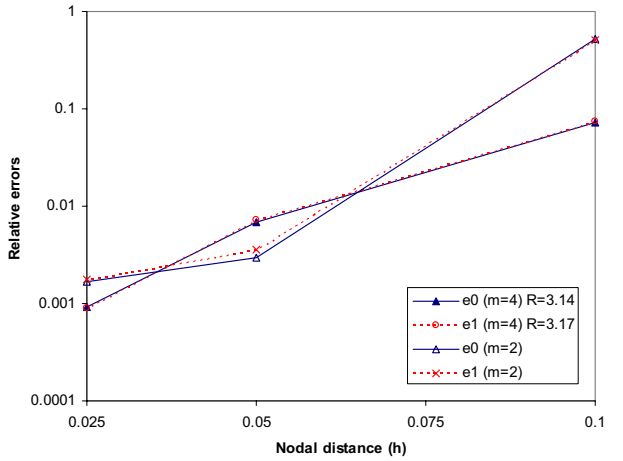

Figure 44 : Convergence rate in relative errors e_0 and e_1.

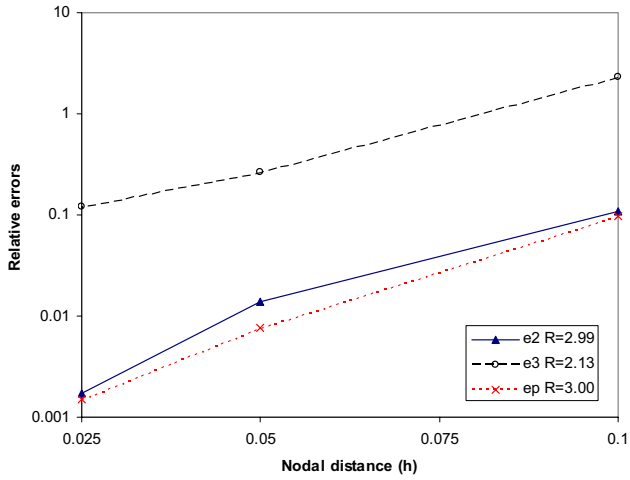

Figure 45 : Convergence rate in relative errors e_2, e_3, and e_p.

search Laboratory, under a cooperative research agreement with the University of California at Irvine. The Cognizant Program Official at the U. S. Army Research Labs is Dr. R. Namburu. Partial support for this work was also provided by the Office of Naval Research, in the program directed by Dr. Y.D.S. Rajapakse.

References

Atluri, S. N. (2004): *The meshless method (MLPG) for domain & bie discretizations*. Tech Science Press, USA, 680 pages.

Atluri, S. N.; Cho, J. Y.; Kim, H. G. (1999): Analysis of the beams, using the meshless local Petrov-Galerkin method, with generalized moving least squares interpolations. *Comput. Mech.* 24: 334-347.

Atluri, S. N.; Han, Z. D.; Rajendran, A. M. (2004): A new implementation of the meshless finite volume method, through the MLPG "mixed" approach. *CMES: Computer Modeling in Engineering & Sciences* 6(6): 491-513.

Atluri, S. N.; Shen, S. (2002a): *The meshless local Petrov-Galerkin (MLPG) method*. Tech Science Press, USA, 440 pages.

Atluri, S. N.; Shen, S. (2002b): The meshless local Petrov-Galerkin (MLPG) method: A simple & less-costly alternative to the finite element and boundary element method. *CMES: Computer Modeling in Engineering & Sciences* 3 (1): 11-52.

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

Batra, R. C.; Ching, H. K. (2002): Analysis of elastodynamic deformations near a crack/notch tip by the meshless local Petrov-Galerkin (MLPG) method. *CMES: Computer Modeling in Engineering & Sciences* 3 (6): 717-730.

Cho, J. Y.; Atluri, S. N. (2001): Analysis of shear flexible beams, using the meshless local Petrov-Galerkin

method, based on a locking-free formulation. *Eng. Comput.* 18 (1-2): 215-240.

Han, Z. D.; Atluri, S. N. (2004a): Meshless local Petrov-Galerkin (MLPG) approaches for solving 3D problems in elasto-statics. *CMES: Computer Modeling in Engineering & Sciences* 6 (2): 169-188.

Han, Z. D.; Atluri, S. N. (2004b): A meshless local Petrov-Galerkin (MLPG) approaches for solving 3-Dimesnsional elasto-dynamics. *CMC: Computer, Materials & Continua* 1 (2): 129-140.

Li, Q.; Shen, S.; Han, Z. D.; Atluri, S. N. (2003): Application of Meshless Local Petrov-Galerkin (MLPG) to Problems with Singularities, and Material Discontinuities, in 3-D Elasticity. *CMES: Computer Modeling in Engineering & Sciences* 4(5): 567-581.

Lin, H.; Atluri, S. N. (2000): Meshless local Petrov-Galerkin (MPLG) method for convection-diffusion problems. *CMES: Computer Modeling in Engineering & Sciences* 1 (2): 45-60

Lin, H.; Atluri, S. N. (2001): The meshless local Petrov-Galerkin (MPLG) method for solving incompressible Navier-stokes equations. *CMES: Computer Modeling in Engineering & Sciences* 2 (2): 117-142

Long, S. Y.; Atluri, S. N. (2002): A meshless local Petrov-Galerkin method for solving the bending problem of a thin plate. *CMES: Computer Modeling in Engineering & Sciences* 3 (1): 53-63.

Raju, I. S.; Phillips, D. R. (2003): Further Developments in the MLPG Method for Beam Problems. *CMES: Computer Modeling in Engineering & Sciences* 4 (1): 141-160.

Shen, S.; Atluri, S. N. (2004): Multiscale simulation based on the meshless local Petrov-Galerkin (MLPG) method. *CMES: Computer Modeling in Engineering & Sciences* 5(3): 235-255.

Shen, S.; Atluri, S. N. (2005): A tangent stiffness MLPG method for atom/continuum multiscale simulation. *CMES: Computer Modeling in Engineering & Sciences* 7(1): 49-67.