Engineering Analysis with Boundary Elements 98 (2019) 310–327

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

journal homepage: www.elsevier.com/locate/enganabound

New strong formulation for material nonlinear problems based on the particle difference method

Young-Cheol Yoon [a,*], Peter Schaefferkoetter [b], Timon Rabczuk [c], Jeong-Hoon Song [b,*]

[a] Department of Civil Engineering, Myongji College, 356-1 Hongeun-Dong, Seodaemoon-Ku, Seoul 120-776, South Korea
[b] Department of Civil, Environmental, and Architectural Engineering, University of Colorado, Boulder, CO 80309, USA
[c] Institute of Structural Mechanics, Bauhaus-University Weimar, Marienstrasse 15, 99423 Weimar, Germany

ARTICLE INFO

Keywords:
Strong form meshfree method
Material nonlinear
Newton's method
Particle difference method
Double derivative approximation

ABSTRACT

In this paper, we propose a strong form method for analyzing material nonlinear problems. The method utilizes the Particle Difference Method (PDM) which is classified as a strong form meshfree method and discretizes the governing equations based on a complete nodal computation without any integral formulation. The conventional strong form meshfree methods could not explicitly handle the nonlinear material model since they are mostly discretized based on the Navier's equation where kinematic variables are unified into displacement. To explicitly treat the nonlinear constitutive equation in the framework of strong formulation, a double derivative approximation is devised, which removes the need for the use of second order derivative approximation. The momentum equation is directly discretized through the double derivative approximation and is linearized by Newton's method to yield an iterative procedure for finding a converged solution. Stresses and internal variables are updated by the return mapping algorithm and efficiency of the iterative procedure is dramatically improved by the algorithmic tangent modulus. The consistency of the double derivative approximation was shown by the reproducing test. The accuracy and robustness of the developed nonlinear procedure were then verified through various inelastic material problems in one and two-dimensions.

1. Introduction

The Particle Difference Method (PDM) is a strong form collocation method based on the moving least squares (MLS) minimization of Taylor approximation. The PDM discretizes the domain and boundary with a set of spatially distributed collocation points, where at each collocation point, the strong form governing equation and boundary conditions are evaluated. To this end, derivative approximation operators are implicitly constructed in terms of MLS-based Taylor approximation procedure. Though the collocation points should be appropriately distributed, mesh or grid structure is not required. A weighting function used in the MLS approach ensures the global continuity of the approximated point-wise solution and its derivative fields. Note that weighting function is defined with the size of the neighborhood, i.e. influence domain, around a given collocation point such that collocation points within the same neighborhood interact or influence one another. Unlike the conventional finite element method, the derivative approximation is not constrained by compatibility condition along the element boundary, and it automatically satisfies the reproducing property (or consistency condition). This is an important component for not only for the interpolation process, but also for the strong formulation based on direct discretization of governing equations. For a discussion on the mathematics of derivative approximation refer to Yoon and Song [26]. Gains in computational efficiency are achieved by the fact that PDM method requires no actual differentiation in the construction of derivative operators, especially in case of higher order derivatives. These gains are further elevated by the fact that the PDM operates on the strong form of the governing equation thereby negating the need for numerical integration with quadrature. However, so far, these merits of PDM have not been fully utilized in solving the solid mechanics problems since many meshfree methods are more or less associated with the weak form; excellent review for various meshfree methods can be found in Chen et al. [4]. Furthermore, the conventional strong form meshfree methods fail to properly address the nonlinear constitutive equations, i.e. plasticity. For example, the use of the deformation theory does not help improve the computational efficiency when iteratively updating the nonlinear constitutive model. To alleviate this difficulty, a plethora of meshfree methods were proposed based on the weak formulation [3,5,7,13–17,22]. In a broad sense, they substitute the meshfree approximation for the finite element approximation but cannot achieve the computational efficiency of the finite element approach. The solution procedure from the previous studies employed deformation theory whereas this paper utilizes the

* Corresponding authors.
E-mail addresses: ycyoon@mjc.ac.kr (Y.-C. Yoon), jh.song@colorado.edu (J.-H. Song).

https://doi.org/10.1016/j.enganabound.2018.10.015
Received 29 May 2018; Received in revised form 18 September 2018; Accepted 22 October 2018

return mapping stress update algorithm involving the algorithmic tangent modulus. In fact, meshfree methods, based either on the weak or the strong formulation, have mostly employed the Hencky's total deformation theory; by doing so, a faster and more effective stress update algorithm equipped with the algorithmic tangent modulus could not be used such that efficiencies equivalent to those realized with nonlinear FEM could not be acquired.

Recently, the PDM has been applied to various problems in mechanics such as weak and strong discontinuity problems [26,27], moving boundary problem [28] dynamic crack problem [10], proportionally damped and cracked concrete beam problem [24] and phase field analysis for solidification [6,20]. The PDM has been shown to successfully predict interface evolution and crack growth phenomena. However, when it comes to the material nonlinearity, the PDM also encounters the aforementioned difficulties. At present, many conventional meshfree methods working with the strong form have solved the Navier's equation which unifies kinematic variables into displacement but requires the second order derivative of displacement [1,9,11]. Recently, a new method that combines the finite difference and strong form collocation method has been developed and successfully applied for solving two dimensional and three dimensional linear elasticity problems with complex geometries [29]. However, in these formulations, the constitutive equation cannot be explicitly treated since the stress tensor or its rate form does not appear in the equilibrium equation. Navier's equation for analyzing solid mechanics problem is given by

$$\mu \nabla^2 \mathbf{u} + (\lambda + \mu)\nabla(\nabla \cdot \mathbf{u}) = -\mathbf{b} \quad \text{in } \Omega \tag{1}$$

where $\mathbf{b}$ is the body force, and λ and μ are Lamé constants. In fact, the Navier's equation is only effective for elastic material unless the Lamé constants are continuously updated to address the incremental deformation. However, both theory and application to follow such an approach have been barely developed, thereby preventing various existing plasticity models from being implemented in the Navier's equation's framework. Note that when solving the Navier's equation with the conventional C^0 approach, discretizing $\nabla^2\mathbf{u}$ and $\nabla(\nabla \bullet \mathbf{u})$ terms in Eq. (1) often entails numerical difficulties.

In a general nonlinear analysis, the constitutive model is given by the relation between stress and strain or their rate forms. To accelerate plastic stage computation, a stress integration algorithm involving the algorithmic tangent modulus is frequently employed. To solve the material nonlinearity problem involving these features, this study presents an iterative algorithm for the PDM. The equilibrium equation or linear momentum equation Eq. (2), expressed by the divergence of stress tensor, is directly discretized so that the constitutive equation can be explicitly implemented in the strong form. In other words, the linear momentum equation given as in Eq. (2) is directly solved in its current form:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = 0 \quad \text{in } \Omega \tag{2}$$

Note that unlike the Navier's equation, Eq. (2) does not include the second order derivative apparently. Instead, the first order differentiations appear both in the divergence computation for stress and the strain computation; the strain is expressed by the first order differentiation for the displacement and is included in the stress through the constitutive relation. Thus, it can be noticed that repeated appearance of the first order differentiation substitutes for the second order differentiation. This study separately discretizes the strain and divergence of stress in Eq. (2) by using a successive application of the first order derivative approximation. By so doing, the constitutive equation for inelastic material can be explicitly handled in the strong form. In the following chapters, this technique is named the double derivative approximation. It provides the PDM with computational efficiency in the strong formulation involving a nonlinear procedure; in fact, the computational efficiency is attributed to the flexibility of the MLS approximation where the derivatives of kinematic variables can be determined by interpolating the nodal variables. The PDM combined with double derivative approximation can handle kinematic variables only at the collocation point. Hence the strain is first computed by the derivative approximation for the nodal displacement and then the divergence of stress tensor is evaluated by the derivative approximation for the nodal stress; after the former process, the stress is determined from the strain together with the tangent modulus. Also, it will be shown that the derivative computation can be conducted using a global matrix that contains the first order derivative approximation for nodal shape function. The double derivative approximation is then calculated by a scalar product of the matrices. Although this approach seems somewhat unconventional to those who are familiar with the weak formulation like the finite element method, it is inevitable in the strong formulation like the PDM to explicitly deal with the constitutional model.

Topological constraints associated with meshing are circumvented with PDM since nodal computation is not limited by mesh or grid structure. In addition, the compatibility condition, a requirement for finite element shape functions, is automatically guaranteed during construction of approximated derivative operators due to the reproducing property [26,27]. The flexibility of incorporating the derivative approximation into the iterative algorithm for material nonlinearity problem will be shown. One concern in this study is, of course, the validity of the double derivative. The validity is investigated through the numerical experiment. Another concern is the effectiveness of the newly developed iterative algorithm which is designed to operate on the strong form of the governing equation. More specifically, the algorithm covers the residual equations for Newton method, and application of conventional inelastic material models.

This paper presents a nonlinear numerical procedure for solving material nonlinearity problems. The procedure includes discretization of equilibrium equation, construction of residual equations and development of return mapping algorithm combined with the algorithmic tangent modulus. During the iterative process, the double derivative approximation plays a key role in the construction of residual equations. The stress update via return mapping algorithm and introduction of the algorithmic tangent modulus effectively accelerates computation speed. Solving the total system yields the converged displacement solution which appropriately describes the inelastic behavior of material. The main merit of PDM, the ability to operate with the strong form without the need of a mesh, results in gains in computational efficiency when solving problems dealing with material nonlinearity. In future studies, this merit is expected to be extended to solving large deformation problems that involves both material *and* geometric nonlinearity.

2. One dimensional material nonlinear problem

2.1. Discretization of equilibrium equation

When a body force is absent, solid mechanics problem restricted to small displacement is described by the boundary value problem as follows

$$\nabla \cdot \boldsymbol{\sigma} = 0 \quad \text{in } \Omega \tag{3}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{in } \Gamma_t \tag{4}$$

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{in } \Gamma_u \tag{5}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\nabla \bullet \boldsymbol{\sigma}$ is the divergence of stress, $\bar{\mathbf{t}}$ is the traction, $\mathbf{n}$ is the unit normal vector to domain Γ_t, $\bar{\mathbf{u}}$ is the displacement prescribed along Γ_u, and $\mathbf{u}$ is the unknown displacement vector. In one-dimensional case, Eqs. (3)–(5) simply become $\frac{\partial \sigma}{\partial x} = 0$, $\sigma = \bar{t}$ and $u = \bar{u}$, respectively.

To derive one dimensional derivative approximation, m-th order Taylor polynomial is defined at an arbitrary position y as following

$$\begin{aligned} u_L^m(x, y) &= u(y) + \left(\tfrac{x-y}{\rho}\right)\tfrac{\rho}{1!}u^{(1)}(y) + \cdots + \left(\tfrac{x-y}{\rho}\right)^m \tfrac{\rho^m}{m!}u^{(m)}(y) \\ &= \mathbf{p}_m^T\left(\tfrac{x-y}{\rho}\right)\mathbf{a}(y) \end{aligned} \tag{6}$$

where $\mathbf{p}_m^T(\frac{x-y}{\rho})$ is mth order polynomial vector and $\mathbf{a}(y)$ is unknown derivative coefficient vector which includes all the derivatives of $u(x)$ at y up to mth order. ρ is a radius of support covering neighbor nodes of y. The superscript with a parenthesis indicates the order of derivative. The derivative coefficients, $u(y)$, …, $u^{(m)}(x)$, are obtained by minimizing the residual equation based on the moving least squares method; note that the coefficients are known as the particle derivative approximation in Yoon and Song [26] and Yoon and Song [28]. The αth order derivative of $u(x)$ is then approximated by the following derivative approximation:

$$\frac{\partial^\alpha u(x)}{\partial x^\alpha} := \sum_I \Phi_I^{[\alpha]}(x) u_I \tag{7}$$

where u_I denotes the nodal solution for displacement at x_I (i.e., $u(x_I)$), and $\Phi_I^{[\alpha]}(x)$ is the generalized shape function.$\Phi_I^{[\alpha]}(x)$ denotes the αth order derivative approximation of $\Phi_I^{[(0)]}(x)$ and is analogous with the meshfree shape function in the conventional meshfree methods.

Navier's equation unifies the unknown kinematic variables into the displacement. In linear elastic problems the tangent modulus reduces to Young's Modulus, E, and Navier's equation is simply written as

$$\frac{\partial \sigma(x)}{\partial x} = \frac{\partial}{\partial x}(E\varepsilon(x)) = E\frac{\partial^2 u(x)}{\partial x^2} \tag{8}$$

However, in the nonlinear region, the tangent modulus changes after yielding, causing a change in the stress state and now is dependent on the spatial coordinate, x. As a remedy, the equilibrium equation as written in Eq. (3) is directly discretized to explicitly consider the constitutive equation. In this process, it is advantageous to preserve a conventional nonlinear computational framework like the nonlinear FEM procedure. To implement the elastoplastic tangent modulus, the equilibrium equation is rewritten as following

$$\frac{\partial \sigma(x)}{\partial x} = \frac{\partial}{\partial x}\left(E^{E-P}(x)\varepsilon(x)\right) = \frac{\partial}{\partial x}\left(E^{E-P}(x)\frac{\partial u(x)}{\partial x}\right) \tag{9}$$

Take note that second order differentiation is not used in Eq. (9). Instead, application of the first order differentiation occurs twice, first in the divergence computation for stress, and then for the strain computation for displacement. By using the derivative approximation, Eq. (9) can then be discretized at x_J as follows

$$\begin{aligned}\sum_I \Phi_I^{[(1)]}(x_J)\sigma_I &= \sum_I \Phi_I^{[(1)]}(x_J)\left(E^{E-P}(x_I)\varepsilon_I\right)\\ &= \sum_I \Phi_I^{[(1)]}(x_J)\left(E^{E-P}(x_I)\sum_K \Phi_K^{[(1)]}(x_I)u_K\right)\\ &= \sum_I\sum_K \Phi_I^{[(1)]}(x_J)E^{E-P}(x_I)\Phi_K^{[(1)]}(x_I)u_K\end{aligned} \tag{10}$$

where $\Phi_I^{[(1)]}(x_J)$ indicates the first order derivative approximation for the nodal shape function $\Phi_I^{[(0)]}(x_J)$. Note that $\Phi_I^{[(1)]}(x_J)$ is evaluated for node I at x_J while $\Phi_J^{[(1)]}(x_I)$ for node J at x_I.; while the constitutive equation, $\sigma_I = E^{E-P}(x_I)\varepsilon_I$, involves x_I or subscript I, the compatibility equation, $\varepsilon_I = \sum_K \Phi_K^{[(1)]}(x_I)u_K$ associates K to compute a kinetic variable at x_I.

Note that the repeated use of the first order derivative approximation successfully substitutes for the second order derivative; hereinafter this process is named the double derivative approximation. The linear momentum equation can be discretized without the need for second order derivative approximation even though no weak formulation is employed. In addition, the double derivative approximation enables the elastoplastic tangent modulus to be explicitly implemented in the discrete system. In one dimensional case, Eq. (10) can be assembled for all interior nodes to yield the system of equations as follows

$$\begin{aligned}&\begin{pmatrix}\sum_I \Phi_I^{[(1)]}(x_1)\sigma_I\\ \vdots\\ \sum_I \Phi_I^{[(1)]}(x_N)\sigma_I\end{pmatrix}\\ &= \begin{pmatrix}\Phi_1^{[(1)]}(x_1) & \cdots & \Phi_N^{[(1)]}(x_1)\\ \vdots & \ddots & \vdots\\ \Phi_1^{[(1)]}(x_N) & \cdots & \Phi_N^{[(1)]}(x_N)\end{pmatrix}\begin{pmatrix}E^{E-P}(x_1) & & \\ 0 & \ddots & 0\\ & & E^{E-P}(x_N)\end{pmatrix}\\ &\times\begin{pmatrix}\Phi_1^{[(1)]}(x_1) & \cdots & \Phi_N^{[(1)]}(x_1)\\ \vdots & \ddots & \vdots\\ \Phi_1^{[(1)]}(x_N) & \cdots & \Phi_N^{[(1)]}(x_N)\end{pmatrix}\begin{pmatrix}\Delta u_1\\ \vdots\\ \Delta u_N\end{pmatrix}\end{aligned} \tag{11}$$

Here, Eq. (11) can be rewritten in a matrix form as following

$$\begin{pmatrix}\sum_I \Phi_I^{[(1)]}(x_1)\sigma_I\\ \vdots\\ \sum_I \Phi_I^{[(1)]}(x_N)\sigma_I\end{pmatrix} = \mathbf{\Psi}^T\mathbf{D}\mathbf{\Psi}\,\Delta\mathbf{u} \tag{12}$$

where it can be noticed that $\mathbf{\Psi}^T$ is analogous to $\mathbf{\Psi}$ specifically for one dimensional case. However, for multi-dimensional case, they are not the same. $\mathbf{\Psi}$, $\mathbf{D}$, and $\Delta\mathbf{u}$ are written as below

$$\mathbf{\Psi} = \begin{pmatrix}\Phi_1^{[(1)]}(x_1) & \cdots & \Phi_N^{[(1)]}(x_1)\\ \vdots & \ddots & \vdots\\ \Phi_1^{[(1)]}(x_N) & \cdots & \Phi_N^{[(1)]}(x_N)\end{pmatrix} \tag{13}$$

$$\mathbf{D} = Diag\left(E^{E-P}(x_1), \cdots, E^{E-P}(x_N)\right) \tag{14}$$

$$\Delta\mathbf{u} = \left(\Delta u_1, \cdots, \Delta u_N\right)^T \tag{15}$$

where $\mathbf{\Psi}^T$ comes from the divergence calculation of stress and $\mathbf{\Psi}$ functions as a derivative operator for strain computation. It is interesting that $\mathbf{\Psi}^T\mathbf{D}\mathbf{\Psi}$ in Eq. (12) looks quite similar with the integrand ($\mathbf{B}^T\mathbf{D}\mathbf{B}$) of stiffness term ($\int_\Omega \mathbf{B}^T\mathbf{D}\mathbf{B}\,d\Omega$) in the weak formulation. Here, similarity exists between $\mathbf{\Psi}$ and $\mathbf{B}$ which is so-called 'B' matrix found in the stiffness matrix. Eq. (12) is the strong form of the discrete equilibrium equation, and yet it resembles the integrand of stiffness matrix of weak form. It is satisfied at the node level whereas the weak form is satisfied in a variational (or an average) sense involving numerical quadrature.

As seen in Eq. (13), $\mathbf{\Psi}$ (or $\mathbf{\Psi}^T$) is assemblage of the first order derivative approximations for nodal shape functions. The Ith row of $\mathbf{\Psi}$ contains the derivative approximations for all neighbor nodes of x_I. The Jth column is a collection of all the derivative approximations of node J for different x_I's. As seen in Eq. (10), the nodal equilibrium equation given in Eq. (10) is equivalent to the row vector of Eq. (12) as following

$$\sum_I \Phi_I^{[(1)]}(x_J)\sigma_I = \left(Row^{J-th}\,\mathbf{\Psi}^T\right)\mathbf{D}\mathbf{\Psi}\,\Delta u \tag{16}$$

where $Row^{J-th}\,\mathbf{\Psi}^T$ indicates the Jth row vector of $\mathbf{\Psi}^T$. On the other hand, note that the natural and essential boundary conditions are treated as in conventional strong form based meshfree methods like the PDM. When a node belongs to the boundary, the corresponding boundary condition is implemented in assemblage of the system of equations. Construction of the aforementioned discrete equations are based on the global coordinate system so that computer programming becomes quite simple; unlike the FEM, local coordinate system and mapping between local and global coordinate are not required. Compared to the conventional PDM using the second order derivative approximation, the newly developed method does not demand remarkable additional complicated process nor computational effort.

2.2. One dimensional iterative solution procedure

To solve material nonlinear problems, an iterative procedure is required and the residual equations are constructed by linearizing governing equations by way of Newton's method. The iterative algorithm seeks

a converged solution at each incremental step. For an interior node, the residual equation can be written for $(k+1)$ iteration of $i+1$ step as follows

$$\begin{aligned}\overset{\frown}{R}_{i+1}^{(k+1)} &= \overset{\frown}{R}_{i+1}^{(k)} + \frac{\partial}{\partial u_{i+1}^{(k)}}\left(\frac{\partial \sigma_{i+1}^{(k)}}{\partial x}\right)\Delta u_{i+1}^{(k+1)} \\ &= \overset{\frown}{R}_{i+1}^{(k)} + \frac{\partial}{\partial x}\left(\frac{\partial \sigma_{i+1}^{(k)}}{\partial \varepsilon_{i+1}^{(k)}}\frac{\partial \varepsilon_{i+1}^{(k)}}{\partial u_{i+1}^{(k)}}\right)\Delta u_{i+1}^{(k+1)}\end{aligned} \tag{17}$$

where $\Delta u_{i+1}^{(k+1)}$ is the displacement increment which should be found at the current iteration. $\overset{\frown}{R}_{i+1}^{(k)}(=\frac{\partial \sigma_{i+1}^{(k)}}{\partial x})$ is the residual for the divergence of stress at (k) iteration of $i+1$ step. $\frac{\partial \sigma_{i+1}^{(k)}}{\partial \varepsilon_{i+1}^{(k)}}$ denotes the elastoplastic tangent modulus. For multi-dimensional problem, the algorithmic tangent moduli are used to ensure the efficiency and accuracy in plastic computation [18]. Note that in Eq. (17), positions of the differential operators, ∂x and $\partial u_{i+1}^{(k)}$, are mutually exchanged; the operators are commutable after the linearization because the discrete equation might not be governed by the precise mathematical theory. $\sigma_{i+1}^{(k)}$ is calculated by the return mapping algorithm and kinematic variables are updated by backward Euler scheme as in Simo and Taylor [18].

When assuming $\overset{\frown}{R}_{i+1}^{(k+1)}(x) = 0$ and considering Eq. (12) or (16), Eq. (17) can be evaluated at x_J as follows

$$\overset{\frown}{R}_{i+1}^{(k)}(x_J) = -\left(Row^{J-th}\boldsymbol{\Psi}^T\right)\left(\mathbf{D}_{i+1}^{(k)}\boldsymbol{\Psi}\right)\Delta\mathbf{u}_{i+1}^{(k+1)} \tag{18}$$

When setting aside the treatment for the boundary equation, assembling the above residual equations for all interior nodes generates the system of equations and the system can be solved with respect to displacement increment giving the following expression:

$$\Delta\mathbf{u}_{i+1}^{(k+1)} = -\left(\boldsymbol{\Psi}^T\,\mathbf{D}_{i+1}^{(k)}\,\boldsymbol{\Psi}\right)^{-1}\mathbf{R}_{i+1}^{(k)} \tag{19}$$

where $\mathbf{R}_{i+1}^{(k)} = (\overset{\frown}{R}_{i+1}^{(k)}(x_1), \cdots, \overset{\frown}{R}_{i+1}^{(k)}(x_N))^T$ includes the nodal residual vectors. As a result, the elastoplastic tangent modulus was taken out in the form of $\mathbf{D}_{i+1}^{(k)}$ by introducing the double derivative approximation; the equilibrium equation, which originally involves the second order Partial Differential Equation (PDE) in conjunction with the material nonlinearity, was successfully discretized in the framework of strong formulation. This methodology can be readily extended to multi-dimension problems. The symmetricity of coefficient matrix in Eq. (19) is not guaranteed since for convenience, the residual equations are usually assembled by the node numbering order. Furthermore, to solve the material nonlinear problems without the high order derivative computation, the accuracy of the double derivative approximation and the stability of iterative procedure should be secured.

3. Multi-dimensional material nonlinear problem

3.1. Derivative approximation for PDM

The multi-index notion is used hereinafter for more convenient expression in multi-dimensional formulation. The $\boldsymbol{\alpha}$th power of n dimensional real vector, $\mathbf{x} \in \mathrm{R}^n$, is defined as follows

$$\mathbf{x}^{\boldsymbol{\alpha}} := x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} \tag{20}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$. The $\boldsymbol{\alpha}$th order derivative of a differentiable function, $f(\mathbf{x})$, with respect to $\mathbf{x}$ is given by

$$D_{\mathbf{x}}^{\boldsymbol{\alpha}} f(\mathbf{x}) := \frac{\partial^{|\boldsymbol{\alpha}|} f(\mathbf{x})}{\partial {x_1}^{\alpha_1} \cdots \partial {x_n}^{\alpha_n}} \tag{21}$$

where $D_{\mathbf{x}}^{\boldsymbol{\alpha}}$ denotes the $\boldsymbol{\alpha}$th order partial differential operator with respect to $\mathbf{x}$ and $|\boldsymbol{\alpha}| = \sum_{i=1}^{n} \alpha_i$. For example, in two dimensional case, $u(\mathbf{x})$ and $\frac{\partial u(\mathbf{x})}{\partial {x_2}^2}$ are expressed by $D_{\mathbf{x}}^{(0,0)}u(\mathbf{x})$ and $D_{\mathbf{x}}^{(0,2)}u(\mathbf{x})$, respectively.

Now, a given differentiable function, $u(\mathbf{x})(\in C^m(\Omega))$, is expressed by the multivariable Taylor theorem. The mth order Taylor polynomial is written by collecting the derivative coefficients of the given function combined with the polynomial basis as follows

$$u(\mathbf{x};\mathbf{y}) = \sum_{|\boldsymbol{\alpha}|\le m}\frac{(\mathbf{x}-\mathbf{y})^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}D_{\mathbf{x}}^{\boldsymbol{\alpha}}u(\mathbf{y}) = \mathbf{p}_m^T(\mathbf{x};\mathbf{y})\mathbf{a}(\mathbf{y}) \quad \text{on } \mathbf{y}\in\Omega \tag{22}$$

where $\boldsymbol{\alpha}! := \prod_{i=1}^{n}\alpha_i!$ is the factorial, $\mathbf{y}$ is a local center at which the Taylor polynomial is expanded and m denotes the order of consistency. The Taylor polynomial is composed of the complete m-th order polynomial vector, $\mathbf{p}$, and derivative coefficient vector, $\mathbf{a}$. The polynomial vector is given by

$$\mathbf{p}_m^T(\mathbf{x};\mathbf{y}) := \left(\frac{(\mathbf{x}-\mathbf{y})^{\boldsymbol{\alpha}_1}}{\boldsymbol{\alpha}_1!}, \cdots, \frac{(\mathbf{x}-\mathbf{y})^{\boldsymbol{\alpha}_K}}{\boldsymbol{\alpha}_K!}\right) \tag{23}$$

where $\boldsymbol{\alpha}_1 = (0, \cdots, 0)$ and $\boldsymbol{\alpha}_K = (0, \cdots, m)$. The derivative coefficient vector is written as

$$\mathbf{a}(\mathbf{y}) = \begin{pmatrix} D_{\mathbf{x}}^{\boldsymbol{\alpha}_1}u(\mathbf{y}) \\ \vdots \\ D_{\mathbf{x}}^{\boldsymbol{\alpha}_K}u(\mathbf{y}) \end{pmatrix}. \tag{24}$$

The lengths of $\mathbf{p}_m(\mathbf{x};\mathbf{y})$ and $\mathbf{a}(\mathbf{y})$ are $K = \frac{(n+m)!}{n!m!}$. Each component of $\mathbf{a}(\mathbf{y})$ indicates the derivative approximation for $u(\mathbf{x})$ evaluated at $\mathbf{y}$.

To get $\mathbf{a}(\mathbf{y})$, a weighted residual functional is constructed in the following manner:

$$J(\mathbf{a}) := \sum_{I=1}^{N} w\left(\frac{\mathbf{x}_I - \mathbf{y}}{\rho}\right)\left(\mathbf{p}_m^T(\mathbf{x}_I;\mathbf{y})\mathbf{a}(\mathbf{y}) - u_I\right)^2 \quad \text{on } \mathbf{y}\in\Omega \tag{25}$$

where $w(\frac{\mathbf{x}_I-\mathbf{y}}{\rho})$ is the weight function with a compact support; the size of compact support (or domain of influence) is determined by ρ. N is the number of nodes included in the support of weight function. Note that with the moving least squares procedure, the Taylor polynomial is valid within the range of compact support. The greatest accuracy is achieved when $\mathbf{x} = \mathbf{y}$. u_I is the nodal solution (for displacement) for neighbor node I. For problems involving material nonlinearity, nodal solutions for u_i are found through an iterative procedure. In this study, the variable, ρ, is used to identify the proper number of nodes that should participate in the Taylor polynomial, regardless of the node density or position, so that the resolution of derivative approximation is regularly preserved over all the position in numerical model. This is an advantageous feature for developing an adaptive modeling scheme. As identified by Yoon and Song [26], the computational efficiency and moment matrix invertibility are associated with the upper and lower bounds of the number of nodes included in the derivative approximation, respectively. In addition, the weight function might take an arbitrary shape as long as it has a peak at its center and smoothly vanishes since the derivative approximation no longer requires the differentiability of weight function.

The derivative coefficient vector is then obtained by minimizing the residual functional with respect to $\mathbf{a}(\mathbf{y})$ as follows

$$\begin{aligned}\mathbf{a}(\mathbf{x}) = &\sum_{I=1}^{N}\left(\mathbf{p}_m(\mathbf{x}_I;\mathbf{x})w\left(\frac{\mathbf{x}_I-\mathbf{x}}{\rho}\right)\mathbf{p}_m^T(\mathbf{x}_I;\mathbf{x})\right)^{-1} \\ &\cdot\left(w\left(\frac{\mathbf{x}_1-\mathbf{x}}{\rho}\right)\mathbf{p}_m(\mathbf{x}_1;\mathbf{x}), \cdots, w\left(\frac{\mathbf{x}_N-\mathbf{x}}{\rho}\right)\mathbf{p}_m(\mathbf{x}_N;\mathbf{x})\right)\cdot\begin{pmatrix}u_1\\ \vdots \\ u_N\end{pmatrix}\end{aligned} \tag{26}$$

where note that $\mathbf{y}$ is replaced by $\mathbf{x}$. It can be noticed that when calculating $\mathbf{a}(\mathbf{x})$, the derivative approximations up to the order of consistency are obtained without any actual differentiation for the approximation function $D_{\mathbf{x}}^{(0,0)}u(\mathbf{x})$. The derivative coefficient vector can be expressed in a matrix form as following

$$\begin{pmatrix} D_{\mathbf{x}}^{\boldsymbol{\alpha}_1}u(\mathbf{x}) \\ \vdots \\ D_{\mathbf{x}}^{\boldsymbol{\alpha}_L}u(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \Phi_1^{[\boldsymbol{\alpha}_1]}(\mathbf{x}), \cdots, \Phi_N^{[\boldsymbol{\alpha}_1]}(\mathbf{x}) \\ \vdots \quad \ddots \quad \vdots \\ \Phi_1^{[\boldsymbol{\alpha}_L]}(\mathbf{x}), \cdots, \Phi_N^{[\boldsymbol{\alpha}_L]}(\mathbf{x}) \end{pmatrix}\cdot\begin{pmatrix}u_1\\ \vdots \\ u_N\end{pmatrix} \tag{27}$$

where $\Phi_I^{[\boldsymbol{\alpha}]}(\mathbf{x})$ is the $\boldsymbol{\alpha}$th generalized shape function; more specifically, $\Phi_I^{[\boldsymbol{\alpha}]}(\mathbf{x})$ indicates the $\boldsymbol{\alpha}$th derivative approximation of the shape function for node I. It is the derivative approximation of $\Phi_I^{[\mathbf{0}]}(\mathbf{x})$ but is not the exact derivative of $\Phi_I^{[\mathbf{0}]}(\mathbf{x})$. In the right-hand side of Eq. (27), the $\boldsymbol{\alpha}$th row of the coefficient matrix embraces the $\boldsymbol{\alpha}$-th derivative approximations of shape function for the neighbor nodes. Here, a useful expression for the $\boldsymbol{\alpha}$-th derivative approximation can be written as follows

$$D_{\mathbf{x}}^{\boldsymbol{\alpha}} u(\mathbf{x}) := \sum_{I=1}^{N} \Phi_I^{[\boldsymbol{\alpha}]}(\mathbf{x}) u_I \tag{28}$$

where note that u_I is equivalent to $u(\mathbf{x}_I)$ within the context of the PDM. The little difference between u_I and $u(\mathbf{x}_I)$ is attributed to the absence of constraint formulation for essential boundary enforcement which the conventional weak form based meshfree methods cannot avoid. The derivative coefficient vector provides an excellent approximation for the derivative function of $u(\mathbf{x})$ and use of higher order polynomial vector is expected to provide the higher level of accuracy for derivative approximation. Yoon and Song [26] showed that the derivative approximation successfully approximates the derivative of the original function. Except for the associated error in solving the nonlinear system, most error results from the construction of derivative approximation and the mathematical validity of derivative approximation results from the consistency of the Taylor polynomial expanded by the moving least squares method. Note that this is not a new approach for many previous studies which use meshfree point collocation methods [8,9,25] employ the meshfree derivative computation without the actual differentiation for the approximation function.

3.2. Double derivative approximation

In the PDM, the α-th order derivative function of $u(x)$ can be directly approximated by the α-th order derivative approximation which includes the $\boldsymbol{\alpha}$-th generalized shape function combined with the nodal solution. However, as already shown in one dimensional case, the second order derivative approximation is decomposed into two first order derivative approximations such that the elastoplastic tangent modulus explicitly appears in the discretization process for the linear momentum equation which describes the material nonlinear problem. In this section, validity of the double derivative approximation is investigated; it will be shown in the next chapter that the derivative approximation is a successful substitute for the second order derivative approximation. Interestingly, when employing the double derivative approximation, the neighbor nodes of neighbor nodes of the original node participate in the discretization of equilibrium equation because each first order derivative approximation involves different neighbor node set. This overlap results in an extended domain of influence as compared to the conventional second order derivative approximation. This viewpoint may seem to be unusual from the perspective of FEM. However, it demonstrates the flexibility of PDM in constructing the derivative approximation without the constraint of a mesh or grid structure.

To better understand the key idea of double derivative approximation, recall the one dimensional second order derivative approximation which is written in a matrix form from Eq. (7) as follows

$$\begin{pmatrix} \sum_I \Phi_I^{[(2)]}(x_1) u_I \\ \vdots \\ \sum_I \Phi_I^{[(2)]}(x_N) u_I \end{pmatrix} = \begin{pmatrix} \Phi_1^{[(2)]}(x_1) & \cdots & \Phi_N^{[(2)]}(x_1) \\ \vdots & \ddots & \vdots \\ \Phi_1^{[(2)]}(x_N) & \cdots & \Phi_N^{[(2)]}(x_N) \end{pmatrix} \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix}. \tag{29}$$

However, the double derivative approximation generates a different form of the second order derivative approximation as following

$$\begin{aligned} \sum_I \Phi_I^{[(2)]}(x_J) u_I &\cong \sum_I \Phi_I^{[(1)]}(x_J) \left(\sum_K \Phi_K^{[(1)]}(x_I) u_K \right) \\ &= \sum_I \sum_K \Phi_I^{[(1)]}(x_J) \Phi_K^{[(1)]}(x_I) u_K \end{aligned} \tag{30}$$

where note that $u_I = \sum_{K=1}^{N} \Phi_K^{[1]}(\mathbf{x}_I) u_K$; u_I is approximated by the separate first order derivative approximation. Thus, additional summation for K comes out and the generalized shape functions for I and K are involved together in the double derivative approximation. In fact, as seen in Eq. (10) or (11), we already utilized this approach in discretizing the linear momentum equation. A vectorial assemblage of the double derivative approximations over all the nodes yields a following matrix expression:

$$\begin{aligned} &\begin{pmatrix} \sum_I \sum_K \Phi_I^{[(1)]}(x_1) \Phi_K^{[(1)]}(x_I) u_K \\ \vdots \\ \sum_I \sum_K \Phi_I^{[(1)]}(x_N) \Phi_K^{[(1)]}(x_I) u_K \end{pmatrix} \\ &= \begin{pmatrix} \Phi_1^{[(1)]}(x_1) & \cdots & \Phi_N^{[(1)]}(x_1) \\ \vdots & \ddots & \vdots \\ \Phi_1^{[(1)]}(x_N) & \cdots & \Phi_N^{[(1)]}(x_N) \end{pmatrix} \begin{pmatrix} \Phi_1^{[(1)]}(x_1) & \cdots & \Phi_N^{[(1)]}(x_1) \\ \vdots & \ddots & \vdots \\ \Phi_1^{[(1)]}(x_N) & \cdots & \Phi_N^{[(1)]}(x_N) \end{pmatrix} \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix}. \end{aligned} \tag{31}$$

The multi-dimensional version of the double derivative approximation for the second order, i.e., $|\boldsymbol{\alpha}| = 2$, $|\boldsymbol{\beta}| = |\boldsymbol{\alpha} - \boldsymbol{\beta}| = 1$, can be written as

$$\begin{aligned} D_{\mathbf{x}}^{\boldsymbol{\alpha}} u(\mathbf{x}_J) &= \sum_I \Phi_I^{[\boldsymbol{\beta}]}(\mathbf{x}_J) D_{\mathbf{x}}^{\boldsymbol{\alpha}-\boldsymbol{\beta}} u(\mathbf{x}_I) \\ &= \sum_I \Phi_I^{[\boldsymbol{\beta}]}(\mathbf{x}_J) \left(\sum_{K=1}^{N} \Phi_K^{[\boldsymbol{\alpha}-\boldsymbol{\beta}]}(\mathbf{x}_I) u(\mathbf{x}_K) \right) \\ &= \sum_I \sum_K \Phi_I^{[\boldsymbol{\beta}]}(\mathbf{x}_J) \Phi_K^{[\boldsymbol{\alpha}-\boldsymbol{\beta}]}(\mathbf{x}_I) u_K \end{aligned} \tag{32}$$

where it is noted that $D_{\mathbf{x}}^{\boldsymbol{\alpha}-\boldsymbol{\beta}} u(\mathbf{x}_I)$ is replaced by the derivative approximation, $\sum_{K=1}^{N} \Phi_K^{[\boldsymbol{\alpha}-\boldsymbol{\beta}]}(\mathbf{x}_I) u(\mathbf{x}_K)$. In two-dimensional case, if $\boldsymbol{\alpha} = (2, 0)$ then $\boldsymbol{\beta} = \boldsymbol{\alpha} - \boldsymbol{\beta} = (1, 0)$ where $|\boldsymbol{\beta}| = |\boldsymbol{\alpha} - \boldsymbol{\beta}| = 1$ and $|\boldsymbol{\alpha}| = 2$. Similarly, if $\boldsymbol{\alpha} = (1, 1)$ then $\boldsymbol{\beta} = (1, 0)$ and $\boldsymbol{\alpha} - \boldsymbol{\beta} = (0, 1)$ or vice versa. Therefore, the second derivative functions of $u(\mathbf{x})$ can be computed by

$$D_{\mathbf{x}}^{(2,0)} u(\mathbf{x}_J) = \sum_I \sum_K \Phi_I^{[(1,0)]}(\mathbf{x}_J) \Phi_K^{[(1,0)]}(\mathbf{x}_I) u_K \tag{33}$$

$$D_{\mathbf{x}}^{(0,2)} u(\mathbf{x}_J) = \sum_I \sum_K \Phi_I^{[(0,1)]}(\mathbf{x}_J) \Phi_K^{[(0,1)]}(\mathbf{x}_I) u_K \tag{34}$$

$$D_{\mathbf{x}}^{(1,1)} u(\mathbf{x}_J) = \sum_I \sum_K \Phi_I^{[(1,0)]}(\mathbf{x}_J) \Phi_K^{[(0,1)]}(\mathbf{x}_I) u_K \tag{35}$$

where recall that $D_{\mathbf{x}}^{(2,0)} u(\mathbf{x}) = \frac{\partial^2 u(\mathbf{x})}{\partial^2 x_1}$, $D_{\mathbf{x}}^{(0,2)} u(\mathbf{x}) = \frac{\partial^2 u(\mathbf{x})}{\partial^2 x_2}$, and $D_{\mathbf{x}}^{(1,1)} u(\mathbf{x}) = \frac{\partial^2 u(\mathbf{x})}{\partial x_1 \partial x_2}$. Also note that u_K's are the solutions not for the neighbor nodes of $\mathbf{x}_J$ but for the neighbor nodes of $\mathbf{x}_I$. This illustrates how the neighboring nodes of neighbor nodes of the original node are involved in the double derivative approximation. However, when applying the conventional derivative approximation, the second order derivative approximations are given by

$$D_{\mathbf{x}}^{(2,0)} u(\mathbf{x}_J) = \sum_K \Phi_K^{[(2,0)]}(\mathbf{x}_J) u_K \tag{36}$$

$$D_{\mathbf{x}}^{(0,2)} u(\mathbf{x}_J) = \sum_K \Phi_K^{[(0,2)]}(\mathbf{x}_J) u_K \tag{37}$$

$$D_{\mathbf{x}}^{(1,1)} u(\mathbf{x}_J) = \sum_K \Phi_K^{[(1,1)]}(\mathbf{x}_J) u_K \tag{38}$$

where u_K's are the nodal solutions for the neighbor nodes of $\mathbf{x}_J$. Fig. 1(a) and (b) are schematics of domains of influence for the conventional derivative approximation and the double derivative approximation for the second order derivative computation, respectively. As seen in the figure, only nine nodes are included in the conventional derivative approximation while twenty-five nodes are included in the double derivative approximation.

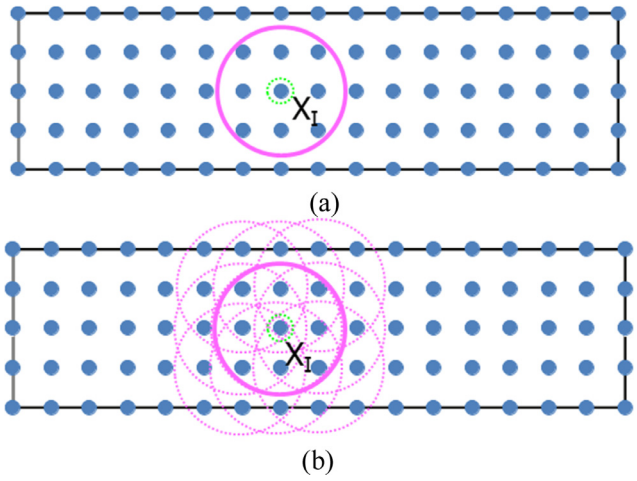

Fig. 1. Schematics of domains of influence for (a) the conventional derivative approximation and (b) the double derivative approximation.

Note that the differential operator, $D_{\mathbf{x}}^{\boldsymbol{\alpha}}$, can be decomposed into $D_{\mathbf{x}}^{\boldsymbol{\beta}}$ and $D_{\mathbf{x}}^{\boldsymbol{\alpha}-\boldsymbol{\beta}}$ through the double derivative approximation, as will be shown in the next section that the decomposition enables the PDM to use the first order derivative approximation in discretizing the second order PDE. This feature is quite interesting because the strong formulation inherently involves the equivalent differentiation order to the original PDE while the weak formulation beneficially utilizes the reduced order of differentiation by the introduction of the integral equation. It shows that the differential operator can be decomposed into the lower order differential operators in the numerical scheme although more precise mathematical analysis is left for future study. Furthermore, this approach might provide valuable insight to the governing equation in the strong form because the order of derivative approximation used to describe the original PDE, can be effectively reduced without the use of weak formulation.

3.3. Strongly formulated iterative procedure for plane stress problem

The divergence of stress tensor expresses the equilibrium equation for the multi-dimensional case. The double derivative approximation is applied in both divergence and stress computations. By doing so, the elastoplastic tangent modulus explicitly appears in the discrete form of the equation and is manipulated in the strong formulation for material nonlinear problem. Newton's method requires linearization of the equilibrium equation. Absent of the body force, the equilibrium equation can be linearized with respect to $\Delta\mathbf{u}$ as follows

$$\frac{\partial}{\partial \mathbf{u}}\left(\frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{x}}\right)\Delta\mathbf{u} = \frac{\partial}{\partial \mathbf{x}}\left(\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}}\frac{\partial \boldsymbol{\varepsilon}}{\partial \mathbf{u}}\right)\Delta\mathbf{u}. \tag{39}$$

In Eq. (39), the partial differentiations with respect to $\mathbf{u}$ and $\mathbf{x}$ are involved; note that as seen in one-dimensional formulation in Section 2.2, positions of the differentiations are exchangeable, which is crucial process to extract the elastoplastic tangent modulus from the linear momentum equation. Although this is not always applicable in the canonical mathematical derivation, the exchange of differentiation order makes no difference in the discrete form of equilibrium equation. In fact, the exchange simply changes the order of matrices containing the derivative approximations. Also, it will be shown that discretizations for $\frac{\partial}{\partial \mathbf{x}}$ and $\frac{\partial \boldsymbol{\varepsilon}}{\partial \mathbf{u}}$ yield the matrices which resemble the so-called 'B' matrix in the FEM formulation.

To set up the residual equation for the iterative procedure, Eq. (39) is linearized at $\mathbf{x}_J$ as following

$$\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k+1)}(\mathbf{x}_J) = \overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_J) + \left[\frac{\partial}{\partial \mathbf{x}}\left(\frac{\partial \boldsymbol{\sigma}_{i+1}^{(k)}}{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}\frac{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}{\partial \mathbf{u}_{i+1}^{(k)}}\right)\Delta\widehat{\mathbf{u}}_{i+1}^{(k+1)}\right]_{\mathbf{x}=\mathbf{x}_J} \tag{40}$$

where $\Delta\overset{\sim}{\mathbf{u}}{}_{i+1}^{(k+1)}(\mathbf{x}_J)$ is an increment of nodal displacement at $(k+1)$ iteration of $i+1$ step; '$\widehat{\bullet}$' indicates that the value results from computation at a node. $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k+1)}(\mathbf{x}_J)$ and $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_J)$ are the residual vectors for the divergence of stress at $(k+1)$ and (k) iterations, respectively. $\frac{\partial \boldsymbol{\sigma}_{i+1}^{(k)}}{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}$ denotes the elastoplastic tangent moduli. To calculate $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x})$, $\boldsymbol{\sigma}_{i+1}^{(k)}(\mathbf{x}_J)$ is evaluated using the return mapping algorithm [18] which consists of the elastic prediction and plastic correction. Other kinematic variables are updated by the backward Euler method.

Now the second term in the right hand side of Eq. (40) can be discretized by introducing the double derivative approximation as follows

$$\left[\frac{\partial}{\partial \mathbf{x}}\left(\frac{\partial \boldsymbol{\sigma}_{i+1}^{(k)}}{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}\frac{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}{\partial \mathbf{u}_{i+1}^{(k)}}\right)\Delta\overset{\sim}{\mathbf{u}}{}_{i+1}^{(k+1)}\right]_{\mathbf{x}=\mathbf{x}_J} = \left(Row^{J-th}\boldsymbol{\Psi}^T\right)\mathbf{D}_{i+1}^{(k)}\boldsymbol{\Psi}\,\Delta\mathbf{u}_{i+1}^{(k+1)} \tag{41}$$

where $\Delta\mathbf{u}_{i+1}^{(k+1)}$ is the global vector containing displacement increments for all nodes, which is defined by

$$\Delta\mathbf{u}_{i+1}^{(k+1)} = \begin{pmatrix} \Delta u_{1,i+1}^{(k+1)}(\mathbf{x}_1) \\ \Delta u_{2,i+1}^{(k+1)}(\mathbf{x}_1) \\ \vdots \\ \Delta u_{1,i+1}^{(k+1)}(\mathbf{x}_N) \\ \Delta u_{2,i+1}^{(k+1)}(\mathbf{x}_N) \end{pmatrix} \tag{42}$$

Note that $\boldsymbol{\Psi}$ does not have superscript or subscript because material nonlinearity without consideration of geometric nonlinearity makes no change in the generalized shape functions as the incremental step advances. When $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k+1)}(\mathbf{x}) = 0$ is assumed at each incremental step, assembling $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_I)$ yields the following system of equations:

$$-\left(\boldsymbol{\Psi}^T\mathbf{D}_{i+1}^{(k)}\boldsymbol{\Psi}\right)\Delta\mathbf{u}_{i+1}^{(k+1)} = \mathbf{R}_{i+1}^{\Omega,(k)} \tag{43}$$

where the symmetry of the system is not guaranteed. Note the similarity between coefficient matrix in Eq. (43) and the stiffness matrix that appears in the weak formulation. Solving Eq. (43) for $\Delta\mathbf{u}_{i+1}^{(k+1)}$ generates the following discrete equation:

$$\Delta\mathbf{u}_{i+1}^{(k+1)} = -\left(\boldsymbol{\Psi}^T\mathbf{D}_{i+1}^{(k)}\boldsymbol{\Psi}\right)^{-1}\mathbf{R}_{i+1}^{\Omega,(k)} \tag{44}$$

Eq. (44) is analogous with the one-dimensional system in Eq. (19). In addition, to construct the total system of equations, the residual equations for the natural and essential boundary conditions should be applied to Eq. (44). In Eq. (44), $\mathbf{R}_{i+1}^{\Omega,(k)}$ is the global residual vector which collects $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_I)$ for all interior nodes as follows

$$\mathbf{R}_{i+1}^{\Omega,(k)} = \begin{pmatrix} \overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_1) \\ \vdots \\ \overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_N) \end{pmatrix} \tag{45}$$

where $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_J)$ occupies the (2J-1)-th and (2J)-th slots of $\mathbf{R}_{i+1}^{\Omega,(k)}$. Since $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_I) = \nabla\cdot\boldsymbol{\sigma}_{i+1}^{(k)}(\mathbf{x}_I)$, $\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_I)$ is obtained by

$$\overset{\sim}{\mathbf{R}}{}_{i+1}^{\Omega,(k)}(\mathbf{x}_I) = \sum_J \boldsymbol{\Phi}_{IJ}^T\cdot\overset{\sim}{\boldsymbol{\sigma}}{}_{i+1}^{(k)}(\mathbf{x}_J) \tag{46}$$

where

$$\boldsymbol{\Phi}_{IJ}^T = \begin{pmatrix} \Phi_J^{[(1,0)]}(\mathbf{x}_I) & 0 & \Phi_J^{[(0,1)]}(\mathbf{x}_I) \\ 0 & \Phi_J^{[(0,1)]}(\mathbf{x}_I) & \Phi_J^{[(1,0)]}(\mathbf{x}_I) \end{pmatrix} \tag{47}$$

$$\widehat{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_J) = \begin{pmatrix} \sigma_{11,i+1}^{(k)}(\mathbf{x}_J) \\ \sigma_{22,i+1}^{(k)}(\mathbf{x}_J) \\ \sigma_{12,i+1}^{(k)}(\mathbf{x}_J) \end{pmatrix} \tag{48}$$

In Eq. (47), the subscript I of $\boldsymbol{\Phi}_{IJ}^T$ is related to the position ($\mathbf{x}_I$) at which $\boldsymbol{\Phi}_{IJ}^T$ is constructed while the subscript J implies the neighbor nodes of $\mathbf{x}_I$. It is noticed that $\boldsymbol{\Phi}_{IJ}^T$ is quite similar with the transpose of 'B' matrix in the conventional FEM formulation.$\sigma_{ij,i+1}^{(k)}(\mathbf{x}_I)$ is computed with the aforementioned return mapping algorithm [18]. Then $\boldsymbol{\Psi}$ is composed of $\boldsymbol{\Phi}_{IJ}$ as following

$$\boldsymbol{\Psi} = \begin{pmatrix} \boldsymbol{\Phi}_{11} & \cdots & \boldsymbol{\Phi}_{1N} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\Phi}_{N1} & \cdots & \boldsymbol{\Phi}_{NN} \end{pmatrix} \tag{49}$$

and $\boldsymbol{\Psi}^T$ is given by

$$\boldsymbol{\Psi}^T = \begin{pmatrix} \boldsymbol{\Phi}_{11}^T & \cdots & \boldsymbol{\Phi}_{1N}^T \\ \vdots & \ddots & \vdots \\ \boldsymbol{\Phi}_{N1}^T & \cdots & \boldsymbol{\Phi}_{NN}^T \end{pmatrix} \tag{50}$$

$\boldsymbol{\Psi}^T$ is the discrete differential operator for computing the divergence of stress while $\boldsymbol{\Psi}$ is associated with the strain calculation; i.e., $\frac{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}{\partial \mathbf{u}_{i+1}^{(k)}}$ in Eq. (40) or (41). $\boldsymbol{\Psi}^T$ is not the transpose of $\boldsymbol{\Psi}$; instead, its components, $\boldsymbol{\Phi}_{IJ}$'s, are transposed.

The material coefficient matrix ($\mathbf{D}_{i+1}^{(k)}$) is given in a diagonal matrix form as follows

$$\mathbf{D}_{i+1}^{(k)} = \begin{pmatrix} \widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_1) & & \\ 0 & \ddots & 0 \\ & & \widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_N) \end{pmatrix} \tag{51}$$

Note that $\mathbf{D}_{i+1}^{(k)}$ is not the differential operator $D_{\mathbf{x}}^{\boldsymbol{\alpha}}$ defined in Eq. (21). For the two-dimensional case, the tangent modulus at $\mathbf{x}_I$ is given in a matrix form as following

$$\widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I) = \begin{pmatrix} D_{11,i+1}^{(k)}(\mathbf{x}_I) & D_{12,i+1}^{(k)}(\mathbf{x}_I) & D_{13,i+1}^{(k)}(\mathbf{x}_I) \\ D_{21,i+1}^{(k)}(\mathbf{x}_I) & D_{22,i+1}^{(k)}(\mathbf{x}_I) & D_{23,i+1}^{(k)}(\mathbf{x}_I) \\ D_{31,i+1}^{(k)}(\mathbf{x}_I) & D_{32,i+1}^{(k)}(\mathbf{x}_I) & D_{33,i+1}^{(k)}(\mathbf{x}_I) \end{pmatrix}. \tag{52}$$

The components of nodal stress vector in Eq. (48) are obtained using the nodal strain which is computed by

$$\widehat{\boldsymbol{\varepsilon}}_{i+1}^{(k)}(\mathbf{x}_I) = \sum_J \boldsymbol{\Phi}_{IJ} \cdot \left(\widehat{\mathbf{u}}_i(\mathbf{x}_J) + \delta \widehat{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_J) \right) \tag{53}$$

where $\widehat{\mathbf{u}}_i(\mathbf{x}_I)$ is the converged displacement vector at the previous step. $\delta \widehat{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_J)$ is the accumulated displacement increment during the iterations at current step and is expressed by $\delta \widehat{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_J) = \sum_{l=1}^{k} \Delta \widehat{\mathbf{u}}_{i+1}^{(l)}(\mathbf{x}_J)$. In the iterative procedure, the nodal stress, $\widehat{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_J)$, is then updated at every iteration as follows

$$\begin{aligned} \widehat{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I) &= \widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I) \widehat{\boldsymbol{\varepsilon}}_{i+1}^{(k)}(\mathbf{x}_I) \\ &= \widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I) \sum_J \boldsymbol{\Phi}_{IJ} \cdot \left(\widehat{\mathbf{u}}_i(\mathbf{x}_J) + \delta \widehat{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_J) \right) \end{aligned} \tag{54}$$

It is worth noting that $Row^{J-th}\,\boldsymbol{\Psi}^T$ in Eq. (41) can be rewritten in terms of $\boldsymbol{\Phi}_{IJ}^T$ as following

$$\left(Row^{J-th}\,\boldsymbol{\Psi}^T \right) = \left(\boldsymbol{\Phi}_{J1}^T, \cdots, \boldsymbol{\Phi}_{JN}^T \right) \tag{55}$$

where Row^{J-th} indicates the row vector involving node J at which the generalized shape function is constructed; in two-dimensional case, it involves (*2J-1*)-th and (*2J*)-th rows.

Taken note that in the FEM, the equilibrium equation is assembled after it is evaluated on an element level where the integrand is calculated at integration points by way of numerical quadrature. However, in the PDM, the equilibrium equation is constructed at the node. From Eq. (41), the nodal discrete equilibrium equation at $\mathbf{x}_I$ is written by

$$\left[\frac{\partial}{\partial \mathbf{x}} \left(\frac{\partial \boldsymbol{\sigma}_{i+1}^{(k)}}{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}} \frac{\partial \boldsymbol{\varepsilon}_{i+1}^{(k)}}{\partial \mathbf{u}_{i+1}^{(k)}} \right) \right]_{\mathbf{x}=\mathbf{x}_I} = \sum_J \boldsymbol{\Phi}_{IJ}^T \widehat{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I) \boldsymbol{\Phi}_{IJ} \tag{56}$$

where it is seen that in the discretization process, $\frac{\partial}{\partial \mathbf{x}}$ and $\frac{\partial \boldsymbol{\varepsilon}}{\partial \mathbf{u}}$ are transformed into $\boldsymbol{\Phi}_{IJ}^T$ and $\boldsymbol{\Phi}_{IJ}$, respectively. Also, the discrete equilibrium equation employs a single summation for neighbor node J while in particular, the discrete equation of weak formulation requires a double summation for I and J during the numerical quadrature for each quadrature point.

The plane stress condition is considered in this study where interior nodal stresses for each iterative step are updated with the return mapping algorithm [18]. Compared to the FEM for material nonlinear problem, the PDM provokes no difference in the numerical implementation for plane stress case. Thus, this study strictly follows the numerical implementation procedure of the algorithm proposed by Simo and Hughes [19]. The increment of consistency parameter($\Delta\gamma$) is determined during the stress update procedure. It depends on the incremental step size and is determined from a discrete consistency condition, $g(\Delta\gamma) = 0$. It is solved by a separated Newton iterative procedure and this local iterative procedure takes a very short computation time since it provokes very simple arithmetic calculation; see, Simo and Hughes [19] for further details.

3.4. Total system of equations

As mentioned previously, elemental equilibrium equations generated via the FEM are often integrated numerically with quadrature rules during which, a mapping from the local coordinate to the global coordinate usually occurs. However, the PDM operates on the strong form of equilibrium; integration and mapping is not required and the computations are carried out in a node-wise manner. The PDM total system of equations can be constructed node by node from the start of the discretization process. Governing equations for interior nodes, traction boundary nodes, and essential boundary nodes, are linearized to yield the residual equations and the total system is constructed by assembling all the residual equations to be solved by the iterative procedure based on Newton's method. During construction of the total system of equations, assemblage of the residual equations follows the node numbering order. This aspect presents convenience from a programming perspective since the residual equations need not be classified according to the region to which the relevant node belongs. The process does not generate a symmetric system, and it should be noted that strong form methods like PDM inherently yield the non-symmetric sparse system [8,9,25–27].

At the boundary, Eqs. (4) and (5) are used to construct residual equations. When $\mathbf{x}_I$ belongs to Γ_t, the natural boundary condition in Eq. (4) is discretized to yield the residual equation as follows

$$\widehat{\mathbf{R}}_{i+1}^{\Gamma_t,(k+1)}(\mathbf{x}_I) = \widehat{\mathbf{R}}_{i+1}^{\Gamma_t,(k)}(\mathbf{x}_I) + \frac{\partial}{\partial \mathbf{u}_{i+1}^{(k)}} \left(\mathbf{n}^T(\mathbf{x}_I) \cdot \widehat{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I) \right) \Delta \widehat{\mathbf{u}}_{i+1}^{(k+1)}(\mathbf{x}_I) \quad \text{on } \Gamma_t \tag{57}$$

where $\mathbf{n}^T(\mathbf{x}_I)$ is the transpose of a matrix representing the unit normal vector at $\mathbf{x}_I$ and is written by

$$\mathbf{n}^T(\mathbf{x}_I) = \begin{pmatrix} n_1(\mathbf{x}_I) & 0 & n_2(\mathbf{x}_I) \\ 0 & n_2(\mathbf{x}_I) & n_1(\mathbf{x}_I) \end{pmatrix}. \tag{58}$$

The residual vector at the previous iteration is defined as follows

$$\widehat{\mathbf{R}}_{i+1}^{\Gamma_t,(k)}(\mathbf{x}_I) = \mathbf{n}^T(\mathbf{x}_I) \cdot \widehat{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I) - \bar{\mathbf{t}}(\mathbf{x}_I) \tag{59}$$

See Eq. (54) for an explanation of $\overset{\frown}{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I)$. $\mathbf{n}^T(\mathbf{x}_I)\cdot\overset{\frown}{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I)$ in Eq. (57) is discretized by

$$\mathbf{n}^T(\mathbf{x}_I)\cdot\overset{\frown}{\boldsymbol{\sigma}}_{i+1}^{(k)}(\mathbf{x}_I)=\mathbf{n}^T(\mathbf{x}_I)\cdot\overset{\frown}{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I)\overset{\frown}{\boldsymbol{\varepsilon}}_{i+1}^{(k)}(\mathbf{x}_I) \tag{60}$$

where $\overset{\frown}{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I)$ and $\overset{\frown}{\boldsymbol{\varepsilon}}_{i+1}^{(k)}(\mathbf{x}_I)$ are defined in Eqs. (52) and (53), respectively. To complete the construction of Eq. (57), the second term in the right hand side of Eq. (57) is then discretized as follows

$$\left[\frac{\partial}{\partial\mathbf{u}_{i+1}^{(k)}}\left(\mathbf{n}^T\cdot\overset{\frown}{\boldsymbol{\sigma}}_{i+1}^{(k)}\right)\right]_{\mathbf{x}=\mathbf{x}_I}=\sum_J\mathbf{n}^T(\mathbf{x}_I)\cdot\overset{\frown}{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I)\boldsymbol{\Phi}_{IJ} \tag{61}$$

On the other hand, when $\mathbf{x}_I$ belongs to Γ_u, the residual equation for the essential boundary condition in Eq. (5) is given as

$$\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k+1)}(\mathbf{x}_I)=\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k)}(\mathbf{x}_I)+\frac{\partial\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_I)}{\partial\mathbf{u}_{i+1}^{(k)}}\Delta\overset{\frown}{\mathbf{u}}_{i+1}^{(k+1)}(\mathbf{x}_I)\quad\text{on }\Gamma_u \tag{62}$$

where the residual vector for the essential boundary node is written by

$$\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k)}(\mathbf{x}_I)=\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_I)-\bar{\mathbf{u}}(\mathbf{x}_I). \tag{63}$$

The displacement vector, $\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_I)$, needs to be updated at every iteration and is given by

$$\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_I)=\sum_J\Phi_J^{[(0,0)]}(\mathbf{x}_I)\cdot\left(\widehat{\mathbf{u}}_i(\mathbf{x}_J)+\delta\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}(\mathbf{x}_J)\right) \tag{64}$$

The computation of Eq. (62) can be finished by discretizing the second term in the right hand side of Eq. (62) as follows

$$\left[\frac{\partial\overset{\frown}{\mathbf{u}}_{i+1}^{(k)}}{\partial\mathbf{u}_{i+1}^{(k)}}\right]_{\mathbf{x}=\mathbf{x}_I}=\sum_J\begin{pmatrix}\Phi_J^{[(0,0)]}(\mathbf{x}_I) & 0\\ 0 & \Phi_J^{[(0,0)]}(\mathbf{x}_I)\end{pmatrix} \tag{65}$$

where it is noticed that the second term in the right hand side of Eq. (62) consists of the zeroth order generalized shape function, $\Phi_J^{[(0,0)]}(\mathbf{x}_I)$.

Similar to the interior node case shown in Eq. (45), the global residual vectors $\mathbf{R}_{i+1}^{\Gamma_t,(k)}$ and $\mathbf{R}_{i+1}^{\Gamma_u,(k)}$ in Eqs. (57) and (62) are constructed by assembling $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_t,(k)}(\mathbf{x}_I)$ and $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k)}(\mathbf{x}_I)$ for natural and essential boundary nodes, respectively. Then the total residual vector($\mathbf{R}_{i+1}^{(k)}$) is built up by assembling $\overset{\frown}{\mathbf{R}}_{i+1}^{\Omega,(k)}(\mathbf{x}_I)$, $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_t,(k)}(\mathbf{x}_I)$ and $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k)}(\mathbf{x}_I)$ when $\overset{\frown}{\mathbf{R}}_{i+1}^{\Omega,(k+1)}(\mathbf{x}_I)=0$, $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_t,(k+1)}(\mathbf{x}_I)=0$ and $\overset{\frown}{\mathbf{R}}_{i+1}^{\Gamma_u,(k+1)}(\mathbf{x}_I)=0$ are assumed. After the total system of equations for i + 1(k + 1 iteration) step are constructed, the displacement increment, $\Delta\mathbf{u}_{i+1}^{(k+1)}$, is found by solving the following total system of equations:

$$\mathbf{K}_{i+1}^{(k)}\cdot\Delta\mathbf{u}_{i+1}^{(k+1)}=\mathbf{R}_{i+1}^{(k)} \tag{66}$$

Note that the assemblage of residual equations is conducted in accordance with the order of node numbering in order to achieve convenience in a computer programming. However, as seen in Fig. 2, to arrange the discrete equations in the assembling process, all nodes still need be classified according to the region to which they belong. In this scenario, the row directional components of $\mathbf{K}_{i+1}^{(k)}$ are supposed to be arranged by the order of node numbering but the order of residual equation does not affect the solution process as long as they are independent of each other. Here, for convenience's sake, the stiffness matrix and residual vector are expressed according to the node classification as following

$$\mathbf{K}_{i+1}^{(k)}=\begin{pmatrix}\mathbf{K}_{i+1}^{\Omega,(k)}\\ \mathbf{K}_{i+1}^{\Gamma_t,(k)}\\ \mathbf{K}_{i+1}^{\Gamma_u,(k)}\end{pmatrix} \tag{67}$$

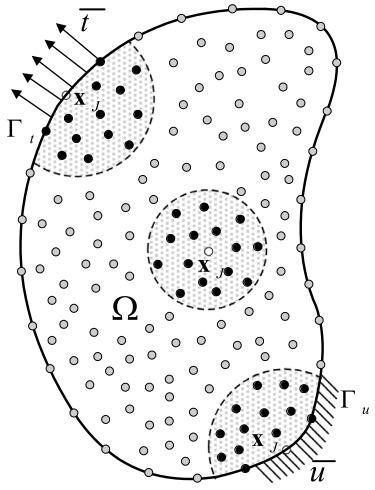

Fig. 2. Classification of nodes for constructing total algebraic system.

and

$$\mathbf{R}_{i+1}^{(k)}=\begin{pmatrix}\mathbf{R}_{i+1}^{\Omega,(k)}\\ \mathbf{R}_{i+1}^{\Gamma_t,(k)}\\ \mathbf{R}_{i+1}^{\Gamma_u,(k)}\end{pmatrix} \tag{68}$$

where the components of Eq. (67) can be found in Eqs. (56), (61), and (65), respectively; more specifically, they can be calculated as follows

$$\mathbf{K}_{IJ,i+1}^{\Omega,(k)}=\sum_J\boldsymbol{\Phi}_{IJ}^T\overset{\frown}{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I)\boldsymbol{\Phi}_{IJ} \tag{69}$$

$$\mathbf{K}_{IJ,i+1}^{\Gamma_t,(k)}=\sum_J\mathbf{n}^T(\mathbf{x}_I)\cdot\overset{\frown}{\mathbf{D}}_{i+1}^{(k)}(\mathbf{x}_I)\boldsymbol{\Phi}_{IJ} \tag{70}$$

$$\mathbf{K}_{IJ,i+1}^{\Gamma_u,(k)}=\sum_J\begin{pmatrix}\Phi_J^{[(0,0)]}(\mathbf{x}_I) & 0\\ 0 & \Phi_J^{[(0,0)]}(\mathbf{x}_I)\end{pmatrix} \tag{71}$$

where $\mathbf{K}_{IJ,i+1}^{\Omega,(k)}$ indicates the J-th component of the I-th row vector of $\mathbf{K}_{i+1}^{\Omega,(k)}$. As a result, the components of total system are allocated for the corresponding address according to the node number and directional component. For instance, the (2I-1)-th and (2I)-th components of $\mathbf{R}_{i+1}^{(k)}$ have the components for the residual equation corresponding to the node I.

From the Eq. (66), the displacement increment for the current iteration can be computed by

$$\Delta\mathbf{u}_{i+1}^{(k+1)}=-\left(\mathbf{K}_{i+1}^{(k)}\right)^{-1}\cdot\mathbf{R}_{i+1}^{(k)} \tag{72}$$

where the size of stiffness matrix is $(n\times N_t)$ by $(n\times N_t)$ and the sizes of residual vector and displacement vector are $(n\times N_t)$ by 1; N_t is the total number of nodes and n is the space dimension. For each step during the iterative procedure, the stiffness matrix is repeatedly computed until convergence criterion is satisfied while the global residual vector is calculated once. Once satisfied, through solving Eq. (72), the displacement is updated and the loading process advances to the next step. The overall procedure for iterative computation is analogous to that of the nonlinear finite element method.

4. Numerical examples

4.1. Verification of the double derivative approximation for one dimensional problem

In this section, mathematical feasibility of the double derivative approximation is numerically verified against the analytical solution of one dimensional elastic rod problem. Details of the double derivative

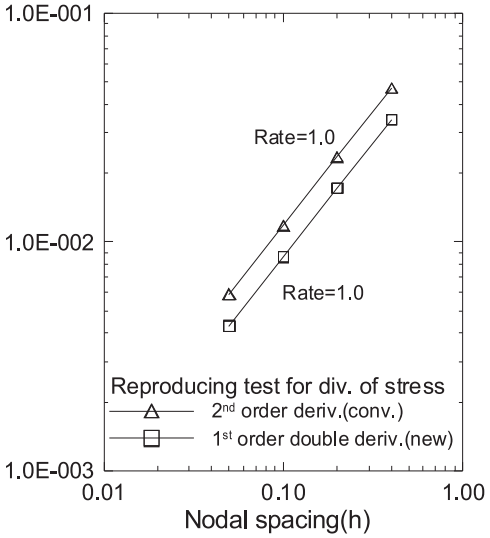

Fig. 3. Reproducing test results for the divergence of stress computation using the double derivative approximation and the conventional second order derivative approximation.

approximation can be found in Section 3.2 of this paper. Solutions to the posed problem, are computed and compared with the conventional second order derivative approximation and the double derivative approximation. A rod problem described by the displacement field in Eq. (73) with the divergence of stress given in Eq. (74) is considered:

$$u(x) = \frac{1}{E}\left(\frac{a_1}{6}x^3 + a_2 x + a_3\right) \tag{73}$$

$$\frac{\partial \sigma(x)}{\partial x} = a_1 x \tag{74}$$

In the above, E, a_1, a_2, and a_3 are Young's modulus and arbitrary constants, respectively. The conventional second order derivative approximation and double derivative approximation, the divergence of stress is computed by using

$$\sigma_{x,x}(x) = \sum_I E\,\Phi_I^{[2]}(x_J)u_I \quad \text{and} \tag{75}$$

$$\sigma_{x,x}(x) = \sum_I \sum_K \Phi_I^{[1]}(x_J) E\,\Phi_K^{[1]}(x_I)u_K \tag{76}$$

To numerically investigate the validity of Eq. (30), the accuracies of the derivative approximations are evaluated through the reproducing property for the divergence of stress. The following measurement is introduced to ascertain the error in the reproducing process based on L^∞:

$$E_{L_\sigma^\infty} = \frac{Sup_{I\in\Lambda}\left\|\sigma_{x,x}^{nu}(\mathbf{x}_I) - \sigma_{x,x}^{ex}(\mathbf{x}_I)\right\|}{Sup_{I\in\Lambda}\left\|\sigma_{x,x}^{ex}(\mathbf{x}_I)\right\|} \tag{77}$$

where $\sigma_{x,x}^{ex}(\mathbf{x}_I)$ and $\sigma_{x,x}^{nu}(\mathbf{x}_I)$ are exact and numerical computations for the divergence of stress, and Λ denotes a set for all nodes. Since the closed-form solution is already given in Eqs. (73) and (74), the relative error in the reproduced divergence of stress can be measured. The reader is referred to Lee and Yoon [9] and Yoon and Song [27] for further details on the reproducing test.

The test results are shown in Fig. 3 below. The double derivative approximation, Eq. (76), and the conventional second order derivative approximation, Eq. (75), yield almost same convergence rates. Both derivative approximations yield the first order convergence rate as expected by numerical theory. Here, the quadratic polynomial vector is adopted for the double derivative and the conventional second order derivative approximations. The magnitude of relative error generated with the

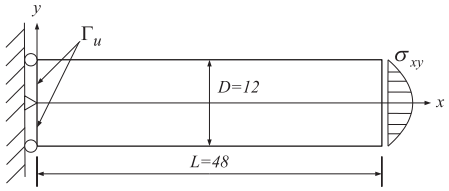

Fig. 4. Schematic drawing of an elastic cantilever beam.

double derivative approximation is slightly smaller than the conventional derivative approximation; it implies that the double derivative approximation is an adequate substitute for the conventional second order derivative approximation in the PDM formulation for nonlinear material analysis. Similar performance in multi-dimensional problems is observed and verifications for two dimensional problems are presented in the following sections.

4.2. Convergence study for elastic beam problem

In this section, a convergence study for two-dimensional elastic cantilever beam problem is performed to investigate the robustness of double derivative approximation. Unlike the reproducing test performed in the previous section, the accuracy of numerical solution obtained by solving the PDE is examined here. First, the PDM equipped with the double derivative approximation is exploited for the linear elastic case followed by a study on the nonlinear case. Analytical solutions of a linear elastic cantilever beam subjected to an end load as shown in Fig. 4 are given by Timoshenko and Goodier [21] as follows

$$u_1 = \frac{-Px_2}{6EI}\left[(6L - 3x_1)x_1 + (2+\nu)\left({x_2}^2 - \frac{D^2}{4}\right)\right] \tag{78}$$

$$u_2 = \frac{P}{6EI}\left[3\nu {x_2}^2(L - x_1) + (4+5\nu)\frac{D^2 x_1}{4} + (3L - x_1){x_1}^2\right] \tag{79}$$

The closed-form solution for stresses are given by

$$\sigma_{11} = -\frac{P(L - x_1)x_2}{I} \tag{80}$$

$$\sigma_{22} = 0 \tag{81}$$

$$\sigma_{12} = \frac{P}{2I}\left(\frac{D^2}{4} - {x_2}^2\right) \tag{82}$$

where $I = \frac{D^3}{12}$ is the moment of inertia for a beam with rectangular cross-section. Eqs. (78) and (79) present essential boundary condition at $x = 0$, from $-D/2 \le y \le D/2$ and traction boundary condition on the remaining surfaces are imposed using Eqs. (80)–(82). Plane stress is assumed with Young's modulus $E = 10,000psi$ and Poisson's ratio $\nu = 0.3$. Note that all the boundary conditions in PDM should be explicitly enforced. This is not the case when operating with the weak form of the governing PDE, such as those produced with Galerkin methods where the traction-free boundary condition is automatically met without an additional treatment.

The relative errors in numerical solution are measured both in L^2 and L^∞ senses. When computing L^2 error, a background mesh is employed although it is not required in the PDM. The relative L^2 error in displacement norm and the energy error norm are defined as follows

$$E_{L^2} = \frac{\left[\int_\Omega (\mathbf{u}_{nu} - \mathbf{u}_{ex})^T(\mathbf{u}_{nu} - \mathbf{u}_{ex})d\Omega\right]^{1/2}}{\left[\int_\Omega \mathbf{u}_{ex}^T\mathbf{u}_{ex}d\Omega\right]^{1/2}} \tag{83}$$

$$E_{Energy} = \frac{\left[\int_\Omega (\boldsymbol{\varepsilon}_{nu} - \boldsymbol{\varepsilon}_{ex})^T(\boldsymbol{\sigma}_{nu} - \boldsymbol{\sigma}_{ex})d\Omega\right]^{1/2}}{\left[\int_\Omega \boldsymbol{\varepsilon}_{ex}^T\boldsymbol{\sigma}_{ex}d\Omega\right]^{1/2}} \tag{84}$$

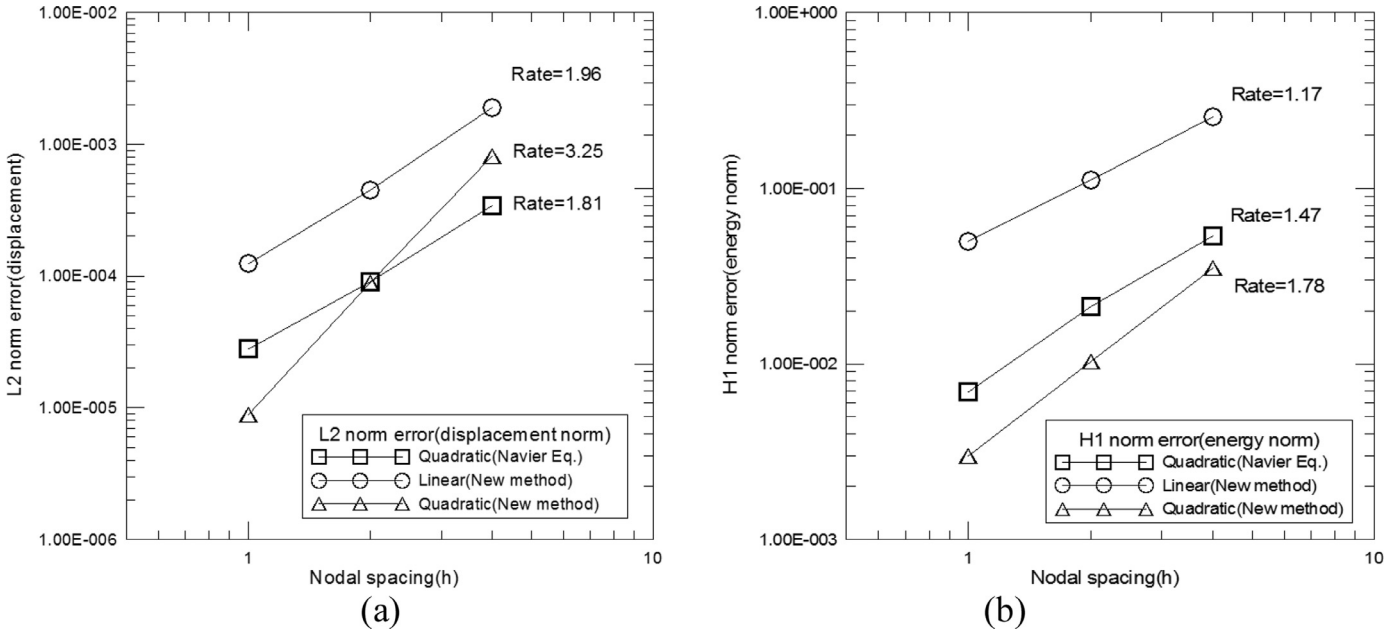

Fig. 5. Convergences of L^2 error norms for 2D elastic cantilever beam problem (a) relative L^2 error norm in displacement and (b) relative energy error norm.

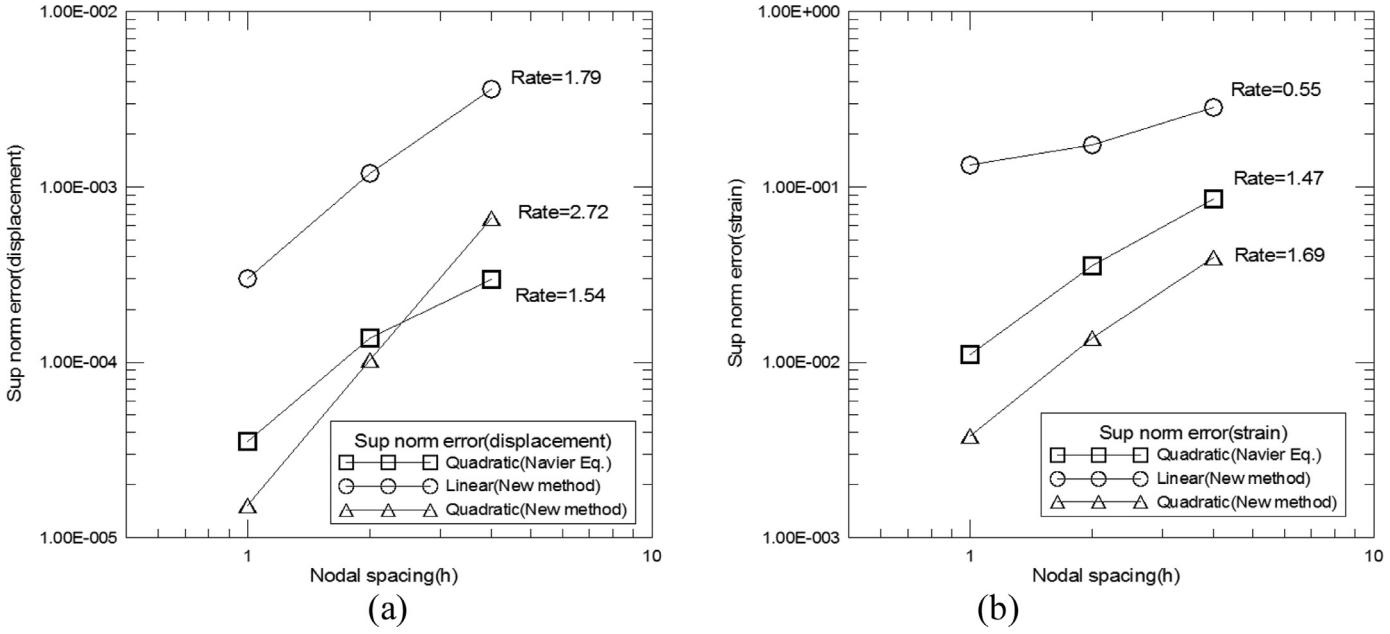

Fig. 6. Convergences of L^∞ error norms for 2D elastic cantilever beam problem (a) relative L^∞ error norm in displacement and (b) relative L^∞ error norm in strain.

where subscript '*ex*' indicates exact solutions and subscript '*nu*' indicates the numerical solutions. In the L^∞ error norm, the sup norm is measured to investigate the maximum nodal error for the kinematic variables of interest; it is a preferable and appropriate measure of error for the strong formulations like PDM because no mesh structure is involved in those formulations. In addition, error measurement in an integral sense like L^2 error may not be suitable for measuring the error of PDM solution since all the computations are conducted at nodes. The L^2 error norm represents the error accumulated over the whole problem domain, whereas the L^∞ error norm denotes the local error occurred in a highly error-inducing region. It is evaluated at node so that it implies the maximum nodal error over all the given nodes.

The relative L^∞ error norms for displacement and strain are defined as following

$$E_{L_u^\infty} = \frac{\underset{I\in\Lambda}{Sup}\left\|u_1^{nu}(\mathbf{x}_I) - u_1^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|u_2^{nu}(\mathbf{x}_I) - u_2^{ex}(\mathbf{x}_I)\right\|}{\underset{I\in\Lambda}{Sup}\left\|u_1^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|u_2^{ex}(\mathbf{x}_I)\right\|} \tag{85}$$

$$E_{L_\varepsilon^\infty} = \frac{\underset{I\in\Lambda}{Sup}\left\|\varepsilon_{11}^{nu}(\mathbf{x}_I) - \varepsilon_{11}^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|\varepsilon_{22}^{nu}(\mathbf{x}_I) - \varepsilon_{22}^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|2\varepsilon_{12}^{nu}(\mathbf{x}_I) - 2\varepsilon_{12}^{ex}(\mathbf{x}_I)\right\|}{\underset{I\in\Lambda}{Sup}\left\|\varepsilon_{11}^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|\varepsilon_{22}^{ex}(\mathbf{x}_I)\right\| + \underset{I\in\Lambda}{Sup}\left\|2\varepsilon_{12}^{ex}(\mathbf{x}_I)\right\|} \tag{86}$$

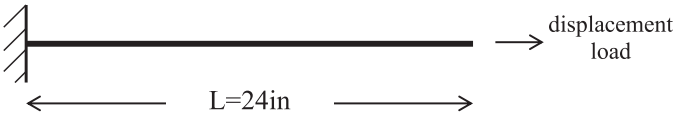

Fig. 7. One dimensional inelastic rod.

where $u_1^{nu}(\mathbf{x}_I)$, and $u_2^{nu}(\mathbf{x}_I)$ are x- and y-directional displacements obtained at $\mathbf{x}_I$ while $u_1^{ex}(\mathbf{x}_I)$, and $u_2^{ex}(\mathbf{x}_I)$ are the exact solutions at the same position. Similarly, $\varepsilon_{11}^{nu}(\mathbf{x}_I)$, $\varepsilon_{22}^{nu}(\mathbf{x}_I)$, and $\varepsilon_{12}^{nu}(\mathbf{x}_I)$ are strain components computed at $\mathbf{x}_I$ while $\varepsilon_{xx}^{ex}(\mathbf{x}_I)$, $\varepsilon_{yy}^{ex}(\mathbf{x}_I)$, and $\varepsilon_{xy}^{ex}(\mathbf{x}_I)$ are the exact values.

Fig. 5(a) and (b) presents convergence rates for the relative L^2 error norm in displacement and energy, respectively. As is well known, for linear approximations of the weak formulation like FEM, the optimal convergence rates for L^2 error are two and one for the displacement and energy, respectively. Utilizing a Taylor polynomial in the PDM yields the convergence rates of 3.25 and 1.78 for the L^2 displacement error norm and energy error norm, respectively. Larger convergence rates are observed as compared to the optimal rates owing to the use of quadratic Taylor polynomial. Greater convergence rates are also observed as compared to the conventional method which utilizes the second order derivative approximation in Navier's equation. The performance of the PDM combined with the linear Taylor polynomial is also shown. Note that the developed PDM, which can be referred to as a '*truly*' strong formulation, is able to use the linear Taylor polynomial in solving the second order PDE even though it produces the larger convergence when combined with the quadratic Taylor polynomial. As shown in Fig. 5(a) and (b), application of a linear Taylor polynomial is applied produces convergence rates of 1.96 and 1.17. These rates are very close to the aforementioned optimal convergence rates, thereby showing that either linear or quadratic Taylor polynomials used in the PDM results in competitive or higher convergence rates than the conventional method.

In terms of magnitude, using the quadratic Taylor polynomial generates the smallest relative error. Fig. 6(a) and (b) present convergence rates for the relative L^∞ error norms in displacement and strain fields, respectively. When the PDM is combined with the quadratic Taylor polynomial, it yields the convergence rates of 2.72 and 1.69 for displacement and strain fields. The PDM combined with the linear Taylor polynomial yields convergence rates of 1.79 and 0.55 for displacement and strain fields, respectively. The conventional method involving the Navier's equation and the second order derivative approximation generates the convergence rates 1.54 and 1.47 for the displacement and strain, respectively. The convergence rates measured in L^∞ sense show similar trends with the L^2 error cases. Therefore, it is recommended to use the quadratic Taylor polynomial for constructing the double

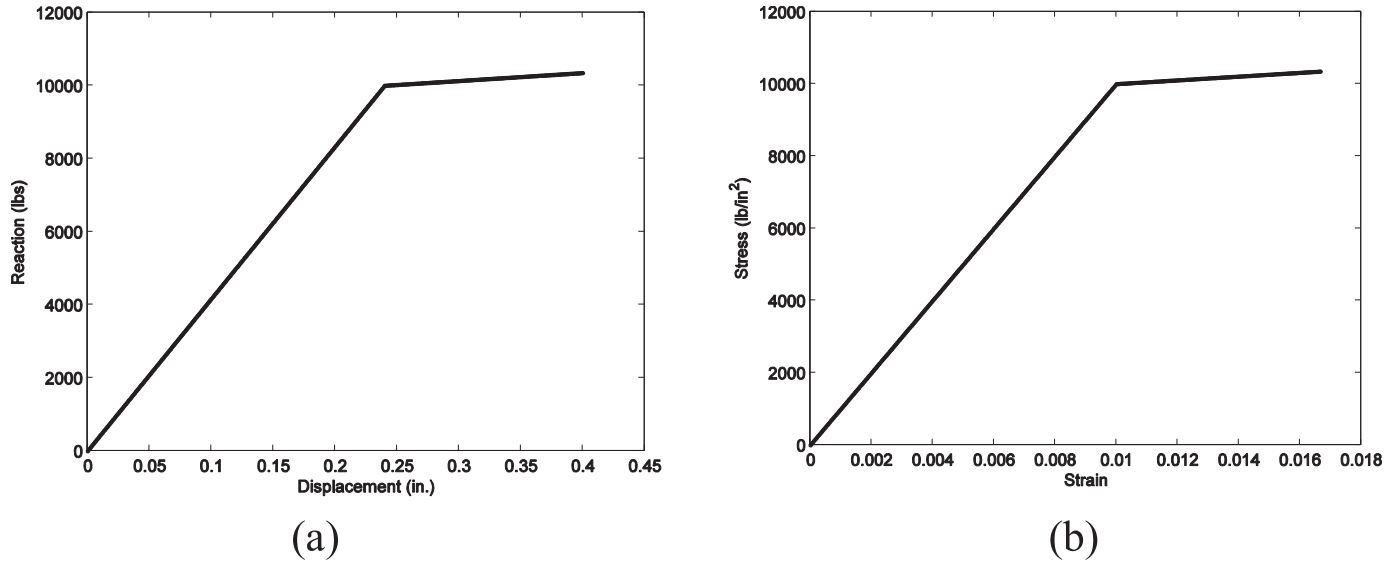

Fig. 8. Numerical results for one dimensional rod elongation problem (a) displacement and reaction force and (b) stress and strain response at the center node.

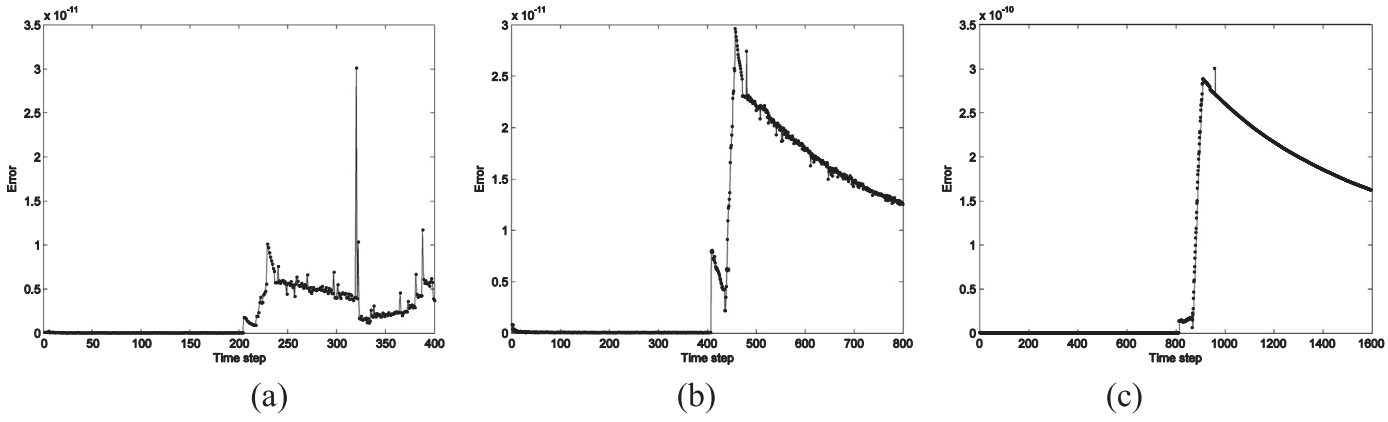

Fig. 9. Relative L^2 error norm in displacement according to loading steps for one dimensional rod elongation problem for (a) 61 node model (b) 121 node model and (c) 241 node model.

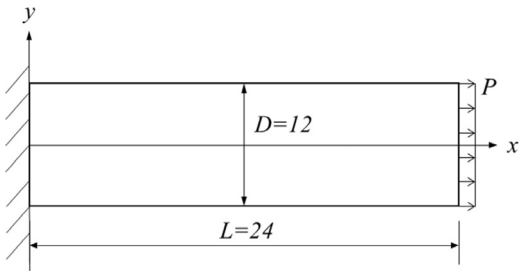

Fig. 10. 2-D inelastic beam problem under uniform tensile load.

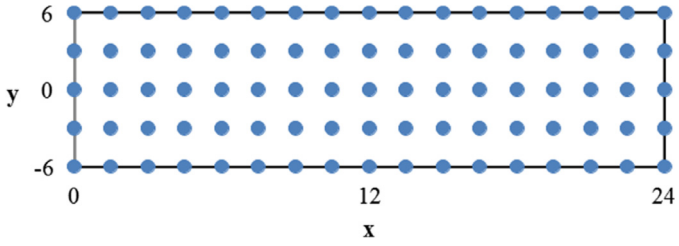

Fig. 11. Node distribution of 2-D inelastic beam problem.

derivative approximation because the quadratic Taylor polynomial does not produce a notable increase in computational effort as compared to the linear Taylor polynomial. Furthermore, it achieves significant improvement in solution accuracy thereby guaranteeing smaller error and faster convergence so that it is justified to be implemented in the iterative algorithm for solving nonlinear material problems.

4.3. Elongation of one dimensional inelastic rod

A one dimensional inelastic rod subjected to a tensile load is analyzed by the newly developed PDM (See Fig. 7). Material parameters applied are $E=1\times10^5$ *psi* (Young's modulus), $\sigma_y=1\times10^4$ *psi*, $K=5\times10^4$ *psi* (plastic modulus), $H=5\times10^3$ *psi* (kinematic hardening modulus). 61

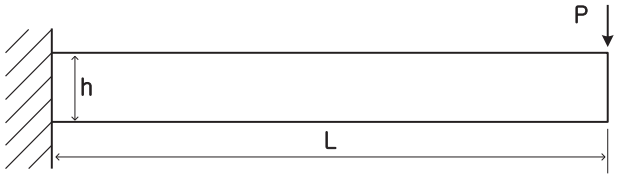

Fig. 13. Cantilever beam subjected to a concentrated load.

nodes and the quadratic Taylor polynomial are used in the numerical model and the displacement at the right end is controlled during the simulation. The reaction force response as a function of displacement is shown in Fig. 8(a) where the strain hardening behavior is clearly seen. In addition, Fig. 8(b) shows the stress-strain curve computed at the middle of rod; the linear hardening behavior after yielding is accurately described.

Convergence with the PDM is well behaved when finding nonlinear solutions. During the analysis, the number of iterations to reach convergence during plastic stage computation never exceeded five. To investigate stability of the iterative algorithm, the relative L^2 error norm defined in Eq. (83) was computed after achieving convergence for each incremental step and was plotted in Fig. 9(a)–(c). The magnitude of the relative error is less than 1.0×10^{-10} at all time steps and the numerical error exhibits relatively stable behavior for both elastic and plastic regimes; yielding begins around 200, 400, and 800 steps for 61, 121, and 241 node models, respectively. As the number of nodes increases, the absolute value of relative error does not decrease but shows quite stable decreasing trend. This shows the PDM with the double derivative approximation performs well in the analysis of one dimensional material nonlinear problem.

4.4. Two-dimensional inelastic beam under tensile load

This section considers a fixed-free two-dimensional inelastic rod model which was solved in the previous section using one-dimensional model. Fig. 10 presents an illustrative description for the simulation. Material parameters applied in the simulation are the same as those

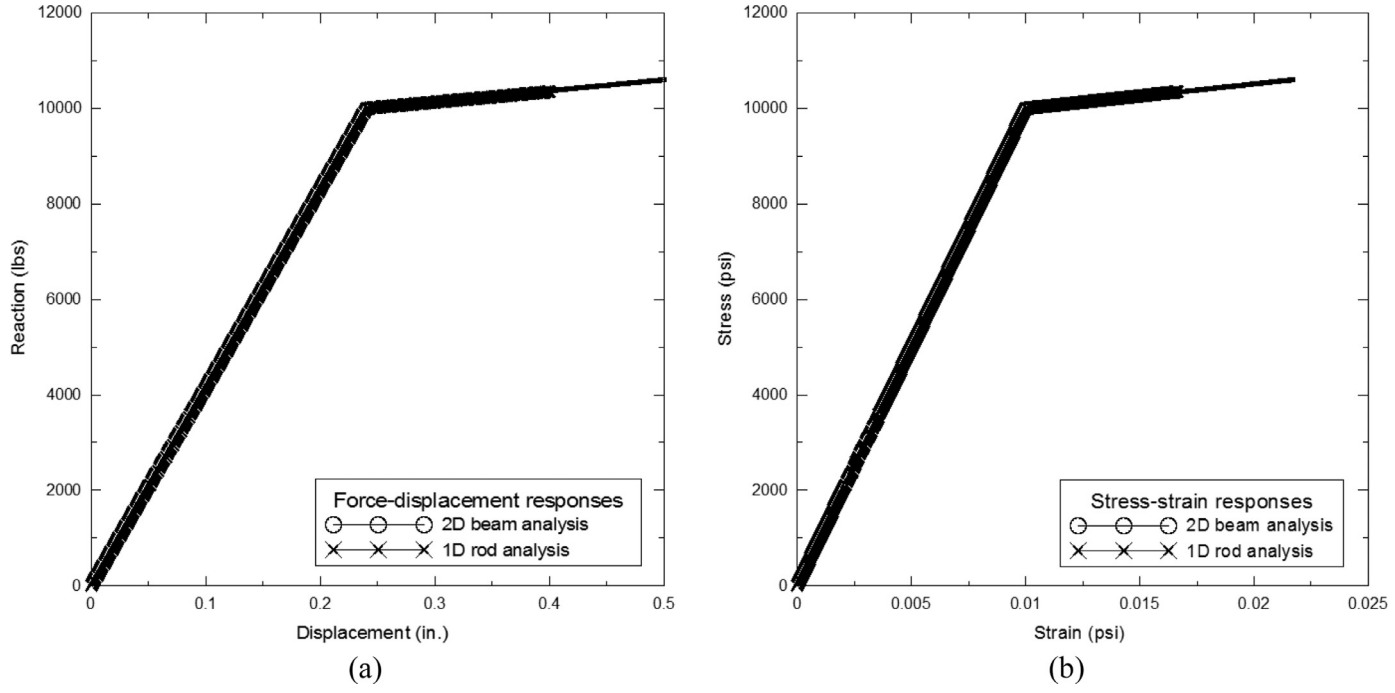

Fig. 12. Comparisons of numerical results for two dimensional beam elongation problems compared with one dimensional simulation case (a) reaction force and displacement responses and (b) stress and strain responses at the center node.

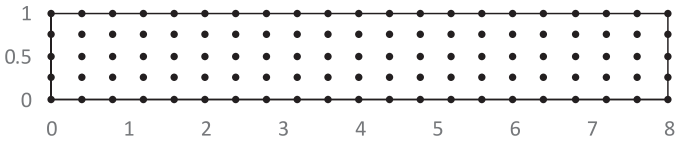

Fig. 14. Node distribution for a cantilever beam problem.

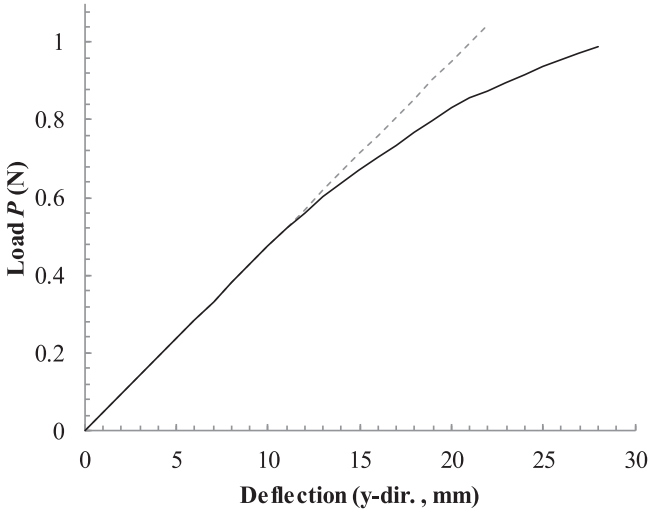

Fig. 15. Load-displacement curve for the cantilever beam under concentrated load (at the end of the beam).

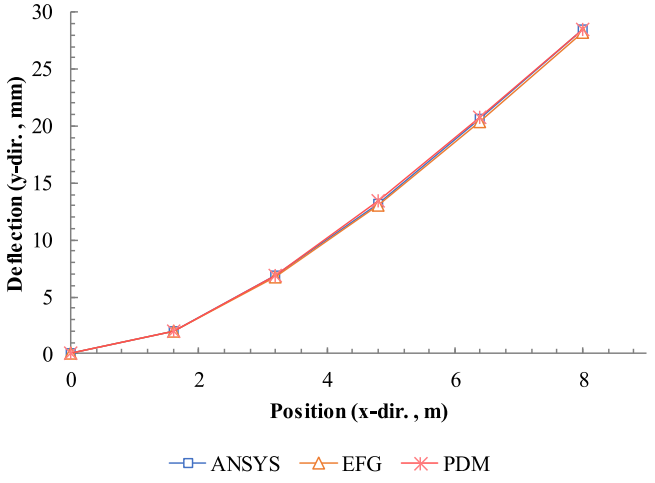

Fig. 16. Comparisons of deflection curves for the cantilever beam under concentrated load problem (at the last loading step).

in one dimensional case; however, the Poisson's ratio is assumed to be zero to equalize the simulation condition with the one-dimensional case. Fig. 11 depicts the numerical model where 153(= 17 × 9) nodes are regularly distributed. The analysis assumes a plane stress condition and employs the quadratic Taylor polynomial in constructing the double derivative approximation.

Fig. 12(a) provides a comparison of the reaction force-displacement curves for the one and two-dimensional cases. The reaction force was calculated by summing up the nodal forces along the displacement boundary. As seen in the figure, the results for both cases are in agreement with each other. Fig. 12(b) presents the stress-strain relation for the node placed at the center of beam. The two-dimensional case agrees pretty well with the one-dimensional result. For most incremental steps, the number of iterations needed to attain convergence during the plastic stage computations approximately ranged from five to six; in all incremental steps, the iteration number did not exceed ten. The convergence behavior remains stable without any abrupt increase in iteration number when passing from the elastic to plastic regions. The associated computational efficiency is mostly attributed to the adoption of the algorithmic tangent moduli as well as robustness of the double derivative approximation. It was proven that the double derivative approximation provides sufficient accuracy in derivative computation required in the strong formulation for the material nonlinear problem. Therefore, the PDE, which includes the second order derivative function, can be successfully discretized by the first order derivative approximation without involving any weak formulation which essentially requires the numerical integration process.

4.5. Two dimensional beam under concentrated load

A cantilever beam subjected to a concentrated load at the end is analyzed. Fig. 13 shows a schematic of problem; the length of the beam is $L = 8\ m$, the height is $h = 1\ m$ and the concentrated load is $P = 1\ N$. Material parameters such as Young's modulus of $E = 1 \times 10^5\ N/m^2$, Poisson's ratio of $\nu = 0.25$, yield stress of $\sigma_y = 25 N/m^2$ are applied in the analysis. The linear hardening elasto-plastic model is adopted with $E^{E-P} = 0.2E$. Fig. 14 provides the node arrangement using 105(= 21 × 5) nodes. The simulation considers a quadratic Taylor polynomial and assumes a plane stress condition. Peng et al. (2011) solved this problem by using a form of element-free Galerkin method which was named as the complex variable element-free Galerkin method.

Load-deflection curve is presented in Fig. 15 where the load and deflection are taken at the end of the beam. Fig. 16 presents deflection curves which are evaluated along the center line at the last loading step. For comparison, the deflection curves obtained by the Element-Free Galerkin method [2] and ANSYS are plotted together. All deflection curves are in nearly identical to each other.

4.6. V-notched tension specimen

A V-notched specimen under tension is analyzed using the developed PDM. Plane stress condition is assumed and a perfectly plastic material law governs the material behavior. Material parameters used are Young's modulus of $E = 1.96 \times 10^5 \mathrm{N/mm^2}$, Poisson's ratio of $\nu = 0.3$ and

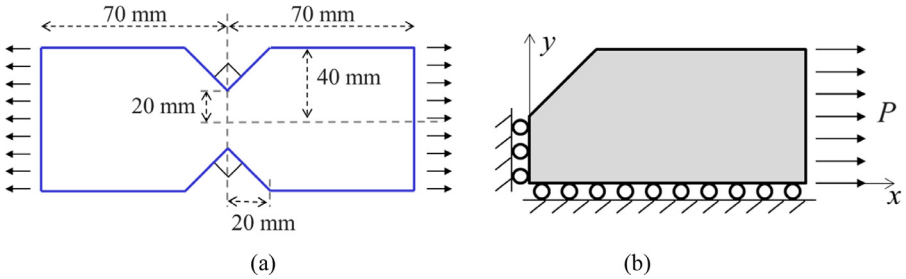

Fig. 17. Schematic drawings of cylinder with internal pressure problem (a) problem description and (b) one quarter numerical model.

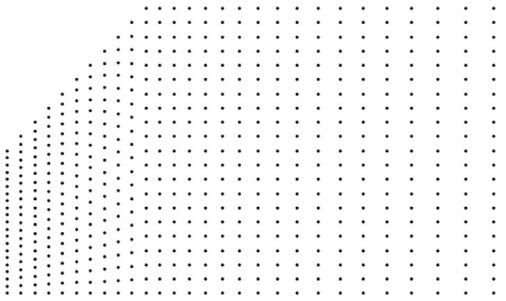

Fig. 18. Numerical model with adaptive node arrangement for V-notched tension specimen.

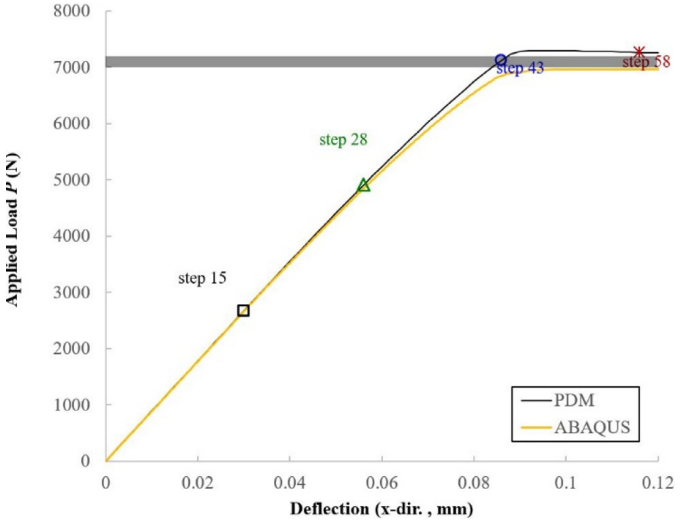

Fig. 19. A comparison of load-deflection curves for the V-notched tension specimen problem where the shaded area denotes the range of peak loads observed by Yamada et al. [23].

yield stress of $\sigma_y = 294$ N/mm^2. Fig. 17(a) and (b) show a schematic of the problem and a quarter model for numerical simulation, respectively. Due to the symmetry in geometry and loading, only one quarter (the upper right quadrant) of the specimen is analyzed. The boundary conditions are explanatorily depicted in the numerical model; P denotes the load acting on half-width of the specimen. Fig. 18 presents the numerical model where 588 nodes are adaptively distributed around the notch. The reader is referred to Dai et al. [5] for which this problem was analyzed by using a weak-form Radial Point Interpolation Method based on the deformation theory.

The V-notch inherently induces a severe stress concentration deep in the sharp angle; this phenomenon naturally provokes strain localization leading to material yielding. Fig. 19 shows load-deflection curves for the PDM and FEM (ABAQUS) models. Nonlinear effects for the system are observed when load increment step reaches 48-th step (i.e., applied load around 4900 N). At this load, the system exceeds the critical stress in the neighborhood around the notch and begins a transition into the plastic regime. As shown in Fig. 19, the evolution from the elastic to plastic regime for the system occurs at a higher rate in FEM (ABAQUS) model as compared to the PDM model. This is attributed to differences in each method where the PDM employs higher order interpolation functions that depend on neighboring points within the region of compact support such that the displacement field is smooth and continuous across the body. Overall, the results for both models agree well with the range of critical loads presented by Yamada et al. [23]. In Fig. 20, normal stress distributions along the vertical minimum section($x = 0$) are plotted for four selected loading steps which are already depicted in Fig. 19. As the loading step advances, the normal stress distributions gradually converge to the yielding state corresponding to a yield stress of 294 N/mm^2. It is observed that when the loading reaches $P = 3027$ N (at 17-th step), iteration number begins to abruptly increase which is attributed to yielding initiation near the notch root while most other part is still in elastic state. At the 28-th loading step, the stress near the notch root reaches the specific stress level driving the material to enter the plastic stage. At the 52th loading step ($P = 7287$ N), the normal stresses of all nodes located along $x = 0$ reach the yield stress of 294N/mm^2 so that the central elastic region disappears and the plastic region extends to the

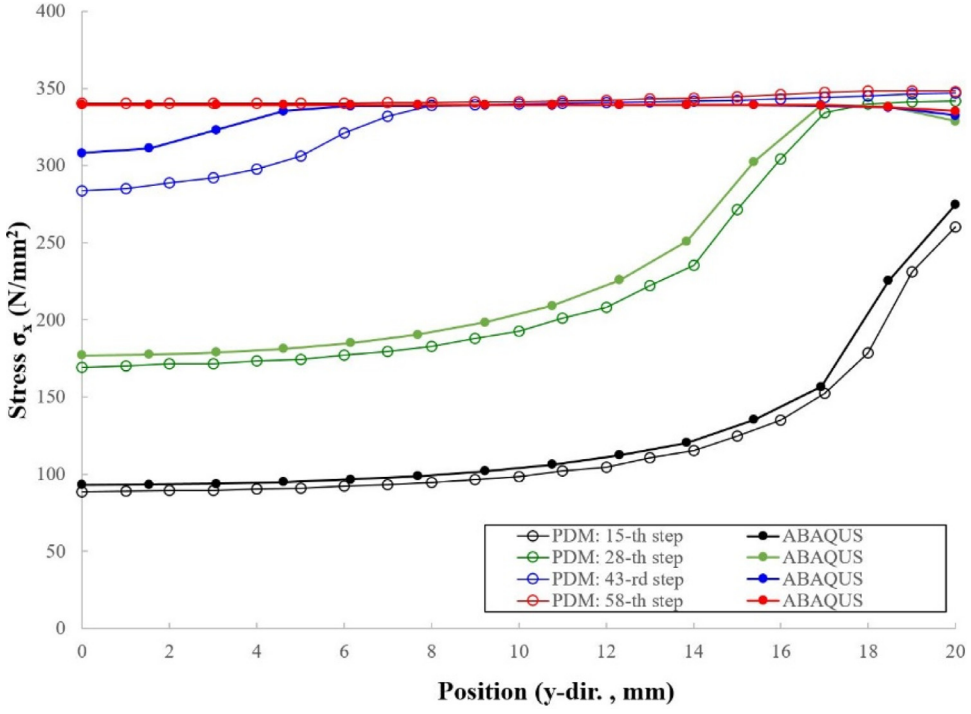

Fig. 20. Normal stress distributions along $x = 0$ for V-notched tension specimen problem.

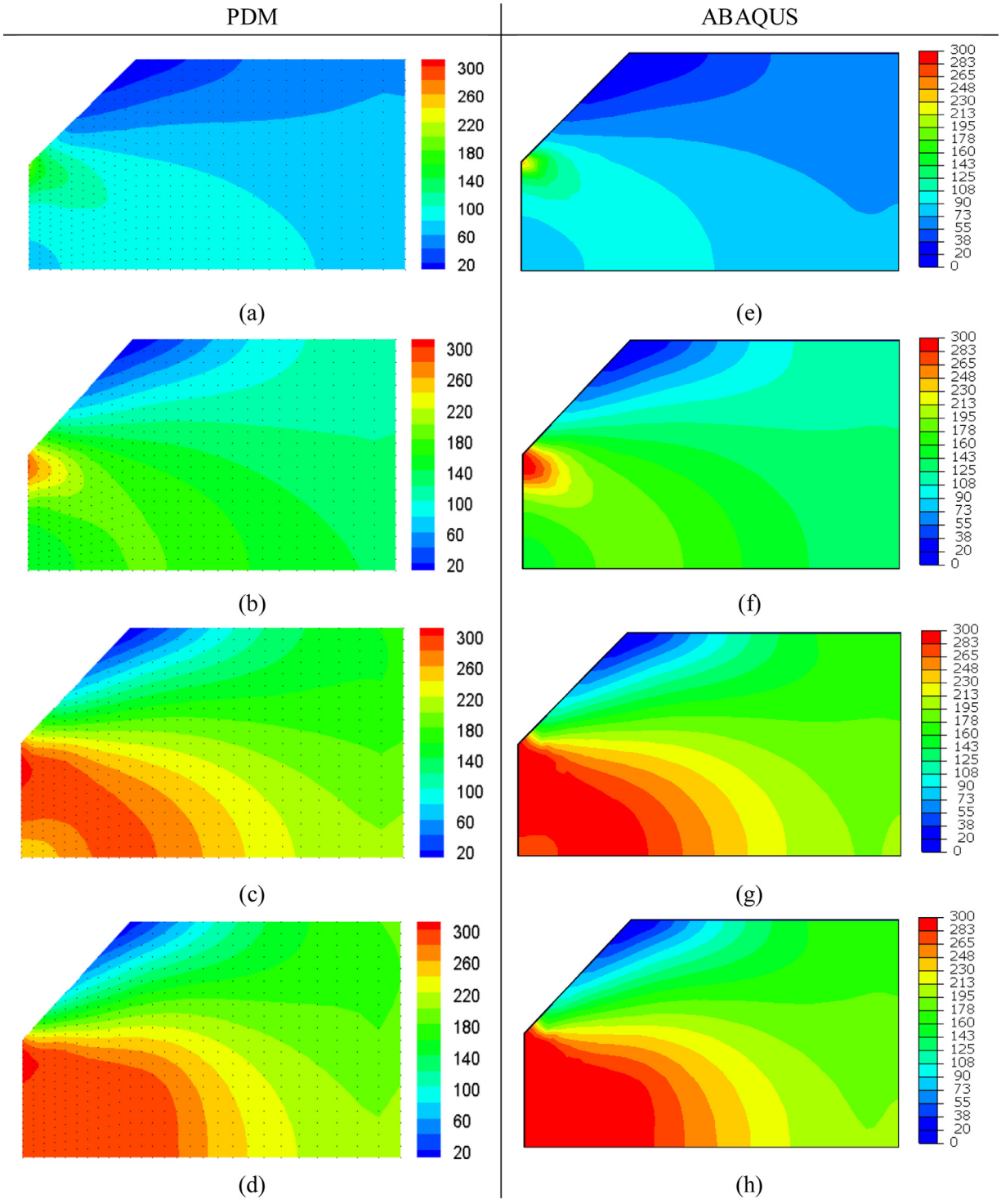

Fig. 21. The PDM (a-d) and corresponding FEM (ABAQUS) (e-h) contour plots of von-Mises stress distributions for V-notched tension specimen (unit: N/mm^2); (a)-(d) correspond to 15-th, 28-th, 43-rd, and 58-th load steps.

entire minimum section of the specimen. Note that the closer a node is in proximity to the notch root, the earlier the stress state approaches to the yield state. In addition, Fig. 21(a)–(d) depict contour plots for von-Mises stress at the selected loading steps. They clearly show that how the plastic zone develops and extends from the region near the notch root to other remaining part of the specimen. When a collapse load is assumed to be the value at which the plastic region develops to the entire minimum section of the specimen, it is determined to be $P = 7287\ N$ by the numerical simulation; it agrees moderately well with $P_c = 7197\ N$ given by Yamada et al. [23]. Also, the excellence in numerical stability is verified because even after the 52th loading step yielding the collapse load, the numerical procedure still gave the converged solution (See Figs. 19–21); other numerical schemes like Dai et al. [5] often failed to generate the converged solution after reaching the collapse load. Thus, the present solution still looks promising.

4.7. Perforated plate with circular hole

A perforated plate with a circular hole under uniform tension is analyzed using a plastic material law under the assumption of plane

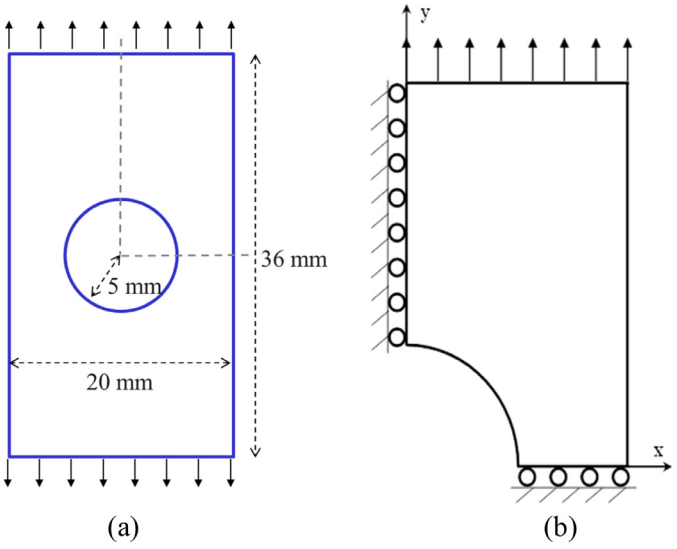

Fig. 22. Perforated plate problem description and an illustrative quarter model (a) geometry (b) problem domain.

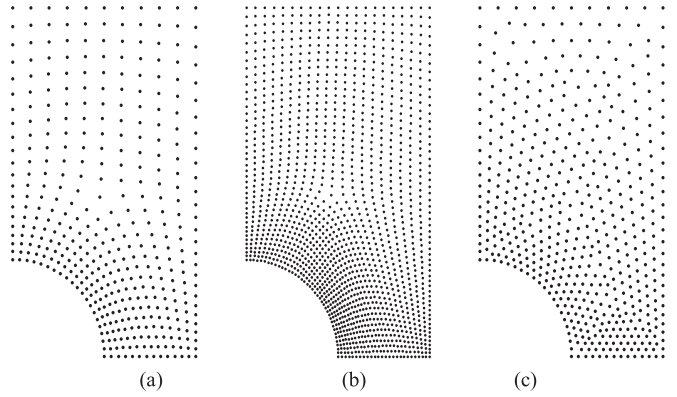

Fig. 23. Numerical models for the perforated plate problem (a) 401 nodes model with regular distribution (b) 1486 nodes model with regular distribution(c) 524 nodes model with irregular distribution.

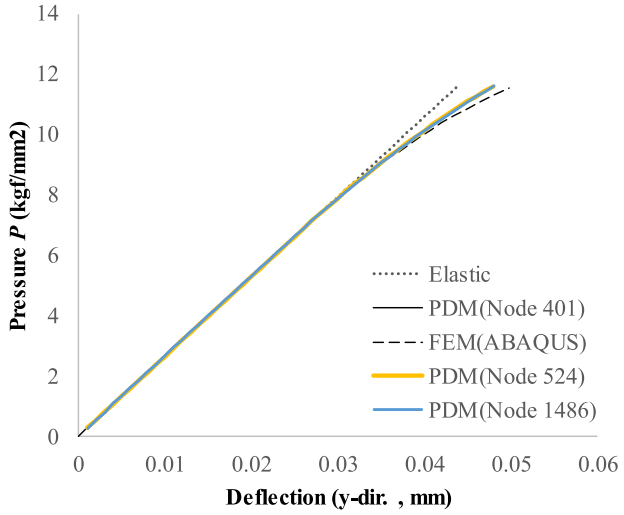

Fig. 24. Vertical displacement on top surface according to applied pressure.

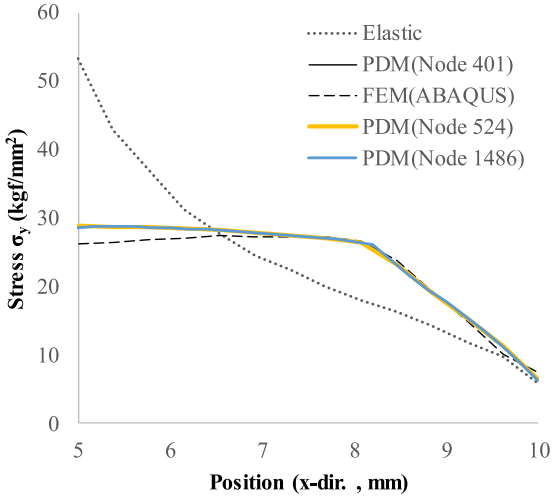

Fig. 25. Normal stress(σ_{22}) distribution along y = 0 for perforated plate problem.

stress condition. The work hardening material is considered with following material parameters; Young's modulus of $E = 7000\ \mathrm{kgf/mm^2}$, plastic modulus of $K = 217\ \mathrm{kgf/mm^2}$, Poisson's ratio of $\nu = 0.2$ and yield stress of $\sigma_y = 24.3\ \mathrm{kgf/mm^2}$. The problem configuration and upper right quadrant are shown in Fig. 22(a) and (b), respectively. Boundary conditions are defined considering the symmetry condition; the upper side of the plate is uniformly stretched while the horizontal component of the left side and the vertical component of the bottom side are constrained. Fig. 23(a)–(c) present the numerical models; Fig. 23(a) and (b) show that 401 and 1486 nodes are regularly and adaptively distributed around the hole, respectively while Fig. 23(c) depicts the 524 nodes model with an irregular distribution.

In Fig. 24, the displacements at the middle point on the top surface are plotted according to the applied pressure; the pressure-displacement responses obtained by the developed PDM based on 401, 1486 and 524 nodes models agree well with that computed by the FEM (ABAQUS). For comparison, the numerical result for elastic material is plotted together. It is seen that the pressure initially increases in accordance with the trajectory of elastic material behavior and then decelerates after the material begins to yield. Fig. 25 presents the computed stress profiles (σ_{22}) along the horizontal minimum section of plate (y = 0). For comparison, the stress distribution obtained by FEM(ABAQUS) is plotted together; for a reference, the stress distribution computed considering elastic material property is also presented. It is observed that stress distributions obtained by the developed PDM and FEM are in quite good agreement. Regular and irregular models with 401, 1486 and 524 nodes produce almost identical stress distributions. However, slight discrepancy in the stress profiles near the natural boundary occurs because the PDM involves the first order derivative computation in the natural boundary enforcement and only half of the influence domain is formed on the boundary. So, the accuracy of traction boundary enforcement is slightly inferior to that of the essential boundary node which employs the zeroth order derivative approximation and that of the interior node which can construct a complete circular influence domain; the similar phenomenon occurs in the finite difference scheme and in fact, it can be effectively improved by introducing various modified differencing techniques [12]. In addition, Fig. 26(a) presents a contour plot for von-Mises stress computed at the last loading step using the PDM; in Fig. 26(b), the stress contour plot obtained by FEM is presented for comparison. It is noticed that the stress distributions obtained by the PDM and FEM seem to reasonably agree to each other and a shear band is about to develop along the diagonal direction from the right edge of the hole. Furthermore, Fig. 26(c) and (d) show that stress distributions computed by the model with regularly distributed 1486 nodes and irregularly distributed 524 nodes. These results might demonstrate that the PDM seldom provokes the sensitivity problem of node density, even in

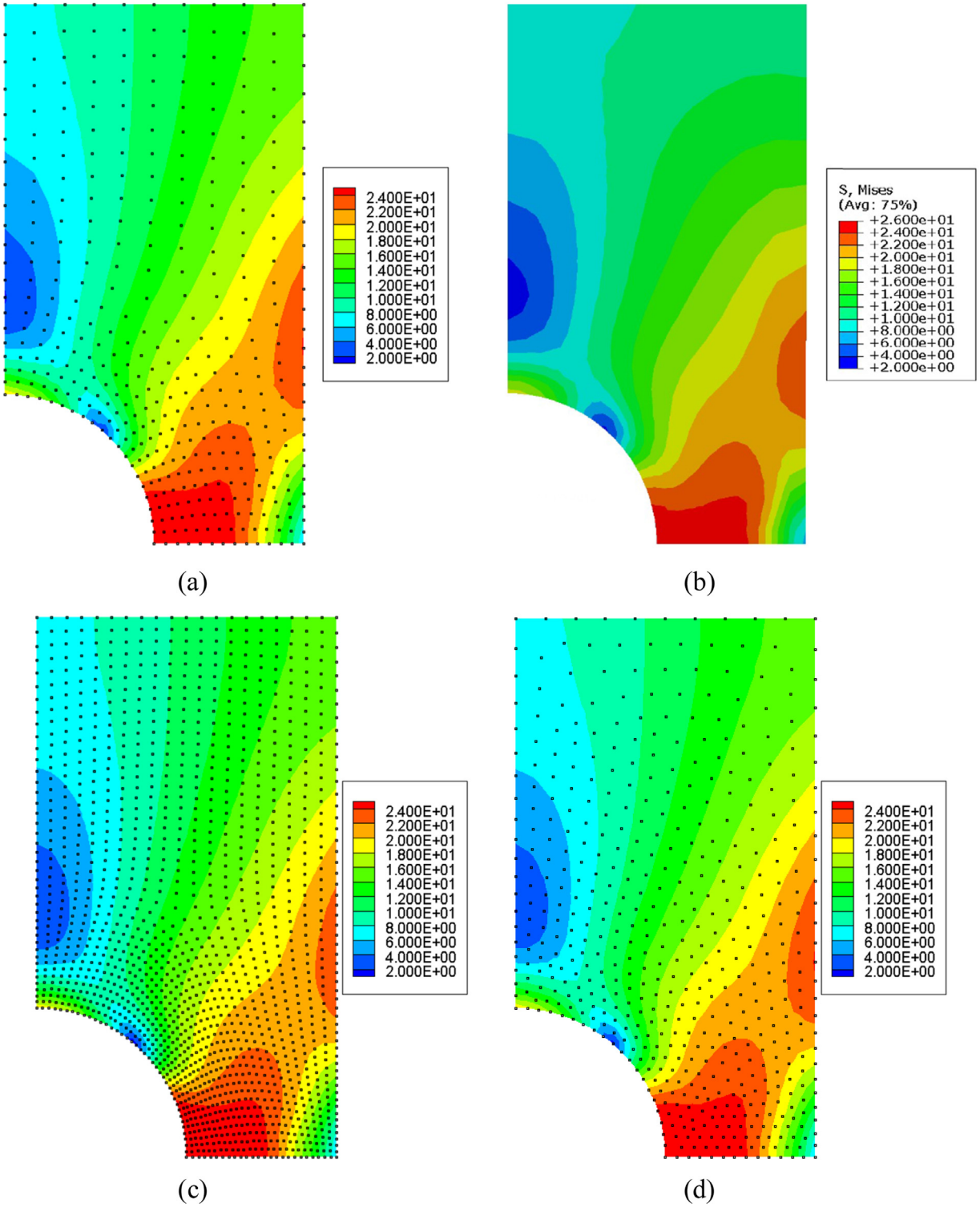

Fig. 26. Contour plots of von-Mises stress distributions at the last loading step obtained by (a) 401 nodes PDM model with regular distribution and (b) FE model(ABAQUS) (c) 1486 nodes PDM model with regular distribution (d) 524 nodes PDM model with irregular distribution (unit: kgf/mm^2).

case of the irregular node distribution although this issue will be more thoroughly dealt with in the separated future works.

5. Conclusions

In this study, a strong form meshfree method based on the Particle Difference Method (PDM) was developed for material nonlinear problems. Previously, the PDM has utilized the Navier's equation which unifies kinematic unknowns into displacement. Although this approach is common in other strong form meshfree schemes, it generated difficulties in handling nonlinear material model. This study further developed the PDM to analyze the material nonlinear behavior in the field of solid mechanics. The double derivative approximation was devised to explicitly implement the inelastic constitutive equation within the

framework of strong formulation. It effectively negates the need for a direct second order derivative approximation which is required for solving Navier's equation. As a result, the linear momentum equation, which is expressed by the divergence of stress tensor, can be directly discretized using the first order derivative approximation only. The divergence of stress and the strain were then computed by applying the double derivative approximation. The application of double derivative approximation enables the momentum equation to be discretized in connection with the constitutive equation so that the conventional constitutive models for nonlinear material can be readily implemented. Furthermore, the nonlinear iterative procedure is based on the Newton method which utilizes the return mapping algorithm to compute and update stresses and kinematic variables thereby improving the computational efficiency of the iterative solution procedure using the algorithmic tangent moduli. Although the developed PDM is a strong formulation, it has no difficulty in utilizing the framework of conventional nonlinear FEM.

The reproducing test was performed to verify the mathematical validity of double derivative approximation. From the comparison with the second order derivative approximation, the double derivative approximation was proved to show the accuracy superior or at least equivalent to that of the second order derivative approximation which was often used with the Navier's equation. In particular, when combined with the quadratic Taylor polynomial, it shows better performance than the second order derivative approximation. The numerical procedure proposed in this paper was benchmarked against various solid mechanics problems involving perfect plastic and strain hardening material behavior; inelastic rods, inelastic beam subjected to tensile and vertical loads, the V-notched specimen under tension, and the perforated plate with a circular hole under uniform tension. In each problem, kinematic variables such as displacement, strain and stress were checked if they correspond to the analytical inelastic behavior. It was shown that the computation results are in agreement with the closed-form solutions or the results obtained by the FEM. Also, it was briefly verified that the PDM seldom provokes the node density sensitivity problem even with the irregular node distribution. The fact that the iteration number for finding the converged solution was less than five for one dimensional case and less than ten for two dimensional cases demonstrates efficiency of the nonlinear procedure. The compatibility between the PDM and the developed nonlinear procedure involving double derivative approximation was successfully demonstrated implementing the return mapping algorithm and algorithmic tangent moduli. Therefore, the new strong formulation has proven to solve material nonlinear problems accurately and efficiently and is readily expected to extend to the large deformation problem involving both material and geometric nonlinearities.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT &Future Planning (NRF-2017R1A1A1A05001196).

References

[1] Aluru NR. A point collocation method based on reproducing kernel approximations. Int J Numer Methods Eng 2000;47:1083–121.
[2] Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. Int J Numer Methods Eng 1994;37:229–56.
[3] Bordas S, Rabczuk T, Zi G. Three-dimensional crack initiation, propagation, branching and junction in non-linear materials by an extended meshfree method without asymptotic enrichment. Eng Fract Mech 2008;75(5):943–60.
[4] Chen J-S, Hillman M, Chi S-W. Meshfree methods: progress made after 20 years. J Eng Mech 2017;143(4):04017001.
[5] Dai KY, Liu GR, Han X, Li Y. Inelastic analysis of 2D solids using a weak-form RPIM based on deformation theory. Comput Methods Appl Mech Eng 2006;195:4179–93.
[6] Fu Y, Michopoulos JG, Song J-H. Bridging the multi-phase field model with the molecular dynamics for the solidification of nano-crystals. J Comput Sci 2017;20:187–97.
[7] Gu YT, Wang QX, Lam KY, Dai KY. A pseudo-elastic local meshless method for analysis of material nonlinear problems in solids. Eng Anal Bound Elem 2007;31:771–82.
[8] Kim DW, Kim YS. Point collocation method using the fast moving least-square reproducing kernel approximation. Int J Numer Methods Eng 2003;56:1445–64.
[9] Lee S-H, Yoon Y-C. Meshfree point collocation method for elasticity and crack problem. Int J Numer Methods Eng 2004;61:22–48.
[10] Lee S-H, Kim K-H, Yoon Y-C. Particle difference method for dynamic crack propagation. Int J Impact Eng 2016;87:132–45.
[11] Luo Y, Häussler-Combe U. A generalized finite-difference method based on minimizing global residual. Comput Methods Appl Mech Eng 2002;191:1421–38.
[12] Morton KW, Mayers DF. Numerical solution of partial differential equations. 2nd ed. Cambridge; 2005.
[13] Pozo PL, Perazzo F, Angulo A. A meshless FPM model for solving nonlinear material problems with proportional loading based on deformation theory. Adv Eng Softw 2009;40:1148–54.
[14] Rabczuk T, Eibl J. Modelling dynamic failure of concrete with meshfree methods. Int J Impact Eng 2006;32:1878–97.
[15] Rabczuk T, Belytschko T, Xiao SP. Stable particle method based on Lagrangian Kernels. Comput Methods Appl Mech Eng 2004;193:1035–63.
[16] Rabczuk T, Song J-H, Belytschko T. Simulations of instability in dynamic fracture by the cracking particles method. Eng Fract Mech 2009;76(6):730–41.
[17] Rabczuk T, Gracie R, Song J-H, Belytschko T. Immersed particle method for fluid–structure interaction. Int J Numer Methods Eng 2010;81(1):48–71.
[18] Simo JC, Taylor RL. Consistent tangent operators for rate-independent elastoplasticity. Comput Methods Appl Mech Eng 1985;48:101–18.
[19] Simo JC, Hughes TJR. Computational inelasticity. New York: Springer-Verlag; 1998.
[20] Song J-H, Fu Y, Kim T-Y, Yoon Y-C, Michopoulos JG, Rabczuk T. Phase field simulations of coupled microstructure solidification problems via the strong form particle difference method. Int J Mech Mater Des 2017. doi:10.1007/s10999-017-9386-1.
[21] Timoshenko SP, Goodier IN. Theory of elasticity. 3rd ed. MaGraw-Hill; 1970.
[22] Tu W, Gu YT, Wen PH. Effective shear modulus approach for two dimensional solids and plate bending problems by meshless point collocation method. Eng Anal Bound Elem 2012;36:675–84.
[23] Yamada Y, Yoshimura N, Sakurai T. Plastic stress-strain matrix and its application for solution of elastic plastic problems by finite element method. Int J Mech Sci 1968;10:343–54.
[24] Yoon Y-C, Kim K-H, Lee S-H. Dynamic particle difference method for the analysis of proportionally damped system and cracked concrete beam. Int J Fract 2016;203(1):237–62.
[25] Yoon Y-C, Lee S-H, Belytschko T. Enriched meshfree collocation method with diffuse derivatives for elastic fracture. Comput Math Appl 2006;51(8):1349–66.
[26] Yoon Y-C, Song J-H. Extended particle difference method for weak and strong discontinuity problems: part1. Derivation of the extended particle derivative approximation for the representation of weak and strong discontinuities. Comput Mech 2014a;53(6):1087–103.
[27] Yoon Y-C, Song J-H. Extended particle difference method for weak and strong discontinuity problems: part2. Formulations and applications for various interfacial singularity problems. Comput Mech 2014b;53(6):1105–28.
[28] Yoon Y-C, Song J-H. Extended particle difference method for moving boundary problems. Comput Mech 2007c;54(3):723–43.
[29] Bourantas GC, Mountris KA, Loukopoulos VC, Lavier L, Joldes GR, Wittek A, Miller K. Strong-form approach to elasticity: hybrid finite difference-meshless collocation method (FDMCM). Appl Math Model 2018;57:316–38.