

The Global Nonlinear Galerkin Method for the Analysis of Elastic Large Deflections of Plates under Combined Loads: A Scalar Homotopy Method for the Direct Solution of Nonlinear Algebraic Equations

Hong-Hua Dai^{1,2}, Jeom Kee Paik³ and Satya N. Atluri²

Abstract: In this paper, the global nonlinear Galerkin method is used to perform an accurate and efficient analysis of the large deflection behavior of a simply-supported rectangular plate under combined loads. Through applying the Galerkin method to the governing nonlinear partial differential equations (PDEs) of the plate, we derive a system of coupled third order nonlinear algebraic equations (NAEs). However, the resultant system of NAEs is thought to be hard to tackle because one has to find the one physical solution from among the possible multiple solutions. Therefore, a suitable initial guess is required to lead to the real solution for given load conditions. The feature of this paper is that we apply the global nonlinear Galerkin method to the governing PDEs and solve the resultant NAEs directly in each load step. To keep track of the physical solution, the initial guess for the current load step is provided by taking the solution of the NAEs for the last step as the initial guess. Besides, the size of the NAEs grows dramatically larger, with the increase of the number of terms of the trial functions, which will cost much more computational efforts. An exponentially convergent scalar homotopy algorithm (ECSHA) is introduced to solve the large set of NAEs. The approach in the present paper is more direct and simpler than either the incremental global Galerkin method, or the incremental local Galerkin method (finite element method) based on a symmetric incremental weak-form; both of which methods lead to the inversion of tangent stiffness matrices and Newton-Raphson iterations in each load step. The present method of exponentially convergent scalar homotopy of directly solving the NAEs is much better than the quadratically convergent Newton-Raphson method. Several numerical examples are provided to validate the feasibility and efficiency of the proposed scheme.

¹ College of Astronautics, Northwestern Polytechnical University, Xi'an 710072, P.R. China

² Center for Aerospace Research & Education, University of California, Irvine

³ Lloyd's Register Educational Trust (LRET) Center of Excellence, Pusan National University, Korea

Keywords: large deflections, combined loads, nonlinear global Galerkin method, nonlinear algebraic equations (NAEs), initial guess, exponentially convergent scalar homotopy algorithm (ECSHA).

1 Introduction

The large deflection behavior of plates under combined loads has been a subject of interest for many years due to its significant role in various fields in industry. Therefore, a lot of researches have been conducted towards the analysis of large deflection plates since the end of the nineteenth century. Among them, the most seminal work should be credited to von Karman who first developed in the nonlinear theory governing the moderately large deflection of plates in 1910. Nowadays, most of the researches dealing with large deflection plates are based on von Karman Equations. Ingenious ways of using von Karman's nonlinear theory in an updated Lagrangian corotational frame, for analyzing large rotations, and large deformation of plates and shells, have been proposed by Cai, Paik and Atluri (2009a, 2009b, 2010a, 2010b) and Zhu, Cai, Paik and Atluri (2010). In von Karman's theory, the large deflection behavior of plates with initial deflection is described by two nonlinear PDEs which are notoriously difficult to solve. In general, the exact analytical solution of PDEs are possible only in the simplest geometrical domains, and only mostly for linear problems [Atluri 2002]. Therefore, for solving the von Karman PDEs, researchers turn to the numerical methods.

The finite element method (FEM) originated from the need for solving complex elasticity and structural analysis problems in civil and aeronautical engineering. Its development can be traced back to the work by Alexander Hrennikoff (1941) and Richard Courant (1942). The FEM proved to be a powerful tool in structural analysis and many types of elements are available for the analysis of the behavior of plates. Its core characteristic is to mesh a continuous domain into a set of discrete elements. Hence, a continuous problem will mostly be replaced by a discrete problem whose solution is known to approximate that of the continuous problem. For nonlinear problems, such as the von Karman nonlinear theory of plates, it is common to develop the tangent-stiffness finite element method, based on local trial function in each element, using the incremental form of the symmetric Galerkin weak-form. The tangent-stiffness equations of the nonlinear plate theory are solved by using Newton-Raphson iteration scheme for each incremental displacement state, which is only quadratically convergent. Moreover, the Newton method involves the expensive process of inverting the tangent-stiffness at each iteration in each increment.

To avoid the expensive effort due to solving such a large set of equations as in the finite element method, an incremental Galerkin method was first proposed by

Ueda, Rashed and Paik (1987), and applied by Paik, Thayamballi, Lee and Kang (2001), Paik and Lee (2005). In the incremental global Galerkin method, instead of solving the von Karman PDEs directly, an incremental form of governing differential equations is derived. The derived PDEs are a set of piecewise linear partial differential equations. Therefore, upon applying the global Galerkin method to the incremental form of governing differential equations, a set of linear system of simultaneous equations will be obtained. This incremental global Galerkin method naturally leads to a tangent-stiffness matrix which is in general densely populated [as opposed to the sparsely populated tangent-stiffness matrix of the plates, based on the finite element method], but the matrix is of a much smaller size than that in FEM. However, the solution of the nonlinear plate problem, using the incremental global Galerkin method of Ueda, Rashed and Paik (1987) also involves a Newton-Raphson iteration, and the inversion of the tangent-stiffness matrix at each time and is only quadratically convergent.

Unlike the above methods, in the present paper the global Galerkin method is applied directly to the nonlinear PDEs to derive a system of third order coupled NAEs. As a contribution of this study, we solve the resultant NAEs in each load step by the exponentially convergent scalar homotopy algorithm. In general, the resultant NAEs is hard to solve. Firstly, one has to find the one physical solution among the multiple solutions. Therefore, a suitable initial guess is required to lead to the real solution. To keep track of the physical solution, we will solve the sets of NAEs corresponding to gradually increased loads, and take the solution of the last load step as the initial guess for the current NAEs under the current loads. Secondly, the size of NAEs grows large dramatically, with the increase of the number of terms of the deflection function. However, there are few tools to solve such a large system of NAEs directly. The most well-known Newton method suffers from its sensitivity to initial guess and expensiveness for calculating the inverse of the Jacobian matrix at each iteration step. Recently, four algorithms are developed to efficiently deal with the NAEs without calculating the inverse of the Jacobian matrix. They are the fictitious time integration method (FTIM) [Liu and Atluri (2008)], the modified Newton method [Atluri, Liu and Kuo (2009)], the scalar homotopy method (SHM) [Liu, Yeh, Kuo and Atluri (2009)] and the ECSHA [Liu, Ku, Yeh, Fan and Atluri (2010)]. In this study, the ECSHA is applied to transform the NAEs to ordinary differential equations (ODEs) and then the ODEs are numerically integrated by Euler method to find the original solution of the NAEs. In addition, an acceleration technique is proposed to speed up the convergence. Finally, numerical examples are employed to demonstrate the feasibility of the present direct global nonlinear Galerkin method and the efficiency of the ECSHA for solving the NAEs.

2 Governing differential equations of plates and the global nonlinear Galerkin method

Table 1: Notations

a	length of the plate
b	width of the plate
t	thickness of the plate
α	aspect ratio a/b
E	Young's modulus
ν	Poisson's ratio
$D = \frac{Et^3}{12(1-\nu^2)}$	plate bending rigidity
w	added deflection of the plate
w_0	initial deflection of the plate
F	Airy stress function
M	assumed half wave number in the x direction
N	assumed half wave number in the y direction
P_x	compression force in the x direction
P_y	compression force in the y direction
M_x	in-plane bending moment in the x direction
M_y	in-plane bending moment in the y direction
τ	shear stress
Q	lateral pressure
σ_{rx}	residual stress in the x direction
σ_{ry}	residual stress in the y direction

The elastic large deflection response of a plate with initial deflection is governed by two PDEs, which are named von Karman equations. One of them represents the equilibrium condition in the transverse direction, and the other represents the compatibility condition of in-plane strains. The PDEs are as follows:

$$\begin{aligned} \varphi = D \nabla^4 w - t \left[\frac{\partial^2 F}{\partial y^2} \frac{\partial^2 (w + w_0)}{\partial x^2} + \frac{\partial^2 F}{\partial x^2} \frac{\partial^2 (w + w_0)}{\partial y^2} - 2 \frac{\partial^2 F}{\partial x \partial y} \frac{\partial^2 (w + w_0)}{\partial x \partial y} \right] - Q \\ = 0 \end{aligned} \quad (1)$$

$$\nabla^4 F = E \left[\left(\frac{\partial^2 w}{\partial x \partial y} \right)^2 - \frac{\partial^2 w}{\partial x^2} \frac{\partial^2 w}{\partial y^2} + 2 \frac{\partial^2 w_0}{\partial x \partial y} \frac{\partial^2 w}{\partial x \partial y} - \frac{\partial^2 w_0}{\partial x^2} \frac{\partial^2 w}{\partial y^2} - \frac{\partial^2 w}{\partial x^2} \frac{\partial^2 w_0}{\partial y^2} \right] \quad (2)$$

In the above, w_0 is the given initial transverse displacement, w is the additional transverse displacement, and F is the Airy stress function governing the in plane stress resultants. In solving the above PDEs by the direct nonlinear global Galerkin method for capturing elastic large deflections of a simply supported plate, the added deflection w due to the applied load, and the initial deflection w_0 should satisfy the boundary conditions at four edges. In particular, the boundary conditions are as follows:

$$\begin{aligned} w = 0, \quad \frac{\partial^2 w}{\partial y^2} + \nu \frac{\partial^2 w}{\partial x^2} &= 0, \quad \text{at } y = 0, \text{ and } y = b \\ w = 0, \quad \frac{\partial^2 w}{\partial x^2} + \nu \frac{\partial^2 w}{\partial y^2} &= 0, \quad \text{at } x = 0, \text{ and } x = a \end{aligned} \quad (3)$$

To satisfy the boundary conditions, the added deflection function w and the initial deflection w_0 can be assumed in Fourier series,

$$w_0 = \sum_{m=1}^M \sum_{n=1}^N A_{0mn} \sin\left(\frac{m\pi x}{a}\right) \sin\left(\frac{n\pi y}{b}\right) \quad (4)$$

$$w = \sum_{m=1}^M \sum_{n=1}^N A_{mn} \sin\left(\frac{m\pi x}{a}\right) \sin\left(\frac{n\pi y}{b}\right) \quad (5)$$

Where, A_{mn} and A_{0mn} are the unknown and the known coefficients, respectively. The conditions of the combined loads, namely, bi-axial loads, bi-axial in-plane bending and edge shear are given as follows:

$$\begin{aligned} \int_0^b \frac{\partial^2 F}{\partial y^2} t dy &= P_x, \quad \int_0^b \frac{\partial^2 F}{\partial y^2} t \left(y - \frac{b}{2}\right) dy = M_x \quad \text{at } x = 0, \text{ and } x = a \\ \int_0^a \frac{\partial^2 F}{\partial x^2} t dx &= P_y, \quad \int_0^a \frac{\partial^2 F}{\partial x^2} t \left(x - \frac{a}{2}\right) dx = M_y \quad \text{at } y = 0, \text{ and } y = b \\ \frac{\partial^2 F}{\partial x \partial y} &= -\tau, \quad \text{at four edges} \end{aligned} \quad (6)$$

Then the homogenous solution F_h for the Airy stress function F should satisfy the condition of the combined loads acting on the plate. Considering the loading conditions, we can easily find F_h , by assuming F_h as cube polynomials in x and y . Substituting F_h into Eq. (6) we can obtain,

$$F_h = -P_x \frac{y^2}{2bt} - \sigma_{rx} \frac{y^2}{2} - P_y \frac{x^2}{2at} - \sigma_{ry} \frac{x^2}{2} - M_x \frac{y^2(2y-3b)}{b^3t} - M_y \frac{x^2(2x-3a)}{a^3t} - \tau_{xy} xy \quad (7)$$

For simplicity, the following notations are introduced to abbreviate the expressions involving the sine or cosine terms,

$$\begin{aligned}\sin\left(\frac{m\pi x}{a}\right) &= sx(m), \quad \cos\left(\frac{m\pi x}{a}\right) = cx(m) \\ \sin\left(\frac{n\pi y}{b}\right) &= sy(n), \quad \cos\left(\frac{n\pi y}{b}\right) = cy(n)\end{aligned}\quad (8)$$

To find the particular solution F_p , which should satisfy Eq. (2), one can substitute w and w_0 into the right side of Eq. (2), thus obtaining:

$$\begin{aligned}\nabla^4 F_p = & \frac{E\pi^4}{4a^2b^2} \sum_{m=1}^M \sum_{n=1}^N \sum_{k=1}^K \sum_{l=1}^L \{ [A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2] cx(m+k)cy(n+l) \\ & + [A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2] cx(m+k)cy(n-l) \\ & + [A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2] cx(m-k)cy(n+l) \\ & + [A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2] cx(m-k)cy(n-l) \}\end{aligned}\quad (9)$$

Consequently, the particular solution F_p for the Airy stress function can be written in the following way,

$$\begin{aligned}F_p = & \sum_{m=1}^M \sum_{n=1}^N \sum_{k=1}^K \sum_{l=1}^L \{ B_1(m,n,k,l) \times cx(m+k)cy(n+l) \\ & + B_2(m,n,k,l) \times cx(m+k)cy(n-l) \\ & + B_3(m,n,k,l) \times cx(m-k)cy(n+l) \\ & + B_4(m,n,k,l) \times cx(m-k)cy(n-l) \}\end{aligned}\quad (10)$$

Upon substituting F_p into the Eq. (2), the coefficients B_1 , B_2 , B_3 and B_4 are obtained as

$$\begin{aligned}B_1(m,n,k,l) &= \frac{E\alpha^2}{4} \times \frac{A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2}{[(m+k)^2 + (n+l)^2]^2} \\ B_2(m,n,k,l) &= \frac{E\alpha^2}{4} \times \frac{A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2}{[(m+k)^2 + (n-l)^2]^2} \\ B_3(m,n,k,l) &= \frac{E\alpha^2}{4} \times \frac{A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2}{[(m-k)^2 + (n+l)^2]^2} \\ B_4(m,n,k,l) &= \frac{E\alpha^2}{4} \times \frac{A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2}{[(m-k)^2 + (n-l)^2]^2}\end{aligned}\quad (11)$$

Inserting B_1, B_2, B_3 and B_4 in Eq. (10) , we obtain:

$$\begin{aligned}
 F_p = & \frac{E\alpha^2}{4} \\
 & \sum_{m=1}^M \sum_{n=1}^N \sum_{k=1}^K \sum_{l=1}^L \left\{ \frac{A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2}{[(m+k)^2 + (n+l)^2]^2} \times cx(m+k)cy(n+l) \right. \\
 & + \frac{A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2}{[(m+k)^2 + (n-l)^2]^2} \times cx(m+k)cy(n-l) \\
 & + \frac{A_{mn}A_{kl}ml(nk + ml) + A_{kl}A_{0mn}(nk + ml)^2}{[(m-k)^2 + (n+l)^2]^2} \times cx(m-k)cy(n+l) \\
 & \left. + \frac{A_{mn}A_{kl}ml(nk - ml) - A_{kl}A_{0mn}(nk - ml)^2}{[(m-k)^2 + (n-l)^2]^2} \times cx(m-k)cy(n-l) \right\}
 \end{aligned} \tag{12}$$

Then, the Airy stress function F can be obtained by

$$F = F_h + F_p \tag{13}$$

It is evident from Eq. (7), Eq. (12) and Eq. (13) that F is a second order function with regard to the unknown deflection coefficients A_{mn} . To compute the unknown coefficients A_{mn} , the global Galerkin method is applied to the equilibrium Eq. (1),

$$\iiint_V \phi(x, y, z) s_x(i) s_y(j) dx dy dz = 0, \quad i = 1, 2, 3 \dots \quad j = 1, 2, 3 \dots \tag{14}$$

Upon substituting Eq. (13) into Eq. (1), and then Eq. (1) to Eq. (14) after a lengthy derivation, we obtain a system of third order coupled NAEs, with respect to the

unknown coefficients A_{mn} , the expression of the derived NAEs is

$$\begin{aligned}
& \sum_{m=1}^M \sum_{n=1}^N A_{mn} \times D\pi^4 \left(\frac{m^2}{a^2} + \frac{n^2}{b^2} \right)^2 H_{01}(i, j, m, n) \\
& + \sum_{m=1}^M \sum_{n=1}^N \sum_{k=1}^K \sum_{l=1}^L \sum_{r=1}^R \sum_{s=1}^S A_{mn} A_{kl} A_{rs} \times (-t) \\
& \frac{E\alpha^2\pi^4}{4a^2b^2} (H_1 + H_2 + H_3 + H_4 - 2H_9 - 2H_{10} - 2H_{11} - 2H_{12}) \\
& + \sum_{m=1}^M \sum_{n=1}^N \sum_{k=1}^K \sum_{l=1}^L A_{mn} A_{kl} \times (-t) \\
& \frac{E\alpha^2\pi^4}{4a^2b^2} \sum_{r=1}^R \sum_{s=1}^S A_{0rs} (H_1 + H_2 + H_3 + H_4 - 2H_9 - 2H_{10} - 2H_{11} - 2H_{12}) \\
& + \sum_{k=1}^K \sum_{l=1}^L \sum_{r=1}^R \sum_{s=1}^S A_{kl} A_{rs} \times (-t) \\
& \frac{E\alpha^2\pi^4}{4a^2b^2} \sum_{m=1}^M \sum_{n=1}^N A_{0mn} (H_6 + H_7 - H_5 - H_8 + 2H_{13} - 2H_{14} - 2H_{15} + 2H_{16}) \\
& + \sum_{k=1}^K \sum_{l=1}^L A_{kl} \times (-t) \\
& \frac{E\alpha^2\pi^4}{4a^2b^2} \sum_{m=1}^M \sum_{n=1}^N \sum_{r=1}^R \sum_{s=1}^S A_{0mn} A_{0rs} (H_6 + H_7 - H_5 - H_8 + 2H_{13} - 2H_{14} - 2H_{15} + 2H_{16}) \\
& + \sum_{m=1}^M \sum_{n=1}^N A_{mn} \times (-t) \\
& \left\{ \frac{m^2\pi^2}{a^2} \left[\left(\frac{P_x}{bt} + \sigma_{rx} - \frac{6}{b^2t} M_x \right) H_{01}(i, j, m, n) + \frac{12}{b^3t} M_x H_{03}(i, j, m, n) \right] \right. \\
& + \frac{n^2\pi^2}{b^2} \left[\left(\frac{P_y}{at} + \sigma_{ry} - \frac{6}{a^2t} M_y \right) H_{01}(i, j, m, n) + \frac{12}{a^3t} M_y H_{02}(i, j, m, n) \right] \\
& \left. + \frac{2\tau\pi^2}{ab} mn \times H_{04}(i, j, m, n) \right\}
\end{aligned}$$

$$\begin{aligned}
& + \sum_{m=1}^M \sum_{n=1}^N A_{0mn} \times (-t) \left\{ \frac{m^2 \pi^2}{a^2} \right. \\
& \left[\left(\frac{P_x}{bt} + \sigma_{rx} - \frac{6}{b^2 t} M_x \right) H_{01}(i, j, m, n) + \frac{12}{b^3 t} M_x H_{03}(i, j, m, n) \right] \\
& + \frac{n^2 \pi^2}{b^2} \left[\left(\frac{P_y}{at} + \sigma_{ry} - \frac{6}{a^2 t} M_y \right) H_{01}(i, j, m, n) + \frac{12}{a^3 t} M_y H_{02}(i, j, m, n) \right] \\
& + \frac{2\tau \pi^2}{ab} mn \times H_{04}(i, j, m, n) \left. \vphantom{\sum_{m=1}^M \sum_{n=1}^N} \right\} \\
& - Q \times H_{00}(i, j) = 0
\end{aligned} \tag{15}$$

Where, for simplicity, the coefficient matrix $H_1(i, j, m, n, k, l, r, s)$ is denoted by H_1 and so forth. All the coefficient matrices can be obtained by performing integration over the whole volume of the plate. We can write the Eq. (15) in a matrix form,

$$[K_f]_{MN \times MN} A_f + [K_s]_{MN \times (MN)^2} A_s + [K_t]_{MN \times (MN)^3} A_t + [C]_{MN \times 1} = 0 \tag{16}$$

Where $[C]_{MN \times 1}$ is the constant column matrix, $[K_f]_{MN \times MN}$, $[K_s]_{MN \times (MN)^2}$ and $[K_t]_{MN \times (MN)^3}$ are the first order, second order, third order coefficient matrices, respectively, with their subscripts being their dimensions. A_f, A_s, A_t are the first order, second order and third order unknown vectors, respectively. The exact descriptions of the matrices and vectors in Eq. (16) are given in the Appendix.

We can see from Eq. (16) that the number of nonlinear terms of the NAEs becomes larger dramatically with the increase of deflection function terms $M \times N$. For instance, if we take $M = N = 2$, $M = N = 3$, $M = N = 4$ and $M = N = 5$ the number of third order terms in one equation is 64, 729, 4096 and 15625, respectively. Therefore, solving the system of third order simultaneous equations to solve for the coefficients A_{MN} normally requires a large amount of computational effort, especially when $M \times N$ are not small. Moreover, since the solution of each coefficient should be unique, one will have to construct a suitable initial guess for the NAEs to find the one physical solution among the multiple solutions. Because of these two reasons, it has hitherto been considered to be an impossible task to solve such a set of highly nonlinear third order simultaneous equations [Paik, Thayamballi, Lee and Kang 2001].

In section 3, the exponentially convergent scalar homotopy algorithm is introduced, which can be used to solve a large set of NAEs. In section 4, approaches for providing the proper initial guess to directly solve the highly nonlinear algebraic equations are discussed.

3 An Exponentially Convergent Scalar Homotopy Algorithm

The ECSHA, which is first proposed by Liu, Ku, Yeih, Fan and Atluri (2010), is based on the construction of a scalar homotopy function to transform a vector function into a time-dependent scalar function by introducing a fictitious time-like variable. Taking advantage of the time-dependent scalar function, the proposed ECSHA does not need to calculate the inverse of the Jacobian matrix at every iteration step, such that it can greatly reduce the cost of the computational time. Moreover, the ECSHA can solve a large class of NAEs effectively and is insensitive to the initial guess demonstrated by Liu, Ku, Yeih, Fan and Atluri (2010). To begin with, we consider the following NAEs:

$$\mathbf{F}(\mathbf{x}) = \mathbf{0}, \quad (17)$$

where $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, and $\mathbf{F} = (F_1, F_2, \dots, F_n)^T$.

Traditionally, the Newton's method for solving these NAEs is given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}^{-1}(\mathbf{x}^k) \mathbf{F}(\mathbf{x}^k) \quad (18)$$

Where $\mathbf{B}$ denotes the Jacobian matrix of $\mathbf{F}(\mathbf{x})$, and $\mathbf{x}^{k+1}$ is the $(k+1)th$ iteration for $\mathbf{x}$. Newton's method has an advantage, in that it is quadratically convergent. However, its convergence depends on the initial guess of the solution. If the initial guess is beyond the attracting zone, the Newton's method fails. In addition, Newton's method is numerically expensive to compute the inverse of the Jacobian matrix at every iteration step.

Many contributions have been made to avoid the shortcomings of Newton's method. Davidenko (1953) first developed a homotopy method to solve NAEs by numerically integrating $\dot{\mathbf{x}}(t) = -\mathbf{H}_x^{-1} \mathbf{H}_t(\mathbf{x}, t)$, $\mathbf{x}(0) = \mathbf{a}$, where $\mathbf{H}$ is a vector homotopy function. Thus, it is called a vector homotopy method. This vector homotopy method is global convergent. However, it suffers a slow convergence speed due to the inverse of matrix and a required small time step.

To take advantage of the global convergence of the homotopy method and also to avoid computing the inverse of the Jacobian matrix, the scalar homotopy method (SHM), was developed by Liu, Yeih, Kuo and Atluri (2009). In their study, instead of using a vector function, they introduced a scalar function

$$h(\mathbf{x}, t) = \frac{1}{2} \left[t \|\mathbf{F}(\mathbf{x})\|^2 - (1-t) \|\mathbf{x} - \mathbf{a}\|^2 \right] = 0 \quad (19)$$

as an auxiliary function. The scalar homotopy method basically aims to construct a path from the solution of the auxiliary scalar function to the solution of the desired function continuously. The SHM shows many merits to deal with a variety of

engineering problems [Liu, Yeih, Kuo and Atluri (2009), Fan, Liu, Yeih and Chan (2010)]. Furthermore, the exponentially convergent scalar homotopy algorithm developed by Liu, Ku, Yeih and Atluri (2010) shows a better performance in solving a large system of NAEs. To be different from the former SHM, the ECSHA is based on a Newton scalar homotopy function

$$h(\mathbf{x}, t) = \frac{1}{2}Q(t) \|\mathbf{F}(\mathbf{x})\|^2 - \frac{1}{2} \|\mathbf{F}(\mathbf{x}_0)\|^2 = 0 \quad (20)$$

Where $Q(t)$ is a monotonically increasing function of t , and $Q(0) = 1$, $Q(\infty) = \infty$. Considering the consistency condition, the derivative of $h(\mathbf{x}, t)$ with respect to t should vanish, that is

$$\frac{dh}{dt} = \frac{\partial h}{\partial t} + \frac{\partial h}{\partial \mathbf{x}} \cdot \frac{d\mathbf{x}}{dt} = 0 \quad (21)$$

By solving the above equation, we obtain

$$\dot{\mathbf{x}} = - \frac{\frac{\partial h}{\partial t}}{\left\| \frac{\partial h}{\partial \mathbf{x}} \right\|^2} \frac{\partial h}{\partial \mathbf{x}} \quad (22)$$

The derivatives of $h(\mathbf{x}, t)$ with respect to $\mathbf{x}$ and t are

$$\frac{\partial h}{\partial t} = \frac{1}{2} \dot{Q}(t) \|\mathbf{F}(\mathbf{x})\|^2 \quad (23)$$

$$\frac{\partial h}{\partial \mathbf{x}} = Q(t) \mathbf{B}^T \mathbf{F}(\mathbf{x}) \quad (24)$$

By substituting Eq. (22), Eq. (23) and Eq. (24) to Eq. (21), we obtain,

$$\dot{\mathbf{x}} = \frac{\dot{Q}(t)}{2Q(t)} \frac{\|\mathbf{F}(\mathbf{x})\|^2}{\|\mathbf{B}^T \mathbf{F}(\mathbf{x})\|^2} \mathbf{B}^T \mathbf{F}(\mathbf{x}) \quad (25)$$

In the Eq. (25), there are many choices for the monotonically increasing function $Q(t)$, in the study of Liu, Ku, Yeih and Atluri (2010), they let

$$\frac{\dot{Q}(t)}{Q(t)} = - \frac{v}{(1+t)^m}, \quad 0 < m \leq 1 \quad (26)$$

Hence,

$$Q(t) = e^{\frac{v}{1-m} [(1+t)^{1-m} - 1]} \quad (27)$$

Finally, we derive that

$$\dot{\mathbf{x}} = \frac{-v}{2(1+t)^m} \frac{\|\mathbf{F}(\mathbf{x})\|^2}{\|\mathbf{B}^T \mathbf{F}(\mathbf{x})\|^2} \mathbf{B}^T \mathbf{F}(\mathbf{x}) \quad (28)$$

Where, v is the damping constant of ECSHA, t is the fictitious time, m is a constant related to the convergence speed. Virtually, the ECSHA transforms the target NAEs into an equivalent system of first order ODEs as Eq. (28) shows. To solve the resultant ODEs, we can use several numerical integration methods such as Euler method, Runge-Kutta approach and Group Preserving Scheme [Liu and Atluri (2008)]. In this study, a forward Euler scheme is employed to perform the integration, and the following equation is obtained:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{hv}{2(1+t)^m} \frac{\|\mathbf{F}(\mathbf{x}^k)\|^2}{\|\mathbf{B}^T \mathbf{F}(\mathbf{x}^k)\|^2} \mathbf{B}^T \mathbf{F}(\mathbf{x}^k) \quad (29)$$

Where h is the fictitious time step for the fictitious time t . In Eq. (29), we can see that one need not to invert the Jacobian matrix at all.

4 Selection of the Initial Guess Solution, and an Acceleration Technique

4.1 Initial guess selection

When an iterative method is employed to solve the NAEs, the initial guess of the solution is of great importance. In general, when an initial guess is in the vicinity of a solution, it may significantly reduce the number of iterations and also avoid deviating from the current solution. Consider a simple case, a rectangular plate subjected to uniaxial compression load P . The Eq. (16) is its governing equations.

When P is small compared with P_{cr} , the linear terms of Eq. (16) play a dominate role in the whole equation since the deflection is small and the nonlinear terms can be quite small. Based on this observation, one can throw off the nonlinear terms in Eq. (16), and solve the linear part of the NAEs quite easily. Intuitively, the solution of the linear equations is taken as a reasonable initial guess for the NAEs, when the applied loads are small. However, with the increase of P , the nonlinear terms grow large quickly. When it reaches a certain level, the magnitude of the nonlinear terms becomes comparable to that of the linear terms. Thus, the solution of the linear equations may not be a good initial guess any more. Therefore, this approach fails when the plate deflects finitely.

Another approach to construct a proper initial guess for the NAEs is to take the solution of the last load step as the initial guess of the current step when the two

loads are reasonably close to each other. For instance, we can use the solution of the NAEs with load P as the initial guess for the NAEs with load $P + \Delta P$ where ΔP is relatively small compared with P . It makes sense since a small change of the load will results in a small change of the deflection, thus, a small difference between the solutions. This approach makes use of the approximation between solutions of two close loads. Theoretically, this load-tracking approach is applicable to any situation when the plate deflects finitely so we use this approach to keep track of the physical solution in each load step in the numerical illustrations. Although we take the compression load as an example, this approach still makes sense when the plate is subjected to a combination of loading conditions.

4.2 Acceleration technique

Although the load-tracking approach guarantees that when the load increases gradually we can keep track of the physical solution, the computing effort may be very expensive especially when the number of terms of the deflection function is relatively large. Therefore, an acceleration technique is proposed to speed up the convergence of the ECSHA. The motivation of the acceleration technique is to make the initial guess in a close vicinity of the physical solution by using the approximation between the solutions of the cases with $M \times N$ terms and $(M + 1) \times (N + 1)$ terms under the same load.

Let the deflection function with $M \times N$ terms and $(M + 1) \times (N + 1)$ terms be $w_{(M,N)}$, $w_{(M+1,N+1)}$, respectively. Both $w_{(M,N)}$ and $w_{(M+1,N+1)}$ represent the added deflection of a plate under the same load. Physically, $w_{(M,N)}$ should closely approximate $w_{(M+1,N+1)}$. Simply says, $w_{(M+1,N+1)} \approx w_{(M,N)}$. According to Eq. (5), we can express the relationship between the solution of $M \times N$ terms and that of $(M + 1) \times (N + 1)$ terms in a matrix form,

$$\begin{pmatrix} A_{11}^{(M+1,N+1)} & A_{12}^{(M+1,N+1)} & \dots & A_{1,N}^{(M+1,N+1)} & A_{1,N+1}^{(M+1,N+1)} \\ A_{21}^{(M+1,N+1)} & A_{22}^{(M+1,N+1)} & \dots & A_{2,N}^{(M+1,N+1)} & A_{2,N+1}^{(M+1,N+1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{M,1}^{(M+1,N+1)} & A_{M,2}^{(M+1,N+1)} & \dots & A_{M,N}^{(M+1,N+1)} & A_{M,N+1}^{(M+1,N+1)} \\ A_{M+1,1}^{(M+1,N+1)} & A_{M+1,2}^{(M+1,N+1)} & \dots & A_{M+1,N}^{(M+1,N+1)} & A_{M+1,N+1}^{(M+1,N+1)} \end{pmatrix} \approx \begin{pmatrix} A_{11}^{(M,N)} & A_{12}^{(M,N)} & \dots & A_{1,N}^{(M,N)} & 0 \\ A_{21}^{(M,N)} & A_{22}^{(M,N)} & \dots & A_{2,N}^{(M,N)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{M,1}^{(M,N)} & A_{M,2}^{(M,N)} & \dots & A_{M,N}^{(M,N)} & 0 \\ 0 & 0 & \dots & 0 & 0 \end{pmatrix} \quad (30)$$

If we arrange the coefficients in a vector form, the above Eq. (30) indicates that

$$\left[A_{11}^{(M,N)}, A_{12}^{(M,N)}, \dots, A_{1,N}^{(M,N)}, 0, A_{21}^{(M,N)}, A_{22}^{(M,N)}, \dots, A_{2,N}^{(M,N)}, 0, \dots, 0, 0, \dots, 0 \right]^T$$

is a reasonable guess of

$$\left[A_{11}^{(M+1,N+1)}, A_{12}^{(M+1,N+1)}, \dots, A_{1,N+1}^{(M+1,N+1)}, A_{21}^{(M+1,N+1)}, A_{22}^{(M+1,N+1)}, \dots, A_{2,N+1}^{(M+1,N+1)}, \dots, A_{M+1,1}^{(M+1,N+1)}, A_{M+1,2}^{(M+1,N+1)}, \dots, A_{M+1,N+1}^{(M+1,N+1)} \right]^T$$

Where, $A_{i,j}^{(M,N)}$ ($i = 1, 2, \dots, M; j = 1, 2, \dots, N$) and $A_{i,j}^{(M+1,N+1)}$ ($i = 1, 2, \dots, M+1; j = 1, 2, \dots, N+1$) are the unknown coefficients for the trial functions with $M \times N$ and $(M+1) \times (N+1)$ terms, respectively.

In practical applications, we employ the load-tracking approach to provide the initial guess for the NAEs with $M \times N$ terms to keep track of the physical solutions. The acceleration technique can be carried out for the case with $(M+1) \times (N+1)$ terms to speed up the convergence.

5 Numerical illustrations

In this section, several numerical examples are provided to demonstrate the validity of the proposed scheme, which is applying the global Galerkin method directly to the highly nonlinear PDEs and directly solving the resultant NAEs at every load step, by analyzing the large deflection of a simply supported rectangular plate subjected to different loading conditions. In addition, the efficiency of the ECSHA for solving a large system of NAEs is investigated. Besides, the effectiveness of the acceleration technique is confirmed. The Young's modulus and Poisson's ratio are assumed to be $E = 205.8 \text{ GPa}$ and $\nu = 0.3$, respectively for all examples. For applying the ECSHA, the parameters h, m, ν are set to be 1, 0.01 and 2 respectively. The parameters may influence the convergence property and numerical stability of the ECSHA. However, they are not our concerns in this study. To further understand these parameters, one can refer to the paper by Liu, Ku, Yeih, Fan and Atluri (2010).

5.1 A square plate under uniaxial compression

In this example, a simply supported square plate under uniaxial compression is analyzed. The dimensions of this plate are $a = 1, b = 1, t = 0.009$, where a, b, t represent length, width and thickness respectively. All dimensions in this study are

in metres unless otherwise mentioned. According to Eq. (4) and Eq. (5), the initial deflection is assumed to consists of $M \times N$ terms,

$$w_0 = \sum_{m=1}^M \sum_{n=1}^N A_{0mn} s x(m) s y(n)$$

Where, A_{0mn} is the known coefficients with A_{011} being 0.45×10^{-3} and other elements being zeros. The deflection function with $M \times N$ terms is,

$$w = \sum_{m=1}^M \sum_{n=1}^N A_{mn} s x(m) s y(n)$$

The global Galerkin method is applied to deal with five cases wherein the deflection functions are assumed with 1×1 term, 2×2 terms, 3×3 terms, 4×4 terms and 5×5 terms, respectively. A case of the incremental global Galerkin method developed by Ueda, Rashed and Paik (1987) is cited to compare with the present global direct nonlinear Galerkin method. Figure 1 displays curves that plot the compression load against the maximum deflection of the plate. The compression load acting on the plate varies from 0 to 2 (P_{cr}) with load step being 0.1. Therefore, for each case, there are 21 load steps, hence 20 sets of NAEs to solve.

It may be seen from Figure 1 that the results of the present nonlinear global Galerkin method and the incremental global Galerkin method are in good agreement. Figure 1 also provides the comparison of the results of the present global nonlinear Galerkin method with different order trigonometric functions. We only plot three of the five cases for sake of visual clarity. We can see that all the three cases with 1×1 , 3×3 and 5×5 terms are in very good agreement. In detail, the three cases coincide with each other when load is under approximately 1.5. As load increases, the results of the cases with 3×3 and 5×5 terms still coincide while the case 1×1 begins to differ slightly from them. It indicates that the present global nonlinear Galerkin method works reasonably well even with few deflection function terms.

We see from Table 2 that the size of the NAEs becomes large dramatically with the increase of the number of terms of the deflection function. The ECSHA is employed to deal with the resultant NAEs and the load-tracking approach is adopted to provide the initial guess. For the case with 3×3 terms, the time for solving 20 sets of NAEs is 5969.17s (1.7h) in PC Core2. The time for solving the 4×4 and 5×5 are 224680.00s (62.4h) and 2375733.68s (659.9h), respectively.

Table 3 gives the comparison of the computational time for solving 3×3 , 4×4 , and 5×5 cases with and without acceleration technique. The results given confirm that the acceleration technique can speed up the convergence significantly especially when the size of the NAEs is large.

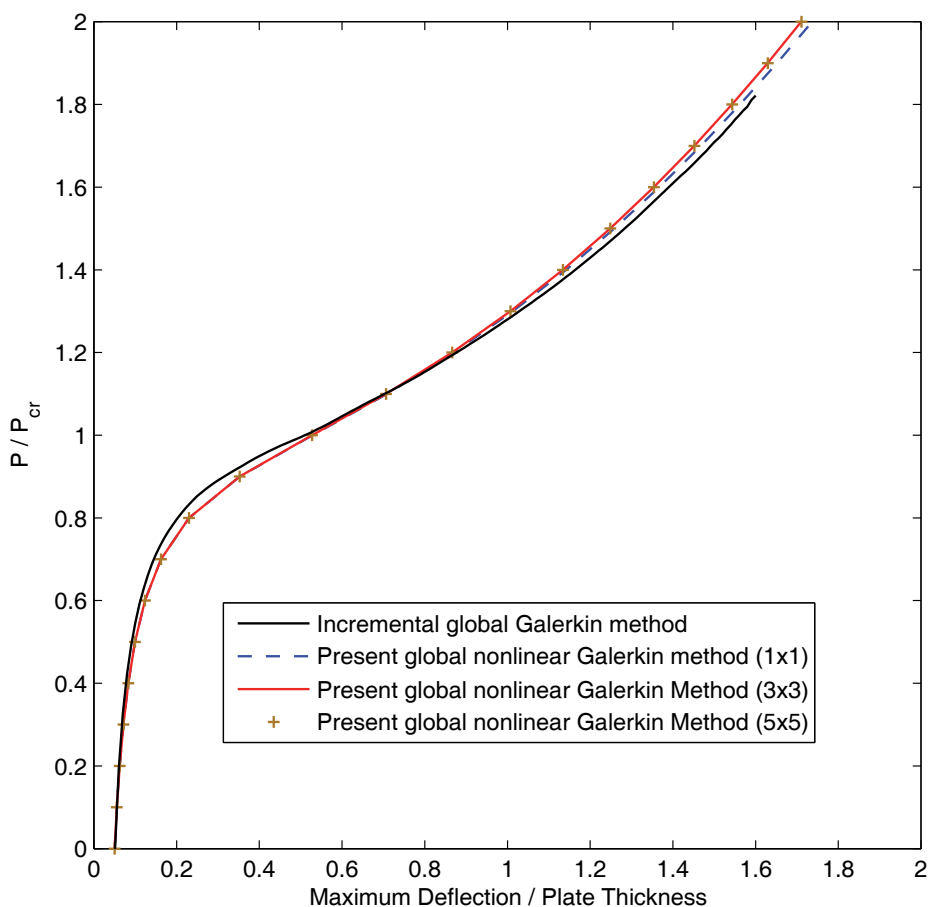


Figure 1: Comparison of the load-deflection curves for the present global nonlinear Galerkin method and the incremental global Galerkin method

5.2 A rectangular plate under uniaxial compression

A simply supported rectangular plate under uniaxial compression is considered. Its dimensions are $a = 1.68$, $b = 0.98$, $t = 0.011$. The pattern of the initial deflection and the deflection function is given by Eq. (4) and Eq. (5). Here $A_{0mn} = 0$ is taken except $A_{011} = 1.1 \times 10^{-3}$ and $A_{021} = 0.22 \times 10^{-3}$.

The present global nonlinear Galerkin method is applied to solve two cases wherein the deflection functions are assumed with 2×1 terms, 3×2 terms. For comparison the analysis is also carried out by the FEM using rectangular, four node, noncon-

Table 2: Sizes of the NAEs and the computational efforts without acceleration technique

Cases($M \times N$)	N_{eqs}	N_{3th}	ε	N_{it}	T
1×1	1	1	10^{-7}	106	8.38s
2×2	4	64	10^{-7}	3833	70.70s
3×3	9	729	10^{-5}	24302	5969.17s
4×4	16	4096	10^{-5}	83483	224680.00s
5×5	25	15625	10^{-3}	95094	2375733.68s
N_{eqs} is the number of equations; N_{3th} is the number of third order terms in one equation; ε is the convergence criterion; N_{it} is the number of total iterations; T is the computational time for solving 20 sets of NAEs.					

Table 3: Comparison of the computational time with and without acceleration

Cases($M \times N$)	T		
	without acceleration	acceleration	ratio
3×3	5969.17s	4192.13s	1.4 :1
4×4	224680.00s	5419.73s	41.5:1
5×5	2375733.68s	108720.75s	21.9:1
The computational time for that with acceleration technique is a summation of the current case and the last case. Because the current case is based on the result of the last one.			

forming plate elements with five degrees freedom at each node. 7×18 elements for half of the plate [Ueda, Rashed and Paik 1987]. Figure 2 displays curves that plot the compression load against the deflections of two points A and B whose positions are $(0.25a, 0.5b)$ and $(0.75a, 0.5b)$ respectively if we set the lower left corner of the plate $(0, 0)$ and upper right corner (a, b) .

It may be seen from Figure 2 that the results of the present global nonlinear Galerkin method and that of the tangent stiffness FEM are in good agreement. Figure 2 also indicates that the two cases of the present method with 2×1 and 3×2 terms agree well with each other. Table 4 provides the computational information of ECSHA for solving the NAEs. In summary, the results given confirm the accuracy and efficiency of the present scheme in the case of rectangular plates.

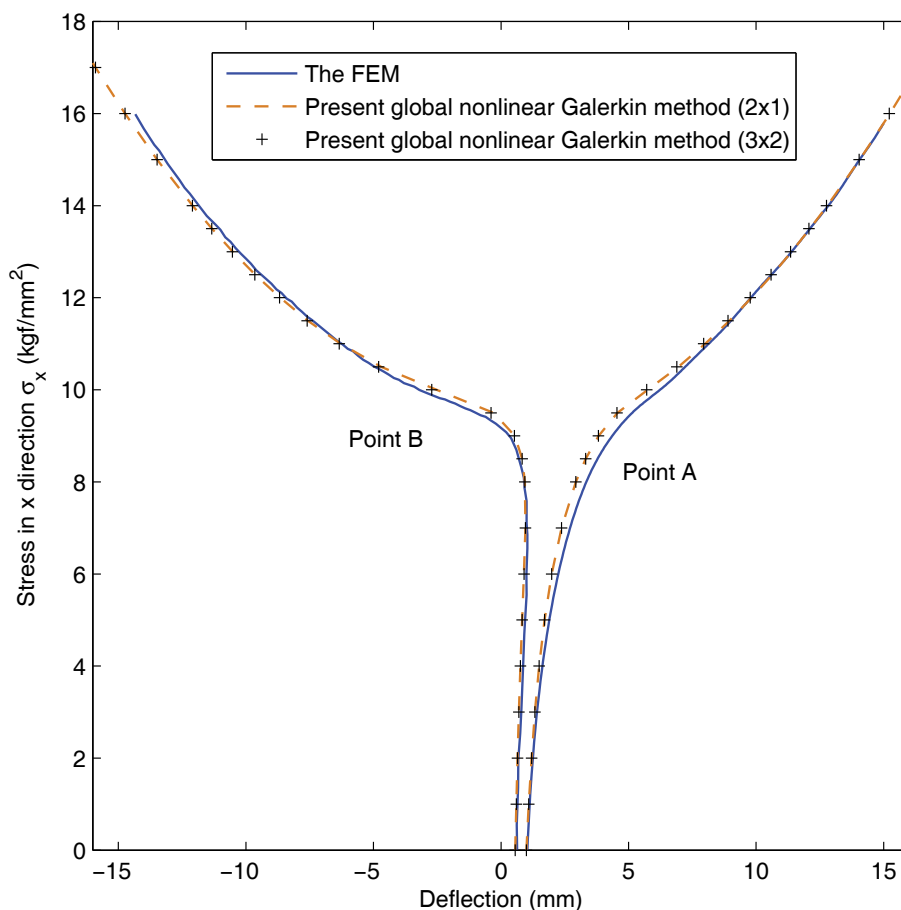


Figure 2: Comparison of the stress versus the deflection of point A and B for the global nonlinear Galerkin method and the finite element method

Table 4: Computational efforts without acceleration technique

Cases($M \times N$)	N_{eqs}	N_{3th}	ϵ	N_{it}	T
2×1	2	8	10^{-7}	595	2.41s
3×2	6	216	10^{-5}	14176	658.50s

5.3 A square plate subjected to lateral load

A square plate subjected to a uniformly distributed lateral load Q is considered in this example. Its dimensions are $a = 1$, $b = 1$, $t = 0.009$. The deflection function

is in the same pattern as before. The initial deflection is assumed to be zero such that $A_{0mn} = 0$. The present global nonlinear Galerkin method is applied to solve the case with 2×2 terms. It indicates in Figure 3 that the present method is quite accurate in the case of lateral load.

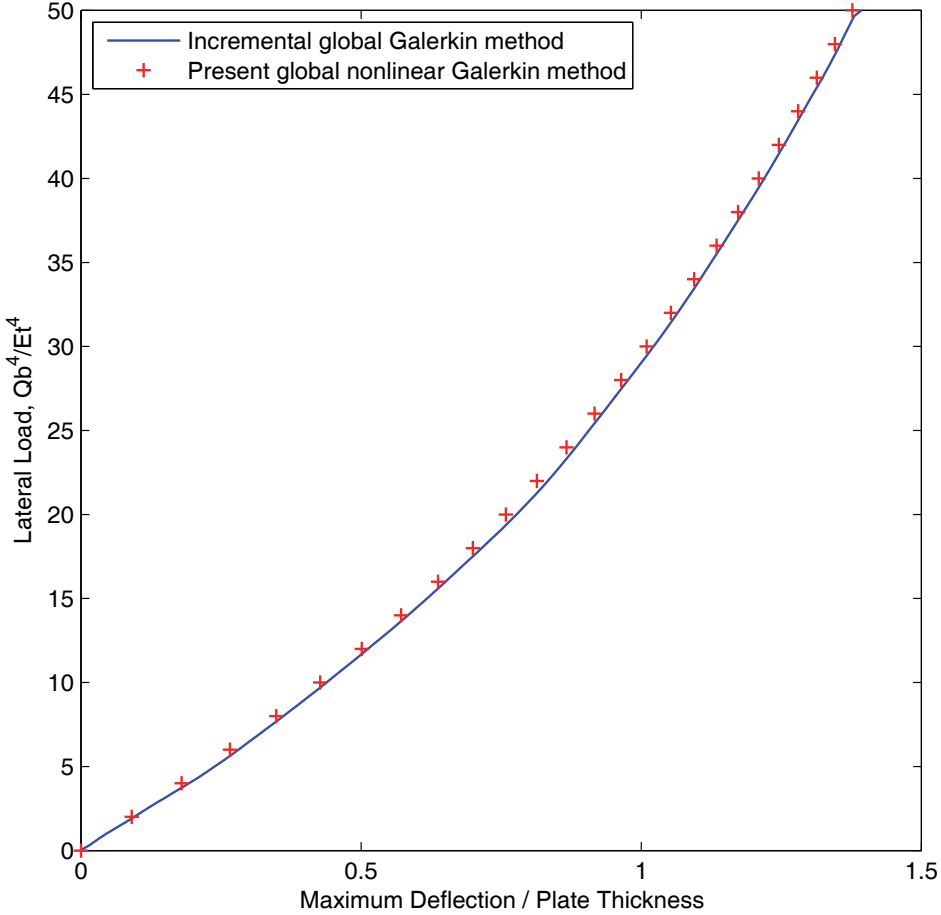


Figure 3: Comparison of the deflection of a square plate under uniform lateral load for the present method and the incremental global Galerkin method

5.4 A square plate subjected to lateral pressure combined with uniaxial compression

In this example, a square plate subjected to lateral pressure combined with uniaxial compression is considered. The compression load acting on the plate is a constant

value $0.6P_{cr}$. The lateral pressure acting on the plate changes as shown in the Figure 4. Its dimensions are $a = 1$, $b = 1$, $t = 0.02$. The deflection function is in the same form as the above examples. The initial deflection is assumed to be zero. In order to compare with the present method, this plate is analyzed by the FEM in ANSYS (brick element with 8 node and three degrees freedom at each node, $50 \times 50 \times 1$ elements for the whole volume). The present global nonlinear Galerkin method is applied to solve the case with 2×2 terms. Figure 4 shows the results obtained from the present method compared with the FEM. It is seen that the results of the present method are quite in accord with the FEM when the lateral load is below approximate 10. When lateral load becomes bigger, the discrepancy exists between the two methods.

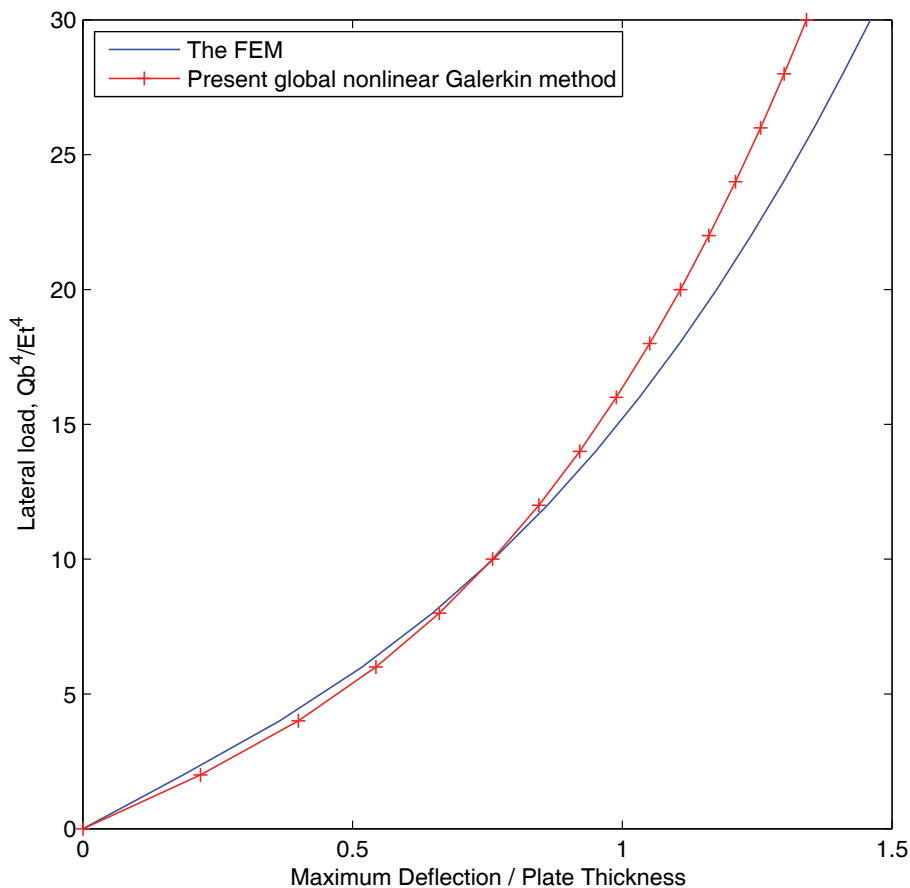


Figure 4: Comparison of the load-deflection curves by FEM and the present global nonlinear Galerkin method

6 Conclusions

The aim of this paper is to present a reasonably accurate and efficient scheme for analyzing the large deflection behavior of a simply-supported rectangular plate under a combination of biaxial compression/tension, biaxial in-plane bending, edge shear and lateral pressure loads. In this scheme, the global Galerkin method is applied directly to the governing highly nonlinear PDEs to derive a system of third order coupled NAEs. The external load is applied incrementally to the plate and the resultant NAEs are solved directly at every load increment. To guarantee the physical solution, the load-tracking approach is introduced to provide the initial guess to directly solving the NAEs for each load. To efficiently solve the large set of NAEs, the ECSHA is employed. In addition, an acceleration technique is proposed to speed up the convergence for solving a large system of NAEs. Four examples are used to verify the accuracy and feasibility of the present scheme for different plates under different loading conditions by comparing the results of the present global nonlinear Galerkin method with the incremental global Galerkin method and the FEM based on the incremental symmetric Galerkin weak-form and local trial functions. The present nonlinear global Galerkin method yields results which are in excellent agreement with the FEM tangent-stiffness method. However, the tangent-stiffness FEM requires degrees of freedom which are about two orders of magnitude larger in number than the number of coupled NAEs in the present nonlinear global Galerkin method. In addition, the presented examples also illustrate the efficiency of the ECSHA for solving the case with up to 5×5 terms, as well as the effectiveness of the introduced acceleration technique.

Acknowledgement: The first author gratefully acknowledges the support from the China Scholarship Council and the support from college of astronautics in NPU. At UCI, this research was supported by the Army Research Laboratory, with Drs. A. Ghoshal and Dy Le as the Program Officials. This research was also supported by the World Class University (WCU) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (Grant no.: R33-10049). The second author is also pleased to acknowledge the support of The Lloyd's Register Educational Trust (The LRET) which is an independent charity working to achieve advances in transportation, science, engineering and technology education, training and research worldwide for the benefit of all.

References

- Atluri, S. N.** (2002): *Methods of Computer Modeling in Engineering and Science*. Tech. Science Press, 1400 pages.
- Atluri, S. N.; Shen, S.** (2002): The meshless local Petrov-Galerkin (MLPG) method: a simple & less-costly alternative to finite element and boundary element methods. *CMES: Computer Modeling in Engineering & Sciences*, vol. 3, pp. 11-51.
- Atluri, S. N.; Liu, H. T.; Han, Z. D.** (2006): Meshless local Petrov-Galerkin (MLPG) mixed collocation method for elasticity problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 14, pp. 141-152.
- Cai Y. C.; Paik J. K.; Atluri S. N.** (2009): Large Deformation Analyses of Space-Frame Structures, with Members of arbitrary Cross-Section, Using Explicit Tangent Stiffness Matrices, Based on a von Karman Type Nonlinear Theory in Rotated Reference Frames. *CMES: Computer Modeling in Engineering & Sciences*, vol. 53, pp. 117-145.
- Cai Y. C.; Paik J. K.; Atluri S. N.** (2009): Large Deformation Analyses of Space-Frame Structures, Using Explicit Tangent Stiffness Matrices, Based on the Reissner variational principle and a von Karman Type Nonlinear Theory in Rotated Reference Frames. *CMES: Computer Modeling in Engineering & Sciences*, vol. 54, pp. 335-368.
- Cai Y. C.; Paik J. K.; Atluri S. N.** (2010): Locking-free Thick-Thin Rod/Beam Element for Large Deformation Analyses of Space-Frame Structures, Based on the Reissner variational Principle and A von Karman Type Nonlinear Theory. *CMES: Computer Modeling in Engineering & Sciences*, vol. 58, pp. 75-108.
- Cai Y. C.; Paik J. K.; Atluri S. N.** (2010): A Triangular Plate Element with Drilling Degrees of Freedom, for Large Rotation Analyses of Built-up Plate/Shell Structures, Based on the Reissner Variational Principle and the von Karman Non-linear Theory in the Co-rotational Reference Frame. *CMES: Computer Modeling in Engineering & Sciences*, vol.61, pp.273-312.
- Fan, C. M.; Liu, C. S.; Yeih, W. C.; Chan, H. F.** (2010): The Scalar Homotopy Method for Solving Non-Linear Obstacle Problems. *CMC: Computers, Materials & Continua*, vol. 15, pp. 67-86.
- Ku, C. Y.; Yeih, W. C.; Liu, C. S.** (2010): Solving Non-Linear Algebraic Equations by a Scalar Newton-homotopy Continuation Method. *International Journal of Nonlinear Science & Numerical Simulation*, vol. 11, pp. 435-450.
- Liu, C. S.; Atluri, S. N.** (2009): A Novel Time Integration Method for Solving A Large System of Non-Linear Algebraic Equations. *CMES: Computer Modeling in*

Engineering & Sciences, vol. 31, pp. 71-83.

Liu, C. S.; Atluri, S. N. (2008): A Fictitious Time Integration Method (FTIM) for Solving Mixed Complementarity Problems with Applications to Non-Linear Optimization. *CMES: Computer Modeling in Engineering & Sciences*, vol. 34, pp. 155-178.

Liu, C. S.; Atluri, S. N. (2009): A Fictitious Time Integration Method for the Numerical Solution of the Fredholm Integral Equation and for Numerical Differentiation of Noisy Data, and Its Relation to the Filter Theory. *CMES: Computer Modeling in Engineering & Sciences*, vol. 41, pp. 243-261.

Liu, C. S.; Ku, Y. C.; Yeih, W. C.; Fan, C. M.; Atluri, S. N. (2010): An Exponentially Convergent Scalar Homotopy Algorithm for Solving A Determinate/Indeterminate System of Non-Linear Algebraic Equations. *CMES: Computer Modeling in Engineering & Sciences*

Liu, C. S.; Yeih, W. C.; Atluri, S. N. (2010): An Enhanced Fictitious Time Integration Method for Non-Linear Algebraic Equations with Multiple Solutions: Boundary Layer, Boundary Value and Eigenvalue Problems. *CMES: Computer Modeling in Engineering & Sciences*, vol. 59, pp. 301-323.

Liu, C. S.; Yeih, W. C.; Kuo, L. C.; Atluri, S. N. (2009): A Scalar Homotopy Method for Solving an Over/Under-determined System of Non-Linear Algebraic Equations. *CMES: Computer Modeling in Engineering & Sciences*, vol. 53, pp. 47-71.

Paik, J. K.; Thayamballi, A. K.; Lee, S. K.; Kang, S. J. (2001): A semi-analytical method for the elastic-plastic large deflection analysis of welded steel or aluminum plating under combined in-plane and lateral pressure loads. *Thin-Walled Structure*, vol. 39, pp.125–152.

Paik, J. K.; Lee, M. S. (2005): A semi-analytical method for the elastic-plastic large deflection analysis of stiffened panels under combined biaxial compression/tension, biaxial in-plane bending, edge shear, and lateral pressure loads. *Thin-Walled Structure*, vol. 43, pp.375–410.

Timoshenko, S.; Krieger, S. W. (1959): *Theory of Plates and Shells*. McGraw-Hill Companies, 580 pages

Ueda, Y.; Rashed, S. M. H.; Paik, J. K. (1987): An Incremental Galerkin Method for Plates and Stiffened Plates. *Computers & Structures*, vol. 27, pp. 147-156.

Zhu H. H.; Cai Y. C.; Paik J. K.; Atluri S. N. (2010): Locking-free Thick-Thin Rod/Beam Element Based on a von Karman Type Nonlinear Theory in Rotated Reference Frames For Large Deformation Analyses of Space-Frame Structures. *CMES: Computer Modeling in Engineering & Sciences*, vol. 57, pp. 175-204.

Zhu, T.; Zhang, J.; Atluri, S. N. (1999): A meshless numerical method based on the local boundary integral equation (LBIE) to solve linear and non-linear boundary value problems. *Eng. Anal. Bound. Elem.*, vol. 23, pp. 375-389

Appendix A

In the appendix, for simplicity we denote $H_p(i, j, k, l, m, n, r, s)$ by H_p where p is from 1 to 16. The exact expression of the matrices and vectors in the derived Eq. (16) is given as follows:

1. The expression of the first order matrix $K_f = K_{f1} + K_{f2} + K_{f3}$

(a) K_{f1} associated with the first row of Eq. (15) can be written as

$$K_{f1} = \begin{pmatrix} K_{f1}(1,1) & K_{f1}(1,2) & \cdots & K_{f1}(1,N_j) & \cdots & K_{f1}(1,N_xN_y) \\ K_{f1}(2,1) & K_{f1}(2,2) & \cdots & K_{f1}(2,N_j) & \cdots & K_{f1}(2,N_xN_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f1}(N_i,1) & K_{f1}(N_i,2) & \cdots & K_{f1}(N_i,N_j) & \cdots & K_{f1}(N_i,N_xN_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f1}(N_xN_y,1) & K_{f1}(N_xN_y,2) & \cdots & K_{f1}(N_xN_y,N_j) & \cdots & K_{f1}(N_xN_y,N_xN_y) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$K_{f1}(N_i, N_j) = D \left(\frac{m^2}{a^2} + \frac{n^2}{b^2} \right)^2 H_{01}(i, j, m, n)$$

for

$$\begin{aligned} i, m &= 1, 2, \dots, N_x; \\ j, n &= 1, 2, \dots, N_y; \\ N_i &= (i-1)N_y + j; \\ N_j &= (m-1)N_y + n \end{aligned}$$

(b) K_{f2} associated with the 5th row of Eq. (15) can be obtained as

$$K_{f2} = \begin{pmatrix} K_{f2}(1,1) & K_{f2}(1,2) & \cdots & K_{f2}(1,N_j) & \cdots & K_{f2}(1,N_xN_y) \\ K_{f2}(2,1) & K_{f2}(2,2) & \cdots & K_{f2}(2,N_j) & \cdots & K_{f2}(2,N_xN_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f2}(N_i,1) & K_{f2}(N_i,2) & \cdots & K_{f2}(N_i,N_j) & \cdots & K_{f2}(N_i,N_xN_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f2}(N_xN_y,1) & K_{f2}(N_xN_y,2) & \cdots & K_{f2}(N_xN_y,N_j) & \cdots & K_{f2}(N_xN_y,N_xN_y) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$K_{f2}(N_i, N_j) = (-t) \frac{E\alpha^2\pi^4}{4a^2b^2}$$

$$\sum_{m=1}^M \sum_{n=1}^N \sum_{r=1}^R \sum_{s=1}^S A_{0mn} A_{0rs} (H_6 + H_7 - H_5 - H_8 + 2H_{13} - 2H_{14} - 2H_{15} + 2H_{16})$$

for

$$\begin{aligned} i, k &= 1, 2, \dots, N_x; \\ j, l &= 1, 2, \dots, N_y; \\ N_i &= (i-1)N_y + j; \\ N_j &= (k-1)N_y + l \end{aligned}$$

(c) K_{f3} associated with the 6^{th} row of Eq. (15) can be obtained as

$$K_{f3} = \begin{pmatrix} K_{f3}(1, 1) & K_{f3}(1, 2) & \dots & K_{f3}(1, N_j) & \dots & K_{f3}(1, N_x N_y) \\ K_{f3}(2, 1) & K_{f3}(2, 2) & \dots & K_{f3}(2, N_j) & \dots & K_{f3}(2, N_x N_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f3}(N_i, 1) & K_{f3}(N_i, 2) & \dots & K_{f3}(N_i, N_j) & \dots & K_{f3}(N_i, N_x N_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{f3}(N_x N_y, 1) & K_{f3}(N_x N_y, 2) & \dots & K_{f3}(N_x N_y, N_j) & \dots & K_{f3}(N_x N_y, N_x N_y) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$\begin{aligned} K_{f3}(N_i, N_j) = & (-t) \left\{ \frac{m^2\pi^2}{a^2} \left[\left(\frac{P_x}{bt} + \sigma_{rx} - \frac{6}{b^2t} M_x \right) H_{01}(i, j, m, n) + \frac{12}{b^3t} M_x H_{03}(i, j, m, n) \right] \right. \\ & + \frac{n^2\pi^2}{b^2} \left[\left(\frac{P_y}{at} + \sigma_{ry} - \frac{6}{a^2t} M_y \right) H_{01}(i, j, m, n) + \frac{12}{a^3t} M_y H_{02}(i, j, m, n) \right] \\ & \left. + \frac{2\tau\pi^2}{ab} mn \times H_{04}(i, j, m, n) \right\} \end{aligned}$$

for

$$\begin{aligned} i, m &= 1, 2, \dots, N_x; \\ j, n &= 1, 2, \dots, N_y; \\ N_i &= (i-1)N_y + j; \\ N_j &= (m-1)N_y + n \end{aligned}$$

2. The expression of the second order matrix $K_s = K_{s1} + K_{s2}$

(a) K_{s1} associated with the 3^{th} row of Eq. (15) can be obtained as

$$K_{s1} =$$

$$\begin{pmatrix} K_{s1}(1,1) & K_{s1}(1,2) & \dots & K_{s1}(1,N_j) & \dots & K_{s1}(1,N_x^2 N_y^2) \\ K_{s1}(2,1) & K_{s1}(2,2) & \dots & K_{s1}(2,N_j) & \dots & K_{s1}(2,N_x^2 N_y^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{s1}(N_i,1) & K_{s1}(N_i,2) & \dots & K_{s1}(N_i,N_j) & \dots & K_{s1}(N_i,N_x^2 N_y^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{s1}(N_x N_y,1) & K_{s1}(N_x N_y,2) & \dots & K_{s1}(N_x N_y,N_j) & \dots & K_{s1}(N_x N_y,N_x^2 N_y^2) \end{pmatrix}$$

where each component in the above matrix can be calculated by,

$$K_{s1}(N_i, N_j) =$$

$$(-t) \frac{E\alpha^2\pi^4}{4a^2b^2} \sum_{r=1}^R \sum_{s=1}^S A_{0rs} (H_1 + H_2 + H_3 + H_4 - 2H_9 - 2H_{10} - 2H_{11} - 2H_{12})$$

for

$$i, m, k = 1, 2, \dots, N_x;$$

$$j, n, l = 1, 2, \dots, N_y;$$

$$N_i = (i-1)N_y + j;$$

$$N_j = (m-1)N_x N_y^2 + (n-1)N_x N_y + (k-1)N_y + l$$

(b) K_{s2} associated with the 4^{th} row of Eq. (15) can be obtained as

$$K_{s2} =$$

$$\begin{pmatrix} K_{s2}(1,1) & K_{s2}(1,2) & \dots & K_{s2}(1,N_j) & \dots & K_{s2}(1,N_x^2 N_y^2) \\ K_{s2}(2,1) & K_{s2}(2,2) & \dots & K_{s2}(2,N_j) & \dots & K_{s2}(2,N_x^2 N_y^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{s2}(N_i,1) & K_{s2}(N_i,2) & \dots & K_{s2}(N_i,N_j) & \dots & K_{s2}(N_i,N_x^2 N_y^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{s2}(N_x N_y,1) & K_{s2}(N_x N_y,2) & \dots & K_{s2}(N_x N_y,N_j) & \dots & K_{s2}(N_x N_y,N_x^2 N_y^2) \end{pmatrix}$$

where each component in the above matrix can be calculated by,

$$K_{s2}(N_i, N_j) =$$

$$(-t) \frac{E\alpha^2\pi^4}{4a^2b^2} \sum_{m=1}^M \sum_{n=1}^N A_{0mn} (H_6 + H_7 - H_5 - H_8 + 2H_{13} - 2H_{14} - 2H_{15} + 2H_{16})$$

for

$$\begin{aligned} i, k, r &= 1, 2, \dots, N_x; \\ j, l, s &= 1, 2, \dots, N_y; \\ N_i &= (i-1)N_y + j; \\ N_j &= (k-1)N_xN_y^2 + (l-1)N_xN_y + (r-1)N_y + s \end{aligned}$$

3. The expression of the third order matrix K_t associated with the 2^{th} row of Eq. (15)

$$K_t = \begin{pmatrix} K_t(1,1) & K_t(1,2) & \dots & K_t(1,N_j) & \dots & K_t(1,N_x^3N_y^3) \\ K_t(2,1) & K_t(2,2) & \dots & K_t(2,N_j) & \dots & K_t(2,N_x^2N_y^2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_t(N_i,1) & K_t(N_i,2) & \dots & K_t(N_i,N_j) & \dots & K_t(N_i,N_x^3N_y^3) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ K_t(N_xN_y,1) & K_t(N_xN_y,2) & \dots & K_t(N_xN_y,N_j) & \dots & K_t(N_xN_y,N_x^3N_y^3) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$K_t(N_i, N_j) = (-t) \frac{E\alpha^2\pi^4}{4a^2b^2} (H_1 + H_2 + H_3 + H_4 - 2H_9 - 2H_{10} - 2H_{11} - 2H_{12})$$

for

$$\begin{aligned} i, m, k, r &= 1, 2, \dots, N_x; \\ j, n, l, s &= 1, 2, \dots, N_y; \\ N_i &= (i-1)N_y + j; \\ N_j &= (m-1)N_x^2N_y^3 + (n-1)N_x^2N_y^2 + (k-1)N_xN_y^2 + (l-1)N_xN_y + (r-1)N_y + s \end{aligned}$$

4. The expression of the constant vector $C = C_1 + C_2$

(a) C_1 associated with the 7^{th} row of Eq. (15) can be obtained as,

$$C_1 = \begin{pmatrix} C_1(1) \\ C_1(2) \\ \vdots \\ C_1(N_i) \\ \vdots \\ C_1(N_xN_y) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$C_1(N_i) = (-t) \left\{ \frac{m^2 \pi^2}{a^2} \left[\left(\frac{P_x}{bt} + \sigma_{rx} - \frac{6}{b^2 t} M_x \right) H_{01}(i, j, m, n) + \frac{12}{b^3 t} M_x H_{03}(i, j, m, n) \right] \right. \\ \left. + \frac{n^2 \pi^2}{b^2} \left[\left(\frac{P_y}{at} + \sigma_{ry} - \frac{6}{a^2 t} M_y \right) H_{01}(i, j, m, n) + \frac{12}{a^3 t} M_y H_{02}(i, j, m, n) \right] \right. \\ \left. + \frac{2\tau \pi^2}{ab} mn \times H_{04}(i, j, m, n) \right\}$$

for

$$i = 1, 2, \dots, N_x; \\ j = 1, 2, \dots, N_y; \\ N_i = (i-1)N_y + j;$$

(b) C_2 associated with the 8th row of Eq. (15) can be obtained as,

$$C_2 = \begin{pmatrix} C_2(1) \\ C_2(2) \\ \vdots \\ C_2(N_i) \\ \vdots \\ C_2(N_x N_y) \end{pmatrix}$$

Where each component in the above matrix can be calculated by,

$$C_2(N_i) = -P \times H_{00}(i, j)$$

for

$$i = 1, 2, \dots, N_x; \\ j = 1, 2, \dots, N_y; \\ N_i = (i-1)N_y + j;$$

5. The expression of the unknown vectors

(a) The expression of the unknown vector A_f

$$A_f = [A_1 \quad A_2 \quad \dots \quad A_{N_x N_y}]^T$$

(b) The expression of the unknown vector A_s

$$A_s = \begin{bmatrix} A_1 A_1 & A_1 A_2 & \cdots & A_1 A_{N_x N_y} & A_2 A_1 & A_2 A_2 & \cdots & A_2 A_{N_x N_y} \\ \cdots & A_{N_x N_y} A_1 & A_{N_x N_y} A_2 & \cdots & A_{N_x N_y} A_{N_x N_y} \end{bmatrix}^T$$

(c) The expression of the unknown vector A_t

$$A_t = \begin{bmatrix} A_1 A_1 A_1 & A_1 A_1 A_2 & \cdots & A_1 A_1 A_{N_x N_y} & A_1 A_2 A_1 & A_1 A_2 A_2 & \cdots & A_1 A_2 A_{N_x N_y} \\ \cdots & A_{N_x N_y} A_{N_x N_y} A_1 & A_{N_x N_y} A_{N_x N_y} A_2 & \cdots & A_{N_x N_y} A_{N_x N_y} A_{N_x N_y} \end{bmatrix}^T$$

6. The coefficient matrices H are given below,

$$H_{00}(i, j) = \iiint_V sx(i)sy(j)dxdydz$$

$$H_{01}(i, j, m, n) = \iiint_V sx(m)sy(n)sx(i)sy(j)dxdydz$$

$$H_{02}(i, j, m, n) = \iiint_V x \times sx(m)sy(n)sx(i)sy(j)dxdydz$$

$$H_{03}(i, j, m, n) = \iiint_V y \times sx(m)sy(n)sx(i)sy(j)dxdydz$$

$$H_{04}(i, j, m, n) = \iiint_V cx(m)cy(n)sx(i)sy(j)dxdydz$$

$$H_1(i, j, m, n, k, l, r, s) = \frac{(m+k)^2 s^2 + (n+l)^2 r^2}{\left[(m+k)^2 + \alpha^2 (n+l)^2 \right]^2} ml (nk - ml) \\ \times \iiint_V cx(m+k)cy(n+l)sx(r)sy(s)sx(i)sy(j)dxdydz$$

$$H_2(i, j, m, n, k, l, r, s) = \frac{(m+k)^2 s^2 + (n-l)^2 r^2}{\left[(m+k)^2 + \alpha^2 (n-l)^2 \right]^2} ml (nk + ml) \\ \times \iiint_V cx(m+k)cy(n-l)sx(r)sy(s)sx(i)sy(j)dxdydz$$

$$H_3(i, j, m, n, k, l, r, s) = \frac{(m-k)^2 s^2 + (n+l)^2 r^2}{\left[(m-k)^2 + \alpha^2 (n+l)^2 \right]^2} ml (nk + ml) \\ \times \iiint_V cx(m-k)cy(n+l)sx(r)sy(s)sx(i)sy(j)dxdydz$$

$$\begin{aligned}
H_4(i, j, m, n, k, l, r, s) &= \frac{(m-k)^2 s^2 + (n-l)^2 r^2}{\left[(m-k)^2 + \alpha^2 (n-l)^2\right]^2} ml (nk - ml) \\
&\quad \times \iiint_V cx(m-k)cy(n-l)sx(r)sy(s)sx(i)sy(j)dx dy dz \\
H_5(i, j, m, n, k, l, r, s) &= \frac{(m+k)^2 s^2 + (n+l)^2 r^2}{\left[(m+k)^2 + \alpha^2 (n+l)^2\right]^2} (nk - ml)^2 \\
&\quad \times \iiint_V cx(m+k)cy(n+l)sx(r)sy(s)sx(i)sy(j)dx dy dz \\
H_6(i, j, m, n, k, l, r, s) &= \frac{(m+k)^2 s^2 + (n-l)^2 r^2}{\left[(m+k)^2 + \alpha^2 (n-l)^2\right]^2} (nk + ml)^2 \\
&\quad \times \iiint_V cx(m+k)cy(n-l)sx(r)sy(s)sx(i)sy(j)dx dy dz \\
H_7(i, j, m, n, k, l, r, s) &= \frac{(m-k)^2 s^2 + (n+l)^2 r^2}{\left[(m-k)^2 + \alpha^2 (n+l)^2\right]^2} (nk + ml)^2 \\
&\quad \times \iiint_V cx(m-k)cy(n+l)sx(r)sy(s)sx(i)sy(j)dx dy dz \\
H_8(i, j, m, n, k, l, r, s) &= \frac{(m-k)^2 s^2 + (n-l)^2 r^2}{\left[(m-k)^2 + \alpha^2 (n-l)^2\right]^2} (nk - ml)^2 \\
&\quad \times \iiint_V cx(m-k)cy(n-l)sx(r)sy(s)sx(i)sy(j)dx dy dz \\
H_9(i, j, m, n, k, l, r, s) &= \frac{(m+k)(n+l)}{\left[(m+k)^2 + \alpha^2 (n+l)^2\right]^2} mlrs (nk - ml) \\
&\quad \times \iiint_V sx(m+k)sy(n+l)cx(r)cy(s)sx(i)sy(j)dx dy dz \\
H_{10}(i, j, m, n, k, l, r, s) &= \frac{(m+k)(n-l)}{\left[(m+k)^2 + \alpha^2 (n-l)^2\right]^2} mlrs (nk + ml) \\
&\quad \times \iiint_V sx(m+k)sy(n-l)cx(r)cy(s)sx(i)sy(j)dx dy dz \\
H_{11}(i, j, m, n, k, l, r, s) &= \frac{(m-k)(n+l)}{\left[(m-k)^2 + \alpha^2 (n+l)^2\right]^2} mlrs (nk + ml) \\
&\quad \times \iiint_V sx(m-k)sy(n+l)cx(r)cy(s)sx(i)sy(j)dx dy dz
\end{aligned}$$

$$H_{12}(i, j, m, n, k, l, r, s) = \frac{(m-k)(n-l)}{\left[(m-k)^2 + \alpha^2(n-l)^2\right]^2} m l r s (nk - ml) \\ \times \iiint_V s x(m-k) s y(n-l) c x(r) c y(s) s x(i) s y(j) d x d y d z$$

$$H_{13}(i, j, m, n, k, l, r, s) = \frac{(m+k)(n+l)}{\left[(m+k)^2 + \alpha^2(n+l)^2\right]^2} r s (nk - ml)^2 \\ \times \iiint_V s x(m+k) s y(n+l) c x(r) c y(s) s x(i) s y(j) d x d y d z$$

$$H_{14}(i, j, m, n, k, l, r, s) = \frac{(m+k)(n-l)}{\left[(m+k)^2 + \alpha^2(n-l)^2\right]^2} r s (nk + ml)^2 \\ \times \iiint_V s x(m+k) s y(n-l) c x(r) c y(s) s x(i) s y(j) d x d y d z$$

$$H_{15}(i, j, m, n, k, l, r, s) = \frac{(m-k)(n+l)}{\left[(m-k)^2 + \alpha^2(n+l)^2\right]^2} r s (nk + ml)^2 \\ \times \iiint_V s x(m-k) s y(n+l) c x(r) c y(s) s x(i) s y(j) d x d y d z$$

$$H_{16}(i, j, m, n, k, l, r, s) = \frac{(m-k)(n-l)}{\left[(m-k)^2 + \alpha^2(n-l)^2\right]^2} r s (nk - ml)^2 \\ \times \iiint_V s x(m-k) s y(n-l) c x(r) c y(s) s x(i) s y(j) d x d y d z$$

