12. DISTRIBUTED GEOMERIC NONLINEAR ANALYSIS OF PLATES

12.1 GENERAL

Thin walled structural elements like plates and shells are commonly found in spacecrafts, missiles, aircrafts, land-based vehicles, underwater vessels and structures, chemical processing equipments and modern housing. A good knowledge of structural behavior of these elements under static, dynamic and environmental loading conditions is required to utilize the materials economically and efficiently. Practically all the problems of mechanics are nonlinear but linearization is commonly used to have approximate solution. Linearized approximate solution of structural mechanics problems is valid for many practical or engineering purposes but in certain situations linearized treatment may be inadequate [130].

Nonlinearities in structural mechanics can arise in different ways. The nonlinearity, in which generalized Hooke's law is not valid i.e. if the material stress-strain behavior is nonlinear, is known as physical or material nonlinearity. Elastic-plastic stress analysis of plate is the example of material nonlinear problem. Alternatively nonlinearity caused by the deformation of elastic body, involving nonlinear strain displacement relation, comes under geometric nonlinear problems. A flexible plate subjected to transverse loading and experiencing large deformation is an example of geometric nonlinear problem. In the present work geometric nonlinearity is considered for analysis [131].

In linear analysis it has been implicitly assumed that both the displacements and strains developed in the structure are small. Practically it means that the geometry of the elements remains basically unchanged during the loading process and linear strain approximation can be used. In practice such assumptions fail frequently even though actual strains may be small and elastic limits of ordinary structural materials not exceeded. If it is required to calculate displacements accurately, geometric nonlinearity may have to be considered when the transverse displacement is not small compared to thickness of element. In this case membrane stresses, which are usually neglected in plate flexure, may cause considerable decrease of displacements as compared with linear solution. The interaction between membrane stresses and curvatures results in the stretching of the median surface, which in turn leads to nonlinear terms in strain displacement relations. Significant saving in weight may be achieved if thin plates are designed with due consideration to large deflection behavior.

The solution of nonlinear problem by displacement based finite element method can be usually attempted by one of three basic techniques: incremental or stepwise procedure, iterative or Newton Raphson type methods and step-iterative or mixed procedures. As these techniques are iterative in nature, the solution of a nonlinear problem may require 10 to 100 times time required for linear solution. Use of high performance computing or distributed computing can improve the computational efficiency. This chapter therefore is devoted to distributed implementation of geometric nonlinear analysis of plate.

12.2 METHOD FOR NONLINEAR ANALYSIS

Whether the displacements or strains are large or small, equilibrium conditions between internal and external forces have to be satisfied. For the case of small strains the stiffness matrix in local coordinates is linear i.e. it is not a function of the displacements and it remains the same for all states of deformation. But due to nonlinearity between the strain and displacements element stiffness matrix in global coordinates varies with the displacement [132].

The discretized nonlinear system can generally be written as a set of algebraic equations in the form of:

$$\psi(\mathbf{a}) \equiv \mathbf{P}(\mathbf{a}) + \mathbf{f} \equiv \mathbf{K}(\mathbf{a}) \,.\, \mathbf{a} + \mathbf{f} = 0 \qquad \ldots (12.1)$$

While the solution of a linear equation system has a form of:

$$\mathbf{K} \,.\, \mathbf{a} + \mathbf{f} = 0 \qquad \ldots (12.2)$$

where **K** is stiffness matrix, **f** is an actual force and **a** is structural displacements. The solution of linear equation can be accomplished without difficulty in a direct manner but for nonlinear systems various iterative methods (indirect methods) are available out of which Newton Raphson method is quiet popular and is described below:

The most obvious and direct solution procedure is one of iteration, which starts from the form

$$\mathbf{K} \, . \, \mathbf{a} + \mathbf{f} = 0 \qquad \ldots (12.3)$$

In which $\mathbf{K} = \mathbf{K}(\mathbf{a})$

If initially some value of $\mathbf{a} = \mathbf{a}^0$ is assumed, the improved approximation is obtained as

$$\mathbf{a}^1 = -(\mathbf{K}^0)^{-1}\mathbf{f} \text{ where } \mathbf{K}^0 = \mathbf{K}(\mathbf{a}^0) \qquad \ldots (12.4)$$

Repetition of the process can be written as

$$\mathbf{a}^n = -(\mathbf{K}^{n-1})^{-1} \, \mathbf{f} \qquad \ldots (12.5)$$

and this iteration is terminated when the error

$$\mathbf{e} = \mathbf{a}^n - \mathbf{a}^{n-1} \qquad \ldots (12.6)$$

becomes sufficiently small. Usually some norm of the error is determined and iteration continues until this is sufficiently small. The process is represented graphically as shown in Fig. 12.1(a).

In Newton Raphson method if an approximate solution $\mathbf{a} = \mathbf{a}^n$ is reached, an improved solution using a curtailed Taylor expression can be written as,

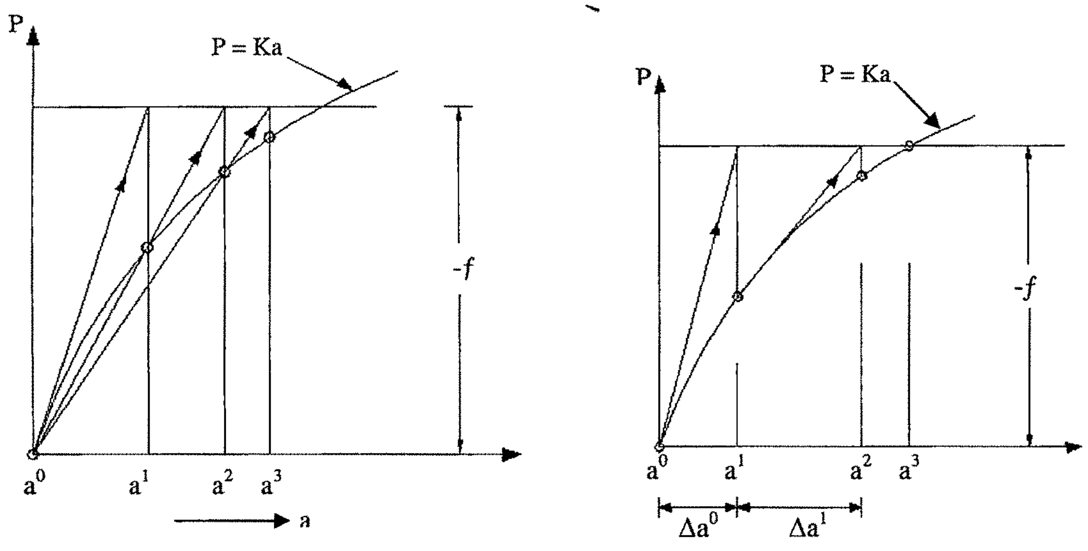
$$\psi(\mathbf{a}^{n+1}) \equiv \psi(\mathbf{a}^n) + (d\psi/d\mathbf{a})_n \; \Delta\mathbf{a}^n = 0 \qquad \ldots (12.7)$$

with $\mathbf{a}^{n+1} = \mathbf{a}^n + \Delta\mathbf{a}^0$

In above Eq. $d\psi/d\mathbf{a} \equiv d\mathbf{P}/d\mathbf{a} \equiv \mathbf{K}_T(\mathbf{a})$ represents a tangential matrix. The improved value of $\mathbf{a}^{n+1}$ can be computed by,

$$\Delta\mathbf{a}^n = -(\mathbf{K}_T{}^n)^{-1}\psi^n = -(\mathbf{K}_T{}^n)^{-1}(\mathbf{P}^n + \mathbf{f}) \qquad \ldots (12.8)$$

In this process at every step of computation a new set of linearized equations has to be solved for $\Delta \mathbf{a}^n$. The process is usually convergent in the vicinity of the solution. Fig. 12.1(b) graphically represents the process.



(a) Direct Method

(b) Newton Raphson Method

FIG. 12.1 ITERATIVE METHOD

In modified Newton Raphson method the tangential stiffness matrix is made constant as,

$$\mathbf{K}_T^n = \mathbf{K}_T^0 \qquad \ldots (12.9)$$

And so modified equilibrium equation will be

$$\Delta \mathbf{a}^n = -(\mathbf{K}_T^0)^{-1}(\mathbf{P}^n + \mathbf{f}) \qquad \ldots (12.10)$$

In this case a simple resolution of the same equation system is repeatedly used. This is more economical at each step but the convergence is slower.

12.3 GEOMETRIC NONLINEAR FINITE ELEMENT FORMULATION

The steps for geometric nonlinear finite element analysis based on Newton Raphson iterative method [133, 134] are as follows:

(a) Initial linear solution is obtained as first approximation of displacements $\mathbf{a}^0$.

(b) Unbalanced load vector ψ^0 is found by $\int \mathbf{B}^T \sigma \, dV - \mathbf{f}$. where σ is internal stress vector and $\mathbf{f}$ is external forces vector. If the displacements are large, strain depends nonlinearly on displacements and **B** depends on **a** and it can be written as $\mathbf{B} = \mathbf{B_0} + \mathbf{B_L}(\mathbf{a})$.

(c) Tangential stiffness matrix $\mathbf{K_T}^0$ is calculated as $\mathbf{K_T} = \mathbf{K_0} + \mathbf{K_L} + \mathbf{K_\sigma}$ where $\mathbf{K_0}$ is small displacement matrix, $\mathbf{K_L}$ is large displacement matrix and $\mathbf{K_\sigma}$ is initial stress matrix.

(d) Correction in displacements is calculated as $\Delta\mathbf{a_0} = -(\mathbf{K_T}^0)^{-1}\psi^0$.

The steps (b), (c) and (d) are repeated until ψ^n becomes sufficiently small or till the desired Euclidean norm displacement criteria is satisfied. The final solution of first load increment becomes initial solution of next load increment in the incremental loading procedure. As the geometric nonlinear analysis requires number of iterations to reach the final solution, it is computationally 10 to 100 times expensive than the linear solution.

In the present work eight-noded isoparametric quadrilateral element as shown in Fig. 12.2 is used. At each node following five degrees of freedom are considered: u, v, w, θ_x, θ_y. i.e. inplane displacements (u, v), lateral displacement (w) and rotations about y and x axes (θ_x, θ_y). The nodal displacement vector is defined as

$$\mathbf{a}^T = \{\mathbf{a}_i^p\ ,\ \mathbf{a}_i^b\}^T = \{u_1,v_1,w_1,\ \theta_{x1},\ \theta_{y1},\ u_2,v_2,w_2,\ \theta_{x2},\ \theta_{y2},\ \ldots\ldots\ u_8,v_8,w_8,\ \theta_{x8},\ \theta_{y8}\}^T$$

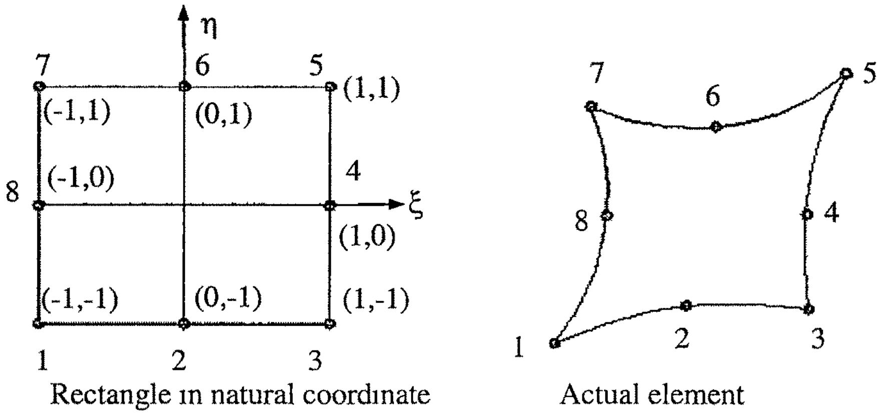


FIG. 12.2 EIGHT NODED ISOPARAMETRIC ELEMENT

The shape functions for this element in terms of the non-dimensional coordinate system are as follows:

$$\begin{array}{ll} N_1 = \xi(\xi - 1)\,\eta(\eta-1) / 4 & N_2 = (1-\xi^2)\,\eta\,(\eta-1) / 2 \\ N_3 = \xi(\xi + 1)\,\eta(\eta-1) / 4 & N_4 = \xi(\xi + 1)\,(1-\eta^2) / 2 \\ N_5 = \xi(\xi + 1)\,\eta(\eta+1) / 4 & N_6 = (1-\xi^2)\,\eta(\eta+1) / 2 \\ N_7 = \xi(\xi - 1)\,\eta(\eta+1) / 4 & N_8 = \xi(\xi - 1)\,(1-\eta^2) / 2 \end{array} \qquad \ldots (12.11)$$

In geometrical nonlinear analysis the lateral displacements will be responsible for development of membrane type strains and now the two problems of 'in-plane' and 'lateral' deformation can no longer be dealt with separately but are coupled.

As shown in Fig. 12.3 transverse displacement 'w' produces some additional extension in X and Y directions of the middle surface and the length dx stretches to

$$dx' = \sqrt{1 + (\partial w/\partial x)^2} \;=\; dx\,(\,1 + (1/2)(\partial w/\partial x)^2 + . \,.\,) \qquad \ldots (12.12)$$

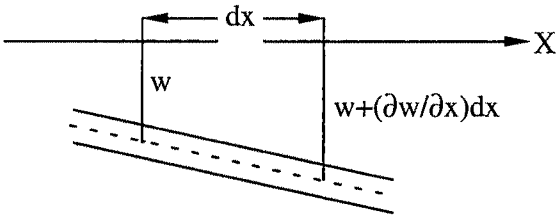


FIG. 12.3 INCREASE IN MID SURFACE LENGTH DUE TO w DISP.

The strain components, in terms of middle surface displacements, for coupled plane and flexural case including shear deformation are as follows:

$$\varepsilon = \left\{\begin{array}{c} \varepsilon_p \\ \varepsilon_b \\ \varepsilon_s \end{array}\right\} = \left\{\begin{array}{cl} & \varepsilon_x \\ & \varepsilon_y \\ & \gamma_{xy} \\ -\partial^2 w/\partial x^2 & = -\partial\theta_x/\partial x \\ -\partial^2 w/\partial y^2 & = -\partial\theta_y/\partial y \\ -2\partial^2 w/\partial x\partial y & = -\partial\theta_x/\partial y - \partial\theta_y/\partial x \\ -\phi_x & = \partial w/\partial x - \theta_x \\ -\phi_y & = \partial w/\partial y - \theta_y \end{array}\right\} \qquad \ldots (12.13)$$

The corresponding stresses (defined in terms of resultants) as shown in Fig. 12.4 are given by

$$\sigma = \left\{\begin{array}{c} \sigma_p \\ \sigma_b \\ \sigma_s \end{array}\right\} = \left[\; T_x \quad T_y \quad T_{xy} \quad M_x \quad M_y \quad M_{xy} \quad Q_x \quad Q_y \;\right]^T \qquad \ldots (12.14)$$

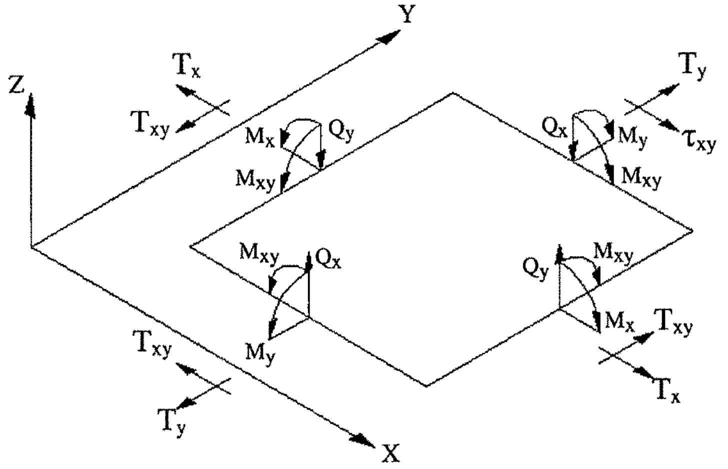


FIG. 12.4 COMPONENTS OF RESULTANT FORCES (IN +VE DIRECTION)

In defining the strain in X direction considering up to second approximation (Fig. 12.3)

$$\varepsilon_x = \partial u/\partial x + (1/2)\,(\partial w/\partial x)^2$$

Considering in a similar way, other components of strain can be written as.

$$\varepsilon = \begin{Bmatrix} \varepsilon_p \\ \varepsilon_b \\ \varepsilon_s \end{Bmatrix} = \begin{Bmatrix} \partial u/\partial x \\ \partial v/\partial y \\ \partial u/\partial y + \partial v/\partial x \\ -\partial^2 w/\partial x^2 \\ -\partial^2 w/\partial y^2 \\ -2\partial^2 w/\partial x\partial y \\ -\phi_x = \partial w/\partial x - \theta_x \\ -\phi_y = \partial w/\partial y - \theta_y \end{Bmatrix} + \begin{Bmatrix} (1/2)(\partial w/\partial x)^2 \\ (1/2)(\partial w/\partial y)^2 \\ (\partial w/\partial x)(\partial w/\partial y) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} \varepsilon_0^{\,p} \\ \varepsilon_0^{\,b} \\ \varepsilon_0^{\,s} \end{Bmatrix} + \begin{Bmatrix} \varepsilon_L^{\,p} \\ 0 \\ 0 \end{Bmatrix} \quad \ldots (12.15)$$

In this first term is related to the linear strain and the second term corresponds to nonlinear contribution.

The relations between stresses and strains are considered linear and constitutive law matrix is similar to that of plane stress and plate bending problems.

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}^p & 0 & 0 \\ 0 & \mathbf{D}^b & 0 \\ 0 & 0 & \mathbf{D}^s \end{bmatrix} \quad \ldots (12.16)$$

where,

$$\mathbf{D}^p = Et/(1-\nu^2)\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ \nu & 1 & 0 \end{bmatrix} \quad \mathbf{D}^b = Et^3/12(1-\nu^2)\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \quad \mathbf{D}^s = Et\,/\,2.4(1+\nu)\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

To derive tangent stiffness matrix relation between $d\psi$ and $d\mathbf{a}$ can be established by taking appropriate variation of ψ with respect to $d\mathbf{a}$,

$$d\psi = \int d\mathbf{B}^T \sigma\, dV + \int \mathbf{B}^T d\sigma\, dV = \mathbf{K}_T d\mathbf{a} \qquad \text{. . . (12.17)}$$

$$d\sigma = \mathbf{D}\, d\varepsilon = \mathbf{D}\,\mathbf{B}\, d\mathbf{a} \qquad \text{. . . (12.18)}$$

$$\text{As } \mathbf{B} = \mathbf{B}_0 + \mathbf{B}_L(\mathbf{a}) \qquad \text{. . . (12.19)}$$

$$d\psi = \int d\mathbf{B}_L^T \sigma\, dV + \int \mathbf{B}^T \mathbf{D}\,\mathbf{B}\, dV\, d\mathbf{a} \qquad \text{. . . (12.20)}$$

$$\text{Therefore } d\psi = \mathbf{K}_\sigma\, d\mathbf{a} + \mathbf{K}'\, d\mathbf{a} \qquad \text{. . . (12.21)}$$

$$\text{where } \mathbf{K}_\sigma = \int d\mathbf{B}_L^T \sigma\, dV \text{ and } \mathbf{K}' = \int \mathbf{B}^T \mathbf{D}\,\mathbf{B}\, dV = \mathbf{K}_0 + \mathbf{K}_L \qquad \text{. . . (12.22)}$$

$$\text{Thus } d\psi = (\mathbf{K}_\sigma + \mathbf{K}_0 + \mathbf{K}_L)d\mathbf{a} = \mathbf{K}_T d\mathbf{a} \qquad \text{. . . (12.23)}$$

Where $\mathbf{K_0}$ is small displacement matrix, $\mathbf{K_L}$ is large displacement matrix and $\mathbf{K}_\sigma$ is initial stress matrix.

The strain displacement relation matrix B, consisting of linear and nonlinear terms can be For calculated as:

$$\mathbf{B} = \mathbf{B}_0 + \mathbf{B}_L \qquad \text{. . . (12.24)}$$

where $\mathbf{B}_0 = \begin{bmatrix} \mathbf{B}_0^p & 0 \\ 0 & \mathbf{B}_0^b \end{bmatrix} \qquad \mathbf{B}_L = \begin{bmatrix} 0 & \mathbf{B}_L^b \\ 0 & 0 \end{bmatrix}$

Here $\mathbf{B}_0^p$ and $\mathbf{B}_0^b$ are standard strain displacement linking matrices for linear in-plane and bending element as:

$$\mathbf{B}_0 = \sum_{i=1}^{nodes} \begin{bmatrix} \partial N_i/\partial x & 0 & 0 & 0 & 0 \\ 0 & \partial N_i/\partial y & 0 & 0 & 0 \\ \partial N_i/\partial y & \partial N_i/\partial x & 0 & 0 & 0 \\ 0 & 0 & 0 & -\partial N_i/\partial x & 0 \\ 0 & 0 & 0 & 0 & -\partial N_i/\partial y \\ 0 & 0 & 0 & -\partial N_i/\partial y & -\partial N_i/\partial x \\ 0 & 0 & \partial N_i/\partial x & -N_i & 0 \\ 0 & 0 & \partial N_i/\partial y & 0 & -N_i \end{bmatrix} \qquad \text{. . . (12.25)}$$

For deriving $\mathbf{B}_L^b$ variation of ε_L^p with respect to parameter $\mathbf{a}^b$, displacements corresponding to plate bending, is taken. The nonlinear strain component can be written conveniently as,

$$\varepsilon_L^p = (1/2) \begin{bmatrix} \partial w/\partial x & 0 \\ 0 & \partial w/\partial y \\ \partial w/\partial y & \partial w/\partial x \end{bmatrix} \begin{Bmatrix} \partial w/\partial x \\ \partial w/\partial y \end{Bmatrix} = (1/2)\ \mathbf{A}\,\theta \qquad \ldots (12.26)$$

The derivatives (slopes) of w can be related to the nodal displacements **a** as,

$$\theta = \begin{Bmatrix} \partial w/\partial x \\ \partial w/\partial y \end{Bmatrix} = \mathbf{G\ a} \qquad \ldots (12.27)$$

In which G matrix is defined purely in terms of the co-ordinates as [135],

$$\mathbf{G} = \sum_{1}^{\text{Nodes}} \begin{bmatrix} 0 & 0 & 0 & \partial N_I/\partial x & 0 & 0 \\ 0 & 0 & 0 & \partial N_I/\partial y & 0 & 0 \end{bmatrix} \qquad \ldots (12.28)$$

The variation of ε_L^p is written as,

$$d\varepsilon_L^p = (1/2)\, d\mathbf{A}\,\theta + (1/2)\,\mathbf{A}\, d\theta = \mathbf{A\,G}\, d\mathbf{a} \qquad \ldots (12.29)$$

So, $$\mathbf{B}_L^b = \mathbf{A\,G} \qquad \ldots (12.30)$$

For calculation of various elements of **A** matrix isoparametric shape functions and initial displacements of a particular iteration are to be considered. So,

$$\partial w/\partial x = (\partial N_1/\partial x)w_1 + (\partial N_2/\partial x)w_2 + \ldots\ldots\ldots + (\partial N_8/\partial x)w_8 \qquad \ldots (12.31)$$

and $$\partial w/\partial y = (\partial N_1/\partial y)w_1 + (\partial N_2/\partial y)w_2 + \ldots\ldots\ldots + (\partial N_8/\partial y)w_8 \qquad \ldots (12.32)$$

After calculation of $\mathbf{B}_0$ and $\mathbf{B}_L^b$ matrices they are superimposed to get **B** matrix from which **K′** matrix is calculated as,

$$\mathbf{K}' = \mathbf{K}_0 + \mathbf{K}_L = \int \mathbf{B}^T \mathbf{D}\,\mathbf{B}\, dV \qquad \ldots (12.33)$$

Finally $\mathbf{K}_\sigma$ is calculated as, defined earlier as $\int d\mathbf{B}_L^T\, \sigma\, dV$,

$$d\mathbf{B}_L^T = \begin{bmatrix} 0 & 0 \\ d\mathbf{B}_L^{bT} & 0 \end{bmatrix} \qquad \ldots (12.34)$$

So, $$\mathbf{K}_\sigma\, da = \int_V \begin{bmatrix} 0 & 0 \\ \mathbf{G}^T d\mathbf{A}^T & 0 \end{bmatrix} \begin{bmatrix} T_x & T_y & T_{xy} & M_x & M_y & M_{xy} & Q_x & Q_y \end{bmatrix}^T dV \qquad \ldots (12.35)$$

$$d\mathbf{A}^T \begin{bmatrix} T_x \\ T_y \\ T_{xy} \end{bmatrix} = \begin{bmatrix} T_x & T_{xy} \\ T_{xy} & T_y \end{bmatrix} d\theta = \begin{bmatrix} T_x & T_{xy} \\ T_{xy} & T_y \end{bmatrix} \mathbf{G}\ d\mathbf{a}^b \qquad \ldots (12.36)$$

This gives the formulation for $\mathbf{K}_\sigma$ as

$$\mathbf{K}_\sigma = \int_V \mathbf{G}^T \begin{bmatrix} T_x & T_{xy} \\ T_{xy} & T_y \end{bmatrix} \mathbf{G}\ dV \qquad \ldots (12.37)$$

From nodal displacements linear and nonlinear terms of strain are calculated and combined. Subsequently using constitutive law matrix **D** as defined earlier the membrane forces, T_x, T_y and T_{xy} can be calculated.

The Gauss quadrature formula is generally used to evaluate the integrals. In the Gauss integration technique, a polynomial of degree (2n-1) can be integrated exactly by n sampling points. Thus the stiffness matrix can be evaluated as follows:

$$\mathbf{K}' = \sum_{a=1}^{NG} \sum_{b=1}^{NG} \{[\mathbf{B}(x,y)]^T[\mathbf{D}][\mathbf{B}(x,y)]\} \, |J| W_a W_b \qquad \ldots (12.38)$$

$$\mathbf{K}_\sigma = \sum_{a=1}^{NG} \sum_{b=1}^{NG} \{[\mathbf{G}(x,y)]^T[\sigma][\mathbf{G}(x,y)]\} \, |J| W_a W_b \qquad \ldots (12.39)$$

where W_a and W_b are the weighting factors corresponding to Gauss sampling points and NG is the number of Gauss points selected for the integration.

Numerically evaluated $\mathbf{K}_0$, $\mathbf{K}_L$ and $\mathbf{K}_\sigma$ are added to have tangent stiffness matrix of an element.

To evaluate the nodal loads due to uniformly distributed normal surface pressure P_0, the displacement normal to the surface of the element is required. The load vector at node i is given by,

$$\mathbf{P} = \int_A P_0 [N_i]^T dA = \sum_{a=1}^{NG} \sum_{b=1}^{NG} P_0 [N_i]^T |J| W_a W_b \qquad \ldots (12.40)$$

12.4 DISTRIBUTED IMPLEMENTATION OF NONLINEAR ANALYSIS

Geometrical nonlinear finite element analysis of plate, using procedure discussed in earlier section, is implemented here over distributed computing through Local

Area Network and WebDedip environment. It is a coarse grain implementation, in which application is divided into small tasks, which can run concurrently over different computers, and communication of data takes place through intermediate files transferred by FTP. Two alternative implementations are considered in this section.

12.4.1 First Approach

In first alternative calculation of unbalanced load vector (ψ), calculation of initial and large displacement stiffness matrix ($\mathbf{K}_0 + \mathbf{K}_L$), and calculation of initial stress stiffness matrix ($\mathbf{K}_\sigma$) are carried out, for entire structure, in parallel on different computers. Subsequently they are assembled on one computer for the solution. In the first iteration displacements obtained from linear elastic analysis are taken as initial displacements. The geometrical data, load data and initial displacements are communicated to three computers, where above three process are carried out. After calculation, the computers communicate stiffness matrix and load vector of entire structure to any one computer where, stiffness matrices ($\mathbf{K}_0 + \mathbf{K}_L$) and ($\mathbf{K}_\sigma$) are added to get tangent stiffness matrix and using unbalanced load vector change in displacement ($\Delta\mathbf{a}$) is calculated. Using this solution revised displacements are calculated and compared with displacement of previous iteration. For comparison Euclidean norm displacement criteria as defined below is used.

$$e = \left(\sqrt{\Sigma \mathbf{a}_{i+1}^2} - \sqrt{\Sigma \mathbf{a}_i^2}\right) / \sqrt{\Sigma \mathbf{a}_i^2} \qquad \ldots (12.41)$$

When the value of e is less than 0.01, the iterations will be stopped and next increment of load is started. For the next increment of load displacements of previous iteration are taken as initial displacements.

Four subprograms are prepared i.e. UNBALOD for calculating unbalanced load vector, STIFK for calculating initial and large displacement stiffness matrix, STIFKS for calculating initial stress stiffness matrix and ASSMBL for assembling stiffness matrices and load vector and for solving equations to get final displacements and checking convergence. The implementation is graphically represented in Fig. 12.5.

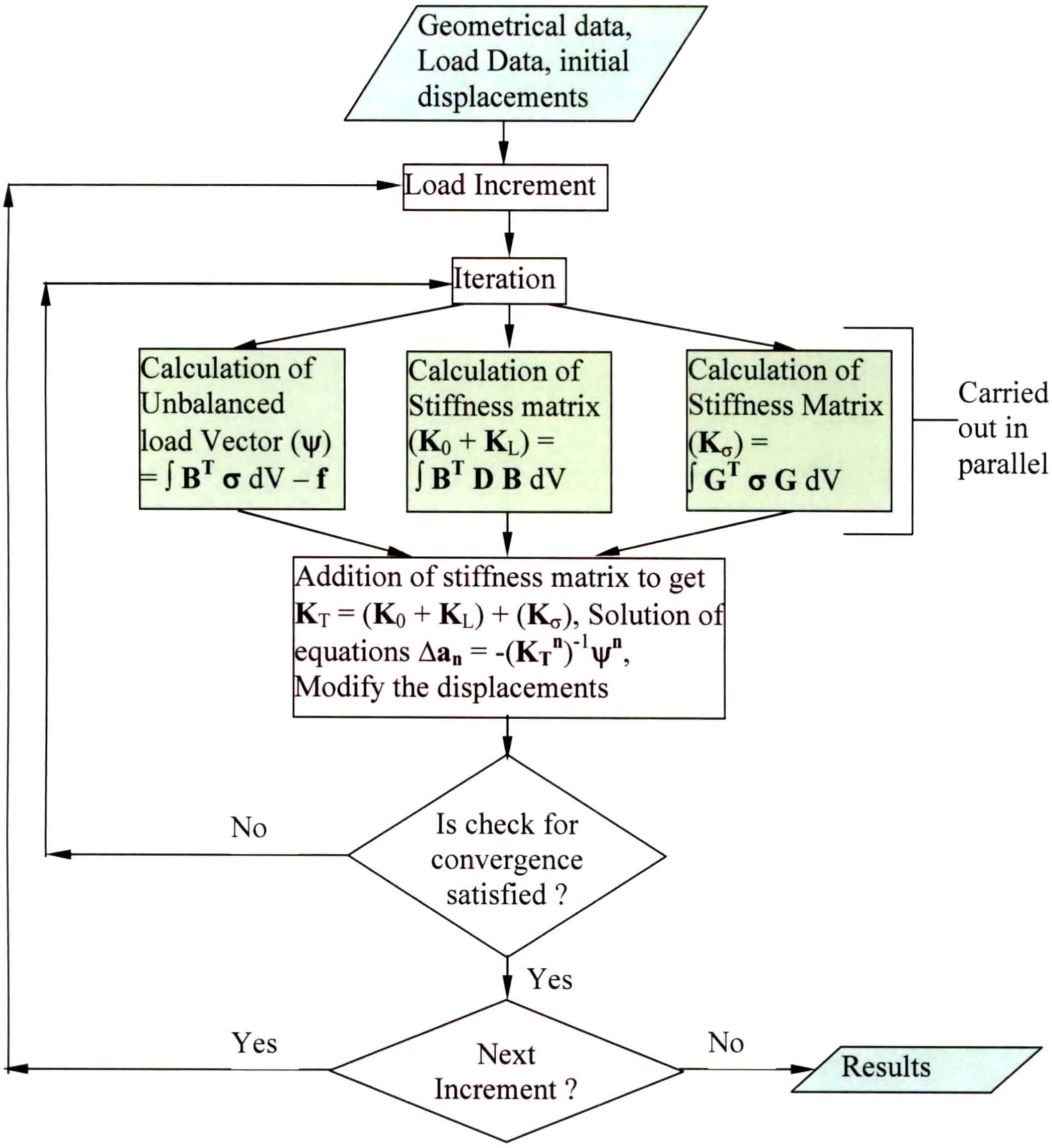

Geometrical data, Load Data, initial displacements
Load Increment
Iteration
Calculation of Unbalanced load Vector (ψ) $= \int \mathbf{B}^T \boldsymbol{\sigma}\, dV - \mathbf{f}$
Calculation of Stiffness matrix $(\mathbf{K}_0 + \mathbf{K}_L) = \int \mathbf{B}^T \mathbf{D}\,\mathbf{B}\, dV$
Calculation of Stiffness Matrix $(\mathbf{K}_\sigma) = \int \mathbf{G}^T \boldsymbol{\sigma}\, \mathbf{G}\, dV$
Carried out in parallel
Addition of stiffness matrix to get $\mathbf{K}_T = (\mathbf{K}_0 + \mathbf{K}_L) + (\mathbf{K}_\sigma)$, Solution of equations $\Delta \mathbf{a}_n = -(\mathbf{K}_T^n)^{-1} \psi^n$, Modify the displacements
No
Is check for convergence satisfied ?
Yes
Yes
Next Increment ?
No
Results

FIG. 12.5 PARALLEL IMPLEMENTATION OF NONLINEAR ANALYSIS

As the entire structure is considered, bandwidth is minimum and so calculation of load vector and stiffness matrices does not take much time. The time taken for calculation of load vector is lowest while the solution of equation takes more time in comparison to other processes. The size of intermediate files depends on size of structure. As communication is done through FTP, transfer of large files takes more time and sometimes problems may occur in transfer, which may terminate the entire application. This approach uses only three computers for small or large size problem, and change in number of computers is not possible. So the scalability in this approach is not very high.

12.4.2 Second Approach

In second alternative finite element mesh of a plate is divided into small parts, known as substructures [136], and analysis is carried out using number of computers equal to substructures. In this approach, calculation of unbalanced load vector and tangent stiffness matrix of each substructure is carried out concurrently on different computers and subsequently they are assembled to get degrees of freedom at interface nodes. From displacements of interface nodes, displacements of internal nodes are calculated again in parallel on different computers. The application is divided into five tasks as shown in Table 12.1.

TABLE 12.1 LIST OF TASKS FOR NONLINEAR ANALYSIS

No.	Task	Function of task
1.	DGNFEP1	Divides entire finite element domain into small parts, known as substructures and distribute data of each substructure i.e. Geometrical, load, initial displacements and load increment to different computers.
2.	DGNFEP2	Calculates tangent stiffness matrix and unbalanced load vector corresponding to boundary / interface degrees of freedom, using static condensation, for each substructure in parallel on different computers and communicates the same for further assembly.
3.	DGNFEP3	Collects tangent stiffness matrix and unbalanced load vector of each substructure and assemble to have stiffness matrix and load vector corresponding to interface degrees of freedom. After incorporating boundary degrees of freedom, of original structure, calculates change in displacements of interface nodes and distributes the appropriate displacements of each substructure to different computers.
4.	DGNFEP4	For each substructure, calculates change in displacements of internal nodes from displacements of boundary nodes and communicates the same for overall results.
5.	DGNFEP5	After combining change in displacements of all substructures, revise the initial displacements. The initial and final displacements are compared and Euclidean norm displacement criteria is checked and if satisfied load is incremented otherwise next iteration will be started and the control is transferred to first process till the end of all load increment. The revised displacements will become initial displacements for next iteration.

In the first task finite element mesh is divided into parts such that each part consists of same number of elements. The first process generates the data for each substructure consisting of geometric and material data, number of load increment, boundary nodes and initial displacements of each node of substructure.

In second task for static condensation, internal nodes are numbered first and subsequently the boundary nodes. In this process bandwidth of substructure increases and from banded matrix of entire substructure degrees of freedom corresponding to internal nodes are condensed to get tangent stiffness matrix of substructure corresponding to boundary nodes. Similarly unbalanced load vector corresponding to boundary nodes are calculated. Second task prepares a intermediate file consisting of substructure's banded tangent stiffness matrix and unbalanced load vector along with corresponding number of boundary degrees of freedom. To keep size of file smaller only banded part of stiffness matrix is stored. But due to larger bandwidth second task takes more time.

Third task assembles tangent stiffness matrix and unbalanced load vector depending on degrees of freedom in a banded form to keep computation time and memory requirements lower. After solution third task prepares an intermediate file consisting of displacements corresponding to boundary nodes for different substructures.

Fourth task prepares substructure's tangent stiffness matrix and unbalanced load vector in a banded form and boundary conditions, i.e. displacements at boundary nodes as supplied by third task, are imposed to get displacements of internal nodes. In this task as all the nodes are numbered in sequence, bandwidth is the lowest and so process is faster and requires less memory. This task prepares an intermediate file consisting of change in displacements at all nodes for entire substructure. After combining results of each substructure and checking convergence, fifth task generates an intermediate file consisting of final displacement at each node after a particular iteration. If a Euclidian norm displacement criterion is satisfied then fifth task increases the load increment number in the intermediate file, otherwise it keeps the same increment number for the further iteration. The file prepared by fifth task is read by first task in subsequent iteration to have initial displacements.

12.5 ANALYSIS PROBLEM AND RESULTS

An example of plate subjected to uniformly distributed load is considered for distributed geometric nonlinear finite element analysis. The size of plate is 12 m × 12 m and is clamped on all edges. Young's Modulus of Elasticity (E) is 2.2 × 10^6 t/m^2, Poisson's Ratio (ν) = 0.3, thickness of plate (t) = 0.1 m and Rigidity of Plate = D = E t^3 /12(1 - ν^2) = 201.465 t-m

Due to symmetry about both the axis only quarter part of the plate is considered for the solution. Again the quarter plate is divided into 64 and 1296 eight-nodded isoparametric element having 1125 and 20165 degrees of freedom. Typical meshing along with element and node number is shown in Fig. 12.6.

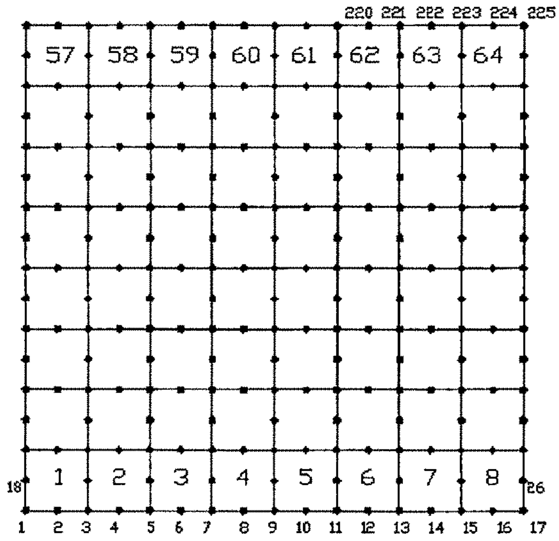


FIG. 12.6 TYPICAL DISCRETIZATION OF PLATE

The maximum uniformly distributed load of 4.04174 t/m^2 is applied in thirteen load increments. Solution in each increment is carried out by Newton type iterative method. The final solution for a particular load increment is considered as an initial displacement for the next increment. Load and final displacement of are represented in dimensionless form as shown in Table 12.2. It also includes the comparison of final displacements obtained by analytical method represented by Timoshenko [116] and with that of Finite Difference Method reported by Alami [137]. The results are shown graphically in Fig. 12.7. The result of finite element formulation matches closely with the available results.

TABLE 12.2 COMPARISON OF DISPLACEMENTS AT DIFFERENT LOADS

Sr. No.	'q' t/m^2	q a^4/Dt	w_{max} / t		w_{max} / t	
			Timoshenko [116]	Alami [137]	FEM - 1125 DOF	FEM - 20165 DOF
1	0.311	20	0.39	0.38	0.37	0.40
2	0.622	40	0.63	0.67	0.64	0.67
3	0.933	60	0.83	0.89	0.85	0.88
4	1.244	80	1.01	1.04	1.03	1.06
5	1.555	100	1.15	1.19	1.15	1.19
6	1.865	120	1.26	1.32	1.28	1.32
7	2.176	140	1.37	1.41	1.40	1.44
8	2.487	160	1.46	1.52	1.46	1.56
9	2.798	180	1.55	1.60	1.59	1.62
10	3.109	200	1.62	1.66	1.65	1.69
11	3.420	220	1.68	1.74	1.72	1.77
12	3.731	240	1.74	1.81	1.79	1.84
13	4.042	260	1.8	1.87	1.86	1.92

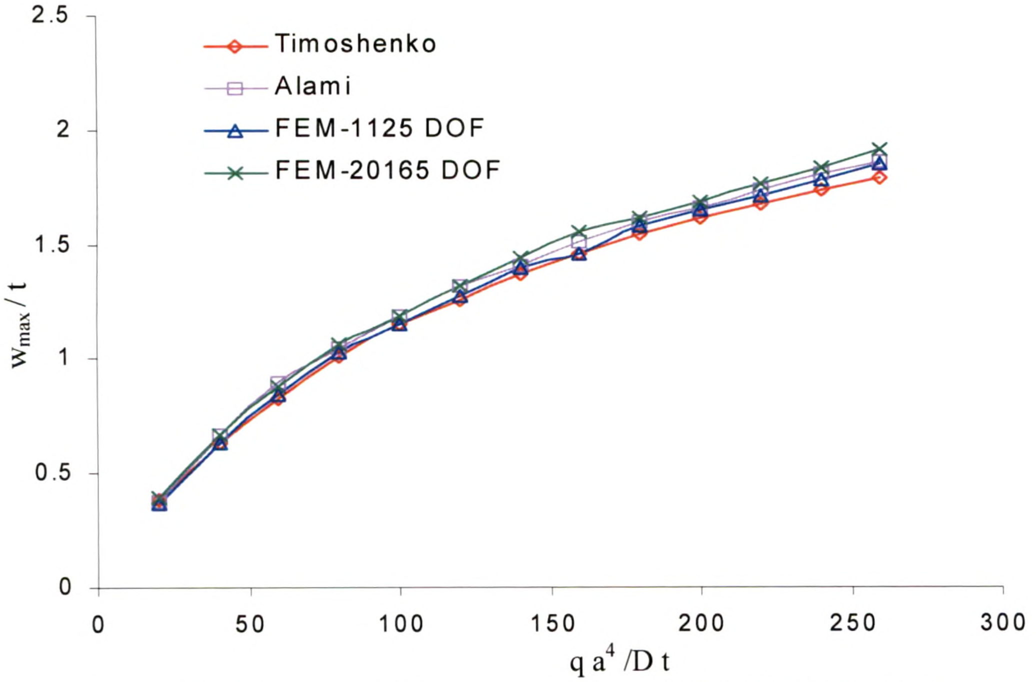


FIG. 12.7 COMPARISON OF NONLINEAR DISPLACEMENTS

To implement distributed computing two approaches, as discussed in earlier section, are used. As per first approach calculation of unbalanced load vector, initial and large displacement stiffness matrices and initial stress stiffness matrix for entire structure is carried out in parallel on different computers. After collecting load vector and stiffness matrices, tangent stiffness matrix is formed and after solution of linear equations, revised displacements are calculated and convergence is checked. If convergence is satisfied next load increment is considered otherwise iterations are continued till convergence is achieved.

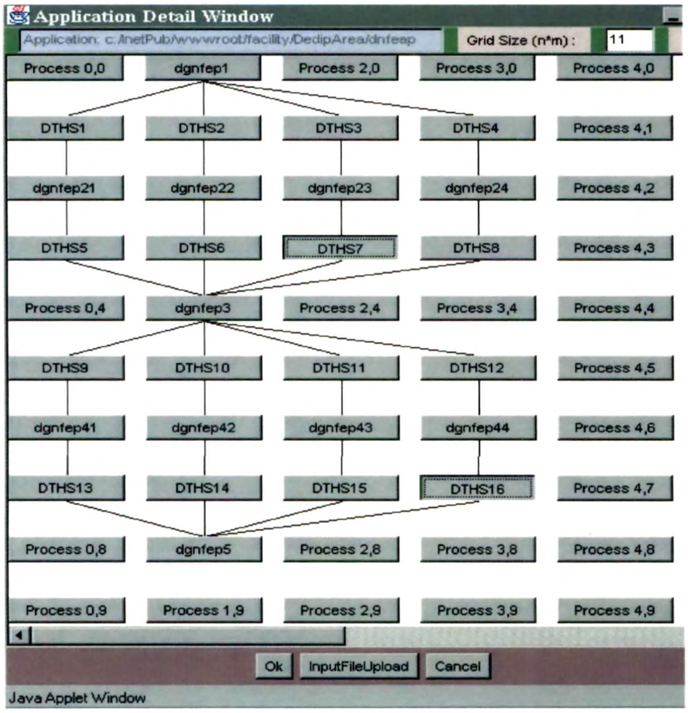
The time taken by various processes to solve problem with 20165 degrees of freedom, is shown in Table 12.4. Three Pentium IV computers running at 1.8 GHz connected in LAN through ethernet were used in this application. In this method process UNBALOD, STIFK and STIFKS are carried out in parallel while process ASSMBL is carried out sequentially. The timing shown here are only for one iteration. Number of such iterations are carried out till final load reaches, but in all iteration time spent in different processes is almost same.

TABLE 12.3 COMPARISON OF TIMING FOR FIRST ALTERNATIVE

Name of Process	NB	NEQ	Process Time (Sec)	Time for Distributed Computing (Sec)		Sequential Time (Sec)	Speedup
				Comput.	Commun.		
UNBALOD	565	20165	10	265	150	355	0.86
STIFK			80				
STIFKS			85				
ASSMBL			180				

In the second approach, substructure technique is used. The problem with 20165 DOF is divided into 2, 3, 4 and 6 substructures and calculation of tangent stiffness matrix and unbalanced load vector for each substructure is carried out in parallel on same number of computers. After calculating displacements of boundary nodes sequentially, calulation of displacements of internal nodes is carried out in parallel. Finally assembly of result of each substructure and check for convergence is carried out sequentially. Screen shot of WebDedip GUI for configuration of application on four computer is shown in Fig. 12.8

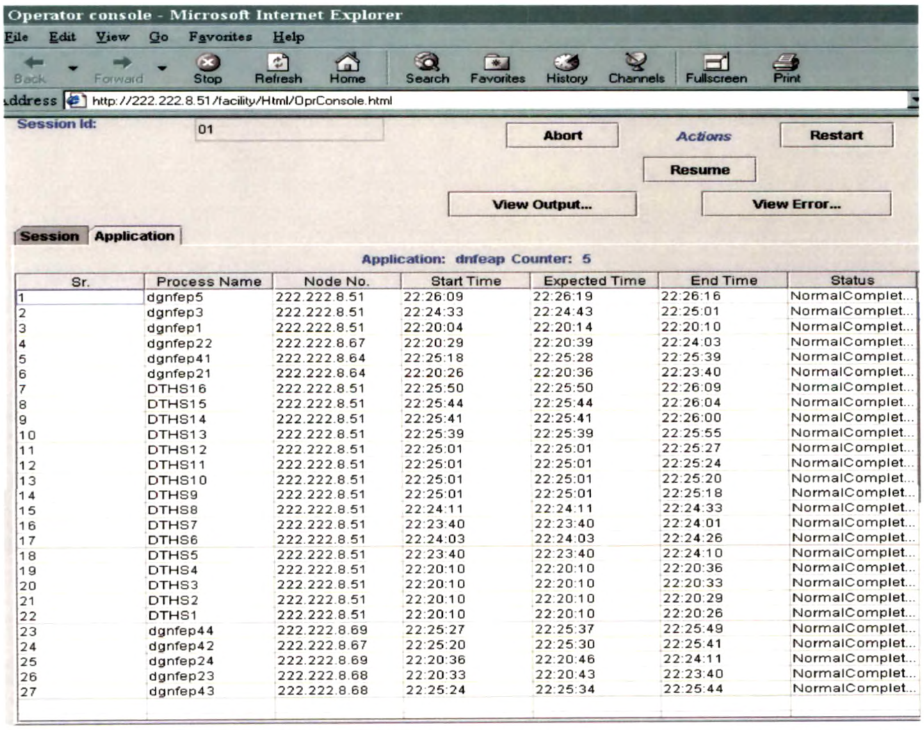
After successful completion of application WebDedip gives the summary of application from which time required for various processes as well as communication time can be known. The screen shot of WebDedip, for summary of application, is shown in Fig. 12.9. The timings have been observed when application was implemented over Local Area Network having Pentium IV computers with 256 MB RAM and running at 1.8 GHz. Similarly after implementation of application over different number of computers, time taken by various process was noted and is reported in Table 12.4. The time taken by different processes and communication time are for one iteration only. For complete nonlinear analysis a number of iterations are required but time required is almost same for each iteration.



Application: c:/InetPub/wwwroot/facility/DedipArea/dnfeap

Grid Size (n*m) : 11

FIG. 12.8 CONFIGURATION OF APPLICATION ON FOUR COMPUTERS



Address http://222.222.8.51/facility/Html/OprConsole.html

Session Id: 01

Application: dnfeap Counter: 5

Sr.	Process Name	Node No.	Start Time	Expected Time	End Time	Status
1	dgnfep5	222.222.8.51	22:26:09	22:26:19	22:26:16	NormalComplet...
2	dgnfep3	222.222.8.51	22:24:33	22:24:43	22:25:01	NormalComplet...
3	dgnfep1	222.222.8.51	22:20:04	22:20:14	22:20:10	NormalComplet...
4	dgnfep22	222.222.8.67	22:20:29	22:20:39	22:24:03	NormalComplet...
5	dgnfep41	222.222.8.64	22:25:18	22:25:28	22:25:39	NormalComplet...
6	dgnfep21	222.222.8.64	22:20:26	22:20:36	22:23:40	NormalComplet...
7	DTHS16	222.222.8.51	22:25:50	22:25:50	22:26:09	NormalComplet...
8	DTHS15	222.222.8.51	22:25:44	22:25:44	22:26:04	NormalComplet...
9	DTHS14	222.222.8.51	22:25:41	22:25:41	22:26:00	NormalComplet...
10	DTHS13	222.222.8.51	22:25:39	22:25:39	22:25:55	NormalComplet...
11	DTHS12	222.222.8.51	22:25:01	22:25:01	22:25:27	NormalComplet...
12	DTHS11	222.222.8.51	22:25:01	22:25:01	22:25:24	NormalComplet...
13	DTHS10	222.222.8.51	22:25:01	22:25:01	22:25:20	NormalComplet...
14	DTHS9	222.222.8.51	22:25:01	22:25:01	22:25:18	NormalComplet...
15	DTHS8	222.222.8.51	22:24:11	22:24:11	22:24:33	NormalComplet...
16	DTHS7	222.222.8.51	22:23:40	22:23:40	22:24:01	NormalComplet...
17	DTHS6	222.222.8.51	22:24:03	22:24:03	22:24:26	NormalComplet...
18	DTHS5	222.222.8.51	22:23:40	22:23:40	22:24:10	NormalComplet...
19	DTHS4	222.222.8.51	22:20:10	22:20:10	22:20:36	NormalComplet...
20	DTHS3	222.222.8.51	22:20:10	22:20:10	22:20:33	NormalComplet...
21	DTHS2	222.222.8.51	22:20:10	22:20:10	22:20:29	NormalComplet...
22	DTHS1	222.222.8.51	22:20:10	22:20:10	22:20:26	NormalComplet...
23	dgnfep44	222.222.8.69	22:25:27	22:25:37	22:25:49	NormalComplet...
24	dgnfep42	222.222.8.67	22:25:20	22:25:30	22:25:41	NormalComplet...
25	dgnfep24	222.222.8.69	22:20:36	22:20:46	22:24:11	NormalComplet...
26	dgnfep23	222.222.8.68	22:20:33	22:20:43	22:23:40	NormalComplet...
27	dgnfep43	222.222.8.68	22:25:24	22:25:34	22:25:44	NormalComplet...

FIG. 12.9 SCREEN SHOT SHOWING SUMMARY OF APPLICATION

TABLE 12.4 COMPARISON OF TIMING FOR SECOND ALTERNATIVE

Number of Substructure	Name of Process	NB	NEQ	Process Time (Sec)	Time for Distributed Computing (Sec)		Sequential Time (Sec)	Speedup
					Comput.	Commun.		
2	DGNFEP1	-	-	6	1178	95	2318	1.82
	DGNFEP2	9565	10265	1100				
	DGNFEP3	1080	1795	25				
	DGNFEP4	565	10265	40				
	DGNFEP5	-	-	7				
3	DGNFEP1	-	-	6	549	99	1569	2.42
	DGNFEP2	6385	6965	480				
	DGNFEP3	960	2150	26				
	DGNFEP4	565	6965	30				
	DGNFEP5	-	-	7				
4	DGNFEP1	-	-	6	278	102	989	2 60
	DGNFEP2	4795	5315	215				
	DGNFEP3	900	2505	28				
	DGNFEP4	565	5315	22				
	DGNFEP5	-	-	7				
6	DGNFEP1	-	-	6	155	108	715	2 72
	DGNFEP2	3205	3665	95				
	DGNFEP3	840	3215	30				
	DGNFEP4	565	3665	17				
	DGNFEP5	-	-	7				

From Table 12.4 it is clear that the maximum time is consumed in formation of tangent stifness matrix and unbalanced load vector of a substructure i.e. process DGNFEP2. As number of substructures increases the tıme taken by process DGNFEP2 reduces. The reason for more time is the bandwidth. As in statıc condensation ınternal nodes are numbered first and boundary nodes are numbered last, bandwith of stifness matrix increases. Further, as the number of substructure ıncreases, computation time reduces and communıcatıon time ıncreases, but overall time to complete one iteratıon reduces. Comaprison of computation and communication time is presented in Fig. 12.10.

The speedup whıch is the ratıo of sequential tıme to parallel tıme do not increases much after four substructures or computers for the sıze of problem considered here. Fig. 12.11 shows comparison of ideal and observed speedup for different number of computers.

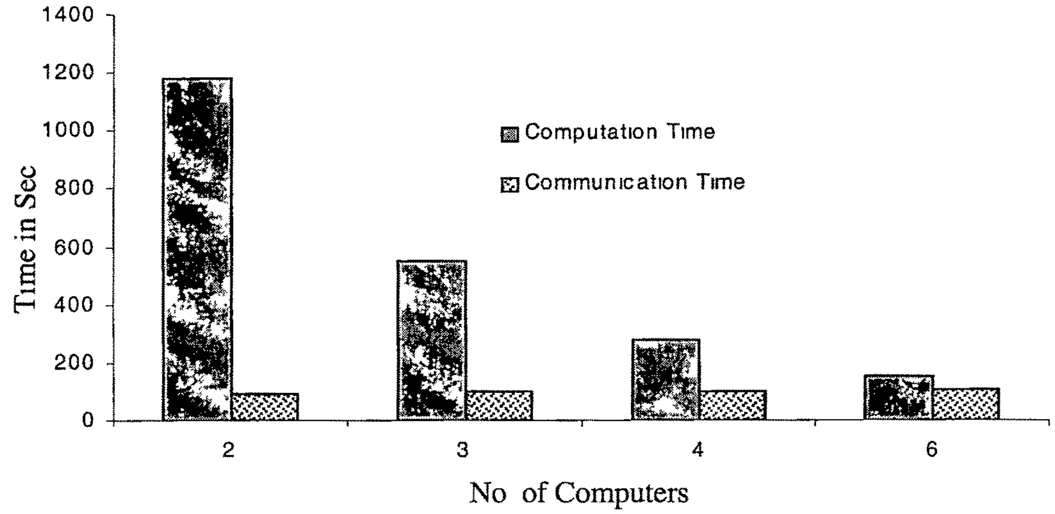


FIG. 12.10 COMPUTATION AND COMMUNICATION TIME

For calculation of speedup, sequential time is obtained when all substructures are solved using one computer and time for distributed computing is obtained by adding computation and communication time. Here each substructure is assigned to one separate computer but if number of substructures are larger than number of computers available, more than one substructure can be assigned to one computer. More speedup can be achieved by keeping computation time larger compared to communication time.

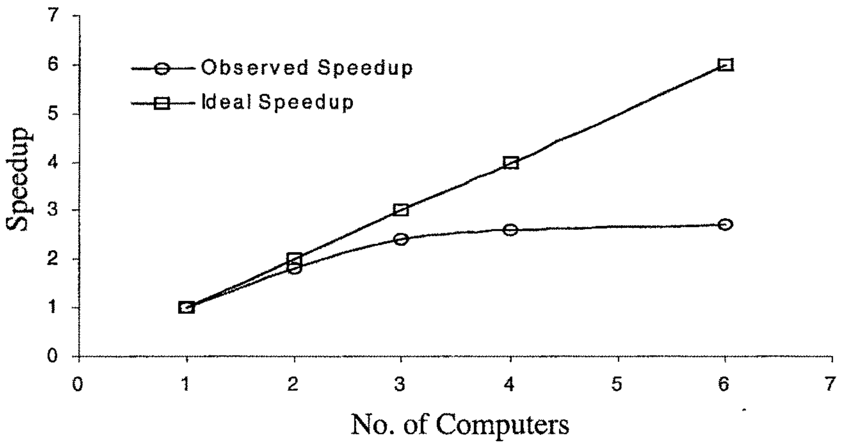


FIG. 12.11 COMPARISON OF SPEEDUP

As communication time increases overhead, speedup with more number of computer reduces. With increas in computers, communication time approaches computation time and further with more number of computers communication time will be greater than computation time and subsequently computational efficiency will decreases. For still larger size of problem more speedup may be observed with increasing computers.

If time required for one iteration is considered for both the strategy as given in Table 12.3 and 12.4, it is observed that, when three computers are used communication time in first stragegy is more while computation time in second strategy is more. This is due to large size of intermediate file required to be transferred in first strategy while more time required for calculating tangent stifness matrix with large bandwidth in second strategy. But this observation may change for still larger size problem. So, for smaller size problem first strategy may be more suitable while for large size problem second strategy may prove better. However, combination of both the strtegy may give more computational efficiency. In combination of both the strtegy, entire finite element mesh can be divided into subdomain and for each domain calculation of unbalanced load vector, linear, nonlinear and initial stress stiffness matrices is carried out in parallel. This strategy is depicted in Fig. 12.12.

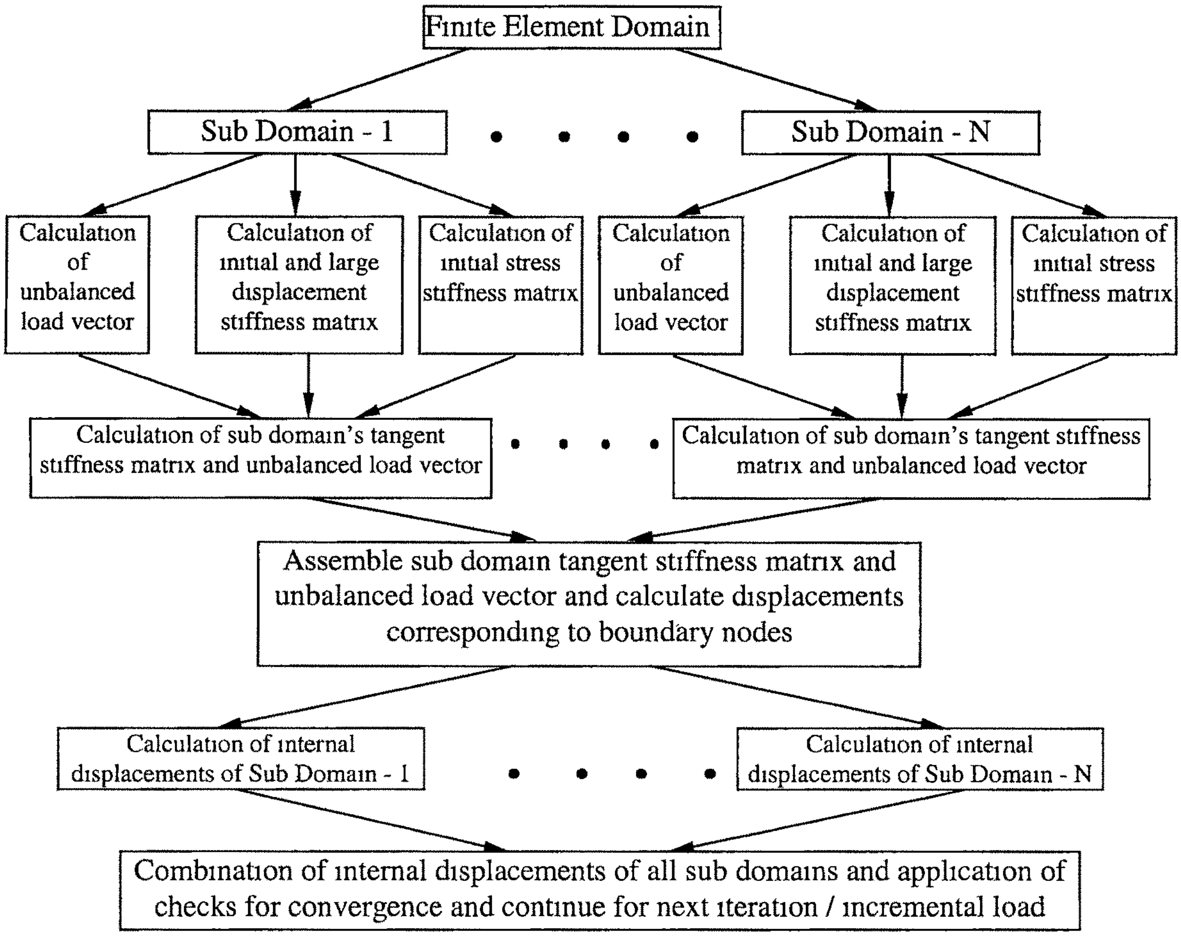

Finite Element Domain
Sub Domain - 1
• • • •
Sub Domain - N
Calculation of unbalanced load vector
Calculation of initial and large displacement stiffness matrix
Calculation of initial stress stiffness matrix
Calculation of unbalanced load vector
Calculation of initial and large displacement stiffness matrix
Calculation of initial stress stiffness matrix
Calculation of sub domain's tangent stiffness matrix and unbalanced load vector
• • • •
Calculation of sub domain's tangent stiffness matrix and unbalanced load vector
Assemble sub domain tangent stiffness matrix and unbalanced load vector and calculate displacements corresponding to boundary nodes
Calculation of internal displacements of Sub Domain - 1
• • • •
Calculation of internal displacements of Sub Domain - N
Combination of internal displacements of all sub domains and application of checks for convergence and continue for next iteration / incremental load

FIG. 12.12 COMBINED STRATEGY FOR DISTRIBUTED NONLINEAR ANALYSIS

12.6 SUMMARY

In this chapter geometric nonlinear finite element analysis was implemented over distributed computing environment using Local Area Network and WebDedip. Two strategies were considered. In first strategy, calculation of unbalanced load vector, initial, large displacement and initial stress stiffness matrices for entire structure was carried out in parallel on different computers. In second strategy, entire structure was divided into number of substructures and calculation of tangent stiffness matrix and unbalanced load vector corresponding to boundary nodes was carried out in parallel on different computers and after calculation of displacement of boundary nodes, calculation of internal displacements was done in parallel. Second strategy proved more efficient and scaleable for large size problem. With different number of computer an efficiency of about 60 to 80% was observed when second strategy was implemented on distributed computing environment of WebDedip.