5. DISTRIBUTED MATRIX ANALYSIS OF SKELETAL STRUCTURES

5.1 GENERAL

Analysis of large size complex structures such as multistoried buildings, long span bridges and tall towers involve formulation and solution of large number of equations. A problem that can arise during analysis of large structures is that the computer may not have sufficient memory to store and process information about the entire structure. A commonly used approach to circumvent this problem is to condense the structure's stiffness equations that are to be solved simultaneously by suppressing some of the degrees of freedom. This process is referred to as static condensation. For very large structures, it may become compulsory to combine condensation with another process called substructuring [100], in which the structure is divided into parts called substructures with the condensed stiffness relations for each substructure generated separately. These are then combined to obtain the stiffness relations for the entire structure.

For the analysis of such large structures, involving large computer time, further, to speed up the analysis time, the parallel processing approach can be used. However a hardware system, dedicated to parallel processing, is expensive. The economical alternative is parallel processing over a network of workstations, which can be visualized as a distributed computing environment for tackling each substructure on a separate workstation using message passing functions [101] for the communication of data between the computers. Distributed computing is a variation of parallel computing in which loosely coupled processors cooperate to solve a problem. Such networks consist of a number of computers that can communicate over the network, but that are otherwise independent. For example, although a program on any given processor has its own data segment, messages can be passed between programs, enabling a kind of cooperative problem solving. However, such communication should be minimized because of relatively high network latencies.

As use of message passing functions complicates programming, use of client server approach can be made without any difficulty of knowing about operating systems, syntax of message passing functions and theoretical background of parallel processing.

In the present chapter feasibility study of distributed computing for analysis of skeletal structure is carried out. The parallel processing requirement is fulfilled using WebDedip, which helps the user to develop his parallel application easily over a network of heterogeneous systems. In this environment the user needs not to use any message passing function in program as is generally required in various available distributed processing environments. In WebDedip the user has to assign various processes to different computers and he has to define communication path between various processes using preprocessor. The communication or message passing between computers is then automatically taken care of by WebDedip. Thus, the user having no theoretical background of parallel processing can also develop application using this environment with small modifications in the program as discussed in Chapter 4.

A problem of static analysis of microwave tower is attempted here using distributed processing concept by dividing the structure into a number of substructures. A program is developed in VC++ based on direct stiffness method. The work of generation of overall stiffness matrix and load vector for different substructures is distributed to different processors and thus each substructure is analysed in parallel on different computers. Finally results are combined to get final displacements and member forces.

5.2 SUBSTRUCTURE TECHNIQUE FOR MATRIX ANALYSIS OF STRUCTURE

The method of substructuring for static structural analysis is based on subdividing the large structure into smaller parts, known as substructures, to obtain the relationship between forces and displacements at the common interfaces or boundaries. These boundary variables are then determined and are used to obtain the unknowns within each substructure. The division of the structure into smaller parts is totally left to analyst, but it will affect the communications between the computer and subsequently efficiency of computation.

The substructure technique has been in use since earlier generations of computers, when the computers were not having large memory and faster processors. To solve large size problem using such machines, to reduce memory consumption and to increase computational speed by reducing number of unknowns, substructure technique was used. In present era of supercomputers also the substructure technique is having its own relevance because of its parallel implementation potential. The computation corresponding to substructures can be carried out on different computers concurrently and solution for boundary degrees of freedom can be carried out sequentially. In this section basic formulation of substructure technique is given in detail [102].

In displacement formulation for structural analysis the basic equation used is the equilibrium equation applied to the structure as a whole and is given by

$$[K]\,\{r\} = \{P\} \qquad \ldots (5.1)$$

where [K] is the stiffness matrix, {r} is the displacement vector and {P} is the load vector.

Using substructure technique the above equilibrium equation is obtained by the assemblage of substructure equations. For each substructure, the stiffness matrix, the displacement vector and the load vector are partitioned corresponding to internal and boundary degrees of freedom $\{d_i\}$ and $\{d_b\}$ respectively as follows:

$$\begin{bmatrix} [k_{ii}] & [k_{ib}] \\ [k_{bi}] & [k_{bb}] \end{bmatrix} \begin{Bmatrix} \{d_i\} \\ \{d_b\} \end{Bmatrix} = \begin{Bmatrix} \{Q_i\} \\ \{Q_b\} \end{Bmatrix} \qquad \ldots (5.2)$$

In the above equation, a boundary node is defined as a node, which is a part of more than one substructure, and the degrees of freedom at the boundary nodes are termed as boundary degrees of freedom.

Now the analysis can be performed in two stages,

1. Considering degrees of freedom at the boundaries as fixed, each substructure is analysed on different computers in parallel. Denote the solution obtained from this step by a superscript α.
2. Combine the condensed stiffness of the substructures from different computers to get the global structure stiffness matrix and analyse the

assemblage by releasing the boundary degrees of freedom. Denote the processing carried out in this step by a superscript β.

The displacement and load vectors can now be expressed as the sum of above two cases as,

$$\begin{Bmatrix} \{d_i\} \\ \hline \{d_b\} \end{Bmatrix} = \begin{Bmatrix} \{d_i^\alpha\} \\ \hline \{d_b^\alpha\} \end{Bmatrix} + \begin{Bmatrix} \{d_i^\beta\} \\ \hline \{d_b^\beta\} \end{Bmatrix} \qquad \ldots (5.3)$$

and

$$\begin{Bmatrix} \{Q_i\} \\ \hline \{Q_b\} \end{Bmatrix} = \begin{Bmatrix} \{Q_i^\alpha\} \\ \hline \{Q_b^\alpha\} \end{Bmatrix} + \begin{Bmatrix} \{Q_i^\beta\} \\ \hline \{Q_b^\beta\} \end{Bmatrix} \qquad \ldots (5.4)$$

where subscript i and b denote the terms corresponding to the internal and boundary degrees of freedom respectively. Obviously as $\{d_b^\alpha\}$ is the displacement at the boundary degrees of freedom, when boundaries are fixed it will be zero. Thus

$$\{d_b^\alpha\} = \{0\} \qquad \ldots (5.5)$$

Also in the first stage of the analysis, all the forces are applied at the internal nodes of the substructure and hence these forces do not appear at the second stage. Hence,

$$\{Q_i^\beta\} = \{0\} \text{ and } \{Q_i^\alpha\} = \{Q_i\} \qquad \ldots (5.6)$$

STAGE I - ANALYSIS WITH FIXED BOUNDARIES

Substituting the value of $\{d_b^\alpha\} = \{0\}$ from Eq. 5.5 into the equilibrium Eq. 5.2, the set of equations for the first stage of analysis with boundaries of substructure fixed can be written as,

$$\left[\begin{array}{c|c} [k_{ii}] & [k_{ib}] \\ \hline [k_{bi}] & [k_{bb}] \end{array}\right] \begin{Bmatrix} \{d_i^\alpha\} \\ \hline \{0\} \end{Bmatrix} = \begin{Bmatrix} \{Q_i\} \\ \hline \{Q_b^\alpha\} \end{Bmatrix} \qquad \ldots (5.7)$$

Solving the first set of above equation ,

$$\{d_i^\alpha\} = [k_{ii}]^{-1} \{Q_i\} \qquad \ldots (5.8)$$

Substituting the value of $\{d_i^\alpha\}$ in the second equation

$$\{Q_b^\alpha\} = [k_{bi}][k_{ii}]^{-1} \{Q_i\} \qquad \ldots (5.9)$$

Here $\{Q_b^\alpha\}$ is the force required to be applied at the substructure boundaries to keep the boundary displacements equal to zero. The above analysis is performed on all the substructures in parallel on different computers.

STAGE II - ANALYSIS WITH BOUNDARIES RELEASED

Again substituting the value of $\{Q_b^\beta\}$ in Eq. 5.2, the set of equations for the second stage of analysis with boundaries released can be written as,

$$\begin{bmatrix} [k_{ii}] & [k_{ib}] \\ \hline [k_{bi}] & [k_{bb}] \end{bmatrix} \begin{Bmatrix} \{d_i^\beta\} \\ \hline \{d_b^\beta\} \end{Bmatrix} = \begin{Bmatrix} \{0\} \\ \hline \{Q_b^\beta\} \end{Bmatrix} \quad \ldots (5.10)$$

Solving the first set of above equations,

$$\{d_i^\beta\} = -[k_{ii}]^{-1} [k_{ib}]\{d_b^\beta\} \quad \ldots (5.11)$$

Solving the second set of equations,

$$[k_{bi}]\{d_i^\beta\} + [k_{bb}]\{d_b^\beta\} = \{Q_b^\beta\} \quad \ldots (5.12)$$

Substituting from Eq. 5.11 for $\{d_i^\beta\}$ into Eq. 5.12

$$-[k_{bi}][k_{ii}]^{-1} [k_{ib}]\{d_b^\beta\} + [k_{bb}]\{d_b^\beta\} = \{Q_b^\beta\} \quad \ldots (5.13)$$

or $$[k^*] \{d_b^\beta\} = \{Q_b^\beta\} \quad \ldots (5.14)$$

where $$[k^*] = [k_{bb}] - [k_{bi}][k_{ii}]^{-1} [k_{ib}] \quad \ldots (5.15)$$

The Eq. 5.15 is the equilibrium equation for the substructure in terms of its boundary degrees of freedom and [k*] is the corresponding stiffness matrix called as condensed stiffness matrix or substructure stiffness matrix. This analysis can be carried out in parallel for all the substructures on different computers and the condensed stiffness matrix for each substructure can be assembled to form the global structure stiffness matrix. Thus,

$$[K] = \sum_{s=1}^{s=n} [k^*]_s \quad \ldots (5.16)$$

and $$\{P\} = \{Q_b\} - \sum_{s=1}^{s=n}\{Q_b^\alpha\}_s \quad \ldots (5.17)$$

In the above equations n stands for the number of substructures, which is equal to the number of computers. The assemblage of the substructures through Eq. 5.16 and 5.17 leads to Eq. 5.1 where all the degrees of freedom are along the common boundaries of the substructures. Solution of Eq. 5.1 gives the global displacements along the boundaries of the substructure. Now, picking up the appropriate displacements, the vector $\{d_b^\beta\}$ can be obtained for each substructure, which can be communicated to different computers and from that $\{d_i^\beta\}$ can be determined using Eq. 5.11.

Thus all the values of {d} required in Eq. 5.3 are known for each substructure and from that other quantities like member end forces can be calculated.

The substructure technique for the analysis of large structures is essentially the same as the standard stiffness method. However, each substructure is treated as an ordinary member of structure and the degrees of freedom of only those joints through which the substructures are connected to each other and / or to supports are considered to be the structure's degrees of freedom. The structure's stiffness matrix and load vector are assembled, respectively, from the substructure stiffness matrices and fixed joint force vectors, which are expressed in terms of external coordinates of substructures only. The equations thus obtained can then be solved for the joint displacements. Some of advantages of analysis of structure using substructure technique are number of unknown displacements to be found at any time will be considerably less and all substructures can be analysed in parallel on different computers. It is also observed that during this process there is no loss of accuracy in the final results. The division of the structure into smaller parts is totally left to analyst, but it will affect the communication time between the computers and subsequently efficiency of computation.

5.3 SUBSTRUCTURING IN DISTRIBUTED ENVIRONMENT

The substructure technique, for the analysis of large structures, can be implemented easily over distributed computing environment. In this process some of the calculations can be carried out in parallel, while some are to be carried out sequentially. Stiffness matrix (Eq. 5.15) and load vector (Eq. 5.9) of substructures are calculated in parallel on different computers. The stiffness matrix and load vector of all substructures are assembled (Eq. 5.16 and 5.17) on one computer and displacements corresponding to boundary degrees of freedom are calculated sequentially. Subsequently from boundary degrees of freedom internal displacements (Eq. 5.11), total displacements of all nodes of substructure (Eq. 5.3) and member end actions of all members of substructures are calculated in parallel. Finally results of all substructures are combined on one computer. The above process is represented graphically in Fig. 5.1.

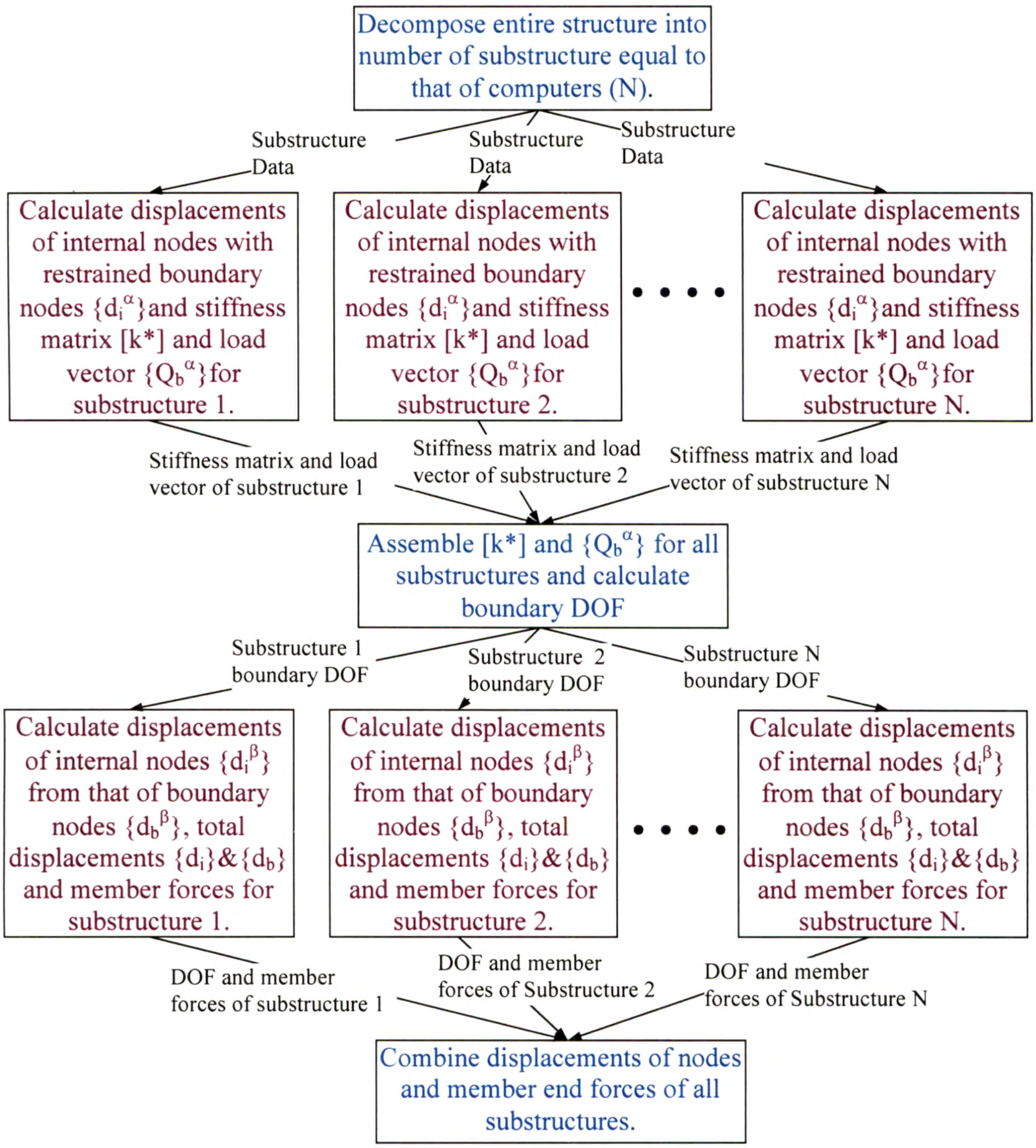


FIG. 5.1 SUBSTRUCTURING IN DISTRIBUTED COMPUTING ENVIRONMENT

In distributed implementation communication between computers takes place four times. In first communication data corresponding to substructures are distributed to different computers. In second, computers communicate stiffness matrix and load vector of substructure to have total equilibrium equations corresponding to boundary nodes. Thirdly, displacements of boundary nodes are distributed to appropriate computers for calculation of total displacements and member end actions. Finally, in fourth communication, results of substructures are collected from different computers. As communication time and time spent for sequential calculations reduce computational efficiency, they should be kept minimum. The division of structure is to be done in such a way that each substructure should have equal number of members (as far as possible) for equal computational loads among computers and number of boundary nodes should be minimum for less communication and minimum sequential computation.

In the WebDedip environment, the whole process is subdivided in to five sub tasks. The first sub task (parant1.c) decomposes the whole structure in to substructures and distributes the substructure data to different computers. The second subtask (parant2.c) analyses each substructure separately on different computers, and communicates substructure stiffness matrix and load vector. The third sub task (parant3.c) assembles the substructure stiffness matrix and load vector of all substructures and after imposing actual boundary conditions calculates displacements at boundary nodes and distribute back to different computers for correction of displacements obtained in sub task two. The fourth sub task (parant4.c) calculates actual displacements of the internal nodes of each substructure on separate computers considering displacements at boundary nodes. It also calculates final member end forces. The fifth sub task (parant5.c) collects the final displacements at nodes of substructure and member end forces from different computers and displays the final result. The interface among different tasks is carried out using intermediate files. All the tasks are then inter linked using graphical user interface of DEDIP to configure the application for parallel processing. These programs are not containing any message passing functions and data to be communicated is to be written in files. The syntax of file input and output is slightly modified to suit distributed processing, which is already described in Chapter 4.

Before implementation of above process over distributed environment, it is essential to check the functioning of all programs on single computer. In this process during configuration of application each task is assigned to same computer with appropriate dependency information. As all the tasks are carried out on same computer, communication of data is not required. Successful implementation of application on single computer indicates proper functioning of program as well as proper execution of tasks as per configuration. This procedure makes debugging of the programs, to be implemented over distributed environment, easier otherwise to find error either in program or in communication may be time consuming. Subsequently all tasks are distributed over different computers and communications between them are defined through configuration of application. Fig. 5.2 shows screen shot depicting the required interdependency. The DEDIP GUI is used to provide the information about remote node on which the process is to be executed.

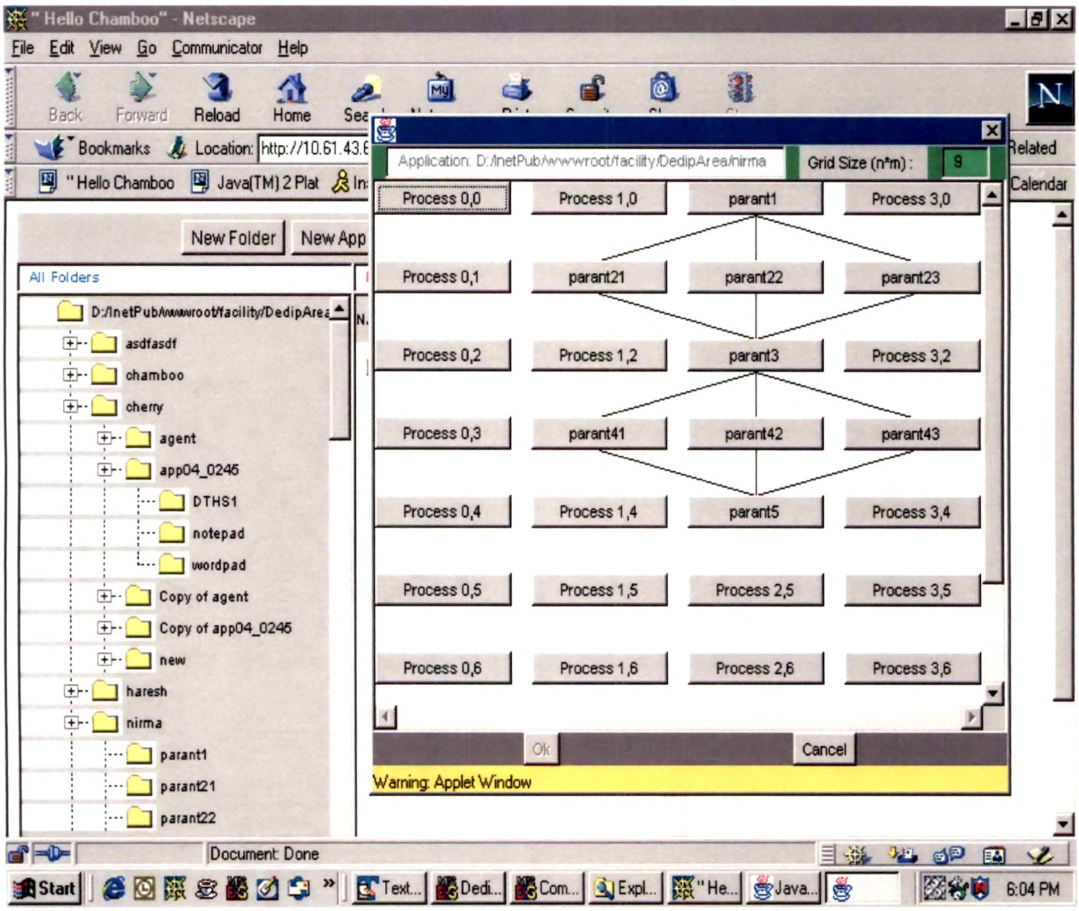


FIG. 5.2 DEDIP GUI FOR CONFIGURATION OF DISTRIBUTED APPLICATION

5.4 EXAMPLE OF STATIC ANALYSIS OF A MICROWAVE TOWER

A microwave tower is modeled as a three dimensional pin jointed structure subjected to various types of loading like dead load, weight of antenna, wind load and earthquake loads. Generally loads are applied at nodes. The stiffness matrix of space truss member with three translational degrees of freedom at a joint can be determined easily by direct approach [103].

$$[k] = \frac{AE}{L}\begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{. . . (5.18)}$$

where E is modulus of elasticity of material, A is cross sectional area, L is length of member

The stiffness can be transformed to structural axis by using rotation transformation matrix as,

$$[k]_s = [R]^T [k] [R] \quad \text{. . . (5.19)}$$

where [R] is rotation transformation matrix and is given by [103].

$$[R] = [R_\gamma] [R_\beta] [R_\alpha] \quad \text{. . . (5.20)}$$

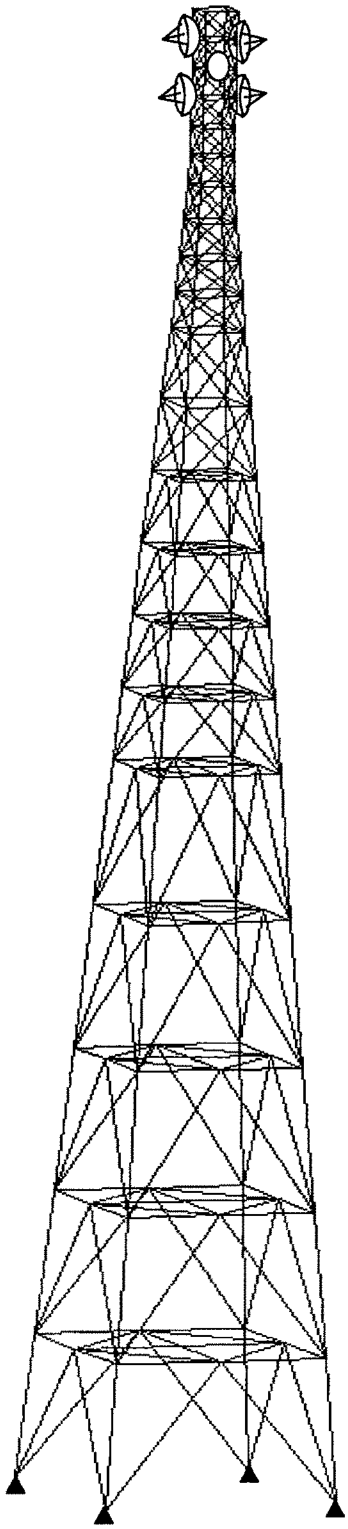
with $[R_\alpha] = \begin{bmatrix} c_x & 0 & c_z \\ 0 & 1 & 0 \\ -c_z & 0 & c_x \end{bmatrix}$ $[R_\beta] = \begin{bmatrix} c_x & c_y & 0 \\ -c_y & c_x & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $[R_\gamma] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_x & c_z \\ 0 & -c_z & c_y \end{bmatrix}$

Where $c_x = (x_k - x_j) / L$, $c_y = (y_k - y_j) / L$, $c_z = (z_k - z_j) / L$

and $L = \sqrt{(x_k - x_j)^2 + (y_k - y_j)^2 + (z_k - z_j)^2}$

(x_j, y_j, z_j) and $x_k, y_k, z_k)$ are the coordinates of the j and k end of the member respectively.

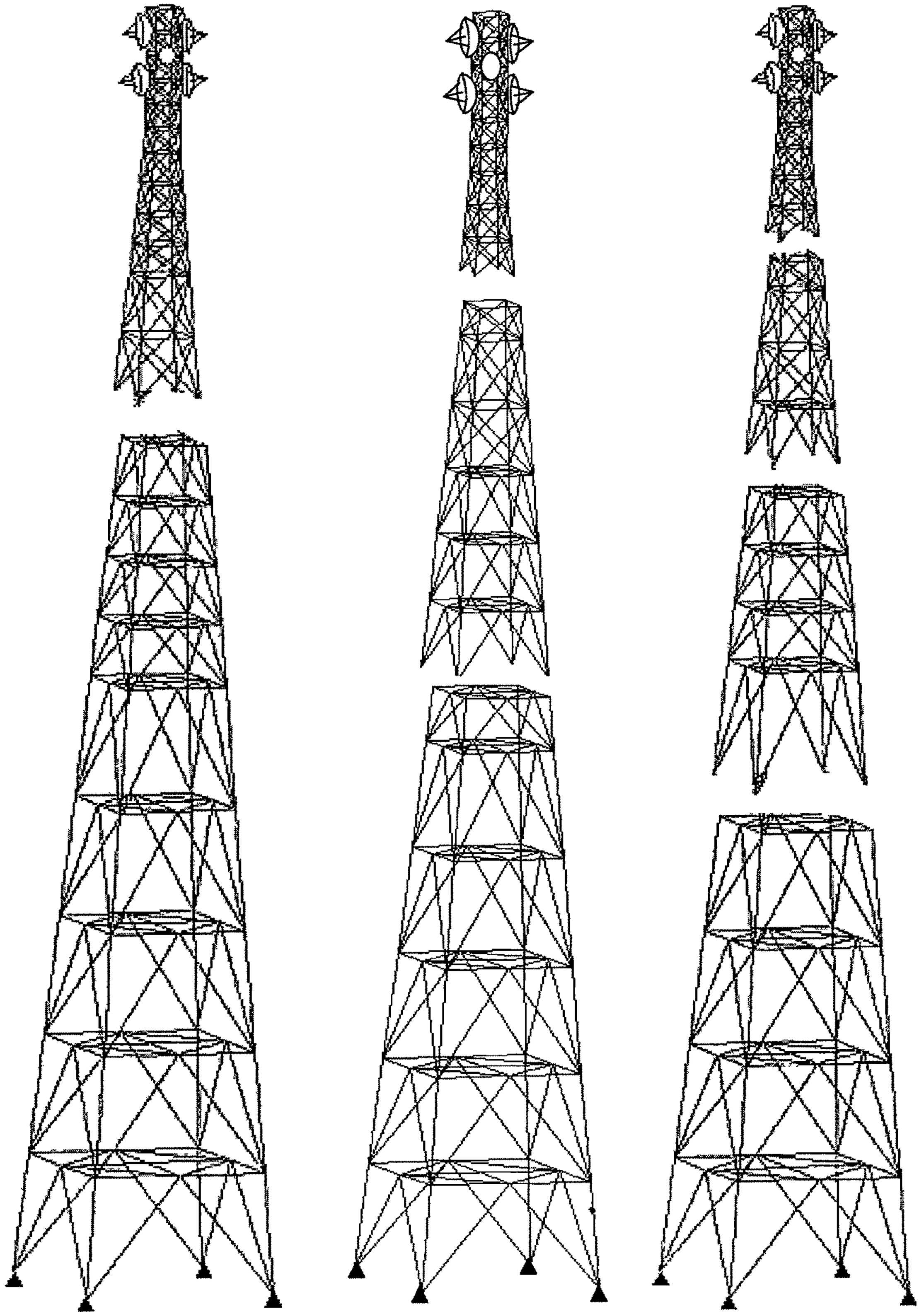
A microwave tower with 102 m height is shown in Fig. 5.3. It is modeled with 426 members and 124 joints. It is analysed for static conditions when subjected to dead load and wind load. The whole structure is divided into two, three and four substructures as shown in Fig. 5.4. On the host computer the structure is divided into substructures (parant1.c) and data files corresponding to each substructure is distributed to slave computers. Analysis of each substructure, including generation of substructure stiffness matrix, is carried out on different slave computers (parant2.c). Then substructure matrix and load vector of each substructure are communicated back from slave to host computer. On the host computer substructure stiffness matrix and load vector corresponding to boundary nodes of all sub substructures are assembled to find the displacements at the boundary nodes (parant3.c). These degrees of freedom are distributed by host computer to different slave computers as per the boundary nodes of each substructure to find the displacements at the internal nodes and forces in each member. After calculating displacements and member end forces (parant4.c) slave computers communicate the results of each substructure to the host computer. Finally displacements of all nodes and member end forces are assembled on host computer (parant5.c) to have final result of analysis.



GEOMETRICAL DETAILS OF THE MICROWAVE TOWER

- Base width 18.00 m
- Top width 2.50 m
- Total tapered height 90.00 m
- Total uniform height 12.00 m
- Types of tapered panels 3
- First five tapered panels of height 10.00 m
- Next six tapered panels of height 6.00 m
- Next four tapered panels of height 2.50 m
- Next six uniform panels of height 2 00 m

FIG. 5.3 3-D VIEW OF MICROWAVE TOWER



(a) 2 - Substructures (b) 3 - Substructures (c) 4 - Substructures

FIG. 5.4 TOWER DIVIDED INTO DIFFERENT NO. OF SUBSTRUCTURES

For the problem under consideration, the results of analysis (nodal displacements, member end forces and reaction at supports) are found in good agreement with that obtained by STAAD software. The maximum joint displacement at top of tower, maximum axial force in bottom panel and reactions at one of the supports are compared in Table 5.1, 5.2 and 5.3 respectively.

TABLE 5.1 COMPARISON OF MAXIMUM JOINT DISPLACEMENT AT TOP

Result obtained from	Dead load			Wind load in right angle direction			Wind load in diagonal direction		
	Disp-X	Disp-Y	Disp-Z	Disp-X	Disp-Y	Disp-Z	Disp-X	Disp-Y	Disp-Z
STAAD/Pro	0.000	-3.602	0.000	363.668	-11.60	0.004	328.522	0.000	-328.54
2 Substructures	0.000	-3.601	0.000	363.656	-11.59	0.003	328.529	0.000	-328.53
3 Substructures	0.000	-3.603	0.000	363.663	-11.60	0.004	328.525	0.000	-328.53
4 Substructures	0.000	-3.598	0.000	363.670	-11.61	0.004	328.523	0.000	-328.54

(Displacements are in mm)

TABLE 5.2 COMPARISON OF MAXIMUM AXIAL FORCE AT BOTTOM

Result obtained from	Dead load	Wind load in right angle direction	Wind load in diagonal direction
STAAD/Pro	190.917 kN	1202.703 kN	2111.529 kN
2 Substructures	190.920 kN	1202.710 kN	2111.525 kN
3 Substructures	190.919 kN	1202.705 kN	2111.535 kN
4 Substructures	190.921 kN	1202.707 kN	2111.530 kN

TABLE 5.3 COMPARISON OF REACTIONS AT ONE SUPPORT

Result obtained from	Dead load			Wind load in right angle direction			Wind load in diagonal direction		
	Rea-X	Rea-Y	Rea-Z	Rea-X	Rea-Y	Rea-Z	Rea-X	Rea-Y	Rea-Z
STAAD/Pro	25.94	219.98	-25.94	-242.83	-1338.2	115.24	-312.3	-2345.0	312.29
2 Substructures	25.96	220.00	-25.96	-242.90	-1338.2	115.26	-312.3	-2346.0	312.30
3 Substructures	25.95	219.91	-25.95	-242.84	-1338.3	115.27	-312 3	-2345.9	312.31
4 Substructures	25.95	219.95	-25.95	-242.86	-1338.2	115.25	-312.2	-2345 9	312.27

(Reactions are in kN)

5.5 DISCUSSION OF RESULTS

In this chapter feasibility of distributed computing in structural analysis was explored using an example of static analysis of microwave tower. Substructure technique, which can be used for distributed implementation of structural analysis, was discussed. From the comparison of results it is observed that for different number of substructures the results are matching closely with that of entire structure. It proves that the accuracy of solution is not affected by number of substructures.

For the example of static analysis of microwave tower, the use of distributed processing concept takes about 18% less computer time compared to sequential processing. As the size of problem considered here is small size, advantage of distributed processing is not very high. Because for small size problems the time required for computation on different computers is not large compared to communication time and sometimes computation time is smaller than communication time. However, for very large size problems having total number of unknowns more than 10000, it is expected to give significant reduction in computer time.