9. INTERNET BASED COMPUTING USING WEBDEDIP

9.1 THE USE OF INTERNET TECHNOLOGY

The developments in computer science and electronics particularly in networking and communications have led to vide usage of Internet. Communications through computer has become an integral feature of our infrastructure. Networking is used in every aspect of business, education and many organizational set up. Internet connects most of the corporations, colleges and universities as well as government offices. Even private residences have access to the Internet through dial up telephone connection and newer technologies are providing even higher capacity services [123]. Most of the organizations are having their in-house network of computers and Internet connectivity.

The Internet is mainly used for collecting information, communication via email, office and project management in case of wide spread organization. The survey of internet uses indicates that 95% of the time is spent in downloading pages and emails, waiting for users to read and write their mails or to read web pages. This effectively means that CPU is idle for most of the time when these activities are taking place. The idle CPU time of computers connected through internet can be used for computation purpose. The effectiveness of computing power can be increased by utilizing idle CPU time for computation.

The development in software engineering particularly using networks has made it possible to use computers connected through internet as distributed computing resources. Also it has become possible to network computing capabilities of various organizations for effective utilization in research and development. The distributed computing using LAN and / or internet can be economical alternative of high performance computers and major advantages are resource sharing, performance enhancement, availability, extensibility, and reliability. The only disadvantage is communication time, which also is continuously reducing with latest developments in communication technology. This type of environment makes it possible to share professional software, where it is uneconomical to invest large amount for limited usage.

In this chapter an attempt is made to exploit the use of computers connected through Internet for structural engineering applications. WebDedip environment is used to implement the applications over internet. Two applications have been demonstrated in the chapter. In first application, use of software on remote computer through internet is demonstrated. In this application data for structure is sent to remote computer through internet where program for analysis is running. After analysis on remote computer results are displayed on computer from where data was sent. In the second application, computers connected through internet are used for distributed analysis of a structure. In this application entire structure is subdivided into number of parts and they are analyzed separately on remote machines and results of each part are combined to have final results. From these applications it is shown that computational resources available on internet can be effectively harnessed for sharing software and for solving large size structural engineering problems.

9.2 IMPLEMENTATION OF WEBDEDIP ENVIRONMENT OVER INTERNET

In chapter 3 basic of some networking technology was covered. It was clear from the discussion that each networking technology is designed to meet with a specific set of constraints. For example LAN technologies are designed to provide high speed communication across short distance. WAN technologies are designed to provide communication across large areas. No single networking technology is best for all needs. A large organization with diverse networking requirements needs multiple physical networks of several types depending on type of task. In many early installations, each computer attached to a single network and employees had to choose a computer appropriate for each task due to which users were neither satisfied nor productive. Now modern computer communication system allows communication between any two computers analogous to the way a telephone system provides communication between any two telephones which is known as universal service concept. Universal service is a fundamental part of networking with which user can communicate to any other computer of organization. As universal service increases individual productivity, it is desirable. Internetworking is the scheme which uses both hardware and software and provides universal service among heterogeneous networks.

Internetworking is quiet general which is not restricted in size. It contains few networks or thousands of networks. Similarly the number of computers attached to each network in an internet can vary - some networks have no computers attached while others have hundreds. A special purpose computer dedicated to task of interconnecting networks is known as router. A router can connect networks that use different technologies, including different media, physical addressing schemes or frame format. An internet consists of a set of networks interconnected by routers. The internet scheme allows each organization to choose the number and type of networks, the number of routers to use to interconnect them and the exact interconnection topology. Although many protocols have been adapted for use in an internet, but TCP/IP- Internet Protocols is the most widely used for internetworking. TCP/IP defines the term host computer to refer to any computer system that connects to an internet and runs application. A host can be as small as a personal computer or as large as a mainframe. Furthermore, a host's CPU can be slow or fast, the memory can be large or small, and the network to which a host connects can operate at high or low speed. TCP/IP protocols make it possible for any pair of hosts to communicate, despite of hardware differences. Both hosts and router need TCP/IP protocol software.

WebDedip was mainly developed for helping the image processing scientist for parallel and distributed processing over a network of heterogeneous system using Java. WebDedip can convert computers connected through internet into distributed computing environment. To implement WebDedip, one computer of the network is configured as a server and DedipServer is to be installed on it. While number of computers are configured as client and DedipAgent is to be installed on them. Installation of DedipServer and DedipAgent was covered in Chapter 4. Communication between computers is done through intermediate files, which are transferred by FTP. So server and client machine should also run FTP which can transfer files from remote to remote. Several FTP software are available with such facility, but in this implementation EFTP server is used. WebDedip is a browser based environment, enabling the user to operate from any node over the internet. The user initiates the interaction by visiting a predefined website using a standard web browser. The standard web server loads the required GUI on the web browser. It has back-end DedipServer running on the web site. Typical configuration of WebDedip over internet is shown in Fig. 9.1.

When any client submits the request to the DedipServer, it reads the application configuration information from the configuration file and accordingly instruct agents running on remote machine to run the specified tasks and agent submits the status when process is completed. It finds out the dependent process on the successful completion of the process and initiates the execution of each such process. The required files are transferred from one node to another. WebDedip has callable library in Java to make interface with FTP server that helps in transferring the files. The required process is automatically inserted in the configuration when IP designer inserts the IO dependency information between two processes.

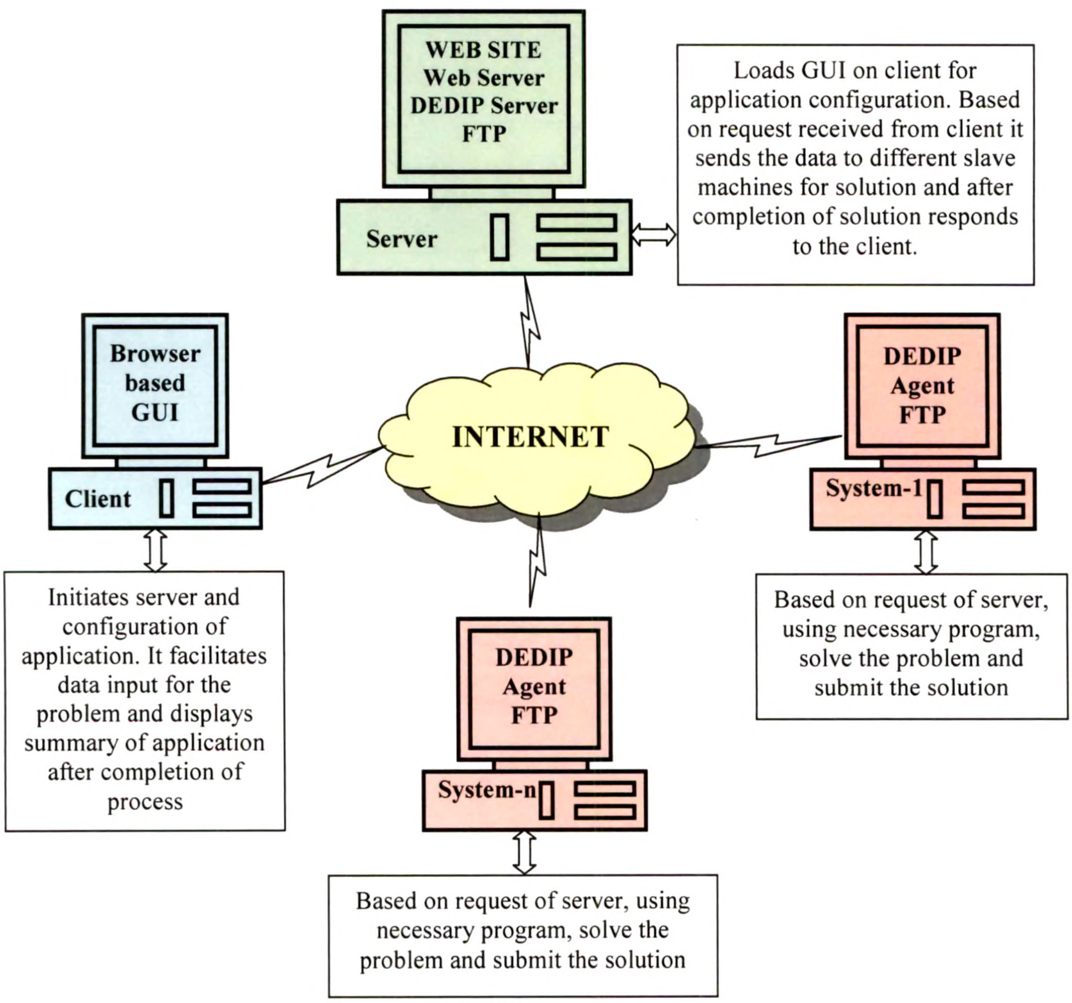


FIG. 9.1 INTERNET BASED COMPUTING USING WEBDEDIP

9.3 ACCESSING SOFTWARE OF REMOTE COMPUTER APPLICATION

In this application two computers are connected through internet. While keeping one computer in the institute as a part of LAN the other computer is connected through telephone landline. Computer connected through LAN is generally protected by proxy server and can not be accessed by other outside computer. So to put it on internet in place of local IP address a live IP address, provided by Internet Services Provider, is to be assigned. The computer on telephone line is assigned dynamic IP address, each time when it connects to internet. Once IP addresses of all computers, to be used for distributed computing are known, configuration of Dedip and DedipArea is to be carried out by specifying IP addresses of server and slave machines as discussed in chapter 4. While DedipServer, standard Web Server and FTP are running on a computer of the institute on the other computer DedipAgent and FTP are processed.

After setting up distributed computing environment, application is to be configured through any client or slave machine through JAVA applet running on server. The configuration of application consists of specifying the process information i.e. IP address of machine on which it is running as well as process on which it is depending and dependent process. It also consists of data transfer information i.e. name of source and destination processes. Various JAVA applets are available for entering this information. The screen shot for configuration of application is shown in Fig. 9.2.

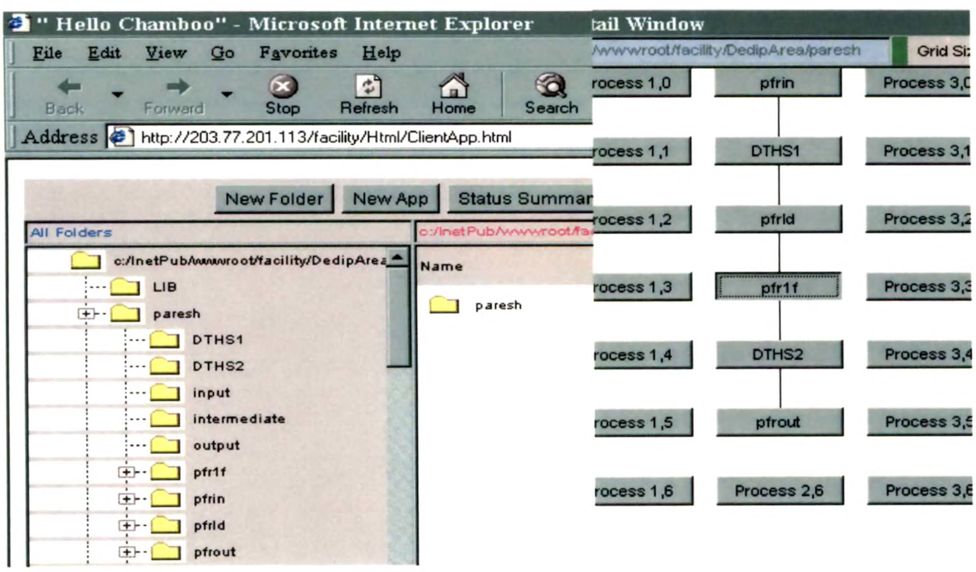


FIG 9.2 SCREEN SHOT FOR CONFIGURATION OF APPLICATION

The application consists of various processes like PFRIN, PFRLD, PFR1F and PFROUT. The PFRIN process copy the data files corresponding to geometry and load in to frame.i and load.i in DedipArea from which they are transferred to remote computer. PFRLD process calculates the load vector from geometrical and load data. PFR1F process analyse the frame and store the analysis results into a file frame.d, which is transferred back to computer from where data is supplied. PFROUT process copy the output file of DedipArea as received from remote computer to the desired folder.

In this implementation while PFRIN and PFROUT processes are running on slave computer (IP 61.1.50.136) connected to internet through telephone line, the PFRLD and PFR1F processes are running on server computer (IP 203.77.201.113). The screen shots for detailed process information are depicted in Fig. 9.3.

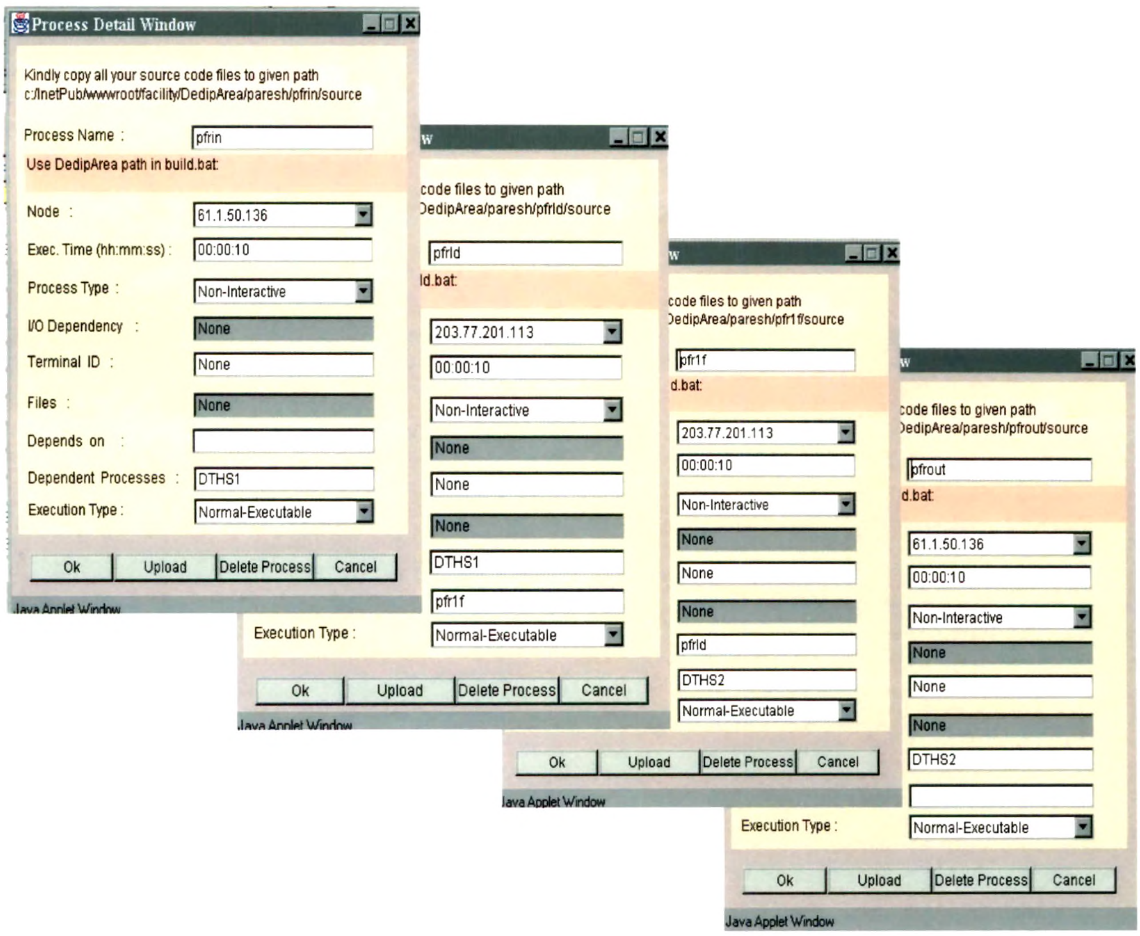


FIG 9.3 SCREEN SHOTS SHOWING DETAILED PROCESS INFORMATION

The data transfer takes place twice. The input data of structure is transferred from slave computer to server and output file consisting of analysis result is transferred from server to slave computer. The screen shots for specifying data transfer information are shown in Fig. 9.4.

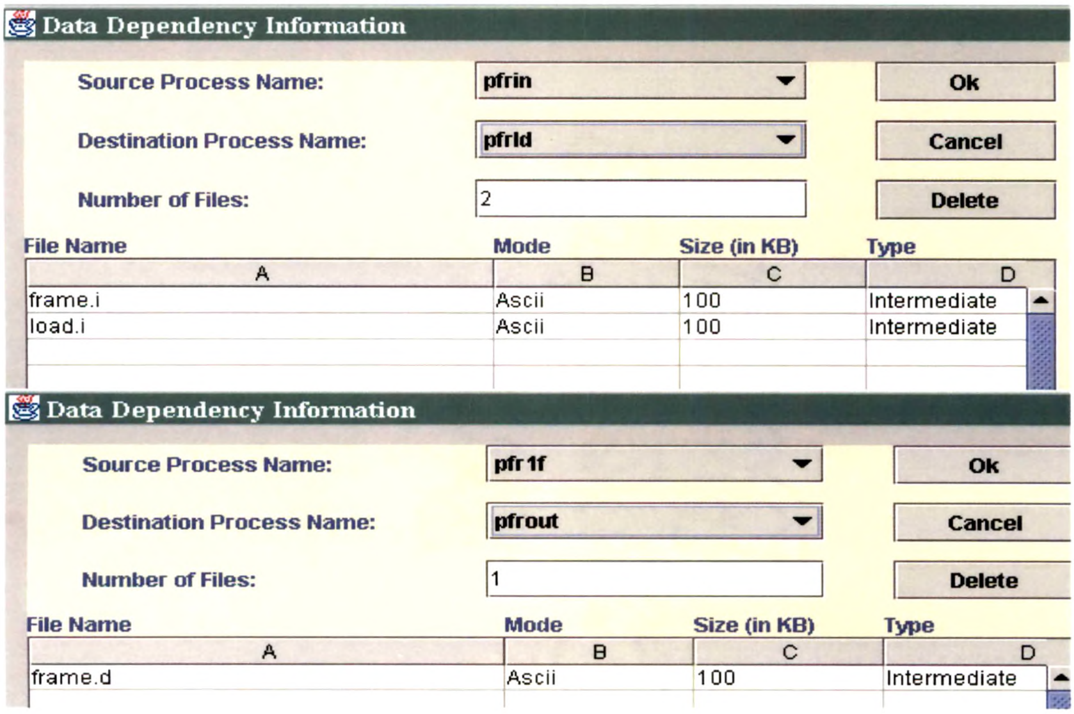
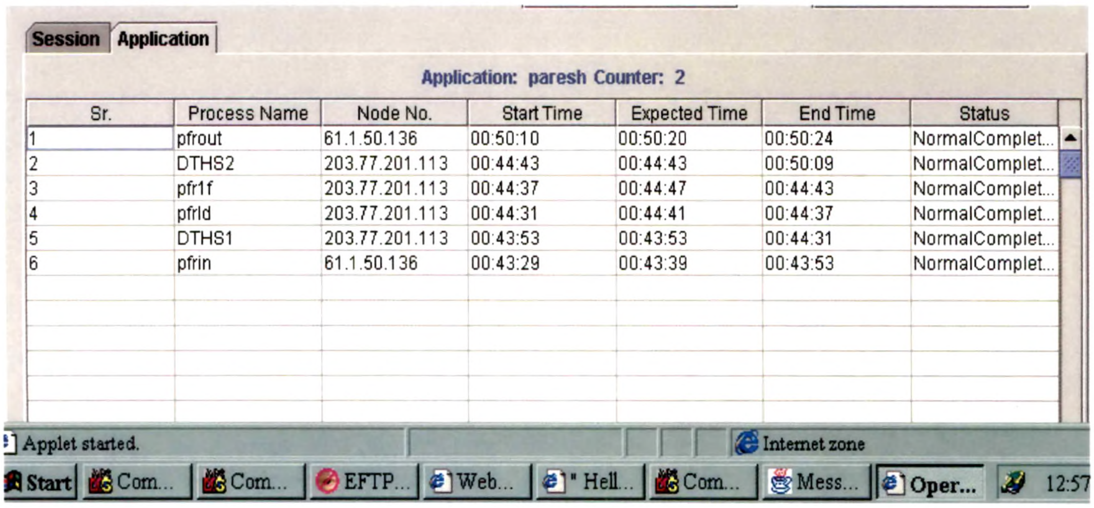


FIG 9.4 SCREEN SHOTS SHOWING DATA TRANSFER INFORMATION

After configuration, when application is run, the server starts the first process as mentioned in the configuration. As it is to be run on remote slave computer, server instructs agent running on that computer to start the process. After completion of first process the next process of configuration DTHS1 starts. DTHS is short form of Data Transfer from Host to Slave. In this process the specified files (frame.i and load.i) are transferred from slave computer to server using FTP. After successful receipt of files on server the next process PFRLD starts and after its completion PFR1F starts and finally it writes output file consisting of analysis results. This output file is transferred through process DTHS2 to computer from where input data was sent. Finally, after successful completion of PFROUT process the application stops. The server gives summary of applications i.e. IP address on which process executed, start time, end time and status. The screen shot showing summary of application is shown in Fig. 9.5. If any error occurs in running process on any computer or in data transfer, the same can be viewed from this JAVA applet.



Sr.	Process Name	Node No.	Start Time	Expected Time	End Time	Status
1	pfrout	61.1.50.136	00:50:10	00:50:20	00:50:24	NormalComplet...
2	DTHS2	203.77.201.113	00:44:43	00:44:43	00:50:09	NormalComplet...
3	pfr1f	203.77.201.113	00:44:37	00:44:47	00:44:43	NormalComplet...
4	pfrld	203.77.201.113	00:44:31	00:44:41	00:44:37	NormalComplet...
5	DTHS1	203.77.201.113	00:43:53	00:43:53	00:44:31	NormalComplet...
6	pfrin	61.1.50.136	00:43:29	00:43:39	00:43:53	NormalComplet...

FIG 9.5 SCREEN SHOT SHOWING APPLICATION SUMMARY

9.4 PLANE FRAME ANALYSIS EXAMPLE

To demonstrate the application of Internet in accessing software of remote computer an example of plane frame analysis is considered. As the main objective of this example is to explore the feasibility of utilizing resources available on Internet, the size of problem considered is very small. A plane frame having two bays and two storey subjected to loads as shown in Fig. 9.6 is chosen.

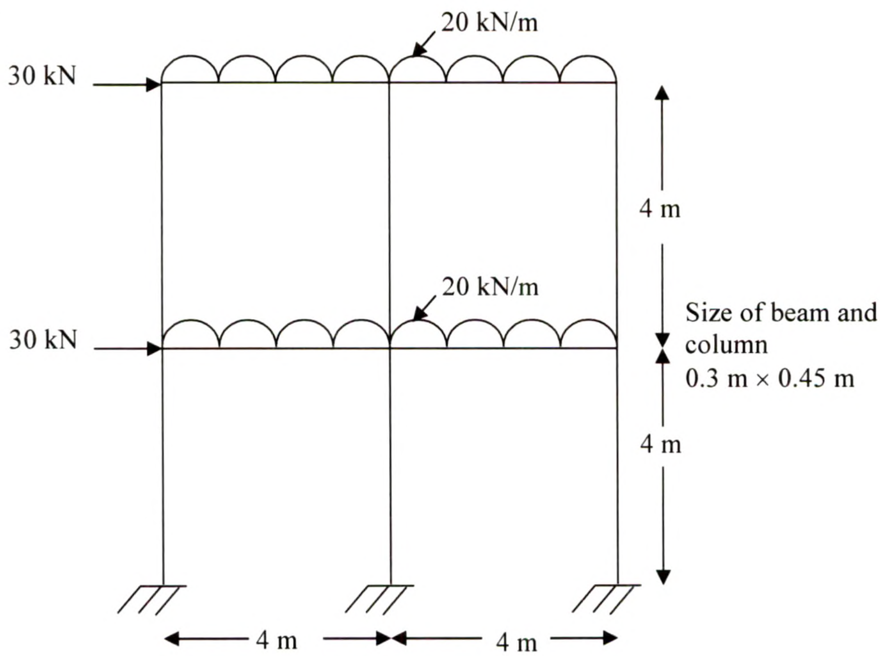


FIG. 9.6 PLANE FRAME EXAMPLE

As discussed in the configuration of application the geometrical and load data of plane frame example is on the remote machine. After starting the application this data is transferred from remote / slave machine to server machine. After receipt of data the server machine starts the process 'pfrld', which calculates the load vector and subsequently the process 'pfr1f' is started to analyze the structure. After completion of analysis the analysis results are sent to remote machine from where the data was sent. After receipt of output on remote machine, the results are verified with that of STAAD software. The time spent in various processes, as can be seen from screen shot (Fig. 9.5), is given in Table 9.1.

TABLE 9.1 TIME CONSUMED IN VARIOUS PROCESSES

Sr. No.	Process	Function	Carried out at	Time consumed
1	pfrin	Copy the data files in DedipArea	Remote machine	24 Sec
2	DTHS1	Transfer data files from remote to server machine	Server machine	38 Sec
3	pfrld	Calculate load vector	Server machine	6 Sec
4	Pfr1f	Analyze plane frame	Server machine	6 Sec
5	DTHS2	Transfer output of analysis from server to remote machine	Server machine	326 Sec
6	pfrout	Copy output file from DedipArea to user directory	Remote machine	5 Sec

From Table 9.1 it can be seen that the time required for completing the analysis is not much compared to time spent for data transfer. Particularly the time spent in transfer of output file from server to remote machine is maximum. One of the reasons is communication traffic or network congestion. As the same communication channel is utilized by number of users simultaneously the time required for successful transfer of data can be more. But due to advances in communication technology this problem may be resolved in future.

9.5 DISTRIBUTED COMPUTING APPLICATION

To implement distributed computing in structural analysis, substructure technique is generally used as discussed in Chapter 5 and 6. In present study distributed finite element analysis is implemented over computers connected through internet. One computer in the institute with live IP address and two computers connected through telephone line are used for the implementation. The server (IP 203.77.201.113) which is in institute is running web server, DedipServer, FTP, DedipAgent and on two slave machine (IP 61.1.49.132 and 61.1.50.136) DedipAgent and FTP are running. After configuring Dedip and DedipArea, distributed application is configured as described in earlier application. This application consists of five processes like CSUB1 to CSUB5. CSUB1 is running on server and it divides the entire structure into number of substructures and prepares data files corresponding to each substructure. These data files are transferred to slave computers using FTP. CSUB2 prepares substructure stiffness matrix and load vector corresponding to boundary DOF for each substructures in parallel on different slave computers. These matrices are transferred to server. CSUB3 assembles stiffness matrices and load vectors of all substructures to form overall matrices, the solution of which yields the displacements corresponding to boundary DOF for all substructures. CSUB3 is running on server. The displacements corresponding to boundary DOF are communicated to slave computers. CSUB4 calculates the internal displacements and element stresses based on boundary DOF and loading condition in parallel on different slave computers. These displacements and element stresses of all substructures are communicated to server to have overall results of entire structure. CSUB5, running on server, collects joint displacements and element stresses of all substructures and prepares overall analysis results for entire structure. The entire process with data communication is shown in Fig. 9.7.

After configuration and uploading required data file and programs the application is run. DedipServer reads the configuration file and initiates the first process and after its completion next process of configuration is started. The process can be either a program to perform different tasks on various computers or it can be data transfer. If any process is to be carried out on remote computer, server instructs the agent running on it for the same. Server coordinates all the tasks and waits for the processes to be completed on slave computers to start the next process. If any error occurs during execution of any program or in communication, server reports the same. After successful completion of application, DedipServer gives the summary of application. From summary the time required in computation and communication can be obtained.

To implement the distributed finite element analysis over computers connected through internet, an example of deep beam as discussed in section 6.7 is selected. The finite element mesh is divided into two substructures and analysis is carried out as shown in Fig. 9.7. The results are verified by those reported by Krishnamoorthy [102]. In this implementation also it is observed that the time spent in various processes for analysis is less compared to communication time spent in data transfer between server and slave machines.

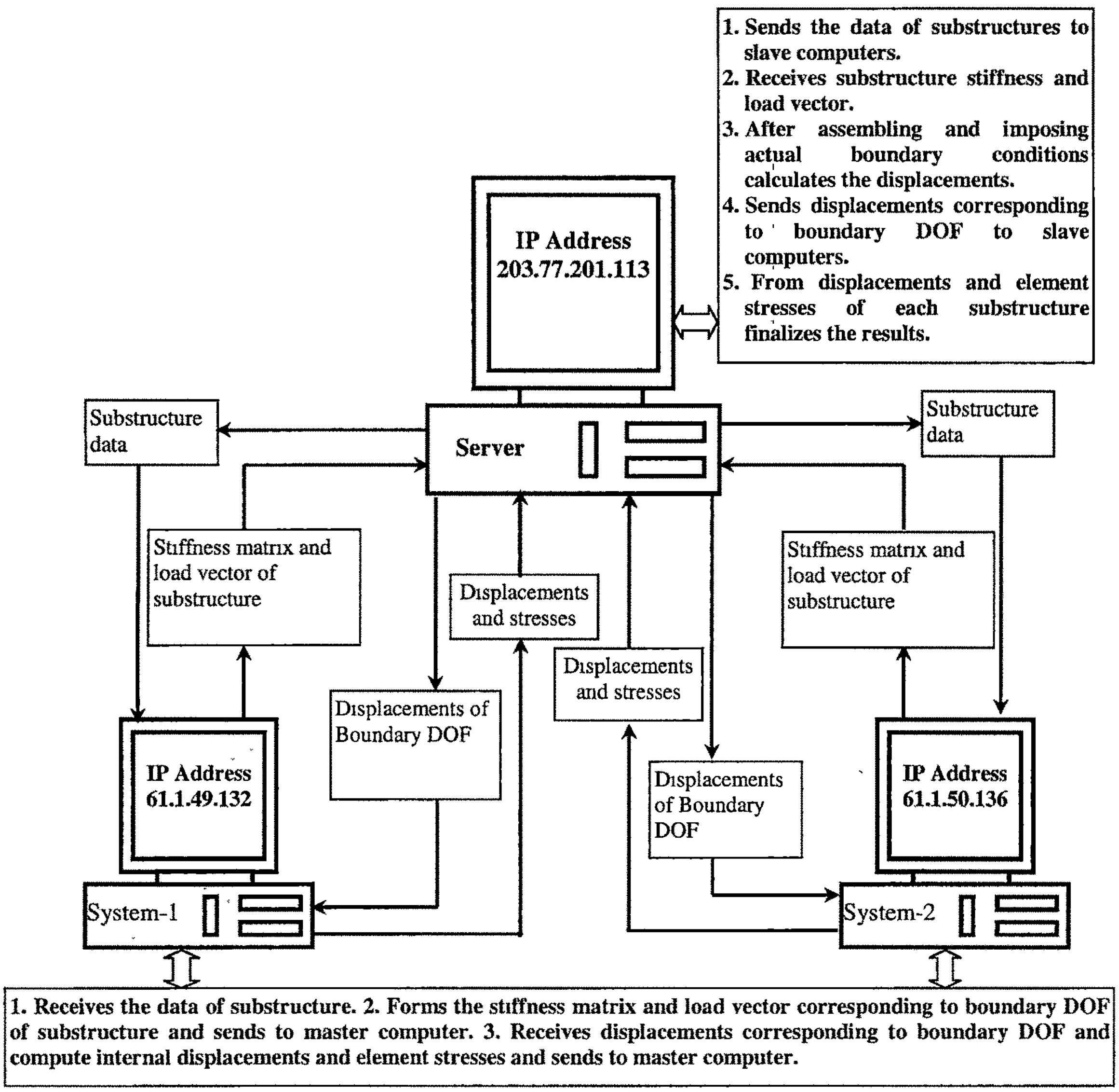


FIG. 9.7 INTERNET BASED DISTRIBUTED COMPUTING

The implementation of substructure technique on distributed computing environment is also possible with message passing libraries like PVM and MPI. The use of the libraries requires message passing functions to be incorporated in the program to send and receive the data. This may complicate the programming and also it may require some theoretical background of parallel processing. The use of WebDedip simplifies the implementation on distributed computing environment, as message passing functions are not required in the program. It is therefore possible to use WebDedip by any application developer who can visualize the distributed computing possibilities in the area of interest.

9.6 PROBLEMS FACED DURING IMPLEMENTATION

Power failure: During the process if power fails at any of the computer, then entire process is to be restarted. Due to power failure process running on a particular computer can not be completed, the processes depending on that can not be initialized.

Disconnection: The computers connected through telephone line often face the problem of disconnection from the internet. As these computers are given dynamic IP address, on reconnection the new IP address will have to be assigned. This will require modification of Dedip configuration which may take some time and also configuration of application, consisting of process information, is required to be changed. After due modifications the application can be restarted.

Communication: The data transfer between the computers is done through FTP. Sometimes problems occur in transferring files from one computer to other due to problems of network. If communication cannot occur properly the application cannot be completed and it may give some error.

9.7 THE CLOSURE

In this chapter use of computers connected through Internet for distributed structural engineering applications was explored. Two applications were implanted over internet. Through first application it has been demonstrated that a user on remote computer can send the data for analysis for which one is not having the facility and can get the analysis result. Using this concept different kind of software can be made available to the user. Without purchasing the costly software thus one can use software available on remote computer by paying some nominal charges. The advantage of this facility is not only the economy but also the availability of the latest version of the software at his disposal.

As during the use of internet most of the time CPU is idle, CPU time of internet connected computers can be effectively utilized for solution of computationally intensive problems. It has been demonstrated through the second application that the WebDedip can be used to develop distributed computing environment over which structural engineering applications can be implemented very easily. Advantage of high performance computing can be obtained by computers connected through internet without any additional cost.

As the objective of this work is to explore the possibility of computing using internet, here only two slave computers have been used, but more number of slave computers can be used for large size structure.