第22卷第2期
2005年4月

计 算 力 学 学 报
Chinese Journal of Computational Mechanics

Vol. 22, No. 2
April 2005

文章编号:1007-4708(2005)02-0202-05

Network-based numerical analysis framework

LI Hai-jiang

(Institute of Road and Bridge Engineering, Dalian Maritime University, Dalian, China)

Abstract: This paper presents a network-based numerical analysis framework (NetNA) which integrates cluster parallel FEA (Finite Element Analysis) together with Java distributed web computing. First, the structure of NetNA is described generally, then the processes for implementing the system are given, some examples are introduced to show the applications of NetNA, finally, some conclusions are given. The framework mentioned above introduces network technology and OO (Object Oriented) method into the field of FEA and civil engineering, which shows some very promising results: the Internet may become the daily design center for our engineers.

Key words: NetNA (Network-based Numerical Analysis); OOP (Object Oriented Programming); cluster; web computing

1 Introduction

NetNA introduces some new computer technologies to develop a kind of network computing platform which integrates parallel FEA together with distributed web computing, and different parts in the same seamless system perform different tasks. The existing gravity wharf CAD system has been transferred onto the Internet as one part of NetNA with the long-term aim to make the Internet a daily design center for civil engineers. Meanwhile, all of the relevant communications between different parts are realized by Java, like Java applet, servlet, Java socket programming and so on; one simple but fully functional finite element analysis module realized in Java has been added into NetNA, which makes NetNA a good long-distance CAI system.

2 Building the hardware platform

The parallel finite element analysis has become more and more widely used in China with the great popularity of PCs cluster technology[1]. With the popular use of message passing library, like PVM[2], especially the emerging of MPI[3], the parallel programming has become much easier than ever before, since the algorithm realized on cluster can run very well on most of the super parallel computers without any modification, and it is much easier and cheaper now to build a cluster. The first step to construct NetNA is to build the PCs cluster, and the processes to build the cluster include the following steps: requirement analysis, structure design, components installation and system test[4]. The following Fig. 1 shows the structure of built cluster.

3 Building the OO parallel FEA kernel

It was ten years ago when OO software design method was first introduced into the field of FEA[5]. Since more and more researchers noticed its remarkable success in most of other applied fields, they are trying to study deeply its application in FEA in order to change the passive and lagging situation compared to other research fields. There is no doubt about the advantages in OOP, especially compared with the former POP (Procedure Oriented Programming) method , so

Received by: 2003-02-23; **Revised by**: 2003-05-14.
Project supported by the National Grand Fundamental Research 973 Program of China under Grant No. G1999032805 (Large Scale Scientific Computing Research)
Corresponding authors: LI Hai-jiang (1973-), Male, Ph. D. Associate Professor.

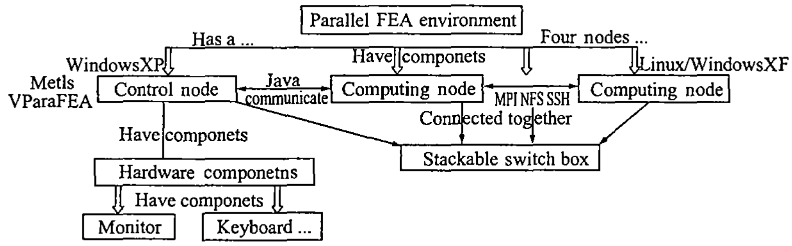

Fig. 1 The structure of parallel FEA PCs cluster

NetNA uses the OO language C++[6] to develop the finite element analysis kernel. From the experiences we get, it really demonstrates the developer has lots of merits in the program efficiency, stability, complexity control and etc.

3.1 System analysis

To analyze the structure of class library is the first step to develop a successful OO FEA program system[7]. First, analyzing the process of finite element method in detail is quite necessary, if some concept owns and manages its data, we can make this concept a class theoretically, such as node, degree of freedom; triangle element, isoparametric element, beam, truss, frame, shell and super-element...; vector, matrix, diagonal matrix, polynomial...; material, elastic material, plastic material...; load, constraint, nodal-load, self-gravity...; Gauss solver, LDLT solver, NewtonRaphson nonlinear solver; point, line, grid, domain, digital model and so on, but in fact, before getting the final copy of a FEA class library, the developer has to analyze and select again and again, to classify those concepts according to the derived relationship between them, such as node, degree of freedom...; element...; vector, matrix...; material...; external load...; linear solver, nonlinear solver...; point... and so on, the concepts in this level can be made as the base class of their original concepts, such as element can serve as the base class of triangle element, beam and so on; linear solver can serve as the base class of the Gauss solver, LDLT solver and so on. Besides the derived relationship between classes, sometimes, the developer can also use the integration to make a new class, for example, the class of degree of freedom can be included in the class of node, the material class can be included in the class of element, and all that. By doing so, the developer can further decrease the number of independent classes, so the complexity of programming can be decreased too.

After finishing the class hierarchy, the designer needs to realize the interface for each class. At this step, to define the member data and member function is the main job, and this is different from the detailed programming. This step is so important that it decides whether or not the OO method can be applied in the finite element programming successfully. OO method can help the developer to use the existing codes very easily by means of deriving. As mentioned above, there are already some finite element class libraries in the world[8,9], ParaFEA made its own class library based on analyzing and deriving those of forerunners, furthermore, it integrates the parallel finite element module, pre and post process module, together with the framework of Visual C++ MFC class library, in the end, ParaFEA becomes a highly integrated and visual OO finite element analysis kernel.

3.2 Serial /parallel finite element analysis

As to the detailed realization of each class in the above class hierarchy, since the length limitation of this paper, here only some main classes will be described according to the inheritance feature in OO finite element programming. Class FEMaterial and class Element are two key classes in ParaFEA. The Fig. 5 describes these two classes and their derived classes, the meaning of each parameter or function in their definitions is obviously clear since the long name method is used here, one can get the meaning right from the word itself. The class BaseObject is the basic one in the finite element class hierarchy, it is the MFC that gives us the way to design the class library since all classes in MFC are derived from CObject, by doing so, all classes in the library are connected together by the common base class. Class ElasticMaterial and class VonMisesMaterial are both derived from the class FEMaterial, and they work for linear elastic problems and nonlinear plastic problems respectively. Class TwoDimEle is derived from its base class element, and class Quad8IsoEle is derived from its base class TwoDimEle. From the above definitions of classes, some rules and advantages about the OOP is coming out: the first one is the data encapsulation, that means data belongs to some specific class object, none of others can get the data directly, they can get the data only by using some common member function, by doing this, the data safety has been improved greatly; the unique inheritance is the key feature of OOP, in order to make a new class, only very few codes need to be added, the new class can use the codes of its base class automatically, this can greatly decrease the lines of source codes undoubtedly; and the polymorphism is another important factor in OOP, by declaring a Virtual function, the base class and its derived class can share this function with the same name and parameter list, but they will be differentiated during the running process by the way of dynamic binding without any problem.

From the above pseudo codes, we can find out some other features of OOP. After defining the class library, the detailed OO programming becomes simple and intuitive, what the developer needs is to define some class objects and to control these objects to perform some specific jobs in the program. In the above analysis procedure, a digital model class object of specific problem is defined first, after that, the authority to control the whole program falls into the hands of this class object of digital model. It gets the original input data to analyze and then gives the results. The Fig. 2 shows class hierarchy of ParaFEA, and the Fig. 3 describes several examples.

MathAPIs	DataCenter	FEABase	DIgitalModel	Graph	App. Framework
ExtMathLib	BaseObject	BaseObject	BaseObject	BaseObject	MFC library
CMathRoutines	String	DegreeofFreedom	Domain	Point	(Mswindows)
Petsc:parallel	DataIO	Point	DynamicMode	Point3D	
math library	DataIOut	Point3D	Dyn-NLMode	BLine	Cobject
	DataIn	Node	StaticMode	SubArea	CAppView
	Directory	FEMaterial	Static-NLMode	Volume	CAppDoc
SelfDevMathLib	Explorer	ElasticMaterial	PushOverMode	...	CAppFrame
Vector	...	VonMisesMaterial	OptMode		...
...		TrescaMaterial...	...	or	or OWL library
Matrix	Pair	Element	SubDomain	based on	(Borland
DiagMatrix	PairContainer	TwoDimEle	(Parallel PCG)	OpenGL	windows
TriangleMatrix	...	TrussEle	...	graphical	class library)
...		FrameEle		library	...
BaseObject	DynArray	TriangleEle	TimeStep	...	or Application
SystemEquations	jQuadIso8List	QuadIso4Ele	LoadFunType		Class Library
EquationSolver	jTriaEleList	QuadIso8Ele	...	or	Frame on
LinearSolver	jLoadList	ThreeDimEle		based on	Linux like
LDLTMethod	jNodeList	BrickIso20Ele	MPI or PVM	AutoCAD-	KDevelop
NonlinearSolver	jMaterialList	SerialSuperEle	as Message	ADS/ARX	...
ConstStiff	...	ParaSuperEle	Passing	develop-	
N-RMethod		...	Library	ment	Metis or Para-
BFGSMethod	or based on	LoadType	for Parallel	...	Metis as
ParaSovler	some data-	Boundary	Finite		domain auto-
ParaCholesky	base like	BodyLoad	Element		decomposition
ParaPCG...	Access	GravityLoad	Analysis		library for
	...	NodalLoad...			Parallel use

Fig. 2 Finite element class hierarchy of ParaFEA

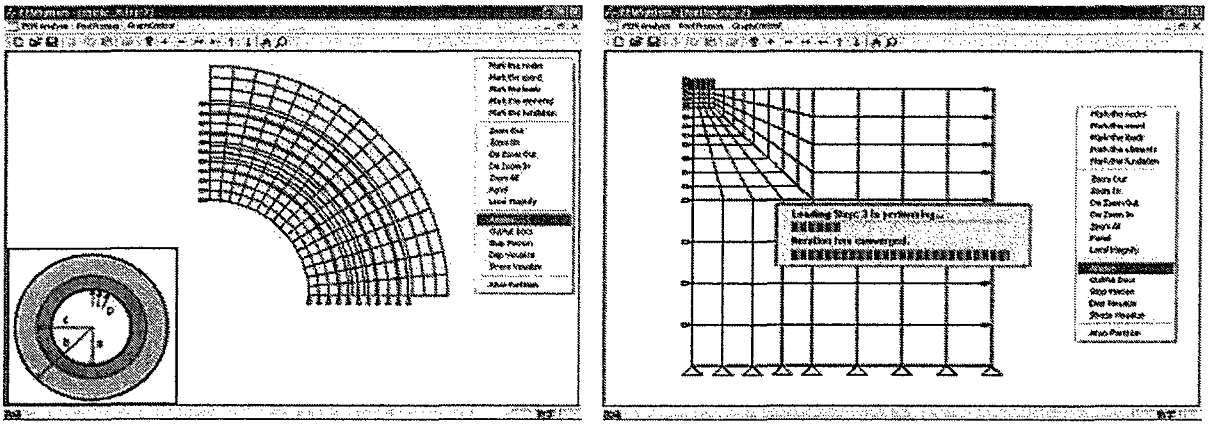

Fig. 3　Examples of OOP in finite element analysis

4　Applications of NetNA

The key point of Internet-based GWCAD is the realization of data transferring protocol. Netscape and IE are the two main internet explorers at present, Netscape introduces push-server as its data transferring protocol, it cuts data into several parts automatically when they are transferring, and the old data will be refreshed by the new data in time. Although this protocol seems right for us, the main problem is that Netscape is never widely used as before. IE instead introduces pop-server protocol which keeps on refreshing the data no matter whether or not they are changed, so lots of unnecessary data transferring happens. According to the requirement of Internet-based GWCAD, a new data transferring protocol has been made. The kernel of that protocol is to transfer data string code, but not the data itself and the data file repeatedly, the forming of that data string code is based on the studying of data character of GWCAD deeply, we find most of the necessary data can be represented by some data selection items, each of these items has five bits in all, the first three bits represent the kinds of data, the last two bits represent the selection of one specific data. Many items form a data string code and it is this string code that is transferred on the Internet, so the speed is so fast that the user can not feel any latency at all. The Fig. 4 shows the data transferring process of Internet-based GWCAD.

GWCAD can be added into NetNA just because of its moderate size in its model and database. All client interfaces are realized by the way of Java applet and html, but since the efficiency of Java program is not high enough, big systems can not run well on the Internet right now. Wharf structural engineers can use this interface to input parameters to form a digital model; meanwhile through the computational capability of Java, they can visualize and modify the model many times until it is acceptable, then submit their design. Web server responds to this requirement and transfers the task to GWCAD server which is responsible for detailed analysis.

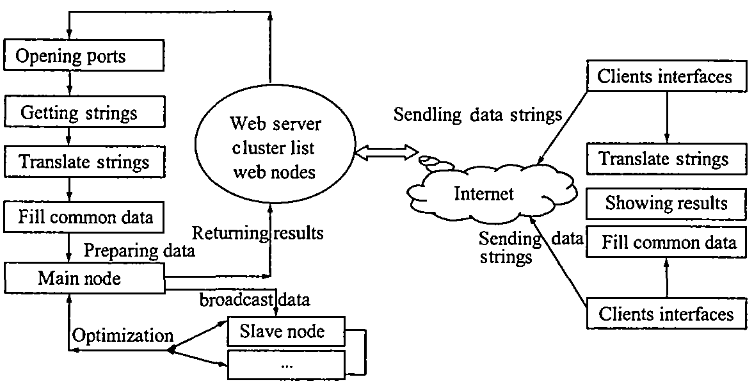

Fig. 4　Data transferring process of Internet-based GWCAD

5 Conclusions

The new era filled with new computer technologies is coming, like cluster, network/Intranet/Internet technology, OO programming software design method and so on, but the application of these technologies in FEA and civil engineering in China is far behind compared to those in other research fields. NetNA combines the cluster OO parallel finite element analysis kernel together with web computing interface, builds a good PCs cluster as the developing hardware platform, and realizes Internet-based gravity wharf analysis and design CAD system. In order to make NetNA practical in business on the Internet, a new concept "charging only for service" has been put forward, and it has been tested successfully in our private LAN. This concept fits right for the new trends to introduce Intranet/Internet technology into the present and future engineering software and the experience to develop NetNA will serve as a good base for the future application study of grid computing.

References:

[1] Mark Baker. *Cluster Computing White Paper*[M]. The Press of University of Portsmouth, UK,2000.

[2] Al Geist, Adam Beguelin, et al. *PVM: Parallel Virtual Machine A Users' Guide and Tutorial for netwoked Parallel Computing* [M]. MIT Press, 1994.

[3] Gropp, Lusk, Skjellum. *Using MPI: Portable Parallel Programming with the Message-Passing Interface*[M]. MIT. Press,1999.

[4] Thomas L. Sterling, et al. *How to Build a Beowulf* [M]. The MIT Press, London, 1999.

[5] Forde BWR, Foschi R O, Stiemer S F. Object oriented finite element analysis[J]. *Computers & Structures*,1990,**34**(3):355-374.

[6] Bruce Eckel. *Thinking in C++*[M]. Prentice Hall Inc. ,1998.

[7] Zimmerman T, Dubois P Y, Bomme P. Object-Oriented finite element programming. I. Governing Principles [J]. *Computer Methods in Applied Mechanics and Engineering*,1992,**98**(2):291-303.

[8] Mackie R I. Object-Oriented programming of the finite element method[J]. *International Journal for Numerical Methods in Engineering*, 1992, **35**(2): 425-436.

[9] Kong X A, Chen D P. An object-oriented design of FEM program[J]. Computers & Structures,1995, **51**:157-166.

基于网络的数值分析系统

李海江

(大连海事大学 道路与桥梁工程研究所,辽宁 大连 116026)

摘　要:结合并行有限元分析以及Java分布式Web计算,介绍了基于网络的数值分析系统。首先介绍了系统的结构组成,进而给出系统的具体构建过程,并结合一些算例表明系统在土木工程中的实际应用,最后给出了结论。本系统将计算机网络技术以及面向对象的方法引入有限元分析以及土木工程领域,进行了有益的探索,研究可望将互联网变成工程师的日常工作中心。

关键词:网络数值分析; 面向对象;集群,Web计算

中图分类号:U651;TP311　　**文献标识码**:A

收稿日期:2003-02-23;**修改稿收到日期**:2003-05-14.

基金项目:国家973项目大规模数值计算(G1999032805)资助项目.

作者简介:李海江(1973-),男,博士,副教授.