*Structural Engineering and Mechanics, Vol. 37, No. 1 (2011) 17-37*

# An Internet-based computing framework for the simulation of multi-scale response of structural systems

Hung-Ming Chen* and Yu-Chih Lin[a]

*Department of Construction Engineering, National Taiwan University of Science and Technology,*
*43, Sec. 4, Keelung Rd., Taipei 106, Taiwan*

*(Received September 22, 2009, Accepted September 7, 2010)*

**Abstract.** This paper presents a new Internet-based computational framework for the realistic simulation of multi-scale response of structural systems. Two levels of parallel processing are involved in this framework: multiple local distributed computing environments connected by the Internet to form a cluster-to-cluster distributed computing environment. To utilize such a computing environment for a realistic simulation, the simulation task of a structural system has been separated into a simulation of a simplified global model in association with several detailed component models using various scales. These related multi-scale simulation tasks are distributed amongst clusters and connected to form a multi-level hierarchy. The Internet is used to coordinate geographically distributed simulation tasks. This paper also presents the development of a software framework that can support the multi-level hierarchical simulation approach, in a cluster-to-cluster distributed computing environment. The architectural design of the program also allows the integration of several multi-scale models to be clients and servers under a single platform. Such integration can combine geographically distributed computing resources to produce realistic simulations of structural systems.

**Keywords:** Internet computing; distributed processing; simulation; structural systems; hierarchical modeling.

## 1. Introduction

Advances in computer technology and numerical methods have allowed the simulation of engineering problems that traditionally have been addressed only through experimentation and theoretical models. Some industries have been able to design sophisticated engineered systems based solely on computer simulation. In addition, many complex phenomena, such as airplane crashes and car accidents, can now be analyzed through computer simulations. In structural engineering, using a computer simulation to realistically represent the detailed behavior of structural systems in various situations, such as the global response and the detailed damage to a structure during a major earthquake, is a goal which remains to be achieved by engineers.

In comparison with other engineering systems, structural engineering systems, including bridges and buildings, are usually large-scale and contain the effects of various structural components and

---

*Corresponding author, Associate Professor, E-mail: hungming@mail.ntust.edu.tw

[a] Ph.D. Candidate

materials at many scales. Therefore, to successfully create a realistic simulation of a structural system, the global model must be able to capture the true behavior of the global system, including the detailed local mechanisms. Modeling the whole structural system in every detail, using very fine meshes, is one effective approach to simulate the responses of structural components. However, the resulting models become enormous and are difficult to process using current computational power.

Presently, Internet computing, which is also called Grid computing, is being perceived as a promising avenue to achieve further computational power. Virtually an infinite number of machines can be connected via the Internet, theoretically leading to unlimited computational capabilities. However, Internet is comparatively slow and over-utilized communication channel for being used in parallel processing. Unless each subtask is highly independent, Internet communication can render impossible any possible gain in efficiency from an Internet-based distributed application. Depending on the time of the day, the traffic on the Internet can be such that no timely communication is possible. Since decomposing a complex model and distributing the decomposed tasks to all available machines over the Internet is not expected to work well due to the Internet-imposed communication time obstacle, parallel simulation method that can especially minimize the quantity and the frequency of data communication via Internet is the key to successfully fulfill realistic simulation in an Internet-based computing environment.

## 2. The simulation method for Internet computing

A new Internet-based simulation method for the realistic simulation of structural engineering systems is proposed and investigated in this paper as one of the possible solutions to enable Internet computing platform for structural simulation. Two levels of parallel processing will be involved in this framework: (1) multiple locally distributed computing environments connected by the Internet to form (2) an Internet-based cluster-to-cluster distributed computing environment. To accomplish a realistic simulation in this computing environment, a large-scale structural simulation task has been separated into two distinct categories of simulations: (1) a simplified global model, and (2) several detailed component models at various scales.

For illustration, the modeling of a structural component by coupling of a simplified component model (SCM) and a rigorous component model (RCM) is shown in Fig. 1. These two separated numerical columns, which are modeled on two scales, actually represent the same structural column in the real structural system. The integration of these scaled models according to their respective frames of reference is shown conceptually in Fig. 2. As the figure indicates, a simplified component model resides in a macro/global system to obtain the actions applied by the rest of the system. A rigorous component model (RCM) is analyzed in isolation to obtain its detailed responses and behavior. The details of the synchronization of these two models will be discussed in the system requirements section of this paper.

Since the synchronic strategy of the multi-scale models of various scales has been developed, we now introduce a method to systematically perform these simulation tasks in the proposed Internet-based computing environment. These correlated multi-scale simulation tasks are distributed amongst clusters connected together by the Internet to form a multi-level modeling hierarchy. These simulations, in separated clusters, coordinate with each other through the Internet to complete a realistic simulation of a whole structural system. This paper also presents a software framework for supporting the proposed realistic simulation approach in an Internet-based, cluster-to-cluster

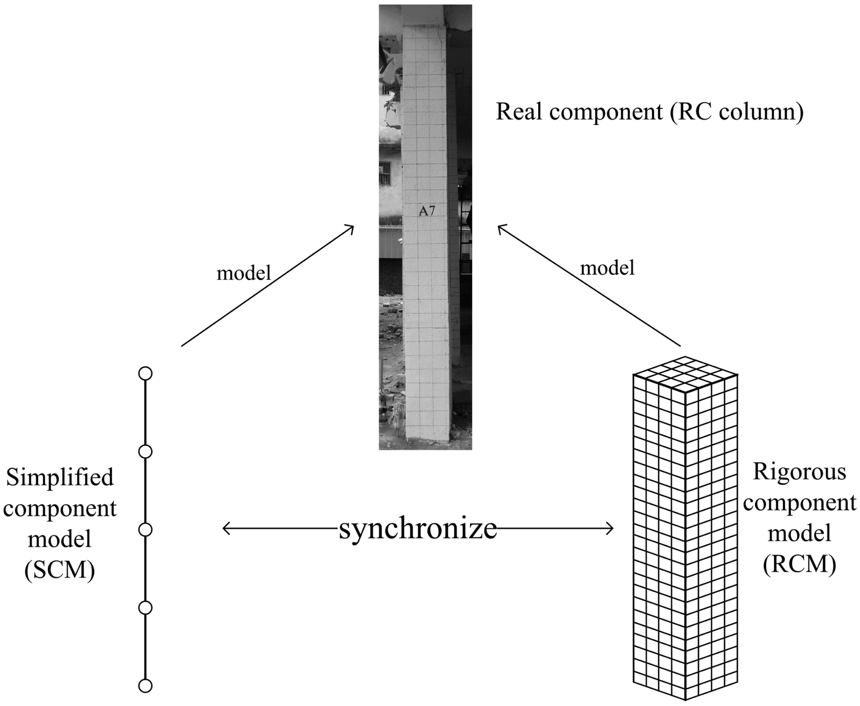


Fig. 1 The modeling of a real component by coupling of an SCM and an RCM

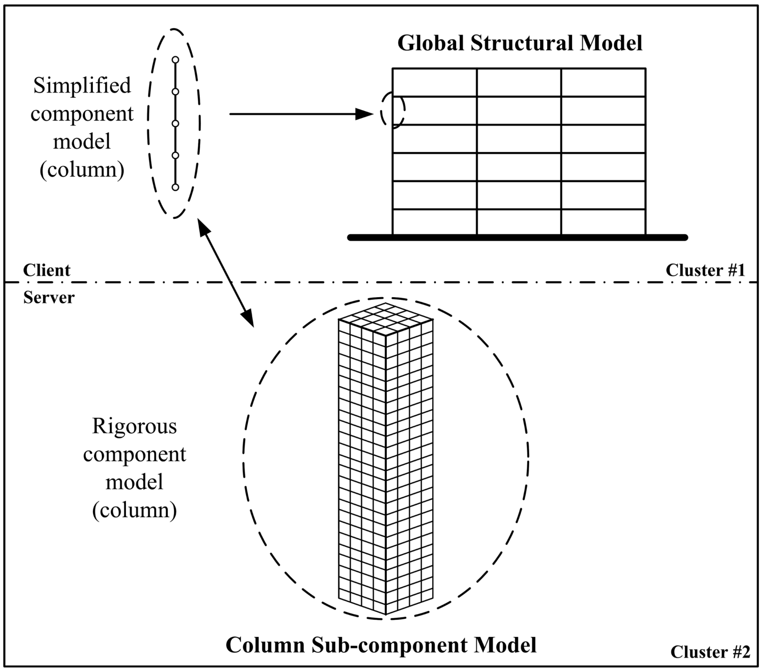


Fig. 2 The simulation method for Internet computing

distributed computing environment. The architectural design of the program allows the integration of several multi-scale models as clients and servers under a single platform. Such integration will facilitate a more realistic simulation of a structural system.

## 3. Related research

The researches related to this study include the applications of Internet computing in engineering, and the computational methods by decomposing task into an assembly of subtasks. Recently, Internet or Grid technology applications for engineering problems have begun to emerge. These applications fall into one of three categories: (1) system integration, (2) distributed experimental simulation and (3) structural design and analysis. In this section, several related studies will be briefly introduced.

In system integration, Zheng *et al.* (2004) developed the MASSIVE framework, which aims to use Grid technology to establish an enabling environment for distributed simulation and visualization of large-scale scientific and engineering research. This project focuses on collaborative numerical simulation and visualization in a Grid environment. It also develops Grid-enabled capability and products for use by the wider community such as industry and academia. Dolenc *et al.* (2007, 2008) developed the InteliGrid framework which provides standards-based collection of ontology based services and Grid middleware in support of dynamic virtual organizations as well as Grid enabled engineering applications in AEC (Architectural Design, Engineering Design and Construction) industries. This study shows Grid technology can provide such an infrastructure for collaboration and interoperability in the AEC industries. In addition, several issues, such as business concepts, which need to be attended to for virtual organizations are addressed and generic business-object-aware extensions to Grid middleware are developed in the framework. These implementations successfully used Internet to integrate geographically distributed systems and resources for providing services cooperatively on information sharing and system usage, but not focused on engineering computing.

In distributed experimental simulation, some studies focus on linking geographically distributed computing resources and experimental equipment via the Internet, to cooperatively perform earthquake simulations for structural systems. Examples of such implementations include Spenser *et al*. (2004) for the NEESgrid project, Kwon *et al*. (2005, 2008), Pan *et al*. (2005), Takahashi and Fenves (2006) and Yang *et al*. (2007). These studies all successfully integrate geographically distributed experimental recourses and environments using Internet to collaborate on structural simulations.

In structural design and analysis, some researchers applied Internet technology to provide analytical services to be accessible remotely over the Internet. Examples of such implementations include Peng and Law (2002, 2004), Nuggehally *et al*. (2003), Yang *et al*. (2004), Vacharasintopchai *et al*. (2007) and Chen and Lin (2008). These studies successfully enable users to access centralized computing recourses remotely via the Internet.

On the other hand, some researchers applied Internet technology to integrate distributed computing resources to cooperate on computational analyses. Song *et al*. (2003) solved a 2D airfoil shape optimization problem using computational fluid dynamics (CFD) within a Grid computing environment implemented in Matlab. The adoption of Grid technologies not only simplifies the integration of proprietary software, but also makes it possible to harness distributed computational power in a consistent and flexible manner. Chen (2005) presented an email-based communication technology of global coordinative optimization of distributed structural systems. Li *et al*. (2006) applied Grid-based computing technologies in the field of engineering design optimization using the Finite Element Method (FEM). Three essential elements in FEM-based structure optimization problems (CAD modeling, mesh and solution, and optimization) are integrated and automated in a Grid-enabled computation environment. This FEM-based optimization was utilized by remote users and was applied to help FEM-based excavator working equipment analysis. Lam *et al*. (2002, 2004) presented an Internet-based environment for distributed collaborative performance-based building design and evaluation. Alonso *et al*. (2007) developed a Grid computing application for the 3D dynamic analysis of large dimension buildings. Performing distributed executions has enabled a considerable reduction in the global execution time of structural studies. Later, Alonso *et al*. (2008) developed a high performance computing-based application for 3D structural analysis of buildings. A Grid structural analysis service, which integrates the parallel application, has been implemented, taking advantage of computers geographically distributed on the Internet. This approach incorporates remote computing recourses to concurrently process a large number of independent structural analysis tasks. These studies successfully integrate geographically distributed computers to cooperate on solving large amount of independent tasks in parallel.

In summary, all the above works prove that the Internet or Grid technology has potential applications for many engineering problems. The approaches proposed in these studies use Grid technology to gather together and coordinate the enormous computational resources available over the Internet to provide the computing energy necessary to undertake sophisticated engineering problems, among many other applications. The logistical delegation of responsibilities can also improve the accuracy and efficiency of the original solutions. The work of this study is in the category of structural design & analysis, where all the Internet-related applications were focused on providing computational services remotely or processing many individual tasks concurrently using Grid technology. However, using a Grid computing platform to process a single large-scale task in parallel has not yet been achieved due to slow Internet communication. Therefore, the goal of this study is to propose a modeling procedure and a software framework for analyzing a large-scale structural system with detailed results based on the Internet computing platform and technology.

On the other hand, the computational strategy is decomposing a problem into correlated sub-problems in a hierarchy. There are solution algorithms using a similar strategy, which is generally called domain decomposition, in parallel processing. Examples of domain decomposition methods include substructuring (static condensation, dynamic reduction), the parallel central difference algorithm by Hajjar and Abel (1989), Finite Element Tearing and Interconnecting (FETI) by Farhat and Roux (1991), the parallel conjugated gradient methods as presented in Adeli and Kumar (1995) and Kumar and Adeli (1995), the Iterative Group Implicit (IGI) algorithm by Modak and Sotelino (2000), the parallel multigrid approaches as presented in Adams *et al*. (2004), and the nonlinear substructuring algorithm by Chen and Archer (2005). The common solution process of these domain decomposition methods is that all sub-problems are modeled numerically only once in a single scale, and then they are correlated and solved by a specific parallel algorithm using the numerical objects, such as matrix and vector, converted from their numerical models. These domain decomposition methods also have the potential to be adopted and extended for applications in Grid computing environment. The computational method in this study is an attempt to provide an alternative approach which is fundamentally different in two aspects in comparison with the domain decomposition methods. First, all sub-problems are modeled twice as a pair of models (SCM and

RCM) in very different scales and complexities. Second, all sub-problems are correlated by assembling their SCMs in a global model, instead of numerical objects converted from their RCMs. Since the data transferred for synchronizing a pair of SCM and RCM usually are only a few nodal displacements and the analyses of different RCMs are highly independent, the quantity and the frequency of data communication via Internet can therefore be greatly minimized.

## 4. The proposed Internet-based simulation framework

The present enterprise, using Internet-based technology to produce a realistic simulation of a large-scale structural system, has one critical weakness, which is the low efficiency and reliability of Internet communication. To conquer this problem, the proposed cluster-to-cluster computing environment and the multi-level hierarchical modeling and simulation method are combined to form an Internet-based framework for large-scale structural simulation in this study.

### 4.1 The cluster-to-cluster distributed computing environment

A simple one-level Internet computing environment can be described as a massive collection of heterogeneous machines connected by comparatively slow, over-utilized communication channels. The computer applications running in this kind of environment must assume that any communication with other machines is done over the Internet, no matter whether the machines are local or remote. Thus, this kind of framework is only suitable for distributed computing problems for which each subtask is more or less independent, such that the time required for Internet

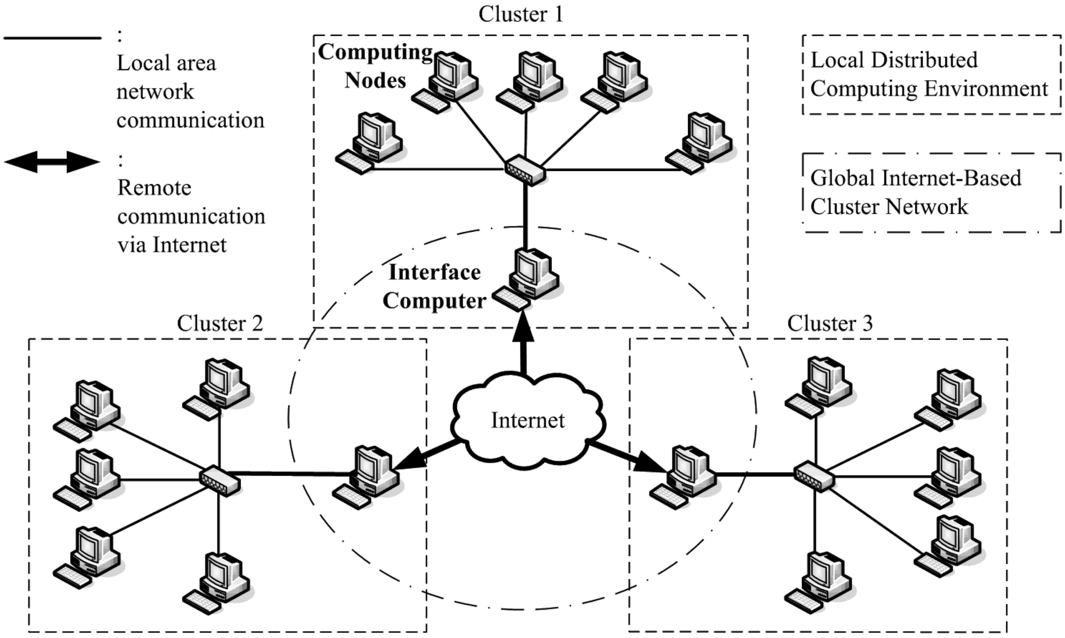


Fig. 3 Cluster-to-cluster distributed computing environment

communication among massive machines is thereby rendered insignificant. Here, please note that for existing distributed finite element analysis algorithms, the communication time is significant and there is a limitation to the number of processors which can yield an efficient computational process. Therefore, there is no way to extend existing algorithms to be applicable in a simple one-level Internet computing environment.

The Internet communication bottleneck can be significantly bypassed by organizing the hardware configuration of an Internet-based environment into a cluster-to-cluster distributed computing framework, as shown conceptually in Fig. 3. This framework involves two levels of parallel processing: (1) multiple locally distributed computing environments, which are connected by the Internet to form a (2) Internet-based cluster-to-cluster distributed computing environment. In fact, each of the clusters in the framework could be comprised of a number of different kinds of processing equipment, including, but not limited to, a distributed memory supercomputer, a shared memory supercomputer, or just a personal computer. The bulk of communication takes place between the computing nodes within a cluster, similar to a traditional cluster computing environment. Only those messages to be exchanged between clusters are required to communicate between the interface computers on the Internet.

### *4.2 The multi-level hierarchical modeling and simulation*

In simulating structural engineering systems, finite element analyses are often used to represent the behavior of the system as a whole. In this type of analysis, simplistic nonlinear models, which consist of one or a few nonlinear elements, can be used for representing the global behaviors of various structural components, such as beamcolumns, walls, and connections. Although the global responses of structural systems can be simulated in this way, the detailed responses of their components cannot be obtained using simplistic models. On the other hand, much research has been conducted on the analysis of individual structural components. In this type of analysis, the structural components are modeled in a very fine mesh to produce detailed responses. However, these models are isolated without consideration of the relationship between their behavior and that of the rest of the system. Therefore, these two levels of knowledge should be integrated under a single platform to produce a more realistic simulation of structural engineering systems.

Modeling the whole structural system in every detail using very fine meshes as the way to simulate the responses of structural components is one approach to achieve this. However, the resulting models would be enormous, awkward and inefficient using current computational power and analysis techniques. Another possible simulation approach for an Internet computing environment is proposed in this study. The proposed approach uses independent models of various scales for simulating the global system of a structure and its detailed components. In the beginning, a structural system is analyzed in the ways traditional structural analyses are conducted. Once done, the simulations of the structure's multi-scale component models are integrated and correlated with the analysis result of the structural system to achieve a realistic simulation of the whole structure. In this way, a detailed simulation of the whole system can be decomposed into several detailed simulations of its individual components.

Fig. 4 shows that the multi-level hierarchical modeling and simulation method can fit into the proposed cluster-to-cluster computing environment nicely. Here, a joint component and a beam component are both separately modeled at two scales. For example, the beam component of the SCM resides in the global structural system, analyzed by Cluster #1, and then it applies the

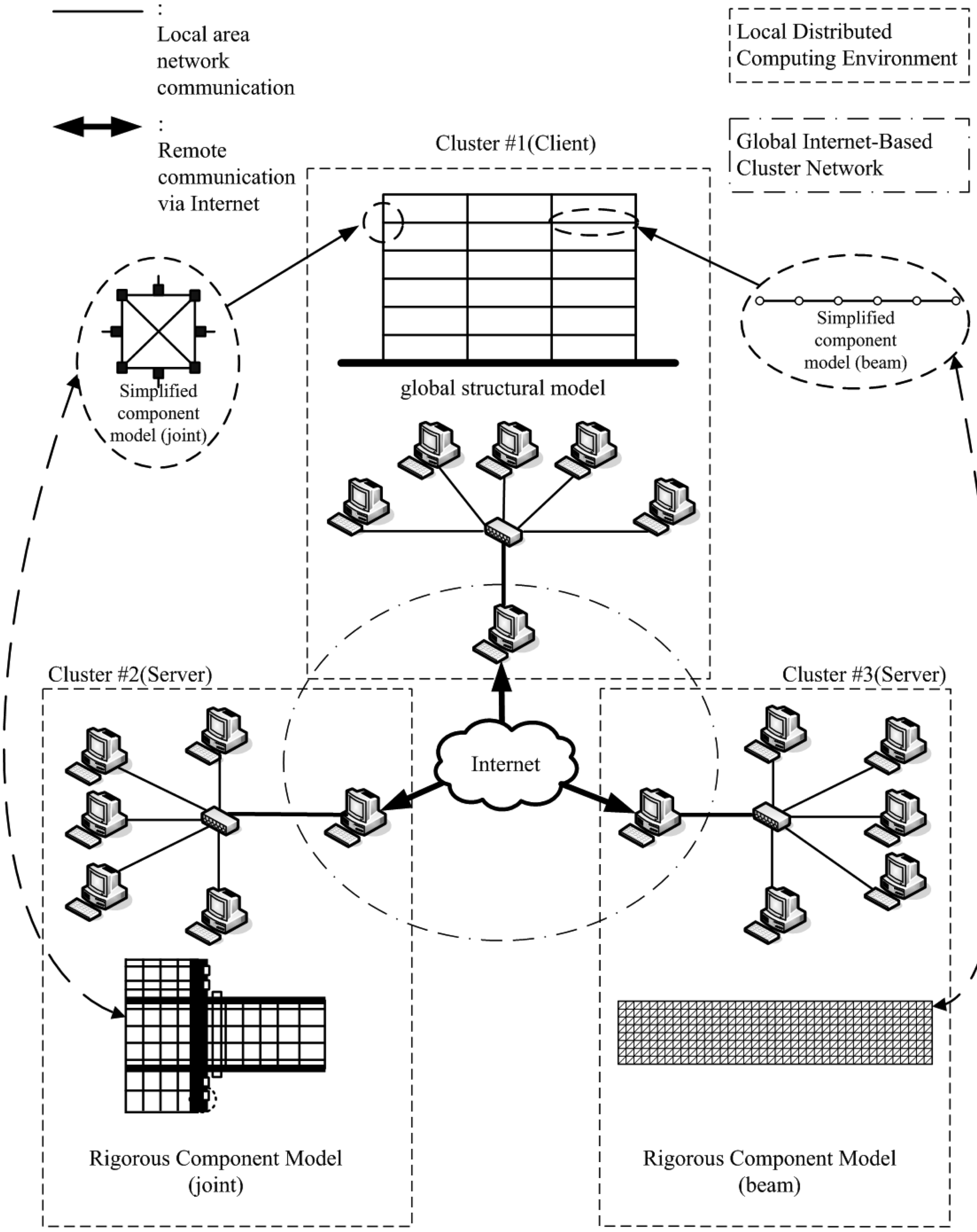


Fig. 4 Hierarchical modeling and simulation in the cluster-to-cluster distributed computing environment

reactions obtained from the rest of the system. Through the global Internet-based network, Cluster #3 receives the effects of the global system on the beam component of the SCM, and these effects are then applied to the beam of the RCM, analyzed by Cluster #3, to obtain the detailed responses and behaviors.

Regarding the allocation of computational tasks amongst clusters, the global analysis of a structural system with SCMs is preferably to be assigned to a local cluster, whereas the local component simulations using RCMs are assigned to available remote clusters prior to the simulation by the user. Since load-balancing is not in the scope of this study, the assignment of tasks remains static for the entire duration of the simulation.

Chen and Iranata (2008) performed a study for verifying the accuracy and demonstrating the applications of the proposed multi-level hierarchical modeling and simulation method. In that study, accuracy of the proposed method is evaluated using a simple portal frame example. The frame is first modeled as a two-level hierarchical model and analyzed using the proposed method. The two-level hierarchical model is composed of a global model in line elements for the whole structure, and a detailed component model in plane elements for its beam portion only. Then the frame is analyzed using a model which is totally constructed using plane elements only with the same mesh size as the component model of the previous two-level model. The resulting comparison of the two analyses indicates that the peak and average percentage of error of strain in the beam section are less than 1%. With this level of accuracy, the proposed simulation method should be appropriate to be used for simulating the response of real structural systems. Afterward, that study presents several case studies for simulating large-scale structural experiments by applying the proposed multi-level simulation method. The comparison of the simulation results with the real experimental results shows high levels of agreement and consistency.

## 5. System requirements

In developing a suitable software system, certain considerations are vital for building an Internet-based computing environment that can process the multi-level hierarchical modeling and simulation method in an Internet computing environment. These requirements include (1) the cluster-to-cluster communication protocol, (2) the synchronization of the multi-level hierarchical models, (3) the accommodation of nonlinear effects, (4) the integration of commercial analysis software, (5) the user interface, and (6) the fault tolerance. A discussion of these requirements follows with some solutions provided.

### *5.1 The cluster-to-cluster communication protocol*

A simple one-level Internet computing environment can be described as a massive collection of heterogeneous machines connected by comparatively slow, over-utilized communication channels. The computer applications running in this kind of environment must assume that any communication with other machines is done over the Internet, no matter whether the machines are local or remote. Thus, this kind of framework is only suitable for distributed computing problems for which each subtask is more or less independent, such that the time required for Internet communication among massive machines is thereby rendered insignificant. Here, please note that for existing distributed finite element analysis algorithms, the communication time is significant and there is a limitation to the number of processors which can yield an efficient computational process. Therefore, there is no way to extend existing algorithms to be applicable in a simple one-level Internet computing environment.

The Internet communication bottleneck can be significantly bypassed by organizing the hardware configuration of an Internet-based environment into a cluster-to-cluster distributed computing framework, as shown conceptually in Fig. 3. This framework involves two levels of parallel processing: (1) multiple locally distributed computing environments, which are connected by the Internet to form a (2) Internet-based cluster-to-cluster distributed computing environment. In fact, each of the clusters in the framework could be comprised of a number of different kinds of processing equipment, including, but not limited to, a distributed memory supercomputer, a shared memory supercomputer, or just a personal computer. The bulk of communication takes place between the computing nodes within a cluster, similar to a traditional cluster computing environment. Only those messages to be exchanged between clusters are required to communicate between the interface computers on the Internet.

### *5.2 The synchronization of the multi-level hierarchical models*

In order to synchronize the multi-level hierarchical analysis of structural systems with the modeling approach, there are two necessary requirements. First, the analysis algorithm must be able to handle the transition between distinct scales by translating a component's global behavior, obtained from its macro-level model, to actions on its micro-level model. Likewise, a component's localized effects obtained from its micro-level model must be coordinated with the behavior of the macro-level model in the global system. Thus, mechanisms are required for extracting the values of displacements and stresses in an SCM, and then appropriately translating these quantities to the known values in the analysis of the corresponding RCM, and vice versa. Second, the SCM must accurately describe the global behavior of the component. Otherwise, the resulting global actions become invalid. Many researchers have developed simplified models that are generally effective for various structural components, such as the beam-column element by D'Ambrisi and Fillippou (1999).

Fig. 5 illustrates the translation of the displacements between the SCM and the RCM by a beam

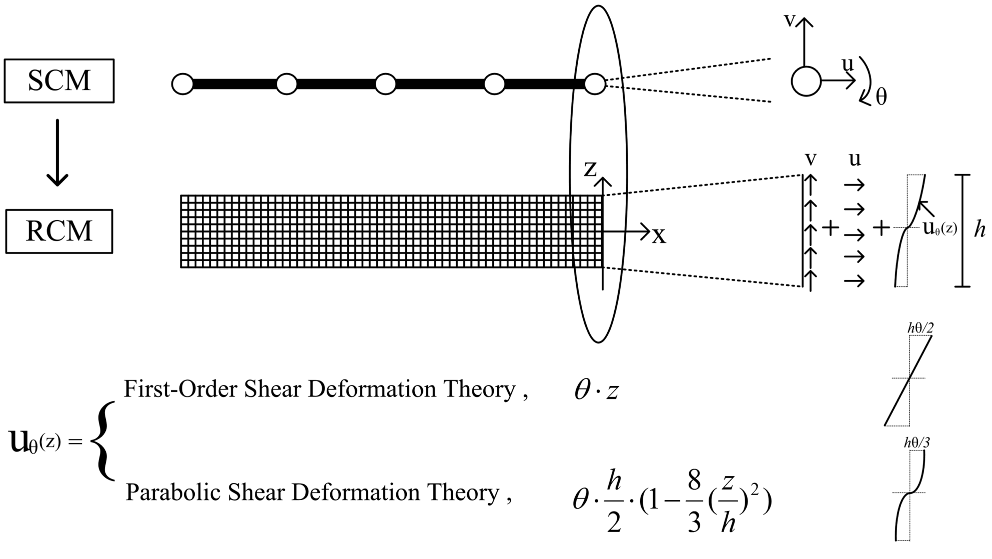


Fig. 5 Theory for displacements translation

example. Here, the displacement of a node in the beam-column element (SCM) is translated to become the displacement field occurring at the corresponding cross-section of an RCM in CST (Constant-Strain Triangle) elements. Basically, the algorithm for such a translation must be derived from appropriate theory in structural mechanics. Using beam or column components as an example, appropriate beam theories such as First-order Shear Deformation Theory (FSDT) and Parabolic Shear Deformation Theory (PSDT) can be adopted for deriving such translations at different levels of accuracy. Fig. 5 shows the equations and illustrates the mechanism for extracting and translating a single nodal displacement of the SCM into several nodal displacements on the corresponding plane of the RCM in a two-dimensional situation. Each single SCM node has three displacement components, namely the vertical displacement, horizontal displacement, and rotation. Those displacements should be extracted and translated into the corresponding plane of the RCM. As it is modeled as the continuum element, the nodal displacement of the RCM consists only of vertical and horizontal displacements. The vertical displacement of the SCM can be directly translated into several nodal displacements in the corresponding RCM plane as it is not a function of the z-axis. However, for horizontal displacement, which is a function of the z-axis, the single nodal displacement of the SCM should be extracted using the FSDT or PSDT equation into several nodal displacements and translated into the corresponding RCM plane. This 2D mechanism is the basis for developing the extracting and translating mechanism in a three-dimensional situation. In addition, the PSDT equation in Fig. 5 is for rectangular cross sections only. The translation equation for other cross sections, such as I beams, can also be derived and applied based on the same theory.

Following the above method, displacement compatibility is used to synchronize SCM with RCM. Thus, the equilibrium between RCM and SCM may not be exactly satisfied. Several verifications were subsequently conducted to assess the SCM-RCM consistency, which found a minimal (insignificant) imbalance of force. Accuracy verification work performed by Chen and Iranata (2008) further demonstrated the effectiveness of employing the proposed simulation method to simulate the response of real structural systems. Therefore, the simulation design and system implementation of this study synchronize SCM with RCM based on displacement compatibility.

Equilibrium transformation represents another potential approach to synchronizing SCM and RCM. Instead of extracting SCM displacements, which resides in the global structure, SCM nodal forces are retrieved from the global structure and applied to RCM. Basically, RCM is a digital model that lacks constraints. Therefore, applying force to RCM during analysis yields a free body motion. To handle this situation, several nodes in the central region of RCM were fixed and defined as boundary conditions. The RCM simulation result can be seen as reactions with respect to the fixed nodes of the pre-defined central region.

### 5.3 The accommodation of nonlinear effects

The simulation of nonlinear behavior of a structural component is handled on both its SCM and RCM in correlation. The SCM of a component usually is composed of nonlinear line elements with plastic hinges. The RCM of a component usually is composed of plane or brick elements with nonlinear constitutive laws of its materials in fine meshes. The global nonlinear behavior of a structural component is determined by the nonlinear elements of its SCM. Usually, each nonlinear element of an SCM describes a deformation mechanism that affects the overall nonlinear behavior of the structural component. The behavior of a nonlinear element is described by one or a few performance curves determined by the geometry and material properties of the cross sections corresponding to its plastic hinges based on codifying experimental results or theorems. On the other hand, the local nonlinear behaviors of a structural component are determined by the detailed meshes of its RCM. Each overstressed mesh can have material nonlinearity occur at each of its integration points, where an appropriate constitutive law defining the nonlinear stress-strain relationship of its material, such as steel or concrete, is specified. The detailed nonlinear responses, such as concentrated stresses and failure locations, can then be presented. The nonlinearities of a pair of correlated SCM and RCM at two different levels are synchronized by translating displacement data in between as pre-defined boundary conditions using the methods introduced in the previous section.

During the step-to-step iteration process of a nonlinear analysis, the internal state of a component is kept by its SCM in the global model at each time step. This is the same as the typical nonlinear analysis of frame structures. On the other hand, the state of an SCM at that step will be translated to be the boundary conditions applied on the corresponding RCM which inherit the internal states at integration points by mapping from the result RCM at previous step for a detailed response of the component at that step. In this process, each RCM analysis is produced and started automatically by the server-side system upon receiving a new state of the corresponding SCM. The model of each RCM analysis is created in a file by merging a predefined partial file, which contained the RCM without subjecting it to any boundary conditions, with the internal state mapping from previous step and the boundary conditions translated from the current state of the corresponding SCM received based on the predefined mapping of degrees of freedom between SCM and RCM.

Regarding the specification of locations where nonlinearities can occur in a structural component, it is handled differently in its SCM and RCM. For the SCM of a structural component, the location of plastic hinges usually is set at each of the critical points, such as the mid-span, the end and the location of applied force, which are predictable according to its position in the whole structural system in advance. This is the typical way to set the plastic hinges for building a nonlinear analytical model of a structure. On the other hand, the nonlinearities of RCM are set and can occur at any integration points, which are usually widely spread within the meshes of the model at a very fine level. Therefore, all the possible local nonlinearities can be captured and presented without knowing their specific locations in advance.

### 5.4 The integration of commercial analysis software

There are many popular commercial structural analysis software products available, such as ABAQUS and LS_DYNA. Because of their outstanding performance and the many different features that they offer, many engineers and researchers from various disciplines regularly use them as analyzing tools. In the past decade, with increasing demands for computing power, these software venders have used parallelization technology to improve the computational capacity and efficiency of their products. Therefore, considering the benefits of software analysis performance, and its general familiarity to the public, we have chosen to use ABAQUS for each cluster of the proposed Internet-based computing environment. In other words, if a cluster system has the ABAQUS software program installed, it can easily become one of the analysis servers of the proposed platform, simply by installing the server-side program of the core analysis module, which will be introduced in a later section. Therefore, those who use the proposed platform can create new system or component models through the pre-processing interface of ABAQUS. After the model has been analyzed, the post-process interface can then show the visualized results.

There is no need to mention that different commercial analysis software programs have their own unique and specific input file formats. Therefore, some rules should be established for identifying the objects with the proposed software system, in order to enable the platform to recognize the data format of different analysis software programs in a dynamic and automatic fashion, without needing to modify the code of the system.

There are two interface requirements that need to be established before the commercial analysis software can be integrated into the proposed analysis platform. First, the system should be able to recognize the content of the input file of the commercial analysis software and convert it to the corresponding object types of the proposed software system. These object types include nodes, elements, forces, boundary conditions, etc. This step is necessary because the global model and the rigorous component models of the whole simulation problem will be displayed on the graphical user interface of this system. The second requirement is that the system should be able to read the results from each of the models studied by the commercial analysis software. Therefore, after the global model is completed, the nodal displacements in the global model can then be extracted from the SCM. These extracted values of the displacements are then translated to become the initial boundary conditions of the corresponding RCM.

### 5.5 The graphical user interface

In addition to the proposed simulation algorithm, a graphical user interface which provides users a visualized way to examine and manage the designed simulation problem is a basic requirement for modern, computer-aided, user-friendly systems. For most cases, the simulation problem will contain one global model and several component models, and each of them will have their own specific properties and physical exterior. Therefore, the graphical user interface on the program's working panel provides users with a multi-window environment to view these models, simultaneously when necessary. Ideally, users should be able to "walk through" the virtual model display to view and inspect the design from several different viewpoints.

For the convenient control and maintenance of the hierarchical models, an explorer-based management window has been developed to display the relation of the global model to each of the component models by inter-connected representing nodes in a foldable tree hierarchy. Upon the clicking of each node in the hierarchy, a model setting interface will be opened to enable users to configure and modify the simulation settings, which include the model file, server IP address, number of processors requested, etc., for the model which the node represents. The management window also includes several buttons to open or close display windows for the model.

### 5.6 The fault tolerance

As mentioned in introducing the multi-level hierarchical modeling and simulation method, the assignment of tasks remains static for the entire duration of the simulation since load-balancing is not in the scope of this study. However, issues like faults or network disruptions need to be handled in such a static task allocation. Basically, the proposed system adopts and extends the fault tolerance mechanism proposed for the server-side system of Web-FEM (Chen and Lin 2008). In the case of network disruptions, the tasks submitted to remote server clusters will not be cancelled and the results will be stored in the pre-specified directories. When the network is recovered, the local client system will automatically reconnect with remote tasks or retrieve results of submitted tasks based on the pre-specified connection and deployment information. On the other hand, any system faults on the client or a server cluster must be resolved and the assigned simulation task must be recovered manually by its system administrator. Then the restarted task will automatically reconnect with remote task(s) by following the same method used in handling network disruptions.

## 6. System design and implementation

The proposed cluster-to-cluster distributed computing environment will require a modified client-server module with a formatted architecture to perform the multi-level hierarchical modeling and simulation. The software framework consists of two modules, as shown in Fig. 6. They are the core analysis module and the graphical user interface module. Both modules are implemented using Java, a platform-independent, net infrastructure language, so that the system can run on any (interface) computer connected to the Internet.

In the core analysis module, the interactions between the client and server are managed by the *Client Communicator* class and *Server Communicator* class. On the Client side, the *Coordinator* class has been designed to control and manage the simulation processes. And its corresponding class on the server side is the *RCM Analysis Server* class which receives the analysis tasks and controls the analysis program to analyze the task, and queue specific tasks for execution. The *Analyzer* classes, which handle the integrated analysis programs, have a synchronization property allowing them to establish the queuing function of the *RCM Analysis Server* class.

The hardware configuration is shown in Fig. 7. In the same way, the hardware configuration is divided into two parts. The graphical user interface module is installed on a personal computer (PC), and the core analysis module is orchestrated by one client cluster and at least one server cluster. As the figure shows, in the core analysis module, each cluster uses an interface computer to communicate, and the analyzing work is thereby administered by its own cluster system.

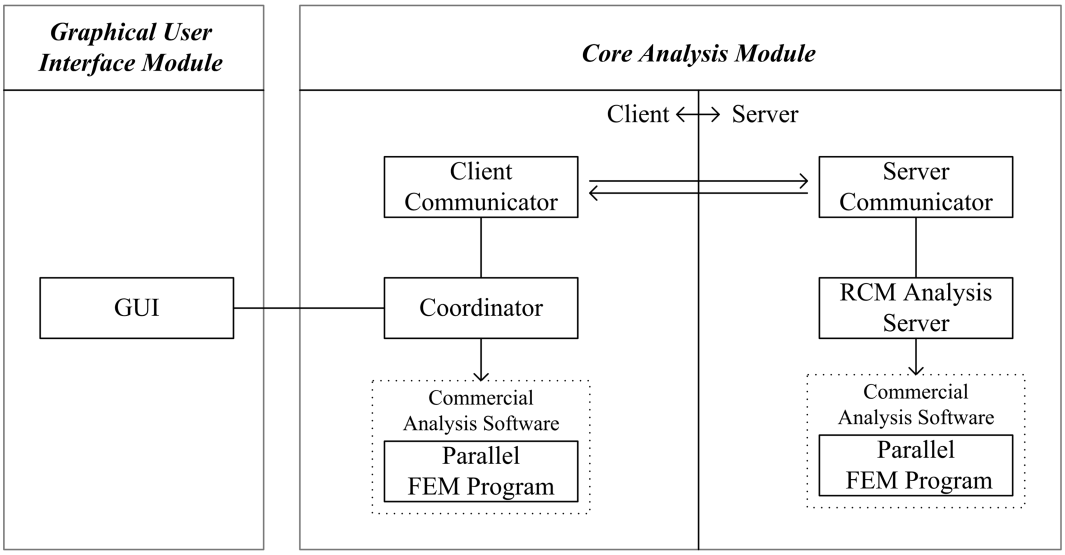


Fig. 6 Software framework of the proposed Internet-based simulation

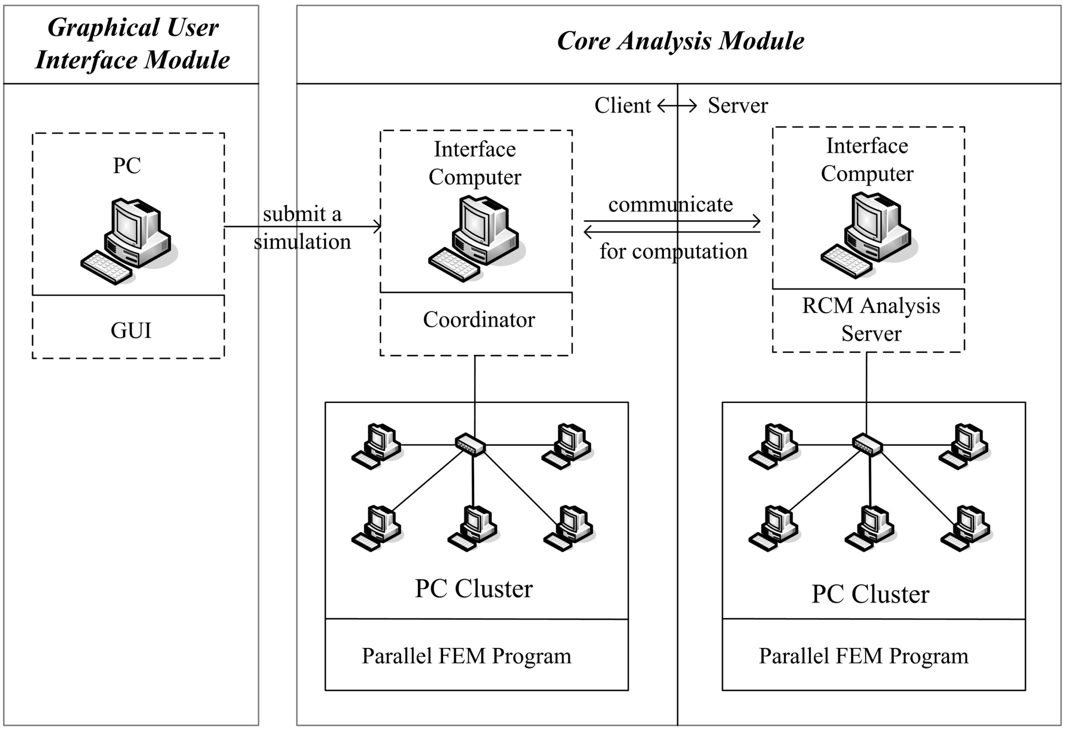


Fig. 7 Hardware configuration

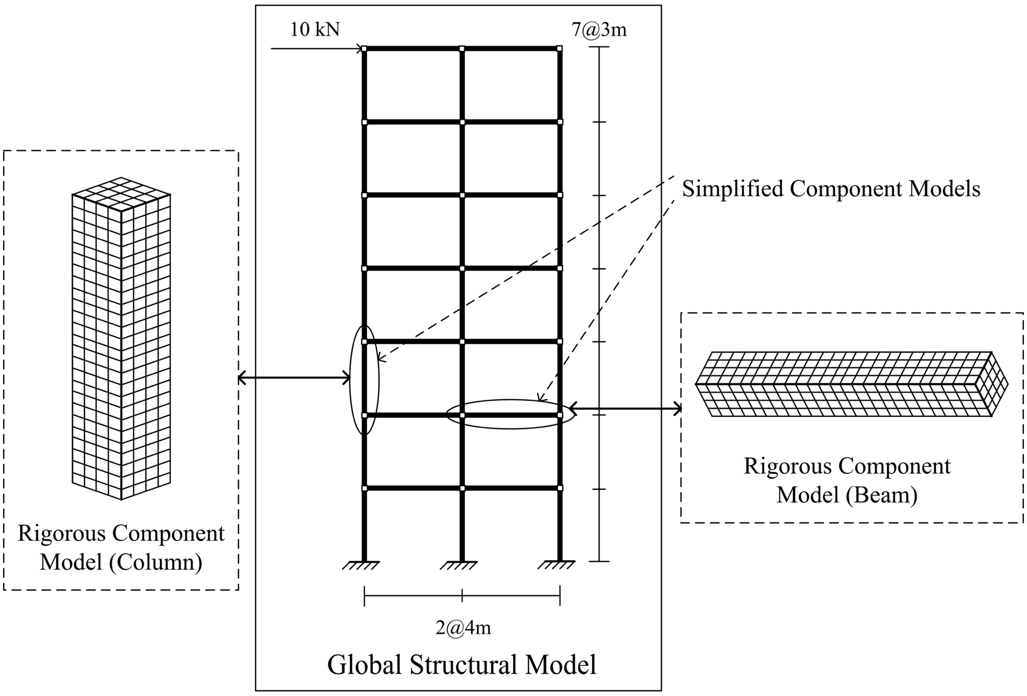
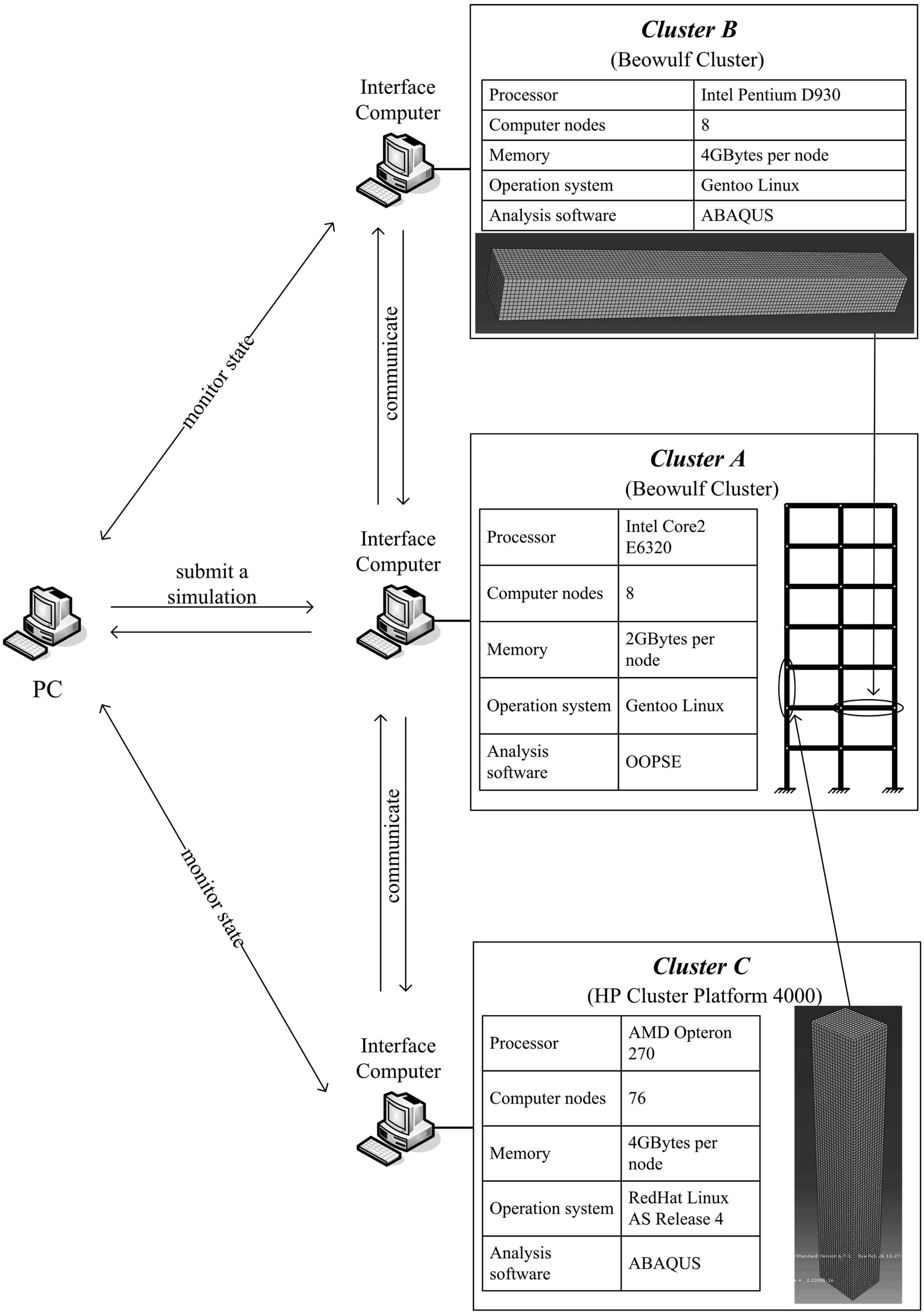


Fig. 8 The hierarchical model of the numerical case study for demonstration



***Cluster B***
(Beowulf Cluster)

| | |
|---|---|
| Processor | Intel Pentium D930 |
| Computer nodes | 8 |
| Memory | 4GBytes per node |
| Operation system | Gentoo Linux |
| Analysis software | ABAQUS |

***Cluster A***
(Beowulf Cluster)

| | |
|---|---|
| Processor | Intel Core2 E6320 |
| Computer nodes | 8 |
| Memory | 2GBytes per node |
| Operation system | Gentoo Linux |
| Analysis software | OOPSE |

***Cluster C***
(HP Cluster Platform 4000)

| | |
|---|---|
| Processor | AMD Opteron 270 |
| Computer nodes | 76 |
| Memory | 4GBytes per node |
| Operation system | RedHat Linux AS Release 4 |
| Analysis software | ABAQUS |

Fig. 9 The hardware and software configurations of the numerical case study

## 7. Numerical case study

The simulation of a simple two-level hierarchical model of a frame structure is presented to demonstrate the proposed system. As shown in Fig. 8, the global model is a 7-story frame structure comprised of hinged beam-column elements. A pushover analysis is performed by applying a point load at the upper left roof corner. Two structural components have been selected to be analyzed in detail using the rigorous models. For the hardware configurations, there are three clusters and one personal computer participating in the simulation. As shown in Fig. 9, the overall computational abilities and the assigned rigorous component models are shown beside these clusters. The state of each cluster is also monitored by the client in order to provide information to the user on the progress of each process, and also to reveal the source of any bottleneck in the process that might occur.

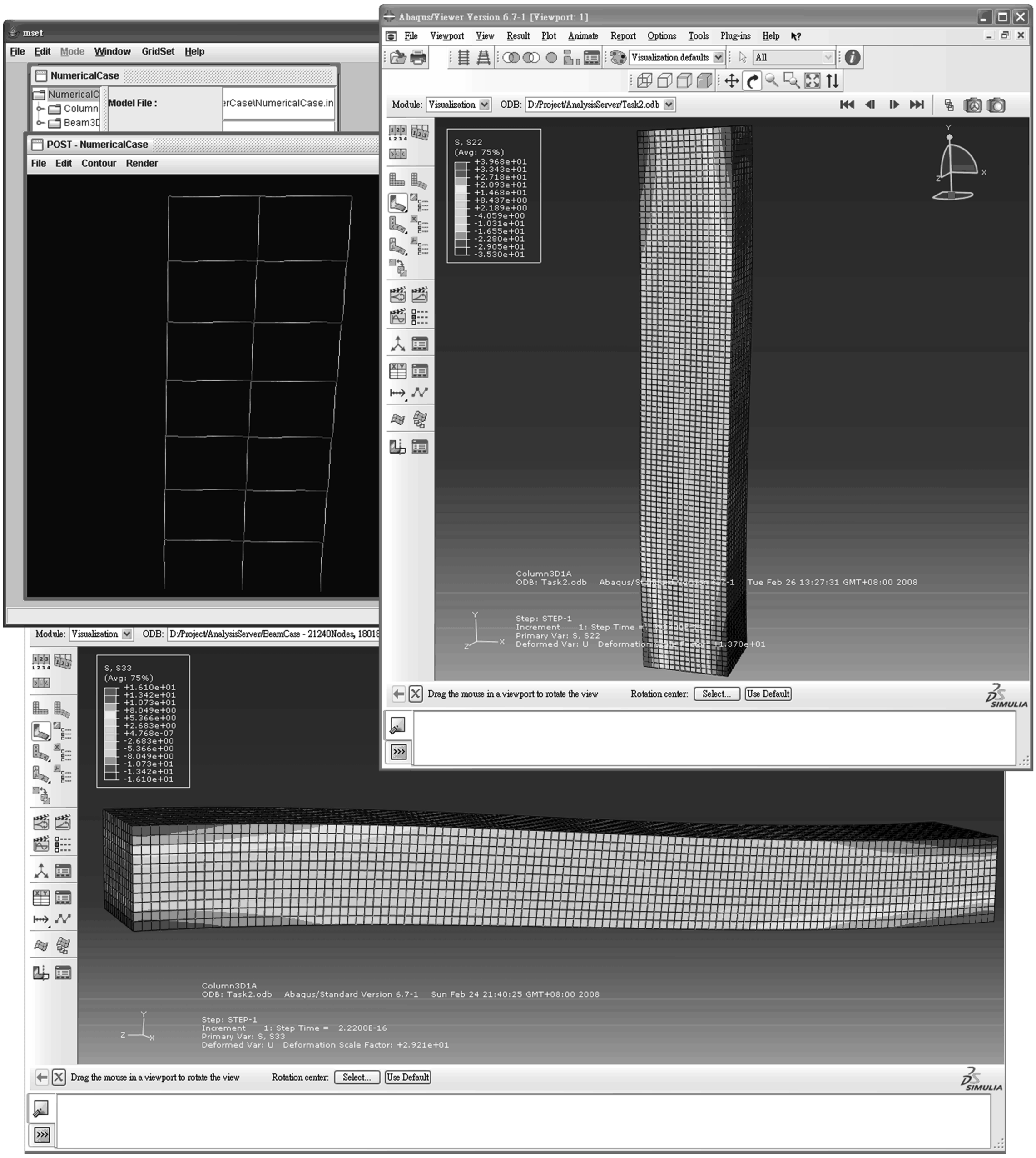
The sensitivity issues, such as the number of RCM and the mesh size of RCM, must be considered in developing a valid hierarchical model such as this example. Actually, the number of localized RCM analyses in conjunction with the global analysis can be freely chosen, since the global structural behavior of the system is mainly determined by its constitutive SCMs. Generally, RCM simulations are only provided for those critical members of concern or interest. However, providing RCM for each of the structural member is also doable if the available computing resources allow such a full simulation. On the other hand, the mesh size of RCM must be fine enough to accurately produce realistic behavior and response of the modeled component.

For the analysis tools, first, the global model is analyzed using a parallelized program, OOPSE, developed by Chen and Archer (2001), at cluster A. Cluster A is equipped with eight computers, all having a 2.4 GHz Intel processor and a 2GB memory. Then, the first column component, which is located on the third floor, is modeled using fine meshes consisting of the C3D8 3D solid elements provided by ABAQUS for stress analysis. Cluster B, assigned to analyze the corresponding rigorous model, is equipped with eight computers, each having a 2.8 GHz Intel processor and a 2GB memory. The second component chosen for analysis, located on the third floor, is modeled in detail using the same C3D8 3D elements. The rigorous component model is analyzed in cluster C. Although cluster C has a computing power of 0.92 TeraFlops, only four nodes are assigned there, so as to analyze the rigorous models equally in all cluster systems.

The nodal displacements of the simplified model in the global system are extracted and translated into the nodal displacements on the corresponding planes of the rigorous model, shown as known boundary conditions, to obtain the detailed responses of each component. The simulated results of this two-level hierarchical frame model, using the proposed system, are shown in Fig. 10.

This simulation took 90 seconds to complete by using the proposed cluster-to-cluster Internet computing framework. For comparison, the same simulation was also run by using a single computer, and it took 155 seconds to complete. As can be seen, the simulation time can be reduced by using the proposed framework. However, the main goal of this study is not to maximize the speedup performance. Instead, the goal is to integrate the available computing resources which are geographically distributed by Internet technology, since the computing units in the proposed computing environment are expected to be highly heterogeneous. In addition, the distributed computational tasks (SCMs) of the proposed simulation method are also expected to be unbalanced. Therefore, the traditional speedup plot for evaluating the performance of parallel processing was not done and presented for comparison.

As shown in Fig. 10, the numerical results of this simulation can provide both the global response



Model File : erCase\NumericalCase.in

Module: Visualization ODB: D:/Project/AnalysisServer/Task2.odb

S, S22
(Avg: 75%)
+3.968e+01
+3.343e+01
+2.718e+01
+2.093e+01
+1.468e+01
+8.437e+00
+2.189e+00
-4.059e+00
-1.031e+01
-1.655e+01
-2.280e+01
-2.905e+01
-3.530e+01

Column3D1A
ODB: Task2.odb Abaqus/S… …-1 Tue Feb 26 13:27:31 GMT+08:00 2008

Step: STEP-1
Increment 1: Step Time = …
Primary Var: S, S22
Deformed Var: U Deformation … +1.370e+01

Drag the mouse in a viewport to rotate the view Rotation center: Select... Use Default

Module: Visualization ODB: D:/Project/AnalysisServer/BeamCase - 21240Nodes, 18018

S, S33
(Avg: 75%)
+1.610e+01
+1.342e+01
+1.073e+01
+8.049e+00
+5.366e+00
+2.683e+00
+4.768e-07
-2.683e+00
-5.366e+00
-8.049e+00
-1.073e+01
-1.342e+01
-1.610e+01

Column3D1A
ODB: Task2.odb Abaqus/Standard Version 6.7-1 Sun Feb 24 21:40:25 GMT+08:00 2008

Step: STEP-1
Increment 1: Step Time = 2.2200E-16
Primary Var: S, S33
Deformed Var: U Deformation Scale Factor: +2.921e+01

Drag the mouse in a viewport to rotate the view Rotation center: Select... Use Default

Fig. 10 The simulation results of the numerical case study

of the modeled frame structure by the SCM-based global analysis, and the detailed responses of its critical components by the corresponding localized RCM analysis. As mentioned in proposing the hierarchical modeling and simulation method, most researches on analysis or simulation of structural behaviors in the published literature are either focused on dealing with a structural system using frame type model only, or an isolated structural component using mesh type model only. The proposed method actually integrates and correlates the two types of structural analysis to produce a more realistic simulation for structural systems in manageable detail. Therefore, in comparison with results from other research, the main difference is the proposed simulation method can provide both macro and micro level responses in an integrated and correlated result set.

A single-scale detail model was built to compare the solution accuracy of the proposed method in the numerical example. This single-scale detail model consists of 0.77 million C3D8 3D elements and 0.8 million nodes. Analysis work requires at least 4.5 GB of memory. With a parallel version of ABAQUS and two computing nodes (each running an Intel E6320 CPU with 4GB of memory), simulation of the detail took 1,200 seconds to complete. The left-hand side of the RCM beam was chosen to compare solution accuracy. The correlated interface of the single-scale detail model was the left-most part of the right-hand side beam section located on the third floor. The resulting comparison of the single-scale model and the numerical example indicated average percentage of error of strain in the beam section to be 3.08%.

## 8. Conclusions

This paper has presented a framework for an Internet-based structural simulation by integrating a cluster-to-cluster distributed computing environment and a multi-level hierarchical method of modeling and simulation. The purpose of this study is to utilize the idle and available computational resources which are geographically distributed on the Internet for providing the computing power needed for processing large-scale structural simulations. However, slow Internet communication is expected to be a significant bottleneck. To conquer this Internet-imposed obstacle, the Internet computing environment is first organized as a two-level parallel platform, which first utilizes local cluster computing, and then remote, cluster-to-cluster computing. A hierarchical modeling approach and computational procedures for the proposed cluster-to-cluster computing environment have been added to streamline the process and avoid excessive Internet communication. To fulfill the proposed concept, a prototype software system has been designed and implemented to perform the multi-level hierarchical modeling and simulation in a cluster-to-cluster distributed computing environment.

At present, most successful Internet applications address computational problems which contain a large number of independent computational tasks to be done in a distributed manner. Unlike such Internet applications, this study tried to execute a single large-scale computational task, which is the detailed behavior simulation of structural systems, using an Internet platform. For such a computational problem, which involves high coupling between decomposed tasks, proposing a computational method and a software framework as one of the possible solutions to realize such applications by overcoming the relatively low communication bandwidth of the Internet in an Internet-based cluster-to-cluster computational platform is the main focus of this study.

The principal focus of this study was to apply the proposed simulation methods in non-linear static analysis. At present, the proposed 2-level coupled RCM-SCM method is inappropriate for dynamic problems, as several issues must be overcome. Considering the vibration mode in the structure dynamic, RCM dynamic behavior should represent a combination of its major modes. Due to scale differences between SCM and RCM nodes, using displacement compatibility to synchronize SCM with RCM may not provide adequate information to reflect RCM dynamic behavior correctly. Thus, it is not appropriate to use displacement compatibility to synchronize RCM wave propagation with SCM. A potentially more rewarding approach to synchronize the two in dynamic is to apply major RCM modes to the displacement compatibility derived from SCM. Such will produce additional compatibility conditions. Another potentially rewarding approach would be to increase SCM nodes in order to provide greater compatibility information. These two methods may be studied and discussed further to evaluate their possible future applications to dynamic problems.

## References

Adams, M.F., Bayraktar, H.H., Keaveny, T.M. and Papadopoulos, P. (2004), "Ultrascalable implicit finite element analyses in solid mechanics with over a half a billion degrees of freedom", *Proceedings of the SC2004 Conference: High Performance Networking and Computing*, ACM/IEEE, Pittsburgh, PA, USA.

Adeli, H. and Kumar, S. (1995), "Distributed finite element analysis on network of workstations-algorithms", *J. Struct. Eng.-ASCE*, **121**(10), 1448-1455.

Alonso, J.M., De Alfonso, C., Garcia, G. and Hernandez, V. (2007), "GRID technology for structural analysis", *Adv. Eng. Softw.*, **38**(11-12), 738-749.

Alonso, J.M., Hernandez, V. and Molto, G. (2008), "A high-throughput application for the dynamic analysis of structures on a grid environment", *Adv. Eng. Softw.*, **39**(10), 839-848.

Chen, H.M. and Archer, G.C. (2001), "A distributed object-oriented finite element analysis program architecture", *J. Comput. Aid. Civil Infrastruct. E.,* **16**(5), 326-336.

Chen, H.M. and Archer, G.C. (2005), "New domain decomposition algorithm for nonlinear substructures", *J. Comput. Civil Eng.-ASCE*, **19**(2), 148-159.

Chen, H.M. and Iranata, D. (2008), "Realistic simulation of reinforced concrete structural systems with combine of simplified and rigorous component model", *Struct. Eng. Mech.*, **30**(5), 619-645.

Chen, H.M. and Lin, Y.C. (2008), "Web-FEM: An Internet-based finite element analysis framework with 3d graphics and parallel computing environment", *Adv. Eng. Softw.*, **39**(1), 55-68.

Chen, S.X. (2005), "A communication technology for Internet-based collaboration optimization for distributed structural systems", *Struct. Multidiscip. O.,* **29**(5), 391-397.

D'Ambrisi, A. and Fillippou, F.C. (1999), "Modeling of cyclic shear behavior in RC Members", *J. Struct. Eng.-ASCE*, **125**(10), 1143-1150.

Dolenc, M., Katranuschkov, P., Gehre, A., Kurowski, K. and Turk, Z. (2007), "The inteligrid platform for virtual organisations interoperability", *Electon. J. Inform. Tech. Constr.,* **12**, 459-477.

Dolenc, M., Kline, R., Turk, Z., Katranuschkov, P. and Kurowski, K. (2008), "Semantic grid platform in support of engineering virtual organizations", *Informatica (Ljubljana)*, **32**(1), 39-49.

Farhat, C. and Roux, F.X. (1991), "A method of finite element tearing and interconnecting and its parallel solution algorithm", *Int. J. Numer. Meth. Eng.*, **32**(6), 1205-1227.

Hajjar, J.F. and Abel, J.F. (1989), "Parallel processing of central difference transient analysis for three-dimensional nonlinear framed structures", *Commun. Appl. Numer. Meth.*, **5**(1), 39-46.

Kumar, S. and Adeli, H. (1995), "Distributed finite element analysis on network of workstations-implementation and applications", *J. Struct. Eng.-ASCE*, **121**(10), 1456-1462.

Kwon, O.S., Nakata, N., Elnashai, A.S. and Spencer, B. (2005), "A framework for multi-site distributed simulation and application to complex structural systems", *J. Earthq. Eng.*, **9**(5), 741-753.

Kwon, O.S., Elnashai, A.S. and Spencer, B.F. (2008), "A framework for distributed analytical and hybrid simulations", *Struct. Eng. Mech.*, **30**(3), 331-350.

Lam, K.P., Mahdavi, A., Gupta, S., Wong, N.H., Brahme, R. and Kang, Z. (2002), "Integrated and distributed computational support for building performance evaluation", *Adv. Eng. Softw.*, **33**(4), 199-206.

Lam, K.P., Wong, N.H., Mahdavi, A., Chan, K.K., Kang, Z. and Gupta, S. (2004), "SEMPER-II: An Internet-based multi-domain building performance simulation environment for early design support", *Automat. Construct.*, **13**(5), 651-663.

Li, Y., Xu, X. and Qiu, Q. (2006), "FEM-based structure optimization with grid-enabled analysis environment", *Proceedings of the 6th World Congress on Intelligent Control and Automation*, Dalian, China.

Modak, S. and Sotelino, E.D. (2000), "The iterative group implicit algorithm for parallel transient finite element analysis", *Int. J. Numer. Meth. Eng.*, **47**(4), 869-885.

Nuggehally, M., Liu, Y.J., Chaudhari, S.B. and Thampi, P. (2003), "An Internet-based computing platform for the boundary element method", *Adv. Eng. Softw.*, **34**(5), 261-269.

Pan, P., Tada, M. and Nakashima, M. (2005), "Online hybrid test by Internet linkage of distributed test-analysis domains", *Earthq. Eng. Struct. Dyn.*, **34**(10), 1407-1425.

Peng, J. and Law, K.H. (2002), "A prototype software framework for Internet-enabled collaborative development of a structural analysis program", *Eng. Comput.*, **18**(1), 38-49.

Peng, J. and Law, K.H. (2004), "Building finite element analysis programs in distributed services environment", *Comput. Struct.*, **82**(22), 1813-1833.

Song, W., Keane, A., Eres, H., Pound, G. and Cox, S. (2003), "Two dimensional airfoil optimization using CFD in a grid computing environment", *Euro-Par 2003, LNCS 2790*, 525-532.

Spenser, Jr., B., Finholt, T.A., Foster, I., Kesselman, C., Beldica, C., Futrelle, J., Gullapalli, S., Hubbard, P., Liming, L., Marcusiu, D., Pearlman, L., Severance, C. and Yang, G. (2004), "NEESgird: A distributed collaboratory for advanced earthquake engineering experiment and simulation", *Proceedings of the 13th World Conference on Earthquake Engineering*, Vancouver, B.C., Canada.

Takahashi, Y. and Fenves, G.L. (2006), "Software framework for distributed experimental-computational simulation of structural systems", *Earthq. Eng. Struct. Dyn.*, **35**(3), 267-291.

Vacharasintopchai, T., Barry, W., Wuwongse, V. and Kanok-Nukulchai, W. (2007), "Semantic web services framework for computational mechanics", *J. Comput. Civil Eng.*, **21**(2), 65-77.

Yang, Y.S., Hsieh, S.H., Tsai, K.C., Wang, S.J., Wang, K.J., Cheng, W.C. and Hsu, C.W. (2007), "ISEE: Ny 66 simulation for earthquake engineering - Part I: Database approach", *Earthq. Eng. Struct. Dyn.*, **35**(15), 2291-2306.

Yang, Y., Lu, J. and Elgamal, A. (2004), "A web-based platform for computer simulation of seismic ground response", *Adv. Eng. Softw.*, **35**(5), 249-259.

Zheng, Y., Song, G., Zhang, J. and Chen, J. (2004), "An enabling environment for distributed simulation and visualization", *Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing*, Pittsburgh, USA.