Journal of Applied Mechanics Vol.12, pp.623-629 (August 2009) JSCE

BASIC STUDY ON ENHANCING IES WITH PARALLEL COMPUTATION

Gholamreza SOBHANINEJAD*, Muneo HORI**, Toshikazu KABEYASAWA***

*Graduate Student, Earthquake Research Institute, University of Tokyo (Bunkyo, Tokyo 113-0032)
E-mail: sobhani@eri.u-tokyo.ac.jp
**Professor, Earthquake Research Institute, University of Tokyo (Bunkyo, Tokyo 113-0032)
E-mail: hori@eri.u-tokyo.ac.jp
***Assistant Professor, Earthquake Research Institute, University of Tokyo (Bunkyo, Tokyo 113-0032)
E-mail: tosikazu@eri.u-tokyo.ac.jp

Integrated earthquake simulation (IES) is seamless simulation of the three earthquake processes, namely, earthquake hazard process, earthquake disaster process and anti-disaster action process. High performance computing is essential if IES, or particularly, simulation of earthquake disaster process is applied to such an urban area in which structures of 10^6 are located. IES is enhanced with parallel computation, and its performance is examined, so that earthquake disaster simulation will be made for a model of an actual city, by inputting strong ground motion. It is shown that parallel IES has fairly well scalability even when advanced non-linear seismic structure analysis method is employed.

Key Words : *integrated earthquake simulation, high performance computing, parallel computation performance, software engineering*

1. Introduction

While basic characteristics of earthquake hazard are not changed, earthquake disaster *evolves* itself as society is changed. In Japan, for instance, a tremendous impact will be made on domestic and global economy if an inter-plate earthquake hits a large area. Even if buildings, structures or lifelines are not collapsed, mal-functioning of a number of these assets will be a source of such impact. Furthermore, longer period will be needed to repair mal-functioning due to shortage of work force. To make better preparation, it will be the first step to develop a hazard map of next generation that is able to show consequences of such earthquakes.

The authors have been developing integrated earthquake simulation (IES)[1)], in order to make a hazard map which is based on numerical simulation. Seamless simulation is made for the three processes; 1) earthquake hazard process in which earthquake waves propagate in the crust; 2) earthquake disaster process in which structures are shaken by strong ground motion; and 3) anti-disaster action process which includes urgent evacuation and recovery of social functions. Simulation of the third process is essential in predicting secondary or indirect disasters of an earthquake which includes impacts on economic activities.

High performance computation (HPC) is required for IES, when it is applied for a larger urban area. In particular, HPC will be essential in simulating the earthquake disaster process; in order to analyze seismic response, which may lead to damage or collapse, for at most 10^6 structures, use must be made of an advanced non-linear numerical method of analyzing seismic structure response, which needs moderate or massive computer resources, depending on the size of a target structure and the characteristics of the numerical method. The performance of HPC should be evaluated if IES is practically used as a hazard map of next generation. Faster prediction of earthquake hazard and disaster will be required for IES.

The contents of this paper are as follows: first, the need of HPC for IES is explained in Section 2, considering the cases when IES is used to make urgent prediction of earthquake disaster when an earthquake actually happens. In Section 3, the techniques that are used in enhancing IES with parallel computation is explained, emphasizing the treatment of data with complex structure and huge amount via MPI library. In Section 4, numerical experiments are made to evaluate the performance of parallelized IES; a model for a part of Tokyo in which around 10,000 structures are located is used. It should be noted that IES uses C++ and a class is used to designate an object which is coded in developing IES.

2. Need of HPC for IES

In constructing a hazard map, statistical analysis which is based on past earthquake disaster records is applied to a target area so that a possible earthquake disaster is estimated for a predicted earthquake hazard. The source of the statistical analysis is fragility curve that relates a certain seismic index to damage possibility of structures, depending on their type or age. The quality of the statistical analysis is not highest and there is limitation of predicting structure damage based on the fragility curves. It is structure analysis which is used in design; the analysis examines the performance of a structure for a given strong ground motion by computing the seismic responses.

IES is a possibly better replacement of the fragility curves, since it numerically analyzes seismic responses of structures. More accurate prediction is made for earthquake disaster, if a better structure analysis method with a better model is used as well more accurate information for strong ground motion is given. While the basic concept of IES of simulation-based earthquake prediction is appealing, the verification of IES results is difficult, this is mainly because only limited data are available for mechanical properties of structures and buildings; see [2].

Beside for constructing a hazard map, it is important to make urgent prediction of earthquake disaster when a large earthquake happens. IES is able to make such prediction if observed strong ground motion is input to a model of the target area. For such urgent prediction, the speed of simulation is important; to establish fast and quick response to earthquake disaster, more emphasis should be put on the speed even with the accuracy being sacrificed to some extent.

In order to increase the speed of simulation, IES has to take advantage of HPC. A most efficient execution is required so that the urgent prediction of earthquake disaster provided by IES will be useful; indeed, the execution time must be evaluated. There are several factors which influence the execution time. Two standard parameters are the computer node number and the problem size, namely, the number of structures which are analyzed. Also, a parameter which is inherent to IES is the numerical method of analyzing seismic structure responses that is enrolled in it.

It is worth to mention a *virtual urgent prediction* of earthquake disaster, which simulates the earthquake disaster process for various urban areas by inputting strong ground motion which is observed for an actual earthquake. Literally, this prediction is virtual since target areas are not hit by the earthquake. However, it appeals public concerns for earthquake disaster which drastically increases right after a big earthquake. The timing of releasing the virtual urgent prediction is important not to miss the best opportunity that individuals, government officials and industries recognize the threat of a possible earthquake and are readily driven to consider earthquake mitigation.

3. Implementation of Parallel Computation

Among HPC architectures, cluster system is more popular since it is economical and accessible. Message Passing Interface (MPI) library[3),4),5),6)] is available to make parallel computation at hand. For IES, however, the use of the standard MPI library is not feasible. This is because data used in IES are complicated. In order to enhance IES with parallel computation, the authors develop two abstract classes, namely, Generic Distributable Class (GDC) and MPI_Process_Manager Class, which will be explained later in this section.

At this moment, IES adapts layer-based design[7)] which consists of data, simulation and visualization layers; see Appendix A for the old design of IES. Common Modeling Data (CMD) is used for inter-layer data transfer; see **Fig. 1** and **Fig. 2**. Coding effort is minimized by introducing CMD. For instance, suppose that there are N, M and L elements in the data, simulation and visualization layers. The number of the inter-layer data transfer becomes $N \times M \times L$ if one element is converted to another element. The number is reduced to $N + M + L$ if all elements are converted to CMD. Accordingly, coding effort for the data conversion is reduced, even though the conversion must be twice for one inter-layer data transfer.

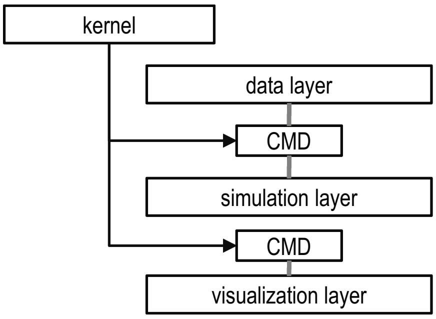

Fig. 1 Schematic view of layer-based design of IES.

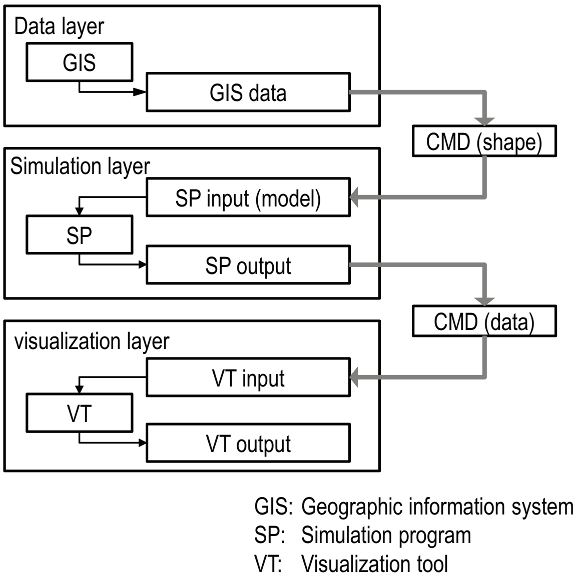

Fig. 2 Intra- and inter-layer data transfer of IES.

3.1 GDC

GDC is an abstract class which offers two functionalities, Serialization() and De-serialization(). In **Fig. 3**, presented is an inheritance model of GDC which posses these two functionalities. Serialization() interprets a class data structure as a primitive data structure, and De-serialization() regenerates the class from the interpreted data. In another word, Serialization() and De-serialization() are overloading MPI functions of Pack() and Unpack() with specific consideration made to the complex data structure of CMD.

As an example, Shape class has two member functions to read and write data of Shape class in a file or another input-output stream. These two member functions can be used as a function pointer argument to accomplish appropriate Serialization() and De-serialization() of Shape class that is achieved with

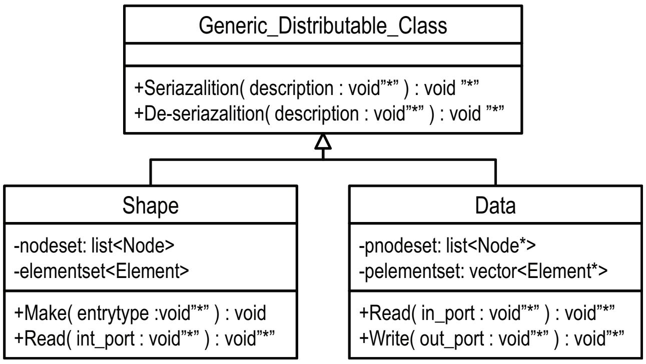

Fig. 3 Design of GDC.

GDC.

3.2 MPI_Process_Manager class

MPI_Process_Manager class is introduced to make easy use of the MPI library in IES, achieving the implementation of parallel computation environment to IES in a general manner so that this class could be used for other simulation systems; see **Fig. 4**. MPI_Process_Manager class is responsible of initialization cluster nodes, sending to and receiving from each cluster node in a processor group. Correspondingly, Send_Serializable_Object() and Receive_Serializable_Object() are made so that they play a role of a wrapper for Serialization() and De-serialization() of a GDC object. Access to Serialization() and De-serialization() is prepared through a function pointer which encapsulates implementation of these two functions.

3.3 IES process hierarchy

A clientserver model is selected for the parallel computing architectural model, and first-come-first-served is used as load balancing strategy. The kernel in association with the data layer works as server, while aggregation of simulation and visualization layers make client parts.

Parallel communication diagram of IES is shown in **Fig. 5**. Server side consists of the kernel, and the kernel initializes computational nodes of client side. A client sends confirmation message for the server side, and starts the process of modeling and simulation. After the client processes the visualization, it requests a new task for the server side. When the kernel finishes all the simulations assigned to the client sides, it finalizes all the computational nodes.

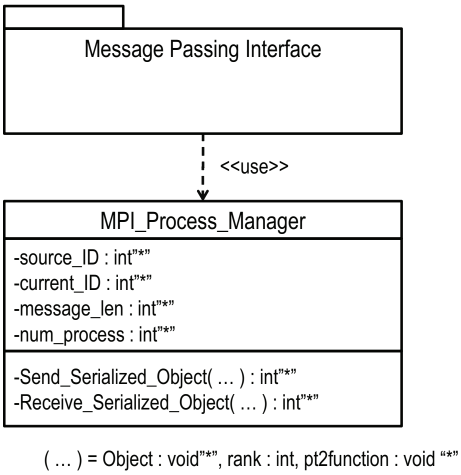

Fig. 4 Design of MPI_Process_Manager Class.

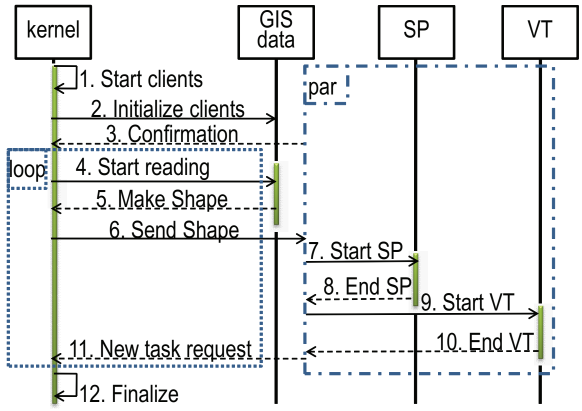

Fig. 5 Schematic view of IES parallel computation process.

4. Numerical Experiments

The performance of HPC enhanced IES is studied by carrying out numerical experiments. In particular, the complexity of the analysis methods is examined by using two seismic analysis methods which require utterly different computational resources. The methods are linear Multi-Degree-Of-Freedom (MDOF) analysis and non-linear Distinct Element Method (DEM) analysis. The performance is measured in terms of speedup and efficiency, and scalability is estimated using iso-efficiency function as an index.

4.1 Model construction

The data conversion from GIS data to two analysis models for one structure is schematically shown in **Fig. 6**. After GIS data are transform into a CMD entity (which is called Shape), two analysis models are constructed from this entity.

The procedures of the data conversion for MDOF are summarized as follows: 1) the number of floors is calculated from the structure height; 2) for each floor, the weight is calculated from the floor area; and 3) spring constants between the floors are determined so that the natural frequency and damping constant of the first and second modes coincide with the empirical formulas.**Table 1** shows assumed relation between floor area, height and structural type; see [8] for more detailed information of constructing an MDOF model.

DEM plugged in IES uses two sets of spring and

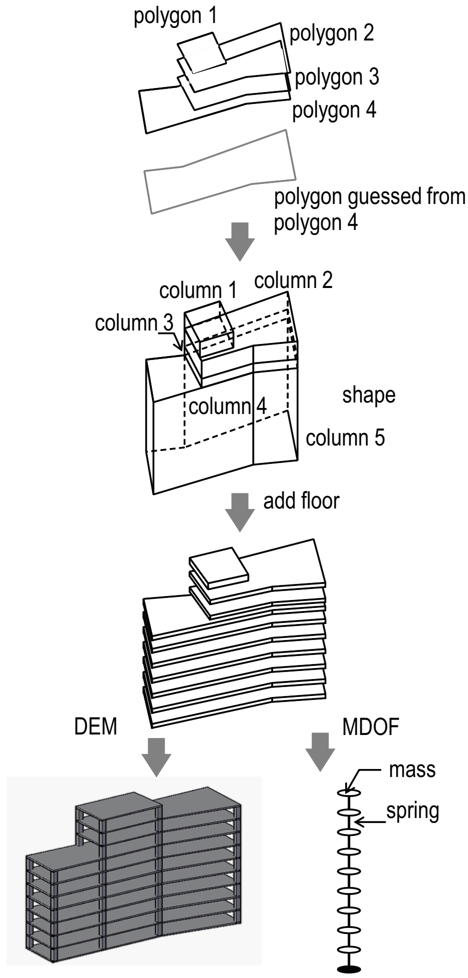

Fig. 6 Schematic view of data conversion from GIS data to structural model.

dashpot. The first set represents contact between elements, and the second set represents joint of wooden house whose mechanical properties have been experi-

Table 1 Structure categorization based on floor area and building height.

Height (m)	Floor area (m²)	
	10 ~100	>100
3-12	Timber	Steel
12-50	RC	RC
>50	Steel RC	Steel

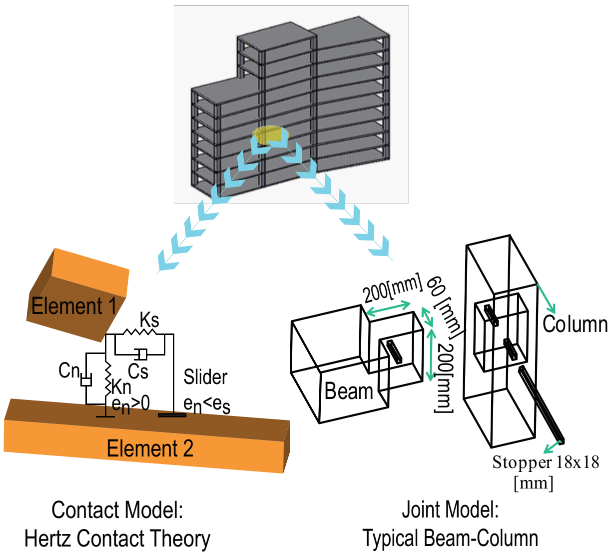

Fig. 7 DEM joint models.

mentally measured. **Fig. 7** shows the two sets of joint that the present DEM analysis uses. All structures in target area are assumed as a wooden house when the non-linear DEM is applied; see [8] for more detailed information of constructing a DEM model.

4.2 Performance measurement

Speedup, efficiency, and iso-efficiency function are used to measure the performance of HPC enhanced IES. These indexes are obtained by executing the IES simulation for 100, 1,000, 5,000, and 10,000 structures on 8, 16, 32, and 64 computational nodes. A cluster with eight AMD Opteron2356 QuadCore processors (2.3 GHz) with 4GB memory for each processor is used for the performance measurement. Input strong motion is synthesized in the simulation of earthquake hazard process; see [8] for more detailed information of synthesizing the strong ground motion.

The execution time of the IES simulation that uses p computational nodes is denoted by $T(p)$. Speedup is then defined in terms of this execution time as follows:

$$S(p) = \frac{T(1)}{T(p)}. \tag{1}$$

Efficiency is defined by normalizing speedup by the number of computational nodes, i.e.,

$$E(p) = \frac{S(p)}{p}. \tag{2}$$

Fig. 8 shows speedup for MDOF and DEM analyses. In the ideal case of parallel computing, the relation between the number of processors and the speedup becomes linear. As is seen, the speedup curve tends to be close to this ideal case as the number of processors or the number of simulated structures increases. The tendency is more obvious for the DEM analysis than for the MDOF analysis. The message passing overhead reveals when analysis method becomes less time-consuming. Since IES is eventually aimed at mega-cities like Tokyo, the increase in speedup for larger number of structures is a good sign of the performance.

Fig. 9 illustrates efficiency for the both analyses. Implicitly, the efficiency indicates the amount of time in which one computational node is working. As is seen, for the MDOF analysis, the efficiency is small; this is another evidence that message passing overhead is dominant in saved time for the MDOF analysis. As the number of analyzed structures increases, the efficiency increases for a fixed number of computational nodes. However, the efficiency decreases as the number of computational nodes increases.

A system, a program or an algorithm is called scalable if there is a function of the number of processors which gives the problem size that is solved with constant efficiency. The following equation gives the relation between number of processors and the number of structures that have more or less same efficiency:

$$n = 1.44 * p^{2.23}, \tag{3}$$

where n is the number of analyzed structures.

Fig. 10 shows estimated iso-efficiency function; the thick line is the iso-efficiency function given as Eq. 3. The iso-efficiency function is of the power low, and hence it is shown that the present IES has scalability

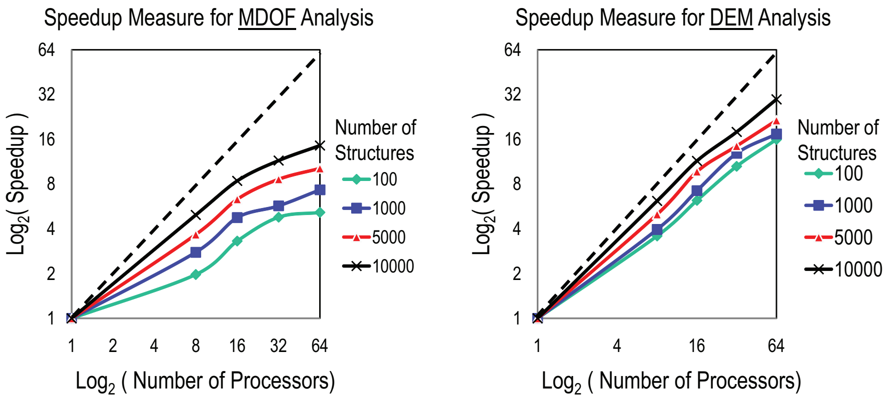

Fig. 8 Speedup for MDOF and DEM analyses.

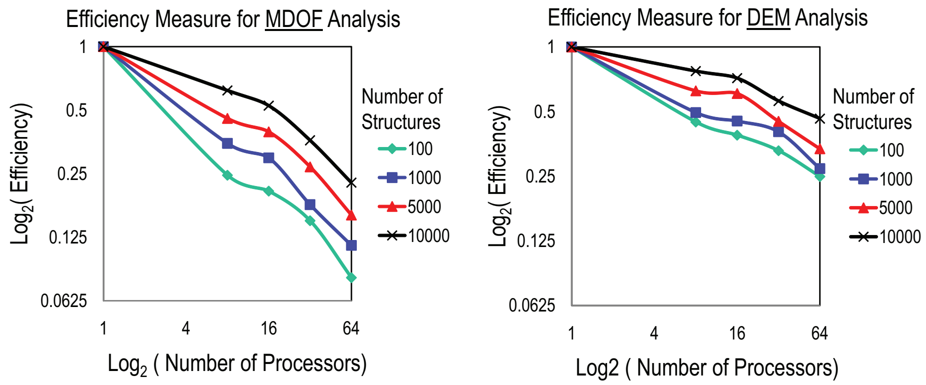

Fig. 9 Efficiency for MDOF and DEM analyses.

from degree two. Several factors influence the scalability degree, such as message passing latency, external storage access. Non-homogeneity of analysis models in IES environment could be a significant factor of the scalability; non-homogeneity is caused by different type of structures which needs specific model construction from Shape entity and an analysis method inherent to it so that execution and data processing times are changed.

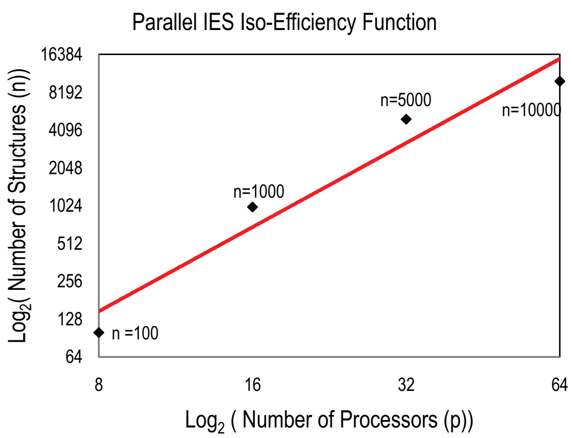

Fig. 10 Iso-efficiency function for MDOF and DEM analyses.

5. Concluding Remarks

This paper presents the enhancement of IES with parallel computation by using GDC and MPI_Message_Passing Class. Although the authors do not think that the basic concept of them is original, well-structured classes are coded so that complicated and huge data which are used in IES are easily handled through several layers of IES.

IES scalability is studied for the linear MDOF analysis and the nonlinear DEM analysis. The iso-efficiency functions show that IES scalability is independent from the analysis methods, even though the value of the efficiency is 0.25 and 0.47 for the MDOF and DEM analyses, respectively. It is thus expected that IES has similar efficiency for other analysis methods.

Efficiency can be improved by using shared-memory parallelization in client side, which should be studied in future studies. This will contribute to increase the speed of IES execution so that it can be used as a hazard map of next generation or a virtual urgent prediction of earthquake disaster.

REFERENCES

1) M. Hori, *Introduction to Computational Earthquake Engineering*, Imperial College Press, 2006.
2) T. Ichimura, M. Hori, K. Terada and T. Yamakawa, On integrated earthquake simulator prototype: combination of numerical simulation and geographical information system, *Structural Eng./Earthquake Eng.*, JSCE, 9, 4, 1-12, 2004.
3) W. Gropp, E. Lusk, A. Skjellum, *Using MPI, portable parallel programming with message-passing interface*, The MIT Press, 1999.
4) B. C. McCandless, J. M Squyres, A. Lumsdaine, Object Oriented MPI (OOMPI), a class library for the Message Passing Interface, *MPI Developer's Conference*, Proceedings, 87-94, 1996.
5) A. Grama, G. Karypis, V. Kumar, A. Gupta, *Introduction to Parallel Computing, 2/E*, Addison-Wesley, 2003.
6) P. S. Pacheco, *Parallel programming with MPI*, Morgan Kaufmann Publishers, Inc, 1996.
7) S. Nallathamby, M. Hori, Y. Ariga, Improvement of IES Wrapper for Plug-in of Advanced Simulation Programs, *Journal of Applied Mechanics*, 10 , 573-579, 2007.
8) Gh. Sobhaninejad, M. Hori, T. Kabayeswa, nhancing Integrated Earthquake Simulation Environment with High Performance Computing, *Journal of Advances in Engineering Software*, in review.

Appendix A Improvement of IES

Originally, IES adopts wrapper-based design; a wrapper is developed for each simulation program, so that it handles communication with the IES system kernel; communication includes receiving input data, executing the program and sending output data; see

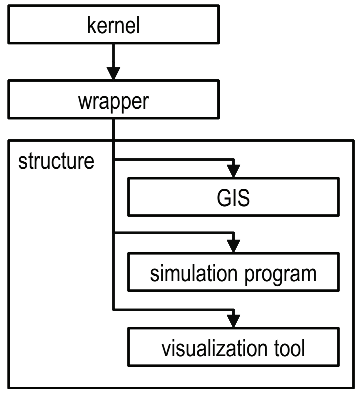

Fig. 11 Schematic view of wrapper-based design of IES.

Fig. 11. As mentioned in Section 3, each wrapper has to communicate data sets and visualization tools, and coding efforts increases as the number of the sets and tools increases. Moreover, there is a possibility that wrappers increase cohesion between processes during IES hierarchy; for instance, cohesion happens if several wrappers simultaneous use the same data set.

To overcome the disadvantages of wrapper design, the current IES adapts layer-based design. As shown in **Fig. 1**, the layer-based design is described in terms of three layers; each layer consists of aggregated abstract classes, and the kernel manages initialization, connection, and execution of each layer as control room.

The key feature of the layer-based design is the data conversion between two neighboring layers. CMD is developed for this purpose. The data structure of CMD is similar to input and output data of finite element analysis. It is certainly true that quantifying the improvement in the simulation coherency is difficult by employing the layer-based design instead of the wrapper-based design. However, by nature, less conjunction of the simulation coherency could happen for the layer-based design that limits the data conversion between neighboring layers.

(Received April 9, 2009)