



ELSEVIER

Advances in Engineering Software 34 (2003) 261–269

ADVANCES IN  
ENGINEERING  
SOFTWARE

[www.elsevier.com/locate/advengsoft](http://www.elsevier.com/locate/advengsoft)

# An Internet-based computing platform for the boundary element method

M. Nuggehally, Y.J. Liu\*, S.B. Chaudhari, P. Thampi

*Computer-Aided Engineering Research Laboratory, Department of Mechanical Engineering, University of Cincinnati,  
P.O. Box 210072, Cincinnati, OH 45221-0072, USA*

Received 28 May 2002; revised 4 November 2002; accepted 3 February 2003

## Abstract

Computers combined with the Internet are dramatically changing the engineering practices in design and analysis. More and more engineering software applications are moving to the Internet, prompted by the promising advantages, such as easy access for users everywhere with an Internet connection, easy upgrade of the software, easy control of the software. All these advantages will contribute to faster and cost-effective engineering software development and applications. Successful applications of the Internet computing depend largely on the speed of data transfer on the network. The boundary element method (BEM) has the inherent advantages over other domain-based numerical methods, because the size of the BEM model files are always considerably smaller, leading to faster data transfer on the current network. In this paper, a successful investigation in implementing the BEM on the Internet is presented. A BEM code for 2D elastostatic problems is used as the first solver in this work. A graphical-user interface (GUI) for the pre- and post-processing using Java, which provides platform-independent applications, is developed and implemented on the Internet. Demonstration problems using the developed platform clearly show the feasibility of the Internet-based computing and the potentials of this platform for future BEM development with further research.

© 2003 Elsevier Science Ltd. All rights reserved.

*Keywords:* Internet computing; Boundary element method; 2D elasticity; Java

## 1. Introduction

As information technology (IT) is penetrating rapidly into every corners of our society, many engineering practices have changed dramatically. Traditional design and analysis practices using hand drawings and calculators have been replaced by computer-aided design (CAD) and computer-aided engineering (CAE). Applications of CAD/CAE have significantly shortened the product development life cycle, reduced product cost and increased its competitiveness. With the popularity of CAD/CAE in various industries, demand has also been high for engineers who have adequate and most up-to-date training in both IT and engineering technologies. Researchers, engineers, educators and students in the traditional engineering fields, are consequently facing great opportunities as well as serious challenges with the endless tidal waves from IT innovations. For example, researchers doing cutting-edge research often have to utilize numerous simulation software tools for their

project, such as, in the multi-scale modeling of advanced nanomaterials, where simulation software must be able to cover the length scales from nano, micro to macro meters. These software tools are often developed by different research groups or companies and thus need different training to use them. Engineers are facing constant needs to update their knowledge and learn new computer languages and software tools also at their jobs. However, the time and cost in purchasing, installing, maintaining and learning the vast arrays of CAD/CAE or other engineering software, as is well known, have increased substantially in all industries. One possible solution to these problems is to develop Internet-based computing platform, which promises to eliminate these barriers associated with the applications of various CAD/CAE software as mentioned above.

Increased use of the Internet in engineering has led to the concept of computing through the Internet. Users of a software do not have to install the software locally anymore. Instead, users can access the software through the Internet at no charge or pay only for the actual use. This concept can dramatically change the way engineers do computing. It can significantly reduce the cost in software applications and

\* Corresponding author. Tel.: +1-513-556-4607; fax: +1-513-556-3390.  
E-mail address: [yijun.liu@uc.edu](mailto:yijun.liu@uc.edu) (Y.J. Liu).

greatly increase the accessibility. With quality online documentation and communication channels, the time and cost in learning various new software and in doing collaborative work can also be reduced dramatically. It is strongly believed that the Internet-based computing will be the future-computing model in engineering.

This paper describes a study in exploring the advantages and potentials of the Internet-based computing concept by the authors. An Internet-based platform for the boundary element method (BEM) has been developed as a first attempt in this direction. A successful experiment in implementing the BEM on the Internet is presented. A BEM code for 2D elastostatic problems is used as the solver in this test. A graphical-user interface (GUI) for the pre- and post-processing using the Java language, which renders platform-independent applications, has been developed and implemented on the Internet. The purpose of this paper is to stimulate more interests and activities in the CAE research community on the development of Internet-based applications, since the Internet computing will almost certainly become a dominant model in engineering design and analysis.

## 2. The boundary element method

The BEM, based on boundary integral equation formulations, is a boundary-based numerical solution technique, which was pioneered for potential and elasticity problems in the 1960s [1,2]. Instead of solving the original partial differential equations in the problem domain, these differential equations are first transformed into boundary integral equations using the Green's identity and fundamental solution associated with the problem. For example, for an elastostatic problem defined on a domain  $V$ , the boundary integral equation is given by [3,4]

$$C_{ij}(P_0)u_j(P_0) = \int_S [U_{ij}(P, P_0)t_j(P) - T_{ij}(P, P_0)u_j(P)]dS(P),$$

$$\forall P_0 \in S, \quad (1)$$

where  $S$  is the boundary of the domain under consideration;  $u_i$  and  $t_i$  are the displacement and traction tensors on the boundary, respectively;  $U_{ij}(P, P_0)$  and  $T_{ij}(P, P_0)$  are the displacement and traction components of the fundamental solution for the elastostatic problem, respectively; and  $C_{ij}(P_0) = \frac{1}{2}\delta_{ij}$  for a smooth boundary with  $\delta_{ij}$  being the Kronecker  $\delta$ . This boundary integral equation relates the unknown boundary data (either traction or displacement) to the given boundary data. By solving Eq. (1), one can find the unknown boundary values of the traction or displacement. Then, by using a representation integral, similar to Eq. (1), but with the source point  $P_0$  inside domain  $V$  and  $C_{ij}(P_0) = \delta_{ij}$ , one can determine the displacement or stress at any point inside the domain  $V$ . In the BEM, elements are introduced on the boundary of

the domain, for example, surface elements for 3D problems and line elements for 2D problems, in order to numerically solve Eq. (1). Eq. (1) is subsequently reduced to algebraic equations involving the physical quantities (e.g. displacements and tractions) at a finite number of nodes on the boundary. By solving this system of algebraic equations, one obtains the boundary solutions (displacements and tractions), and then the solution at any interior point through the use of the representation integrals containing the boundary values. Some references on the BEM can be found in Refs. [3–7].

For the last thirty years or so, the BEM has been developed into a viable alternative or complement to the more general finite element method (FEM), due to its features of boundary discretization and high accuracy. The BEM has been applied to problems in solid mechanics, fracture mechanics, acoustics, elastodynamics, electromagnetics, heat transfer, fluid mechanics, contact mechanics, biomechanics, manufacturing process, and many other multi-physics phenomena. In recent years, the BEM has been found especially suitable for analyzing the mechanics problems of many advanced materials, such as composite materials [8,9], thin films or coatings [10,11], piezoelectric materials [12–16], and the micro-electro-mechanical systems (MEMS) [17,18]. The solution efficiency of the BEM has also been improved dramatically in the last few years by the so-called fast multi-pole expansion methods [19–22] and iterative solvers. In some studies [18], the BEM has surpassed the FEM regarding the solution efficiency in large-scale simulations, besides the efficiency of the BEM in the modeling (meshing) stage.

With the rapid development of the BEM in new application areas and with the new technologies demonstrating superior capabilities, it is highly valuable and beneficial to introduce the BEM to a broader engineering community and apply it to solve challenging research and real application problems. The Internet will be the best platform for disseminating the knowledge and software, and for promoting applications of the BEM in new areas.

## 3. An Internet-based computing platform—UCwaves

Foreseeing the forthcoming age of Internet computing, a prototype study on the concept of Internet computing was initiated by the second author with his students in 1998. The development of an online software, named UCWaves, was developed based on the Java technology (<http://java.sun.com/>) and using a 2D stress analysis BEM code as the first example. The developed GUI for UCWaves and the 2D BEM solver were successfully implemented on the Internet in 2000 (Fig. 1). Users can access this boundary element code through the Internet using a web browser. Once the UCWaves software is launched from this web site, users can open a model (such as the 2D gear model shown in the GUI in Fig. 1), modify geometry, apply boundary conditions, and

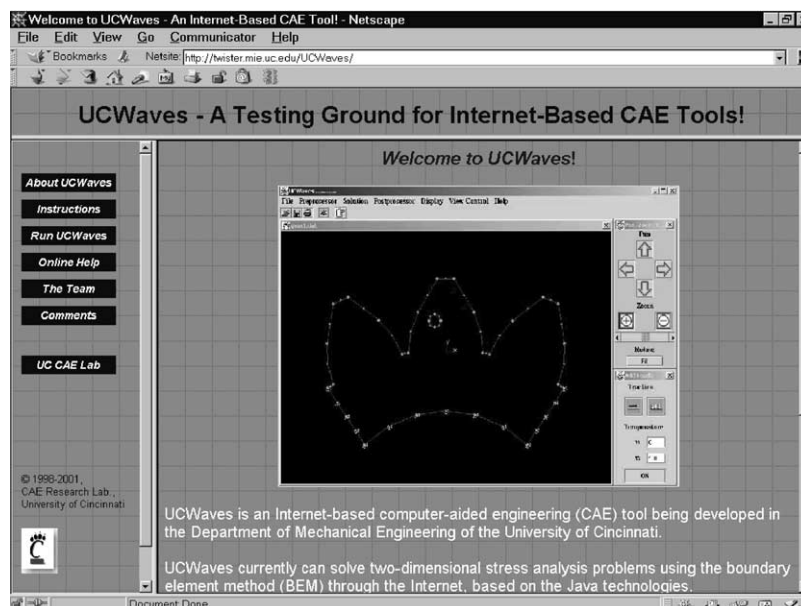


Fig. 1. Current UCWaves web-site (GUI shown in the main window will pop out on the user's computer when the **Run UCWaves** button is pressed).

solve the problem without ever installing the program on his or her own local computer. The web site for the UCWaves is at: <http://twister.mie.uc.edu/UCWaves/>.

This study, exploratory in nature, demonstrates that there are many advantages in Internet-based computing. For example, it will be much easier to upgrade the software and license the software (if necessary) by username and password controls; very convenient to collaborate with partners at other locations on a project; very cost-effective to increase the awareness and use of the BEM software; easier to achieve platform independence, and thin and faster clients. With this Internet-based approach, the developed BEM software can be accessed by any researchers or designers in industries through the Internet anywhere in the world. The usefulness of the BEM software based on this approach will be much broader than that of a traditional, standalone BEM software.

Implementing the Internet-based computing strategy is a great challenge, given the many uncertainties and difficulties, including the speed of current network, security issues, lack of standards, and the inadequacy of training to prepare users to use and explore this promising technology. On the other hand, due to the limitations of the current network speed, no large-scale commercial CAE software packages have been converted from stand-alone versions to online versions. The legacy CAE packages, developed in the last 30 years or so and based on the domain based approaches such as the FEM, have grown into huge programs with millions of lines of code and several hundreds of Mega bytes or even Giga bytes in sizes. The FEM model files can be equally huge for complicated mechanical parts or systems. It will take quite some time for these huge legacy CAE programs to be implemented on the Internet, if it is at all feasible in their current forms and technologies.

With the BEM, however, it can be much easier to assess the Internet computing concept. BEM code can be very small in sizes and input/output files are always considerably smaller in sizes due to the boundary-only discretizations, compared with the FEM. With the use of the Java technology in developing the GUI, as in the case of UCWaves, the size of the GUI can also be very small (e.g. the current size of the UCWaves GUI, shown in the main window in Fig. 1, is only a few hundreds of Kilo bytes). With smaller sizes of the GUI, solvers and data files, the BEM computing can be done relatively fast on the current network for relatively large models and on future network for even larger ones. Therefore, it is believed that the BEM offers a good and feasible choice for starting the Internet computing in CAE. To the best knowledge of the authors, no other BEM code has been implemented on the Internet so far, other than UCWaves.

#### 4. The design and implementation of UCWaves

Major components of UCWaves are: the GUI as a Java applet, BEM solver, UCWaves Web Server for hosting the website, and UCWaves Java Server (UJS) for hosting the BEM solver (Fig. 2). Each of these components will be discussed in the following, along with other related issues, such as security.

##### 4.1. Architecture of UCWaves

The present architecture of UCWaves (Fig. 2) is based on a two-tier client-server concept where the front end is the web browser, and the UCWaves Web Server and Java Server act as the backend. The GUI applet loaded through

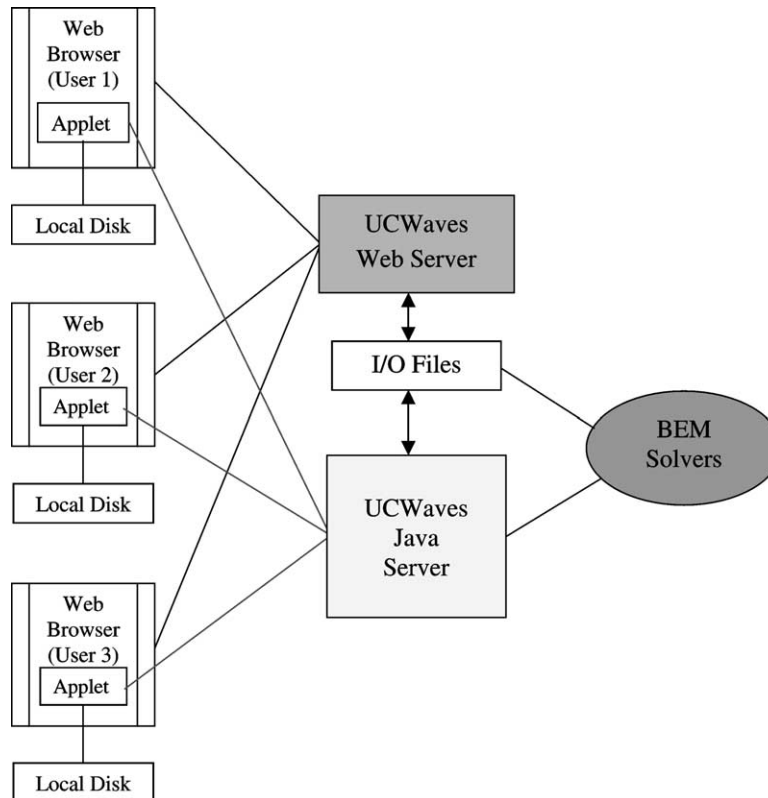


Fig. 2. Current two-tier architecture of UCWaves.

using the web browser is opened in a new frame. Model input files residing on client's local machine can be displayed in the UCWaves GUI window. Then the applet sends all the data related to the model file to the UJS. User can solve for displacements and stresses by clicking the *Solution/Start Solver* button in the GUI window. When the client gives commands like *Solve*, *Plot*, etc. these events start a connection with the UJS. The *Start Solver* command invokes the BEM solver on the server and a result file consisting of both nodal displacements and stresses is generated. The applet reads the result file using the web server. The nodal displacements can be displayed on the client screen by invoking the *Plot* command in the post-processing menu.

The main Java classes for the UCWaves GUI are shown in Fig. 3. The structure of the GUI is very similar to those of other CAE packages. However, the main advantage of the UCWaves GUI is the use of Java language (<http://java.sun.com/>), which presumably can provide operating-system-independent and smaller-size programs. Detailed descriptions of each of the four groups of classes shown in Fig. 3 can be found in Ref. [23], which is available at the UCWaves website.

#### 4.2. Implementation of the UCWaves web page

A key component in UCWaves is the web page for invoking the UCWaves program. This page is opened by

clicking on the *Run UCWaves* button on the UCWaves web-site (Fig. 1). This web page is an active web page in the sense that users are able to invoke the UCWaves GUI and BEM solver hosted on the server through this web page. The web page is constructed using embedded html. The embedded html consists of the simple html tags along with embedded objects. For example, the following lines show one of the object tags:

```

<OBJECT classid = "clsid:8AD9C840-044E-11D1-
B3E9-00805F499D93"
WIDTH = 500 HEIGHT = 500
codebase = "http://java.sun.com/products/plugin/1.2/
jinstall-12-win32.cab#Version = 1,2,0,0">
<PARAM NAME = CODE VALUE = "classes.
graphics.UCWaves">
<PARAM NAME = ARCHIVE VALUE = "ucwaves1
.jar">
  >
  
```

The particular tag here tells the browser to open the object denoted by the *classid*. In this example, the *classid* belongs to the Java Plug-in. The Java Plug-in *classid* can be obtained from the installation directory of the Java Plug-in. The browser calls the *Run UCWaves* web page from UCWaves Web Server. This web page calls Java Plug-in using embedded objects from local hard disk. Then the Plug-in calls the GUI applet, which is downloaded to

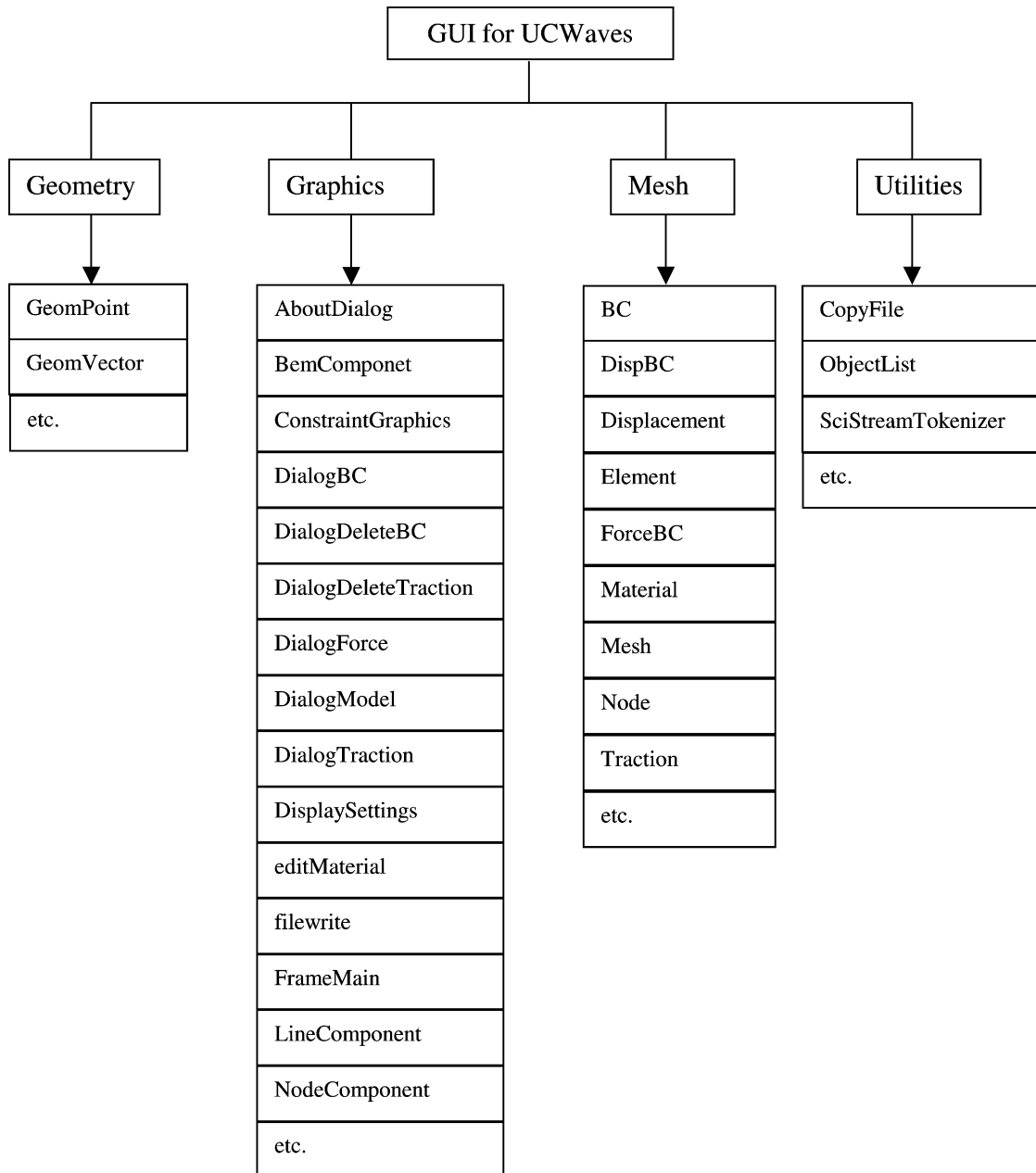


Fig. 3. Main classes for the UCWaves GUI.

the user's machine from the UCWaves Web Server and is connected to the UJS.

#### 4.3. Implementation of GUI Applet

There are unique and critical issues associated with online computing using the Java applet approach. For UCWaves, the GUI applet initiates a new frame, which is loaded through the browser window. The files are transferred to and from the server using socket streams. Following issues, mainly concerning the security of this approach, have to be addressed in the implementation of UCWaves GUI applet.

(a) *Security issues.* UCWaves is designed in such a way that a user will be able to upload a model file and view it in GUI. Due to security issues, applets are not allowed to access the local hard disk [24]. To circumvent this security restriction imposed by browsers as well as Java Virtual Machine (JVM), user needs to install a policy file [25] (as currently used for UCWaves), or to purchase a commercial security certificate service.

(b) *The Policy File.* One way the Java platform provides protection from attack of a virus is through the use of a security manager. Currently Java Development Kit (JDK) system code invokes security manager methods to perform resource access control checks.

Most browsers install a security manager, so that applets typically run under the scrutiny of a security manager. An applet is not allowed to access resources unless it is explicitly granted permission to do so by the security policy in effect. In Java platforms that are compatible with JDK 1.2 or later version, the permission must be granted by an entry in a policy file.

A policy file is an ASCII file and can be composed using any text editor. The following lines are the contents of the policy file currently used by UCWaves and users need to install this file on their local machines:

```
/* AUTOMATICALLY GENERATED ON Wed Apr 19
18:43:14 EDT 2000 */
/* DO NOT EDIT */
grant codeBase "http://twister.mie.uc.edu/" {
permission java.security.AllPermission;
};
```

This policy file states that all permissions to an applet coming from <http://twister.mie.uc.edu/> are granted. Instructions on how to install this policy file are given at the above UCWaves website.

(c) *Java Plug-in.* Java Plug-in software enables users to direct applets to run using Sun Microsystems' virtual machine instead of the web browser's default virtual machine. Virtual machine is the component needed to run Java class files. Instead of using browser's virtual machine, Java applet will use virtual machine residing inside the Plug-in. This will provide consistency and reliability while working with applets. This is important because different browsers can have different virtual machines installed, which can cause problems while executing Java applets.

For example, UCWaves GUI applet contains some Java swing components. Swing components are graphical components used in the earlier versions of JDK to create the GUI. However, the present versions of Netscape and Internet Explorer browsers do not support these components. Therefore specific Java Plug-in is needed in order to run the application in the browser. Instructions on how to install the required Plug-in, the Java runtime environment (JRE) needed for UCWaves, are provided on the UCWaves website.

#### 4.4. UCWaves java server

UJS is a multi-threaded Java program which acts like a server for the BEM solver. The server listens on a port for the applet clients. Whenever a new client accesses UCWaves on the web site, UJS creates a new thread. According to the commands sent by each client, the server invokes BEM solver or creates files for plotting. The solver is invoked by creating a new 'process'. The files are transferred to and from between the client machine and the server using socket streams. The UJS is installed on the server machine as a Windows 2000 service. This service is

always running on the server machine even if no user is logged on.

#### 4.5. Menu structure of the GUI

The main menu structure of UCWaves GUI is shown in Fig. 4. Options for creating a new file, opening an existing file, saving the current file are available in the *File* menu. The *Preprocessor* menu has options for manipulating the displayed model file such as specify material properties, add or delete displacement and traction boundary conditions, selecting nodes or elements. *Start Solver* command used to invoke the BEM solver is under the *Solution* menu. The *Postprocessor* menu has options for plotting and listing the results obtained. The *Display* menu is for selective display of nodes or elements of the model. The *View Control* menu is for pan-zoom-rotation of the model, for selecting view direction and for selecting view style. The *Help* menu has some example model files, along with links to online help materials to be added in the future.

#### 4.6. BEM solver

The current solver available in UCWaves is a 2D structural analysis BEM code written in C++ [10,11]. This code can solve multi-domain stress analysis problems, including structures or materials of extremely thin shapes. Quadratic line elements (with three nodes on each element) are used in this BEM code. Several example problems are available through the UCWaves *Help* menu. The format for the input and output files for this BEM solver can be viewed by listing these files within UCWaves. Other BEM or FEM codes can be added to UCWaves relatively easily in the future.

### 5. Demonstration cases

To demonstrate the feasibilities of the developed UCWaves software, three 2D stress analysis problems are presented. The model files for these three examples are also available on the UCWaves *Help* menu. The purpose of using these examples is to test the developed platform for applications of the BEM code through the Internet. They are not meant to show the already established capabilities of the BEM itself. For this latter purpose, readers can consult the work in Refs. [3–7] and numerous technical papers in the literature.

The first test case is the analysis of a square plate with a hole and a crack on the edge of the hole. The plate is under a tension in the *y*-direction. It is sufficient to analyze one quarter of the plate due to the symmetry in geometry and boundary conditions. A quarter-symmetry model with appropriate boundary conditions is shown in Fig. 5(a). These boundary conditions can be applied and viewed within UCWaves. After press the *Start Solver* button in

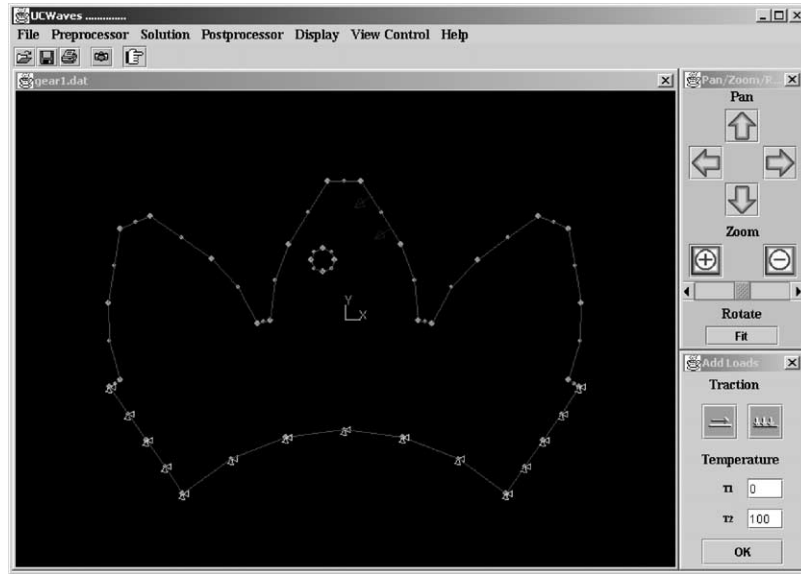


Fig. 4. UCWaves main menu structure.

the Solution menu, the model file is sent to the server and the BEM solver is invoked. The result file containing both the displacement and stress data is sent back to the user’s machine for post-processing, once the solution is done. The deformation of the plate considered is shown in Fig. 5(b). The stress contour plots are not available for the current version of UCWaves.

UCWaves is capable of solving displacements and stresses for multi-domain problems. An example of two-domain problems is shown in Fig. 6(a). This is a square region with a circular inclusion (such as a unit cell model of fiber-reinforced composites). A tensile load is applied to the model in the  $x$ -direction. Fig. 6(b) shows the deformed shape of the model obtained using UCWaves.

The third test case is an analysis of a 2D model of a gear with a cooling hole. The load on the tooth when it is in

contact with another gear is normal to the surface of tooth. This is shown in the displayed model in Fig. 7(a). Only a portion of the gear is considered for the analysis. Appropriate boundary conditions arising from taking only a portion of the gear for the analysis is also shown. Deformed shape for the portion of the gear taken for the analysis is shown in Fig. 7(b).

All the above and other examples can be accessed and tested in UCWaves in the Help menu (by clicking Help/Examples in the UCWaves GUI). All these test problems solved by using the UCWaves demonstrate that the Internet-based computing platform is feasible and very promising, although the speed of the Internet is still the bottleneck. With further development of the GUI and various BEM/FEM solvers for online applications, it can become a powerful analysis tool for engineering analysis in the future.

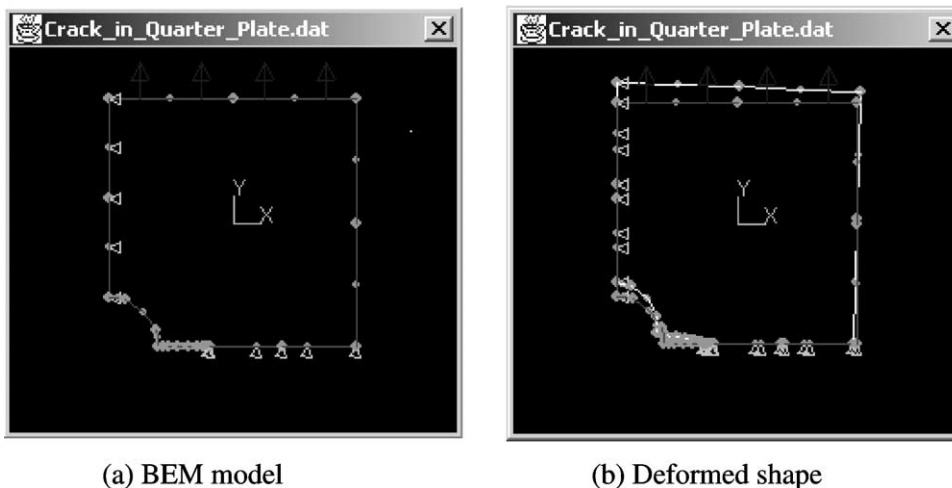


Fig. 5. Square plate with a hole and edge crack.

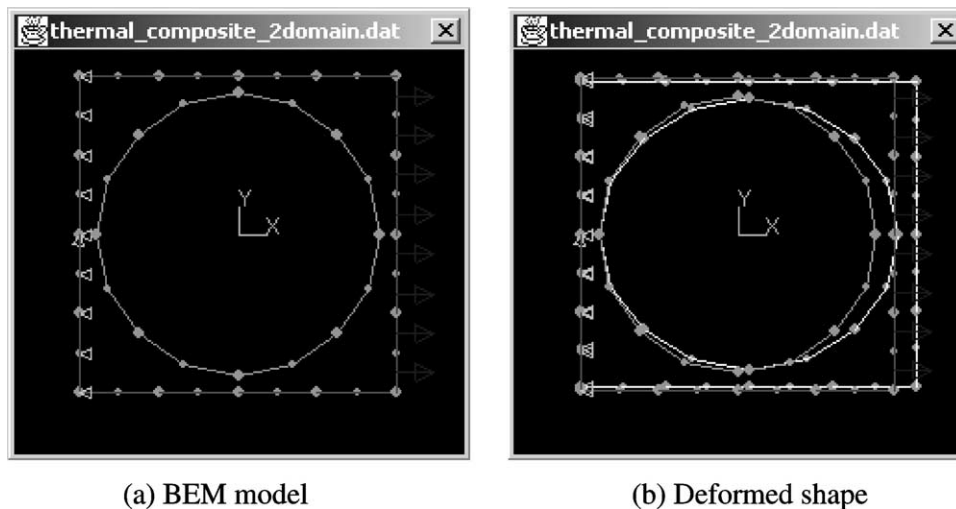


Fig. 6. Two-domain model (An inclusion).

## 6. Discussions

An online computing platform for the BEM is presented in this paper, which is a first step toward fully Internet-based computing for the BEM. The developed online UCWaves software is still in its early stage and much work needs to be done before it can be applied to solve any large scale practical problems, beyond the ones shown in the previous section for the research and education purposes. The main goal of this paper is to demonstrate that Internet-based approach for the BEM is feasible and promising, and to discuss the advantages of using the Internet as future computing platform for engineering software applications. To realize the potentials of Internet computing, many critical and difficult issues must be addressed.

First, the speed of the Internet is still a problem. Although there are many internal network with transfer speed at the level of Giga bytes per second, the speed for an application through the Internet is always determined

by the weakest link, which may have the speed of a modem. Speed of the server is also critical for solving any real application problems, especially when there are many users using the software concurrently. One solution to this problem, besides a more powerful server, is to run the solver locally on the users' computer. This may complicate the management of the solver code, but it can be at least an option. The security is another serious problem for any Internet-based software. Unless there is absolute certainty that the site hosting the applications is safe from conducting virus or other malicious attacks, users will be skeptical about using the online software. Using a commercial security certificate service may be a better choice than employing the policy file as used in this work. Standards in Internet computing and measurements of its effectiveness are all urgently needed for its further development.

Currently, UCWaves can only handle 2D models in stress analysis. Features dealing with 3D models are being

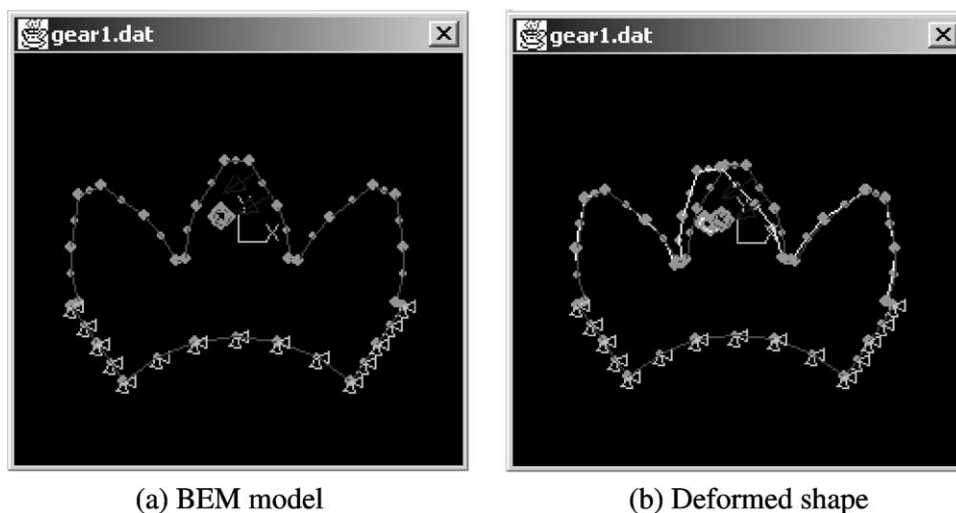


Fig. 7. Gear teeth with a hole.



developed, including display of models, applications of boundary conditions, solution of the BEM model, and post-processing of the results for 3D models. Geometric modeling and meshing capabilities are also being developed. More detailed and systematic online documentations will be developed, which should be an integral part of any Internet-based computing platform. Other boundary element and finite element solvers can also be added to this Internet platform in the future.

### Acknowledgements

Many former and current graduate students in the Department of Mechanical Engineering at the University of Cincinnati contributed to the development of the concept of Internet computing and the UCWaves research in one way or another. They are S.R. Karmalkar, J.F. Luo, N. Xu, K. Puppala, S. Kolluri, G. Deshmukh, B.B. Velkur, S. Annamdevula and X.L. Chen. The authors sincerely acknowledge their contributions.

### References

- [1] Jaswon MA. Integral equation methods in potential theory. I. *Proc R Soc Lond A* 1963;275:23–32.
- [2] Rizzo FJ. An integral equation approach to boundary value problems of classical elastostatics. *Q Appl Math* 1967;25:83–95.
- [3] Banerjee PK. *The boundary element methods in engineering*, 2nd ed. New York: McGraw-Hill; 1994.
- [4] Brebbia CA, Dominguez J. *Boundary elements—an introductory course*. New York: McGraw-Hill; 1989.
- [5] Cruse TA. *Boundary element analysis in computational fracture mechanics*. Dordrecht, The Netherlands: Kluwer; 1988.
- [6] Mukherjee S. *Boundary element methods in creep and fracture*. New York: Applied Science Publishers; 1982.
- [7] Kane JH. *Boundary element analysis in engineering continuum mechanics*. Englewood Cliffs, NJ: Prentice Hall; 1994.
- [8] Liu YJ, Xu N, Luo JF. Modeling of interphases in fiber-reinforced composites under transverse loading using the boundary element method. *J Appl Mech* 2000;67:41–9.
- [9] Chen XL, Liu YJ. Multiple-cell modeling of fiber-reinforced composites with the presence of interphases using the boundary element method. *Comput Mater Sci* 2001;21:86–94.
- [10] Luo JF, Liu YJ, Berger EJ. Analysis of two-dimensional thin structures (from micro- to nano-scales) using the boundary element method. *Comput Mech* 1998;22:404–12.
- [11] Luo JF, Liu YJ, Berger EJ. Interfacial stress analysis for multi-coating systems using an advanced boundary element method. *Comput Mech* 2000;24:448–55.
- [12] Chen T, Lin FZ. Boundary integral formulations for three-dimensional anisotropic piezoelectric solids. *Comput Mech* 1995;15:485–96.
- [13] Ding H, Liang J. The fundamental solutions for transversely isotropic piezoelectricity and boundary element method. *Comput Struct* 1999;71:447–55.
- [14] Pan E. A BEM analysis of fracture mechanics in 2D anisotropic piezoelectric solids. *Engng Anal Bound Elem* 1999;23:67–76.
- [15] Liu YJ, Fan H. On the conventional boundary integral equation formulation for piezoelectric solids with defects or of thin shapes. *Engng Anal Bound Elem* 2001;25:77–91.
- [16] Liu YJ, Fan H. Analysis of thin piezoelectric solids by the boundary element method. *Comput Meth Appl Mech Engng* 2002;191:2297–315.
- [17] Ye W, Mukherjee S. Design and fabrication of an electrostatic variable gap comb drive in micro-electro-mechanical systems. *Comput Model Engng Sci* 2000;1:111–20.
- [18] Ljung P, Bachtold M, Spasojevic M. Analysis of realistic large MEMS devices. *Comput Model Engng Sci* 2000;1:21–30.
- [19] Gomez JE, Power H. A multipole direct and indirect BEM for 2D cavity flow at low Reynolds number. *Engng Anal Bound Elem* 1997;19:17–31.
- [20] Fu Y, Klimkowski KJ, Rodin GJ, Berger E, Browne JC, Singer JK, Geijn RAVD, Vemaganti KS. A fast solution method for three-dimensional many-particle problems of linear elasticity. *Int J Numer Meth Engng* 1998;42:1215–29.
- [21] Nishimura N, Yoshida K-i, Kobayashi S. A fast multipole boundary integral equation method for crack problems in 3D. *Engng Anal Bound Elem* 1999;23:97–105.
- [22] Mammoli AA, Ingber MS. Stokes flow around cylinders in a bounded two-dimensional domain using multipole-accelerated boundary element methods. *Int J Numer Meth Engng* 1999;44:897–917.
- [23] Nugehally M. *Java 3D for UCWaves*. MS Thesis. Department of Mechanical Engineering, University of Cincinnati; 2002.
- [24] Rubin A, Geer D, Ranum M. *A complete guide to web security*. New York: Wiley; 1999.
- [25] McGraw G, Felten E. *Java security hostile applets, holes, and antidotes*. New York: Wiley; 1999.