CFDNET − RECENT DEVELOPMENTS AND FUTURE WORK

J. Militzer and Theo A. Bell
Department of Mechanical Engineering
Dalhousie University
Halifax, P.O.Box 1000, Nova Scotia, B3J 2X4, Canada
http://cfdnet.com, e−mail: julio.militzer@dal.ca

ABSTRACT

CFDnet stands for computational fluid dynamics (**CFD**) software accessible over the Inter**net**. Users dynamically control the server's meshing and solving routines from a GUI (Graphical User Interface) written in Java, running in their web browser. Users can set up, solve, and visualize their own fluid flow problems. By using the platform−independent Java language to provide the user interface, student access to CFDnet is made independent of the type of client computer used. Additionally, by performing the computationally intensive meshing, solving, and visualization processes on a server−side network of powerful computers, this flow modeling tool is made available to users with relatively low speed, inexpensive computers. Classroom experience shows the usefulness in teaching sometimes very difficult Fluid Dynamics concepts. Furthermore, the availability of CFDnet greatly reduces the need for expensive and cumbersome laboratory experiments and demonstrations. CFDnet has been integrated into Fluid Dynamics courses at several universities. This contribution presents some of the main characteristics of CFDnet, some examples of class room application, as well as results student evaluation of CFDnet. Some future projects are also presented, including a joint project proposal to develop a Numerical Wind/Water Tunnel on a national GRID of supercomputers.

INTRODUCTION

CFD can be a very powerful tool in teaching Fluid Dynamics. Complicated flow features can be visualized with relative ease, without the need of complicated and time consuming experiments. Furthermore, many graduating students are expected to carry out CFD simulations in their first job. Most commercial CFD programs are designed with the industrial user in mind and as such are too complicated for the use in a general Fluid Dynamics course. Furthermore, they are usually expensive to purchase, and require maintenance and frequent and costly upgrades. The advent of the Internet followed by Java Applets [1,2] offered a unique opportunity to offer access to relatively complex software to any user with a Java enabled web browser. The main objectives of CFDnet development were:

- Create free software accessible to the largest possible number of users.
- Develop a simple, modular user interface that minimizes both the learning time and the specialized knowledge required.

The Java programming language offers the additional attraction of platform independence, requiring the development of a single code for many different types of computers. Low Internet bandwidth, however, currently makes it impractical to develop large software packages such as an entire CFD software package as an embedded Java Applet. Even when Internet bandwidth and the resulting transfer speeds become significantly faster, other issues make it undesirable to download and run the software entirely on the user's client computer. These issues include our desire to make the computationally intensive processes efficient for all users, even those with less powerful computers, and our desire to protect the developed software package from piracy.

CFDnet uses a Java Applet running on the client computer for problem set up (geometry, physical properties, boundary conditions and solution parameters). The meshing and solver routines reside on the server and are invoked by the client from the Java based GUI. To handle a large number of simultaneous users the server distributes the jobs to several computers using PVM (Parallel Virtual Machine).

The present version of CFDnet is limited to simulating two−dimensional incompressible, laminar or turbulent, steady or unsteady flows and is available on the Internet at http://cfdnet.com.

THE JAVA APPLET−BASED USER INTERFACE

When a student accesses the CFDnet web site (http://cfdnet.com) with a Java−enabled web browser (e.g. Netscape Navigator or MS Internet Explorer), the Java Applet−based graphical user interface (GUI) is automatically loaded, along with the text and images of the web page. In its present form, the user interface consists of several class files and associated images that total about 80 Kbytes, making the download time about 20 seconds for modem connections, and virtually unnoticeable for the higher speed connections normally available at universities. Once started by the

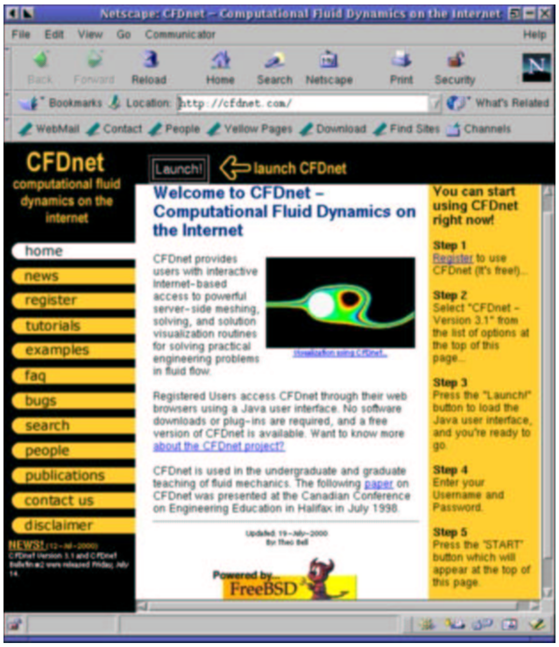

Figure 1 − CFDnet opening page (http://cfdnet.com).

student, the GUI has the look and feel of a native application, and can be used to input the geometry, select the type of problem, specify the boundary conditions and solution parameters. The GUI is also used to initiate and communicate with server−based processes including mesh generation, solution of the flow field, and the generation of visualizations. Once a problem is solved, the same GUI facilitates inspection of the solution. Figure 1 shows the opening page of CFDnet's Java Applet−based GUI, while Figure 2 shows the GUI under step 2 of the Wizard (used to guide the user through a step by step sequence of problem set up). For example, in step 2 the user can choose between several different geometries and even change their proportions. If however he prefers he can draw his own geometry, using the drawing tools available on the GUI. CFDnet uses body fitted coordinates developed according to [3]. Figure 3 illustrates the meshing step, which is done automatically with the aid of the Wizard. Cell size is suggested by the wizard, but can, within limits, be changed by the client.

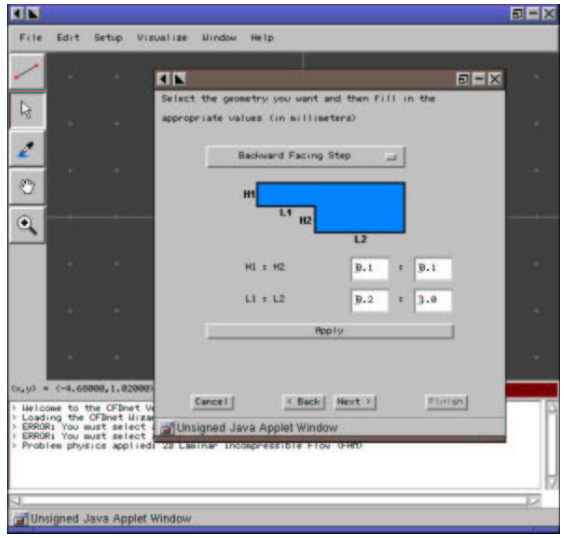

Figure 2 − GUI and Wizard.

Requiring the user to download the GUI from our server means that he is always accessing the most recent, up to date version − i.e. "upgrades" are effectively transparent. In addition, no software installation by the user is required Minimizing perceived response time by the user was the overriding factor in determining whether a particular process should be handled locally at the client, or remotely at the server. When a particular action requires a response time of less than several seconds (e.g. selecting a boundary, applying a boundary condition), all associated coding was provided with the GUI. For actions where a longer response time can be tolerated or is expected (e.g. generating the mesh, or solving the

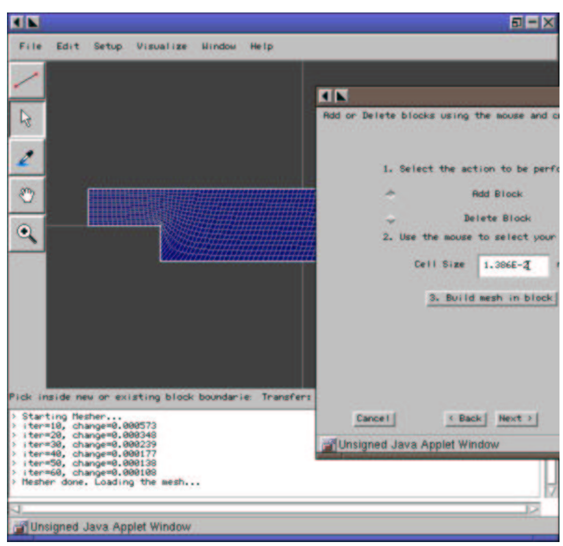

Figure 3 − Meshing.

flow problem) the process is carried out on the server. The GUI code only initiates, interprets and displays the results. This philosophy tends to minimize the size of the GUI, but can result in a single user making many simultaneous attempts to establish server connections. To overcome this potential bottleneck, a special client class was developed as part of the GUI that established and maintained a single connection to the server to support all interactive requests. To free up server resources, this client class was designed to disconnect after 15 seconds of inactivity.

All Java GUI development, debugging, and compilation was performed using Sun Microsystem's Java Development Kit [freely available at http://developer.javasoft.com]. During the initial development of CFDnet, an effort was made to avoid writing a specialized server daemon, and use a standard http daemon in conjunction with server−side CGI programs to facilitate client−server communication. This arrangement did not provide truly interactive, two−way communication between client and server processes, and it was necessary to develop a specialized server daemon, which we call "artifacto". Written in ANSI C, and running on a UNIX server, artifacto directs the connection requests from the client GUI's on to the appropriate server based programs. Consequently, customized binary streams of data can be exchanged between client and server, permitting interactive client control of potentially very powerful programs.

SERVER BASED PROCESSES

To allow the CFDnet server to make use of all computers on the local network as potential solvers, the meshing, solving, and visualization routines incorporate the publicly available parallel virtual machine (PVM) libraries and software [freely available at ftp://netlib2.cs.utk.edu/pvm3]. Initially developed by Los Alamos Laboratories, PVM provides a standard message passing interface, and allows the client's requested process to be initiated and run on any one of the network computers. Through this parallelization strategy, the requirements of a large number of users can be served simultaneously, and the server itself does not need to run any computationally intensive processes, thus improving response time. In its present form, individual CFDnet processes themselves are not parallelized.

Although it is not the purpose of this paper to present the details of the flow solution algorithms used by CFDnet, the following information and references are provided for the interested reader. CFDnet uses multi−block structured meshes, generated using elliptic mesh generation with control functions selected to produce mesh orthogonality at the block boundaries [3]. The governing equations of fluid flow are discretized on the body−fitted grid either using the Finite Analytic Method (FAM) [4,5and 6], or the power law [7]. The FAM being less subject to numerical diffusion provides a better solution, however it requires greater CPU time and is sometimes unstable (for large cell Reynolds numbers). Thus, the client can choose which solver is more suited to his problem. The primitive flow variables − u, v, and p − are solved using an iterative pressure correction scheme similar to SIMPLER [7]. These choices for mesh generation and flow solution algorithms were made because suitable meshing and solving routines already existed within our research group. The Internet−based philosophy of CFDnet could, however, easily be extended to other solution methods (e.g. finite volume or finite element methods, unstructured meshes, etc.).

CFDNET IN THE CLASS ROOM

To make CFDnet a powerful educational tool for engineering students, the program described thus far is complemented by an expanding set of web−based help pages and tutorials. These pages are designed to be referenced by the students during their CFDnet session, and teach them how to use CFDnet through the setup and solution of some simple but illustrative Fluid Dynamics problems. Students have also access to some general CFD pages, that are use to give a brief introduction into the subject. Another tool made available for the students is a problem set up wizard. It automatically pops up when a GUI is loaded and guides the user through the necessary steps for the problem set up.

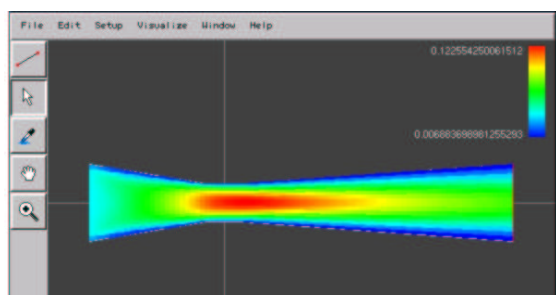

Figure 4 − Laminar and turbulent convergent divergent channel flow.

CFDnet has been used by colleagues for their Fluid Dynamics courses at UVic, University of Connecticut and Dalhousie University. The following is the description of a typical two hour session held last Fall at Dalhousie University. During the allotted time students were expected to analyze the three flows shown in Figures 4, 5, and 6.

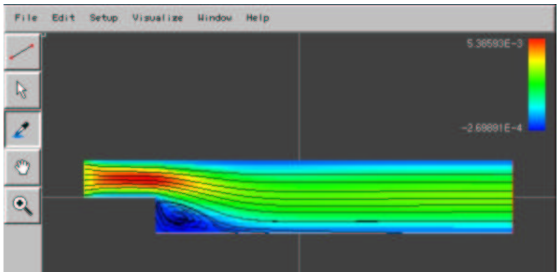

Figure 5. Laminar backward facing step flow

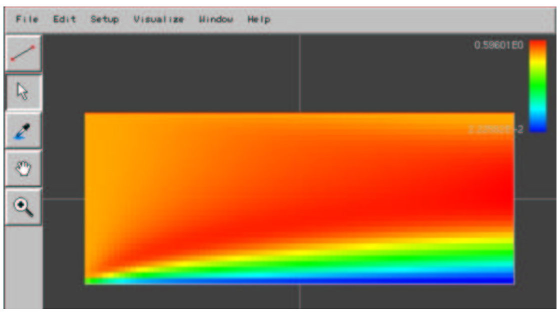

Figure 6. Laminar boundary flat plate flow.

Before this class students have already been given a brief introduction into CFD through an assignment where they use MS Excel to solve numerically with finite differences the parabolic Blasius laminar boundary layer equations. In the first hour CFDnet session students are given a 30 minutes introduction into CFDnet's main features. Each student is seated in front of a PC's with an internet connection to http://cfdnet.com. On the web they can find each of the tutorials, which guide them through the problem set up and solution. At the end of each tutorial they are given a series of assignment type questions, which require the students to carry out further runs. For example for the convergent divergent channel they have to run a laminar and a turbulent case and compare the velocity distributions at the throat. For the backward facing step flow they are required to run several cases for different Reynolds numbers and compare the reattachment distance with available results (provided). Finally, for the flat plate case they are required to plot the velocity distributions at two locations along the plate and compare them with the Blasius solutions, which they had obtained in the previous assignment described above.

STUDENT EVALUATIONS

Currently there are almost 1,200 registered users of CFDnet. Table 1 provides their composition. As the table shows around 70% of users can be identified as academics (assuming that half of the unknown users are academic).

Table 1. CFDnet user distribution

	# Users	%
Academic	685	59
Commercial	243	21
Unknown	238	20
Total	1166	100

To understand better how students perceive CFDnet, we usually carry out a survey at the end of the second class. It is interesting to see their opinion even after such a short period of exposure. The most recent evaluation was carried out last fall with a class of approximately 60 undergraduate Fluid Dynamics students. Table 2 summarizes their responses.

Table 2 − Summary of student responses to CFDnet survey.

	Ease of Use	Simulation	Flexibility
Very Good	7%	24%	10%
Good	76%	76%	79%
Poor	14%	0%	7%
Unacceptable	3%	0%	3%

Some students complained about slow response time (bandwidth problem) as well as having to "wait too long" for an answer. The first problem is gradually being resolved with the advent of ever higher speed internet connections. Part of the second problem is a result of lack of exposure of the students to iterative large processes of the kind required to solve CFD problems. Nevertheless, we are already addressing this problem by developing a national GRID of supercomputers which will greatly improve computation speeds. (see item on the subject under "Canadian C3 GRID").

In general, however, student reaction was very positive. CFDnet provided students with a sense of empowerment that allowed them to use their Fluid Dynamics knowledge to solve a flow problem and immediately visualize the solution. Student comments also helped to identify many improvements that will be incorporated in future versions of the program, such as easier method of exporting results into a spreadsheet or plotting routine.

WHAT IS NEXT ?

We are currently making final adjustments to the upcoming release of CFDnet 4. It will incorporate several new features, among those the most important is the introduction of multi block grids. It will enable users to solve steady and unsteady external flows, such as the flow around an airfoil or a cylinder. Figure 7 shows an example of one of the preliminary results for the flow over a round cylinder.

In addition we will incorporate the ability of creating animated gifs to better present unsteady solutions.

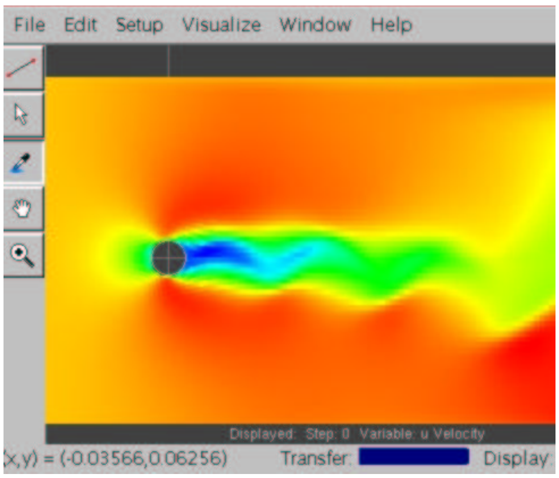

Figure 7. Oscillating flow around a cylinder.

CANADIAN C3 GRID

At the same time we are developing a much more ambitious project. This would is a joint effort involving C3.ca, NRC, CERCA (Montreal), UVic, UNB and Dalhousie University. The goal of this project is to implement a GRID system for use by Canadian research and development workers. Figure 8 shows a schematic of the proposed network for the C3 GRID implementation. The GRID system is designed to enable the sharing of advanced computing and telecommunications resources, thereby making it possible to conduct research and development activities that would ordinarily be limited by the scale of locally available data processing facilities.

The proposed Canadian C3 GRID is a form of a virtual private network implemented using a software suite, termed “middleware”, that integrates the functions of network access, routing, process submission and scheduling, computational resource allocation and control, user interface facilities, resource accounting and security. Using appropriate middleware, a single process stream can be constructed that can effectively use the advanced communications facilities and bandwidth of a network like CA*net3 to distribute processing requirements to one or more geographically dispersed data processing sites. The synergy between the high bandwidth backbone network, the middleware, and the high capacity shared computational resources delivers to the user the capability to complete large computational tasks and to view output using advanced visualization techniques that would not be feasible in a non−GRID environment.

The project has an initial duration of one year . The Dalhousie portion of the project will be the development of a 3D CFD solver to simulate a Numerical Wind/Water Tunnel. The Computational Fluid Dynamics (CFD) solver will carry out time−dependent Navier−Stokes simulation on a parallel distributed memory cluster of super computers. The solver will integrate the existing technologies of CFDnet (Dalhousie University), Parallel Tool Kit (PTK, University of Waterloo), the Message Passing Interface (MPI), and potentially Parallel Virtual Machine (PVM) to handle the Direct Numerical Simulation of incompressible flows using the Immersed Boundary Method for handling complex geometries and moving boundaries. Extension to flows at all speeds (i.e. compressible flows) and incorporation of sub−grid scale models for Large Eddy Simulation (LES) or turbulence models for Unsteady Reynolds−Averaged Navier Stokes (URANS) simulation will be considered in the development process, but will likely be too ambitious for the March 31, 2002 date. The solver will be integrated with the VU flow visualization environment of Centre de recherché en calcul applique (CERCA, Montreal) for problem setup and manipulation of 3−dimensional geometry in the numerical wind tunnel, computational steering, and flow visualization and analysis. To test the final product colleagues from different parts of the country will carry out simulations of complex flows, such as: High Reynolds viscous flow simulation for submarine geometry; transition to turbulence in the flow along blades of compressors and low−pressure turbines; hydrodynamics of a model swimming fish and the simulations of the flow over the wing/body of a jet airplane.

CONCLUSIONS

CFDnet has demonstrated the possibilities of integrating Java with server−based processing to significantly boost the interactive power of Internet for engineering students. After a short introductory session, undergraduate engineering students were able to solve relevant and interesting Fluid Dynamics problems. Our experience has clearly demonstrated the value of introducing CFD in the undergraduate teaching of Fluid Dynamics. Not only for the purpose of teaching the principles of CFD, but also more

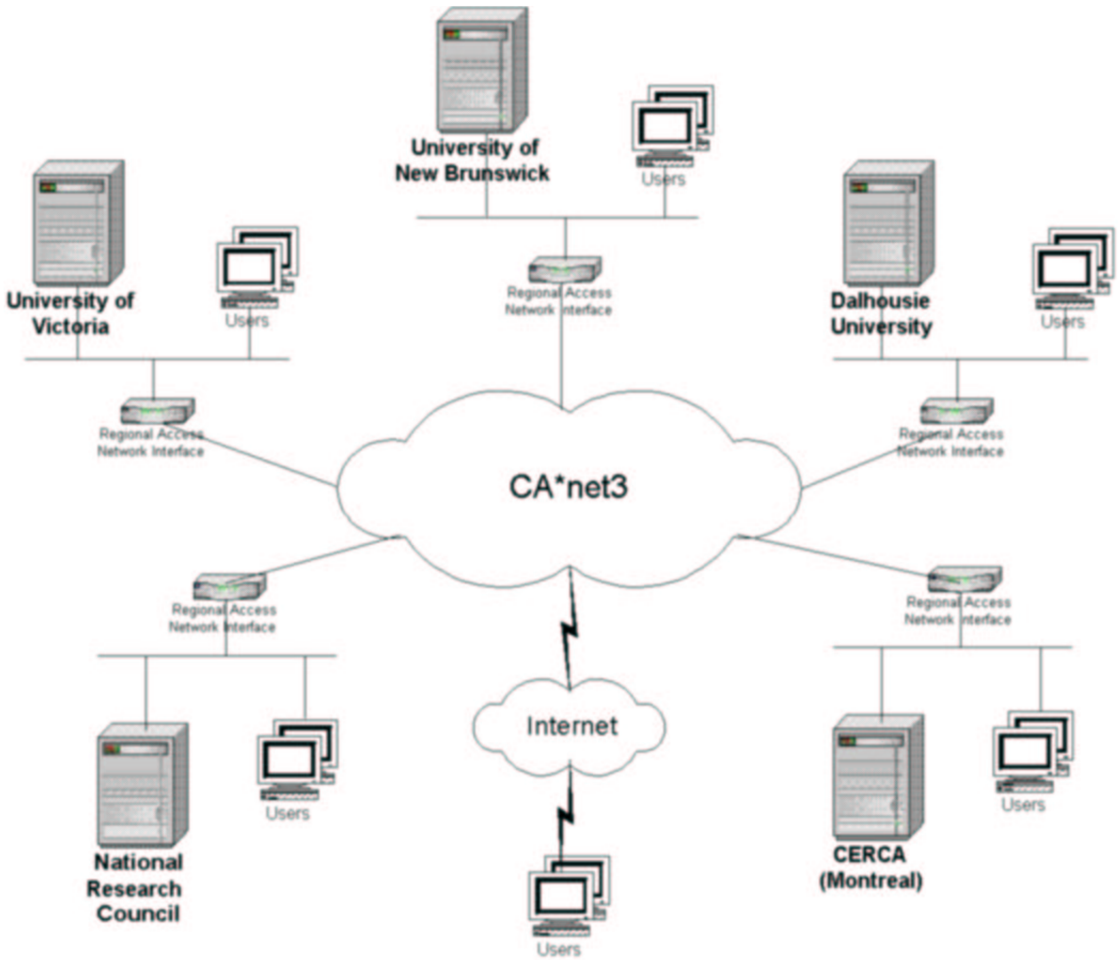

Figure 8− Schematic of proposed network for the C3 GRID implementation.

importantly as a tool to teach the principles of Fluid Dynamics and even as a tool to reduce the need for actual hands on laboratory experimentation, which can be in many cases be simulated more effectively with CFDnet.

CFDnet is continually being improved. Currently there almost 1200 registered users from all corners of the world. The objective is to make CFDnet into an effective teaching aid for Fluid Dynamics. With this in mind we are continually introducing new tutorials as well as improving the solver.

The C3.ca GRID project will offer a complete 3D CFD program, available to the whole research and educational community and accessible over the Internet.

REFERENCES

1. Dietz, D. (1996) Java: A New Tool For Engineering. Mechanical Engineering, Vol. 118 No. 4 April 1996, pp 68−72.
2. Deitz, D. (1997) Product Development On The Web. Mechanical Engineering, Vol. 119 No. 1 January 1997, pp 66−69.
3. Thompson, J.F., Warsi, Z.U.A., and Mastin, C.W. (1985) Numerical Grid Generation, Foundations and Applications. North Holland, Amsterdam.
4. Sun, Y. and Militzer, J. (1992a) The Piece−Wise Parabolic Finite Analytic Method − Part 1: Theory. Applied Mathematical Modeling, Vol 16, pp. 576−588, Oct. 1992.
5. Sun, Y. and Militzer, J. (1992b) The Piece−Wise Parabolic Finite Analytic Method − Part 2: Application. Applied Mathematical Modeling, Vol. 16, pp. 125−132, February 1993.
6. Ham, F. and Militzer, J. (1997) Development of the Finite Analytic Method for Body Fitted Multi−Zoned Structured Meshes. Proceeding of CFD97, The Fifth Annual Conference of the Computational Fluid Dynamics Society of Canada, May 25−27, 1997, Victoria, Canada.
7. Patankar, S. V. (1980) Numerical Heat Transfer and Fluid Flow. Hemisphere Publishing Corporation.