SOME RECENT ADVANCES FOR PRACTICAL FINITE ELEMENT ANALYSIS

K. J. Bathe,† J. Walczak‡ and H. Zhang‡

†Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.
‡ADINA R&D, Inc., 71 Elton Avenue, Watertown, MA 02172, U.S.A.

Abstract—This paper illustrates some recent advances in finite element procedures that are important for practical engineering analysis. Capabilities to solve large finite element systems using iterative methods, and some significant advances in finite element procedures for the nonlinear analysis of structures and fluids are demonstrated. The emphasis is on presenting solution results that demonstrate the applicability of finite element procedures rather than on theoretical developments (which are available in references).

1. INTRODUCTION

The use of finite element methods in engineering practice is increasing rapidly. This trend is observed because the integrated use of finite element software in CAD can have a significant benefit on the productivity of a design team, increase the functionality of the design and decrease its cost.

A great impetus for the use of finite element methods has been without doubt the dramatic decrease of hardware costs to run finite element codes. In addition, however, finite element software capabilities have significantly increased over the last decade. Of course, one important feature is that finite element programs are now much more user-friendly and have, to some degree, also been coupled to CAD software. However, another feature is that great strides have been accomplished in improving and enlarging the solution capabilities of finite element programs. Today, problems can be solved—without much difficulty—of a complexity that a decade ago, an engineering analyst would probably not have dared to tackle.

The objective in this paper is to briefly focus on some significant advances in problem solving solution capabilities. We do not present the associated theoretical developments, and do not give any details of the algorithms employed (which can be found elsewhere, see [1, 2]). The only purpose of this paper is to show models and results that demonstrate powerful solution capabilities and hopefully attract the reader to develop interest in finite element methods.

The presentation given below is concerned with the solution of structural and fluid flow problems. We used the ADINA system, version 6.1, for the analysis of the problems. The models have in most cases been furnished to us by companies in the U.S., Europe and Japan. Due to proprietary considerations, the company names and details of the models are not given. However, sufficient detail is given to

Original figures for this paper were generated with a color-producing terminal and submitted in color.

demonstrate the use of finite element methods and learn about the power of the techniques in engineering design.

2. THE USE OF ITERATIVE SOLVERS IN STRUCTURAL ANALYSIS

A most significant development regarding structural finite element software is the re-emerging use of iterative solution techniques. Of course, iterative methods have been in use all along successfully with other numerical methods, and were also employed in the very early finite element computer programs. However, it is only now that iterative methods are being incorporated in widely used commercial structural finite element software. The primary reasons for the re-emerging use of iterative methods in finite element structural analysis are that increasingly larger finite element systems are being considered, for which direct solution techniques are expensive (both in computer time used and required disk storage), and that effective pre-conditioners have been identified.

The basic equations that need to be solved in finite element analysis are

$$Ax = b, \tag{1}$$

where A is the finite element (effective) coefficient matrix, b is the (effective) load vector and x is the solution vector. In linear analysis $A = K$, $x = U$, $b = R$, whereas in nonlinear analysis a tangent stiffness matrix and incremental displacements and loads are used. In dynamic analysis (with implicit time integration), also the effect of the inertia forces enters A and b [2]. In all cases, however, as the size of the finite element system increases, the number of zero entries in the matrix A under the skyline becomes very large.

Until very recently, the solution of eqn (1) was obtained in all commercial finite element codes using a form of direct Gauss elimination. These direct procedures have been refined to excellent performance. However, the inherent disadvantage of these
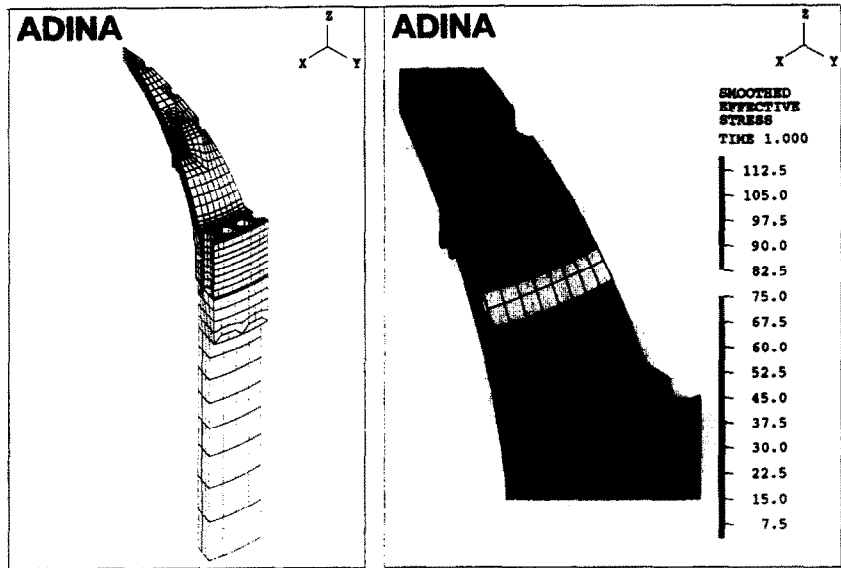
Fig. 1. Model of cover of a reactor vessel.

solution methods is that the zero entries under the skyline in the matrix **A** need be processed because almost all these entries become nonzero during the Gauss elimination. Since the required storage is (roughly) proportional to $N \cdot m$, and the number of numerical operations are (roughly) proportional to $\frac{1}{2}Nm^2$, where N = number of degrees of freedom, m = half bandwidth, and (roughly) $m \sim \alpha \sqrt{N}$, where α = a factor, the expense of storing and solving the system in eqn (1) increases rapidly as N increases.

Using an iterative method, on the other hand, only the actually nonzero entries under the skyline need be stored (with suitable pointer arrays that tell the location of these entries), and operations are only performed on these originally nonzero entries. For large finite element systems, the percentage of originally nonzero entries of all entries below the skyline is typically less than 5% (because the 'nonzero entry half bandwidth' does not depend on the total number of degrees of freedom but only on the number of finite elements coupling into a node). Therefore, for large systems, the required storage will always be less in the iterative solution, but whether the required solution time is less depends of course on the specific iterative scheme used and on the conditioning of the coefficient matrix. The conditioning is in static analysis generally good when three-dimensional (3-D) continua are analyzed, and not favorable when shell-type structures are considered. In dynamic analysis, the conditioning usually improves very significantly due to the mass matrix effects in the matrix **A** (representing physically the inertia effects) [2].

While there are frequently the above significant advantages in using an iterative method, the disadvantages are that, firstly, it is not possible to predict

in a simple way how many iterations will be needed for convergence, and, secondly, if multiple load cases are considered in linear static analysis, the iterative scheme requires (except for the calculation of the pre-conditioner) for each load case a completely new solution (whereas in a direct solution, the factorization of **A** is only performed once, and each load case requires only a forward reduction and back-substitution).

In practice, an iterative scheme is probably best available in a finite element program as an alternative solution strategy to direct solution. The analyst will then obtain sufficient experience to know reasonably well in advance whether the iterative method will be effective for his/her types of analyses, and only then use the iterative scheme. The solutions discussed below should add to the experience base with the iterative methods used in ADINA.

We recall that in ADINA the conjugate gradient method is used with an incomplete Cholesky pre-conditioner [3]. The iterative solver can be employed in static and dynamic analyses with all other modeling options available in the program.

Table 1. Analysis of cover of a reactor pressure vessel

Number of equations: 58,707
Elements used: 3-D SOLID, 20-node elements
Type of analysis: linear static
Number of iterations (ITESOL): 496
Computer: IBM RISC 6000/550 workstation

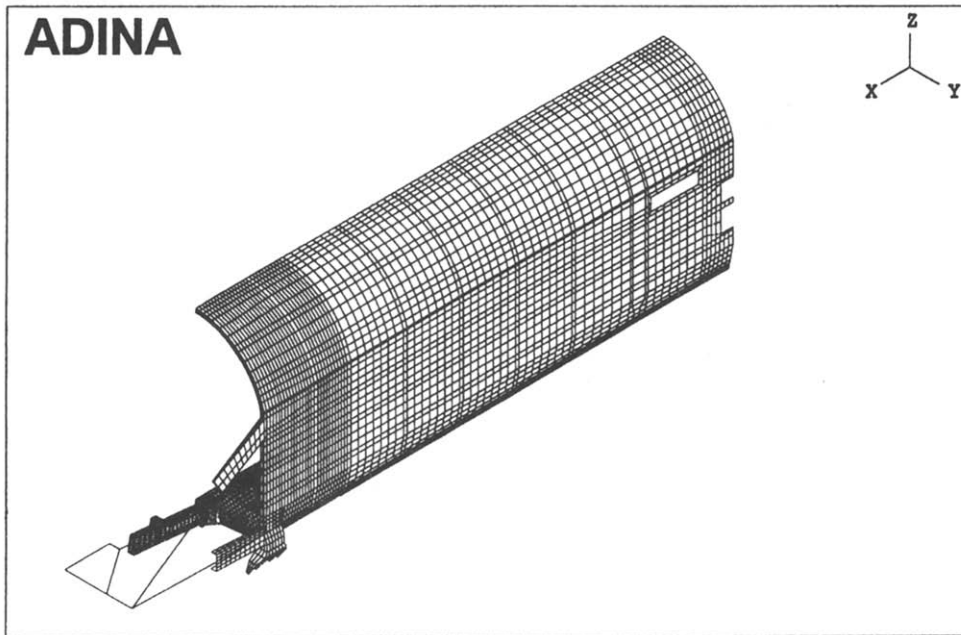|	COLSOL	ITESOL	Ratio
CPU time	6137 sec ($\sim$1.75 hr)	1034 sec ($\sim$0.25 hr)	5.94
Storage requirements (words)	191,419,272	12,863,211	14.9

Fig. 2. Model of a railroad car.

2.1. Analysis of cover of a reactor pressure vessel

Figure 1 shows the finite element model considered. The model consists of 3-D solid 20-node isoparametric elements. The model was run on an IBM RS 6000/550 workstation.

Table 1 summarizes data of the model and the solution statistics. We see that in this analysis, the use of the iterative solution scheme resulted in very significant savings.

2.2. Analysis of a railroad car

Figure 2 shows the finite element model of the railroad car analyzed. This model consists of four-node shell elements with some beam elements, and was run for a linear analysis on an IBM RS 6000/550 workstation.

Table 2 summarizes the solution statistics, and we see that in this case no gain in solution time was obtained by use of the iterative solver, but there were significant storage savings.

We want to mention here another area where in the development of finite element procedures significant advances have been made during the last decade; namely, the development of effective general shell elements that are reliable and versatile (e.g. the elements do not contain spurious zero energy modes and pass the patch test) [4].

2.3. Analysis of a slice of a turbine disk

Figure 3 shows the slice of a turbine disk. The model consists of 3-D solid elements with elasto-plasticity and contact surfaces.

Table 3 gives the solution statistics of this analysis. Here significant savings occurred in the solution time and required disk storage when using the iterative solver.

Table 2. Analysis of a railroad car

Number of equations: 90,825
Elements used: four-node SHELL, MITC4 elements, BEAM elements
Type of analysis: linear static
Number of iterations (ITESOL): 3779
Computer: IBM RISC 6000/550 workstation

	COLSOL	ITESOL	Ratio
CPU time	4578 sec ($\sim$1.25 hr)	4743 sec ($\sim$1.25 hr)	0.97
Storage requirements (words)	212,472,244	7,779,693	27.3

Table 3. Analysis of a slice of a turbine disk

Number of equations: 20,046
Elements used: 3-D SOLID, eight-node elements
Type of analysis: elasto-plastic with 3-D contact, 5 steps
Mean number of Newton–Raphson iterations per step: 4
Mean number of iterations per equation solution (ITESOL): 332
Computer: DEC Alpha AXP 3000 workstation

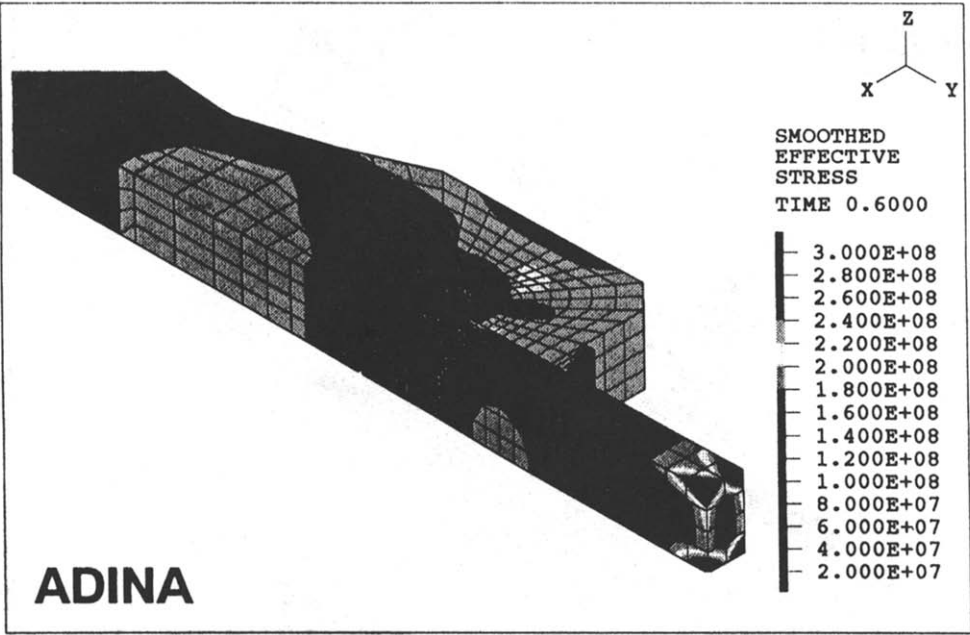|	Direct solver	Iterative solver	Ratio
CPU time	17,160 sec ($\sim$4.75 hr)	3689 sec ($\sim$1 hr)	4.65
Elapsed time	45,977 sec ($\sim$12.75 hr)	4289 sec ($\sim$1.25 hr)	10.7
Storage requirements (words)	25,312,018	1,976,514	12.8

Fig. 3. Model of slice of a turbine disk.

2.4. *Analysis of a dam*

Figure 4 shows the dam analyzed with ADINA for its linear response. The concrete dam and the foundation were modeled using 20-node 3-D solid and eight-node (MITC8) shell elements.

Table 4 summarizes the solution statistics. The benefits of using the iterative solver are quite significant.

2.5. *Analysis of an F-16 jet fighter wheel*

Figure 5 shows the jet fighter wheel considered in this analysis. The model consists of 3-D solid elements and many contact surfaces (shown by thick lines).

Table 5 gives the solution statistics. The analysis using the iterative solver required significantly less disk storage.

3. SOLUTION OF NAVIER–STOKES FLUIDS INCLUDING FLUID–STRUCTURAL INTERACTIONS

In recent years, fluid flow analysis programs have been used to an increasing extent in mechanical and other industries. In the ADINA system, the CFD module ADINA-F is employed to solve 2- and 3-D Navier–Stokes flows, with or without heat transfer, and fluid–structure coupled problems can also be analyzed by use of ADINA and ADINA-F.

Of particular practical interest is the solution of fluid flows of high Peclet numbers (due to high convective heat transfer effects), and of high Reynolds numbers. Of course, the CFD capability should also be able to solve low Reynolds number flows.. Hence, an effective finite element code must be based on elements that satisfy the ellipticity and inf–sup conditions for stability [2, 5], and upwinding

Table 4. Static analysis of a dam

Number of equations: 22,103
Elements used: 3-D SOLID, 20-node elements, eight-node SHELL, MITC8 elements
Type of analysis: linear static
Number of iterations (ITESOL): 547
Computer: DEC Alpha AXP 3000 workstation

	Direct solver	Iterative solver	Ratio
CPU time	1006 sec (~0.25 hr)	386 sec (~6 min)	2.61
Elapsed time	2179 sec (~0.5 hr)	507 sec (~8 min)	4.30
Storage requirements (words)	30,322,866	2,614,260	11.6

Table 5. Three-dimensional contact analysis of an F-16 jet wheel

Number of equations: 51,159
Elements used: 3-D SOLID, 20-node elements
Type of analysis: elastic with 3-D contact
Number of Newton–Raphson iterations: 8
Mean number of iterations per equation solution (ITESOL): 759
Computer: DEC Alpha AXP 3000 workstation

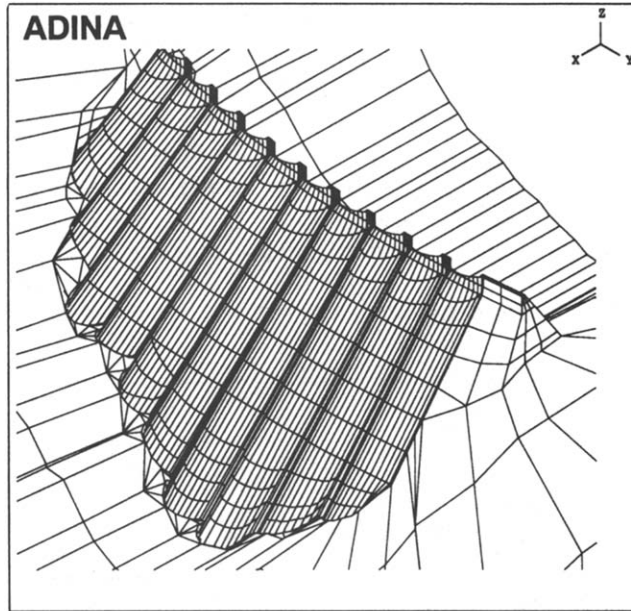|	Direct solver	Iterative solver	Ratio
CPU time	15,923 sec (~4.5 hr)	13,726 sec (~3.75 hr)	1.16
Elapsed time	49,987 sec (~14 hr)	15,131 sec (~4.25 hr)	3.30
Storage requirements (words)	119,210,734	9,523,056	12.5

Fig. 4. Model of a dam and foundation.

techniques that allow the solution at high Reynolds and Peclet numbers with reasonable discretizations [2, 6].

In ADINA-F we use the MINI element (triangle for 2-D flows, tetrahedron for 3-D flows) with techniques of upwinding based on control volume methods. This element shows excellent accuracy for low Reynolds and Peclet number flows (and is mathematically proven to do so, see for example [7]), and with our control volume type upwinding procedures can also be used efficiently for high Reynolds and Peclet number flows.

In addition to the MINI element, ADINA-F also offers the Taylor–Hood nine-node (for 2-D solutions) and 27-node (for 3-D solutions) elements to analyze low Reynolds and Peclet number flows.

The finite element models can be solved using a direct or an iterative solver. For the iterative solution, the biconjugate gradient or GMRES methods with an incomplete Cholesky pre-conditioner can be used [3].
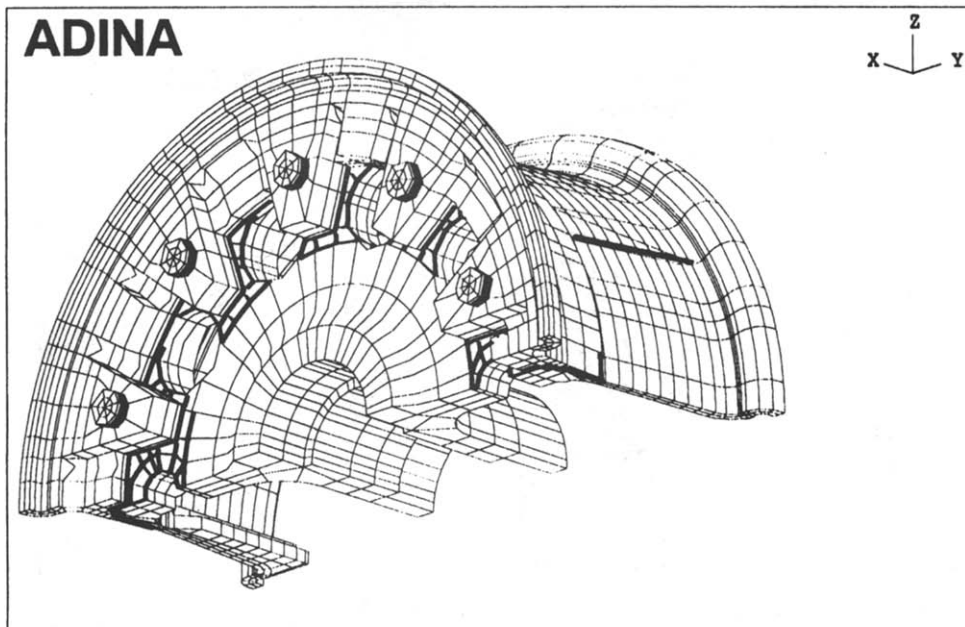

Fig. 5. Model of an F-16 jet fighter wheel.

In the following, we show the solution of fluid flow problems to demonstrate some of the mentioned capabilities.

3.1. *Airflow in a tractor cab*

Figure 6 shows results obtained in the analysis of the airflow in a tractor cab. The air flow was analyzed for laminar and turbulent flows, at Re $\sim 1.25 \times 10^5$, and the figure shows the response predicted. For the turbulent flow the $k-\epsilon$ model was used. The overall flow response predicted in the two analyses is quite similar but, of course, the near wall velocities are different.

The two solutions were both obtained on an HP 9000/730 computer using less than one hour of CPU time. We note that this kind of analysis can, without difficulty, be performed on today's workstations.

3.2. *High Reynolds number flow in a 3-D channel*

Figure 7 depicts the analysis of the 3-D channel flow. The analysis was performed for Re $\sim 10^5$, and using a finer discretization, on workstations, the solution could only be obtained using the iterative solver. For example, using an HP 9000/730 EWS, for
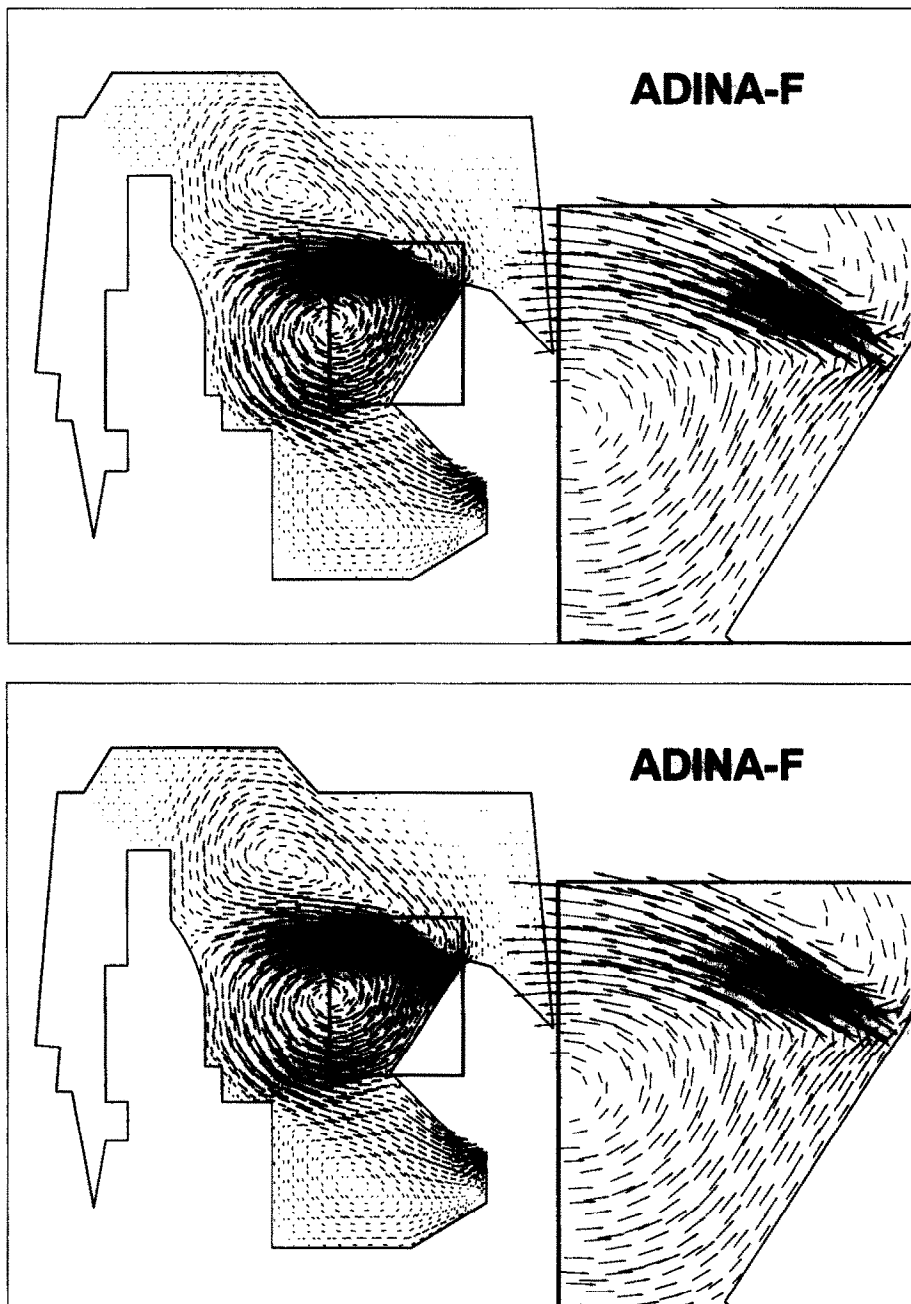

Fig. 6. Airflow in a tractor cab; top: laminar flow; bottom: turbulent flow ($k-\epsilon$ model).
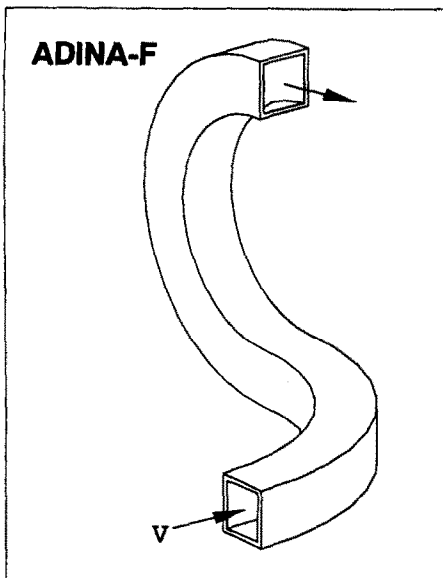
Fig. 7. Three-dimensional channel.

a system of 43,386 degrees of freedom, the CPU time was 2734 sec. This experience shows the importance of an effective iterative scheme for the solution of large finite element models.

3.3. Fluid–structure interactions

The analysis of fluid–structure interactions, in which the fluid is modeled as a Navier–Stokes fluid and general structural deformations are taken into account, can be an important asset in the analysis/design of a mechanical component. Here we show the results of three different analyses.

The Reynolds numbers for these flows are between 10^4 and 10^6. The coupled solutions have been obtained using ADINA (the structural analysis is based on the Lagrangian formulation) with ADINA-F (the fluid–flow analysis is based on an arbitrary Lagrangian–Eulerian formulation).

Figure 8 shows the results of analysis of a control valve, in which the fluid opening depends on the fluid velocity. The figure shows the fluid velocities, pressure distribution and the mesh near the opening with and without flow.

Figure 9 shows the fluid flow in a flexible (rubber) duct. The structural deformations of the originally constant radius duct and the nonuniform pressure distribution are clearly seen.

Figure 10 shows the structural response of a right-angled nozzle and the fluid flow therein. The horizontal top part of the nozzle is a flexible membrane while the other structural sides are relatively stiff. The responses for three different fluid inlet velocities are shown. The deformations of the structure are very large in the highest velocity case.

4. ANALYSIS OF RADIATION BETWEEN COMPLEX SURFACES

The solution of radiative heat exchange between complex surfaces generally represents a difficult computational problem. Response predictions are frequently based on major geometric assumptions. However, it is now possible to directly discretize structures and include the radiative heat exchange between any surfaces without geometric simplifications (beyond those generally used in a finite element solution). This analysis capability is available in the
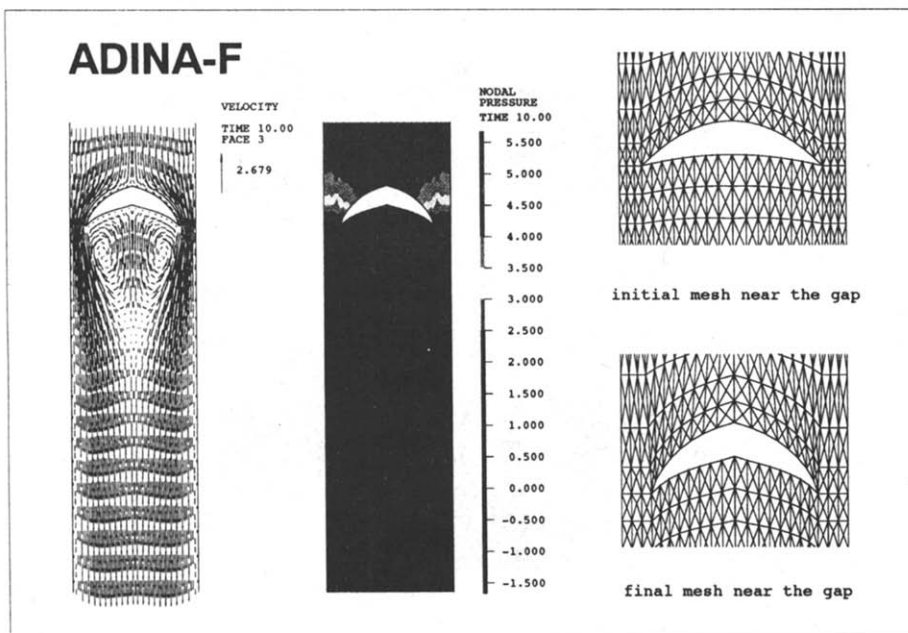

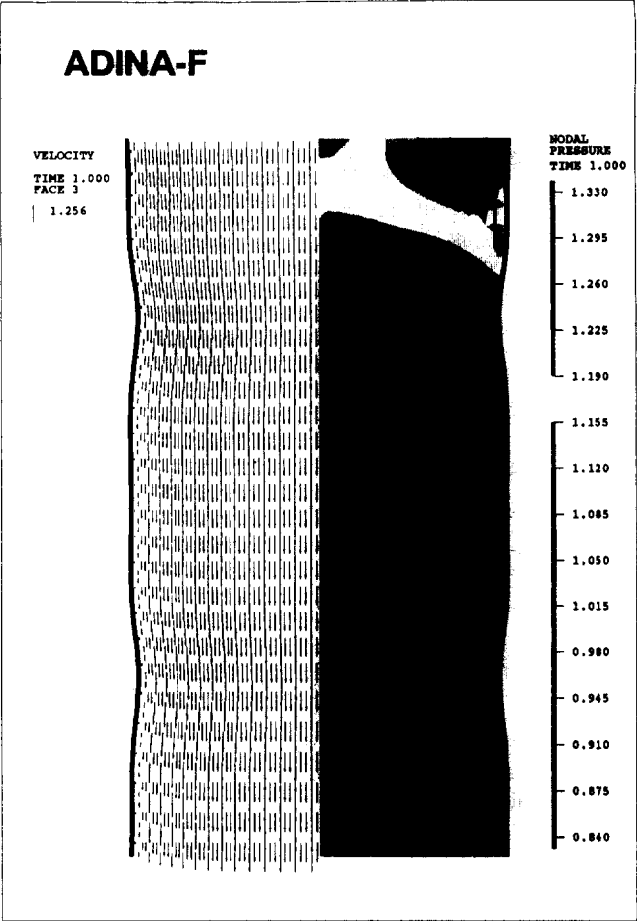Fig. 8. Flow in a control valve. Structural deformations increase flow.

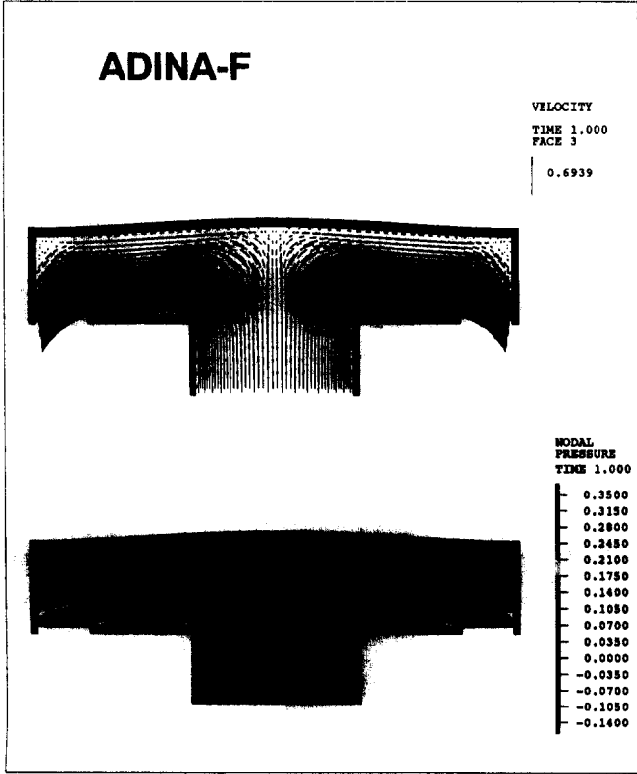Fig. 9. Flow in a rubber duct. Effect of flexibility of duct on pressure distribution.

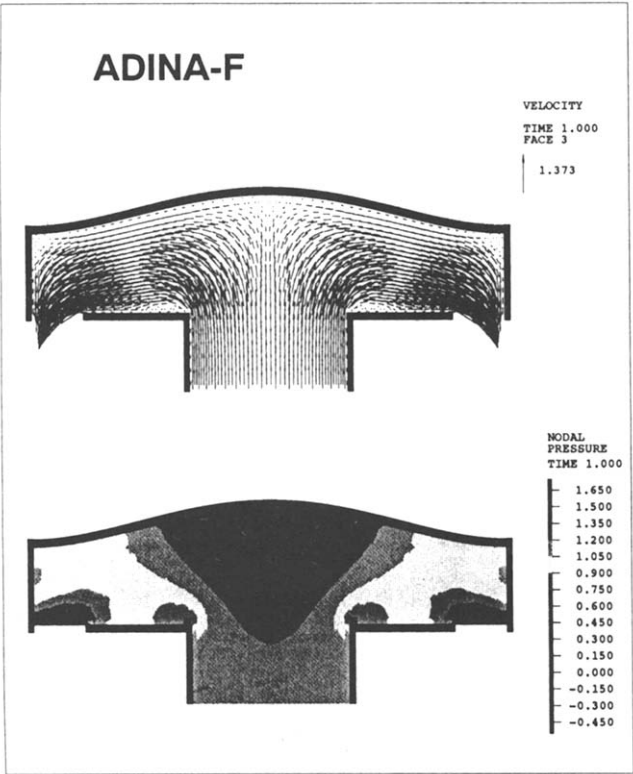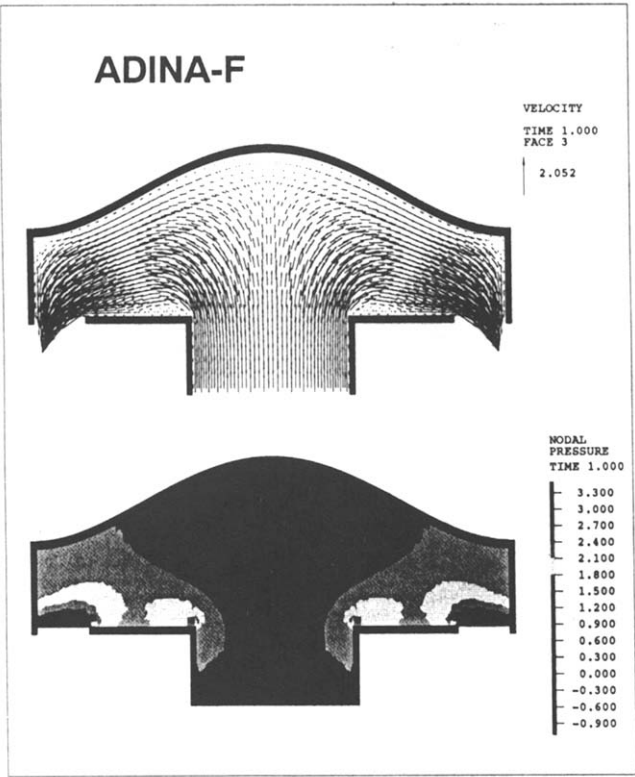Fig. 10a. (*Caption on opposite page.*)

Fig. 10b.

Fig. 10c.

Fig. 10. Flow in a nozzle. Large deformations of initially horizontal cap change flow and pressure distributions: (a) low inlet flow velocity, (b) medium inlet flow velocity, and (c) high inlet flow velocity.
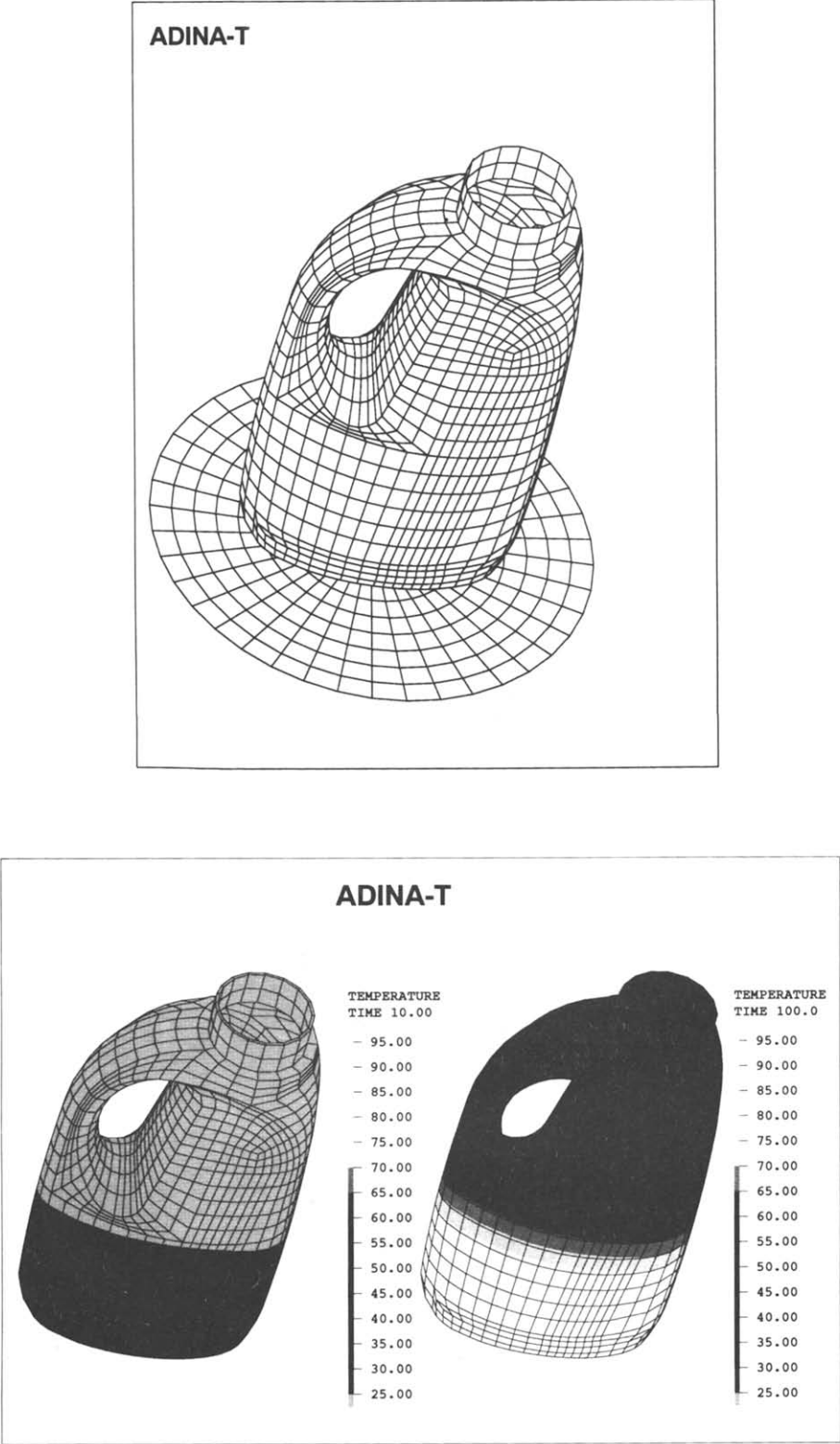
Fig. 11. Temperature analysis of bottle due to radiative heat exchange between bottle and plate; top: model of bottle and plate; bottom: temperature distributions at times 10 and 100 sec.

ADINA system by use of radiosity degrees of freedom (in addition to temperature degrees of freedom) [2].

Figure 11 shows the shell model of a bottle exposed to radiation from a horizontal round plate. A transient analysis was performed, and Fig. 11 shows some solution results. The radiative interaction between the plate and bottle surfaces (including the effects of the bottle surface onto itself) was directly included in the analysis.

5. CONCLUDING REMARKS

The solution of complex engineering analysis problems can now be obtained at a much lower cost than a few years ago. Of course, the price of hardware continues to decrease, but also software capabilities have increased significantly. The improvements in the finite element program capabilities pertain to the preparation of the models—possibly using interfaces to CAD programs—the solution of these models, and the evaluation of the results.

In this paper we briefly summarized some solution experiences with emphasis on large practical models solved on widely available workstations. The objective was to illustrate—albeit rather briefly—currently available capabilities and how these are used in engineering environments.

REFERENCES

1. K. J. Bathe, Some remarks and references on recent developments in finite element analysis procedures. *Comput. Struct.* **40**, 201–202 (1991).
2. K. J. Bathe, *Finite Element Procedures in Engineering Analysis*, 2nd Edn. Prentice-Hall, Englewood Cliffs, NJ, to be published.
3. L. H. Tan and K. J. Bathe, Studies of finite element procedures—the conjugate gradient and GMRES methods in ADINA and ADINA-F. *Comput. Struct.* **40**, 441–449 (1991).
4. K. J. Bathe and E. N. Dvorkin, A formulation of general shell elements—the use of mixed interpolation of tensorial components. *Int. J. Numer. Meth. Engng* **22**, 697–722 (1986).
5. F. Brezzi and K. J. Bathe, A discourse on the stability conditions for mixed finite element formulations. *Comput. Meth. Appl. Mech. Engng.* **82**, 27–57 (1990).
6. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing (1980).
7. D. Chapelle and K. J. Bathe, The inf–sup test. *Comput. Struct.* **47**, 537–545.