PARALLEL FINITE ELEMENT COMPUTATION OF MISSILE AERODYNAMICS

W. B. STUREK[1,*], S. RAY[1], S. ALIABADI[2], C. WATERS[2] AND T. E. TEZDUYAR[2]

[1] *U.S. Army Research Laboratory, Aberdeen Proving Ground, MD 21005-5066, U.S.A.*
[2] *Department of Aerospace Engineering and Mechanics, Army High Performance Computing Research Center, University of Minnesota, 1100 Washington Avenue South, Minneapolis, MN 55415, U.S.A.*

SUMMARY

A flow simulation tool, developed by the authors at the Army HPC Research Center, for compressible flows governed by the Navier–Stokes equations is used to study missile aerodynamics at supersonic speeds, high angles of attack and for large Reynolds numbers. The goal of this study is the evaluation of this Navier–Stokes computational technique for the prediction of separated flow fields around high-length-to-diameter (L/D) bodies. In particular, this paper addresses two issues: (i) turbulence modelling with a finite element computational technique and (ii) efficient performance of the computational technique on two different multiprocessor mainframes, the Thinking Machines CM-5 and CRAY T3D. The paper first provides a discussion of the Navier–Stokes computational technique and the algorithm issues for achieving efficient performance on the CM-5 and T3D. Next, comparisons are shown between the computation and experiment for supersonic ramp flow to evaluate the suitability of the turbulence model. Following that, results of the computations for missile flow fields are shown for laminar and turbulent viscous effects. ©1997 by John Wiley & Sons, Ltd.

Int. J. Numer. Meth. Fluids, **24**: 1417–1432, 1997

No. of Figures: 13. No. of Tables: 0. No. of References: 15.

KEY WORDS: parallel computing methods; compressible flows; missile aerodynamics

1. INTRODUCTION

The compressible Navier–Stokes flow simulation tool developed by the authors at the Army HPC Research Center (AHPCRC) is used to study missile aerodynamics at supersonic speeds and high angles of attack for large-Reynolds-numbers flow conditions. The goal of this study is the evaluation of this Navier–Stokes computational technique for the prediction of separated flow fields around high-length-to-diameter (L/D) bodies. In particular, this paper addresses two issues: (i) turbulence modelling within the finite element computational technique and (ii) efficient performance of the computational technique on the Thinking Machines CM-5 and CRAY T3D, both multiprocessor mainframes.

This study is also part of the effort in participating in a joint collaborative working group (KTA 2-12) under the auspices of The Technical Cooperation Program (TTCP). TTCP is a joint program with the United States, United Kingdom, Canada, Australia and New Zealand which was established for the purpose of exchange of technical information between government agencies of these five countries.

*Correspondence to: W.B. Sturek, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD 21005-5066, U.S.A.

CCC 0271–2091/97/121417–16 $17.50
Received 15 April 1996
Revised 10 July 1996

Since Army missiles and projectiles fly under conditions of moderate to high angles of attack at transonic and supersonic velocities, turbulent viscous effects are important and must be modelled accurately in order for the predicted results to be of value in the design process. Also, owing to the complex geometry of missiles, which includes fin surfaces for guidance and flight stability, significant computational resources in terms of memory and CPU time are required to obtain solutions. Thus the efficient performance of the Navier–Stokes solver on large, scalable computing mainframes is also of keen interest.

Through participation in the TTCP working group (KTA 2-12), a set of experimental data has become available for comparison with the computations. These computations have provided guidance for evaluation of the turbulent viscous effects. Initial comparisons indicate that laminar viscous computations are unable to adequately predict important features of the flow field, including surface pressure distributions and the separated vortex flow field. The KTA study has not yet been completed, so direct comparisons of the computational results with experiment will not be shown here. However, experimental measurements of a turbulent boundary layer in a supersonic compression ramp flow are available for evaluation of the ability of the computational technique to predict this turbulent viscous flow field.

This paper provides a discussion of the Navier–Stokes computational technique and the algorithm issues for achieving efficient performance on the CM-5 and T3D. Next, comparisons are shown between the computation and experiment for supersonic ramp flow to judge the suitability of the turbulence model. Finally, results of the computations for the missile flow field are shown for laminar and turbulent viscous effects.

2. GOVERNING EQUATIONS

Let $\Omega \subset \mathbb{R}^{n_{sd}}$ and (0, T) be the spatial and temporal domains respectively, where n_{sd} is the number of space dimensions, and let Γ denote the boundary of Ω The spatial and temporal co-ordinates are denoted by $\mathbf{x}$ and t respectively. We consider the Navier–Stokes equations for 3D unsteady, compressible flows. These equations in conservation law form can be written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{u}\rho) = 0 \quad \text{on } \Omega, \quad \forall t \in (0, T), \tag{1}$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla \cdot (\mathbf{u}\rho\mathbf{u}) = -\nabla p + \nabla \cdot \mathbf{T} \quad \text{on } \Omega, \quad \forall t \in (0, T), \tag{2}$$

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\mathbf{u}\rho e) = -\nabla \cdot \mathbf{q} - \nabla \cdot (p\mathbf{u}) + \nabla \cdot (\mathbf{T}\mathbf{u}) \quad \text{on } \Omega, \quad \forall t \in (0, T). \tag{3}$$

Here $\rho(\mathbf{x}, t)$, $\mathbf{u}(\mathbf{x}, t)$, $p(\mathbf{x}, t)$ and $e(\mathbf{x}, t)$ are respectively density, velocity, pressure and the total energy per unit mass. The viscous stress tensor and heat flux vector are denoted by $\mathbf{T}$ and $\mathbf{q}$ respectively. Pressure is related to the other state variables with the equation of state of the form

$$p = p(\rho, i), \tag{4}$$

where i is the internal energy

$$i = e - \tfrac{1}{2} \| \mathbf{u} \|^2 . \tag{5}$$

For ideal gases the equation of state takes the form

$$p = (\gamma - 1)\rho i, \tag{6}$$

where γ is the ratio of the specific heats. The viscous stress tensor **T** and heat flux vector **q** are defined as

$$\mathbf{T} = \lambda(\nabla \cdot \mathbf{u})\mathbf{I} + 2\mu\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} = (\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}})/2, \tag{7}$$

$$\mathbf{q} = -\kappa\nabla\theta, \tag{8}$$

where **I** is the identity tensor, κ is the conductivity and θ is the temperature with the following relationship to the internal energy:

$$i = \frac{R\theta}{\gamma - 1}. \tag{9}$$

Here *R* is the ideal gas constant and it is assumed that the viscosity coefficients λ and μ are related:

$$\lambda = -\tfrac{2}{3}\mu. \tag{10}$$

The variation in the viscosity with temperature is modelled by Sutherland's empirical formula:

$$\mu = \mu_{\mathrm{r}}\left(\frac{\theta}{\theta_{\mathrm{r}}}\right)^{3/2}\frac{\theta_{\mathrm{r}} + \theta_0}{\theta + \theta_0}, \tag{11}$$

where θ_0 is an experimentally determined constant and μ_{r} is the viscosity at the reference temperature θ_{r}. The Prandtl number *Pr*, assumed to be given, relates the heat conductivity to the viscosity according to

$$\kappa = \frac{\gamma R\mu}{(\gamma - 1)Pr}. \tag{12}$$

In terms of conservation variables the Navier–Stokes equations of compressible flows given by equations (1)–(3) can be written in the vector form

$$\frac{\partial\mathbf{U}}{\partial t} + \frac{\partial\mathbf{F}_i}{\partial x_i} - \frac{\partial\mathbf{E}_i}{\partial x_i} = \mathbf{0} \quad \text{on } \Omega, \quad \forall t \in (0, T), \tag{13}$$

where $\mathbf{U} = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho e)$ is the vector of conservation variables and $\mathbf{F}_i$ and $\mathbf{E}_i$ are respectively the Euler and viscous flux vectors defined as

$$\mathbf{F}_i = \begin{pmatrix} u_i\rho \\ u_i\rho u_1 + \delta_{i1}p \\ u_i\rho u_2 + \delta_{i2}p \\ u_i\rho u_3 + \delta_{i3}p \\ u_i(\rho e + p) \end{pmatrix}, \tag{14}$$

$$\mathbf{E}_i = \begin{pmatrix} 0 \\ [\mathbf{T}]_{i1} \\ [\mathbf{T}]_{i2} \\ [\mathbf{T}]_{i3} \\ -q_i + [\mathbf{T}]_{ik}u_k \end{pmatrix}, \tag{15}$$

with u_i, q_i and $[\mathbf{T}]_{ij}$ denoting the components of velocity, heat flux and viscous stress tensor respectively. Here repeated indices imply summation over the range of the spatial dimension and the identity tensor is denoted by δ_{ij}.

To provide a convenient set-up for our finite element formulations, equation (13) is written in the form

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}\frac{\partial \mathbf{U}}{\partial x_j}\right) = \mathbf{0} \quad \text{on } \Omega, \quad \forall t \in (0, T), \tag{16}$$

where

$$\mathbf{A}_i = \frac{\partial \mathbf{F}_i}{\partial \mathbf{U}} \tag{17}$$

is the Euler Jacobian matrix and $\mathbf{K}_{ij}$ is the diffusivity matrix satisfying

$$\mathbf{K}_{ij}\frac{\partial \mathbf{U}}{\partial x_j} = \mathbf{E}_i. \tag{18}$$

The explicit definitions of $\mathbf{A}_i$ and $\mathbf{K}_{ij}$ are provided in Reference 1.

Equation (16) is complemented with initial and boundary conditions of the form

$$\mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0, \tag{19}$$

$$\mathbf{U} \cdot \mathbf{e}_k = g_k \quad \text{on } (\Gamma)_{g_k}, \quad k = 1, \ldots, n_{\text{dof}}, \tag{20}$$

$$(n_i \cdot \mathbf{E}_i) \cdot \mathbf{e}_k = h_k \quad \text{on } (\Gamma)_{h_k}, \quad k = 1, \ldots, n_{\text{dof}}, \tag{21}$$

where $\mathbf{e}_k$ is an orthonormal basis function in $\mathbb{R}^{n_{\text{dof}}}$ and n_{dof} is the number of degrees of freedom.

The turbulence computations are carried out using the Smagorinsky turbulence model.[2] In this model, μ is replaced by

$$\mu \leftarrow \mu + \mu_{\text{turb}}, \tag{22}$$

where μ_{turb} is the turbulence diffusion defined as

$$\mu_{\text{turb}} = \rho(0{\cdot}15h)^2(2[\boldsymbol{\epsilon}]_{ij}[\boldsymbol{\epsilon}]_{ij})^{1/2}, \tag{23}$$

with h denoting the element length.

3. LIMITATIONS OF THE SMAGORINSKY TURBULENCE MODEL

The Smagorinsky turbulence model[2] is normally used in large-eddy simulations (LES) where the large-scale components are resolved directly and the small-scale fluctuations are modelled to reflect the effects of subgrid stresses on the large-scale field. It is well known that any class of LES requires a higher degree of grid resolution compared with Reynolds-averaged Navier–Stokes (RANS) models.[3–5] As mentioned in Reference 6, LES of the aerodynamics of an aerofoil at a Reynolds number of 10^7 requires 80 million grid points. Therefore 3D simulation of Navier–Stokes application problems at high Reynolds numbers is not feasible with the LES method. Up to now the Smagorinsky turbulence model has been widely used to simulate 3D problems at moderate Reynolds numbers up to 10,000 with affordable mesh resolution.

The structural simplicity of the algebraic Smagorinsky turbulence model compared with RANS models has made it a practical choice. This model is especially well suited to finite element methods, since the eddy viscosity can be obtained from the quantities available within each individual element. On the other hand, there are fundamental issues concerning the applicability of this model to flows of engineering interest.[6] Also, one should be aware of some practical deficiencies of this model. As mentioned by Moin and Jimenez,[6] these deficiencies include (i) tuning of the model constant for different flows, (ii) the non-vanishing behaviour near solid boundaries and (iii) the presence of the model in laminar regions.

In this article we use the Smagorinsky turbulence model with meshes which are typical of RANS computations. Here we address the usability of this model in conjunction with the most powerful supercomputers available to model supersonic flow problems. Simulations are carried out for two types of flow problems with a level of grid refinement so as to obtain a solution in a reasonable engineering time frame. The grid resolution in our approach does not necessarily push these supercomputers to their limits of computational power.

The steady state solutions reported in this article are obtained using the time-dependent code and relatively large time increments. The backward Euler scheme is also used to insure fast convergence to the steady state solutions. It is obvious that this time integration strategy removes any unsteady behaviour. As a result, the possibility of small-scale unsteadiness will be totally eliminated from the solutions.

In our experiments the results indicate that simulations using the Smagorinsky turbulence model produce acceptable solutions for complicated 3D problems. However, the reliability and suitability of this model for these types of flow simulations in general remains an open issue and requires further investigation.

4. SEMIDISCRETE FINITE ELEMENT FORMULATION

Consider a finite element discretization of a fixed spatial domain Ω into subdomains $\Omega^e, e=1,2,\ldots,n_{\mathrm{el}}$, where n_{el} is the number of elements. Based on this discretization, corresponding to the trial solutions and weighting functions respectively, we define the finite element function spaces $\mathcal{S}^h$ and $\mathcal{V}^h$ for conservation variables. These function spaces are selected as subsets of $[\mathbf{H}^{1h}(\Omega)]^{n_{\mathrm{dof}}}$, where $\mathbf{H}^{1h}(\Omega)$ is the finite-dimensional function space over Ω

$$\mathcal{S}^h=\{\mathbf{U}^h|\mathbf{U}^h\in[\mathbf{H}^{1h}(\Omega)]^{n_{\mathrm{dof}}},\mathbf{U}^h|_{\Omega^e}\in[P^1(\Omega^e)]^{n_{\mathrm{dof}}},\mathbf{U}^h\cdot\mathbf{e}_k\doteq g_k\text{ on }\Gamma_{g_k}\},\tag{24}$$

$$\mathcal{V}^h=\{\mathbf{W}^h|\mathbf{W}^h\in[\mathbf{H}^{1h}(\Omega)]^{n_{\mathrm{dof}}},\mathbf{W}^h|_{\Omega^e}\in[P^1(\Omega^e)]^{n_{\mathrm{dof}}},\mathbf{W}_h\cdot\mathbf{e}_k\doteq 0\text{ on }\Gamma_{g_k}\},\tag{25}$$

where $[P^1(\Omega^e)]$ represents the first-order polynomial in Ω^e.

The stabilized finite element formulation of (16) is written as follows: find $\mathbf{U}^h\in\mathcal{S}^h$ such that $\forall\mathbf{W}^h\in\mathcal{V}^h$

$$\begin{aligned}&\int_\Omega\mathbf{W}^h\cdot\left(\frac{\partial\mathbf{U}^h}{\partial t}+\mathbf{A}_i^h\frac{\partial\mathbf{U}^h}{\partial x_i}\right)\mathrm{d}\Omega+\int_\Omega\left(\frac{\partial\mathbf{W}^h}{\partial x_i}\right)\cdot\left(\mathbf{K}_{ij}^h\frac{\partial\mathbf{U}^h}{\partial x_j}\right)\mathrm{d}\Omega\\&+\sum_{e=1}^{n_{\mathrm{el}}}\int_{\Omega^e}\tau(\mathbf{A}_k^h)^{\mathrm{T}}\left(\frac{\partial\mathbf{W}^h}{\partial x_k}\right)\cdot\left[\frac{\partial\mathbf{U}^h}{\partial t}+\mathbf{A}_i^h\frac{\partial\mathbf{U}^h}{\partial x_i}-\frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}^h\frac{\partial\mathbf{U}^h}{\partial x_j}\right)\right]\mathrm{d}\Omega\\&+\sum_{e=1}^{n_{\mathrm{el}}}\int_{\Omega^e}\delta\left(\frac{\partial\mathbf{W}^h}{\partial x_i}\right)\cdot\left(\frac{\partial\mathbf{U}^h}{\partial x_i}\right)\mathrm{d}\Omega=\int_{\Gamma_{\mathrm{h}}}\mathbf{W}^h\cdot\mathbf{h}^h\mathrm{d}\Gamma.\end{aligned}\tag{26}$$

In the variational formulation given by (26), the first two terms together with the right-hand-side term constitute the Galerkin formulation of the problem. The first series of element-level integrals in (26) consists of the SUPG stabilization terms added to the variational formulation to prevent spatial oscillations in the advection-dominated range. The second series of element-level integrals in (26) consists of the shock-capturing terms added to the formulation to ensure stability at high Mach and Reynolds number.

The definitions of the diagonal stabilization matrix $\boldsymbol{\tau}$ and shock-capturing parameter δ are given in Reference 7.

5. NON-LINEAR BOUNDARY CONDITIONS

The boundary conditions associated with the finite element formulation (26) are in terms of conservation variables. These variables are different from the primitive variables which are normally used to define the physical boundary conditions.

The specification of the physical boundary conditions can be incorporated into the iterative algorithm within the framework of the conservation variables. In the compression ramp problem presented in this article, the temperature needs to be specified on the solid surface. Note that on a fixed solid surface $\mathbf{u}=0$ and therefore

$$\theta=\frac{\gamma-1}{R}\frac{U_5}{U_1}. \tag{27}$$

The variation of equation (27) is

$$\delta\theta=\frac{\gamma-1}{R\rho}\delta U_5-\frac{\theta}{\rho}\delta U_1. \tag{28}$$

Since the weighting functions can be seen as variations in the corresponding variables, equation (28) can be written in terms of the weighting functions:

$$W_\theta=\frac{\gamma-1}{R\rho}W_5-\frac{\theta}{\rho}W_1, \tag{29}$$

where W_θ is the weighting function corresponding to the temperature.

At any arbitrary node A, using equation (29), a new set of weighting functions can be defined such that

$$\mathbf{W}=\mathbf{S}_\text{A}\mathbf{W}^*, \tag{30}$$

where

$$\mathbf{W}^*=\begin{pmatrix}W_1\\W_2\\W_3\\W_4\\W_\theta\end{pmatrix},\qquad \mathbf{S}_\text{A}=\begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&1&0\\\dfrac{R}{\gamma-1}\theta&0&0&0&\dfrac{R}{\gamma-1}\rho\end{pmatrix}. \tag{31}$$

In the iterative algorithm the left-hand-side matrix is derived using the increment of $\mathbf{U}$. Thus, following the same procedure, the increment of $\mathbf{U}$ at node A can be replaced by

$$\Delta\mathbf{U}=\mathbf{S}_\text{A}\Delta\mathbf{U}^*, \tag{32}$$

where

$$\mathbf{U}^*=\begin{pmatrix}U_1\\U_2\\U_3\\U_4\\\theta\end{pmatrix}. \tag{33}$$

To impose the temperature at point A, the finite element formulation (26) needs to be modified locally according to the algorithm outlined. The result of this modification is imposed at the element level. Owing to this transformation, the element-level quantities are replaced recursively as

$$\mathbf{m}^e \longleftarrow (\mathbf{S}^e)^{\mathrm{T}} \mathbf{m}^e \mathbf{S}^e, \tag{34}$$

$$\mathbf{r}^e \longleftarrow (\mathbf{S}^e)^{\mathrm{T}} \mathbf{r}^e, \tag{35}$$

where $[\mathbf{m}^e]_{(n_{\mathrm{dof}}\cdot n_{\mathrm{en}})\times(n_{\mathrm{dof}}\cdot n_{\mathrm{en}})}$ and $[\mathbf{r}^e]_{(n_{\mathrm{dof}}\cdot n_{\mathrm{en}})}$ are the element-level left-hand-side matrix and residual vector respectively. Here $\mathbf{S}^e = \mathrm{diag}(\mathbf{S}_1, \ldots, \mathbf{S}_{n_{\mathrm{en}}})$ is an $(n_{\mathrm{dof}} \cdot n_{\mathrm{en}}) \times (n_{\mathrm{dof}} \cdot n_{\mathrm{en}})$ matrix with $\mathbf{S}_{\mathrm{A}}$ as the nodal block diagonal entry and n_{en} is the number of local nodes in an element. The details of the derivation can be found in Reference 1.

6. PARALLEL COMPUTATION

The stabilized finite element formulation of the Navier–Stokes equations of compressible flows is implemented on the distributed memory CM-5 and T3D supercomputers. The implementation on the CM-5 is based on the data-parallel computing paradigm and the CMF (Connection Machine Fortran) language.[8] Our implementation[9,10] on the T3D takes advantage of the Parallel Virtual Machine (PVM) library.

For efficient computation a mesh-partitioning method is used on both supercomputers to minimize the interprocessor communication. In this method the finite element mesh is partitioned into the desired number of subdomains such that the number of nodes at the interface of the subdomains is minimized. Then the gather and scatter operations, which are basically the transfer of data from the node level to the element level and vice versa, are performed in two steps: on-processor and off-processor. In the on-processor step the gather and scatter operations are local to the processor with no interprocessor communication. The communication between the processors which are mostly related to nodes on the boundary of each subdomain is performed in the off-processor step.

On the T3D the two-step gather and scatter operations are carried out using the PVM-based routines. On the CM-5 there are routines supported by the Connection Machine Scientific Software Library (CMSSL) which allow the user to perform two-step gather and scatter operations. This feature makes the finite element programming relatively easier on the CM-5 compared with the T3D.

The compressible Navier–Stokes finite element formulation gives rise to very large systems of coupled, non-linear equations which require the use of iterative strategies with update techniques such as GMRES[11] for their solution. To further reduce the memory requirements, we use matrix-free iterations and thus eliminate the need to store element-level matrices.[1,12]

The overall performance of our codes is comparable on each supercomputer. The computational speed measured for problems reported in this article is around 20 Mflops per processor on both machines.[13,14]

7. COMPUTATIONAL RESULTS

7.1. Compression Ramp

The experimental study was performed at the Supersonic Wind Tunnels of the Army Ballistic Research Laboratory and is reported in Reference 15. The wind tunnel was a boundary layer channel with a cross-section and height in the test section of 6×6 in^2. The freestream conditions are Mach 3·5, total pressure 53·35 lbf in^{-1} and total temperature 540 °R, giving a Reynolds number for the freestream flow of $6{\cdot}4 \times 10^6$ per foot. The measurements were made on the tunnel lower wall by surveys of the turbulent boundary layer perpendicular to the local surface using flattened pitot pressure and total temperature probes. Extensive detailed measurements of the turbulent boundary layer velocity profiles, surface pressure and skin friction are available for comparison with the computational results.

The computations were carried out using experimental data as the upstream boundary condition (Figure 1) and assumming an outflow outer boundary. The upper wall was modelled by turning the flow parallel to the wall. The computational grid is shown in Figure 2, where only every other grid in each direction is displayed for clarity. Also shown in Figure 2 are the identification and position of the test stations where comparisons are made between the computations and experiments. A zero-pressure-gradient condition exists for 5 in, followed by adverse pressure gradient compression ramp flow for 10 in.

A plot of the surface pressure versus axial position is shown in Plate 1 along with density contours for the laminar solution. For this condition the solution indicates two positions where local flow separation occurred which was not observed in the experiment. The density contour plot shows the development of the laminar viscous layer and the compression of the boundary layer in the flow over the ramp.

A plot of the surface pressure versus axial position is shown in Plate 2 along with density contours for the turbulent viscous solution. For this condition the flow remains attached throughout the flow as observed in the experiment. The density contour plot shows the development of the turbulent viscous layer and the compression of the boundary layer as the flow develops in the adverse pressure gradient.

Comparisons of the computations with experimental data for the boundary layer profile parameters of static pressure, static temperature, density and velocity are shown in Figures 3–6. The results indicate very good agreement at station 10, the last station in the zero-pressure-gradient region. As the flow develops over the ramp, the agreement between computation and experiment, although quite

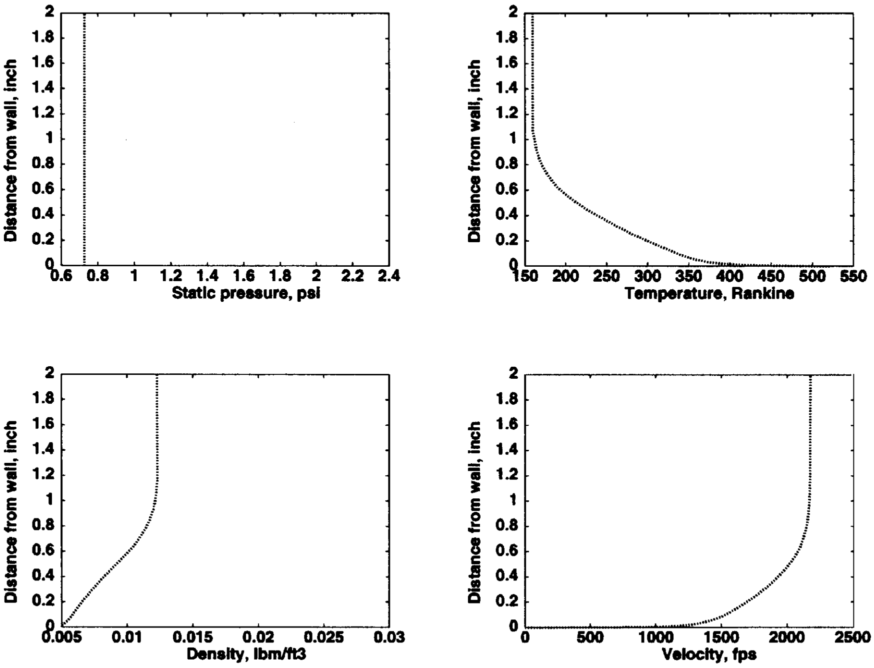

Figure 1. Compression ramp: upstream boundary values (experimental)

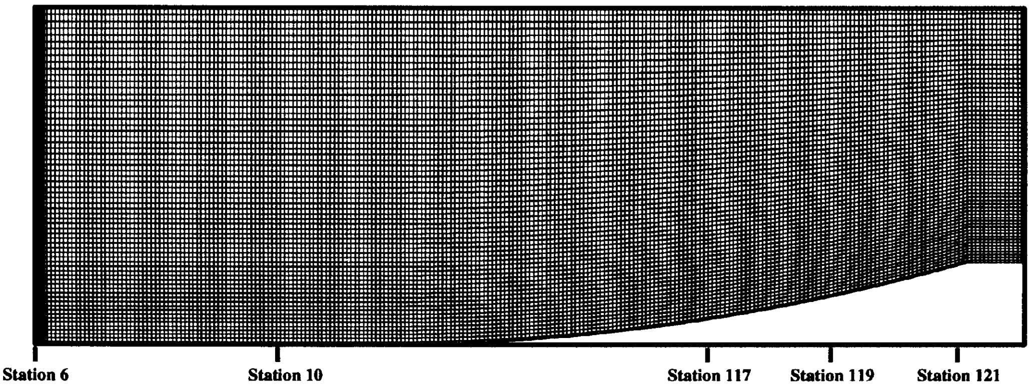

Figure 2. Compression ramp: finite element mesh (every other mesh node plotted)

good at stations 117 and 119, is not acceptable at the last station on the ramp, station 121. Although this could be an indication of inaccuracy for the turbulence model, it could also be the result of upstream influence resulting from the outflow boundary condition. Further studies will address this.

These results were obtained on the T3D using 16 processors and using a finite element mesh with 500 ×150 elements. The number of equations solved at each pseudo-time step is 381,600. The parallel implementation was as described in Section 4. For this problem a converged solution was obtained after 400 pseudo-time steps which required 50 s of CPU time for each step.

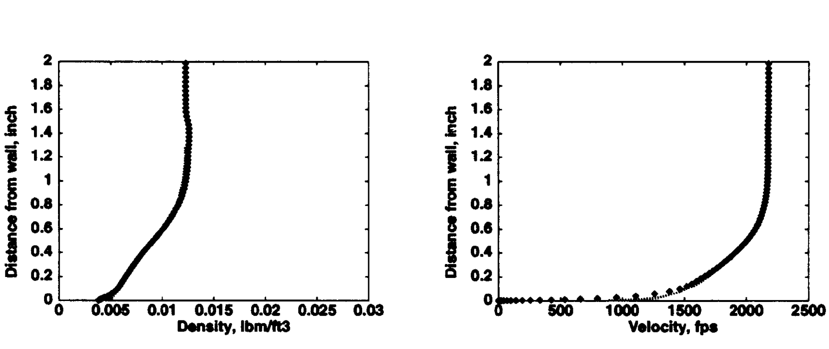

Figure 3. Compression ramp: comparison between computed (◆) and expermental (···) results at station 10

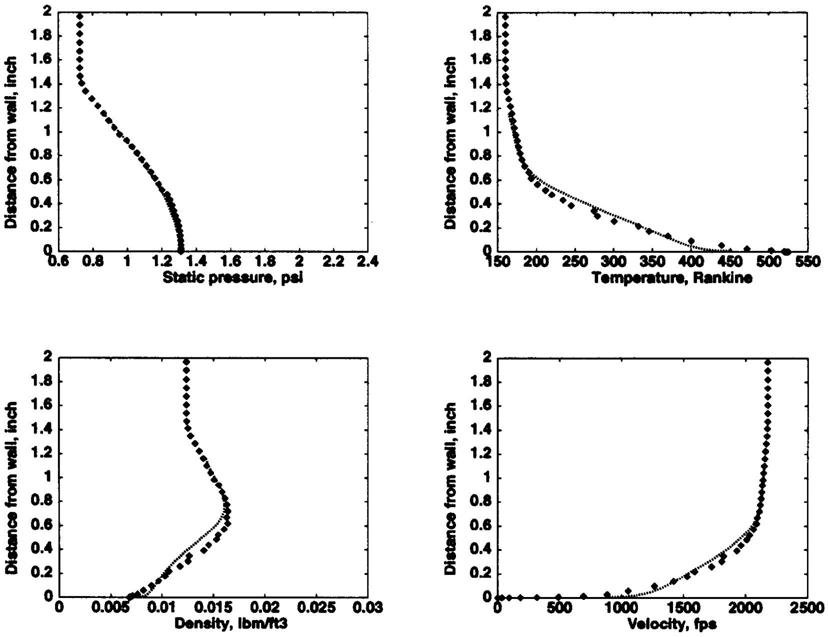

Figure 4. Compression ramp: comparison between computed (◆) and expermental (···) results at station 117

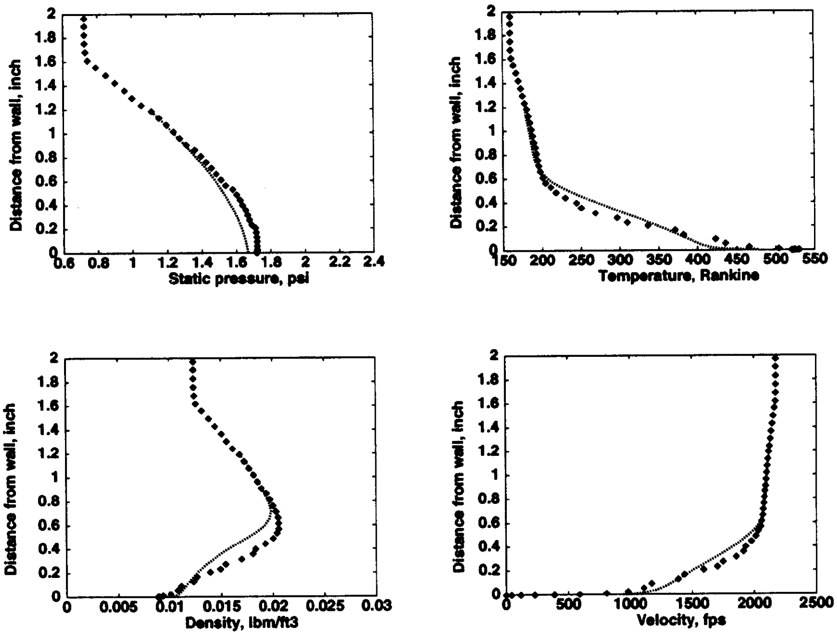

Figure 5. Compression ramp: comparison between computed (◆) and expermental (···) results at station 119

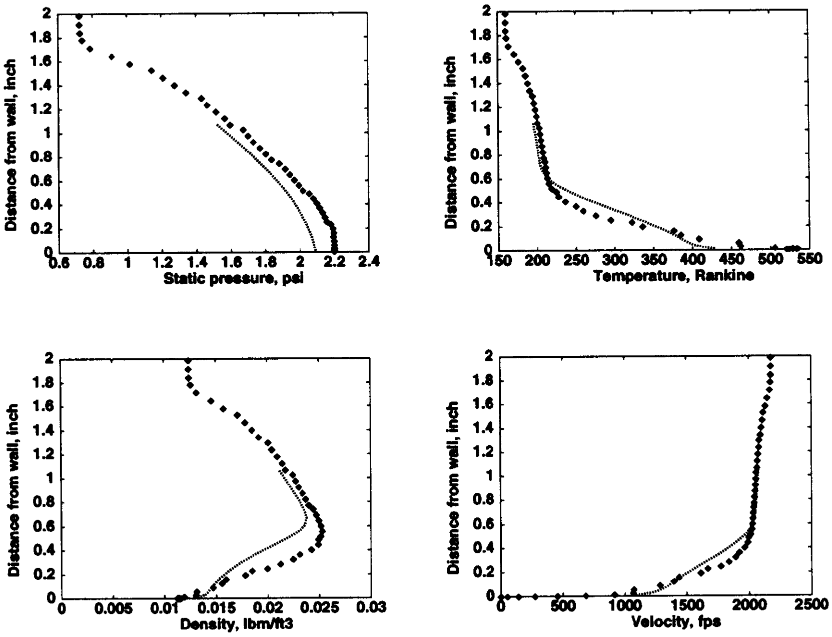

Figure 6. Compression ramp: comparison between computed (◆) and expermental (···) results at station 121

7.2. Missile

As previously discussed, the missile test case is part of a collaborative study which is being carried out under the auspices of The Technical Cooperation Program Panel W2. The participants include representatives from Canada, the U.K. and the U.S. The purpose of the study is to apply Navier–Stokes computational techniques to a complex flow field with highly separated flow for a missile shape to evaluate the predictive technology. The predictive technology includes code accuracy, usability issues, turbulence models, grid generation and flow field visualization.

The KTA study includes the flow about a missile configuration at transonic and supersonic velocities and high angles of attack. The computational results are being compared with experimental measurements for surface pressure, pitot surveys of the outer flow field and strain gauge force measurements. Since this study has not been completed, comparisons with the experimental data will not be shown at this time. Instead, we will show comparisons between results for laminar and turbulent viscous effects and discuss the performance of the computational technique on the multiprocessor computers.

In this paper, preliminary results will be shown for freestream conditions of Mach 2·5 and 14° angle of attack for laminar and turbulent viscous effects. Plate 3 shows the model configuration along with contours of the flow field static pressure on the model surface at three axial stations in the cross-plane. The regions of high pressure on the windward side at the nose and low-pressure separated flow on the lee side of the cylindrical portion of the model are clearly distinguishable.

Examples of the surface pressure at a series of axial stations are shown in Figure 7 and 8. As the flow proceeds down the missile body, increasing regions of separated flow are seen. The laminar viscous results indicate more oscillations in the surface pressure on the lee side of the model than do the turbulent viscous results. These oscillations are a result of the effects of boundary layer separation

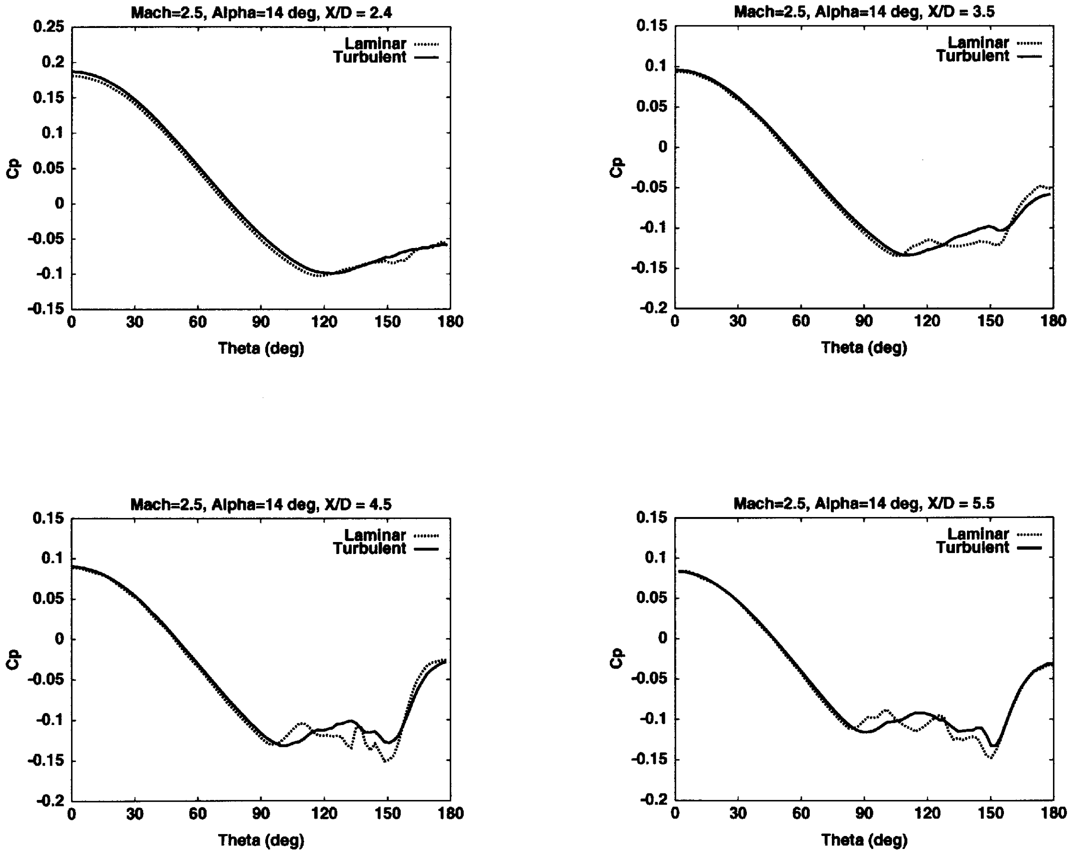

Figure 7. Missile: comparison of computed results on T3D for laminar and turbulent viscous effects for x/d =2·4, 3·5, 4·5 and 5·5

to which the laminar viscous layer is sensitive. At x/d greater than 6·5, separation prior to the 90° circumferential position is indicated for both the laminar and turbulent viscous results.

These separated flow regions are very challenging for computational prediction and require careful evaluation of the computational results with the experimental measurements to assess the validity of the turbulence modelling. The results shown here are considered to be preliminary. Evaluations are ongoing and include a study of the ability to predict the separated outer flow field as well.

The computations of the missile configuration for both laminar and turbulent flows were accomplished on the T3D computer using 256 processors with a finite element mesh consisting of 944,366 nodes and 918,000 elements. This resulted in the solution of 4,610,378 coupled, non-linear equations at each pseudo-time step. Each pseudo-time step required 130 s of CPU time. The solutions were obtained in 400 steps.

Computations for the laminar and turbulent flows were also carried out on the CM-5 with 512 processors. The mesh used in these computations consists of 763,323 nodes and 729,600 elements. During each pseudo-time step, 3,610,964 coupled, non-linear equations were solved. Each pseudo-time step required approximately 50 s of CPU time and convergence was achieved after 150 pseudo-time steps.

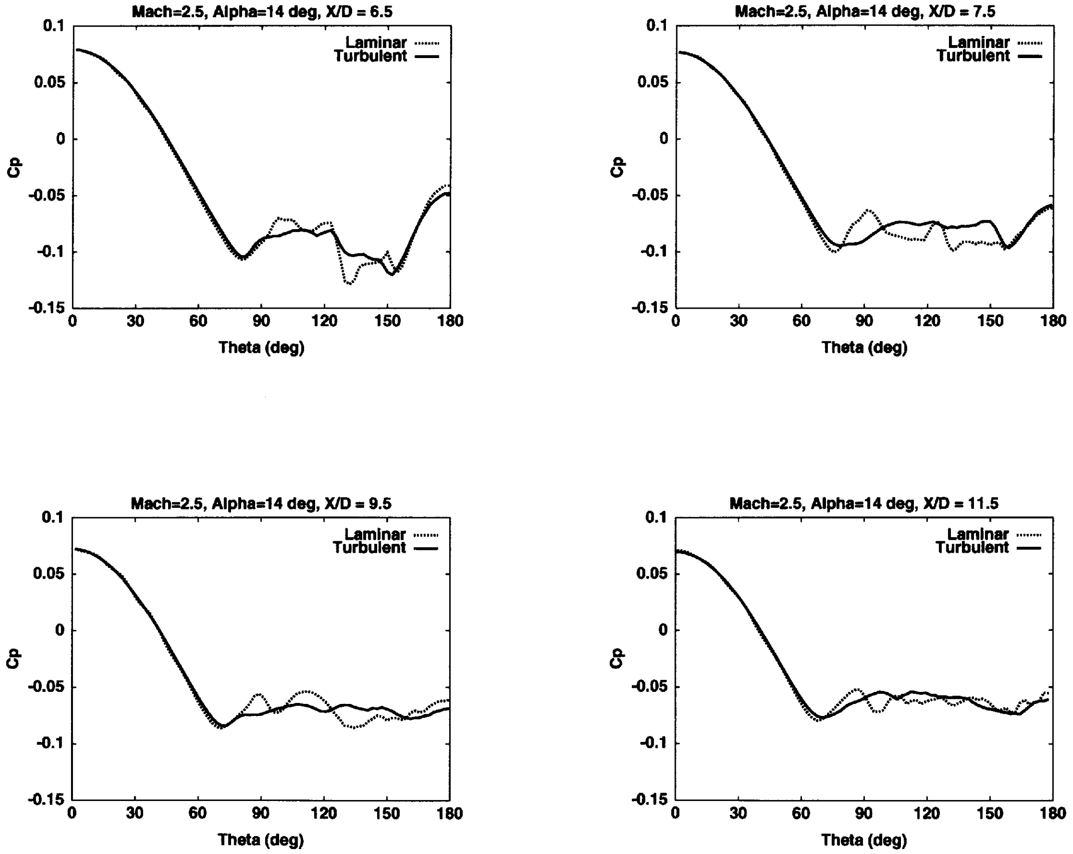

Figure 8. Missile: comparison of computed results on T3D for laminar and turbulent viscous effects for x/d =6·5, 7·5, 9·5 and 11·5

Values for the pressure coefficients at various x/d locations from the CM-5 computations are shown in Figures 9 and 10 comparing laminar and turbulent viscous results. The laminar solution appears to be more turbulent-like than those obtained for the finer mesh T3D results shown in Figures 7 and 8. The turbulent results are only moderately changed from the laminar results. This is an indication that additional effort is needed to evaluate the suitability and accuracy of the turbulence model used.

8. CONCLUDING REMARKS

A computational study has been conducted to evaluate the ability of a finite element compressible Navier–Stokes computational technique to perform computations for high-Reynolds-number flows of interest to Army missile and projectile flow fields. The results for predicting turbulent viscous effects are encouraging; however, additional evaluation is required to reach a meaningful assessment of the capability.

The computational performances achieved on the CM-5 and T3D computers indicate that highly efficient scalable performance has been achieved. This is an important consideration since missile

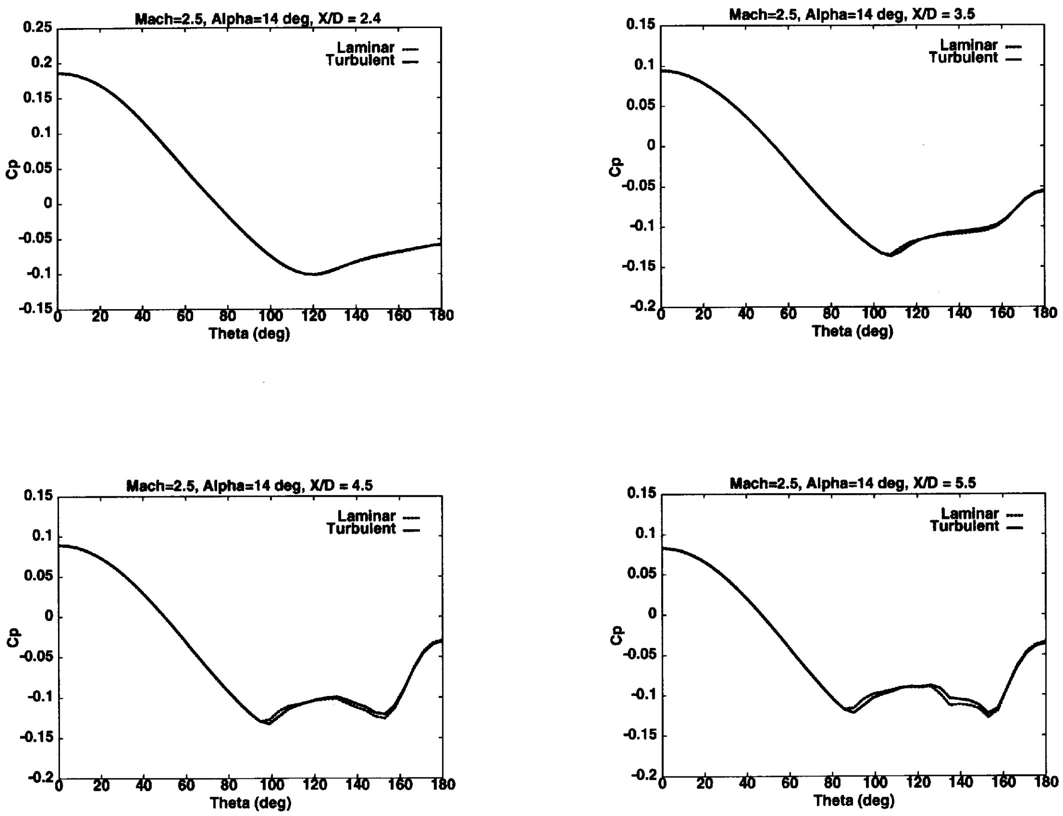

Figure 9. Missile: comparison of computed results on CM-5 for laminar and turbulent viscous effects for $x/d = 2{\cdot}4, 3{\cdot}5, 4{\cdot}5$ and $5{\cdot}5$

configurations currently under development include features such as fins for stability and guidance, reactions jets for guidance, and rocket propulsion. These additional complexities will require substantial computational resources to model accurately and in a timely manner. It appears that only through highly efficient utilization of scalable computers will the capability to adequately address the full scope of these problems be realized.

ACKNOWLEDGEMENTS

This work is sponsored by ARO, ARPA, NASA-JSC and by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory cooperative agreement number DAAH-4-95-2-0003/contract number DAHH04-95-0008. The content does not necessarily reflect the position or the policy of the government and no official endorsement should be inferred. CRAY C90 time and support for the third author were provided in part by the Minnesota Supercomputer Institute.

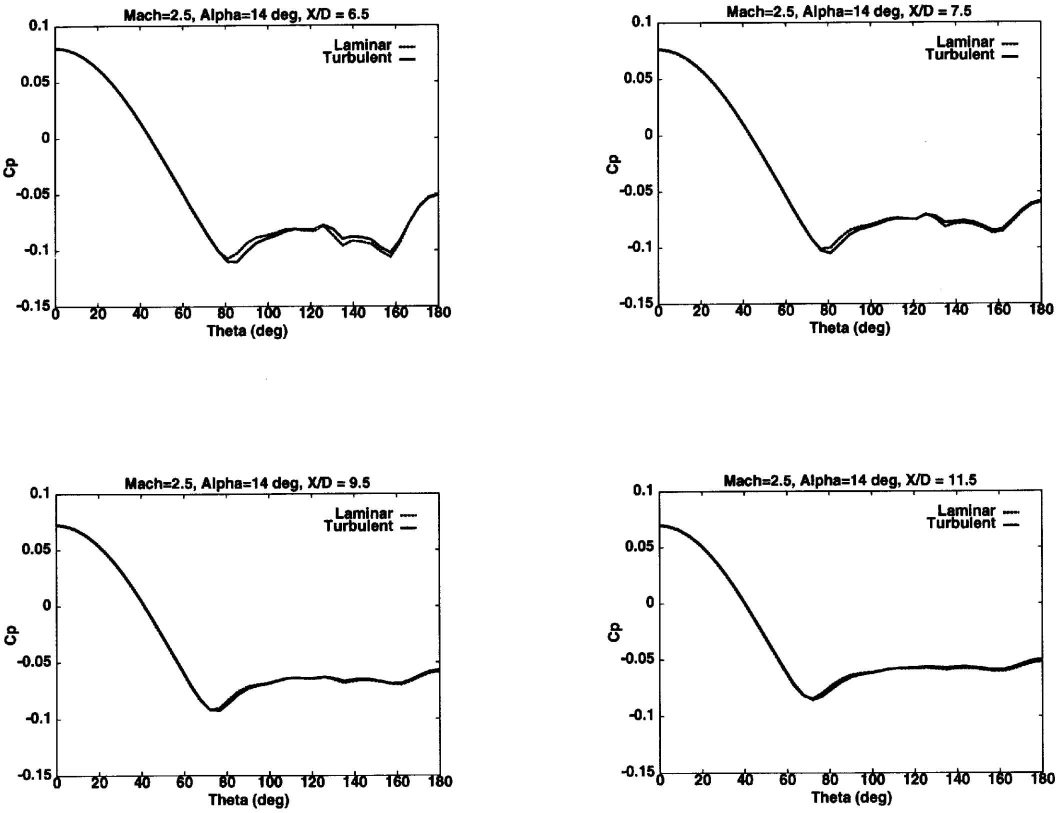

Figure 10. Missile: comparison of computed results on CM-5 for laminar and turbulent viscous effects for $x/d = 6{\cdot}5, 7{\cdot}5, 9{\cdot}5$ and 11·5

REFERENCES

1. S. K. Aliabadi, 'Parallel finite element computations in aerospace applications', *Ph.D. Thesis,* Department of Aerospace Engineering and Mechanics, University of Minnesota, 1994.
2. C. Kato and M. Ikegawa, 'Large eddy simulation of unsteady turbulent wake of a circular cylinder using the finite element method', in I. Celik, T. Kobayashi, K. N. Ghia and J. Kurokawa (eds), *Advances in Numerical Simulation of Turbulent Flows,* FED Vol. 117, ASME, New York, 1991, pp. 49–56.
3. B. E. Launder and D. B. Spalding, 'The numerical computation of turbulent flows', *Comput. Methods Appl. Mech. Eng.*, **3**, 269–289 (1974).
4. F. Chalot and Z. Johan, 'Development of a finite element Navier–Stokes solver with applications to turbulent and hypersonic flows', *AIAA Paper 92-0670,* 1983.
5. B. S. Baldwin and T. J. Barth, 'A one-equation turbulence transport model for high reynolds number wall-bounded flows', *Tech. Rep. 102847,* NASA Ames Research Center, 1990.
6. P. Moin and J. Jimenéz, 'Large eddy simulation of complex turbulent flows', *AIAA Paper 93-3099,* 1993.
7. S. K. Aliabadi and T. E. Tezduyar, 'Parallel fluid dynamics computations in aerospace applications', *Int. J. Numer. Methods Fluids,* **21**, 783–805 (1995).
8. J. G. Kennedy, M. Behr, V. Kalro and T. E. Tezduyar, 'Implementation of implicit finite element methods for incompressible flows on the CM-5', *Comput. Methods Appl. Mech. Eng.,* **119**, 95–111 (1994).
9. T. E. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, V. Kalro and C. Waters, '3D simulation of flow problems with parallel finite element computations on the Cray T3D', in *Computational Mechanics '95, Proc. Int. Conf. on Computational Engineering Science,* Mauna Lani, HI, 1995.
10. T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, V. Kalro and M. Litke, 'High performance computing techniques for flow simulations', in M. Papadrakakis (ed), *Solving Large-Scale Problems in Mechanics: Parallel and Distributed Computer Applications,* Wiley–Interscience, Chichester, 1996.
11. Y. Saad and M. Schultz, 'GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems', *SIAM J. Sci. Stat. Comput.,* **7**, 856–869 (1986).

12. Z. Johan, T. J. R. Hughes and F. Shakib, ‘A globally convergent matrix-free algorithm for implicit time-marching scheme arising in finite element analysis in fluids’, *Comput. Methods Appl. Mech. Eng.*, **87**, 281–304 (1991).
13. T. E. Tezduyar, S. K. Aliabadi, M. Behr and S. Mittal, ‘Massively parallel finite element simulation of compressible and incompressible flows’, *Comput. Methods Appl. Mech. Eng.*, **119**, 157–177 (1994).
14. T. E. Tezduyar, M. Behr and T. J. R. Hughes, ‘High performance finite element computation of fluid dynamics problems’, in M. Hafez and K. Oshima (eds.), *Computational Fluid Dynamics Review 1995*, Wiley, New York, 1995, pp. 390–321.
15. W. B. Sturek, ‘An experimental investigation of the supersonic turbulent boundary layer in a moderate adverse pressure gradient. Part I. A detailed description of the experiment and data tabulation’, *BRL Rep. 1506*, U. S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, MD, 1970.