Comparative Study of Advanced Turbulence Models for Turbomachinery

Ali H. Hadid and Munir M. Sindir

Rocketdyne Division
Boeing North American Inc.
CFD Technology Center
Canoga Park, CA 91309

Contract NAS8-38860
Final Report

Approved by

Glenn Havskjold
Program Manager

October 1996
# Comparative Study of Advanced Turbulence Models for Turbomachinery

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>4</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>5</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td></td>
</tr>
<tr>
<td>1.1 Background</td>
<td>6</td>
</tr>
<tr>
<td>1.2 Outline of Present Study</td>
<td>7</td>
</tr>
<tr>
<td>2. 2D/Axisymmetric $k$-$\varepsilon$ Model</td>
<td></td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>9</td>
</tr>
<tr>
<td>2.2 Theory and Model Equations</td>
<td>10</td>
</tr>
<tr>
<td>2.3 Module Evaluation</td>
<td>15</td>
</tr>
<tr>
<td>References</td>
<td>18</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix A</td>
<td>20</td>
</tr>
<tr>
<td>3. 2D/Axisymmetric Multi-Scale $k$-$\varepsilon$ Model</td>
<td></td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>29</td>
</tr>
<tr>
<td>3.2 Theory and Model Equations</td>
<td>30</td>
</tr>
<tr>
<td>3.3 Module Evaluation</td>
<td>34</td>
</tr>
<tr>
<td>References</td>
<td>36</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix B</td>
<td>37</td>
</tr>
<tr>
<td>4. 2D/Axisymmetric Algebraic Stress Model</td>
<td></td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>48</td>
</tr>
<tr>
<td>4.2 Theory and Model Equations</td>
<td>49</td>
</tr>
</tbody>
</table>
4.3 Module Evaluation

References 52
Figures 53
Appendix C 54

5. 2D/Axisymmetric Reynolds Stress Model 58
5.1 Introduction 59
5.2 Theory and Model Equations 59
5.3 Boundary Conditions 67
5.4 Numerical Procedure 68
5.4.1 Wall Reflection Treatment 71
5.5 Module Evaluation 72
References 74
Figures 76
Appendix D 76

6. 3D $k-\varepsilon$ Turbulence Model 80

6.1 Introduction 81
6.2 Theory and Model Equations 81
6.3 Module Evaluation 82
References 82
Figures 83
Appendix E 83

7. 3D Algebraic Stress Model 90
7.1 Introduction 91
7.2 Theory and Model Equations 91
References 94
Appendix F 95

8. Related Publications and presentations 98
ACKNOWLEDGEMENTS

This work was supported jointly by NASA Marshall Space Flight Center under contract NAS8-38860 and by Rocketdyne Division of Rockwell International under discretionary funds. The authors gratefully acknowledge the efforts of NASA contract monitors Dr. Paul McConnaughey and Mr. Robert Garcia and Rocketdyn program manager Dr. Glenn Havskjold. The authors would also like to acknowledge the efforts of Professor C.-P. Chen and his colleagues Dr. Hong Wei and Ms. X. Li at the University of Alabama at Huntsville for their help in testing the modules. The assistance of Mr. Steve Barson in managing the project and the efforts of Ms. Michelle DeCroix and Mr. Armen Darian of Rocketdyne in testing some of the modules are greatly appreciated.
A computational study has been undertaken to study the performance of advanced phenomenological turbulence models coded in a modular form to describe incompressible turbulent flow behavior in two dimensional/axisymmetric and three dimensional complex geometry. The models include a variety of two equation models (single and multi-scale $k$-$\varepsilon$ models with different near wall treatments) and second moment algebraic and full Reynolds stress closure models. These models were systematically assessed to evaluate their performance in complex flows with rotation, curvature and separation. The models are coded as self contained modules that can be interfaced with a number of flow solvers. These modules are stand alone satellite programs that come with their own formulation, finite-volume discretization scheme, solver and boundary condition implementation. They will take as input (from any generic Navier-Stokes solver) the velocity field, grid (structured H-type grid) and computational domain specification (boundary conditions), and will deliver, depending on the model used, turbulent viscosity, or the components of the Reynolds stress tensor $\overline{u_iu_j}$. There are separate 2D/axisymmetric and/or 3D decks for each module considered.

The modules are tested using Rocketdyn's proprietary code REACT. The code utilizes an efficient solution procedure to solve Navier-Stokes equations in a non-orthogonal body-fitted coordinate system. The differential equations are discretized over a finite-volume grid using a non-staggered variable arrangement and an efficient solution procedure based on the SIMPLE algorithm for the velocity-pressure coupling is used. The modules developed have been interfaced and tested using finite-volume, pressure-correction CFD solvers which are widely used in the CFD community. Other solvers can also be used to test these modules since they are independently structured with their own discretization scheme and solver methodology. Many of these modules have been independently tested by Professor C.P. Chen and his group at the University of Alabama at Huntsville (UAH) by interfacing them with own flow solver (MAST).
CHAPTER 1
Introduction

1.1 Background

Computational Fluid Dynamics (CFD) has been used extensively for the last decade or so in analyzing complex flow phenomenon for many industrial applications, such as combustion and turbomachinery. Most flows of practical interest are turbulent and for many of them, relatively simple prediction methods are sufficient to produce results of engineering accuracy. For others, mainly flows in complex geometry with large body forces such as curvature, rotation and separation, more complex prediction methods are required.

With advancing state-of-the-art of computer technology, the range, size and complexity of flow models being applied have increased. Users have become more sophisticated and there is a constant demand for improvement. CFD codes have adapted to this demand and many general-purpose computer codes have been developed and used. As these general purpose codes become larger, their code structure becomes sophisticated and in general this structure can be divided into three main areas;

1) Numerical algorithms which include discretization methods and solution techniques.
2) Methods of dealing with complex geometry, such as grid generation, structured or unstructured grids.
3) Physical models which include turbulence models, porosity, combustion kinetics, multi-phase flows, etc.

It seems, therefore, that the practicing engineer must have the knowledge of all these elements of the CFD program in order to successfully utilize the code. Modularization of the code structure may then become necessary in order to obtain the maximum benefits from these general-purpose CFD codes. This means developing individual modular routines for the solver and other physical models. If such modules are successful they would allow users to concentrate their talents on developing and improving physical hypothesis such as turbulence models that can be easily tested using these modules.
In general, the physics of turbulence can be captured by solving the full time-dependent Navier-Stokes equations in what is termed as Direct Numerical Simulation (DNS). However, DNS is not practical for engineering purposes mainly because it is restricted to flows at low Reynolds numbers. Large Eddy Simulations (LES) are now competitive with DNS in accuracy at an order of magnitude less cost, however, it is still expensive for routine engineering calculations. Therefore, current engineering prediction methods are based on Reynolds-averaged equations, with models for the unknown Reynolds stresses which appear as the result of time-averaging the nonlinear Navier-Stokes equations. These models fall mainly into three categories; "eddy-viscosity" models, where a relation between the Reynolds stresses and mean velocity gradients at the same point in space is sought. Algebraic stress models, where the Reynolds stresses are expressed as an algebraic relation of turbulence production and dissipation. Reynolds stress models where the exact partial differential equations for the Reynolds stresses are solved after closing the higher order terms. These transport equations account for the dependence of Reynolds stresses on the history of the flow and should perform better than the eddy-viscosity models.

1.2 Outline of the Present Study

In the present work, phenomenological, single-point turbulence models coded in a modular format are developed as self-contained code decks that can be interfaced with a number of flow solvers to analyze turbulent flows in complex 2D/axisymmetric or 3D geometry. These modules are validated using Rocketdyn's REACT code and are independently tested at UAH using own code MAST.

The models that are developed in a modular form include;

1. 2D/axisymmetric single-scale \( k-\varepsilon \) model with three options for near wall treatment that include;
   - Standard Launder and Splading wall functions.
   - Chen and Patel two-layer model.
   - Lam and Bremhorst low-Reynolds number model.

2. 2D/axisymmetric multi-scale \( k-\varepsilon \) model with the standard wall function and Chen & Patel two-layer near wall treatment.

3. 2D/axisymmetric implicit algebraic stress model (ASM) based on the original work of Rodi.

4. 2D/axisymmetric full Reynolds stress turbulence model (RSM) based on the simplified linear
second moment closure model of Launder, Reece and Rodi (LRR) second moment closure.

5. 3D standard $k$-$\varepsilon$ turbulence model with wall function and two-layer near wall treatments.

6. 3D algebraic stress model (ASM).

Each model is coded as a self contained, stand alone module deck that can be interfaced with a number of CFD solvers to analyze turbulent flows in complex geometry. The user can use these modules without concern as to how they are implemented and solved. The input to the modules are the mean flow variables, boundary and geometric information which are to be provided by a mean flow solver. The output of the module are the turbulent eddy-viscosity for the eddy-viscosity models and the Reynolds stresses for the second moment closure models. Moreover, source terms which are needed for the mean flow calculations are calculated and must be passed to the main solver. The modules are tested using the finite-volume REACT code and the results compared with available experimental data.

Full details of each module are given in the next chapters. Chapter 2 discusses the theory and model equations for the two-equation $k$-$\varepsilon$ model used in the 2D/axisymmetric module deck. The module is evaluated with a number of benchmark problems and detailed description of the module variable names together with the input/output structure are given in appendix A. The complete listing of the module is provided at the end of the chapter. Similarly, chapter 3 discusses the theory and model equations for the 2D/axisymmetric multi-time-scale $k$-$\varepsilon$ model. The 2D/axisymmetric Algebraic stress module is presented in chapter 4 and chapter 5 discusses the 2D/axisymmetric Reynolds stress module deck. Full description of the 3D $k$-$\varepsilon$ turbulence model is given in chapter 6 and chapter 7 presents a full description of the 3D algebraic stress model together with module description and code listing in the appendix. Finally in chapter 8, copies of related turbulence work that are presented or published elsewhere are attached.
CHAPTER 2

2D/Axisymmetric $k-\varepsilon$ Turbulence Model

Table of Contents

2.1 Introduction .............................................. 10
2.2 Theory and Model Equations ......................... 10
2.3 Module Evaluation ......................................... 15
References ....................................................... 18
Figures ......................................................... 20
Appendix A ...................................................... 20
2.1 Introduction

In this section a description of the standard k-ε turbulence model that is coded as a self contained computer program to compute turbulent flow quantities in two-dimensional planar or axisymmetric geometry is given. Detailed description of the module structure, variables used and how to interface the module with CFD flow solvers are given in Appendix A. The module has been tested as a separate self-contained unit using the REACT code [1] and was independently tested at the University of Alabama at Huntsville (UAH) using own code (MAST).

2.2 Theory and Model Equations

The k-ε turbulence module is based on the widely used single-scale two equation k-ε turbulence model (k is the turbulent kinetic energy and ε is the energy dissipation rate). The model developed originally by Launder and Spalding [2] was successful in providing good predictions for a wide range of turbulent flows. The k and ε -equations can be derived from the transport equations for the Reynolds stresses assuming fully turbulent flow.

For low-Reynolds number flows close to solid boundaries, adjustments to the model are needed to bridge the viscous dominated sublayer region with the fully turbulent flow region. The success of the wall function method depends on the universality of the turbulent flow structure near the wall. In many complex flows, however, the flow field near the wall has to be determined accurately and the traditional wall-function method is not satisfactory. This is because the specification of all turbulence quantities in terms of the friction velocity fail at separation where the flow near the wall is no longer controlled by the wall shear stress. Patel et al [3] assessed the relative performance of various models which describe the near-wall flows and found that there are still areas of improvements needed to accurately model flow behavior near the wall.

Jones and Launder [4] extended the original k-ε model to the low-Reynolds number form which allowed the calculation to be performed all the way to the wall. Numerical difficulties of accurately resolving the large gradients close to the wall necessitates resolving the wall region with a very fine grid structure. Chen and Patel [5] introduced a method to resolve the near-wall region which combines the standard k-ε model with the one-equation model of Wolfshtein [6] near the wall. In this "two-layer" model an algebraically prescribed eddy-viscosity for the wall region is coupled to the k-ε model to describe the details of the flow in the vicinity of the wall.
Momentum and continuity equations are solved up to the wall and this reduces the physical uncertainties of near-wall turbulence and the numerical difficulties of resolving the very large gradients of turbulence parameters.

For an incompressible, steady and axisymmetric turbulent flow, the Reynolds averaged momentum and continuity equations can be expressed in a generalized form as:

\[
\frac{\partial (\rho u \Phi)}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (\rho u r \Phi) = \frac{\partial}{\partial x} (\Gamma \Phi_x \frac{\partial \Phi}{\partial x}) + \frac{1}{r} \frac{\partial}{\partial r} (r \Gamma \Phi_r \frac{\partial \Phi}{\partial r}) + S_\Phi
\]

where \( \Phi \) is the dependent variable, which stands for \( \Phi = u, v, w \) for the axial, radial and tangential velocities respectively. \( \rho \) is the fluid density, \( \Gamma \Phi_x \) and \( \Gamma \Phi_r \) are exchange coefficients in \( x \) and \( r \)-directions, respectively, and \( S_\Phi \) is the source term for the variable \( \Phi \).

The source terms for the dependent variable are:

- **Axial direction**, \( \Phi = u \), \( \Gamma \Phi_x = 2\mu_e \), \( \Gamma \Phi_r = \mu_e \) and

\[
S_u = \frac{\partial P}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (\mu_e \frac{\partial u}{\partial x})
\]

where \( \mu_e \) is the eddy viscosity and \( P \) is the pressure

- **Radial direction**, \( \Phi = v \), \( \Gamma \Phi_x = \mu_e \), \( \Gamma \Phi_r = 2\mu_e \) and

\[
S_v = -\frac{\partial}{\partial x} \left( \mu_e \frac{\partial u}{\partial x} \right) - 2\mu_e \frac{v}{r^2} + \frac{\rho w^2}{r} - \frac{\partial P}{\partial r}
\]

- **Tangential direction**, \( \Phi = w \), \( \Gamma \Phi_x = \mu_e \), \( \Gamma \Phi_r = \mu_e \) and

\[
S_w = -\frac{\rho v w}{r} - \frac{w}{r^2} \frac{\partial}{\partial r} (r\mu_e)
\]

Equations 2, 3, and 4 above are the momentum equations that are solved by the CFD solvers. However, in order to close the equations and determine the eddy viscosity different turbulence models are used.
The present module utilizes the $k$-$\varepsilon$ model. In this model two equations for the turbulent kinetic energy $k$ and its dissipation $\varepsilon$ which have the same general form as equation (1) are solved.

For the turbulent kinetic energy equation

$$\Phi = k, \quad \Gamma \Phi_x = \Gamma \Phi_r = \mu + \frac{\mu_t}{\sigma_k} \quad \text{and} \quad S \Phi = G - \rho \varepsilon$$  \hspace{1cm} (5)$$

For the energy dissipation equation

$$\Phi = \varepsilon, \quad \Gamma \Phi_x = \Gamma \Phi_r = \mu + \frac{\mu_t}{\sigma_e} \quad \text{and} \quad S \Phi = \frac{\varepsilon}{k} (C_1 f_1 G - C_2 f_2 \rho \varepsilon)$$  \hspace{1cm} (6)$$

where $\sigma_k$ and $\sigma_e$ are turbulent Prandtl/Schmidt numbers for $k$ and $\varepsilon$ respectively, and $G$ denotes the rate of production of the turbulent kinetic energy and is expressed as:

$$G = \mu_e \left\{ 2 \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial r} \right)^2 + \left( \frac{\partial w}{\partial r} \right)^2 \right] + \left( \frac{\partial u}{\partial r} + \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial r} - \frac{w}{r} \right)^2 \right\}$$  \hspace{1cm} (7)$$

where $\mu$ is the dynamic viscosity, and $\mu_t$ is the turbulent viscosity,

$$\mu_t = C_\mu \mu \rho \frac{k^2}{\varepsilon}$$  \hspace{1cm} (8)$$

and $\mu_e = \mu + \mu_t$

$C_\mu, C_1, C_2, \sigma_k$ and $\sigma_e$ are constants whose values are 0.09, 1.44, 1.92, 1.0, 1.0, respectively and $f_1, f_2$ and $f_m$ are damping functions.

Near a wall, turbulent flow can be divided into two regions, the inner viscous sublayer where low turbulent Reynolds number effects are important and the velocities decrease rapidly to zero at the wall, and the outer fully turbulent region. The successful application of the $k$-$\varepsilon$ turbulence model for many complex flows depends to a large extent on how accurately the flow field near the wall is
determined. In the present module three different models are used to treat this thin sublayer region, they include;

Wall function method, where

\[ u^+ = y^+ \quad \text{at } y^+ \leq 11.6 \]  
\[ u^+ = \frac{1}{\kappa} \ln (Ey^+) \quad \text{at } y^+ > 11.6 \]  

where, \( u^+ = \frac{u}{u_T}, \quad y^+ = \frac{u_T y}{v} \) and \( u_T = \sqrt{\tau_w/\rho} \)

\( \tau_w \) is the wall shear stress which can be determined from

\[ \tau_w = \frac{\mu u_p}{\delta} \quad \text{for } y^+ \leq 11.6 \]  
\[ \tau_w = \frac{\kappa C_{\mu}^{0.25} \rho u_p k^{0.5}}{\ln [E C_{\mu}^{0.25} \rho k^{0.5}/\mu]} \quad \text{for } y^+ > 11.6 \]

Here, \( u_p \) denotes the velocity component parallel to the wall at the first grid point \( p \) from the wall.

\( \delta \) is the normal distance from the wall and \( \kappa \) is a constant = 0.42.

In this approach, \( k \) and \( \varepsilon \) equations are solved with \( f_\mu = f_I = f_2 = 1 \), only in the fully turbulent region beyond some distance from the wall. Boundary conditions i.e., velocity components and turbulent parameters at that distance are specified in terms of the friction velocity \( u_T \).

In the low-Reynolds number model, the flow is resolved all the way to the wall with a very fine mesh. Many models have been proposed that are based on the \( k-\varepsilon \) model and differ mainly in the choice of the damping functions \( f_\mu, f_I \) and \( f_2 \) to bridge the gap between the sublayer and the fully turbulent region. The model due to Lam & Bremhorst [7] is used in this work, where:

\[ f_\mu = \left[ 1 - \exp \left( -0.016 R_y \right) \right]^{1/2} \left( 1 + \frac{20.5}{R_t} \right) \]
\[ f_1 = 1 + \left( \frac{0.06}{f_\mu} \right)^3 \quad \text{and} \quad f_2 = 1 - \exp\left( -R_t^2 \right) \]

where, \( R_y = \frac{k^{1/2}}{\nu} \) and \( R_t = \frac{k^2}{\nu \varepsilon} \) are turbulent Reynolds number.

These damping functions tend to unity with increasing distance from the wall.

In the two-layer model due to Chen and Patel [5], a simple algebraically prescribed eddy-viscosity model for the wall region is coupled to the \( k-\varepsilon \) model for the outer flow to describe the flow details. Unlike the low-Reynolds number model that requires the solution of transport equations for both \( k \) and \( \varepsilon \) all the way to the wall, the one-equation model requires the solution of only the turbulent kinetic energy equation in the sublayer region while algebraically specifying the eddy viscosity and energy dissipation.

\[ \nu_t = C_\mu \frac{k^{1/2}}{L_\mu} \quad \text{and} \quad \varepsilon = \frac{k^{3/2}}{L_\varepsilon}. \]

The length scales \( L_\mu \) and \( L_\varepsilon \) contain the necessary damping effects in the near-wall region in terms of the turbulence Reynolds number \( R_y \).

\[ L_\mu = C_1 y \left[ 1 - \exp\left( -R_y/A_\mu \right) \right] \] \hspace{1cm} (13)

\[ L_\varepsilon = C_1 y \left[ 1 - \exp\left( -R_y/A_\varepsilon \right) \right] \] \hspace{1cm} (14)

\( L_\mu \) and \( L_\varepsilon \) become linear and approach \( C_1 y \) with increasing distance from the wall.

\( C_1 = \kappa C_\mu^{0.75} \) and \( A_\varepsilon = 2C_1 \). Chen and Patel [5] used \( A_\mu = 70 \).

The damping effects decay rapidly with distance from the wall independent of the magnitude of the wall shear stress. The matching between the one-equation and the standard \( k-\varepsilon \) models is carried along prescribed grid lines where \( R_y \approx 200 \).

For flows in rotating ducts a modification was made by Chen and Guo [8] to reflect the effects of a system rotation on the length scales \( L_\mu \) and \( L_\varepsilon \), as;
Moreover, the function \( f_2 \) in the dissipation equation is modified to

\[
f_2 = f_2 + Ri
\]

where \( Ri \) is a Richardson number to reflect the effects of streamline curvature due to rotation and is defined as

\[
R_i = (0.4 \omega_k - 0.8 \Omega_k) \Omega_k \left( \frac{k}{\varepsilon} \right)^2
\]

where \( \omega_k = \varepsilon_{ijk} \frac{\partial U_i}{\partial x_j} \) is the local mean vorticity.

The above modification to account for streamline curvature and rotation seemed adequate in the framework of two equation k-\( \varepsilon \) modeling. Other modifications have also been considered but not implemented in this module and can be referred to in Hadid and Sindir [9].

### 2.3 Module Evaluation

The single scale k-\( \varepsilon \) turbulent module was evaluated by comparison with published experimental data. One of the test problems considered is the two dimensional incompressible turbulent flow over a backward facing step with and without rotation (see figure 1) to compare with the experiment of Rothe and Johnston[10]. While the mean flow is in the \( x-y \) plane, the channel is rotated with constant angular velocity \( \Omega \) about the \( z \)-axis. The ratio of the channel width to the step height is very large so that the secondary flow can be ignored, which made the flow remain two dimensional. The channel height to step ratio was set to 2 and the inlet channel height (\( h \)) equals to the step height (\( H \)). The Reynolds number based on the uniform inlet velocity was about 5500. The rotation number (\( Ro = \Omega h/U \)) was varied between +0.06 and -0.06.

The streamline patterns for the three different rotation numbers \( Ro = -0.06, 0.0, +0.06 \) by using the three different wall treatments are shown in figures 2-4. In each figure, the upper (a) and lower (c) parts correspond to \( Ro = +0.06 \) and \( Ro = -0.06 \) respectively. While the middle part (b) is the non-rotating case. It is observed that the streamline patterns are influenced by the system rotation. Suction side step extends the recirculation zone and the pressure side step reduces the recirculation zone. The reattachment length for \( Ro = -0.06 \) using the wall functions is larger compared to the
experimental results. This is due to the fact that no Coriolis effect is accounted for in the law of the wall. The predicted variation of reattachment length with $Ro$ (figure 5) shows reasonable correlation with the experimental data of Rothe and Johnston [10].

The single scale $k-\varepsilon$ model using three different wall treatments with rotational stress generation terms embodied seems to capture the main effects of system rotation on turbulence structure, i.e. the suppression of turbulence level with clockwise rotation and enhancement of turbulence level with counterclockwise rotation. The effects are also noticeable in the corresponding increase in the reattachment length with clockwise rotation and its decrease with counterclockwise rotation.

The other two test cases were those of Daily and Nece [11] where rotating disk cavity circulation and secondary flows are induced by a rotating wall, and Roback and Johnson [12] for a confined double concentric jets with a sudden expansion. Flow swirl in this case is induced by imposing a tangential velocity component at the outer jet. Figure (6) shows the two-dimensional axisymmetric rotating lid cavity of Daily and Nece. The flow is bounded by a disk (rotor) and a stationary end wall (stator) of a chamber. The ratio of the axial clearance between the rotor and the stator ($s$) to the radius of the disk ($a$) is 0.0255. The disk rotates with a rotational Reynolds number $R=4.4\times10^6$ defined as $R = \Omega a^2/v$, where $\Omega$ is the disk rotational speed and $v$ is the kinematic viscosity.

Computations were performed on a 33x75 grid with different grid clustering near the walls for the different near-wall models. Figure (7) shows the velocity vectors at the top region of the cavity using the wall function model. Centrifugal forces move the fluid radially outward on the disk, axially away from the disk on the wall casing, and radially inwards on the stationary end wall. Figure 8, shows the axial variations of the radial velocity component at a radial position $r/a=0.765$. The agreement is fair with some discrepancy for all near-wall models close to the rotating disk. Figure (9), shows the axial variation of the tangential velocity component at the radial position $r/a=0.765$. At the rotating disk ($x=0$), the tangential velocity approaches the value ($a\Omega$). The two-layer near wall model seem to offer closer agreement with the data than the other two models. The presence of corner regions presents a difficulty in defining the normal distances used in the definition of turbulent Reynolds number. In the present analysis, values of the normal distance were based on the normal distance to the nearest solid boundary.

Predictions of the experiments of Roback and Johnson [12] have been presented by several workers, e.g. Sloan et al. [13] and Durst and Wennerberg [14]. Unfortunately, inlet flow profiles were not provided in the experiment. Therefore, the present calculations were started at the
expansion plane using the measured velocity profile at 5 mm downstream of the expansion after some adjustments near the edges of the coaxial jets. Measurements of main turbulent intensities were used to calculate inlet values of the turbulent kinetic energy. Energy dissipation rate was estimated from $\varepsilon = C_L \mu^{2/3}/L$, where $L$ is a length scale of turbulence at the inlet of the order of $10^{-4}$ m.

Figure 10, shows an illustration of the test chamber geometry. The chamber diameter is about twice the secondary tube diameter. The exit from the 8-bladed, 30°, free vortex swirl generator is located approximately 0.005 m upstream from the confluence plane.

A prominent phenomenon in axisymmetric swirling flows in such geometry is the "bubble" or vortex breakdown which has been studied extensively [15-18]. In the present numerical simulation of the experiment, a 150x100 grid nodes was used with different clustering on the walls for the different near-wall models used. Figure 11, shows the velocity vectors indicating the presence of a closed recirculation zone at the center with additional zones at the corner downstream and between the inner jet and the outward diverted secondary jet. The figure also shows flow diversion outwards with high gradients characterized by large turbulent shear and fluctuation levels. Comparisons were made of the radial variations of flow variables at two axial locations, $x=0.025m$ upstream of the vortex bubble and $x=0.102m$ located inside the bubble. Figure (12), shows the radial variation of the axial velocity profile at $x=0.025m$ using the wall function, two-layer and the low Reynolds number models. Fair agreement by the different models is shown. They also seem to predict small negative velocities at a radial position $r=0.015m$ (the interface between the inner and outer jets), slightly under predicted in strength and width. Figure (13) shows the radial variation of the axial velocity profiles at $x=0.102m$. The two-layer model shows a better agreement with the experimental data.

Radial variations of the tangential velocity at $x=0.025m$ is shown in figures 14. The figure shows that the two-layer model offers better agreement with the experiment as compared with the wall function or the low Reynolds number models.

In general, the calculations shown above indicate that the two layer model seem to offer a better comparisons with the experimental results. The three near-wall models are built in the standard two-dimensional/axisymmetric $k-\varepsilon$ turbulence module. The structure of the module will be discussed next together with the details of interfacing with a flow solver and descriptions of variables.
REFERENCES


11. Daily, J. W. and Nece, R. E. "Chamber Dimension Effects on Induced Flow and Frictional


Figure 1. Rotating backward facing step

Figure 2. Stream-function contours using wall function near wall treatment
Figure 3. Stream-function contours using the two-layer near-wall treatment
Figure 4. Stream-function contours using the low-Reynolds number near-wall treatment
Figure 5. Reattachment length as a function of the rotation number
Figure 6. Rotating lid cavity

Figure 7. Velocity vectors ($Re = 4.4 \times 10^6$)
Figure 8. Axial distribution of radial velocity at $r/a = 0.765$

Figure 9. Axial distribution of tangential velocity at $r/a = 0.765$
Figure 10. Roback & Johnson's swirling coaxial jets discharging into an expanded duct

Figure 11. Velocity vectors
APPENDIX A

2D/Axisymmetric $k$-$\varepsilon$ Turbulence Module Deck

A.1 Introduction

In an attempt to modularize the $k$-$\varepsilon$ turbulent physical model - a difficult task as many CFD users may know. A self-contained, stand-alone turbulence module has been constructed that computes turbulent flow quantities using the standard $k$-$\varepsilon$ turbulence model. The module is structured to be flexible with options for three near-wall treatments. It can be easily accessed by the user and interfaced with own CFD solvers to calculate turbulent flows.

It is hoped that the program is sufficiently “full proof” and user friendly. However, care must be exercised to identify the limitations of the module to be compatible with the flow solver. Module capabilities and input/output structure is described next in details followed by a FORTRAN listing of the module.

A.2 Program KEMOD

This is basically the solver for the $k$ and $\varepsilon$ - transport equations. It reads through its argument list different variables from the calling flow solver. These variables are described below where, each variable name ends with either an (I) for Integer variable, (R) for Real variable or (L) for Logical variable.

The flow chart of the program is shown in Figure A.1. It shows the main operations performed by the code.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIMI</td>
<td>Number of cell nodes in the I- or $\xi$-coordinate lines. (input from flow solver)</td>
</tr>
<tr>
<td>NJMI</td>
<td>Number of cell nodes in the J- or $\eta$-coordinate lines. (input from flow solver)</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>XR</td>
<td>Grid node locations in the $x$ or $\xi$-direction, dimensioned to XR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>YR</td>
<td>Grid node locations in the $y$ or $\eta$-direction, dimensioned to YR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>UR</td>
<td>Axial or $x$-direction velocity ($u$), dimensioned as UR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>VR</td>
<td>Radial or $y$-direction velocity ($v$), also dimensional as VR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>WR</td>
<td>Azimuthal velocity ($w$), dimensional WR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>TER</td>
<td>Turbulence kinetic energy $k$, dimensioned TER (NX,NY) (calculated in KEMOD and returned to flow solver)</td>
</tr>
<tr>
<td>EDR</td>
<td>Turbulent energy dissipation rate $\epsilon$, dimensioned EDR (NX,NY) (calculated in KEMOD and returned to flow solver)</td>
</tr>
<tr>
<td>URFKR</td>
<td>Under-relaxation factor for $k$-equation (input from flow solver)</td>
</tr>
<tr>
<td>URFER</td>
<td>Under-relaxation factor for $\epsilon$-equation (input from flow solver)</td>
</tr>
<tr>
<td>PRTKR</td>
<td>Prandtl/Schmidt number for turbulent energy-equation, assumed known (input from flow solver)</td>
</tr>
<tr>
<td>PRTER</td>
<td>Prandtl/Schmidt number for turbulent energy dissipation equation, assumed known (input from flow solver)</td>
</tr>
</tbody>
</table>
| GR     | $= 1.0$ if second order upwinding is desired  
$= 0.0$ if first order upwinding is used  
(input from flow solver. Usually calculation of $k$ and $\epsilon$ are not very sensitive to the order of upwinding used) |
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1R</td>
<td>Mass flux variable at cell faces in x- or $\xi$-direction, dimensioned $F1R$ (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>F2R</td>
<td>Mass flux variable at cell faces in y or $\eta$-direction, dimensioned $F2R$ (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>ITERI</td>
<td>Iteration number (input from flow solver)</td>
</tr>
<tr>
<td>VISCOSR</td>
<td>Dynamic viscosity (input from flow solver)</td>
</tr>
<tr>
<td>VISR</td>
<td>Eddy viscosity, dimensioned $VISR$ (NX,NY) (calculated in KEMOD and returned to main solver)</td>
</tr>
<tr>
<td>AKSIL</td>
<td>Logical variable for axisymmetric geometry ($AKSIL=\cdot$TRUE$\cdot$) or plain geometry ($AKSIL=\cdot$FALSE$\cdot$) (input from flow solver)</td>
</tr>
<tr>
<td>LREL</td>
<td>Logical variable for Lam &amp; Bremhorst's low-Reynolds number model ($LREL=\cdot$TRUE$\cdot$) or others ($LREL=\cdot$FALSE$\cdot$) (input from flow solver)</td>
</tr>
<tr>
<td>LAY2L</td>
<td>Logical variable for Patel's two-layer model if ($LAY2L=\cdot$TRUE$\cdot$) or others ($LAY2L = \cdot$FALSE$\cdot$) (input from flow solver)</td>
</tr>
<tr>
<td>C1R</td>
<td>Turbulence model constant, $C_1$ (input from flow solver)</td>
</tr>
<tr>
<td>C2R</td>
<td>Turbulence model constant, $C_2$ (input from flow solver)</td>
</tr>
<tr>
<td>CMUR</td>
<td>Turbulence model constant, $C_\mu$ (input from flow solver)</td>
</tr>
<tr>
<td>I2LWI</td>
<td>Grid line location from the west wall in the x-direction for the two-layer model (input from flow solver)</td>
</tr>
<tr>
<td>I2LEI</td>
<td>Grid line location from the east wall in the x-direction for the two-layer model (input from flow solver)</td>
</tr>
<tr>
<td>J2LSI</td>
<td>Grid line location from the south wall in the y-direction for the two-layer model (input from flow solver)</td>
</tr>
</tbody>
</table>
Grid line location from the north wall in the y-direction for the two-layer model (input from flow solver)

Boundary condition flag along east boundary must have one for each boundary node set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall e.g., for an outlet boundary condition on the east boundary set JTBEI to NJ*2, and similarly for other boundaries, dimensioned JTBEI (NY) (input from flow solver)

Boundary condition flag along west boundary, dimensioned JTBWI (NY) (input from flow solver)

Boundary condition flag along north boundary, dimensioned ITBNI (NX) (input from flow solver)

Boundary condition flag along south boundary, dimensioned ITBSI (NY) (input from flow solver)

Program KEMOD is interfaced with the main flow solver by a call to KEMOD with its arguments. For iterative flow solvers KEMOD is called within the iteration sequence after the solution of the momentum equations where the mean velocities are passed to KEMOD. There are different flow solvers utilizing different schemes from staggered to nonstaggered grid arrangement and for nonorthogonal coordinate system there are at least three alternatives to the choice of the velocity components

i. Cartesian velocity components

ii. Contravariant velocity components

iii. Covariant velocity components

The Cartesian velocity components are the most widely used and have the advantage of simple formulation of the governing equations. Whatever the arrangement used, mass fluxes at cell faces are required and passed to KEMOD as F1R and F2R in both directions. The location of other variables such as k and ε are at the cell center or cell nodes.
The module starts by reassigning variable names passed to it from flow solver to names that are shared with the different subroutines of the module in a common statement file "KEMOD•COMMON". Then a check is made if it is the first iteration in which case the grid file "GRIDF" is called -after passing the grid node locations XR & YR in KEMOD- in order to calculate grid related quantities which will be explained later. The need to call GRIDG can be waived if all the grid data are passed to the module. That is all the information about the grid such as interpolation factors FX and FY, cell areas (ARE) and volumes (VOL) and normal distances of first grid point from grid boundaries (DNS from south boundary, DNN - from north boundary, DNW - from west boundary and DNE - from east boundary).

After this a call to subroutine CALCE is made to calculate the turbulent kinetic energy $k$ (with the identifier IPHI=1) followed by a check if the low-Reynolds number model or the two-layer model are to be used in which case subroutine TWOLAY is called. The energy dissipation equation is solved next by a call to subroutine CALCE again with the identifier IPHI=2. The turbulent viscosity is updated next by calling subroutine MODVIS. A brief description of each subroutine is given next.

**A.3 Subroutines**

**GRIDG**

Before calling this subroutine, the coordinates of all grid nodes, defined in reference to a fixed Cartesian coordinate frame are read. Figure A.2 shows the position of cell and grid nodes.

This subroutine is called only once to calculate coordinates of grid nodes (intersection of grid lines) and geometrical properties of the grid (cell areas and volumes, interpolation factors, normal distances of near-boundary cell nodes from boundary). All variables including grid node coordinates are converted to one-dimensional arrays. These are formed by scanning the grid in J-direction (figure A.2) for I=1, and then repeating for all I’s. The position of any node in one-dimensional array is therefore defined as;

$$IJ = (I,J) = (I-1) \times NJ + J$$

The actual number of grid nodes is one row and one column less than for all cell nodes. For $I = NI$ and $J = NJ$ fictitious grid nodes are introduced which have the same coordinates as actual nodes on NI-1 in I-direction and NJ-1 in J-direction.

- 24 -
The subroutine then calculates interpolation factors which are associated with cell nodes and are used in the main program to calculate values of dependent variables at locations other than cell nodes (cell centers). Definition of these are given in Figure A.3. Cell areas and volumes are calculated next followed by calculations of normal distances of near-boundary nodes from all four outer boundaries.

**CALCE (PHI, IPHI)**

This subroutine solves the linearized and discretized transport equations for the turbulent energy \( k \) and the energy dissipation rate \( \varepsilon \). The two dummy parameters in the calling statement, PHI and IPHI, represent arrays containing dependent variables for which the equation is to be solved. The subroutine sets up the convective and diffusive coefficients over the entire field. Then it calculates the source terms for either \( k \) or \( \varepsilon \) transport equations. A call is made to entry MODPHI in order to modify these sources and boundary coefficients to suit the particular problem. Moreover, a check is made if the two-layer model is selected then the energy dissipation is set algebraically in the sublayer region.

The discretized equations have the form

\[
A_p \Phi_p = \sum_{i=E,W,N,S} A_i \Phi_i + S_\Phi
\]

where the coefficients \( A_i \) (\( i=E,W,N,S \) see figure A.3) contain both the convective and diffusive fluxes. These equations are assembled and solved by calling subroutine SOLSIP which is based on Stone's Strongly Implicit Solver [19].

**TWOLAY**

This subroutine is called if the two-layer or low-Reynolds number models are used. It calculates the different coefficients needed to describe the energy dissipation and eddy viscosity. In this
subroutine the normal distances used in the definition of the turbulent Reynolds number $R_y$ at corner regions are calculated based on the normal distance nearest to the solid boundary.

**SOLSIP**

This subroutine solves the system of linear algebraic equations for $k$ and $\varepsilon$ using Stone's Implicit Procedure [19]. The array RES (IJ) is used to store residuals. The sum of absolute residuals "RESORP" calculated in the first pass through this part of the routine is used as a measure of convergence of the solution process as a whole and this value is stored in RESOR (IPHI). This variable RESOR (IPHI) is passed to the main solver and if desired can be normalized and compared with the maximum error allowed there. If necessary, inner iterations counter $L$ and the sum of absolute residuals RESORP are printed out to monitor the rate of convergence of $k$ and $\varepsilon$ solution. If the ratio RSM is greater than the maximum allowed for the variable in question, SOR (IPHI), and the number of inner iterations is smaller than a prescribed maximum, NSWP (IPHI), then the routine repeats the sequence of calculating the residuals, increment vectors and updating the dependent variable.

**USERM**

This subroutine has different ENTRY points or sections where variables are updated and boundary conditions are set.

**Section MODVIS**

This section calculates effective viscosity (Eq. 8). It is called after calculating $k$ and $\varepsilon$. At locations where $\varepsilon$ is close to zero (i.e., $\leq 10^{-30}$) viscosity is set to zero. A provision is made for under relaxing changes in effective viscosity which may help to stabilize oscillations and improve convergence rate.

**Section MODPHI**

This section is called from CALCE subroutine and sets the boundary conditions for $k$ and $\varepsilon$ depending on which variable being called (IDIR = 1 for $k$ and IDIR = 2 for $\varepsilon$). For the $k$ -equation, the south boundary is checked first if it is one of four options:
(1) An inflow boundary ITBS(I) = 1, where the source term is set to accept the inlet values at J = 1 (south boundary).

(2) Outflow boundary ITBS(I) = 2, where zero gradient in y or η-direction is employed.

(3) Symmetry boundary, TBS(I) = 3, where gradients normal to symmetry plane are zero.

(4) Wall boundary, ITBS(I) = 4, where the production term GENTS(I) calculated from subroutine WALLFN in program MODIFY is added to the rest of the source term SU(IJ).

Boundary conditions for the ε-equation are similar to those of k except at the wall where they are set to appropriate values for each near wall treatment.

A.4 Program MODIFY

This program is compiled separately and is called from the u and v solver routines. It basically updates the flux source term of the discretized momentum equation due to wall shear stresses. If the u-momentum equation for example is discretized in the form

\[ A_p u_p = \sum_{i=E,W,N,S} A_i u_i + S_u^* \]

where P, E, W, N, S are cell nodes as shown in Figure A.3, and \( A_p^* \) and \( A_i \) contain convective and diffusive coefficients. \( S_u^* \) is the source term containing pressure gradients and cross-derivative diffusion terms and convective terms for second-order upwinding scheme. This source term is usually linearized as \( S_u^* = S_u - B_p u_p \). The term \( B_p \) is usually moved to the left hand side of the equation and modifies the diagonal coefficient \( A_p = A_p^* + B_p \), and the equation can be written as

\[ A_p u_p = \sum_{i=E,W,N,S} A_i u_i + S_u \]
Then $S_u$ and $B_p$ are passed to subroutine MODIFY where they are modified if a wall is present (e.g., ITBS(I) = 4 for south boundary).

For an iterative flow solver using the finite-volume methodology. A typical interface and call to the $k$-$\varepsilon$ module from the main flow solver can be represented by a flow chart as shown in figure A.4.
Figure A.1 2D/axisymmetric $k$-$\varepsilon$ module deck flow chart
Figure A.2 Position of cell and grid nodes
Figure A.3 Definition of the interpolation factors

\[ F_{Xp} = \frac{\overline{Pe}}{\overline{Pe} + eE}, \quad F_{Yp} = \frac{\overline{Pn}}{\overline{Pn} + nN} \]
Figure A.4 Typical main flow solver with calls to the 2D/axisymmetric $k$-$\varepsilon$ module
1 C 2D AXISYMMETRIC SINGLE-SCALE K-E TURBULENCE MODULE C 2 C Rocketdyne CFD Technology Center C 3 C 4 C 5 C 6 C 7 C 8 C 9 C Single Scale 2-Equation with 3 Wall Treatments 10 C 1.) Wall Function 2.) Two Layer 3.) Low Reynolds Number 11 C 12 C 13 C SUBROUTINE KEMOD (NIMI, NJMI, XR, UR, VR, WR, TSR, EDR, 14 & URFER, URFER, PRTK, PRTK, FZ, FZ, FZ, FZ, VISCR, VISCR, 15 & AKSIL, LNDR, LAZ2, C, C, CMUR, A2LWI, L2LEI, J2LSEI, 16 & J2LMEJ, J2BMEI, J2BMHI, J2BHI, J2SBI) 17 C 18 C ------------ INCLUDE 'kemod.h' C 19 C 20 C DIMENSION XR(NX, NY), VR(NX, NY), VR(NX, NY), VR(NX, NY), 21 & VR(NX, NY), VR(NX, NY), EDR(NX, NY), FZ(NX, NY), FZ(NX, NY), 22 & VISCR(NX, NY), J2BEI(NX), J2BEI(NX), J2BEI(NX), J2BEI(NX), 23 C 24 C COMMON/GR/ X(NX, NY), Y(NX, NY) C 25 C 26 C DATA GREAT,SMAAL/1.E30, 1.E-30/ C 27 C DATA MAF, QTR/0.5, 0.25/ C 28 C DATA SOB/0.1, 0.1/ C 29 C DATA NONT/10, 10/ C 30 C DATA WOWN/0.0/ C 31 C 32 C C -- EQUATE VARIABLES IN ARGUMENT LIST TO THOSE IN COMMON BLOCK C 33 C 34 C N1=N1MI 35 C N1=N1MI 36 C N1=N1MI 37 C N1=N1MI 38 C N1=N1MI 39 C URFER=URFER 40 C URFER=URFER 41 C PRTK=PRTK 42 C PRTK=PRTK 43 C 44 C 45 C 46 C 47 C 48 C C 49 C C -- LOGICAL VARIABLES C 50 C 51 C 52 C AKSIL=AKSIL 53 C 54 C 55 C C -- TWO-LAYER TURBULENT PARAMETERS C 56 C 57 C 58 C 59 C 60 C 61 C 62 C 63 C 64 C 65 C CONTINUE C 66 C 67 C C -- BOUNDARY CONDITION IDENTIFIERS C 68 C 69 C 70 C 71 C 72 C 73 C 74 C 75 C 76 C 77 C 78 C 79 C 80 C 81 C 82 C 83 C 84 C 85 C 86 C 87 C 88 C 89 C 90 C 91 C 92 C 93 C 94 C 95 C 96 C C -- CALCULATE GRID GEOMETRIC VARIABLES INITIALLY C 97 C 98 C 99 C IF (ITER.LE.1) THEN C 100 C CALL GRIDC C 101 C ENDIF C 102 C C C C -- CALL KINETIC ENERGY SOLVER C 103 C 104 C 105 C C -- CALL KINETIC ENERGY SOLVER C 106 C 107 C 108 C C -- 2-LAYER OR LOW-RE. MODELS C 109 C IF(LAY2.OR.LRE) CALL TWOLAY C 110 C C -- CALL ENERGY DISSIPATION SOLVER C 111 C CALL CALCE (ED2) C 112 C C 113 C C -- UPDATE AND CALCULATE EDDY VISCOISITY C 114 C CALL NODVIS C 115 C 116 C 117 C 118 C 119 C 120 C 121 C 122 C 123 C 124 C 125 C 126 C 127 C 128 C 129 C C -- SUBROUTINE GRID C 130 C 131 C 132 C 133 C 134 C 135 C 136 C 137 C 138 C 139 C 140 C 141 C 142 C 143 C
<table>
<thead>
<tr>
<th>Oct 12 1996 16:41</th>
<th>ssksmod_2d</th>
<th>Page 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>143</td>
<td>J=N+1M</td>
<td></td>
</tr>
<tr>
<td>144</td>
<td>X(1,J+1)=X(I,J)</td>
<td></td>
</tr>
<tr>
<td>145</td>
<td>Y(I,J+1)=Y(I,J)</td>
<td></td>
</tr>
<tr>
<td>146</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>147</td>
<td>DO 4 J=1,NJ</td>
<td></td>
</tr>
<tr>
<td>148</td>
<td>I=N-1M</td>
<td></td>
</tr>
<tr>
<td>149</td>
<td>X(I-1,J)=X(I,J)</td>
<td></td>
</tr>
<tr>
<td>150</td>
<td>Y(I,J-1)=Y(I,J)</td>
<td></td>
</tr>
<tr>
<td>151</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>152 C</td>
<td>C.... GRID ORIGIN AT X=0, Y=0</td>
<td></td>
</tr>
<tr>
<td>153</td>
<td>DO 5 I=1,NI</td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>DO 5 J=1,NJ</td>
<td></td>
</tr>
<tr>
<td>155</td>
<td>IJ=(I-1)+NJE</td>
<td></td>
</tr>
<tr>
<td>156</td>
<td>XX(I,J)=X(I,J)</td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>YY(I,J)=Y(I,J)</td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>-----CALCULATION OF INTERPOLATION FACTORS</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>DO 6 IJ=1,NINJ</td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>FX(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>FY(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>DO 7 J=2,NJM</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>IJ=1</td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>FY(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>MX=NXI</td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>MX=NXJ</td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>DO 8 I=2,NLX</td>
<td></td>
</tr>
<tr>
<td>171</td>
<td>IJ=IMNI</td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>IPJ=LI</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>LPJ=LI</td>
<td></td>
</tr>
<tr>
<td>174</td>
<td>CJM=1-1</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>IM=1</td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>IM=1-1</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>DX=0.5*(XX(IJ)-XX(IJ+1)-XX(IJ+1)-XX(IJ-1))</td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>DY=0.5*(XX(IJ)-XX(IJ+1)+XX(IJ+1)-XX(IJ-1))</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>DY=0.5*(YY(IJ)-YY(IJ+1)+YY(IJ+1)-YY(IJ-1))</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>DX=0.5*(YY(IJ)-YY(IJ)+YY(IJ+1)-YY(IJ-1))</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>DX=SQRT(DX<em>2+DY</em>2)</td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>FX(IJ)=DEP(DJ+DK)</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>IJ=1</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>FD(IJ)=1.0</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>DO 9 I=1,NL</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>IJ=IMNJ</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>FX(IJ)=FX(IJ-1)</td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>FX(IJ)=FX(IJ+1)</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>DO 10 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>IJ=IMNI+1</td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>FY(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>196</td>
<td>DO 11 J=2,LNJ</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>IJ=IMNJ</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>DO 12 J=1,LNJ</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>IJ=IMNJ</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>DX=0.5*(XX(IJ)+XX(IJ)+XX(IJ)+XX(IJ))</td>
<td></td>
</tr>
<tr>
<td>206</td>
<td>DY=0.5*(YY(IJ)+YY(IJ)+YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>FY=0.5*(YY(IJ)+YY(IJ)+YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>208</td>
<td>DY=0.5*(YY(IJ)+YY(IJ)+YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>DX=SQRT(DX<em>2+DY</em>2)</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>DEP=DEP(DF+DP*2)</td>
<td></td>
</tr>
<tr>
<td>211</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>212</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>213</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Oct 12 1996 16:41</th>
<th>ssksmod_2d</th>
<th>Page 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>214</td>
<td>FY(IJ)=1.0</td>
<td></td>
</tr>
<tr>
<td>215</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>216</td>
<td>DO 13 I=1,NI</td>
<td></td>
</tr>
<tr>
<td>217</td>
<td>FY(IJ)+FY(IJ+1)</td>
<td></td>
</tr>
<tr>
<td>218</td>
<td>IJ=IMNJ+1</td>
<td></td>
</tr>
<tr>
<td>219</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>220</td>
<td>FY(IJ)=FY(IJ-1)</td>
<td></td>
</tr>
<tr>
<td>221</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>222</td>
<td>-----CALCULATION OF CELL AREAS</td>
<td></td>
</tr>
<tr>
<td>223</td>
<td>DO 14 I=1,NI</td>
<td></td>
</tr>
<tr>
<td>224</td>
<td>ARE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>225</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>226</td>
<td>DO 15 I=1,NI</td>
<td></td>
</tr>
<tr>
<td>227</td>
<td>ARE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>228</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>229</td>
<td>ARE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>230</td>
<td>ARE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>231</td>
<td>ARE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>232</td>
<td>ARE(IJ)=0.5*ARE(IJ)</td>
<td></td>
</tr>
<tr>
<td>233</td>
<td>ARE(IJ)=0.5*ARE(IJ)</td>
<td></td>
</tr>
<tr>
<td>234</td>
<td>ARE(IJ)=0.5*ARE(IJ)</td>
<td></td>
</tr>
<tr>
<td>235</td>
<td>ARE(IJ)=0.5*ARE(IJ)</td>
<td></td>
</tr>
<tr>
<td>236</td>
<td>ARE(IJ)=0.5*ARE(IJ)</td>
<td></td>
</tr>
<tr>
<td>237</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>238</td>
<td>-----NORMAL DISTANCE FROM THE WALL</td>
<td></td>
</tr>
<tr>
<td>239</td>
<td>DO 16 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>240</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>241</td>
<td>DO 17 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>242</td>
<td>IM=1</td>
<td></td>
</tr>
<tr>
<td>243</td>
<td>IM=1-1</td>
<td></td>
</tr>
<tr>
<td>244</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>245</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>246</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>247</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>248</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>249</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>251</td>
<td>DDB=0.25*(XX(IJ)+XX(IJ)-XX(IJ)+XX(IJ))</td>
<td></td>
</tr>
<tr>
<td>252</td>
<td>DDBP=0.25*(XX(IJ)+XX(IJ)-XX(IJ)+XX(IJ))</td>
<td></td>
</tr>
<tr>
<td>253</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>254</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>255</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>257</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>258</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>259</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>260</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>261</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>262</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>263</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>264</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>265</td>
<td>DDBP=0.25*(YY(IJ)+YY(IJ)-YY(IJ)+YY(IJ))</td>
<td></td>
</tr>
<tr>
<td>266</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>267</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>268</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>269</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>270</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>271</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>272</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>273</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>274</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>275</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>276</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>277</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>278</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>279</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>280</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>281</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>282</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>283</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td>284</td>
<td>CONTINUE</td>
<td></td>
</tr>
</tbody>
</table>
C --- CALCULATE CELL VOLUMES
C DO 19 IJ=1,NINJ
C VOL(IJ)=ARE(IJ)
C 19 CONTINUE
C IF(AKST) THEN
C SXK=1./6.
C DO 20 I=2,NIM
C I=IMNY(I)
C 20 CONTINUE
C DO 21 J=2,NJM
C IJ=1+J
C 21 CONTINUE
C IKJ=I-J
C IMK=I-J
C ISM=I-J
C 300 RLY+YY(IJ)*1.2
C 301 RIM+YY(IU)*1.2
C 302 RIM+YY(IU)*1.2
C 303 RLY+YY(IU)*1.2
C 304 VOL(IJ)=ORK(*)*(XX(IJ)-XX(IU))**2
C 305 & (XX(IK)-XX(IJ))**2*(RKJ+RKJ+YY(IJ)+YY(IK)+YY(IU))
C 306 & (XX(IK)-XX(IU))**2*(RKJ+RKJ+YY(IK)+YY(IU))
C 307 & (XX(IJ)-XX(IU))**2*(RKJ+RKJ+YY(IJ)+YY(IU))
C 308 CONTINUE
C 309 CONTINUE
C 310 ENDIF
C C --- INITIALIZE VARIABLES INITIALLY
C 311 C HAP=0.5
C 312 C QTR=0.26
C 313 C SMALL=1.E-30
C 314 C GREAT=1.E30
C 315 C DO 22 IJ=1,NINJ
C 316 C DNF(IJ)=DENMET
C 317 C VIS(IJ)=VISO
C 318 C FMU(IJ)=I.1
C 319 C FLR(IJ)=1.2
C 320 C FLR(IJ)=1.2
C 321 C APF(IJ)=0.6
C 322 C AVG(IJ)=0.0
C 323 C ARE(IJ)=0.0
C 324 C ASK(IJ)=0.0
C 325 C AN(IJ)=0.0
C 326 C BE(IJ)=0.0
C 327 C BW(IJ)=0.0
C 328 C BN(IJ)=0.0
C 329 C NS(IJ)=0.0
C 330 C RRS(IJ)=0.0
C 331 C R(IJ)=1.2
C 332 CONTINUE
C 333 IF(AKST) THEN
C 334 DO 23 IJ=1,NINJ
C 335 R(IJ)=YY(IJ)
C 336 CONTINUE
C 337 CONTINUE
C 338 IF(AKST) THEN
C 339 DO 24 IJ=1,NINJ
C 340 CONTINUE
C 341 ENDIF
C C RETURN
C 344 END
C C -------------------------------
C C SUBROUTINE CALCE (PHI,IPHI)
C 347 INCLUDE 'kemod.h'
C C DIMENSION UM(NY),WM(NY),PHI(NXNY),FXD(NY),DM(NY)
C C IF(IPHI.EQ.1) THEN
C 354 URPHI=1./URPF
C 355 PRTHVP=1./PRTH
C 356 ELSE
C 357 URPHI=1./URPF
C 358 PRTHVP=1./PRTH
C 359 ENDIF
C 360 C
C 361 C
C 362 C
C 363 C
C 364 C
C 365 DO 30 J=2,NJM
C 366 IJ=J
C 367 IJM=IJ-1
C 368 IP=IJ+1
C 369 IF(IJ.EQ.1) THEN
C 370 FYM+FFY(IJ)
C 371 FYM+FFY(IJ)
C 372 ARE=HAP*(ARE(IJ)+ARE(IJ))
C 373 DEX+XX(IJ)-XX(IJ)
C 374 DEX+YY(IJ)-YY(IJ)
C 375 VIST+VIS(IJ)-VISO
C 376 GAME-HAP*(VISO+VIST+PRTHVP)*(R(IJ)-R(IJ))
C 377 DM(IJ)=GAME+AREEM*(DEX**2+DYM**2)
C 378 PHISPHIS
C 379 PHINPHIS*(IJ+1)+FNN+PHI(IJ)*FYS
C 380 PHINPHIS*(IJ+1)+FNN+PHI(IJ)*FYS
C 381 WJU(IJ)=U(IJ)
C 382 VM(IJ)=V(IJ)
C 383 WM(IJ)=W(IJ)
C 384 SNW(IJ)=0.
C 385 FNM(IJ)=1.0
C 386 IF(1TBWN(I).EQ.0) TBNW(I).EQ.4) GO TO 30
C 387 DXXS-QTR*X(IJ)-XX(IJ)-XX(IJ)-XX(IJ)-XX(IJ)
C 388 DYSX-QTR*X(IJ)-XX(IJ)-XX(IJ)-XX(IJ)-XX(IJ)
C 389 SNW(IJ)=GAME+AREEM*(DXXS*DXY+DXXS*DYM)*PHISPHIS
C 390 CONTINUE
C 391 CONTINUE
C 392 DO 32 I=2,NIM
C 393 IJ=1
C 394 IJ=IMNY(I)+J
C 395 IJ=J-NJ
C 396 IJ=J-NJ
C 397 IJ=J-NJ
C 398 IJ=J-NJ
C 399 AHEP=ARE(IJ)+ARE(IJ)
C 400 DNX+XX(IJ)-XX(IJ)
C 401 DNX+YY(IJ)-YY(IJ)
C 402 VIST+VIS(IJ)-VISO
C 403 GAME-HAP*(VISO+VIST+PRTHVP)*(R(IJ)-R(IJ))
C 404 DM=GAME+AREEM*(DXXS*DXY+DXXS*DYM)
C 405 FNN=FFY(IJ)
C 406 PHINPHIS*(IJ+1)+FNN+PHI(IJ)*FNN
C 407 U(IJ)=U(IJ)
C 408 W(M=I)
C 409 V(IJ)
C 410 SNW=0.
C 411 IF(1TBS(I).EQ.3) OR ITBS(I).EQ.4) GO TO 33
C 412 DXY+QTR*X(IJ)+XX(IJ)-XX(IJ)-XX(IJ)
C 413 DXY+QTR*X(IJ)+XX(IJ)-XX(IJ)
C 414 SNW=GAME+AREEM*(DXXS*DXY+DXXS*DYM)*PHISPHIS
C 415 DM=GAME+AREEM*(DXXS*DXY+DXXS*DYM)
C 416 CONTINUE
C 417 PHINPHIS+PHINE
C 418 C
C 419 C --- THE MAIN LOOP - ASSEMBLY OF COEFFICIENTS AND SOURCES
C 420 DO 34 J=2,NJM
C 421 IJ=IMNY(I)+J
C 422 IJ=J-NJ
C 423 IJ=J-NJ
C 424 IJ=J-NJ
C 425 IJ=J-NJ
C 426 IJ=J-NJ
C
711 DISNS=GREAT
712 DO 73 J=2,LM,NJM
713 IJ=INMJ(I)+J
714 IJW=INMJ(I)+NJM
715 INMJ,INMJ,NJM
716 DXB=XX(IIJW)-XX(1M JW)
717 DVB=YY(IIJW)-YY(1M JW)
718 XB=HAPF*(XX(IIJW)+XX(1M JW))
719 YB=HAPF*(YY(IIJW)+YY(1M JW))
720 XB=QTR*(XX(IJ)+XX(IJ-NJ-1)+XX(IJ-NJ-1))
721 YB=QTR*(YY(IIJ)+YY(IIJ-1)+YY(IIJ-1))
722 DXPB=XB-XX
723 DYBP=XB-YY
724 DIJSN=DIFL(DXB, DVB, DXPB, DYBP)
725 C--CHECK WEST BOUNDARY
726 IF (ITBN(I).EQ.4) THEN
727 IJW=J
728 IJW=IWM-IWM-1
729 DXB=XX(IIJW)-XX(1M JW)
730 DVB=YY(IIJW)-YY(1M JW)
731 XB=HAPF*(XX(IIJW)+XX(1M JW))
732 YB=HAPF*(YY(IIJW)+YY(1M JW))
733 XB=QTR*(XX(IJ)+XX(IJ-1)+XX(IJ-NJ-1)+XX(IJ-NJ-1))
734 YB=QTR*(YY(IIJ)+YY(IIJ-1)+YY(IIJ-1)+YY(IIJ-1))
735 DXPB=XPB-XX
736 DYPB=XPB-YY
737 DIJSN=DIFL(DXB, DVB, DXPB, DYPB)
738 ENDIF
739 C--CHECK EAST BOUNDARY
740 IF (ITBS(I).EQ.4) THEN
741 IJW=INMJ(I)+J
742 IJW=IWM-IWM-1
743 DXB=XX(IIJW)-XX(1M JW)
744 DVB=YY(IIJW)-YY(1M JW)
745 XB=HAPF*(XX(IIJW)+XX(1M JW))
746 YB=HAPF*(YY(IIJW)+YY(1M JW))
747 XB=QTR*(XX(IJ)+XX(IJ-1)+XX(IJ-NJ-1)+XX(IJ-NJ-1))
748 YB=QTR*(YY(IIJ)+YY(IIJ-1)+YY(IIJ-1)+YY(IIJ-1))
749 DXPB=XPB-XX
750 DYPB=XPB-YY
751 DIJSN=DIFL(DXB, DVB, DXPB, DYPB)
752 ENDIF
753 C DISNP=DISNP/DISNM, DISNP, DISNP
754 RK=DISNP/DEN(IJ)*SQRT(TE(IJ))/VISCOS
755 IF (LAV(2)) THEN
756 ALNM=C*DISNM/(1.0.E-6))/(RK/AMU)
757 ALEO=C*DISNM/(1.0.E-6)/(RK/AED)
758 IF (WONG,.NE.0) THEN
759 AME=.1+4.0E(DYD(IJ)-0.0B00F0)*W00G
760 (*TE(IJ)/((ED(IJ)+SMALL)))*2
761 AMH=ARGE(AME)
762 AMH=ARGE(AME)
763 ALNM=ALNM+AMH*1.5
764 ALEO=ALEO+AMH*2.5
765 END IF
766 ED(IJ)=SQRT(TE(IJ))/**3/ALED
767 VISC(IJ)=VISCOS-DEN(IJ)*CMO*SQR(TE(IJ))**2
768 ELSE
769 RT=DEN(IJ)*TE(IJ)**3/VISCOS*ED(IJ)
770 FMRI(IJ)=(1.0+20.5/R)**2/(1.0-0.0165/R)**2
771 FLR(IJ)=1.0+0.05(FMRI(IJ))**3
772 FLR2(IJ)=1.0-EXP(-RT**RT)
773 ENDIF
774 CONTINUE
775 CONTINUE
776 C
777 C...ALONG THE WEST BOUNDARY
778 C DO 75 J=2,NJM
779 C IF (ITBN(I).NE.4) GO TO 75
780 C
781 C
782 C
783 C
784 C
785 C
786 C
787 C
788 C
789 C
790 C
791 C
792 C
793 C
794 C
795 C
796 C
797 C
798 C
799 C
800 C
801 C
802 C
803 C
804 C
805 C
806 C
807 C
808 C
809 C
810 C
811 C
812 C
813 C
814 C
815 C
816 C
817 C
818 C
819 C
820 C
821 C
822 C
823 C
824 C
825 C
826 C
827 C
828 C
829 C
830 C
831 C
832 C
833 C
834 C
835 C
836 C
837 C
838 C
839 C
840 C
841 C
842 C
843 C
844 C
845 C
846 C
847 C
848 C
849 C
850 C
851 C
852 C
Oct 12 1996 16:41 sskmod_2d

995 IF (RESPOP_GT.RSM.AND.L_GE.NSTP) WRITE(6,2)
996 1 FORMAT(10X,15.,2X,SPIN,RESOR='.',E12.4)
997 2 FORMAT(/,10X, 'SIPIS DID NOT CONVERGE',/)
998 RETURN
999 END
1000 C--------------------------------------------
1001 C--------------------------------------------
1002 C--------------------------------------------
1003 C--------------------------------------------
1004 INCLUDE 'kemod.h'
1005 C--------------------------------------------
1006 C--------------------------------------------
1007 DATA CAPP/A/.4197/
1008 C--------------------------------------------
1009 C--------------------------------------------
1010 ENTRY MODVIS
1011 C--------------------------------------------
1012 DO 80 I=1,NJ
1013 DO J=1,NI
1014 IJ=IMMI(J) + J
1015 VISOLD=VIS(IJ)
1016 VIS(IJ)=VISCRS
1017 IF (ED(IJ).GT.SMALL) VIS(IJ)=FNM(IJ)*DEN(IJ)*TE(IJ)**2*CMU/ED(IJ)+VISCRS
1018 & VIS(IJ)=URPVIS*VIS(IJ)+1.-URPVIS)*VISOLD
1019 80 CONTINUE
1020 C--------------------------------------------
1021 C--------------------------------------------
1022 IF (LAY2) THEN
1023 DO 81 I=2,NIM
1024 IF (TBE(I).NE.4) GO TO 82
1025 DO J=2,JLIE
1026 IJ=IMMI(J) + J
1027 VIS(IJ)=VIS2(IJ)
1028 81 CONTINUE
1029 C--------------------------------------------
1030 IF (LHBE(I).NE.4) GO TO 81
1031 DO J=1,JLIM,NJ
1032 IJ=IMMI(J) + J
1033 VIS(IJ)=VIS2(IJ)
1034 82 CONTINUE
1035 C--------------------------------------------
1036 IF (TBE(I).NE.4) GO TO 86
1037 DO J=1,JLIE,NJ
1038 IJ=IMMI(J) + J
1039 VIS(IJ)=VIS2(IJ)
1040 83 CONTINUE
1041 IF (TBE(I).NE.4) GO TO 85
1042 DO J=2,JLIE
1043 IJ=IMMI(J) + J
1044 VIS(IJ)=VIS2(IJ)
1045 84 CONTINUE
1046 C--------------------------------------------
1047 C--------------------------------------------
1048 C--------------------------------------------
1049 C--------------------------------------------
1050 C--------------------------------------------
1051 C--------------------------------------------
1052 C--------------------------------------------
1053 C--------------------------------------------
1054 C--------------------------------------------
1055 C--------------------------------------------
1056 C--------------------------------------------
1057 C--------------------------------------------
1058 C--------------------------------------------
1059 C--------------------------------------------
1060 C--------------------------------------------
1061 C--------------------------------------------
1062 C--------------------------------------------
1063 C--------------------------------------------
1064 C--------------------------------------------
1065 C--------------------------------------------
1066 C--------------------------------------------
1067 C--------------------------------------------
1068 C--------------------------------------------
1069 C--------------------------------------------
1070 C--------------------------------------------
1071 C--------------------------------------------
1072 C--------------------------------------------
1073 C--------------------------------------------
1074 C--------------------------------------------
1075 C--------------------------------------------
1076 C--------------------------------------------
1077 C--------------------------------------------
1078 C--------------------------------------------
1079 C--------------------------------------------
1080 C--------------------------------------------
1081 C--------------------------------------------
1082 C--------------------------------------------
1083 C--------------------------------------------
1084 C--------------------------------------------
1085 C--------------------------------------------
1086 C--------------------------------------------
1087 C--------------------------------------------
1088 C--------------------------------------------
1089 C--------------------------------------------
1090 C--------------------------------------------
1091 C--------------------------------------------
1092 C--------------------------------------------
1093 C--------------------------------------------
1094 C--------------------------------------------
1095 C--------------------------------------------
1096 C--------------------------------------------
1097 C--------------------------------------------
1098 C--------------------------------------------
1099 C--------------------------------------------
1100 C--------------------------------------------
1101 C--------------------------------------------
1102 C--------------------------------------------
1103 C--------------------------------------------
1104 C--------------------------------------------
1105 C--------------------------------------------
1106 C--------------------------------------------
1107 C--------------------------------------------
1108 C--------------------------------------------
1109 C--------------------------------------------
1110 C--------------------------------------------
1111 C--------------------------------------------
1112 C--------------------------------------------
1113 C--------------------------------------------
1114 C--------------------------------------------
1115 C--------------------------------------------
1116 C--------------------------------------------
1117 C--------------------------------------------
1118 C--------------------------------------------
1119 C--------------------------------------------
1120 C--------------------------------------------
1121 C--------------------------------------------
1122 C--------------------------------------------
1123 C--------------------------------------------
1124 C--------------------------------------------
1125 C--------------------------------------------
1126 C--------------------------------------------
1127 C--------------------------------------------
1128 C--------------------------------------------
1129 C--------------------------------------------
1130 C--------------------------------------------
1131 C--------------------------------------------
1132 C--------------------------------------------
1133 C--------------------------------------------
1134 C--------------------------------------------
1135 C--------------------------------------------
1136 C--------------------------------------------

Printed by yqd449 from mrbig
Oct 12 1996 16:41  sskmod_2d  Page 19

1279      WS=W(IJ)*FY(IJ-1)+W(IJ-1)*(1.0-FX(IJ-1))
1280      WS=W(IJ)*FX(IJ)+W(IJ)*FX(IJ)
1281      W=W(IJ)*FX(IJ-NJ)+W(IJ-NJ)*(1.0-FX(IJ-NJ))
1282      DWM=W=W
1283      DWS=W=WS
1284      GEN(IJ)=GEN(IJ)+ ( (DWS*DNEW*DYEN*DNS)*2 +
1285      (DWM*DNEW*DYEN*DNS-W(IJ)/RP)*ARE(IJ))*2 )
1286      ARE(IJ)=ARE(IJ)**2
1287      ARE(IJ)=ARE(IJ)**2 (V(IJ)/RP)**2 * (VIS(IJ)-VISCOS)
1288      ENDIF
1289      SU(IJ)=AVV(IJ)+GEN(IJ)*VOL(IJ)
1290      845 CONTINUE
1291      AE(IJ)=0.0
1292      RETURN
1293      C
1294      C-----BOUNDARY CONDITIONS FOR DISSIPATION OF KIN. TURB. ENERGY
1295      C
1296      900 CONTINUE
1297      C
1298      CMU5=SQR(SQR(CMU))
1299      CMU7=CMU25+2
1300      C
1301      C-----SOUTH BOUNDARY
1302      DO 910 I=2,91M
1303      IJ=IMNI(I)-2
1304      GO TO (911,912,913,914) ITBS(I)
1305      911 CONTINUE
1306      SU(IJ)=SU(IJ)+AS(IJ)*ED(IJ-1)
1307      HP(IJ)=HP(IJ)+AS(IJ)
1308      GO TO 915
1309      912 ED(IJ-NJ)=ED(IJ)
1310      GO TO 915
1311      913 CONTINUE
1312      IJ=IJ+1
1313      IF(IJ=I+1)
1314      IF(IJ=I+1)
1315      IF(IJ=I+1)
1316      IF(IJ=I+1)
1317      IF(IJ=I+1)
1318      IF(IJ=I+1)
1319      IF(IJ=I+1)
1320      IF(IJ=I+1)
1321      IF(IJ=I+1)
1322      IF(IJ=I+1)
1323      IF(IJ=I+1)
1324      IF(IJ=I+1)
1325      IF(IJ=I+1)
1326      IF(IJ=I+1)
1327      IF(IJ=I+1)
1328      IF(IJ=I+1)
1329      GO TO 915
1330      914 CONTINUE
1331      IF(IJ=I+1)
1332      IF(IJ=I+1)
1333      IF(IJ=I+1)
1334      IF(IJ=I+1)
1335      IF(IJ=I+1)
1336      IF(IJ=I+1)
1337      IF(IJ=I+1)
1338      IF(IJ=I+1)
1339      IF(IJ=I+1)
1340      IF(IJ=I+1)
1341      IF(IJ=I+1)
1342      IF(IJ=I+1)
1343      IF(IJ=I+1)
1344      IF(IJ=I+1)
1345      IF(IJ=I+1)
1346      IF(IJ=I+1)
1347      ELSE
1348      C
1349      TE(IJ)=ABS(TE(IJ))

Oct 12 1996 16:41  sskmod_2d  Page 20

1350      SU(IJ)=CM75*TE(IJ)*SQR(T(TE(IJ)))/(CAPPA*DNS(IJ))*GREAT
1351      BP(IJ)=GREAT
1352      ENDIF
1353      915 CONTINUE
1354      AS(IJ)=0.0
1355      910 CONTINUE
1356      C-----WEST BOUNDARY
1357      DO 920 I=2,91M
1358      IJ=IMNJ(I)+IMN
1359      GO TO (921,922,923,924) ITBN(I)
1360      921 CONTINUE
1361      SU(IJ)=SU(IJ)+AM(IJ)*ED(IJ+1)
1362      BP(IJ)=BP(IJ)+AM(IJ)
1363      GO TO 925
1364      922 ED(IJ+1)=ED(IJ)
1365      GO TO 925
1366      923 CONTINUE
1367      IJ=I+1
1368      IJ=I+1
1369      IJ=I+1
1370      IJ=I+1
1371      IJ=I+1
1372      IJ=I+1
1373      IJ=I+1
1374      IJ=I+1
1375      IJ=I+1
1376      IJ=I+1
1377      IJ=I+1
1378      IJ=I+1
1379      IJ=I+1
1380      IJ=I+1
1381      IJ=I+1
1382      IJ=I+1
1383      924 CONTINUE
1384      IF(LRE) THEN
1385      IJ=I+1
1386      IJ=I+1
1387      IJ=I+1
1388      IJ=I+1
1389      IJ=I+1
1390      IJ=I+1
1391      IJ=I+1
1392      IJ=I+1
1393      IJ=I+1
1394      IJ=I+1
1395      IJ=I+1
1396      IJ=I+1
1397      IJ=I+1
1398      IJ=I+1
1399      ELSE
1400      TE(IJ)=ABS(TE(IJ))
1401      SU(IJ)=CM75*TE(IJ)*SQR(TE(IJ))/(CAPPA*DNS(IJ))*GREAT
1402      BP(IJ)=GREAT
1403      ENDIF
1404      925 CONTINUE
1405      AN(IJ)=0.0
1406      920 CONTINUE
1407      C-----END BOUNDARY
1408      DO 930 I=2,91M
1409      IJ=IMNJ(I)+IMN
1410      GO TO (931,932,933,934) ITBN(I)
1411      931 CONTINUE
1412      SU(IJ)=SU(IJ)+AM(IJ)*ED(IJ+1)
1413      BP(IJ)=BP(IJ)+AM(IJ)
1414      GO TO 935
1415      932 ED(IJ+1)=ED(IJ)
1416      GO TO 935
1417      933 CONTINUE
1418      IJ=I+1
1419      IJ=I+1
1420      IJ=I+1
Oct 12 1996 16:41  sskemod_2d  Page 23

1563 D=V=Y(JJ) - Y(JM)
1564 D=V=P=Q(R)(XX(JJ)+XX(JJ)+XX(JJ)+XX(JJ))
1565 D=V=P=Q(R)(YY(JJ)-YY(JJ)-YY(JJ)-YY(JJ))
1566 FAC=(DBP*DBP*DBP*DBP)/(DBP*DBP*DBP*DBP)
1567 DEL=ED(JJ)+FXE1-ED(JJ)+FXE2+FX1-ED(JM)+FX2
1568 ED(JJ)-ED(JJ)-DEL+FXE1
1569 IJ=I-J-J
1570 ENDIF
1571 A(IJ)+0.0
1572 ENDIF
1573 C - CHECK NORTH WALL BOUNDARY
1574 C
1575 C
1576 JI=IMNJ+IMNJ
1577 IF((TJEW(JJ).EQ.4).THEN
1578 L=J-J
1579 I=I-J-J
1580 TBPRB=QRT(TE(JJ))
1581 DEL=EDN(JJ)
1582 DBX=XX(JJ)-XX(JJ)
1583 D=V=YY(JJ)-YY(JJ)
1584 RB=HAPRI(JJ)
1585 EDN=DEN(JJ)
1586 CALL WALLPN(LB,LM,VISCONS,DENS,DXB,SB,CM,25,LOGIN,CAAPPA
1587 & TAU,SU,SU,SU,SU,SU,SU,SWP,SWP,SWP,GENTE,DELN,TEPR,RF)
1588 S(JJ)+S(JJ)+S(JJ)
1589 BP(JJ)+BP(JJ)+SUP
1590 SUV(JJ)=SUP
1591 S=SUP(JJ)
1592 SUP(JJ)+SUP
1593 GENT(JJ)=GENTE
1594 IF((LBJJ).THEN
1595 IJ=I-J-J
1596 ID=I-J-J
1597 H=J-J-J
1598 J=I-J-J
1599 FXE1-FF(JJ)
1600 FXE2-FF(JM)
1601 FXE1-FF(JM)
1602 FXE2-FF(JM)
1603 FXE1-FF(JM)
1604 FXE2-FF(JM)
1605 FXE1-FF(JM)
1606 FXE2-FF(JM)
1607 FXE1-FF(JM)
1608 FXE2-FF(JM)
1609 ED(JJ)-ED(JJ)-DEL+FXE1
1610 IJ=I-J-J
1611 ENDIF
1612 A(IJ)=0.0
1613 ENDIF
1614 C
1615 CONTINUE
1616 C -- CHECK WALL WEST BOUNDARY
1617 C
1618 DO 620 J=2,NJM
1619 L=JMMNJ(JJ)+J
1620 IF((TJEW(JJ).EQ.4).THEN
1621 L=J-J
1622 L=J-J
1623 L=J-J
1624 TBPRB=QRT(TE(JJ))
1625 DEL=EDN(JJ)
1626 DBX=XX(JJ)-XX(JJ)
1627 D=V=YY(JJ)-YY(JJ)
1628 RB=HAPRI(JJ)
1629 EDN=DEN(JJ)
1630 CALL WALLPN(LB,LM,VISCONS,DENS,DXB,SB,CM,25,LOGIN,CAAPPA
1631 & TAU,SU,SU,SU,SU,SU,SU,SWP,SWP,SWP,GENTE,DELN,TEPR,RF)
1632 S(JJ)+S(JJ)+S(JJ)
1633 BP(JJ)+BP(JJ)+SUP

Oct 12 1996 16:41  sskemod_2d  Page 24

1634 SUP(JJ)+SUP
1635 SUP(JJ)+SUP
1636 SUP(JJ)+SUP
1637 GENT(JJ)=GENTE
1638 IF((LBJJ).THEN
1639 IJ=I-J-J
1640 IJ=I-J-J
1641 IJ=I-J-J
1642 IJ=I-J-J
1643 FY1-FY(JJ)
1644 FY2-FY(JJ)
1645 FY3-FY(JJ)
1646 FY4-FY(JJ)
1647 FY5-FY(JJ)
1648 FY6-FY(JJ)
1649 FY7-FY(JJ)
1650 FY8-FY(JJ)
1651 FY9-FY(JJ)
1652 FY10-FY(JJ)
1653 FY11-FY(JJ)
1654 FY12-FY(JJ)
1655 FY13-FY(JJ)
1656 FY14-FY(JJ)
1657 FY15-FY(JJ)
1658 FY16-FY(JJ)
1659 FY17-FY(JJ)
1660 FY18-FY(JJ)
1661 FY19-FY(JJ)
1662 FY20-FY(JJ)
1663 FY21-FY(JJ)
1664 FY22-FY(JJ)
1665 FY23-FY(JJ)
1666 FY24-FY(JJ)
1667 FY25-FY(JJ)
1668 FY26-FY(JJ)
1669 FY27-FY(JJ)
1670 FY28-FY(JJ)
1671 FY29-FY(JJ)
1672 FY30-FY(JJ)
1673 FY31-FY(JJ)
1674 FY32-FY(JJ)
1675 FY33-FY(JJ)
1676 FY34-FY(JJ)
1677 FY35-FY(JJ)
1678 FY36-FY(JJ)
1679 FY37-FY(JJ)
1680 FY38-FY(JJ)
1681 FY39-FY(JJ)
1682 FY40-FY(JJ)
1683 FY41-FY(JJ)
1684 FY42-FY(JJ)
1685 FY43-FY(JJ)
1686 FY44-FY(JJ)
1687 FY45-FY(JJ)
1688 FY46-FY(JJ)
1689 FY47-FY(JJ)
1690 FY48-FY(JJ)
1691 FY49-FY(JJ)
1692 FY50-FY(JJ)
1693 FY51-FY(JJ)
1694 FY52-FY(JJ)
1695 FY53-FY(JJ)
1696 FY54-FY(JJ)
1697 FY55-FY(JJ)
1698 FY56-FY(JJ)
1699 FY57-FY(JJ)
1700 FY58-FY(JJ)
1701 FY59-FY(JJ)
1702 RETURN
1703 END
1704 C
CHAPTER 3

2D/Axisymmetric Multi-Time-Scale $k-\varepsilon$ Turbulence Model

Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Introduction</td>
<td>30</td>
</tr>
<tr>
<td>3.2 Theory and Model Equations</td>
<td>30</td>
</tr>
<tr>
<td>3.3 Module Evaluation</td>
<td>34</td>
</tr>
<tr>
<td>References</td>
<td>36</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix B</td>
<td>37</td>
</tr>
</tbody>
</table>
In this section a description of the multi-time-scale $k-\varepsilon$ turbulence model that is coded as a self contained computer program to compute turbulent flow quantities in two-dimensional or axisymmetric geometry is given. Detailed description of the module structure, variables used and how to interface the module with CFD flow solvers are given in Appendix B. The module has been tested as a separate self-contained unit using the REACT code [1] and was independently tested at the University of Alabama at Huntsville (UAH) using own code (MAST).

3.1 Introduction

Turbulent flows comprise fluctuating motions with a spectrum of sizes and time scales and different turbulent interactions are associated with different parts of the spectrum. In the single-time-scale turbulence models such as the $k-\varepsilon$ turbulence model it is assumed that a single time scale (proportional to $k/c$) can be used to describe the turbulent flow. In many complex flows turbulence is generally in spectral inequilibrium and a single time scale description is a simplification.

Figure 1, shows a sketch of a typical energy spectrum in a turbulent flow at high Reynolds number in a simplified split spectrum method. Two regions can be identified, the production range (at wave number $\kappa<\kappa'$) where the kinetic energy ($k_p$) leaves this region at a rate ($\varepsilon_p$) and a high wave number or dissipation region ($\kappa>\kappa'$) with kinetic energy ($k_t$) and energy dissipation rate ($\varepsilon_t$).

Hanjalic et al. [2] developed a simple multiple-time-scale turbulence model based on a rational extension of the single scale equation ideas. In their model, a fixed ratio of the turbulent kinetic energy of large eddies ($k_p$) to that of the fine scale eddies ($k_t$) is used to partition the spectrum. Kim and Chen [3] improved on the simplified split spectrum by dynamically determining the location of the partition (i.e $k_p/k_t$) as part of the solution and is dependent on the turbulence intensity, production rate, energy transfer and dissipation rate. The variable partitioning method causes the effective eddy viscosity to decrease when production is high and to increase when production vanishes -a behavior consistent with experimental observations.

3.2 Theory and Model Equations

The multi-time-scale turbulence module is based on the variable partitioning of the turbulent energy spectrum proposed by Kim and Chen [3]. In this model the turbulent kinetic energy spectrum is divided into two sets of wave number regions giving two evolution equations for each region.
These equations represent the kinetic energy \((k_p)\) and the energy transfer rate \((\varepsilon_p)\) in the production range of the spectrum and the kinetic energy \((k_t)\) and the energy dissipation rate \((\varepsilon_t)\) in the dissipation range of the spectrum. This model allows the partition to move toward the high wave number region when production is high and toward the low wave number region when production vanishes.

The equations which describe the multi-time-scale turbulence model used are given below. The turbulent kinetic energy and the energy transfer rate equations for the energy containing large eddies are given as;

\[
\rho \frac{Dk_p}{Dt} = \frac{\partial}{\partial x_i} \left[ (\mu + \frac{\mu_t}{\sigma_{k_p}}) \frac{\partial k_p}{\partial x_i} \right] + G - \rho \varepsilon_p \tag{1}
\]

\[
\rho \frac{D\varepsilon_p}{Dt} = \frac{\partial}{\partial x_i} \left[ (\mu + \frac{\mu_t}{\sigma_{\varepsilon_p}}) \frac{\partial \varepsilon_p}{\partial x_i} \right] + \frac{1}{\rho} C_{p1} \frac{G^2}{k_p} + C_{p2} \frac{G \varepsilon_p}{k_p} - \rho C_{p3} \frac{\varepsilon_p}{k_p} \tag{2}
\]

where \(G\) is the turbulence production rate, given as

\[
G = \mu_e \left[ 2 \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \left( \frac{\partial w}{\partial z} \right)^2 \right] + \left( \frac{\partial u}{\partial y} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial z} \right)^2 + \left( \frac{\partial w}{\partial r} - \frac{w}{r} \right)^2 \tag{3}
\]

where \(\mu\) is the viscosity

\(\mu_t\) is the turbulent viscosity

\(k_p\) is the turbulent kinetic energy in the production range

\(\varepsilon_p\) is the energy transfer rate

\(\sigma_{k_p}\) and \(\sigma_{\varepsilon_p}\) are constants

\(C_{p1}\), \(C_{p2}\) and \(C_{p3}\) are turbulent model constants

The turbulent kinetic energy and the dissipation rate equations for the high wave number small scale eddies region are given as;

\[
\rho \frac{Dk_t}{Dt} = \frac{\partial}{\partial x_i} \left[ (\mu + \frac{\mu_t}{\sigma_{k_t}}) \frac{\partial k_t}{\partial x_i} \right] + \rho \varepsilon_p - \rho \varepsilon_t \tag{4}
\]
\[
\rho \frac{D e_t}{D t} = \frac{\partial}{\partial x_i} \left[ \left( \mu + \frac{\mu_t}{\sigma_{e_t}} \right) \frac{\partial e_t}{\partial x_i} \right] + \rho C_{t1} \frac{\varepsilon_t^2}{k_t} + \rho C_{t2} \frac{\varepsilon_t \varepsilon_p}{k_t} - \rho C_{t3} \frac{\varepsilon_t^2}{k_t}
\]  

(4)

where

- \( k_t \) is the turbulent kinetic energy in the dissipation range
- \( \varepsilon_t \) is the energy dissipation rate
- \( \sigma_{k_t} \) and \( \sigma_{\varepsilon_t} \) are constants
- \( C_{t1}, C_{t2} \) and \( C_{t3} \) are turbulent model constants

The terms \( \frac{1}{\rho} C_{p1} \frac{G^2}{k_p} \) and \( \rho C_{t1} \frac{\varepsilon_t^2}{k_t} \) represent variable energy transfer functions. The first term increases the energy transfer rate when production is high and the second term increases the dissipation rate when the energy transfer rate is high. The turbulent viscosity is given as

\[
\mu_t = \rho C_{\mu f} \frac{k^2}{\varepsilon_p} = \rho C_{\mu} \frac{k^2}{\varepsilon_t}
\]

where \( k = k_p + k_t \) is the total turbulent kinetic energy and \( C_{\mu f} \) is a constant.

The model constants used are similar to those used by Kim and Chen [3]

\( \sigma_{k_p} = 0.75, \quad \sigma_{\varepsilon_p} = 1.15, \quad \sigma_{k_t} = 0.75, \quad \sigma_{\varepsilon_t} = 1.15 \)

\( C_{p1} = 0.21, \quad C_{p2} = 1.24, \quad C_{p3} = 1.84, \quad C_{t1} = 0.29 \)

\( C_{t2} = 1.28, \quad C_{t3} = 1.66 \) and \( C_{\mu f} = 0.09 \)

For turbulent flow analysis, equations (1)-(4) are solved by the module that is interfaced with a Reynolds averaged flow solver to compute the turbulent flow field. For an incompressible, steady and axisymmetric turbulent flow, a generalized equation that expresses the transport of turbulent flow can be written as:

\[
\frac{\partial (\rho u \Phi)}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} (\rho v r \Phi) = \frac{\partial}{\partial x} (\Gamma \Phi x \frac{\partial \Phi}{\partial x}) + \frac{1}{r} \frac{\partial}{\partial r} (r \Gamma \Phi r \frac{\partial \Phi}{\partial r}) + S \Phi
\]

(5)
where $\Phi$ is the dependent variable, which stands for $\Phi = u, v, w$ for the axial, radial and tangential velocities respectively, $\rho$ is the fluid density, $\Gamma_{\phi_x}$ and $\Gamma_{\phi_r}$ are exchange coefficients in $x$ and $r$-directions, respectively, and $S_{\phi}$ is the source term for the variable $\Phi$.

The source terms for the dependent variable are:

- **Axial direction**, $\Phi = u$, $\Gamma_{\phi_x} = 2\mu_e$, $\Gamma_{\phi_r} = \mu_e$ and

\[
S_u = -\frac{\partial P}{\partial x} + \frac{1}{r} \frac{\partial}{\partial r} \left( \mu_e \frac{\partial v}{\partial x} \right)
\]

where $\mu_e$ is the eddy viscosity and $P$ is the pressure.

- **Radial direction**, $\Phi = v$, $\Gamma_{\phi_x} = \mu_e$, $\Gamma_{\phi_r} = 2\mu_e$ and

\[
S_v = -\frac{\partial}{\partial x} \left( \mu_e \frac{\partial u}{\partial r} \right) - 2\mu_e \frac{v}{r^2} + \frac{\rho w^2}{r} - \frac{\partial P}{\partial r}
\]

- **Tangential direction**, $\Phi = w$, $\Gamma_{\phi_x} = \mu_e$, $\Gamma_{\phi_r} = \mu_e$ and

\[
S_w = -\frac{\rho v w}{r} - \frac{w}{r^2} \frac{\partial}{\partial r} (r \mu_e)
\]

Equations (1)-(4) can also be written in a similar form as equation (5) where $\Phi$ stands for;

- **Turbulent kinetic energy in the production range of the energy spectrum**

\[
\Phi = k_p, \quad \Gamma_{\phi_x} = \mu + \frac{\mu_t}{\sigma_{kp}} = \Gamma_{\phi_r} \quad \text{and}
\]

\[
s_{kp} = G - \rho \varepsilon_p
\]

- **Energy transfer rate in the production range of the energy spectrum**

\[
\Phi = \varepsilon_p, \quad \Gamma_{\phi_x} = \mu + \frac{\mu_t}{\sigma_{ep}} = \Gamma_{\phi_r} \quad \text{and}
\]

\[
s_{ep} = \frac{1}{\rho} C_{p1} \frac{G^2}{k_p} + C_{p2} \frac{G\varepsilon_p}{k_p} - \rho C_{p3} \frac{\varepsilon_p^2}{k_p}
\]

- **Turbulent kinetic energy in the dissipation range of the energy spectrum**
\[ \Phi = k_t, \quad \Gamma \Phi_x = \mu^+ \frac{\mu_t}{\sigma_{k_t}} = \Gamma \Phi_r \quad \text{and} \]
\[ s_{k_t} = \rho \varepsilon_p - \rho \varepsilon_t \]

- Energy dissipation rate in the dissipation range of the energy spectrum

\[ \Phi = \varepsilon_b, \quad \Gamma \Phi_x = \mu^+ \frac{\mu_t}{\sigma_{\varepsilon_t}} = \Gamma \Phi_r \quad \text{and} \]
\[ s_{\varepsilon_t} = \rho C_{t1} \frac{\varepsilon_b^2}{k_t} + \rho C_{t2} \frac{\varepsilon_p \varepsilon_t}{k_t} - \rho C_{t3} \frac{\varepsilon_t^2}{k_t} \]

Near a wall, the wall function boundary conditions used are similar to that of Kim and Chen [3].

A two layer model for the multi-time-scale \( k-e \) turbulence model similar to that of Chen and Patel [4] for the single-time-scale \( k-e \) turbulence model is included in the present release.

### 3.3 Model Evaluation

The multi-time-scale \( k-e \) module was evaluated by comparisons with experimental studies. One of the test problems considered was the backward facing step of Driver and Seegmiller [5] where the multi scale \( k-e \) model predicted a recirculation length of 6.14 step heights (H) downstream of the step which is closer to the experimental value (6.10 H) than the standard \( k-e \) model (5.35H).

The majority of the tests were conducted using Roback and Johnson's experimental data [6] for swirling confined double concentric jets. Preliminary analysis indicated some sensitivity to the ratio \( k_p/k_t \) at the inlet boundary, however, a value of 3 was found reasonable in the present analysis. Figures 2 and 3 show the streamline patterns for wall functions and two-layer near wall treatments respectively. The upper (a) and lower (b) parts correspond to the single-scale \( k-e \) and the multi-scale \( k-e \) models respectively. It can be seen from these contours that there are two recirculation zones in the chamber, one is near the expansion corner and another located in the central region and accurate predictions of this central region is very important in combusting swirling flows. Figures 4a and 4b, show the axial velocity along the centerline. In terms of strength and size of the central recirculation zone, the multi-scale \( k-e \) model yields better agreement than the single-scale \( k-e \) model. In the central recirculation region the \( k-e \) model tends to connect the energy transfer rate to the local mean strain rate too strongly while the multi-scale model suppresses this tendency.
Figures 5 and 6 show the radial profiles of the mean axial velocity at different axial locations downstream of the inlet using the wall function and the two-layer near-wall treatments respectively. Similarly, figures 7 and 8 show the corresponding profiles for the tangential velocity, and figures 9 and 10 show the radial profiles of the axial normal turbulent intensity $\langle uu \rangle^{1/2}$ using both the wall function and the two-layer near-wall treatments. In general, the numerical results indicate that the multi-scale model gives better agreement than the standard $k-\varepsilon$ model.
REFERENCES


Figure 1. Description and nomenclature of the multiple-time-scale turbulence model;

\[ k_p = \int_{\kappa=0}^{\kappa=\kappa_1} E \, d\kappa, \quad k_t = \int_{\kappa=\kappa_1}^{\kappa=\infty} E \, d\kappa \]

\( \kappa_1 \) = Partition wave number, \( E \) = Energy spectral density
Figure 2. Stream-function contours of confined swirling jet flow using the wall function near wall treatment
Figure 3. Stream-function contours of confined swirling jet flow using the two-layer near wall treatment
Figure 4. Axial mean velocity along the centerline
(a) wall function model
(b) two-layer model
Figure 5. Radial profiles of mean axial velocity using the wall function near wall treatment

- exp. data
- single scale $k$-$\varepsilon$
- multi-scale $k$-$\varepsilon$
Figure 6. Radial profiles of mean axial velocity using the two-layer near wall treatment

- exp. data

--- single scale $k-\varepsilon$

---- multi-scale $k-\varepsilon$
Figure 7  Radial profiles of mean tangential velocity using the wall function near wall treatment

- exp. data
- single scale $k$-$\varepsilon$
- multi-scale $k$-$\varepsilon$
Figure 8. Radial profiles of mean tangential velocity using the two-layer near wall treatment

- exp. data
- single scale $k$-$\varepsilon$
- multi-scale $k$-$\varepsilon$
Figure 9. Radial profiles of turbulent intensity $(\overline{uu})^{1/2}$ using the wall-function near wall treatment

- exp. data
- single scale $k-\varepsilon$
- multi-scale $k-\varepsilon$
Figure 10. Radial profiles of turbulent intensity

$(\bar{u}\bar{u})^{1/2}$ using the two layer near-wall treatment

- exp. data
- single scale $k$-$\varepsilon$
- multi-scale $k$-$\varepsilon$
APPENDIX B

Multi-Scale $k$-$\varepsilon$ Module Deck

B.1 Introduction

This user's manual describes the multi-scale $k$-$\varepsilon$ module deck. The module is a self contained FORTRAN source code to compute turbulent kinetic energy, energy dissipation and turbulent eddy viscosity using the multi-time-scale $k$-$\varepsilon$ turbulence model. It uses as input the mean flow properties as computed by conventional CFD techniques. The module is constructed to be self-contained, stand alone and compatible with a number of CFD solvers. A discussion of the multi-time-scale $k$-$\varepsilon$ module structure is given next together with flow charts to show how to interface the module with a number of flow solvers. A list of variable names used is also given.

B.2 Program KEMOD

This is basically the solver for the $k$ and $\varepsilon$ - transport equations in both the production and the dissipation regions of the energy spectrum. It reads through its argument list different variables from the calling flow solver. These variables are described below where, each variable name ends with either an (I) for Integer variable, (R) for Real variable or (L) for Logical variable.

The flow chart of the program is shown in Figure B.1. It shows the main operations performed by the code.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XR</td>
<td>Grid node locations in the $x$ or $\xi$-direction, dimensioned to XR (NX,NY) (input from flow solver)</td>
</tr>
<tr>
<td>YR</td>
<td>Grid node locations in the $y$ or $\eta$-direction, dimensioned to YR (NX,NY) (input from flow solver)</td>
</tr>
</tbody>
</table>
Axial or x-direction velocity \((u)\), dimensioned as \(UR (NX,NY)\) (input from flow solver)

Radial or y-direction velocity \((v)\), also dimensional as \(VR (NX,NY)\) (input from flow solver)

Azimuthal velocity \((w)\), dimensional \(WR (NX,NY)\) (input from flow solver)

Large scale turbulence kinetic energy \(k_p\), dimensioned \(TER (NX,NY)\) (calculated in KEMOD and returned to flow solver)

Large scale turbulent energy dissipation rate \(\varepsilon_p\), dimensioned \(EDR (NX,NY)\) (calculated in KEMOD and returned to flow solver)

Small scale turbulence kinetic energy \(k_t\), dimensioned \(TETR (NX,NY)\) (calculated in KEMOD and returned to flow solver)

Small scale turbulent energy dissipation rate \(\varepsilon_t\), dimensioned \(EDTR (NX,NY)\) (calculated in KEMOD and returned to flow solver)

Fluid density, dimensioned \(DENR (NX,NY)\)

Under-relaxation factors dimensioned as \(URFKER(4)\) and specified as follows:

\(URFKER(1)\) for large scale turbulent energy equation
\(URFKER(2)\) for small scale turbulent energy equation
\(URFKER(3)\) for large scale turbulent energy dissipation equation
\(URFKER(4)\) for small scale turbulent energy dissipation equation

Prandtl/Schmidt numbers dimensioned as \(PRTKER(4)\) and specified as follows:

\(PRTKER(1)\) for large scale turbulent energy equation
\(PRTKER(2)\) for small scale turbulent energy equation
\(PRTKER(3)\) for large scale turbulent energy dissipation equation
\(PRTKER(4)\) for small scale turbulent energy dissipation equation

\(GR = 1.0\) if second order upwinding is desired
= 0.0 if first order upwinding is used (input from flow solver).

**F1R**  
Mass flux variable at cell faces in x- or \( \xi \)-direction, dimensioned F1R (NX,NY)  
(input from flow solver)

**F2R**  
Mass flux variable at cell faces in y or \( \eta \)-direction, dimensioned F2R (NX,NY)  
(input from flow solver)

**ITERI**  
Iteration number (input from flow solver), this number must be equal to 1 for a  
restart case

**VISCOSR**  
Dynamic viscosity (input from flow solver)

**VISR**  
Eddy viscosity, dimensioned VISR (NX,NY) (calculated in KEMOD and  
returned to main solver)

**URFVISR**  
Under-relaxation factor for total viscosity calculation

**AKSIL**  
Logical variable for axisymmetric geometry (AKSIL=·TRUE·) or plain  
geometry (AKSIL=·FALSE·) (input from flow solver)

**C1R**  
Turbulence model constant, \( C_1 \) (input from flow solver)

**C2R**  
Turbulence model constant, \( C_2 \) (input from flow solver)

**CMUR**  
Turbulence model constant, \( C_\mu \) (input from flow solver)

**NIMI**  
Number of cell nodes in the I- or \( \xi \)-coordinate lines. (input from flow solver)

**NJMI**  
Number of cell nodes in the J- or \( \eta \)-coordinate lines. (input from flow solver)

**JTBEI**  
Boundary condition flag along east boundary must have one for each boundary  
ode set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall e.g., for an outlet  
boundary condition on the east boundary set JTBEI to NJ*2, and similarly for  
other boundaries, dimensioned JTBEI (NY) (input from flow solver)
**JTBWI**  Boundary condition flag along west boundary, dimensioned JTBWI (NY)  
(input from flow solver)

**ITBNI**  Boundary condition flag along north boundary, dimensioned ITBNI (NX)  
(input from flow solver)

**ITBSI**  Boundary condition flag along south boundary, dimensioned ITBSI (NY)  
(input from flow solver)

Program KEMOD is interfaced with the main flow solver by a call to KEMOD with its arguments. For iterative flow solvers KEMOD is called within the iteration sequence after the solution of the momentum equations where the mean velocities are passed to KEMOD. There are different flow solvers utilizing different schemes from staggered to nonstaggered grid arrangement and for nonorthogonal coordinate system there are at least three alternatives to the choice of the velocity components

i. Cartesian velocity components

ii. Contravariant velocity components

iii. Covariant velocity components

The Cartesian velocity components are the most widely used and have the advantage of simple formulation of the governing equations. Whatever the arrangement used, mass fluxes at cell faces are required and passed to KEMOD as F1R and F2R in both directions. The location of other variables such as k and ε are at the cell center or cell nodes.

The module starts by reassigning variable names passed to it from flow solver to names that are shared with the different subroutines of the module in an include file "mske.h". The user must set the values for NX and NY in mske.h greater than or equal to the maximum grid dimensions. Then a check is made if it is the first iteration in which case the grid file "GRIDG" is called -after passing the grid node locations XR & YR in KEMOD- in order to calculate grid related quantities which will be explained later. The need to call GRIDG can be waived if all the grid data are passed to the module. That is all the information about the grid such as interpolation factors FX and FY, cell areas (ARE) and volumes (VOL) and normal distances of first grid point from grid boundaries
(DNS from south boundary, DNN - from north boundary, DNW - from west boundary and DNE - from east boundary).

After this, two calls to subroutine CALCKE are made to calculate the large and small scale turbulent kinetic energies with the identifier IPHI=1 and 2 respectively. The large and small scale energy dissipation equations are solved next by calling subroutine CALCKE again with the identifiers IPHI=3 and 4 respectively. The effective viscosity is calculated next. At locations where \( \epsilon \) is close to zero (i.e., \( \leq 10^{-30} \)) viscosity is set to zero. A provision is made for under relaxing changes in effective viscosity which may help to stabilize oscillations and improve convergence rate.

**B.3 Subroutines**

**GRIDG**

Before calling this subroutine, the coordinates of all grid nodes, defined in reference to a fixed Cartesian coordinate frame are read. Figure B.2 shows the position of cell and grid nodes.

This subroutine is called only once to calculate coordinates of grid nodes (intersection of grid lines) and geometrical properties of the grid (cell areas and volumes, interpolation factors, normal distances of near-boundary cell nodes from boundary). All variables including grid node coordinates are converted to one-dimensional arrays. These are formed by scanning the grid in J-direction (figure B.2) for I=1, and then repeating for all I's. The position of any node in one-dimensional array is therefore defined as:

\[
IJ = (I,J) = (I-1) \times NJ + J
\]

the actual number of grid nodes is one row and one column less than for all cell nodes. For I = NI and J = NJ fictitious grid nodes are introduced which have the same coordinates as actual nodes on NI-1 in I-direction and NJ-1 in J-direction.

The subroutine then calculates interpolation factors which are associated with cell nodes and are used in the main program to calculate values of dependent variables at locations other than cell nodes (cell centers). Definition of these are given in Figure B.3. Cell areas and volumes are
calculated next followed by calculations of normal distances of near-boundary nodes from all four outer boundaries.

**CALCKE (PHI, IPHI)**

This subroutine solves the linearized and discretized transport equations for the turbulent energies ($k_p$ and $k_t$) and the energy dissipation ($\varepsilon_p$ and $\varepsilon_t$). The two dummy parameters in the calling statement, PHI and IPHI, represent arrays containing dependent variables for which the equation is to be solved. The subroutine sets up the convective and diffusive coefficients over the entire field. Then it calculates the source terms for either $k$ or $\varepsilon$ transport equations. A call is made to entry MODMSKE in order to modify these sources and boundary coefficients to suit the particular problem.

The discretized equations have the form

$$A_p \Phi_p = \sum_{i=E,W,N,S} A_i \Phi_i + S \Phi$$

where the coefficients $A_i$ ($i=E,W,N,S$ see figure B.3) contain both the convective and diffusive fluxes. These equations are assembled and solved by calling subroutine SOLSIP which is based on Stone's Strongly Implicit Solver [7].

**SOLSIP**

This subroutine solves the system of linear algebraic equations for $k$ and $\varepsilon$ using Stone's Implicit Procedure [7]. The array RES (IJ) is used to store residuals. The sum of absolute residuals "RESORP" calculated in the first pass through this part of the routine is used as a measure of convergence of the solution process as a whole and this value is stored in RESOR (IPHI). This variable RESOR (IPHI) is passed to the main solver and if desired can be normalized and compared with the maximum error allowed there. If necessary inner iterations counter L and the sum of absolute residuals RESORP are printed out to monitor the rate of convergence of $k$ and $\varepsilon$ solution. If the ratio RSM is greater than the maximum allowed for the variable in question, SOR (IPHI), and the number of inner iterations is smaller than a prescribed maximum, NSWP (IPHI),
then the routine repeats the sequence of calculating the residuals, increment vectors and updating the dependent variable.

**MODMSKE**

This subroutine is called from CALCKE subroutine and sets the boundary conditions for $k_p$, $k_t$ and $e_p$, $e_t$ depending on which variable being called (IDIR = 1, 2, 3, and 4 for $k_p$, $k_t$, $e_p$, and $e_t$ respectively). Consider the south boundary for example, if it is one of four options:

1. An inflow boundary ITBS(I) = 1, where the source term is set to accept the inlet values at $J = 1$ (south boundary).
2. Outflow boundary ITBS(I) = 2, where zero gradient in $y$ or $\eta$-direction is employed.
3. Symmetry boundary, TBS(I) = 3, where gradients normal to symmetry plane are zero.
4. Wall boundary, ITBS(I) = 4, where the production term GENTS(I) calculated from subroutine WALLFN in program MODIFY is added to the rest of the source term SU(IJ).

**B.4 Program MODIFY**

This subroutine is called from the $u$ and $v$ solver routines. It basically updates the flux source term of the discretized momentum equation due to wall shear stresses. If the $u$-momentum equation for example is discretized in the form

$$A_p^* u_p = \sum_{i=EWNS} A_{i} u_i + S_u^*$$

where $P, E, W, N, S$ are cell nodes as shown in Figure B.3, and $A_p^*$ and $A_i$'s contain convective and diffusive coefficients. $S_u^*$ is the source term containing pressure gradients and cross-derivative diffusion terms and convective terms for second-order upwinding scheme. This source term is
usually linearized as $S_u^* = S_u - B_p u_p$ The term $B_p$ is usually moved to the left hand side of the equation and modifies the diagonal coefficient $A_p = A_p^* + B_p$, and the equation can be written as

$$A_p u_p = \sum_{i=E,W,N,S} A_i u_i + S_u$$

Then $S_u$ and $B_p$ are passed to subroutine MODIFY where they are modified if a wall is present (e.g., ITBS(I) = 4 for south boundary).

**List of Argument Variable Names**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMU</td>
<td>Turbulence model constant, $C_\mu$ (input from flow solver)</td>
</tr>
<tr>
<td>VISCOS</td>
<td>Dynamic viscosity (input from flow solver)</td>
</tr>
<tr>
<td>XX</td>
<td>Grid node locations in the $x$ or $\xi$-direction, dimensioned to XX (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>YY</td>
<td>Grid node locations in the $y$ or $\eta$-direction, dimensioned to YY (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>R</td>
<td>Grid node radius equal to 1 for non-axisymmetric and YY for axisymmetric, dimensioned to R (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>DNS</td>
<td>Normal distance to south, dimensioned to DNS (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>DNN</td>
<td>Normal distance to north, dimensioned to DNN (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>DNW</td>
<td>Normal distance to west, dimensioned to DNW (NX*NY) (input from flow solver)</td>
</tr>
<tr>
<td>DNE</td>
<td>Normal distance to east, dimensioned to DNE (NX*NY) (input from flow solver)</td>
</tr>
</tbody>
</table>
U Axial or x-direction velocity (u), dimensioned as UR (NX*NY) (input from flow solver)

V Radial or y-direction velocity (v), also dimensional as VR (NX*NY) (input from flow solver)

W Azimuthal velocity (w), dimensional WR (NX*NY) (input from flow solver)

DEN Fluid density, dimensional DEN (NX*NY) (input from flow solver)

TE Large scale turbulence kinetic energy $k_p$, dimensioned TE (NX*NY) (calculated in KEMOD and returned to flow solver)

TET Small scale turbulence kinetic energy $k_t$, dimensioned TET (NX*NY) (calculated in KEMOD and returned to flow solver)

SU Variable source term, dimensioned SU (NX*NY)

BP Constant source term, dimensioned BP (NX*NY)

AE Cell area, dimensioned to AE (NX*NY) (input from flow solver)

AW Cell area, dimensioned to AW (NX*NY) (input from flow solver)

AN Cell area, dimensioned to AN (NX*NY) (input from flow solver)

AS Cell area, dimensioned to AS (NX*NY) (input from flow solver)

SUVS,SPVS,SUWS,SPWS
Source terms at south boundary due to wall shear stress, all dimensioned to $S##S$ (NX*NY) (returned to flow solver)

SUVN,SPVN,SUWN,SPWN
Source terms at north boundary due to wall shear stress, all dimensioned to $S##N$ (NX*NY) (returned to flow solver)
SUWV,SPWV,SWW,SPWW
Source terms at west boundary due to wall shear stress, all dimensioned to S##W (NX*NY) (returned to flow solver)

SUVE,SPVE,SUWE,SPWE
Source terms at east boundary due to wall shear stress, all dimensioned to S##E (NX*NY) (returned to flow solver)

GENTS,GENTN,GENTW,GENTEE
Generation terms at south, north, west, and east boundaries respectively due to moving walls, with GENTS(NX), GENTN(NX), GENTW(NY), and GENTEE(NY) (returned to flow solver)

NX
Maximum number of cell nodes in the I- or ξ-coordinate lines. (input from flow solver)

NY
Maximum number of cell nodes in the J- or η-coordinate lines. (input from flow solver)

NXNY
NX*NY

NIM
Number of cell nodes in the I- or ξ-coordinate lines. (input from flow solver)

NJM
Number of cell nodes in the J- or η-coordinate lines. (input from flow solver)

ITBS
Boundary condition flag along south boundary must have one for each boundary node set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall e.g., for an outlet boundary condition on the east boundary set ITBS to NI*2, and similarly for other boundaries, dimensioned ITBS (NX) (input from flow solver)

ITBN
Boundary condition flag along north boundary, dimensioned ITBN (NX) (input from flow solver)

JTBW
Boundary condition flag along west boundary, dimensioned JTBW (NY) (input from flow solver)
JTBE Boundary condition flag along east boundary, dimensioned JTBE (NY) (input from flow solver)

For an iterative flow solver using the finite-volume methodology. A typical interface and call to KEMOD from the main flow solver can be represented by a flow chart as shown in figure B.4.
Figure B.1 Multi-scale $k$-$\varepsilon$ module deck flow chart
Figure B.2 Position of cell and grid nodes
Figure B.3 Definition of interpolation factors

\[
FX_p = \frac{P_e}{P_e + eE}, \quad FY_p = \frac{P_n}{P_n + nN}
\]
Figure B.4 Typical main flow solver flow chart with calls to the multi-scale k-e module.
<table>
<thead>
<tr>
<th>Oct 12 1996 16:38</th>
<th>mskemod_2d</th>
<th>Page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2D/AXISYMMETRIC MULTI-SCALE K-E TURBULENCE MODULE</td>
<td>C</td>
</tr>
<tr>
<td>2</td>
<td>Rocketdyne CFD Technology Center</td>
<td>C</td>
</tr>
<tr>
<td>3</td>
<td>MULTISCALE 2-Equation with 2 Wall Treatments</td>
<td>C</td>
</tr>
<tr>
<td>4</td>
<td>WALL FUNCTION 2-LAYER</td>
<td>C</td>
</tr>
<tr>
<td>5</td>
<td>SUBROUTINE KEMOD (NM1, NMJ, XR, YR, WR, VR, TEY, TETR,</td>
<td>C</td>
</tr>
<tr>
<td>6</td>
<td>&amp; EDF, EDR, UF1KFR, UF2KFR, URKFR, URKFR,</td>
<td>C</td>
</tr>
<tr>
<td>7</td>
<td>&amp; PRFED, PRFED, GR, FI1, FI2, ITB1, VISCM, VISCR, AKSI,</td>
<td>C</td>
</tr>
<tr>
<td>8</td>
<td>&amp; LAY2L, CMUR, I2LWI, 12LSI, J2JIL, J2JNL, JTB1, JTB2, ITBNI,</td>
<td>C</td>
</tr>
<tr>
<td>9</td>
<td>COMMON/G (X(NX,NY), Y(NX,NY)</td>
<td>C</td>
</tr>
<tr>
<td>10</td>
<td>DATA GREAT, SMALL/1.E30, 1.E-30/</td>
<td>C</td>
</tr>
<tr>
<td>11</td>
<td>DATA HAF, QTR, 0.5, 0.25/</td>
<td>C</td>
</tr>
<tr>
<td>12</td>
<td>DATA SOR/0.3, 0.1, 1.0, 1.0/</td>
<td>C</td>
</tr>
<tr>
<td>13</td>
<td>DATA MNDFP/1.0, 1.0, 1.0/</td>
<td>C</td>
</tr>
<tr>
<td>14</td>
<td>DATA CP1, CP2, CP3/0.21, 1.24, 1.84/</td>
<td>C</td>
</tr>
<tr>
<td>15</td>
<td>DATA CT1, CT2, CT3/0.39, 1.28, 1.66/</td>
<td>C</td>
</tr>
<tr>
<td>16</td>
<td>COMMON/GR/ X(NX,NY), Y(NX,NY)</td>
<td>C</td>
</tr>
<tr>
<td>17</td>
<td>INUM=NUMI</td>
<td>C</td>
</tr>
<tr>
<td>18</td>
<td>NJM=NJMI</td>
<td>C</td>
</tr>
<tr>
<td>19</td>
<td>NJ=NJM+1</td>
<td>C</td>
</tr>
<tr>
<td>20</td>
<td>N1=NJM+1</td>
<td>C</td>
</tr>
<tr>
<td>21</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>22</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>23</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>24</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>25</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>26</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>27</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>28</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>29</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>30</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>31</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>32</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>33</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>34</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>35</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>36</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>37</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>38</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>39</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>40</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>41</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>42</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>43</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>44</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>45</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>46</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>47</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>48</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>49</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>50</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>51</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>52</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>53</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>54</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>55</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>56</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>57</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>58</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>59</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>60</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>61</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>62</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>63</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>64</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>65</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>66</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>67</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>68</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>69</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>70</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>71</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>72</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>73</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>74</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>75</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>76</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>77</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>78</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>79</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>80</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>81</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>82</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>83</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>84</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>85</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>86</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>87</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>88</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>89</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>90</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>91</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>92</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>93</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>94</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>95</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>96</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>97</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>98</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>99</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>100</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>101</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>102</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>103</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>104</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>105</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>106</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>107</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>108</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>109</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>110</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>111</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>112</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>113</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>114</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>115</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>116</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>117</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>118</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>119</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>120</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>121</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>122</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>123</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>124</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>125</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>126</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>127</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>128</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>129</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>130</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>131</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>132</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>133</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>134</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>135</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>136</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>137</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>138</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>139</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>140</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>141</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>142</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
<tr>
<td>143</td>
<td>UF1KFR=UF1KFR</td>
<td>C</td>
</tr>
<tr>
<td>144</td>
<td>UF2KFR=UF2KFR</td>
<td>C</td>
</tr>
</tbody>
</table>

Printed by yqd449 from mrtig
143 C     NJ=NJM+1
144 NJ=NJM+1
145 NI=NJM+1
146 NIM=NJM+1
147 DO 2 I=1,NI
148 I=MJ(1)-1,NJ-1)
149 CONTINUE
150 DO 3 I=1,NIM
151 J=NJM
152 X(I,J+1)=X(I,1,J)
153 Y(I,J+1)=Y(I,1,J)
154 3 CONTINUE
155 DO 4 J=1,NJ
156 I=NIM
157 X(I+1,J)=X(I,J)
158 Y(I+1,J)=Y(I,J)
159 4 CONTINUE
160 C
161 C... GRID ORIGIN AT X=0, Y=0
162 DO 5 I=1,NI
163 DO 5 J=1,NJ
164 IJ=I+J
165 XX(IJ)=X(I,J)
166 YY(IJ)=Y(I,J)
167 5 CONTINUE
168 C
169 C-----CALCULATION OF INTERPOLATION FACTORS
170 C
171 DO 6 J=1,NJM
172 FX(I,J)=0.
173 FY(I,J)=0.
174 6 CONTINUE
175 C
176 DO 7 J=2,NJM
177 IJ=J
178 FX(I,J)=0.
179 LIE=NJM-1
180 DO 8 I=2,LIE
181 IJ=I+I
182 IR=IJ+J
183 IJ=LJ-1
184 IK=IJ-J-NJ
185 DXP=0.5X[(X(IJ)-XX(IJM)+XX(IJM)-XX(IJM-1))]
186 DYP=0.5X[(XX(IJM)+XX(IJM)-XX(IJM-1))]
187 8 CONTINUE
188 DYP=0.5X[(YY(IJ)-YY(IJM)+YY(IJM)-YY(IJM-1))]
189 DXP=SQRT(DX**2+DYP**2)
190 DK=SQRT(DX**2+DYP**2)
191 FX(I,J)=DXP/DK
192 CONTINUE
193 IJ=IJ-1
194 9 CONTINUE
195 CONTINUE
196 C
197 DO 9 I=1,NI
198 IJ=I+1
199 FX(I,J)=FX(I,J+1)
200 IJ=IJ+1
201 FX(I,J)=FX(I,J-1)
202 CONTINUE
203 CONTINUE
204 DO 10 J=2,NJM
205 IJ=I+1
206 FY(I,J)=0.
207 LN=NJM-1
208 DO 11 J=2,LNJ
209 IJ=IJ+1
210 IJ=I+1
211 IJ=IJ-1
212 IJ=IJ-J-NJ
213 DXP=0.5X[(XX(IJ)+XX(IJM)-XX(IJM)-XX(IJM-1))]
214 DYP=0.5X[(YY(IJ)+YY(IJM)-YY(IJM)-YY(IJM-1))]
215 DXP=0.5X[(YY(IJ)+YY(IJM)-YY(IJM)-YY(IJM-1))]
216 DYP=0.5X[(YY(IJ)+YY(IJM)-YY(IJM)-YY(IJM-1))]
217 DXP=0.5X[(YY(IJ)+YY(IJM)-YY(IJM)-YY(IJM-1))]
218 DYP=0.5X[(YY(IJ)+YY(IJM)-YY(IJM)-YY(IJM-1))]
219 CONTINUE
220 IJ=IJ+1
221 FY(I,J)=FY(I,J-1)
222 CONTINUE
223 C
224 DO 12 J=1,NJ
225 FY(I,J)=FY(I,J-1)
226 CONTINUE
227 IJ=IJ+1
228 CONTINUE
229 IJ=IJ+1
230 CONTINUE
231 C
232 DO 13 IJ=1,NJM
233 CONTINUE
234 AR(IJ)=0.
235 CONTINUE
236 C
237 DO 14 I=1,NM
238 CONTINUE
239 IJ=IJ+1
240 DYNNE=XX(IJ)-XX(IJM-1)
241 DYNNS=YY(IJ)-YY(IJM-1)
242 DYNNO=XX(IJ)-XX(IJM-1)
243 DYNSS=YY(IJ)-YY(IJM-1)
244 CONTINUE
245 C
246 C------NORMAL DISTANCE FROM THE WALL
247 C
248 DO 15 J=1,NJ
249 DNM=0.
250 DNM=0.
251 CONTINUE
252 C
253 DO 16 J=1,NJ
254 DNM=0.
255 C
256 C
257 C
258 C
259 C
260 C
261 C
262 C
263 C
264 C
265 C
266 C
267 C
268 C
269 C
270 C
271 C
272 C
273 C
274 C
275 C
276 C
277 C
278 C
279 C
280 C
281 C
282 C
283 C
284 C

mskemod_2d

mskemod_2d
E E O >.
E _m e'
D
ml I_l
m!
_n
041
_'_ _ _ _
0
'0
E
E
E
E
E
E
E
E
E _O
E
E
E
E
E
E
E
E
E

Oct 12 1996 16:38 mskemod_2d Page 5

285      IXL+JL-NJ
286      DXB=XX(IJ)-XX(IJ-1)
287      DYB=YY(IJ)-YY(IJ-1)
288      DXB=0.25*(XX(IM)+XX(IJ)+XX(IM-1)+XX(IJ-1))
289      DYB=0.25*(YY(IM)+YY(IJ)+YY(IM-1)+YY(IJ-1))
290      DXY(J)=DELTAX(DXB,DXB,DXB,DXB)
291      CONTINUE
292      CALCULATE CELL VOLUMES
293      DO 19 1J=1,NINJ
294      VOL(IJ)=AM(E(IJ))
295      19 CONTINUE
296      C
297      IF(AKSI) THEN
298      SXX=1./6.
299      DO 20 1I=2,NIM
300      20 CONTINUE
301      DT=1J+2,NIM
302      IJ=1000000000
303      VOL(IJ)=SXX*(XX(IJ)-XX(IJ-1))
304      VOL(IJ-1)=SXX*(XX(IJ-1)-XX(IJ))
305      VOL(IJ+1)=SXX*(XX(IJ+1)-XX(IJ))
306      VOL(IJ-2)=SXX*(XX(IJ-2)-XX(IJ-1))
307      VOL(IJ+2)=SXX*(XX(IJ+2)-XX(IJ+1))
308      CONTINUE
309      INITIALIZE VARIABLES INITIALLY
310      RETURN
311      C
312      C
313      C
314      C
315      C
316      C
317      C
318      C
319      C
320      C
321      C
322      C
323      C
324      SMALL=1.E-30
325      GREAT=1.E30
326      DO 22 1J=1,NINJ
327      D=0.25
328      DENS(IJ)=DENS(IJ)+DEN
329      VIS(IJ)=VISCO
330      HMO(IJ)=1.0
331      PLRI(IJ)=1.0
332      APJ(IJ)=0.0
333      AVJ(IJ)=0.0
334      AVJ(IJ)=0.0
335      AS(IJ)=0.0
336      AN(IJ)=0.0
337      AN(IJ)=0.0
338      BM(IJ)=0.0
339      BM(IJ)=0.0
340      BN(IJ)=0.0
341      BS(IJ)=0.0
342      RES(IJ)=0.0
343      R(IJ)=0.0
344      CONTINUE
345      IF(AKSI) THEN
346      DO 23 1J=1,NINJ
347      R(IJ)=YY(IJ)
348      23 CONTINUE
349      CONTINUE
350      END
351      C
352      C
353      C
354      C
355      C

Oct 12 1996 16:38 mskemod_2d Page 6

356      SUBROUTINE CALCCE(HI,PHI)
357      C
358      C
359      INCLUDE 'mskemod.h'
360      C
361      DIMENSION U(NY),V(NY),W(NY),PHI(NY),FXW(NY),DYN(3)
362      C
363      DO TO (1,2,3,4) PHI
364      CONTINUE
365      C
366      CONTINUE
367      C
368      CONTINUE
369      C
370      PRTINV=1./PRT
371      C
372      CONTINUE
373      C
374      CONTINUE
375      C
376      CONTINUE
377      C
378      CONTINUE
379      C
380      C
381      C
382      C
383      C
384      C
385      C
386      C
387      C
388      C
389      C
390      C
391      C
392      C
393      C
394      C
395      C
396      C
397      C
398      C
399      C
400      C
401      C
402      C
403      C
404      C
405      C
406      C
407      C
408      CONTINUE
409      C
410      DO 100 1J=1,NIM
411      J=1
412      C
413      C
414      C
415      C
416      C
417      C
418      C
419      C
420      C
421      C
422      C
423      C
424      C
425      C
426      C
3
Oct 12 1996 16:38 mskemod_2d Page 7

427 UN=U(IJ)
428 VN=V(IJ)
429 WN=WN(IJ)
430 SD=SD(IJ)
431 IF(ITB6(IJ).EQ.0.OR.ITRS(IJ).EQ.4) GO TO 110
432 DXET+QTR*XX(IJFP)+XX(IJFP-NJ)-XX(IJFP-JJ)-XX(IJFP)
433 +QTR*YY(IJFP)-YY(IJFP-NJ)-YY(IJFP-JJ)-YY(IJFP)
434 SD=SD+GENN(AREN)*/(DXH*DXET+DVY*DYTE) */(PHIN*PHIN(IJ))
435 SDNN=SDNW
436 110 CONTINUE
437 PHIN=PHINE
438 C-----THE MAIN LOOP - ASSEMBLY OF COEFFICIENTS AND SOURCES
439 C
440 D 101 J=2,NJM
441 IF(IJMOD(IJ).EQ.1) J=2,NJM
442 IJ=IJ+2
443 IF(IJ.GT.NJM) IJ=1
444 FXY=FX(IJ)
445 FXY=FXY(IJ)
446 FYS=FYS(IJ)
447 DXX-XX(IJ)+XX(IJ-1)
448 DYY-YY(IJ)-YY(IJ-1)
449 DXX=XX(IJ)+XX(IJ-1)
450 DYY=YY(IJ)-YY(IJ-1)
451 AREH=ARE(H(IJ))+ARE(H(IJ+1))
452 AREH=ARE(H(IJ))+ARE(H(IJ+1))
453 VISE=VISE(IJ)+FXY*VIS(IJ+1)*FXY
454 VISE=VISE(IJ)+FXY*VIS(IJ+1)*FXY
455 GAME=GAME(IJ)+VIS(IJ)*FXY(IJ+1)*FXY
456 GAME=GAME(IJ)+VIS(IJ)*FXY(IJ+1)*FXY
457 C
458 D=SN
459 DEGAME=ARE(IJ)*DXX**2*DXY**2
460 D=GAME/ARE(IJ)*DXX**2*DXY**2
461 C
462 C LINEAR UPWIND DIFFERENCING
463 C
464 AREM=MIN(FI(IJ).EQ.1.00)*FX(IJ)+G
465 AREM=MAX(FI(IJ).EQ.1.00)*FX(IJ)+G
466 AREM=MAX(FI(IJ).EQ.1.00)*FX(IJ)+G
467 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
468 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
469 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
470 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
471 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
472 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
473 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
474 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
475 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
476 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
477 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
478 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
479 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
480 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
481 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
482 AN=MN(FI(IJ).EQ.1.00)*FX(IJ)+G
483 C
484 DXX=QTR-(XX(IJ)+XX(IJ-1)-XX(IJ-1)-XX(IJ-1))
485 DYY=QTR+(YY(IJ)-YY(IJ-1)-YY(IJ-1)-YY(IJ-1))
486 DXET-QTR*(XX(IJ)-XX(IJ-1)+(IJP-1)-XX(IJ-1))
487 DYTE-QTR*(YY(IJ)-YY(IJ-1)-(IJP-1)-YY(IJ-1))
488 C
489 C PHINE-PHINE
490 C
491 C LINEAR UPWIND DIFFERENCING
492 C
493 C INJ1=INJ-1
494 C INJ2=INJ-2
495 C
496 C
497 C
498 C
499 C
500 C
501 C
502 C
503 C
504 C
505 C
506 C
507 C
508 C
509 C
510 C
511 C
512 C
513 C
514 C
515 C
516 C
517 C
518 C
519 C
520 C
521 C
522 C
523 C
524 C
525 C
526 C
527 C
528 C
529 C
530 C
531 C
532 C
533 C
534 C
535 C
536 C
537 C
538 C
539 C
540 C
541 C
542 C
543 C
544 C
545 C
546 C
547 C
548 C
549 C
550 C
551 C
552 C
553 C
554 C
555 C
556 C
557 C
558 C
559 C
560 C
561 C
562 C
563 C
564 C
565 C
566 C
567 C
568 C

mskemod_2d

Oct 12 1996 16:38 mskemod_2d Page 8

mskemod_2d

Printed by yqd449 from mtbig
711  RK=DISN*DEN(J3)*SQRT(TETOT)/VISCONS
712  ALMU=C11*DISN*1.0-EXP(-RK/AMU))
713  ALED=C11*DISN*1.0-EXP(-RK/ALED)
714  ED(JJ)=TETOT+SQRT(TETOT)/ALRU
715  VIS2(JJ)=DISN+DEN(J3)*CU*M*SQRT(TETOT)*ALMU
716  CONTINUE
717  CONTINUE
718  C
719  C...ALONG THE NORTH BOUNDARY
720  C
721  DO 72 J=I,NUM
722  IF(ITTM(J).NE.7) GO TO 72
723  DISNR=DISNR+DISNR
724  DISN=DISN+DISN
725  DISN=DISN+DISN
726  DISN=DISN+DISN
727  DISN=DISN+DISN
728  DISN=DISN+DISN
729  DISN=DISN+DISN
730  DISN=DISN+DISN
731  DISN=DISN+DISN
732  DISN=DISN+DISN
733  DISN=DISN+DISN
734  DISN=DISN+DISN
735  DISN=DISN+DISN
736  DISN=DISN+DISN
737  DISN=DISN+DISN
738  DISN=DISN+DISN
739  DISN=DISN+DISN
740  DISN=DISN+DISN
741  DISN=DISN+DISN
742  DISN=DISN+DISN
743  DISN=DISN+DISN
744  DISN=DISN+DISN
745  DISN=DISN+DISN
746  DISN=DISN+DISN
747  DISN=DISN+DISN
748  DISN=DISN+DISN
749  DISN=DISN+DISN
750  DISN=DISN+DISN
751  DISN=DISN+DISN
752  DISN=DISN+DISN
753  DISN=DISN+DISN
754  DISN=DISN+DISN
755  DISN=DISN+DISN
756  DISN=DISN+DISN
757  DISN=DISN+DISN
758  DISN=DISN+DISN
759  DISN=DISN+DISN
760  DISN=DISN+DISN
761  DISN=DISN+DISN
762  DISN=DISN+DISN
763  DISN=DISN+DISN
764  DISN=DISN+DISN
765  DISN=DISN+DISN
766  ENDIF
767  C
768  C...ALONG THE WEST BOUNDARY
769  C
770  DO 75 J=J-2,NUM
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1279</td>
<td>SU(IJ)=SU(IJ)+AS(IJ)*TET(IJ-1)</td>
<td>Page 19</td>
</tr>
<tr>
<td>1280</td>
<td>BP(IJ)=BP(IJ)+AS(IJ)</td>
<td>mskemod_2d</td>
</tr>
<tr>
<td>1281</td>
<td>GO TO 915</td>
<td></td>
</tr>
<tr>
<td>1282</td>
<td>TET(IJ-1)=TET(IJ)</td>
<td>Oct 12 1996 16:38</td>
</tr>
<tr>
<td>1283</td>
<td>GO TO 915</td>
<td>mskemod_2d</td>
</tr>
<tr>
<td>1284</td>
<td>913 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1285</td>
<td>IJ=IJ-1</td>
<td></td>
</tr>
<tr>
<td>1286</td>
<td>IPJ=IJ*NJ</td>
<td></td>
</tr>
<tr>
<td>1287</td>
<td>IJ=IJ-NJ</td>
<td></td>
</tr>
<tr>
<td>1288</td>
<td>FXE1=FX(IJ)</td>
<td></td>
</tr>
<tr>
<td>1289</td>
<td>FXE2=FX(IM1)</td>
<td></td>
</tr>
<tr>
<td>1290</td>
<td>FXW1=FXE1</td>
<td></td>
</tr>
<tr>
<td>1291</td>
<td>FXW2=FXE2</td>
<td></td>
</tr>
<tr>
<td>1292</td>
<td>DBX=XX(IJ)-XX(IM1)</td>
<td></td>
</tr>
<tr>
<td>1293</td>
<td>DYB=YY(IJ)-YY(IM1)</td>
<td></td>
</tr>
<tr>
<td>1294</td>
<td>DXYB=Q*XX(IJ-1)-XX(IM1)-XX(IM2)</td>
<td></td>
</tr>
<tr>
<td>1295</td>
<td>DXYB=Q*YY(IJ-1)-YY(IM1)-YY(IM2)</td>
<td></td>
</tr>
<tr>
<td>1296</td>
<td>FAC=(DXYB<em>DXYB)/((DXYB+2</em>DXYB+2 SMALL)</td>
<td></td>
</tr>
<tr>
<td>1297</td>
<td>DEL=TET(IJ)*(FXW1+FXE2)+TET(IPJ)*FXE1+TET(IM1)+FXW2</td>
<td></td>
</tr>
<tr>
<td>1298</td>
<td>TET(IJ+1)=TET(IJ)+DEL*FAC</td>
<td></td>
</tr>
<tr>
<td>1299</td>
<td>IJ=IJ+1</td>
<td></td>
</tr>
<tr>
<td>1300</td>
<td>GO TO 915</td>
<td></td>
</tr>
<tr>
<td>1301</td>
<td>914 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1302</td>
<td>SU(IJ)=FRAC*TE(IJ)*GREAT</td>
<td></td>
</tr>
<tr>
<td>1303</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>1304</td>
<td>915 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1305</td>
<td>AS(IJ)=0.0</td>
<td></td>
</tr>
<tr>
<td>1306</td>
<td>910 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1307</td>
<td>C-----NORTH BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1308</td>
<td>DO 920 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>1309</td>
<td>J=MN(I)+1</td>
<td></td>
</tr>
<tr>
<td>1310</td>
<td>GO TO (921,922,923,924)</td>
<td></td>
</tr>
<tr>
<td>1311</td>
<td>921 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1312</td>
<td>SU(IJ)=SU(IJ)+AN(IJ)*TET(IJ+1)</td>
<td></td>
</tr>
<tr>
<td>1313</td>
<td>BP(IJ)=BP(IJ)+AN(IJ)</td>
<td></td>
</tr>
<tr>
<td>1314</td>
<td>GO TO 925</td>
<td></td>
</tr>
<tr>
<td>1315</td>
<td>922 TET(IJ+1)=TET(IJ)</td>
<td></td>
</tr>
<tr>
<td>1316</td>
<td>GO TO 925</td>
<td></td>
</tr>
<tr>
<td>1317</td>
<td>923 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1318</td>
<td>IJ=IJ-1</td>
<td></td>
</tr>
<tr>
<td>1319</td>
<td>IPJ=IJ-NJ</td>
<td></td>
</tr>
<tr>
<td>1320</td>
<td>IJ=IJ-NJ</td>
<td></td>
</tr>
<tr>
<td>1321</td>
<td>FXE1=FX(IJ)</td>
<td></td>
</tr>
<tr>
<td>1322</td>
<td>FXE2=FX(IM1)</td>
<td></td>
</tr>
<tr>
<td>1323</td>
<td>FXW1=FXE1</td>
<td></td>
</tr>
<tr>
<td>1324</td>
<td>FXW2=FXE2</td>
<td></td>
</tr>
<tr>
<td>1325</td>
<td>DBX=XX(IJ)-XX(IM1)</td>
<td></td>
</tr>
<tr>
<td>1326</td>
<td>DYB=YY(IJ)-YY(IM1)</td>
<td></td>
</tr>
<tr>
<td>1327</td>
<td>DXYB=Q*XX(IJ-1)-XX(IM1)-XX(IM2)</td>
<td></td>
</tr>
<tr>
<td>1328</td>
<td>DXYB=Q*YY(IJ-1)-YY(IM1)-YY(IM2)</td>
<td></td>
</tr>
<tr>
<td>1329</td>
<td>FAC=(DXYB<em>DXYB)/((DXYB+2</em>DXYB+2 SMALL)</td>
<td></td>
</tr>
<tr>
<td>1330</td>
<td>DEL=TET(IJ)*(FXW1+FXE2)+TET(IPJ)*FXE1+TET(IM1)+FXW2</td>
<td></td>
</tr>
<tr>
<td>1331</td>
<td>TET(IJ+1)=TET(IJ-1)+DEL*FAC</td>
<td></td>
</tr>
<tr>
<td>1332</td>
<td>IJ=IJ+1</td>
<td></td>
</tr>
<tr>
<td>1333</td>
<td>GO TO 925</td>
<td></td>
</tr>
<tr>
<td>1334</td>
<td>924 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1335</td>
<td>SU(IJ)=FRAC*TE(IJ)*GREAT</td>
<td></td>
</tr>
<tr>
<td>1336</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>1337</td>
<td>925 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1338</td>
<td>AN(IJ)=0.0</td>
<td></td>
</tr>
<tr>
<td>1339</td>
<td>920 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1340</td>
<td>C-----WEST BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1341</td>
<td>DO 930 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>1342</td>
<td>J=MN(I)+1</td>
<td></td>
</tr>
<tr>
<td>1343</td>
<td>GO TO (931,932,933,934)</td>
<td></td>
</tr>
<tr>
<td>1344</td>
<td>931 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1345</td>
<td>SU(IJ)=SU(IJ)+AW(IJ)*TET(IJ-NJ)</td>
<td></td>
</tr>
<tr>
<td>1346</td>
<td>BP(IJ)=BP(IJ)+AW(IJ)</td>
<td></td>
</tr>
<tr>
<td>1347</td>
<td>GO TO 935</td>
<td></td>
</tr>
<tr>
<td>1348</td>
<td>932 TET(IJ-NJ)=TET(IJ)</td>
<td></td>
</tr>
<tr>
<td>1349</td>
<td>GO TO 935</td>
<td></td>
</tr>
<tr>
<td>1350</td>
<td>933 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1351</td>
<td>IJ=J</td>
<td></td>
</tr>
<tr>
<td>1352</td>
<td>IJ=I+1</td>
<td></td>
</tr>
<tr>
<td>1353</td>
<td>LMN(J)=1</td>
<td></td>
</tr>
<tr>
<td>1354</td>
<td>FYN1=FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1355</td>
<td>FYN2=FY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1356</td>
<td>FYS1=F1,FYN1</td>
<td></td>
</tr>
<tr>
<td>1357</td>
<td>FYS2=F1,FYN2</td>
<td></td>
</tr>
<tr>
<td>1358</td>
<td>DXY=YY(IJ)-YY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1359</td>
<td>DXY=XX(IJM)-XX(IJM)</td>
<td></td>
</tr>
<tr>
<td>1360</td>
<td>DXY=Q*XX(IJM-1)-XX(IJM+1)-XX(IJM)</td>
<td></td>
</tr>
<tr>
<td>1361</td>
<td>DXY=Q*YY(IJM)-YY(IJM+1)-YY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1362</td>
<td>FAC=(MXYB,MXYB)/((MXYB+2*MXYB+2 SMALL)</td>
<td></td>
</tr>
<tr>
<td>1363</td>
<td>DEL=TET(IJM)*(FXS1+FXS2)+TET(IPJ)*FXS1+TET(IJM)+FXS2</td>
<td></td>
</tr>
<tr>
<td>1364</td>
<td>TET(IJM)=TET(IJM)+DEL*FAC</td>
<td></td>
</tr>
<tr>
<td>1365</td>
<td>LMN(J)=1</td>
<td></td>
</tr>
<tr>
<td>1366</td>
<td>GO TO 935</td>
<td></td>
</tr>
<tr>
<td>1367</td>
<td>934 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1368</td>
<td>SU(IJ)=FRAC*TE(IJ)*GREAT</td>
<td></td>
</tr>
<tr>
<td>1369</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>1370</td>
<td>935 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1371</td>
<td>AN(IJ)=0.0</td>
<td></td>
</tr>
<tr>
<td>1372</td>
<td>930 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1373</td>
<td>C-----EAST BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1374</td>
<td>DO 940 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>1375</td>
<td>J=MN(I)+1</td>
<td></td>
</tr>
<tr>
<td>1376</td>
<td>GO TO (941,942,943,944)</td>
<td></td>
</tr>
<tr>
<td>1377</td>
<td>941 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1378</td>
<td>SU(IJ)=SU(IJ)+AE(IJ)*TET(IJM)</td>
<td></td>
</tr>
<tr>
<td>1379</td>
<td>BP(IJ)=BP(IJ)+AE(IJ)</td>
<td></td>
</tr>
<tr>
<td>1380</td>
<td>GO TO 945</td>
<td></td>
</tr>
<tr>
<td>1381</td>
<td>942 TET(IJM)=TET(IJ)</td>
<td></td>
</tr>
<tr>
<td>1382</td>
<td>GO TO 945</td>
<td></td>
</tr>
<tr>
<td>1383</td>
<td>943 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1384</td>
<td>IJ=IJ+1</td>
<td></td>
</tr>
<tr>
<td>1385</td>
<td>IJM=I+1</td>
<td></td>
</tr>
<tr>
<td>1386</td>
<td>LMN(J)=1</td>
<td></td>
</tr>
<tr>
<td>1387</td>
<td>FYN1=FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1388</td>
<td>FYN2=FY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1389</td>
<td>FYS1=F1,FYN1</td>
<td></td>
</tr>
<tr>
<td>1390</td>
<td>FYS2=F1,FYN2</td>
<td></td>
</tr>
<tr>
<td>1391</td>
<td>DXY=YY(IJ)-YY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1392</td>
<td>DXY=XX(IJM)-XX(IJM)</td>
<td></td>
</tr>
<tr>
<td>1393</td>
<td>DXY=Q*XX(IJM-1)-XX(IJM+1)-XX(IJM)</td>
<td></td>
</tr>
<tr>
<td>1394</td>
<td>DXY=Q*YY(IJM)-YY(IJM+1)-YY(IJM)</td>
<td></td>
</tr>
<tr>
<td>1395</td>
<td>FAC=(MXYB,MXYB)/((MXYB+2*MXYB+2 SMALL)</td>
<td></td>
</tr>
<tr>
<td>1396</td>
<td>DEL=TET(IJM)*(FXS1+FXS2)+TET(IPJ)*FXS1+TET(IJM)+FXS2</td>
<td></td>
</tr>
<tr>
<td>1397</td>
<td>TET(IJM)=TET(IJM)+DEL*FAC</td>
<td></td>
</tr>
<tr>
<td>1398</td>
<td>LMN(J)=1</td>
<td></td>
</tr>
<tr>
<td>1399</td>
<td>GO TO 945</td>
<td></td>
</tr>
<tr>
<td>1400</td>
<td>944 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1401</td>
<td>SU(IJ)=FRAC*TE(IJ)*GREAT</td>
<td></td>
</tr>
<tr>
<td>1402</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>1403</td>
<td>945 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1404</td>
<td>AE(IJ)=0.0</td>
<td></td>
</tr>
<tr>
<td>1405</td>
<td>940 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1406</td>
<td>RETURN</td>
<td></td>
</tr>
<tr>
<td>1407</td>
<td>C-----</td>
<td></td>
</tr>
<tr>
<td>1408</td>
<td>RETURN</td>
<td></td>
</tr>
<tr>
<td>1409</td>
<td>C-----BOUNDARY CONDITIONS FOR ENERGY TRANSFER RATE (ED)</td>
<td></td>
</tr>
<tr>
<td>1410</td>
<td>C-----</td>
<td></td>
</tr>
<tr>
<td>1411</td>
<td>1000 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1412</td>
<td>CE</td>
<td></td>
</tr>
<tr>
<td>1413</td>
<td>CM=CM=SQRT(SQRT(CM))</td>
<td></td>
</tr>
<tr>
<td>1414</td>
<td>CM=CM=CM<em>CM</em>CM</td>
<td></td>
</tr>
<tr>
<td>1415</td>
<td>CC</td>
<td></td>
</tr>
<tr>
<td>1416</td>
<td>C-----SOUTH BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1417</td>
<td>DO 1010 I=2,NIM</td>
<td></td>
</tr>
<tr>
<td>1418</td>
<td>J=MN(I)+1</td>
<td></td>
</tr>
<tr>
<td>1419</td>
<td>GO TO (1011,1012,1013,1014)</td>
<td></td>
</tr>
<tr>
<td>1420</td>
<td>1011 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>Oct 12 1996 16:38</td>
<td>mskemod_2d</td>
<td>Page 21</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>1421</td>
<td>$SU_{IJ} = SU_{IJ} + AS_{IJ} \cdot ED_{IJ-1}$</td>
<td></td>
</tr>
<tr>
<td>1422</td>
<td>$BP_{IJ} = BP_{IJ} + AS_{IJ}$</td>
<td></td>
</tr>
<tr>
<td>1423</td>
<td>GO TO 1015</td>
<td></td>
</tr>
<tr>
<td>1424</td>
<td>1012</td>
<td>$ED_{IJ-NJ} = ED_{IJ}$</td>
</tr>
<tr>
<td>1425</td>
<td>GO TO 1015</td>
<td></td>
</tr>
<tr>
<td>1426</td>
<td>1013 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1427</td>
<td>I3 = I3 - 1</td>
<td></td>
</tr>
<tr>
<td>1428</td>
<td>IFJ = I3 - NJ</td>
<td></td>
</tr>
<tr>
<td>1429</td>
<td>IN3 = I3 - NJ</td>
<td></td>
</tr>
<tr>
<td>1430</td>
<td>FXE1 = FX1(IJ)</td>
<td></td>
</tr>
<tr>
<td>1431</td>
<td>$FX2 = FX2 - FX1(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1432</td>
<td>$FW1 = FW1 - FXE1$</td>
<td></td>
</tr>
<tr>
<td>1433</td>
<td>$FW2 = FW2 - FX2$</td>
<td></td>
</tr>
<tr>
<td>1434</td>
<td>$DXB = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1435</td>
<td>$DYB = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1436</td>
<td>$DXBP = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1437</td>
<td>$DYBP = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1438</td>
<td>$FX2E = DXB - DYB$</td>
<td></td>
</tr>
<tr>
<td>1439</td>
<td>$FW2E = FW1 - FW2$</td>
<td></td>
</tr>
<tr>
<td>1440</td>
<td>$ED(IJ) = ED(IJ-1) - DEL - FAC$</td>
<td></td>
</tr>
<tr>
<td>1441</td>
<td>I3 = I3 - 1</td>
<td></td>
</tr>
<tr>
<td>1442</td>
<td>GO TO 1015</td>
<td></td>
</tr>
<tr>
<td>1443</td>
<td>1014 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1444</td>
<td>TT = ABS$[TE(IJ) + TET(IJ)]$</td>
<td></td>
</tr>
<tr>
<td>1445</td>
<td>$SU_{IJ} = SU_{IJ} + GREAT \cdot CMS75 + TT \cdot SQRT(TT) / (CAPPAM + DMS(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1446</td>
<td>BP(IJ) = GREAT</td>
<td></td>
</tr>
<tr>
<td>1447</td>
<td>1015 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1448</td>
<td>AS(IJ) = 0.0</td>
<td></td>
</tr>
<tr>
<td>1449</td>
<td>1010 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1450</td>
<td>C---- NORTH BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1451</td>
<td>DO 1020 LD = 2, NUM</td>
<td></td>
</tr>
<tr>
<td>1452</td>
<td>I3 = INM(NJ)</td>
<td></td>
</tr>
<tr>
<td>1453</td>
<td>GO TO (1021, 1022, 1023, 1024, 1025) ITBN(I)</td>
<td></td>
</tr>
<tr>
<td>1454</td>
<td>1021 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1455</td>
<td>$SU_{IJ} = SU_{IJ} + AN(IJ) \cdot ED(IJ-1)$</td>
<td></td>
</tr>
<tr>
<td>1456</td>
<td>BP(IJ) = BP(IJ) + AN(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1457</td>
<td>GO TO 1025</td>
<td></td>
</tr>
<tr>
<td>1458</td>
<td>1022 ED = ED(IJ) ED(IJ)</td>
<td></td>
</tr>
<tr>
<td>1459</td>
<td>GO TO 1025</td>
<td></td>
</tr>
<tr>
<td>1460</td>
<td>1023 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1461</td>
<td>I3 = I3 - 1</td>
<td></td>
</tr>
<tr>
<td>1462</td>
<td>IFJ = I3 - NJ</td>
<td></td>
</tr>
<tr>
<td>1463</td>
<td>INM = I3 - NJ</td>
<td></td>
</tr>
<tr>
<td>1464</td>
<td>$FX2E = FX2E(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1465</td>
<td>$FW2E = FW2E - FXE1$</td>
<td></td>
</tr>
<tr>
<td>1466</td>
<td>$DXB = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1467</td>
<td>$DYB = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1468</td>
<td>$DXBP = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1469</td>
<td>$DYBP = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1470</td>
<td>$FX2 = DXB - DYB$</td>
<td></td>
</tr>
<tr>
<td>1471</td>
<td>$FW2 = FW1 - FW2$</td>
<td></td>
</tr>
<tr>
<td>1472</td>
<td>$ED(IJ) = ED(IJ-1) - DEL - FAC$</td>
<td></td>
</tr>
<tr>
<td>1473</td>
<td>I3 = I3 - 1</td>
<td></td>
</tr>
<tr>
<td>1474</td>
<td>GO TO 1025</td>
<td></td>
</tr>
<tr>
<td>1475</td>
<td>1024 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1476</td>
<td>TT = ABS$(TE(IJ) + TET(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1477</td>
<td>$SU_{IJ} = SU_{IJ} + GREAT \cdot CMS75 + TT \cdot SQRT(TT) / (CAPPAM + DMS(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1478</td>
<td>BP(IJ) = GREAT</td>
<td></td>
</tr>
<tr>
<td>1479</td>
<td>1025 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1480</td>
<td>AN(IJ) = 0.0</td>
<td></td>
</tr>
<tr>
<td>1481</td>
<td>1020 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1482</td>
<td>LD = 2, NUM</td>
<td></td>
</tr>
<tr>
<td>1483</td>
<td>1021 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1484</td>
<td>C----WEST BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1485</td>
<td>DO 1030 JO = 2, NJM</td>
<td></td>
</tr>
<tr>
<td>1486</td>
<td>I3 = INM(JO)</td>
<td></td>
</tr>
<tr>
<td>1487</td>
<td>GO TO (1031, 1032, 1033, 1034) ITBN(J)</td>
<td></td>
</tr>
<tr>
<td>1488</td>
<td>1031 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1489</td>
<td>$SU_{IJ} = SU_{IJ} + AM(IJ) \cdot ED(IJ-NJ)$</td>
<td></td>
</tr>
<tr>
<td>1490</td>
<td>BP(IJ) = BP(IJ) + AM(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1491</td>
<td>GO TO 1015</td>
<td></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Oct 12 1996 16:38</th>
<th>mskemod_2d</th>
<th>Page 22</th>
</tr>
</thead>
<tbody>
<tr>
<td>1492</td>
<td>ED(IJ-NJ) = ED(IJ)</td>
<td></td>
</tr>
<tr>
<td>1493</td>
<td>GO TO 1035</td>
<td></td>
</tr>
<tr>
<td>1494</td>
<td>1033 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1495</td>
<td>I3 = J</td>
<td></td>
</tr>
<tr>
<td>1496</td>
<td>IFJ = I3 + 1</td>
<td></td>
</tr>
<tr>
<td>1497</td>
<td>INM = I3</td>
<td></td>
</tr>
<tr>
<td>1498</td>
<td>FYM1 = FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1499</td>
<td>FYN2 = FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1500</td>
<td>FYS1 = 1.0</td>
<td></td>
</tr>
<tr>
<td>1501</td>
<td>FYS2 = 0.0</td>
<td></td>
</tr>
<tr>
<td>1502</td>
<td>DVB = Y(IJ) \cdot YY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1503</td>
<td>DXX = XX(IJ) - XX(IJ)</td>
<td></td>
</tr>
<tr>
<td>1504</td>
<td>$DXBP = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1505</td>
<td>DYB = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1506</td>
<td>FAC = $DXB + DXBP + DVB$</td>
<td></td>
</tr>
<tr>
<td>1507</td>
<td>DEL = ED(IJ) + FYM1 + FYN2 + FYS1 - FYS2</td>
<td></td>
</tr>
<tr>
<td>1508</td>
<td>ED(IJ-NJ) = ED(IJ-NJ) \cdot FAC</td>
<td></td>
</tr>
<tr>
<td>1509</td>
<td>I3 = I3 + NJ</td>
<td></td>
</tr>
<tr>
<td>1510</td>
<td>GO TO 1035</td>
<td></td>
</tr>
<tr>
<td>1511</td>
<td>1034 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1512</td>
<td>TT = ABS$(TE(IJ) + TET(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1513</td>
<td>$SU_{IJ} = SU_{IJ} + GREAT \cdot CMS75 + TT \cdot SQRT(TT) / (CAPPAM + DMS(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1514</td>
<td>BP(IJ) = GREAT</td>
<td></td>
</tr>
<tr>
<td>1515</td>
<td>1035 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1516</td>
<td>AM = AM(IJ)</td>
<td></td>
</tr>
<tr>
<td>1517</td>
<td>1030 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1518</td>
<td>C----- EAST BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>1519</td>
<td>DO 1040 JO = 2, NUM</td>
<td></td>
</tr>
<tr>
<td>1520</td>
<td>I3 = INM(JO)</td>
<td></td>
</tr>
<tr>
<td>1521</td>
<td>GO TO (1041, 1042, 1043, 1044) ITBN(J)</td>
<td></td>
</tr>
<tr>
<td>1522</td>
<td>1041 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1523</td>
<td>$SU_{IJ} = SU_{IJ} + AM(IJ) \cdot ED(IJ-NJ)$</td>
<td></td>
</tr>
<tr>
<td>1524</td>
<td>BP(IJ) = BP(IJ) + AM(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1525</td>
<td>GO TO 1045</td>
<td></td>
</tr>
<tr>
<td>1526</td>
<td>1042 ED(IJ-NJ) = ED(IJ)</td>
<td></td>
</tr>
<tr>
<td>1527</td>
<td>GO TO 1045</td>
<td></td>
</tr>
<tr>
<td>1528</td>
<td>1043 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1529</td>
<td>I3 = INM(JO)</td>
<td></td>
</tr>
<tr>
<td>1530</td>
<td>IFJ = I3 + 1</td>
<td></td>
</tr>
<tr>
<td>1531</td>
<td>INM = I3</td>
<td></td>
</tr>
<tr>
<td>1532</td>
<td>FYM1 = FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1533</td>
<td>FYN2 = FY(IJ)</td>
<td></td>
</tr>
<tr>
<td>1534</td>
<td>FYS1 = 1.0</td>
<td></td>
</tr>
<tr>
<td>1535</td>
<td>FYS2 = 0.0</td>
<td></td>
</tr>
<tr>
<td>1536</td>
<td>$DXX = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1537</td>
<td>DVB = Y(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1538</td>
<td>$DXBP = XX(IJ) - XX(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1539</td>
<td>$DYBP = YY(IJ) - YY(IJ)$</td>
<td></td>
</tr>
<tr>
<td>1540</td>
<td>FAC = $DXB + DXBP + DVB$</td>
<td></td>
</tr>
<tr>
<td>1541</td>
<td>DEL = ED(IJ) + FYM1 + FYN2 + FYS1 - FYS2</td>
<td></td>
</tr>
<tr>
<td>1542</td>
<td>ED(IJ-NJ) = ED(IJ-NJ) \cdot FAC</td>
<td></td>
</tr>
<tr>
<td>1543</td>
<td>I3 = I3 + NJ</td>
<td></td>
</tr>
<tr>
<td>1544</td>
<td>GO TO 1045</td>
<td></td>
</tr>
<tr>
<td>1545</td>
<td>1044 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1546</td>
<td>TT = ABS$(TE(IJ) + TET(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1547</td>
<td>$SU_{IJ} = SU_{IJ} + GREAT \cdot CMS75 + TT \cdot SQRT(TT) / (CAPPAM + DMS(IJ))$</td>
<td></td>
</tr>
<tr>
<td>1548</td>
<td>BP(IJ) = GREAT</td>
<td></td>
</tr>
<tr>
<td>1549</td>
<td>1045 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1550</td>
<td>AM = AM(IJ)</td>
<td></td>
</tr>
<tr>
<td>1551</td>
<td>1040 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>1552</td>
<td>C---- RETURN</td>
<td></td>
</tr>
<tr>
<td>1553</td>
<td>C---- BOUNDARY CONDITIONS FOR ENERGY DISSIPATION RATE (EDIT)</td>
<td></td>
</tr>
<tr>
<td>1554</td>
<td>1110 I = 2, NUM</td>
<td></td>
</tr>
<tr>
<td>1555</td>
<td>I3 = INM(JO)</td>
<td></td>
</tr>
<tr>
<td>1556</td>
<td>GO TO (1111, 1112, 1113, 1114) ITBN(J)</td>
<td></td>
</tr>
<tr>
<td>Oct 12 1996 16:38</td>
<td>mskemod_2d</td>
<td>Page 23</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>1563 1111 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1564 SU(IJ)+SU(IJ)+AS(IJ)*EDT(IJ-1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1565 BF(IJ)+BF(IJ)+AS(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1566 GO TO 1115</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1567 1112 EDT(IJ-NJ)=EDT(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1568 GO TO 1115</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1569 1113 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1570 IJ=I-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1571 IJ=I+1,NJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1572 IM2=I-1,NJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1573 FEX1=FX(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1574 FEX2=FX(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1575 FXW1=1-FEX1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1576 FXW2=1-FEX2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1577 DXY=XX(IJ)-XX(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1578 DXY=YY(IJ)-YY(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1579 DXY=QTR*(XX(IJ-1)+XX(IJ)+XX(IM2-1)+XX(IM2))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1580 DXY=QTR*(YY(IJ-1)+YY(IJ)+YY(IM2-1)+YY(IM2))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1581 FAC=(DXY<em>DXY</em>DY<em>DY)/(DXY</em>2+DY*2+SMALL)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1582 DEL=EDT(IJ)*FXW1-FEX2+EDT(IJ)*FXW1-EDT(IM2)*FXW2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1583 EDT(IJ-1)=EDT(IJ+1)-DEL*FAC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1584 IJ=I-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1585 GO TO 1115</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1586 1114 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1587 SU(IJ)+GREAT*ED(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1588 BP(IJ)=GREAT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1589 1115 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1590 AS(IJ)=0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1591 1110 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1592 C---- NORTH BOUNDARY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1593 DO 1120 I=2,NJM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1594 IJ=IM2(IJ)+NJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1595 GO TO 1121,1122,1123,1124,1127</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1596 1121 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1597 SU(IJ)+SU(IJ)+AN(IJ)*EDT(IJ+1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1598 BP(IJ)+BP(IJ)+AN(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1599 GO TO 1125</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1600 1122 EDT(IJ)+EDT(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1601 GO TO 1125</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1602 1123 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1603 IJ=I+1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1604 IP=I+1,NJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1605 IM2=I+1,NJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1606 FEX1=FX(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1607 FEX2=FX(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1608 FXW1=1-FEX1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1609 FXW2=1-FEX2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1610 DXY=XX(IJ)-XX(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1611 DXY=YY(IJ)-YY(IM2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1612 DXY=QTR*(XX(IJ-2)+XX(IJ)+XX(IM2-2)+XX(IM2))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1613 DXY=QTR*(YY(IJ-2)+YY(IJ)+YY(IM2-2)+YY(IM2))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1614 FAC=(DXY<em>DXY</em>DY<em>DY)/(DXY</em>2+DY*2+SMALL)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1615 DEL=EDT(IJ)*FXW1-FEX2+EDT(IJ)*FXW1-EDT(IM2)*FXW2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1616 EDT(IJ-1)=EDT(IJ+1)-DEL*FAC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1617 IJ=I+1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1618 GO TO 1125</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1619 1124 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1620 SU(IJ)+GREAT*ED(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1621 BP(IJ)=GREAT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1622 1125 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1623 AN(IJ)=0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1624 1120 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1625 C---- WEST BOUNDARY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1626 DO 1130 J=2,NJM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1627 IJ=INM(IJ)+2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1628 GO TO 1131,1132,1133,1134</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1629 1131 CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1630 SU(IJ)+SU(IJ)+AN(IJ)*EDT(IJ-NJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1631 BF(IJ)+BF(IJ)+AN(IJ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1632 GO TO 1135</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1633 1132 EDT(IJ-NJ)=EDT(IJ)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1705 DO 10 I=1,N1
1706 DO 10 J=1,N2
1707 IJ=IMNJ(1)+J
1708 SU(IJ)+SU(IJ)
1709 BP(IJ)=BP(IJ)
1710 10 CONTINUE
1711 C-- CHECK WALL SOUTH BOUNDARY
1712 C
1713 C
1714 DO 500 I=2,NIM
1715 IJ=IMNJ(I)+2
1716 IF(TMRH(I).EQ.4) THEN
1717 LH=IJ
1718 LW=LJ-1
1719 TERR+SQRT(TE(IJ)+TET(IJ))
1720 DELM+DNS(I)
1721 DXB=XX(IJ)-XX(IJ-NJ-1)
1722 DYY=YY(IJ-1)-YY(IJ-NJ-1)
1723 RB=HAF*(R(IJ-1)+R(IJ-NJ-1))
1724 DENS+DEN(IJ)
1725 CALL WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1726 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1727 SU(IJ)+SU(IJ)+SUP
1728 BP(IJ)+BP(IJ)+SUP
1729 SUPV(I)=SUP
1730 SPW(I)=SUP
1731 SWP(I)=SWP
1732 SPW(I)=SUP
1733 GENTM(I)=GENTE
1734 AS(IJ)=0.0
1735 ENDIF
1736 C-- CHECK NORTH WALL BOUNDARY
1737 C
1738 C
1739 IJ=IMNJ(I)+1
1740 IF(TMRH(I).EQ.4) THEN
1741 LB=IJ
1742 LW=LJ-1
1743 TERR+SQRT(TE(IJ)+TET(IJ))
1744 DELM+DNS(I)
1745 DXB=XX(IJ)-XX(IJ-NJ-1)
1746 DYY=YY(IJ-1)-YY(IJ-NJ-1)
1747 RB=HAF*(R(IJ-1)+R(IJ-NJ-1))
1748 DENS+DEN(IJ)
1749 CALL WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1750 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1751 SU(IJ)+SU(IJ)+SUP
1752 BP(IJ)+BP(IJ)+SUP
1753 SUPV(I)=SUP
1754 SPW(I)=SUP
1755 SWP(I)=SWP
1756 SPW(I)=SUP
1757 GENTM(I)=GENTE
1758 AN(IJ)=0.0
1759 ENDIF
1760 C-- CHECK WALL WEST-BOUNDARY
1761 C
1762 C
1763 DO 620 J=2,NJM
1764 IJ=IMNJ(J)+2
1765 IF(TMRH(J).EQ.4) THEN
1766 LB=IJ
1767 LW=LJ-1
1768 TERR+SQRT(TE(IJ)+TET(IJ))
1769 DELM+DNS(I)
1770 DXB=XX(IJ-1)-XX(IJ-NJ-1)
1771 DYY=YY(IJ-1)-YY(IJ-NJ-1)
1772 RB=HAF*(R(IJ-1)+R(IJ-NJ-1))
1773 DENS+DEN(IJ)
1774 CALL WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1775 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1776 CALL WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1777 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1778 SU(IJ)=SU(IJ)+SU
1779 BP(IJ)=BP(IJ)+SUP
1780 SUPW(J)=SUP
1781 SPW(J)=SUP
1782 SWP(J)=SWP
1783 SPW(J)=SUP
1784 GENTM(J)=GENTE
1785 AM(IJ)=0.0
1786 ENDF
1787 C-- CHECK WALL EAST-BOUNDARY
1788 C
1789 C
1790 IJ=IMNJ(NIM)+J
1791 IF(TJBR(J).EQ.4) THEN
1792 LB=IJ
1793 LW=LJ+1
1794 TERR+SQRT(TE(IJ)+TET(IJ))
1795 DELM+DNS(I)
1796 DXB=XX(IJ)-XX(IJ-1)
1797 DYY=YY(IJ)-YY(IJ-1)
1798 RB=HAF*(R(IJ)+R(IJ-1))
1799 DENS+DEN(IJ)
1800 CALL WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1801 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1802 SU(IJ)=SU(IJ)+SU
1803 BP(IJ)=BP(IJ)+SUP
1804 SUV(J)=SUP
1805 SPV(J)=SUP
1806 SPW(J)=SUP
1807 SWP(J)=SUP
1808 GENTM(J)=GENTE
1809 AE(IJ)=0.0
1810 ENDF
1811 C-- SUBROUTINE 'WALLFN' TO SET WALL FUNCTIONS
1812 C
1813 C
1814 RETURN
1815 C
1816 END
1817 C
1818 C
1819 C
1820 INCLUDE 'mskemod.b'
1821 C
1822 SUBROUTINE WALLFN(LB,LW,VISCONS,DENS,DXB,DYY,CMW25,ELOG,CAPP,
1823 & TAU,SU,SU,SU,SU,SU,SWP,SWP,GENTE,DELM,TEPR,RB)
1824 C
1825 INCLUDE 'mskemod.b'
1826 C
1827 UP=U(LB)
1828 VP=V(LB)
1829 WP=W(LB)
1830 UNAL=U(LW)
1831 VNAL=V(LW)
1832 WNALL=W(LW)
1833 ARW=SQRT(DXX**2+DYY**2)
1834 DXX=DXX/ARW
1835 DYY=DYY/ARW
1836 CONST=DENS*CMW25*TEPR
1837 YPLS=DELM*CONST/VISC
1838 TCOEF=VISC/DELM
1839 IF(LAY2) GOTO 10
1840 IF(YPLS.LE.11.63) GO TO 10
1841 UPLUS=LOG(ELOG*YPLS)/CAPP
1842 TCOEF=CONST/UPLUS
1843 10 CONTINUE
1844 VPINT=UP*DXX+VP*DYY
1845 VPX=VPINT*DXX
1846 VFX=VPINT*DYY
## CHAPTER 4

2D/Axisymmetric Algebraic Stress Turbulence Model

### Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Introduction</td>
<td>49</td>
</tr>
<tr>
<td>4.2 Theory and Model Equations</td>
<td>49</td>
</tr>
<tr>
<td>4.3 Module Evaluation</td>
<td>52</td>
</tr>
<tr>
<td>References</td>
<td>53</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix C</td>
<td>54</td>
</tr>
</tbody>
</table>
4.1 Introduction

In this section a description is given of the two-dimensional/Axisymmetric Algebraic Stress turbulence Model (ASM) based on the work of Rodi [1]. The model is coded as a self contained computer program to compute turbulent flow quantities when interfaced with a CFD solver. Detailed description of the module structure, variables used and how to interface the module with CFD flow solvers are given in Appendix C.

The module uses as input the mean flow properties, as computed by conventional CFD solvers, and calculates the Reynolds stresses, turbulent kinetic energy and the energy dissipation. It is structured to be self-contained and compatible with many CFD codes. It has been tested as a separate unit at Rocketdyne using the finite-volume REACT code [2]. The module has also been tested independently at the University of Alabama at Huntsville (UAH) using own code MAST.

The module computes turbulent flow quantities in two-dimensional planar or axisymmetric geometry with or without swirl. The standard wall functions and the two-layer model of Chen and Patel [3] are used for the near wall treatment.

4.2 Theory and Model Equations

The Algebraic Stress (ASM) module is based on the work of Rodi [1]. The idea is to simplify or truncate the Reynolds stress equation by approximating the convective and diffusive transport of the Reynolds stresses \( \overline{u_iu_j} \) in terms of the corresponding transport of turbulent energy. This allows the transport equation for the stresses to be expressed as a set of algebraic formulae containing the turbulence energy and its rate of dissipation as unknowns in the form:

\[
\overline{u_iu_j} = \frac{k}{(P - \epsilon)} \left[ P_{ij} - \frac{2}{3} \delta_{ij} \epsilon + \Phi_{ij} \right]
\]

where \( P_{ij} = \) Production and \( P = \frac{1}{2} P_{kk} \) and

\[
\Phi_{ij} = \text{pressure-strain redistribution}
\]

\[
\Phi_{ij} = \Phi_{ij,1} + \Phi_{ij,2} + \Phi_{ij,1w} + \Phi_{ij,2w}
\]

- 49 -
Rotta's linear return-to-isotropy concept for the non-linear part of

\[ \Phi_{ij,1} = -C_1 \frac{\varepsilon}{k} (\bar{u}_i \bar{u}_j - \frac{2}{3} k \delta_{ij}) \]

is used and the "isotropization of production" concept for the linear "rapid" part of

\[ \Phi_{ij,2} = -C_2 (P_{ij} - \frac{2}{3} P \delta_{ij}) \]

is used. Gibson and Launder [4] concept for the wall reflection terms is used as

\[ \Phi_{ij,1w} = C_{1w} \rho \frac{\varepsilon}{k} (u_k u_m n_k n_m \delta_{ij} - \frac{3}{2} u_k u_i n_k n_j - \frac{3}{2} u_k u_j n_k n_i) f \]

\[ \Phi_{ij,2w} = C_{2w} (\Phi_{km,2} n_k n_m \delta_{ij} - \frac{3}{2} \Phi_{ik,2} n_k n_j - \frac{3}{2} \Phi_{jk,2} n_k n_i) f \]

where \((n_i)\) is the wall-normal unit vector in the \(i\)-direction. The wall-distance function \((f)\) represents the ratio of the turbulence length scale \((L_\varepsilon = \frac{k^{3/2}}{\varepsilon})\) and the wall distance and is given as

\[ f = \left( \frac{C_m^{0.75} k^{1.5}}{K \varepsilon} \right) \frac{1}{\Delta n} \]

with \(\Delta n\) being the wall-normal distance.

The set of algebraic stress equations can be arranged in the form

\[ A_{ij} \bar{u}^2 + B_{ij} \bar{v}^2 + C_{ij} \bar{w}^2 + D_{ij} \bar{u} \bar{v} + E_{ij} \bar{v} \bar{w} + F_{ij} \bar{u} \bar{w} = G_{ij} \]

where \(A_{ij}, B_{ij}, C_{ij}, D_{ij}, E_{ij}, F_{ij},\) and \(G_{ij}\) are functions of the mean and turbulent flow variables.

The above equation can be solved iteratively in the main flow solver. However, the algebraic system of equations is stiff and convergence difficulties are encountered when solved iteratively. Therefore, the set of equations was cast in the general matrix form \(A^T = B\), where
\[ A = \frac{3\varepsilon}{2\lambda k} + 2 \frac{\partial U}{\partial x} - \frac{\partial V}{\partial y} - \frac{V}{r} - 2 \frac{\partial U}{\partial y} - \frac{\partial V}{\partial x} - \frac{\partial W}{\partial y} + \frac{W}{r} - \frac{\partial W}{\partial x} \]

\[ - \frac{\partial U}{\partial x} + \frac{3\varepsilon}{2\lambda k} + 2 \frac{\partial V}{\partial y} - \frac{V}{r} - 2 \frac{\partial V}{\partial y} - \frac{\partial U}{\partial x} - (\frac{\partial W}{\partial y} + \frac{W}{r}) - \frac{\partial W}{\partial x} \]

\[ - \frac{\partial U}{\partial x} - \frac{\partial V}{\partial y} + \frac{3\varepsilon}{2\lambda k} + 2 \frac{r}{\partial x} - (\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}) - 2 \frac{\partial W}{\partial y} + \frac{W}{r} - 2 \frac{\partial W}{\partial x} \]

\[ \frac{\partial V}{\partial x} \quad \frac{\partial U}{\partial y} \quad 0 \quad \frac{\varepsilon}{\lambda k} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} \quad 0 \quad -\frac{W}{r} \]

\[ 0 \quad \frac{\partial W}{\partial y} \quad -\frac{W}{r} \quad \frac{\partial W}{\partial x} \quad \frac{\varepsilon}{\lambda k} + \frac{\partial V}{\partial y} + \frac{V}{r} \quad \frac{\partial V}{\partial x} \]

\[ \frac{\partial W}{\partial x} \quad 0 \quad 0 \quad \frac{\partial W}{\partial y} \quad \frac{\partial U}{\partial y} \quad \frac{\varepsilon}{\lambda k} + \frac{\partial U}{\partial x} + \frac{V}{r} \]

\[ T = \{ \rho \quad u \quad \frac{v}{v} \quad \rho \quad \frac{w}{w} \quad \rho \quad u \quad \frac{v}{v} \quad \rho \quad \frac{w}{w} \quad \rho \quad \frac{w}{w} \} \]

\[ B = \begin{align*}
\frac{\rho \varepsilon}{\lambda} + \frac{3}{2(1-C_2)} (\Phi_{11,1w} + \Phi_{11,2w}) \\
\frac{\rho \varepsilon}{\lambda} + \frac{3}{2(1-C_2)} (\Phi_{22,1w} + \Phi_{22,2w}) \\
\frac{\rho \varepsilon}{\lambda} + \frac{3}{2(1-C_2)} (\Phi_{33,1w} + \Phi_{33,2w}) \\
\frac{1}{(1-C_2)} (\Phi_{12,1w} + \Phi_{12,2w}) \\
\frac{1}{(1-C_2)} (\Phi_{23,1w} + \Phi_{23,2w}) \\
\frac{1}{(1-C_2)} (\Phi_{13,1w} + \Phi_{13,2w})
\end{align*} \]

where \[ \lambda = \frac{1-C_2}{C_1 + \frac{P}{\rho \varepsilon}} \]
The matrix was inverted at each iteration step to obtain a converged solution. The wall function and a two-layer model were built in the module to model the near-wall region.

4.3 Module Evaluation

The ASM module was evaluated by comparison with experimental data of Driver and Seegmiller [5] for the backward facing step and the data of Roback and Johnson [6]. The effect of the wall reflection term is also studied with both wall function and two-layer near wall models. Figures 1a and 1b show the stream-function contours for a backward facing step flow using the wall function and the two-layer near wall models with reattachment length of 5.59H and 5.83H respectively (H is the step height). Figures 2a and 2b show the stream-function contours for the Roback & Johnson confined swirling jet flow using the wall function near wall model, where (a) includes the wall-reflection term in the pressure-strain redistribution term and (b) without the wall reflection term. Figure 3 shows a comparison of the axial velocity along the centerline with and without wall reflection term. The comparisons of the predicted mean axial velocity, mean tangential velocity, turbulent intensities $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$, and the Reynolds stress $\overline{uw'}$ using the ASM model as compared with the single and multi-scale $k$-$\epsilon$ models are presented in figures 4 to 9 respectively. The figures in general show that the ASM model used here when combined with the wall function near wall treatment predicts better comparisons without using the wall reflection terms. This may be explained by the fact that the wall reflection terms -whose purpose is to damp normal turbulent intensity normal to the wall as the wall is reached- are not effective when using wall functions near the wall. Similar conclusions were also obtained by the UAH group when testing the ASM module using their code (MAST). Also, in the ASM model, a set of algebraic equations for the Reynolds stresses are solved and there is no boundary conditions are needed for the stresses. This is not the same in the full Reynolds stress model (RSM) where a set of nonlinear differential equations are solved and boundary conditions for the stresses are required. More on this will be discussed in detail in the next RSM module. Also, more details will be given on the tensorial incorporation of the wall reflection terms since they are tied to the orientation of the wall through the unit normal vectors.
REFERENCES


Figure 1. Stream-function contours of backward facing step flow using the ASM with (a) wall function and (b) two-layer near wall treatment
Figure 2. Stream-function contours of confined swirling jet flow
Figure 3. Decay of axial velocity along the centerline in confined swirling jet flow

- o Roback & Johnson
- ---- ASM no $\Phi_w$
- ----- ASm with $\Phi_w$
Figure 4. Radial profiles of mean axial velocity in confined swirling jet flow

- M-S model. — ASM. —- k-ε model
Figure 5. Radial profiles of mean tangential velocity in confined swirling jet flow

— M-S model. —— ASM. —— k-ε model
Figure 6. Radial profiles of turbulent intensity ($\bar{u}u$)$^{1/2}$ in confined swirling jet flow
Figure 7. Radial profiles of turbulent intensity $\sqrt{\langle \nu^2 \rangle}$ in confined swirling jet flow.
Figure 8. Radial profiles of turbulent intensity $(\overline{w^2})^{1/2}$ in confined swirling jet flow.
Figure 9. Radial profiles of turbulent shear stress $\overline{uw}$ in confined swirling jet flow
APPENDIX C

2D/Axisymmetric Algebraic Stress Turbulence Module Deck

This module is a FORTRAN source code to solve 2D/Axisymmetric turbulent flow quantities using the algebraic stress model when interfaced with a main flow solver. The module consists of the main routine ASMOD that calls a number of subroutines to perform different functions that will be explained below.

3.1 Subroutine ASMOD

This is basically the main routine that reads through its argument list different variables from the calling flow solver which are described below.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Grid node locations in the x or $\xi$-direction, dimensioned to X(NX*NY)</td>
</tr>
<tr>
<td>Y</td>
<td>Grid node locations in the y or $\eta$-direction, dimensioned to Y(NX*NY)</td>
</tr>
<tr>
<td>FX</td>
<td>Interpolation factor in the x or $\xi$-direction.</td>
</tr>
<tr>
<td>FY</td>
<td>Interpolation factor in the y or $\eta$-direction.</td>
</tr>
<tr>
<td>ARE</td>
<td>Cell areas</td>
</tr>
<tr>
<td>VOL</td>
<td>Cell volumes.</td>
</tr>
<tr>
<td>R</td>
<td>Radial distance in the axisymmetric geometry or 1. for planar geometry.</td>
</tr>
<tr>
<td>DNS</td>
<td>Normal distance of a cell from the south-boundary dimensioned to NX.</td>
</tr>
<tr>
<td>DNN</td>
<td>Normal distance of a cell from the north-boundary dimensioned to NX.</td>
</tr>
<tr>
<td>DNE</td>
<td>Normal distance of a cell from the east-boundary dimensioned to NY.</td>
</tr>
<tr>
<td>DNW</td>
<td>Normal distance of a cell from the west-boundary dimensioned to NY.</td>
</tr>
<tr>
<td>U</td>
<td>Axial or x-direction velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>V</td>
<td>Radial or y-direction velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>W</td>
<td>Tangential or azimuthal velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>TE</td>
<td>Turbulent kinetic energy, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>ED</td>
<td>Turbulent energy dissipation rate, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>DEN</td>
<td>Density (assumed constant for incompressible flows).</td>
</tr>
<tr>
<td>F1</td>
<td>Mass flux at cell faces in the x or $\xi$-direction, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>F2</td>
<td>Mass flux at cell faces in the y or $\eta$-direction, dimensioned to NX*NY.</td>
</tr>
</tbody>
</table>
VISCOUS  Laminar viscosity.
VIS       Eddy viscosity, dimensioned to NX*NY.
RESOR     Residual error for the \(k\) and \(\varepsilon\) -equations solver, dimensioned to 2.
ITBS      Boundary condition flag along the south boundary dimensioned to NX and must have one for each boundary node set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall, e.g., for a wall boundary condition along the south boundary set ITBS to NX*4. Similarly for the other boundaries.
ITBN      Boundary condition flag along the north boundary, dimensioned to NX.
JTBE      Boundary condition flag along the east-boundary dimensioned to NY.
JTBW      Boundary condition flag along the west-boundary dimensioned to NY.
ITER      Iteration number.
FMUU      function used in the two-layer model.
ICAL      = 1 for swirl velocity calculations, 0 otherwise.
AKSI      = 1 for axisymmetric flow, 0 otherwise.
RESTART   = 1 if calculations are restarted from a previous run, 0 otherwise.

ASMOD starts by reading the turbulent flow constants, under-relaxation factors and Prandtl/Schmidt numbers for the \(k\) and \(\varepsilon\) equations. These are:

CD1, CD2  constants in the \(k\) and \(\varepsilon\) -equations and are usually set to 1.44 and 1.92 respectively.
CMU, ELOG, and CAPPA constants in the \(k\) and \(\varepsilon\) -equations and are usually set to 0.09, 9.8 and 0.42 respectively.
LAY2      set to true (T) for two-layer model and false (F) for wall functions.
GKE       is set to 1 for second-order upwinding of the convective terms in the \(k\) and \(\varepsilon\) -equations.
ALFAKE    is the iteration parameter used in the \(k\) and \(\varepsilon\) -equation solver.
URFVIS    is the underrelaxation factor of the viscosity near the wall.
SORKE(1) and SORKE (2) are the degree of accuracy for the \(k\) and \(\varepsilon\) -equation solver respectively.
URFKE(1) and URFKE(2) are the underrelaxation factors for the \(k\) and \(\varepsilon\) -equations respectively.
PRTKE(1) and PRTKE(2) are ratio of Prandtl to Schmidt numbers used in the \(k\) and \(\varepsilon\) -equations in the two-layer model near the wall.
C1, C2    are constants in the ASM model.
C1W and C2W are the two constants in the wall-reflection terms of the pressure-strain redistribution term.

CK and CE constants in the diffusion term of the \( k \) and \( \varepsilon \) -equations.

WREFON = 1 if the wall reflection terms of the pressure-strain term are to be included, 0 otherwise.

All dimensions considered are one-dimensional. The position of any node is defined as \( \text{I,J} = (l,J) = (I-1)*NJ + J \), where \( NI \) and \( NJ \) are the number of grid nodes in the \( X \) and \( Y \)-directions respectively. It is assumed that grid related data such as cell areas, volumes and interpolation factors be passed to the module from an external grid generator.

**Subroutine WALREF**

This subroutine calculates the wall reflection terms in the pressure-strain redistribution term. It calculates the wall unit normal vectors and the normal distance away from the wall. This is needed to resolve the wall tangential and normal velocity components that are needed to obtain the near-wall values of the Reynolds stresses.

**Subroutine CALPIIJ**

This subroutine calculates the production terms of the individual stress components.

**Subroutine CALUIUJ**

This subroutine calculates the individual stress component from its algebraic equation. It sets the coefficients of the algebraic stress equations which are solved implicitly at each iteration step by inverting a 6x6 matrix.

**Subroutine ACALCKE**

This subroutine solves the transport equations for the turbulent energy \((\text{IPHI}=1)\) and energy dissipation.\((\text{IPHI}=2)\). Daly and Harlow [7] gradient stress diffusion form is used in the module instead of the simplified isotropic diffusivity form. The subroutine calls MODPHI subroutine that sets the appropriate boundary conditions for \( k \) and \( \varepsilon \). The set of algebraic difference equations are then solved using Stone's strongly implicit solver ASOLSIP.

- 56 -
Subroutine ATWOLAY

This subroutine calculates the near wall turbulence using Chen and Patel's [3] two layer model.

Subroutine MODPIJ

This subroutine modifies the production terms near the wall using the near wall region model.

Subroutine MODPHI

This subroutine calculates the near wall boundary conditions for the turbulence energy and the energy dissipation.

Subroutine AMODVIS

This subroutine modifies the eddy viscosity close to a wall using the near wall model chosen.

Subroutine ASOLSIP

This subroutine solves the system of linear algebraic equations for $k$ and $\varepsilon$ using Stone's Implicit Procedure [8].

Subroutine AMODIFY

This subroutine is called from the momentum equations solver of the main routine. It updates the flux source terms of the discretized momentum equations due to wall shear stresses and due to the Reynolds stress gradients. The terms SUASM, SVASM and SWASM need to be added to the U, V and W-momentum equations of the main solver. They represent the difference form of the Reynolds stress gradients in the momentum equations.
E E o -Jo E 

 INCLUDE 'gridparam.h'
 INCLUDE 'asm.h'

 DIMENSION X(NXNY), Y(NXNY), FX(NXNY), FY(NXNY), 
 & ARE(NXNY), VOL(NXNY), & (NXNY), 
 DIMENSION DMS(NX), DNM(NX), DNE(NX), DNW(NX), 
 DIMENSION FTS(NX), FTN(NX), FTE(NX), JTBW(NY), 
 DIMENSION E(NXNY), V(NXNY), W(NXNY), D(NXNY), 
 & F1(NXNY), F2(NXNY), FMU(NXNY), 
 DIMENSION PHI(NXNY), RESOR(2), TERS(NXNY).

 NI=NIM+1
 NJ=NNJ+1

 IF (ITER.EQ.1) THEN

 REWIND 41

 READ(41,*), C1, C2, C3, C4, BLOGC, CAPPF
 READ (41,*), CL, LAY2
 READ (41,*), GKE, ALF, AKE, URFVIS
 READ (41,*), D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, D16, D17, D18, D19, D20, D21, D22, D23, D24, D25, D26, D27, D28, D29, D30, D31, D32, D33, D34, D35, D36, D37, D38, D39, D40, D41, D42, D43, D44, D45, D46, D47, D48, D49, D50, D51, D52, D53, D54, D55, D56, D57, D58, D59, D60, D61, D62, D63, D64, D65, D66, D67, D68, D69, D70, D71, D72, D73, D74, D75, D76, D77, D78, D79, D80, D81, D82, D83, D84, D85, D86, D87, D88, D89, D90, D91, D92, D93, D94, D95, D96, D97, D98, D99, D100, D101, D102, D103, D104, D105, D106, D107, D108, D109, D110, D111, D112, D113, D114, D115, D116, D117, D118, D119, D120, D121, D122, D123, D124, D125, D126, D127, D128, D129, D130, D131, D132, D133, D134, D135, D136, D137, D138, D139, D140, D141, D142.
Eg.

"E

IIDI

ml

Z

&

_u

_o

_mo

ed

"0

O_

I

0

E

m) o.

,
Oct 12, 1996 16:36  asmod_2d  Page 11

`DXE=X(IJ)-X(IJM)`  710
`DYE=Y(IJ)-Y(IJM)`  711
`DSN=X(IJ)-X(IJM)`  712
`DYN=Y(IJ)-Y(IJM)`  713
`DKS=QTR(X(IJ))-X(IJM)`  714
`DYQ=Y(IJ)-Y(IJM)`  715
`DVY=V(IJ)-V(IJM)`  716
`DVX=V(IJ)-V(IJM)`  717
`DSQ=QTR(Y(IJ))-Y(IJM)`  718
`DQY=Y(IJ)-Y(IJM)`  719
`DQX=QTR(Y(IJ))-Y(IJM)`  720
`DQV=QTR(Y(IJ))-Y(IJM)`  721
`AREH=H(ARE(IJ)+ARE(IJM))`  722
`ARSH=H(AR(ARE(IJ)+ARE(IJM)))`  723
`CRK=CR`  724
`I(LH,IQ,EQ,2)k=0`  725
`& ARE(IJ)+ARE(IJM)`  726
`& ARE(IJ)+ARE(IJM)`  727
`& ARE(IJ)+ARE(IJM)`  728
`& ARE(IJ)+ARE(IJM)`  729
`& ARE(IJ)+ARE(IJM)`  730
`& ARE(IJ)+ARE(IJM)`  731
`& ARE(IJ)+ARE(IJM)`  732
`& ARE(IJ)+ARE(IJM)`  733
`& ARE(IJ)+ARE(IJM)`  734
`& ARE(IJ)+ARE(IJM)`  735
`& ARE(IJ)+ARE(IJM)`  736
`& ARE(IJ)+ARE(IJM)`  737
`& ARE(IJ)+ARE(IJM)`  738
`& ARE(IJ)+ARE(IJM)`  739
`& ARE(IJ)+ARE(IJM)`  740
`& ARE(IJ)+ARE(IJM)`  741
`& ARE(IJ)+ARE(IJM)`  742
`& ARE(IJ)+ARE(IJM)`  743
`& ARE(IJ)+ARE(IJM)`  744
`& ARE(IJ)+ARE(IJM)`  745
`& ARE(IJ)+ARE(IJM)`  746
`& ARE(IJ)+ARE(IJM)`  747
`& ARE(IJ)+ARE(IJM)`  748
`& ARE(IJ)+ARE(IJM)`  749
`& ARE(IJ)+ARE(IJM)`  750
`& ARE(IJ)+ARE(IJM)`  751
`& ARE(IJ)+ARE(IJM)`  752
`& ARE(IJ)+ARE(IJM)`  753
`& ARE(IJ)+ARE(IJM)`  754
`& ARE(IJ)+ARE(IJM)`  755
`& ARE(IJ)+ARE(IJM)`  756
`& ARE(IJ)+ARE(IJM)`  757
`& ARE(IJ)+ARE(IJM)`  758
`& ARE(IJ)+ARE(IJM)`  759
`& ARE(IJ)+ARE(IJM)`  760
`& ARE(IJ)+ARE(IJM)`  761
`& ARE(IJ)+ARE(IJM)`  762
`& ARE(IJ)+ARE(IJM)`  763
`& ARE(IJ)+ARE(IJM)`  764
`& ARE(IJ)+ARE(IJM)`  765
`& ARE(IJ)+ARE(IJM)`  766
`& ARE(IJ)+ARE(IJM)`  767
`& ARE(IJ)+ARE(IJM)`  768
`& ARE(IJ)+ARE(IJM)`  769
`& ARE(IJ)+ARE(IJM)`  770
`& ARE(IJ)+ARE(IJM)`  771
`& ARE(IJ)+ARE(IJM)`  772
`& ARE(IJ)+ARE(IJM)`  773
`& ARE(IJ)+ARE(IJM)`  774
`& ARE(IJ)+ARE(IJM)`  775
`& ARE(IJ)+ARE(IJM)`  776
`& ARE(IJ)+ARE(IJM)`  777
`& ARE(IJ)+ARE(IJM)`  778
`& ARE(IJ)+ARE(IJM)`  779
`& ARE(IJ)+ARE(IJM)`  780

Oct 12, 1996 16:36  asmod_2d  Page 12

`SNSE=-(GAMNU+DYK*DS+GAMNU+DXK*DS)*`  781
`& (PHI-PHI)*2.0+GAMNU+DYK*DS)*`  782
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  783
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  784
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  785
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  786
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  787
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  788
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  789
`& (PHI-PHI)*2.0+GAMNU+DXK*DS)*`  790

Printed by yq449 from mtbg
852 30 CONTINUE
854 1 DO 40 J=2,NJM
855 IF (JTBW(J),EQ,4,OR,JBTH(J),EQ,4) THEN
856 DO 410 I=2,NIM
857 IF(1&=IMNJ(I)+1) THEN
858 IF(FMU(I,JL),LT,0.95) THEN
859 SU(I,J)=GREAT*ED(I,J)
860 BP(I,J)=GREAT
861 END IF
862 410 CONTINUE
863 40 CONTINUE
864 ENDIF
865 END
866 C
867 C
868 C DO 200 I=2,NIM
869 DO 200 J=2,NJM
870 IJ=IMNJ(I)+J
871 AP(I,J)=AH(I,J)+AE(I,J)+AN(I,J)+AS(I,J)+BP(I,J)
872 AP(I,J)=AP(I,J)*URPHI
873 SU(I,J)=SU(I,J)+1.0-URPHI(IIKHI)*AP(I,J)*PHI(I,J)
874 200 CONTINUE
875 200 CONTINUE
876 C
877 C ----- SOLVING F.D. EQUATIONS
878 C
879 CALL ASOLSIP (PHI,PHI,RESOR)
880 C
881 C
882 C
883 C IF (PHI(I,J).LT.0.) PHI(I,J)=ABS(PHI(I,J))
884 END DO
885 RETURN
886 END
887 C
888 C
889 C --------------- SUBROUTINE ATWOLAY(X,Y,DEP,TE,ED,DISW,ITWS,ITBN,JBTH,JTBW)
890 C
891 C INCLUDE 'gridparam.h'
892 C INCLUDE 'asm.h'
893 C
894 C
895 C DIMENSION X(NNXY), Y(NNXY), DEN(NNXY)
896 C DIMENSION TE(NNXY), ED(NNXY)
897 C DIMENSION ITWS(NX), ITBN(NY), JTBW(NY)
898 C
899 C
900 C NI = NIM + 1
901 NJ = NJM + 1
902 C
903 C
904 C MU25 = SQRT(SQRT(CMU))
905 CYTM = CMU25**2
906 C11 = CAPPA/CMU25
907 AED=2.0*C11
908 AMG=70.0
909 C
910 C DO 10 I=2,NIM
911 C DO 110 J=2,NJM
912 C DISG=DISG+1
913 C DISG=DISG
914 C DISG=DISG+1
917 CJ=IMNJ(I)+J
918 C C ... CHECK THE SOUTH BOUNDARY
919 C C CHECK THE SOUTH BOUNDARY
920 C IF(ITBS(I),EQ,4) THEN
921 C LNW=IMNJ(I)+1
1136  GO TO 825
1137  824 CONTINUE
1138  IF (.NOT. LA72) GEN(IJ) = GENMIN(IJ)
1139  818 CONTINUE
1140  GO TO 825
1141  CONTINUE
1142  AN(IJ) = 0.0
1143  820 CONTINUE
1144  ----- WEST BOUNDARY
1145  DO 810 J = 2, NJM
1146  811 IJ = IMNJ(J) + 2
1147  GO TO (831, 832, 833, 834) JTBW(JJ)
1148  831 CONTINUE
1149  SU(IJ) = SU(IJ) + AW(IJ)*TE(IJ-NJ)
1150  832 CONTINUE
1151  HI(IJ-NJ) = TE(IJ)
1152  833 CONTINUE
1153  GO TO 835
1154  834 CONTINUE
1155  IJ = J
1156  IJ = IJ + 1
1157  IJ = IJ + 1
1158  FYNL-FYNJ
1159  FYN2-FYNJ
1160  FYS1-1- FYN1
1161  FYS2-1- FYN2
1162  DVB-Y(IJ)-Y(IJ)
1163  DXB=ABS(X(IJ)-X(IJ))
1164  DXPQ=QTR(X(IJ-NJ)-X(IJ)+X(IJ-NJ)-X(IJ))
1165  DXPQ=QTR(Y(IJ)-Y(IJ)+Y(IJ)-Y(IJ))
1166  FAC = (DB-DBP+DBP)/DBX2+2+DBX*2/SMALL
1167  DEL-TE(IJ)*FYS1-FYNJ*TE(IJ)*FYN1-TE(IJ)*FYS2
1168  PHI(IJ)-TE(IJ-NJ)-DEL*FAC
1169  IJ = IJ + NJ
1170  GOTO 835
1171  836 CONTINUE
1172  IF (.NOT. LA72) GEN(IJ) = GENMIN(IJ)
1173  818 CONTINUE
1174  SU(IJ) = SU(IJ) + GEN(IJ) * VOL(IJ)
1175  835 CONTINUE
1176  AW(IJ) = 0.0
1177  830 CONTINUE
1178  ----- EAST BOUNDARY
1179  DO 840 J = 2, NJM
1180  841 IJ = 1+NMNJ(J)
1181  GO TO (831, 832, 833, 844) JTBW(JJ)
1182  841 CONTINUE
1183  SU(IJ) = SU(IJ) + AW(IJ)*TE(IJ+NJ)
1184  842 CONTINUE
1185  BP(IJ) = BP(IJ) + AW(IJ)
1186  840 CONTINUE
1187  GO TO 845
1188  843 CONTINUE
1189  IJ = IJ-NJ
1190  IJ = IJ + 1
1191  IJ = IJ + 1
1192  FYN1-FYNJ
1193  FYN2-FYNJ
1194  FYS1-1- FYN1
1195  FYSZ-1-FYN2
1196  DXB=X(IJ)-X(IJ)
1197  DXPQ=QTR(X(IJ-NJ)-X(IJ)+X(IJ-NJ)-X(IJ))
1198  DXPQ=QTR(Y(IJ)-Y(IJ)-Y(IJ)-Y(IJ))
1199  FAC = (DB-DBP+DBP)/DBX2+2+DBX*2/SMALL
1200  DEL-TE(IJ)*FYS1-FYNJ*TE(IJ)*FYN1-TE(IJ)*FYS2
1201  PHI(IJ)-TE(IJ-NJ)-DEL*FAC
1202  IJ = IJ-NJ
1203  GO TO 845
1204  844 CONTINUE
1205  IF (.NOT. LA72) GEN(IJ) = GENMIN(IJ)
1420 C 5 CONTINUE
1421 
1422 C
1423 C
1424 
1425 DO 10 I=2,NIM
1426 DO 10 J=2,NJM
1427 IJ=IN2(I,J)-J
1428 API=1.0/AF(J)
1429 AP(I,J)=1.0
1430 AE(I,J)=AE(I,J)*API
1431 AM(I,J)=AM(I,J)*API
1432 AN(I,J)=AN(I,J)*API
1433 AS(I,J)=AS(I,J)*API
1434 SU(I,J)=SU(I,J)*API
1435 10 CONTINUE
1436 C
1437 DO 20 I=2,NIM
1438 DO 20 J=2,NJM
1439 IJ=IN2(I,J)+J
1440 IM2=IJ-1
1441 IM3=IJ-2
1442 BN(I,J)=AM(I,J)/1.0+ALFAKE*BN(IJ-JNJ)
1443 BS(IJ)=AS(IJ)/1.0+ALFAKE*BE(IJ)
1444 POM1=ALFAKE*BN(IJ)*BN(IM3)
1445 POM2=ALFAKE*BS(IJ)*BN(IM2)
1446 BP(IJ)=API*POM1+POM2-BN(IJ)*BE(IM3)+BH(IM3)*BN(IM2)
1447 BN(IJ)=1.0+AN(IJ)-POM1/BP(IJ)+SMALL
1448 BS(IJ)=1.0+AS(IJ)-POM2/BP(IJ)+SMALL
1449 20 CONTINUE
1450 C
1451 DO 100 L=1,NSWPKE(I PHI)
1452 RES=0
1453 DO 30 I=2,NIM
1454 DO 30 J=2,NJM
1455 IJ=IN2(I,J)+J
1456 RES(IJ)=AM(IJ)*PHI(IJ+1)+AS(IJ)*PHI(IJ-1)+AE(IJ)*PHI(IJ)+AM(IJ-1)*PHI(IJ)+AM(IJ+1)*PHI(IJ)+AM(IJ)*PHI(IJ)
1457 RES=RES+RES(IJ)/RES(IJ)+SMALL
1458 RES=RES(IJ)+RES(IJ)+RES(IJ)+RES(IJ)+RES(IJ)+RES(IJ)
1459 
1460 C
1461 30 CONTINUE
1462 C
1463 IF(L.EQ.1) RESORKE(IPH)=RESORKE
1464 RSM=RESORKE(I PHI)+RESORKE
1465 DO 40 I=2,NIM
1466 I=IMM+2-I
1467 DO 40 J=2,NJM
1468 JJ=NNM+2-J
1469 IJ=IN2(I,J)+J
1470 RES(IJ)=RES(IJ)*BH(IJ)*BE(IJ)*RES(IJ)*RES(IJ)+RES(IJ)
1471 PH1(IJ)+PH2(IJ)+RES(IJ)
1472 40 CONTINUE
1473 IF(RESORKE.EQ.ZERO) RETURN
1474 IF(RESORKE.GE.ZERO) GOTO 200
1475 100 CONTINUE
1476 C
1477 IF(RESORKE.GE.ZERO) RETURN
1478 2 FORMAT(10X,' SOLSIP DID NOT CONVERGE ') 1479 C
1480 CONTINUE
1481 AUX1=0
1482 AUX2=0
1483 DO 50 I=2,NIM
1484 DO 50 J=2,NJM
1485 IJ=IN2(I,J)+J
1486 AUX1=AUX1+ABS(Phi(IJ)-PH(IJ))
1487 AUX2=AUX2+ABS(Phi(IJ))
1488 50 CONTINUE
1489 C
1490 RESOR(IPH)=AUX1/AUX2
E E 0 O'J
O"O*E
E'n
'o
'D
d.O
,m.
(D
O
O_0
O.
.o
4
_4
O_4
4
I
4
O
E(n

---SOUTH BOUNDARY---

DO 600 I=1,2

IQ=IMDL(1)+2

IF(ITBS(I).EQ.4) THEN

DXY=X(IJ-1)-X(IJ-NJ-1)

DY=Y(IJ-1)-Y(IJ-NJ-1)

ARM=SQRT(DBB**2+DYY**2)

DXY=DBX/ARW

DYY=DAY/ARW

CONST=DEN(IJ)*CMU25*SQRT(TE(IJ))

VPLS=DNS(IJ)*CONST/VSICS

IF(VPLS.LE.11.63.OR.LAY2) THEN

TCODE=VSICS/DNS(IJ)

ELSE

UPLUS=LOG(ELOG*VPLS)/CAPPA

TCODE=CONST/UPLOS

ENDIF

VPINT=U(IJ)*DBX+V(IJ)*DYB

VPINT=VPINT+W(IJ)

VPINT=ABS(VPINT-SQRT(U(IJ-1)*U(IJ-1)+W(IJ-1)*W(IJ-1))

GENTN=TCODE*CONST/ABS(VPINT)/(CAPPA*DEN(IJ)*DNS(IJ))

---NORTH BOUNDARY ---

IF(ITBS(I).EQ.4) THEN

DXY=X(IJ-1)-X(IJ-NJ)

DY=Y(IJ-1)-Y(IJ-NJ)

ARM=SQRT(DBB**2+DYY**2)

DXY=DBX/ARW

DYY=DAY/ARW

CONST=DEN(IJ)*CMU25*SQRT(TE(IJ))

VPLS=DNS(IJ)*CONST/VSICS

IF(VPLS.LE.11.63.OR.LAY2) THEN

TCODE=VSICS/DNS(IJ)

ELSE

UPLUS=LOG(ELOG*VPLS)/CAPPA

TCODE=CONST/UPLOS

ENDIF

VPINT=U(IJ)*DBX+V(IJ)*DYB

VPINT=VPINT+W(IJ)

VPINT=ABS(VPINT-SQRT(U(IJ-1)*U(IJ-1)+W(IJ-1)*W(IJ-1))

GENTN=TCODE*CONST/ABS(VPINT)/(CAPPA*DEN(IJ)*DNS(IJ))
# CHAPTER 5

2D/Axisymmetric Full Reynolds Stress (RSM) Turbulence Model

Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction</td>
<td>59</td>
</tr>
<tr>
<td>5.2 Theory and Model Equations</td>
<td>59</td>
</tr>
<tr>
<td>5.3 Boundary Conditions</td>
<td>67</td>
</tr>
<tr>
<td>5.4 Numerical Procedure</td>
<td>68</td>
</tr>
<tr>
<td>5.4.1 Wall Reflection Treatment</td>
<td>71</td>
</tr>
<tr>
<td>5.6 Module Evaluation</td>
<td>72</td>
</tr>
<tr>
<td>References</td>
<td>74</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix D</td>
<td>76</td>
</tr>
</tbody>
</table>
5.1 Introduction

This report describes a self contained FORTRAN source code to compute turbulent quantities using Launder, Reece and Rodi's [1] second order closure, Reynolds stress model. The module deck is designed to interface with a number of flow solvers to analyse incompressible turbulent internal flows. Detailed description of the model used is given with a special emphasis on the coupling of the mean velocity and Reynolds stresses in the discretization procedure of the generalized coordinate system using a co-located finite volume method. The module was interfaced with the REACT flow solver and tested with benchmark flows including the backward-facing step. The module was also successfully interfaced with the MAST code at the University of Alabama at Huntsville (UAH) and independently tested. The Reynolds stress model implemented produced consistently more accurate simulations than the standard $k-\varepsilon$ model.

5.2 Theory and Model Equations

The flow is considered planar or axially symmetric, steady with constant fluid properties. Its mean field may be described by a two-dimensional time averaged equations of continuity and momentum, which can be written as:

$$\frac{\partial \rho U}{\partial x} + \frac{1}{r} \frac{\partial \rho r V}{\partial r} = 0$$

(1)

$$\frac{\partial (\rho U \Phi)}{\partial x} + \frac{\partial (\rho V \Phi)}{\partial r} = \frac{\partial}{\partial x} (\rho \frac{\partial \Phi}{\partial x}) + \frac{\partial}{\partial r} (\rho \frac{\partial \Phi}{\partial r}) + r \Phi$$

(2)

$\Phi$ stands for any of the dependent variables, namely, $U$ and $V$ (axial and radial velocities respectively) and $rW$ (radial distance $r$ multiplied by the tangential velocity $W$). $\rho$ is the fluid density, $\mu$ is the laminar viscosity. $S_\Phi$ is the source term for the variable $\Phi$ and is given by:

- Axial direction, $\Phi = U$ and $S_U = -\frac{\partial \rho}{\partial x} - \frac{\partial \rho \mu^2}{\partial x} - \frac{1}{r} \frac{\partial \rho r u}{\partial r}$

- Radial direction, $\Phi = V$ and $S_V = -\frac{\partial \rho}{\partial r} + \frac{\rho W^2}{r} - \frac{2 \mu V}{r^2} - \frac{1}{r} \frac{\partial r u}{\partial r} - \frac{\partial \rho r v}{\partial x} + \frac{r w^2}{r}$

- Tangential direction, $\Phi = rW$ and $S_W = -\frac{2 \mu}{r} \frac{\partial r W}{\partial r} - \rho \frac{r u}{\partial x} - \rho \frac{r v}{\partial r} - 2 r w$. 

- 59 -
where $u, v$ and $w$ are the fluctuating velocity components in the axial, radial and azimuthal directions respectively.

Turbulence wall effects in the module are represented by Gibson and Launder [2] version of the high Reynolds number stress transport closure of Launder, Reece and Rodi [1]. The stress closure consists essentially of modeled transport equations for the stresses $\overline{u_iu_j}$ and for axisymmetric swirling flow it includes all the six stresses $\overline{u^2}, \overline{v^2}, \overline{w^2}, \overline{uv}, \overline{uw}$ and $\overline{vw}$.

The set of differential equations governing the transport of Reynolds stresses ($\overline{u_iu_j}$) is obtained from Navier-Stokes equations by multiplying the equations for the fluctuating components $(u_i)$ and $(u_j)$ by $(u_i)$ and $(u_j)$ respectively, then summing these equations and time averaging the results. The resulting Reynolds stress transport equations are then solved using the mean flow equations to obtain the mean and turbulent flow quantities.

The full transport equations for the Reynolds stresses can be written in a compact form using Cartesian tensor representation as;

$$\frac{1}{r} \frac{\partial}{\partial x_k} \left( r \frac{\partial}{\partial x_k} \overline{u_iu_j} \right) - \frac{1}{r} \frac{\partial}{\partial x_k} \left( rC_k \rho \left( \frac{\partial}{\partial x_l} \frac{\partial}{\partial x_i} \right) + k \frac{\partial}{\partial x_l} \right) = P_{ij} + D_{ij} + \Phi_{ij} - \epsilon_{ij} \quad (3)$$

Where $U_k$ are the mean velocity components in $x_k$ direction. The right hand side contains the production term $P_{ij}$ given as

$$P_{ij} = - \rho \left( \overline{u_iu_k} \frac{\partial U_j}{\partial x_k} + \overline{u_ju_k} \frac{\partial U_i}{\partial x_k} \right) \quad (4)$$

$P_{ij}$ does not require approximations since it is fully represented by turbulent stresses and mean flow gradients.

The dissipation correlation $\epsilon_{ij}$ arise from the fine-scale of the turbulent motion. At high Reynolds numbers these scales are many orders of magnitude smaller than the large energy containing eddies and turbulence energy cascades down along the eddy-size range with little linkage occurring at intermediate scales, to be ultimately dissipated by the smallest eddies which are unaware of the nature of the mean flow and the large scale turbulence. Therefore, the structure of these fine scale motions responsible for viscous dissipation is isotropic and the dissipation tensor $\epsilon_{ij}$ reduces to
An additional equation for the dissipation $\varepsilon$ is required.

$D_{ij}$ represents the Reynolds stress diffusion which does not in general contribute greatly to the balance of transport of $\overline{u_i u_j}$ except in regions of low stress production by mean strain. This term include contributions of fluctuating pressure-velocity correlations ($\overline{pu_i}$ and $\overline{pu_j}$), triple correlations $\overline{u_i u_j u_k}$ and viscous diffusion $\nu \frac{\partial \overline{u_i u_j}}{\partial x_k}$. Daly and Harlow [3] proposed a simple gradient diffusion hypothesis to model the stress diffusion term in the form

$$D_{ij} = C_s \frac{\partial}{\partial x_k} \rho \frac{k}{\varepsilon} \frac{\partial}{\partial x_l} \overline{u_k u_l} \frac{\partial \overline{u_i u_j}}{\partial x_k}$$

with constant $C_s$ is taken to be 0.22. Lien and Leschziner [4] simplified the treatment of the diffusion term to allow an appropriate isotropic diffusivity in the form

$$D_{ij} = \frac{\partial}{\partial x_k} \left[ \frac{H}{\sigma_k} \frac{\partial}{\partial x_k} (\overline{u_i u_j}) \right]$$

where $\sigma_k$ is a dimensionless constant. Harlow's proposal for the diffusion term is adopted in the present module since it is based on the fundamental conservation equations for the triple correlations, while Lien & Leschziner's form has a weaker basis in this respect.

$\Phi_{ij}$ represents the redistribution of turbulence energy among the normal stresses through the interaction of pressure and strain fluctuations. Modeling the pressure-strain term is the most elaborate and involves the solution of the Poisson equation for pressure fluctuations $p$. The explicit appearance of the pressure in the correlation is eliminated by taking the divergence of the equation for the fluctuating velocity $u_i$, thus obtaining a Poisson equation for $p$. Following a volume integration of the resulting equation subject to the assumption of local mean-flow homogeneity results in three contributions to the pressure-strain correlation $\Phi_{ij}$. One involving just fluctuating quantities $\Phi_{ij,1}$ another arising from the presence of the mean rate of strain $\Phi_{ij,2}$, and a third arising from the surface integral representing wall effects $\Phi_{ij,w}$. Since the primary role of $\Phi_{ij}$ is to guide turbulence towards isotropy, Rotta [5] proposed for $\Phi_{ij,1}$

$$\Phi_{ij,1} = -2 \rho C_i \varepsilon b_{ij}$$
where \( b_{ij} = (\bar{u}_i \bar{u}_j - \frac{2}{3} \delta_{ij} k) / 2k \) is the dimensionless anisotropy parameter. \( C_1 \) is a constant and \( k \) and \( \varepsilon \) are turbulent kinetic energy and energy dissipation respectively. More elaborate models have been proposed such as Lumley [6] and Fu [7] using a nonlinear expression for \( \Phi_{ij,1} \). The term \( \Phi_{ij,2} \) has been the subject of more extensive research. The traditional linear approach similar to Rotta's work simplifies this correlation to:

\[
\Phi_{ij,2} = -C_2 (P_{ij} - \frac{2}{3} \delta_{ij} P)
\]

where \( P \) is the production of turbulent kinetic energy. Analogous to \( \Phi_{ij,1} \) the correlation, \( \Phi_{ij,2} \) represents the isotropization of turbulence production tensor with \( C_2 \) as a constant. More elaborate models such as that of Speziale, Sarkar and Gatski [8] is based on dynamical systems approach and invariancy concepts. Nonlinear models for \( \Phi_{ij,2} \) based on the realizability constraints have been developed, e.g, Shih and Lumley [9] and Fu, Launder and Tselepidakis [10]. The simplified correlations in equations (8) and (9) are used in the present module.

The correlation \( \Phi_{ij,w} \) represents the wall damping effects that counteracts the tendency of \( \Phi_{ij,1} \) and \( \Phi_{ij,2} \) to isotropise the turbulent structure. Since close to a solid wall turbulence approach a state of intense anisotropy associated with a tendency towards a 2D turbulence. Following Shir [11] and Gibson and Launder [2], \( \Phi_{ij,w} \) is modeled as the combination of two separate terms:

\[
\Phi_{ij,1w} = C_{1w} \rho \frac{\varepsilon}{k} [\bar{u}_k \bar{u}_m n_k n_m \delta_{ij} - \frac{3}{2} \bar{u}_k \bar{u}_i n_k n_j - \frac{3}{2} \bar{u}_k \bar{u}_j n_k n_i] f(\frac{1}{l_n})
\]

\[
\Phi_{ij,2w} = C_{2w} [\Phi_{km,2} n_k n_m \delta_{ij} - \frac{3}{2} \Phi_{ij,2} n_k n_j - \frac{3}{2} \Phi_{jk,2} n_k n_i] f(\frac{1}{l_n})
\]

where \( l_n \) is the normal distance from the point in question to the wall and \( l (= \frac{k^{3/2}}{\varepsilon}) \) is the turbulent length scale. The following relationship is used for the wall damping function

\[
f = \frac{C_m^{7/4} k^{3/4}}{\kappa \varepsilon} \frac{1}{<l_n>}
\]

where \( <l_n> \) is the average distance of the point considered from the surrounding surfaces and \( n_i \) is a wall-normal unit vector in the \( i \)-direction. The constants \( C_{1w} \) and \( C_{2w} \) have values of 0.5 and 0.3 respectively.

It will be of some value to list the full Reynolds stress equations for axisymmetric swirling flows. Although, the derivations have been carried out within the constraints of Cartesian coordinates, considerations will be given next to the forms applicable to any general curved coordinate system.
In general the transport equation for the Reynolds stresses \((u_i u_j)\) can be written as:

\[
C_{ij} = D_{ij} + P_{ij} + F_{ij} - \varepsilon_{ij} + R_{ij}
\]

where \(C_{ij}, D_{ij}, P_{ij}, F_{ij}\) and \(\varepsilon\) represent convection, diffusion, production, pressure-strain and dissipation terms. The term \(R_{ij}\) results from the transformation of the equation from plane to axially symmetric conditions and swirl. In Cartesian coordinates, the above terms are summarized below for each stress component;

**\(u^2\) - equation**

\[
C_{11} = \frac{1}{r} \frac{\partial}{\partial x} (\rho r U u_1^2) + \frac{1}{r} \frac{\partial}{\partial \varphi} (\rho r V u_1^2)
\]

\[
D_{11} = \frac{1}{r} \frac{\partial}{\partial x} \left[ \rho r C_k u_1^2 \frac{k \partial u_1^2}{\varepsilon} + \rho r C_k u_1 v_1 \frac{k \partial u_1^2}{\varepsilon} \right] + \frac{1}{r} \frac{\partial}{\partial \varphi} \left[ \rho r C_k u_1 v_1 \frac{k \partial u_1^2}{\varepsilon} + \rho r C_k v_1^2 \frac{k \partial u_1^2}{\varepsilon} \right]
\]

\[
P_{11} = -2r (\frac{u_1^2}{\partial x} + u_1 v_1 \frac{\partial U}{\partial \varphi})
\]

\[
\Phi_{11} = -\rho C_1 \frac{\varepsilon}{k} (u_1^2 - \frac{2}{3} k) - C_2 (P_{11} - \frac{2}{3} P)
\]

\[
+ \rho C_{1\omega} \frac{\varepsilon}{k} [-2 u_1^2 f_x + \frac{v_1^2}{\partial x} - \frac{u_1 v_1 \partial f_{xy}}{\partial x}]
\]

\[
+ C_{2\omega} \left[ 2 C_2 (P_{11} - \frac{2}{3} P) f_x - C_2 (P_{22} - \frac{2}{3} P) f_y + C_2 P_{12} f_{xy} \right]
\]

\[
\varepsilon_{11} = -\frac{2}{3} \rho \varepsilon
\]

**\(v^2\) - equation**

\[
C_{22} = \frac{1}{r} \frac{\partial}{\partial x} (\rho r U v_2^2) + \frac{1}{r} \frac{\partial}{\partial \varphi} (\rho r V v_2^2)
\]
\[ D_{22} = \frac{1}{r} \frac{\partial}{\partial x} \left[ \rho r C_k \frac{k}{\varepsilon} \left( u \frac{\partial \varepsilon}{\partial x} + \frac{\partial \varepsilon}{\partial y} \right) \right] + \frac{1}{r} \frac{\partial}{\partial r} \left[ \rho r C_k \frac{k}{\varepsilon} \left( u \frac{\partial \varepsilon}{\partial x} + v^2 \frac{\partial \varepsilon}{\partial y} \right) \right] \]

\[ P_{22} = -2 \rho \left( u \frac{\partial V}{\partial x} + \frac{\partial V}{\partial r} - \frac{\partial W}{r} \right) \]

\[ \Phi_{22} = -C_1 \rho \frac{\varepsilon}{k} \left( \frac{\partial \varepsilon}{\partial x} \right) \cdot C_2 \left( P_{22} - \frac{2}{3} P \right) \]

\[ + \rho C_{1w} \frac{\varepsilon}{k} \left( u^2 f_x - 2 v^2 f_y - uv f_{xy} \right) \]

\[ + C_{2w} \left[ -C_2 \left( P_{11} - \frac{2}{3} P \right) f_x + 2 C_2 \left( P_{22} - \frac{2}{3} P \right) f_y + C_2 \frac{P_{12}}{P} f_{xy} \right] \]

\[ R_{22} = 2 \rho C_k \frac{k}{\varepsilon} \frac{w^2}{r^2} \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) - 2 \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) \cdot C_2 \left( P_{22} - \frac{2}{3} P \right) f_{xy} \]

\[ -2 \rho C_k \frac{k}{\varepsilon} \frac{w^2}{r^2} \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) - 2 \rho C_k \frac{k}{\varepsilon} \frac{w^2}{r} \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) + 2 \rho \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) \frac{W}{r} \]

\[ \text{--- equation} \]

\[ C_{33} = \frac{1}{r} \frac{\partial}{\partial x} \left( \rho r U w^2 \right) + \frac{1}{r} \frac{\partial}{\partial r} \left( \rho r V w^2 \right) \]

\[ D_{33} = \frac{1}{r} \frac{\partial}{\partial x} \left[ \rho r C_k \frac{k}{\varepsilon} \left( u^2 \frac{\partial w^2}{\partial x} + \frac{\partial w^2}{\partial y} \right) \right] \]

\[ + \frac{1}{r} \frac{\partial}{\partial r} \left[ \rho r C_k \frac{k}{\varepsilon} \left( u^2 \frac{\partial w^2}{\partial x} + \frac{\partial w^2}{\partial y} \right) \right] \]

\[ P_{33} = -2 \rho \left( u w \frac{\partial W}{\partial x} + v w \frac{\partial W}{\partial r} + w^2 \frac{V}{r} \right) \]

\[ F_{33} = -\rho C_1 \frac{\varepsilon}{k} \left( w^2 - \frac{2}{3} k \right) \cdot C_2 \left( P_{33} - \frac{2}{3} P \right) \]

\[ + \rho C_{1w} \frac{\varepsilon}{k} \left( u^2 f_x + \frac{\partial w^2}{\partial y} + 2 uv f_{xy} \right) \]

\[ - C_2 C_{2w} \left[ \left( P_{11} - \frac{2}{3} P \right) f_x + \left( P_{22} - \frac{2}{3} P \right) f_y - 2 C_2 \frac{P_{12}}{P} f_{xy} \right] \]

\[ R_{33} = 2 \rho C_k \frac{k}{\varepsilon} \frac{w^2}{r^2} \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) + 2 \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) \frac{w^2}{r} \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) + 2 \rho \frac{\partial}{\partial x} \left( \frac{\partial \varepsilon}{\partial x} \right) \frac{W}{r} \]
- 2 \rho \ C_k \ \frac{k \ (w^2)^2}{r^2} + 2 \rho \ C_k \ \frac{k \ \bar{u} \ \bar{w} \ \frac{\partial \bar{w}}{\partial \bar{r}}}{r} + 2 \rho \ C_k \ \frac{k \ \bar{w} \ \frac{\partial \bar{w}}{\partial \bar{r}}}{r} - 2 \rho \ \bar{w} \ \frac{W}{r}

\textit{\overline{uv} - equation}

\[ C_{12} = \frac{1}{r} \ \frac{\partial}{\partial \bar{x}} (\rho \bar{r} \bar{U} \bar{u} \bar{v}) + \frac{1}{r} \ \frac{\partial}{\partial \bar{r}} (\rho \bar{r} \bar{V} \bar{u} \bar{v}) \]

\[ D_{12} = \frac{1}{r} \ \frac{\partial}{\partial \bar{x}} \left[ \rho \ C_k \ \frac{\bar{u} \ \frac{\partial \bar{u}}{\partial \bar{x}} + \bar{w} \ \frac{\partial \bar{w}}{\partial \bar{x}}}{r} \right] + \frac{1}{r} \ \frac{\partial}{\partial \bar{r}} \left[ \rho \ C_k \ \frac{\bar{u} \ \frac{\partial \bar{u}}{\partial \bar{r}} + \bar{v} \ \frac{\partial \bar{v}}{\partial \bar{r}}}{r} \right] \]

\[ P_{12} = -\rho \left( \bar{u}^2 \ \frac{\partial \bar{V}}{\partial \bar{x}} - \frac{\bar{u} \ \bar{v}}{r} + \bar{v}^2 \ \frac{\partial \bar{U}}{\partial \bar{r}} - \bar{w} \ \frac{W}{r} \right) \]

\[ \Phi_{12} = -\rho \ C_i \ \frac{\bar{e}}{k} \overline{uv} - C_2 \ P_{12} \]

\[ = \frac{3}{2} \rho C_{lw} \ \frac{\bar{e}}{k} \left( \bar{u} \bar{v} (f_x + f_y) + (\bar{u}^2 + \bar{v}^2) f_{xy} \right) \]

\[ +\frac{3}{2} \ C_2 \ C_{2w} \left( (P_{11} + P_{22} - \frac{4}{3} P) f_{xy} + P_{12} (f_x + f_y) \right) \]

\[ R_{12} = -\rho \ C_k \ \frac{1}{\bar{e}} \ \frac{\bar{w} ^2 \ \bar{u} \bar{v}}{r^2} - \frac{\partial}{\partial \bar{x}} \left( \rho \ C_k \ \frac{k \ (\bar{u} \bar{w})^2}{r} \right) - \frac{1}{r} \ \frac{\partial}{\partial \bar{r}} \left( \rho \ C_k \ \frac{k \ \bar{u} \ \bar{w} \ \bar{v}}{r} \right) \]

\[ -\rho \ C_k \ \frac{1}{r} \ \frac{k \ \bar{w} \ \frac{\partial \bar{u}}{\partial \bar{x}} + \bar{u} \ \bar{w} \ \frac{\partial \bar{w}}{\partial \bar{r}}}{r} + \rho \ \bar{u} \ \bar{w} \ \frac{W}{r} \]

\textit{\overline{vw} - equation}

\[ C_{23} = \frac{1}{r} \ \frac{\partial}{\partial \bar{x}} (\rho \bar{r} \bar{U} \bar{v}) + \frac{1}{r} \ \frac{\partial}{\partial \bar{r}} (\rho \bar{r} \bar{V} \bar{v}) \]

\[ D_{23} = \frac{1}{r} \ \frac{\partial}{\partial \bar{x}} \left[ \rho \ C_k \ \frac{\bar{u} \ \frac{\partial \bar{v}}{\partial \bar{x}} + \bar{w} \ \frac{\partial \bar{w}}{\partial \bar{r}}}{r} \right] \]

\[ + \frac{1}{r} \ \frac{\partial}{\partial \bar{r}} \left[ \rho \ C_k \ \frac{\bar{u} \ \frac{\partial \bar{v}}{\partial \bar{x}} + \bar{v} \ \frac{\partial \bar{v}}{\partial \bar{r}}}{r} \right] \]

\[ P_{23} = -\rho \left( \bar{w} \ \frac{\partial \bar{W}}{\partial \bar{x}} + \bar{u} \ \bar{w} \ \frac{\partial \bar{V}}{\partial \bar{x}} + \bar{v} \ \bar{w} \ \frac{\partial \bar{V}}{\partial \bar{r}} + \bar{w} \ \frac{\bar{V} \ \bar{r} - \bar{w}^2 \ \frac{W}{r}}{r} \right) \]
\[
\Phi_{23} = -C_1 \frac{\varepsilon}{k} \overline{vw} - C_2 P_{23} - \frac{3}{2} \rho C_{1w} \frac{\varepsilon}{k} (\overline{uw} f_{xy} + \overline{vw} f_y)
\]
\[
+ \frac{3}{2} C_2 C_{2w} (P_{13fxy} + P_{23fy})
\]
\[
R_{23} = -\rho \left( \overline{v^2} - \overline{w^2} \right) \frac{W}{r} + \rho C_k \frac{k}{\varepsilon} \frac{1}{r} \overline{vw} \frac{\partial}{\partial r} (\overline{v^2} - \overline{w^2}) + \frac{\partial}{\partial x} (\rho C_k \frac{k}{\varepsilon} \overline{uw} \frac{(\overline{v^2} - \overline{w^2})}{r})
\]
\[
+ \frac{1}{r} \frac{\partial}{\partial r} (\rho C_k \frac{k}{\varepsilon} \overline{vw} (\overline{v^2} - \overline{w^2})) - 4 \rho C_k \frac{k}{\varepsilon} \overline{vw} \overline{w^2} \frac{r}{r} + \rho C_k \frac{k}{\varepsilon} \overline{uw} \frac{\partial}{\partial x} (\overline{v^2} - \overline{w^2})
\]

• \overline{uw}-equation

\[
C_{13} = \frac{1}{r} \frac{\partial}{\partial x} (\rho r U \overline{uw}) + \frac{1}{r} \frac{\partial}{\partial r} (\rho r V \overline{uw})
\]
\[
D_{13} = \frac{1}{r} \frac{\partial}{\partial x} \left[ \rho r C_k \frac{k}{\varepsilon} (u^2 \frac{\partial \overline{uw}}{\partial x} + uv \frac{\partial \overline{uw}}{\partial r}) \right]
\]
\[
+ \frac{1}{r} \frac{\partial}{\partial r} \left[ \rho r C_k \frac{k}{\varepsilon} (uv \frac{\partial \overline{uw}}{\partial x} + v^2 \frac{\partial \overline{uw}}{\partial r}) \right]
\]
\[
P_{13} = -\rho \left( \overline{u^2} \frac{\partial W}{\partial x} + uv \frac{\partial W}{\partial r} + \overline{vw} \frac{\partial U}{\partial x} - \overline{uw} \frac{\partial V}{\partial r} \right)
\]
\[
F_{13} = -\rho C_1 \frac{\varepsilon}{k} \overline{uw} - C_2 P_{13} - \frac{3}{2} \rho C_{1w} \frac{\varepsilon}{k} \overline{uw} f_x
\]
\[
- \frac{3}{2} \rho C_{1w} \frac{\varepsilon}{k} \overline{vw} f_{xy} + \frac{3}{2} C_2 C_{2w} C_2 P_{13fxy} + \frac{3}{2} C_2 C_{2w} P_{23fy}
\]
\[
R_{13} = -\rho \overline{uw} \frac{W}{r} + \rho C_k \frac{k}{\varepsilon} \frac{1}{r} \overline{vw} \frac{\partial \overline{uv}}{\partial r} + \frac{\partial}{\partial x} (\rho C_k \frac{k}{\varepsilon} \overline{uw} \frac{\overline{uv}}{r})
\]
\[
+ \frac{1}{r} \frac{\partial}{\partial r} (\rho C_k \frac{k}{\varepsilon} \overline{vwuv}) + \rho C_k \frac{k}{\varepsilon} \frac{\overline{uw}}{r} \frac{\partial \overline{uw}}{\partial x} - \rho C_k \frac{k}{\varepsilon} \frac{\overline{uw}}{r} \frac{\overline{uw}}{r^2}
\]

The turbulence energy dissipation rate \( \varepsilon \) is determined from its own transport equation;

\[
\frac{1}{r} \frac{\partial}{\partial x} (r C_{e\varepsilon} P_k \frac{k}{\varepsilon} \frac{\partial \varepsilon}{\partial x}) + C_{e_{\varepsilon} k} \frac{\varepsilon}{k} P_k - C_{e_{\varepsilon} 2} \rho \frac{\varepsilon^2}{k}
\]

(14)

where the constants \( C_{e_{\varepsilon} 1} \) and \( C_{e_{\varepsilon} 2} \) have values of 1.44 and 1.92 respectively.
The terms $f_x$, $f_y$, and $f_{xy}$ appearing in the stress-equation are tied to the orientation of the wall through the wall-damping function $f$ and will be explained later in the wall reflection treatment section.

5.3 Boundary Conditions

To solve the transport equations for the Reynolds stresses, boundary conditions for the stresses are needed. In the present module the log-law based relations are used to bridge the gap between the fully turbulent and viscous near-wall regions. Boundary values for the stresses can be derived by applying the Reynolds stress equations to the near-wall equilibrium flow. It can be shown that the stresses are related to the turbulent kinetic energy $\overline{u_i u_j} = C_{ij} k$, where $C_{ij}$ are constants to be determined. Consider as an example, the log-layer turbulent flow, where $S = \frac{\partial U}{\partial y} = \frac{u_\tau}{\kappa y}$, where $u_\tau$ is the friction velocity and $\kappa$ is Von Karman constant. In the log-layer, the limiting form of the stress equation is obtained by neglecting the convective terms and equating the production to dissipation and setting the wall-distance function $f=1$, hence the molecular and turbulent diffusion terms can be neglected. Consequently, the normal stress equation for the wall-normal component when simplified with $\Phi_{22} = \rho \frac{\nu^2}{k} \varepsilon$ is:

$$\frac{\nu^2}{k} = \frac{2 \left( -1 + C_1 + C_2 - 2C_2C_{2w} \right)}{3 \left( C_1 + 2C_{1w} \right)} = C_{22} \tag{15}$$

From experimental data, Lien & Leschziner [4] reported a value of $C_{22} \approx 0.249$ for near wall equilibrium turbulence. The most frequently used value of $C_1 = 1.8$ and $C_2 = 0.6$, and from Gibson and Launder [2] $C_{1w} = 0.5$ and $C_{2w} = 0.3$. Substituting these values into equation (13) give a value $C_{22} = 0.247$ which is close to the experimental value. Similarly, these constants also give $C_{11} = 1.09$, $C_{33} = 0.654$ and $C_{12} = -0.255$. 

-67-
5.4 Numerical Procedure

The conservation equations for the Reynolds stresses and the energy dissipation are integrated over control volumes after transformation of the Cartesian form to body-fitted no-orthogonal coordinates. The equation governing the transport of a scalar property $\Phi$, which stands for the Reynolds stress components and the energy dissipation equation can be written as:

$$\frac{1}{J} \frac{\partial}{\partial \xi^k} \left[ J (\rho U_m \Phi - q_m) \beta_m^k \right] = S^\Phi$$

(16)

where $\xi^k$ represents the curvilinear coordinate frame and $J$ is the Jacobian of the coordinate transformation, and $\beta_m^k$ represents its cofactors and $q_m$ represents the diffusion flux. Equation (16) is then integrated over discrete control volumes where the dependent variables on the volume faces are approximated by finite-difference representation.

In general the diffusion term is represented as

$$q_m = \Gamma_\Phi \frac{\partial \Phi}{\partial \xi^n} \beta_l^n$$

(17)

where $\Gamma_\Phi$ is the diffusion coefficient.

The tensorial form of the diffusivity due to Daly and Harlow [3] is adopted as;

$$\Gamma_\Phi = \rho r C_s^k \frac{u_m u_l}{\epsilon}$$

(18)

instead of the isotropic diffusivity ($\Gamma_\Phi = \mu_t / \sigma_\Phi$). Utilizing the equilibrium assumption and experimental near-wall stress data, the constant $C_s$ is taken to be 0.22 for the Reynolds stress equations and 0.18 for the turbulent energy dissipation equation. The diffusion term is discretized with a second-order central differencing scheme, while the convective terms are discretized using first or second order upwind differencing scheme.

A special discretization practice for the Reynolds stress gradients is introduced into the finite volume procedure with colocated storage arrangement. This is necessary to avoid the problem of mean velocity-Reynolds stress decoupling that can lead to oscillatory solutions or even divergence of the iterative solution algorithm. The procedure adopted in the present work differs from that of Obi & Peric [12] and that of Lien and Leschziner [4] by accounting for all the driving forces of the Reynolds stresses and not only those given by the gradient-diffusion type process. To illustrate the
origin of the problem, consider the Reynolds stress gradient terms in the axial momentum equation in 2D Cartesian uniform grid for simplicity

$$\frac{\partial \overline{u^2}}{\partial x} - \frac{\partial \overline{uv}}{\partial y}$$

Now integrating over a control volume surrounding node $P$ (cf. Figure 1) yields;

$$- \int \frac{\partial \overline{u^2}}{\partial x} \, dx \, dy - \int \frac{\partial \overline{uv}}{\partial y} \, dx \, dy = - \left\{ (\overline{u^2}_e - \overline{u^2}_w) \Delta y_p + (\overline{uv}_n - \overline{uv}_s) \Delta x_p \right\}$$

Now if the cell face values of the $\overline{u^2}_e, \overline{u^2}_w, \overline{uv}_n$ and $\overline{uv}_s$ are evaluated with linear interpolation, the stress difference expression become

$$- \left\{ \left( \frac{\overline{u^2}_e - \overline{u^2}_w}{2} \right) \Delta y_p + \left( \frac{\overline{uv}_n - \overline{uv}_s}{2} \right) \Delta x_p \right\}$$

and since no $P$-node shear stress appear in the resulting expression, a checker-board oscillation, similar to that played by the pressure field appear. Therefore, a non-linear interpolation scheme is needed to avoid these odd-even oscillations in the same context of Rhie and Chow [13] for cell face velocities. This means that any cell-face velocity is not merely sensitized to the pressure differences centered on that face but also to the Reynolds stress differences.

Consider the descretized equation for the axial normal stress component $\overline{u^2}$ in general non-orthogonal coordinates;

$$A_p \overline{u^2}_p = \sum_{i=0}^{n} A_i \overline{u^2}_i + S_{u^2}$$  \hspace{1cm} (19)

where $n$ stands for the cells E, W, N and S neighboring $P, A_i$ are the coefficients for the neighboring cells and $S_{u^2}$ is the source term that includes production, dissipation and pressure-strain redistribution terms as;

$$S_{u^2} = P_{11} - \frac{2}{3} \rho \varepsilon + \Phi_{II}$$

where $\Phi_{II}$ combines Rotta’s stress isotropization model and isotropization of production model and related wall-correction terms due to Gibson and Launder [2].

$$\Phi_{II} = - \rho C_1 \frac{\varepsilon}{k} (\overline{u^2} - \frac{2}{3} k) - C_2 (P_{11} - \frac{2}{3} P)$$

$$+ \rho C_{1w} \frac{\varepsilon}{k} \left( - \overline{2u^2 f_x} + \overline{v^2 f_y} - \overline{uv f_{xy}} \right)$$

$$+ 2 C_2 C_{2w} (P_{11} - \frac{2}{3} P) f_x - C_2 C_{2w} (P_{22} - \frac{2}{3} P) f_y + C_2 C_{2w} P_{12} f_{xy}$$  \hspace{1cm} (20)

Rearranging the production terms that contribute to the stress generation and noting that
\[ P = \frac{1}{2} P_{kk}, \text{ then;} \]
\[ S_{u^2} = AP_{11} + BP_{22} + CP_{33} + DP_{12} + S_{11} \]  

(21)

where
\[ A = 1 - \frac{2}{3} C_2 + \frac{1}{3} C_2 C_{2w}f_y + \frac{4}{3} C_2 C_{2w}f_x \]
\[ B = 1 - \frac{2}{3} C_2 + \frac{1}{3} C_2 C_{2w}f_y + \frac{4}{3} C_2 C_{2w}f_x \]
\[ C = \frac{1}{3} C_2 + \frac{1}{3} C_2 C_{2w} (f_y - 2f_x) \]
\[ D = C_2 C_{2w}f_{xy} \]

and \( S_{11} \) contains the remaining terms.

Substituting for the production terms \( P_{11}, P_{22}, P_{33} \) and \( P_{12} \), then equation (19) becomes;

\[
\overline{u^2}_p = H_P + 2\rho A \left[ \overline{w^2} (D_1 \Delta U \xi + D_2 \Delta U \eta) + \overline{wv} (E_1 \Delta U \eta + E_2 \Delta U \xi) \right]_P \\
+ 2\rho B \left[ \overline{wv} (D_1 \Delta V \xi + D_2 \Delta V \eta) + \overline{v^2} (E_1 \Delta V \xi + E_2 \Delta V \xi) \right]_P + \frac{\overline{vw} W}{A_P} \frac{1}{r} \]

\[
+ 2\rho C \left[ \overline{wv} (D_1 \Delta W \xi + D_2 \Delta W \eta) + \overline{wv} (E_1 \Delta W \xi + E_2 \Delta W \xi) \right]_P - \frac{\overline{w^2} W}{A_P} \frac{1}{r} \]

\[
+ \rho D \left[ \overline{u^2} (D_1 \Delta V \xi + D_2 \Delta V \eta) + \overline{v^2} (E_1 \Delta U \eta + E_2 \Delta U \xi) \right]_P + \frac{\overline{uv} V}{A_P} \frac{1}{r} + \frac{\overline{uw} W}{A_P} \frac{1}{r} \]

\[ + \frac{S_{11}}{A_P} \]  

(22)

where
\[ H_P = \sum_{i}\overline{u^2}_i / A_P \]
\[ D_1 = - \Delta y_{\eta}^\eta / A_P, \quad D_2 = \Delta y_{\eta}^\xi / A_P, \]
\[ E_1 = - \Delta x_{\eta}^\xi / A_P \quad \text{and} \quad E_2 = \Delta x_{\eta}^\eta / A_P \]

here \( \Delta y_{\eta}^\eta = (y_n - y_s), \Delta y_{\eta}^\xi = (y_e - y_w), \) etc

and \( \Delta U \xi = (U_E - U_P), \Delta V \xi = (V_E - V_P), \) etc
Now, performing the interpolation practice to obtain east cell-face value of the normal stress \( \overline{u^2_e} \) we obtain;

\[
\overline{u^2_e} = < \overline{u^2_p} > - 2 \rho A \overline{u^2 D_1} \Delta U \xi > - 2 \rho A \overline{u v E_2} \Delta U \xi > \\
- 2 \rho B \overline{u v D_1} \Delta V \xi > - 2 \rho B \overline{v^2 E_2} \Delta V \xi > \\
- 2 \rho C \overline{u w D_1} \Delta W \xi > - 2 \rho C \overline{v w E_2} \Delta W \xi > \\
- \rho D \overline{u^2 D_1} \Delta V \xi > - \rho D \overline{v^2 E_2} \Delta U \xi > \\
+ 2 \rho A \overline{u^2 D_1} > \Delta U \xi > + 2 \rho A \overline{u v E_2} > \Delta U \xi > \\
+ 2 \rho B \overline{u v D_1} > \Delta V \xi > + 2 \rho B \overline{v^2 E_2} > \Delta V \xi > \\
+ 2 \rho C \overline{u w D_1} > \Delta W \xi > + 2 \rho C \overline{v w E_2} > \Delta W \xi > \\
+ 2 \rho D \overline{u^2 D_1} > \Delta V \xi > + \rho D \overline{v^2 E_2} > \Delta U \xi >
\]

(23)

The brackets < and > denote linear interpolation. For instance, on the east face

\[
< \Phi > = (1-f_{\xi}) \Phi_P + f_{\xi} \Phi_E \quad \text{where,} \quad f_{\xi} = \frac{\Delta x_P}{\Delta x_P + \Delta x_E}
\]

Similar expressions can be obtained for \( \overline{u^2_w}, \overline{u^2_n}, \overline{u^2_s}, \overline{u v_w}, \overline{u v_n}, \overline{u v_s}, \overline{v^2_e}, \overline{v^2_w}, \overline{v^2_n} \) and \( \overline{v^2_s} \) which are then used to calculate the Reynolds stress gradients in the discretized axial momentum equation. Similarly, expressions for \( \overline{u v_e}, \overline{u v_w}, \overline{u v_n}, \overline{u v_s}, \overline{v^2_e}, \overline{v^2_w}, \overline{v^2_n} \) and \( \overline{v^2_s} \) can be obtained for the stress gradients in the radial momentum equation and \( \overline{u w_e}, \overline{u w_w}, \overline{u w_n}, \overline{u w_s}, \overline{v w_e}, \overline{v w_w}, \overline{v w_n} \) and \( \overline{v w_s} \) expressions to evaluate stress gradients in the azimuthal momentum equation.

5.4.1 Wall Reflection Treatment

The wall reflection terms \( \Phi_{ij,w} \) appear in the pressure-strain term correlation as wall correction terms \( \Phi_{ij,1w} \) and \( \Phi_{ij,2w} \) to counteract the tendency of \( \Phi_{ij,1} \) and \( \Phi_{ij,2} \) to isotropise the turbulence structure. Special consideration is given to the wall proximity effects on the redistribution process \( \Phi_{ij,w} \) with relation to the local orthogonal coordinate system at the wall, cf. figure 2.

At a wall, turbulence approach a state of strong anisotropy associated with the tendency towards a 2D turbulence. The wall-reflection terms ensure that normal stress normal to the wall is not too
For body-fitted coordinates, there is a need to consider the tensorial form of the wall reflection terms since they are tied to the orientation of the wall through the damping function term (eq. 12). For a curved surface (figure 2), the wall normal vector \( \mathbf{n} = n_1 \mathbf{i}_1 + n_2 \mathbf{i}_2 \), where \( \mathbf{i}_1 \) and \( \mathbf{i}_2 \) are unit vectors in Cartesian coordinates. The Cartesian components of the wall-distance function \( f \) are given as:

\[
f_x = n_1^2 f, \quad f_y = n_2^2 f \quad \text{and} \quad f_{xy} = n_1 n_2 f
\]

where \( f_x = n_1^2 \left( C_{\mu}^{0.75} k^{1.5}/\kappa \varepsilon \right) / L_n \), for example and \( L_n \) is the normal distance from the wall.

The Reynolds stresses close to the wall are transformed from wall coordinates to Cartesian coordinates by appropriate vector decompositioning to give:

\[
\begin{align*}
\bar{u}^2 &= \bar{u}_1^2 t_1^2 + \bar{v}_2^2 n_1^2 + 2 \bar{uv} t_1 n_1 \\
\bar{v}^2 &= \bar{u}_2^2 t_2^2 + \bar{v}_2^2 n_2^2 + 2 \bar{uv} t_2 n_2 \\
\bar{w}^2 &= \bar{w}^2 \\
\bar{uv} &= \bar{u}_1 t_1 t_2 + \bar{v}_2 n_1 n_2 + \bar{uv} ( t_1 n_2 + t_2 n_1 ) \\
\bar{vw} &= \bar{uw} t_2 + \bar{vw} n_2 \\
\bar{uw} &= \bar{uw} t_1 + \bar{vw} n_1
\end{align*}
\]

where \( \bar{u}^2, \bar{v}^2 \) ... are the Reynolds stresses in Cartesian coordinates and \( \bar{u}_1^2, \bar{v}_2^2 \) ... are the Reynolds stresses in wall-coordinate, \( n_1, n_2 \) are the Cartesian components of the normal vector component and \( t_1, t_2 \) are the Cartesian components of the tangential vector component.

5.5 Module Evaluation

The RSM module was tested at Rocketdyne after interfacing with the CFD solver REACT and at the University of Alabama at Huntsville (UAH) using own solver (MAST). The first test was on fully developed channel flow with length to height ratio of 50 and a Reynolds number of \( 2 \times 10^5 \) based on the channel height. A non-uniform mesh of 101x41 was used with clustering at the walls. Figure 3 shows the fully developed mean velocity profile across the channel. Figure 4 shows the normal Reynolds stress profiles across the channel and figure 5 shows the shear stress profile. Similar results were obtained when the module was interfaced and tested independently at UAH using the MAST code.
The next test problem is that of the backward facing step of Driver and Seegmiller [14]. The calculations were performed using a 101x41 grid points with clustering near the walls. The computational domain had a length of 50H (H is the step height) and a width of 9H. The experimental data were used to specify the inflow conditions for a channel flow calculation where the fully developed profiles at the channel exit were used as the inlet conditions for the backward facing step calculations. Fully developed flow conditions were imposed at the outflow boundary. The boundary conditions for the Reynolds stress equations were arrived at by using the log-law of the wall and assuming local equilibrium conditions close to the wall. It can be shown that the Reynolds stresses are related to the turbulent kinetic energy by;

\[
\overline{u_iu_j} = C_{ij} k
\]

were \( C_{ij} \) is constant. The Reynolds stresses at the vicinity of the wall used are

\[
\overline{u^2} = 1.098 k, \quad \overline{\nu^2} = 0.247 k, \quad \overline{\nu \nu} = -0.255 k \quad \text{and} \quad \overline{w^2} = 2k - \overline{u^2} - \overline{\nu^2} = 0.654 k.
\]

Figure 6 shows the stream lines using Launder, Reece and Rodi's model. The computed reattachment length was about 5.8H which is closer to the experimental value of 6.1H than the standard \( k-\epsilon \) model (5.35H). The figure also shows a small (turbulence driven) secondary flow region at the base corner of the step which cannot be predicted using the isotropic eddy-viscosity \( k-\epsilon \) model. Also, a smaller recirculation region is noted at the top lip of the step which is also driven by turbulence anisotropy (more refined grid may be needed to isolate and study this region).

Figure 7 shows the mean velocity profile across the channel at four step heights downstream of the step as compared with the standard \( k-\epsilon \) turbulence model predictions. The axial normal turbulent intensity \( \overline{u^2} \) profile across the channel at \( x/H=4 \) is shown on figure 8 and the radial normal turbulent intensity \( \overline{\nu^2} \) is shown on figure 9. The shear stress \( \overline{\nu \nu} \) profile across the channel at \( x/H=4 \) is also shown on figure 10. The results show that the module predicts improved results using the RSM model as compared with the standard \( k-\epsilon \) model.
REFERENCES


Figure 1. Control volume for node $P$ and its surroundings

Figure 2. Cartesian and wall-coordinate systems
Figure 3. Mean velocity profile across the channel

Figure 4. Turbulent intensity profiles across the channel

Figure 5. Shear stress profile across the channel
Figure 6. Stream-function contours for the backward facing step flow

Figure 7. Axial mean velocity profile at $X/H = 4$
Figure 8. Turbulent intensity $\overline{uu}$ profile (normalized with $U^2_{ref}$) at $X/H = 4$
Figure 9. Turbulent stress $\overline{vv}$ profile (normalized with $U_{ref}^2$) at $X/H = 4$
Figure 10. Turbulent shear stress $\overline{uv}$ profile (normalized with $U_{ref}^2$) at $X/H = 4$
APPENDIX D

2D/Axisymmetric Reynolds Stress Module Deck

The 2D/axisymmetric Reynolds stress module is a FORTRAN source code to solve 2D/Axisymmetric turbulent flow using the full Reynolds stress model based on Launder, Reece and Rodi[1] when interfaced with a main flow solver. The module consists of the main routine RSMOD that calls a number of subroutines to perform different functions that will be explained below.

3.1 Subroutine RSMOD

This is basically the main routine that reads through its argument list different variables from the calling flow solver which are described below.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Grid node locations in the x or ξ-direction, dimensioned to X(NX*NY)</td>
</tr>
<tr>
<td>Y</td>
<td>Grid node locations in the y or η-direction, dimensioned to Y(NX*NY)</td>
</tr>
<tr>
<td>FX</td>
<td>Interpolation factor in the x or ξ-direction.</td>
</tr>
<tr>
<td>FY</td>
<td>Interpolation factor in the y or η-direction.</td>
</tr>
<tr>
<td>ARE</td>
<td>Control cell areas</td>
</tr>
<tr>
<td>VOL</td>
<td>Control cell volumes.</td>
</tr>
<tr>
<td>R</td>
<td>Radial distance in the axisymmetric geometry or 1. for planar geometry.</td>
</tr>
<tr>
<td>DNS</td>
<td>Normal distance of a cell from the south-boundary dimensioned to NX.</td>
</tr>
<tr>
<td>DNN</td>
<td>Normal distance of a cell from the north-boundary dimensioned to NX.</td>
</tr>
<tr>
<td>DNE</td>
<td>Normal distance of a cell from the east-boundary dimensioned to NY.</td>
</tr>
<tr>
<td>DNW</td>
<td>Normal distance of a cell from the west-boundary dimensioned to NY.</td>
</tr>
<tr>
<td>U</td>
<td>Axial or ξ-direction velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>V</td>
<td>Radial or ψ-direction velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>W</td>
<td>Tangential or azimuthal velocity, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>TE</td>
<td>Turbulent kinetic energy, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>ED</td>
<td>Turbulent energy dissipation rate, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>DEN</td>
<td>Density (assumed constant for incompressible flows).</td>
</tr>
<tr>
<td>F1</td>
<td>Mass flux at cell faces in the x or ξ-direction, dimensioned to NX*NY.</td>
</tr>
<tr>
<td>F2</td>
<td>Mass flux at cell faces in the y or η-direction, dimensioned to NX*NY.</td>
</tr>
</tbody>
</table>
Laminar viscosity.

Eddy viscosity, dimensioned to NX*NY.

Residual error for the equations solver, dimensioned to 8.

Boundary condition flag along the south boundary dimensioned to NX and must have one for each boundary node set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall e.g., for a wall boundary condition along the south boundary set ITBS to NX*4. Similarly for the other boundaries.

Boundary condition flag along the north boundary, dimensioned to NX.

Boundary condition flag along the east-boundary dimensioned to NY.

Boundary condition flag along the west-boundary dimensioned to NY.

Iteration number.

= 1 for swirl velocity calculations, 0 otherwise.

= 1 for axisymmetric flow, 0 otherwise.

= 1 if calculations are restarted from a previous run, 0 otherwise.

RSMOD starts by reading the turbulent flow constants, under-relaxation factors and Prandtl/Schmidt numbers for the $k$ and $\varepsilon$ equations. These are;

- **CD1, CD2**
  
  Constants in the $k$ and $\varepsilon$-equations and are usually set to 1.44 and 1.92 respectively.

- **CMU, ELOG**
  
  Also constants in the $k$ and $\varepsilon$-equations and are usually set to 0.09, 9.8 and 0.42 respectively.

- **C1, C2**
  
  Constants in the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations respectively.

- **GKE**
  
  Is set to 1 for second-order upwinding of the convective terms in the transport equations.

- **ALFAKE**
  
  Is the iteration parameter used in the $k$ and $\varepsilon$-equation solver.

- **URFVIS**
  
  Is the underrelaxation factor of the viscosity near the wall.

- **SORKE(1-8)**
  
  Are the degree of accuracy for the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations solver respectively.

- **URFKE(1-8)**
  
  Are the underrelaxation factors for the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations respectively.

- **PRTKE(1-8)**
  
  Are ratio of Prandtl to Schmidt numbers used in the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations respectively.

- **RSMOD starts by reading the turbulent flow constants, under-relaxation factors and Prandtl/Schmidt numbers for the $k$ and $\varepsilon$ equations. These are;**

- **CD1, CD2**
  
  Constants in the $k$ and $\varepsilon$-equations and are usually set to 1.44 and 1.92 respectively.

- **CMU, ELOG**
  
  Also constants in the $k$ and $\varepsilon$-equations and are usually set to 0.09, 9.8 and 0.42 respectively.

- **GKE**
  
  Is set to 1 for second-order upwinding of the convective terms in the transport equations.

- **ALFAKE**
  
  Is the iteration parameter used in the $k$ and $\varepsilon$-equation solver.

- **URFVIS**
  
  Is the underrelaxation factor of the viscosity near the wall.

- **SORKE(1-8)**
  
  Are the degree of accuracy for the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations solver respectively.

- **URFKE(1-8)**
  
  Are the underrelaxation factors for the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations respectively.

- **PRTKE(1-8)**
  
  Are ratio of Prandtl to Schmidt numbers used in the $k$, $\varepsilon$, $u^2$, $v^2$, $w^2$, $uv$, $vw$, and $uw$-equations respectively.

- **C1, C2**
  
  Are constants in the RSM model.
$C_1p$ and $C_2p$ are the two constants in the wall-reflection terms of the pressure-strain redistribution term.

$C_k$ and $C_\varepsilon$ are the constants in the diffusion term of the $k$ and $\varepsilon$-equations.

$CUU, CVV, CWW, CUV, CVW, CUW$ are the constants multiplying the kinetic energy for the stress values near the wall.

$WREFON$ = 1 if the wall reflection terms of the pressure-strain term are to be included, = 0 otherwise.

All variable dimensions considered are one-dimensional. The position of any node is defined as $IJ = (I,J) = (I-1) \cdot NJ + J$, where $NI$ and $NJ$ are the number of grid nodes in the X and Y-directions respectively. It is assumed that grid related data such as cell areas, volumes and interpolation factors be passed to the module from an external grid generator.

**Subroutine CALPIJ**

This subroutine calculates the production terms of the individual stress components.

**Subroutine CALUIUJ**

This subroutine solves the transport equations for the turbulent energy ($IPHI=1$), energy dissipation ($IPHI=2$) and Reynolds stresses ($IPHI=3, 4, 5, 6, 7, 8$ for $u^2, v^2, w^2, uv, vw, uw$). Daly and Harlow [3] gradient stress diffusion form is used in the module instead of the simplified isotropic diffusivity form. This subroutine calls MODUIUJ subroutine that sets the appropriate boundary conditions for the Reynolds stresses. The set of algebraic difference equations are then solved using Stone's strongly implicit solver SOLSIP.

**Subroutine MODPIJ**

This subroutine modifies the production terms near the wall using the near wall region model.

**Subroutine MODUIUJ**

This subroutine calculates the near wall boundary conditions for all the variables.
Subroutine SOLSIP

This subroutine solves the system of linear algebraic equations for all the variables using Stone's Implicit Procedure.

Subroutine WALREF

This subroutine calculates the wall reflection terms in the pressure-strain redistribution correlation. It calculates the wall unit normal vectors and the normal distance away from the wall. This is needed to resolve the wall tangential and normal velocity components that are needed to obtain the near-wall values of the Reynolds stresses.

SUBROUTINE WALPARA

This subroutine calculates the normal and tangential wall unit vectors.
ENDIF
143 IF (ITBN(1).EQ.4) THEN
144 J = 16NJ(1)+NJ
145 DXY=X(IJ)-X(IJ-NJ)
146 DYZ=Y(IJ)-Y(IJ-NJ)
147 FXY=SQR(DXY**2+DYZ**2)
148 FT1M(I)=DFB*FPHI
149 FT2M(I)=DXY*FPHI
150 FT1N(I)=DYZ*FPHI
151 FT2N(I)=DYZ*FPHI
152 ENDIF
153 CONTINUE
154
155 DO 20 J = 2, NJM
156 IF (ITBN(J).EQ.4) THEN
157 J = J + 1
158 DXY=X(IJ)-X(IJ-1)
159 DYZ=Y(IJ)-Y(IJ-1)
160 FXY=SQR(DXY**2+DYZ**2)
161 FT1M(J)=DFB*FPHI
162 FT2M(J)=DXY*FPHI
163 FT1N(J)=DYZ*FPHI
164 FT2N(J)=DYZ*FPHI
165 ENDIF
166 CONTINUE
167
168 IF (ITBN(J).EQ.4) THEN
169 J = 16NJ(NJM)+NJ
170 DXY=X(IJ)-X(IJ-1)
171 DYZ=Y(IJ)-Y(IJ-1)
172 FXY=SQR(DXY**2+DYZ**2)
173 FT1M(J)=DFB*FPHI
174 FT2M(J)=DXY*FPHI
175 FT1N(J)=DYZ*FPHI
176 FT2N(J)=DYZ*FPHI
177 ENDIF
178 CONTINUE
179
180 CMU25=SQR(SQR(CMU))
181 CMU25=CMU25**3
182 FCMC=CMU25/CAPPA
183
184 DO 80 I = 2, NJM
185 IF (ITBN(I).EQ.4) THEN
186 J = J + 1
187 IJ = 16NJ(I)+J
188 RDIPM=0.0
189 RDIA2=0.0
190 RDINN=0.0
191 RDINS=0.0
192 TE(IJ)=ABS(TI(IJ))
193 TE(IJ)=FCMC*TE(IJ)**1.5/(ED(IJ)+SMALL)
194 START WITH SOUTH BOUNDARY
195 IF (ITBN(I).EQ.4) THEN
196 J = 16NJ(I)+1
197 IJ = IJ+1
198 DXY=X(IJ)-X(IJ-1)
199 DYZ=Y(IJ)-Y(IJ-1)
200 XH=HAF*(X(IJ)+X(IJ-1))
201 YH=HAF*(Y(IJ)+Y(IJ-1))
202 XBP=QTR*(X(IJ)+X(IJ-1))
203 YBP=QTR*(Y(IJ)+Y(IJ-1))
204 DBP=DFB-DBP
205 DYP=DFB-YB
206 RDSN=1.0/(RDSN+SMALL)
207 RDSN=1.0/(RDSN+SMALL)
208 ENDIF
209 CHECK NORTH BOUNDARY
210 IF (ITBN(I).EQ.4) THEN
211 J = 16NJ(I)+1
212 IJ = IJ-1
213 DXY=X(IJ)-X(IJ-1)
214 DXY=X(IJ-1)-X(IJ-1)
215 XH=HAF*(X(IJ-1)+X(IJ-1))
216 YH=HAF*(Y(IJ-1)+Y(IJ-1))
217 XBP=QTR*(X(IJ-1)+X(IJ-1))
218 YBP=QTR*(Y(IJ-1)+Y(IJ-1))
219 DBP=DFB-DBP
220 DYP=DFB-YB
221 RDSN=1.0/(RDSN+SMALL)
222 ENDIF
223
224 DO 240 J = 2, NJM
225 IF (ITBN(J).EQ.4) THEN
226 J = J + 1
227 DXY=X(IJ)-X(IJ-1)
228 DYZ=Y(IJ)-Y(IJ-1)
229 XBP=QTR*(X(IJ)+X(IJ-1))
230 YBP=QTR*(Y(IJ)+Y(IJ-1))
231 DXY=DFB-DBP
232 DYP=DFB-YB
233 RDSN=1.0/(RDSN+SMALL)
234 ENDIF
235
236 DO 300 I = 2, NJM
237 IF (ITBN(I).EQ.4) THEN
238 J = J + 1
239 DXY=X(IJ)-X(IJ-1)
240 DYZ=Y(IJ)-Y(IJ-1)
241 XBP=QTR*(X(IJ)+X(IJ-1))
242 YBP=QTR*(Y(IJ)+Y(IJ-1))
243 DXY=DFB-DBP
244 DYP=DFB-YB
245 RDSN=1.0/(RDSN+SMALL)
246 ENDIF
247
248 DO 500 J = 2, NJM
249 IF (ITBN(J).EQ.4) THEN
250 J = J + 1
251 DXY=X(IJ)-X(IJ-1)
252 DYZ=Y(IJ)-Y(IJ-1)
253 XBP=QTR*(X(IJ)+X(IJ-1))
254 YBP=QTR*(Y(IJ)+Y(IJ-1))
255 DXY=DFB-DBP
256 DYP=DFB-YB
257 RDSN=1.0/(RDSN+SMALL)
258 ENDIF
259
260 DO 700 I = 2, NJM
261 IF (ITBN(I).EQ.4) THEN
262 J = J + 1
263 DXY=X(IJ)-X(IJ-1)
264 DYZ=Y(IJ)-Y(IJ-1)
265 XBP=QTR*(X(IJ)+X(IJ-1))
266 YBP=QTR*(Y(IJ)+Y(IJ-1))
267 DXY=DFB-DBP
268 DYP=DFB-YB
269 RDSN=1.0/(RDSN+SMALL)
270 ENDIF
271
272 DO 900 J = 2, NJM
273 IF (ITBN(J).EQ.4) THEN
274 J = J + 1
275 DXY=X(IJ)-X(IJ-1)
276 DYZ=Y(IJ)-Y(IJ-1)
277 XBP=QTR*(X(IJ)+X(IJ-1))
278 YBP=QTR*(Y(IJ)+Y(IJ-1))
279 DXY=DFB-DBP
280 DYP=DFB-YB
281 RDSN=1.0/(RDSN+SMALL)
282 ENDIF
283
284 DO 1000 I = 2, NJM
285 IF (ITBN(I).EQ.4) THEN
286 J = J + 1
287 DXY=X(IJ)-X(IJ-1)
288 DYZ=Y(IJ)-Y(IJ-1)
289 XBP=QTR*(X(IJ)+X(IJ-1))
290 YBP=QTR*(Y(IJ)+Y(IJ-1))
291 DXY=DFB-DBP
292 DYP=DFB-YB
293 RDSN=1.0/(RDSN+SMALL)
294 ENDIF
295
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>285</td>
<td>U3W(J) = U1W(J)</td>
</tr>
<tr>
<td>286</td>
<td>U2W(J) = V1W(J)</td>
</tr>
<tr>
<td>287</td>
<td>U3W(J) = W1W(J)</td>
</tr>
<tr>
<td>288</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>289</td>
<td>C</td>
</tr>
<tr>
<td>290</td>
<td>DO 101 I = 2, NIM</td>
</tr>
<tr>
<td>291</td>
<td>J = 1</td>
</tr>
<tr>
<td>292</td>
<td>IJ = IMN(I) + J</td>
</tr>
<tr>
<td>293</td>
<td>UN = U(IJ)</td>
</tr>
<tr>
<td>294</td>
<td>VN = V(IJ)</td>
</tr>
<tr>
<td>295</td>
<td>WN = W(IJ)</td>
</tr>
<tr>
<td>296</td>
<td>DO 102 J = 2, NJ</td>
</tr>
<tr>
<td>297</td>
<td>IJ = IMN(I) + J</td>
</tr>
<tr>
<td>298</td>
<td>IFJ = IJ + NJ</td>
</tr>
<tr>
<td>299</td>
<td>IP = IJ - NJ</td>
</tr>
<tr>
<td>300</td>
<td>IJ = IJ + 1</td>
</tr>
<tr>
<td>301</td>
<td>IJ = IJ - 1</td>
</tr>
<tr>
<td>302</td>
<td>FXE = FXE</td>
</tr>
<tr>
<td>303</td>
<td>FXX = FXX</td>
</tr>
<tr>
<td>304</td>
<td>FYX = FYX</td>
</tr>
<tr>
<td>305</td>
<td>FYZ = FYZ</td>
</tr>
<tr>
<td>306</td>
<td>RX = QTR*(R(IJ) + R((IJ-NJ)+(IJ-1) + R((IJ-NJ)) - R((IJ-NJ)) + R((IJ-NJ)))</td>
</tr>
<tr>
<td>307</td>
<td>DXR = HAF*(XX(IJ) - XX(IJ-M) + XX(IJ-M) - XX(IJ-M))</td>
</tr>
<tr>
<td>308</td>
<td>DXR = HAF*(YY(IJ) - YY(IJ-M) + YY(IJ-M) - YY(IJ-M))</td>
</tr>
<tr>
<td>309</td>
<td>DYN = HAF*(YY(IJ) - YY(IJ-M) + YY(IJ-M) - YY(IJ-M))</td>
</tr>
<tr>
<td>310</td>
<td>CS = UN</td>
</tr>
<tr>
<td>311</td>
<td>VS = VN</td>
</tr>
<tr>
<td>312</td>
<td>WS = WN</td>
</tr>
<tr>
<td>313</td>
<td>C</td>
</tr>
<tr>
<td>314</td>
<td>UN = SIU1(IJ) + FYX*(SIU1(IJ) - FYX (IJJ) )</td>
</tr>
<tr>
<td>315</td>
<td>VN = SIV(IJ) + FYX*(SIV(IJ) - FYX (IJJ) )</td>
</tr>
<tr>
<td>316</td>
<td>WN = SIV(IJ) + FYX*(SIV(IJ) - FYX (IJJ) )</td>
</tr>
<tr>
<td>317</td>
<td>C</td>
</tr>
<tr>
<td>318</td>
<td>UE = U1W(J) + FXE</td>
</tr>
<tr>
<td>319</td>
<td>VE = V1W(J) + FXE</td>
</tr>
<tr>
<td>320</td>
<td>WE = W1W(J) + FXE</td>
</tr>
<tr>
<td>321</td>
<td>C</td>
</tr>
<tr>
<td>322</td>
<td>DUNW = U1W(J)</td>
</tr>
<tr>
<td>323</td>
<td>DUNV = W1W(J)</td>
</tr>
<tr>
<td>324</td>
<td>DUNW = U1W(J)</td>
</tr>
<tr>
<td>325</td>
<td>DUNV = W1W(J)</td>
</tr>
<tr>
<td>326</td>
<td>DUNW = U1W(J)</td>
</tr>
<tr>
<td>327</td>
<td>DUNV = W1W(J)</td>
</tr>
<tr>
<td>328</td>
<td>C</td>
</tr>
<tr>
<td>329</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>330</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>331</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>332</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>333</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>334</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>335</td>
<td>DUX = DUX*(DUX<em>DUX-DYX</em>DYX)/ARE(IJ)</td>
</tr>
<tr>
<td>336</td>
<td>C</td>
</tr>
<tr>
<td>337</td>
<td>P111(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>338</td>
<td>P121(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>339</td>
<td>&amp;</td>
</tr>
<tr>
<td>340</td>
<td>VJ = W1W(J)</td>
</tr>
<tr>
<td>341</td>
<td>VJR = 0.0</td>
</tr>
<tr>
<td>342</td>
<td>IF(ASK10) VJR = V1V(IJ)</td>
</tr>
<tr>
<td>343</td>
<td>P311(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>344</td>
<td>&amp;</td>
</tr>
<tr>
<td>345</td>
<td>P121(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>346</td>
<td>&amp;</td>
</tr>
<tr>
<td>347</td>
<td>P121(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>348</td>
<td>&amp;</td>
</tr>
<tr>
<td>349</td>
<td>P121(J) = -2.0*(D101(IJ) + U12(IJ) + DUXVU(IJ) + DUDVU(IJ))</td>
</tr>
<tr>
<td>350</td>
<td>&amp;</td>
</tr>
<tr>
<td>351</td>
<td>V1W(J) = V1W(J)</td>
</tr>
<tr>
<td>352</td>
<td>V2W(J) = V2W(J)</td>
</tr>
<tr>
<td>353</td>
<td>GEN1(IJ) = 0.5*(P1111(J) + P2211(J) + P3311(J))</td>
</tr>
<tr>
<td>354</td>
<td>C</td>
</tr>
<tr>
<td>355</td>
<td>U1W(J) = U1W(J)</td>
</tr>
<tr>
<td>356</td>
<td>U3W(J) = U3W(J)</td>
</tr>
<tr>
<td>357</td>
<td>C</td>
</tr>
<tr>
<td>358</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>359</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>360</td>
<td>C</td>
</tr>
<tr>
<td>361</td>
<td>MODIFY GEN-TERMS CLOSE TO A WALL</td>
</tr>
<tr>
<td>362</td>
<td>CALL MODPI</td>
</tr>
<tr>
<td>363</td>
<td>RETURN</td>
</tr>
<tr>
<td>364</td>
<td>END</td>
</tr>
<tr>
<td>365</td>
<td>C</td>
</tr>
<tr>
<td>366</td>
<td>C</td>
</tr>
<tr>
<td>367</td>
<td>C</td>
</tr>
<tr>
<td>368</td>
<td>C</td>
</tr>
<tr>
<td>369</td>
<td>SUBROUTINE CALCU(J ICA, IPHI, PHI, ASI, R, U, V, W, DEN, TE, ED, VIS,</td>
</tr>
<tr>
<td>370</td>
<td>VISCOD, ITBS, ITBN, ITBM, ITBN, WRFON, IAI,</td>
</tr>
<tr>
<td>371</td>
<td>C</td>
</tr>
<tr>
<td>372</td>
<td>INCLUDE 'gridparam.h'</td>
</tr>
<tr>
<td>373</td>
<td>INCLUDE 'tcm.h'</td>
</tr>
<tr>
<td>374</td>
<td>DIMENSION PHI(NXN), FXNW(NY), DW(NY)</td>
</tr>
<tr>
<td>375</td>
<td>DIMENSION U(NXN), V(NXN), W(NXN), TE(NXN), ED(NXN),</td>
</tr>
<tr>
<td>376</td>
<td>DEN(NXN), R(NXN), A(6,6), B(6,6), VIS(NXN)</td>
</tr>
<tr>
<td>377</td>
<td>DIMENSION ITBS(NX), ITBN(NY), ITBM(NY), ITBN(NY)</td>
</tr>
<tr>
<td>378</td>
<td>DIMENSION F123(NY), F123(NY), F123(NY), F123(NY),</td>
</tr>
<tr>
<td>379</td>
<td>&amp;</td>
</tr>
<tr>
<td>380</td>
<td>&amp;</td>
</tr>
<tr>
<td>381</td>
<td>&amp;</td>
</tr>
<tr>
<td>382</td>
<td>&amp;</td>
</tr>
<tr>
<td>383</td>
<td>&amp;</td>
</tr>
<tr>
<td>384</td>
<td>&amp;</td>
</tr>
<tr>
<td>385</td>
<td>C</td>
</tr>
<tr>
<td>386</td>
<td>LI = 1</td>
</tr>
<tr>
<td>387</td>
<td>PHINE = PHI(IJ)</td>
</tr>
<tr>
<td>388</td>
<td>PHINE = PHI(IJ)</td>
</tr>
<tr>
<td>389</td>
<td>C</td>
</tr>
<tr>
<td>390</td>
<td>DO 11 J = 2, NJ</td>
</tr>
<tr>
<td>391</td>
<td>IJ = J</td>
</tr>
<tr>
<td>392</td>
<td>IJ = IJ - 1</td>
</tr>
<tr>
<td>393</td>
<td>IP = IJ + NJ</td>
</tr>
<tr>
<td>394</td>
<td>AREH = HAF*(ARE(IJ) + ARE(IJ))</td>
</tr>
<tr>
<td>395</td>
<td>DXX = XX(IJ) - XX(IJ-M)</td>
</tr>
<tr>
<td>396</td>
<td>DYY = YY(IJ) - YY(IJ-M)</td>
</tr>
<tr>
<td>397</td>
<td>DXXS = QTR*(XX(IJ) + XX(IJ-M) - XX(IJ-M))</td>
</tr>
<tr>
<td>398</td>
<td>DXXS = QTR*(YY(IJ) + YY(IJ-M) - YY(IJ-M))</td>
</tr>
<tr>
<td>399</td>
<td>C</td>
</tr>
<tr>
<td>400</td>
<td>C</td>
</tr>
<tr>
<td>401</td>
<td>GAMEV2 = 0.0</td>
</tr>
<tr>
<td>402</td>
<td>GAMEV2 = 0.0</td>
</tr>
<tr>
<td>403</td>
<td>C</td>
</tr>
<tr>
<td>404</td>
<td>C</td>
</tr>
<tr>
<td>405</td>
<td>IF(IPHI_EQ.ID) CE = CE</td>
</tr>
<tr>
<td>406</td>
<td>TERM = HAF<em>zeit(IJ) + CKE</em>DEN(IJ) + R(IJ)</td>
</tr>
<tr>
<td>407</td>
<td>GAMEV2 = TERM*V2(IJ)</td>
</tr>
<tr>
<td>408</td>
<td>GAMEV2 = TERM*V2(IJ)</td>
</tr>
<tr>
<td>409</td>
<td>GAMEV2 = TERM*V2(IJ)</td>
</tr>
<tr>
<td>410</td>
<td>END</td>
</tr>
<tr>
<td>411</td>
<td>DW = GAMEV2<em>DYE</em>2 + GAMEV2*DXX**2</td>
</tr>
<tr>
<td>412</td>
<td>C</td>
</tr>
<tr>
<td>413</td>
<td>C</td>
</tr>
<tr>
<td>414</td>
<td>C</td>
</tr>
<tr>
<td>415</td>
<td>C</td>
</tr>
<tr>
<td>416</td>
<td>C</td>
</tr>
<tr>
<td>417</td>
<td>IF(ITBM(J)) EQ 3.0 OR ITBN(J).EQ 4.0 GO TO 10</td>
</tr>
<tr>
<td>418</td>
<td>C</td>
</tr>
<tr>
<td>419</td>
<td>&amp;</td>
</tr>
<tr>
<td>420</td>
<td>&amp;</td>
</tr>
<tr>
<td>421</td>
<td>CONTINUE</td>
</tr>
<tr>
<td>422</td>
<td>ADD GRADIENT TERMS IN RII</td>
</tr>
<tr>
<td>423</td>
<td>C</td>
</tr>
<tr>
<td>424</td>
<td>C</td>
</tr>
<tr>
<td>425</td>
<td>RP = R(IJ)</td>
</tr>
<tr>
<td>426</td>
<td>C</td>
</tr>
<tr>
<td>427</td>
<td>C</td>
</tr>
<tr>
<td>428</td>
<td>C</td>
</tr>
<tr>
<td>429</td>
<td>C</td>
</tr>
</tbody>
</table>
Oct 12 1996 16:39  rsmod_2d  Page 7

427 IF (PHI.EQ. IV2 .AND. ICAL(IVR)) THEN
428 F222M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
429 F322M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)/RP
430 F322M(IJ)=WV(IJ)
431 ENDIF
432 C---- W2-EQUATION
433 IF (PHI.EQ. IW2 .AND. ICAL(IWR)) THEN
434 F332M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
435 F332M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)/RP
436 F332M(IJ)=WV(IJ)
437 F332M(IJ)=WV(IJ)
438 ENDIF
439 C---- U2-EQUATION
440 IF (PHI.EQ. IU2 .AND. ICAL(IUR)) THEN
441 F342M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
442 F342M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)/RP
443 F342M(IJ)=WV(IJ)
444 F342M(IJ)=WV(IJ)
445 ENDIF
446 C---- VW-EQUATION
447 IF (PHI.EQ. IVW) THEN
448 F213M(IJ)=V2(IJ)-W2(IJ)
449 F213M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*V2(IJ)-W2(IJ)/RP
450 F213M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*V2(IJ)-W2(IJ)
451 F213M(IJ)=V2(IJ)
452 F213M(IJ)=V2(IJ)
453 ENDIF
454 C---- UW-EQUATION
455 IF (PHI.EQ. IUW) THEN
456 F113M(IJ)=UW(IJ)
457 F113M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*UW(IJ)/RP
458 F113M(IJ)=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*UW(IJ)
459 F113M(IJ)=UW(IJ)
460 ENDIF
461 C
462 LI CONTINUE
463 C
464 DO 100 I=2,NJM
465 J=1
466 IJ=IM2(IJ)+J
467 IJ=IJ-2
468 IJ=IJ
469 FXE=FX(IJ)
470 FXE=FXE
471 ARE=ARE(IJ)+ARE(IJ)
472 DXK=XX(IJ)-XX(IJ)
473 DYN=YY(IJ)-YY(IJ)
474 DXE=XX(IJ)-XX(IJ)
475 DXF=XX(IJ)-XX(IJ)
476 GAMM2=GM2(IJ)+GM2(IJ)
477 GAMM2=GM2(IJ)+GM2(IJ)
478 GAMM2=GM2(IJ)
479 GAMM2=GM2(IJ)
480 CCK=C
481 IF (PHI.EQ. TED) CCK=C
482 IF (ED(IJ).NE.0.0) THEN
483 TERM+HEED(IJ)*CK*DEN(IJ)*(R(IJ)+R(IJ))
484 GAMM2=TERM*U2(IJ)
485 GAMM2=TERM*V2(IJ)
486 GAMM2=TERM*U2(IJ)
487 GAMM2=TERM*V2(IJ)
488 ENDIF
489 DO=GM2+DYN*2+GM2*DYN+2
490 FYS=1.0
491 PHINE-BH(IJ)+FXY
492 PHINE-BH(IJ)+FXY
493 SDM=0.0
494 IF (IV(IJ).EQ.3 .OR. ITBS(IJ).EQ.4) GO TO 110
495 SSM=GM2*DYN+DYN*GM2+DYN*GM2+(PHINE-PHINE(IJ))
496 SSM=GM2*DYN+DYN*GM2+(PHINE-PHINE(IJ))
497 CONTINUE
498 PHINE=PHINE(IJ)
499 PHINE=PHINE(IJ)
500 C

Oct 12 1996 16:39  rsmod_2d  Page 8

498 C---- GRADIENT TERMS IN RII
499 R=RAP*(R(IJ)+R(IJ))
500 C---- V2-EQUATION
501 IF (PHI.EQ. IV2 .AND. ICAL(IVR)) THEN
502 F222M=0.0
503 F223M=0.0
504 F322M=0.0
505 F322M=0.0
506 IF (RP.GT.0.0) THEN
507 F222M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
508 F322M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)/RP
509 F322M=V2(IJ)
510 F322M=V2(IJ)
511 ENDIF
512 C---- W2-EQUATION
513 IF (PHI.EQ. IW2 .AND. ICAL(IWR)) THEN
514 F213M=0.0
515 F213M=0.0
516 F213M=0.0
517 F213M=0.0
518 F213M=0.0
519 IF (RP.GT.0.0) THEN
520 F213M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
521 F322M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)/RP
522 F322M=V2(IJ)
523 F322M=V2(IJ)
524 ENDIF
525 C---- UW-EQUATION
526 IF (PHI.EQ. IUW) THEN
527 F113M=0.0
528 F113M=0.0
529 F113M=0.0
530 F113M=0.0
531 F113M=0.0
532 IF (RP.GT.0.0) THEN
533 F213M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)**2
534 F322M=DEN(IJ)*CK*TEED(IJ)*WV(IJ)*WV(IJ)
535 F113M=UW(IJ)
536 F113M=UW(IJ)
537 ENDIF
538 C---- VW-EQUATION
539 IF (PHI.EQ. IVW) THEN
540 F213M=0.0
541 F213M=0.0
542 F213M=0.0
543 F213M=0.0
544 F213M=0.0
545 IF (RP.GT.0.0) THEN
546 F213M=V2(IJ)-W2(IJ)
547 F213M=V2(IJ)-W2(IJ)
548 F213M=V2(IJ)-W2(IJ)
549 F213M=V2(IJ)-W2(IJ)
550 ENDIF
551 C---- UW-EQUATION
552 IF (PHI.EQ. IUW) THEN
553 F113M=0.0
554 F113M=0.0
555 F113M=0.0
556 F113M=0.0
557 F113M=0.0
558 IF (RP.GT.0.0) THEN
559 F213M=V2(IJ)-W2(IJ)
560 F213M=V2(IJ)-W2(IJ)
561 F213M=V2(IJ)-W2(IJ)
562 F213M=V2(IJ)-W2(IJ)
563 ENDIF
564 C---- THE MAIN LOOP - ASSEMBLY OF COEFFICIENTS AND SOURCES
565 C
566 DO 101 J=2,NJM

Printed by yqd449 from mrbig
IF(IPHI.EQ.IV2) THEN
    F222S=F222N
713  F322S=F322N
714  F522S=F522N
715  F622S=F622N
716  C
717  FDUMP+FKDE*VW(IJ)*2
718  FDUMP+FKDE*VW(IJ)*2
719  F222N=FDUMP+VW(IJ)/FP
720  FDUMP+FKDE*VW(IJ)*VW(IJ)/RP
721  FDUMP+FKDE*VW(IJ)*VW(IJ)/RP
722  F322N=FDUMP+VW(IJ)/FP
723  F522N=FDUMP+VW(IJ)/FP
724  F622N=FDUMP+VW(IJ)/FP
725  C
726  FDUMP+FKDE*VW(IJ)*2
727  FDUMP+FKDE*VW(IJ)*2
728  F222E+FDUMP+FXN+FDUMP*FXE
729  FDUMP+FKDE*VW(IJ)*VW(IJ)/RP
730  FDUMP+FKDE*VW(IJ)*VW(IJ)/RP
731  F322E+FDUMP+FXN+FDUMP*FXE
732  F522E+FDUMP+FXN+FDUMP*FXE
733  F622E+FDUMP+FXN+FDUMP*FXE
734  C
735  F2NS=F222N-222S
736  F3NS=F322N-322S
737  F5NS=F522N-522S
738  F6NS=F622N-622S
739  C
740  DF2DY=(F2NS*DF2M-F2DM*DFNS)/ARE(IJ)
741  DF3DY=(F3NS*DF3M-F3DM*DFNS)/ARE(IJ)
742  DF5DY=(F5NS*DF5M-F5DM*DFNS)/ARE(IJ)
743  DF6DY=(F6NS*DF6M-F6DM*DFNS)/ARE(IJ)
744  C
745  SU1=(1.-C2)*P22(IJ)+TH*C2*GEN(IJ)+
746  & TH*GEN(IJ)*C1*ED(IJ)+-2.*DEN(IJ)*VW(IJ)*VW(IJ)/RP
747  & (B1=TH*DEN(IJ)+ED(IJ)+C1*FEDK*V2(IJ))/V2(IJ)+SMALL
748  & C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
749  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
750  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
751  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
752  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
753  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
754  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
755  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
756  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
757  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
758  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
759  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
760  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
761  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
762  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
763  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
764  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
765  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
766  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
767  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
768  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
769  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
770  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
771  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
772  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
773  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
774  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
775  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
776  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
777  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
778  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
779  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
780  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)
781  & (C2*C2PUR(F1(IJ)+TH*GEN(IJ))+FUNK(IJ)

--- W-EQUATION SOURCES ---
<table>
<thead>
<tr>
<th>Location</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2415</td>
<td>SU(IJ)=GREAT<em>SIGN</em>UVREAL</td>
<td></td>
</tr>
<tr>
<td>2416</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>2417</td>
<td>1732 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2418</td>
<td>AM(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>2419</td>
<td>1730 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2420</td>
<td>C---- EAST BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>2421</td>
<td>DO 1740 J=2,NJM</td>
<td></td>
</tr>
<tr>
<td>2422</td>
<td>IJ=INM(NIM)+J</td>
<td></td>
</tr>
<tr>
<td>2423</td>
<td>GO TO (1741,1742,1742,1743) JTB(EJ)</td>
<td></td>
</tr>
<tr>
<td>2424</td>
<td>1741 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2425</td>
<td>SU(IJ)=SU(IJ)+AE(IJ)*UV(IJ+1J)</td>
<td></td>
</tr>
<tr>
<td>2426</td>
<td>BP(IJ)=BP(IJ)+AE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2427</td>
<td>GO TO 1742</td>
<td></td>
</tr>
<tr>
<td>2428</td>
<td>1743 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2429</td>
<td>DHX=EX(IJ)-EX(IJ-1)</td>
<td></td>
</tr>
<tr>
<td>2430</td>
<td>DDB=YY(IJ)-YY(IJ-1)</td>
<td></td>
</tr>
<tr>
<td>2431</td>
<td>ARW=SQRT(DBB<strong>2+DBB</strong>2)+SMALL</td>
<td></td>
</tr>
<tr>
<td>2432</td>
<td>DXX=DXX/ARW</td>
<td></td>
</tr>
<tr>
<td>2433</td>
<td>DDB=DDB/ARW</td>
<td></td>
</tr>
<tr>
<td>2434</td>
<td>VP2=U(IJ)*DBB+V(IJ)*DBB</td>
<td></td>
</tr>
<tr>
<td>2435</td>
<td>GRADV2=VWALL-ABS(VP2)</td>
<td></td>
</tr>
<tr>
<td>2436</td>
<td>SIN=1.0</td>
<td></td>
</tr>
<tr>
<td>2437</td>
<td>IF (GRADV2.LT.0.) SIN=-1.0</td>
<td></td>
</tr>
<tr>
<td>2438</td>
<td>TE(IJ)=ABS(TE(IJ))</td>
<td></td>
</tr>
<tr>
<td>2439</td>
<td>UVMAL=CV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2440</td>
<td>VVWAL=CVV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2441</td>
<td>UVMAL=CV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2442</td>
<td>UVMAL=CVV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2443</td>
<td>&amp; UVWAL=PFTE(IJ)+VT2E(IJ)+VW2E(IJ)+VF2E(IJ)</td>
<td></td>
</tr>
<tr>
<td>2444</td>
<td>SU(IJ)=GREAT<em>SIGN</em>UVREAL</td>
<td></td>
</tr>
<tr>
<td>2445</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>2446</td>
<td>1742 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2447</td>
<td>AE(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>2448</td>
<td>1740 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2449</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2450</td>
<td>RETURN</td>
<td></td>
</tr>
<tr>
<td>2451</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2452</td>
<td>C------- BOUNDARY CONDITIONS FOR VW-REYNOLDS STRESS</td>
<td></td>
</tr>
<tr>
<td>2453</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2454</td>
<td>1800 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2455</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2456</td>
<td>SOUTH BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>2457</td>
<td>DO 1810 J=2,NIM</td>
<td></td>
</tr>
<tr>
<td>2458</td>
<td>IJ=INM(NIM)+J</td>
<td></td>
</tr>
<tr>
<td>2459</td>
<td>GO TO (1811,1812,1812,1813) ITBS(I)</td>
<td></td>
</tr>
<tr>
<td>2460</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2461</td>
<td>SU(IJ)=SU(IJ)+AS(IJ)*UV(IJ+1J)</td>
<td></td>
</tr>
<tr>
<td>2462</td>
<td>BP(IJ)=BP(IJ)+AS(IJ)</td>
<td></td>
</tr>
<tr>
<td>2463</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2464</td>
<td>TE(IJ)=ABS(TE(IJ))</td>
<td></td>
</tr>
<tr>
<td>2465</td>
<td>UWV=CV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2466</td>
<td>VWAL=CVV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2467</td>
<td>VWREAL=UWREAL*PFTE(IJ)+VW2E(IJ)+VF2E(IJ)</td>
<td></td>
</tr>
<tr>
<td>2468</td>
<td>SU(IJ)=GREAT*UVREAL</td>
<td></td>
</tr>
<tr>
<td>2469</td>
<td>BP(IJ)=GREAT</td>
<td></td>
</tr>
<tr>
<td>2470</td>
<td>1812 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2471</td>
<td>AS(IJ)=0.</td>
<td></td>
</tr>
<tr>
<td>2472</td>
<td>1810 CONTINUE</td>
<td></td>
</tr>
<tr>
<td>2473</td>
<td>C------- NORTH BOUNDARY</td>
<td></td>
</tr>
<tr>
<td>2474</td>
<td>DO 1820 J=2,NIM</td>
<td></td>
</tr>
<tr>
<td>2475</td>
<td>IJ=INM(NIM)+J</td>
<td></td>
</tr>
<tr>
<td>2476</td>
<td>GO TO (1821,1822,1822,1823) ITBS(I)</td>
<td></td>
</tr>
<tr>
<td>2477</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2478</td>
<td>SU(IJ)=SU(IJ)+AN(IJ)*UV(IJ+1J)</td>
<td></td>
</tr>
<tr>
<td>2479</td>
<td>BP(IJ)=BP(IJ)+AN(IJ)</td>
<td></td>
</tr>
<tr>
<td>2480</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2481</td>
<td>TE(IJ)=ABS(TE(IJ))</td>
<td></td>
</tr>
<tr>
<td>2482</td>
<td>UWV=CV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2483</td>
<td>VWAL=CVV*TE(IJ)</td>
<td></td>
</tr>
<tr>
<td>2484</td>
<td>VWREAL=UWREAL*PFTE(IJ)+VW2E(IJ)</td>
<td></td>
</tr>
</tbody>
</table>
2704 SUBROUTINE SOLV(A,BB,N)
2705 DIMENSION A(N,N),B(N),C(N),BB(N),X(N),MME(N)
2706 EP .1, K-19
2707 DO 10 J=1,N
2708 MME(J)=J
2709 DO 20 J=1,N
2710 Y=0.
2711 DO 30 J=1,N
2712 IF(ABS(A(I,J)).LT.ABS(Y)) GOTO 30
2713 K=J
2714 Y=A(I,J)
2715 30 CONTINUE
2716 C
2717 IF(ABS(Y).LT.EP) THEN
2718 WRITE(*,*)
2719 WRITE(*,*)
2720 DO 35 I=1,N
2721 WRITE(*,1000) (A(I,J),JA=1,N)
2722 35 CONTINUE
2723 PRINT*, 'THERE IS NO CONVERSE MATRIX'
2724 STOP 2222
2725 ENDF
2726 C
2727 Y=1./Y
2728 DO 40 J=1,N
2729 C(J)=A(J,K)
2730 A(K,J)=A(J,K)
2731 A(J,J)=-C(J,J)*Y
2732 B(J)=A(I,J)*Y
2733 DO 40 40 J=1,N
2734 A(I,I)=Y
2735 J=MME(I)
2736 MME(I)=MME(K)
2737 MME(K)=J
2738 DO 50 I=1,K
2739 IF(K.EQ.I) GOTO 51
2740 DO 12 J=1,N
2741 IF(J.EQ.I) GOTO 13
2742 A(K,J)=A(K,J)-B(J)*C(K)
2743 12 CONTINUE
2744 11 CONTINUE
2745 20 CONTINUE
2746 DO 33 I=1,N
2747 DO 44 K=1,N
2748 IF(MME(K).EQ.I) GOTO 55
2749 44 CONTINUE
2750 55 IF(K.EQ.I) GOTO 33
2751 DO 66 J=1,N
2752 W=A(I,J)
2753 A(I,J)=A(K,J)
2754 66 A(K,J)=W
2755 IN=WME(I)
2756 MME(I)=MME(K)
2757 MME(K)=IN
2758 33 CONTINUE
2759 1000 FORMAT(4X,1P5E13.4)
2760 DO 50 I=1,N
2761 X(I)=0.
2762 DO 60 I=1,N
2763 50 X(I)=X(I)+A(I,J)*BB(J)
2764 DO 60 60 I=1,N
2765 60 BB(I)=X(I)
2766 RETURN
2767 C
2768 SUBROUTINE AMODIFY(SUASM,SVASM,SWASM,
2841  ENDIF
2842  600 CONTINUE
2843  C---- WEST BOUNDARY
2844  DO 620 J=2,N,
2845  11+1 MJ(ID) +J
2846  IF (JTBW(J).EQ.4) THEN
2847  DXB=X(IJ-NJ)-X(IJ-NJ-1)
2848  DYB=Y(IJ-NJ)-Y(IJ-NJ-1)
2849  APH=SQRT(DXB**2+DYB**2)
2850  DXB=DXB/APH
2851  DYB=DYB/APH
2852  CONST=DEN(ID)*CMU25*SQRT(TE(J))
2853  YPLS=DM(J)*CONST/VISCOS
2854  IF (YPLS.LE.11.63.OR.LAY2) THEN
2855  TOCEF=VISCOS/DM(J)
2856  ELSE
2857  UPLUS=LOG(ELOG*YPLS)/CAPP
2858  TOCEF=CONST/UPLUS
2859  ENDIF
2860  VPINT=U(IJ)*DXB+V(IJ)*DYB
2861  VPINT=VPINT+W(IJ)
2862  VPINT=ABS(VPINT-SQRT(U(IJ-NJ)+U(IJ-NJ)+
2863       1)*V(IJ-NJ)**2+V(IJ-NJ)**2+V(IJ-NJ))
2864  GENTW(J)=TOCEF*CONST*ABS(VPINT)/(CAPP*DEN(ID)*DM(J))
2865  ENDIF
2866  C---- EAST BOUNDARY
2867  IJ=1MJ(ID) +J
2868  IF (JTEB(J).EQ.4) THEN
2869  UP=U(IJ)
2870  VP=V(IJ)
2871  WP=W(IJ)
2872  UWALL=U(IJ+1)
2873  VWALL=V(IJ+1)
2874  WWALL=W(IJ+1)
2875  TEPW=SQRT(TE(J))
2876  UNIS=DNE(J)
2877  RR=HAP*(R(IJ)+R(IJ-1))
2878  DENS=DEN(ID)
2879  DBX=X(IJ-1)-X(IJ)
2880  DYB=Y(IJ-1)-Y(IJ)
2881  APH=SQRT(DXB**2+DYB**2)
2882  DXB=DXB/APH
2883  DYB=DYB/APH
2884  CONST=DEN(ID)*CMU25*SQRT(TE(J))
2885  YPLS=DM(J)*CONST/VISCOS
2886  IF (YPLS.LE.11.63.OR.LAY2) THEN
2887  TOCEF=VISCOS/DM(J)
2888  ELSE
2889  UPLUS=LOG(ELOG*YPLS)/CAPP
2890  TOCEF=CONST/UPLUS
2891  ENDIF
2892  VPINT=U(IJ)*DXB+V(IJ)*DYB
2893  VPINT=VPINT+W(IJ)
2894  VPINT=ABS(VPINT-SQRT(U(IJ-NJ)+U(IJ-NJ)+
2895       1)*V(IJ-NJ)**2+V(IJ-NJ)**2+V(IJ-NJ))
2896  GENTE(J)=TOCEF*CONST*ABS(VPINT)/(CAPP*DEN(ID)*DNE(J))
2897  ENDIF
2898  620 CONTINUE
2899  RETURN
2900  END
2901  C
2902  C gridparam.h
2903  C
2904  C
2905  C
2906  C
2907  C
2908  C
2909  C
2910  C
2911  C

rsmod_2d
CHAPTER 6
3D $k$-$\varepsilon$ Turbulence Model

Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction</td>
<td>81</td>
</tr>
<tr>
<td>6.2 Theory and Model Equations</td>
<td>81</td>
</tr>
<tr>
<td>6.3 Module Evaluation</td>
<td>82</td>
</tr>
<tr>
<td>6.4 References</td>
<td>82</td>
</tr>
<tr>
<td>Figures</td>
<td></td>
</tr>
<tr>
<td>Appendix E</td>
<td>83</td>
</tr>
</tbody>
</table>
6.1 Introduction

In this section a description of the standard \( k-\varepsilon \) turbulence model that is coded as a self contained computer program to compute turbulent flow quantities in three-dimensional, body-fitted geometry is given. Module structure and variables used are given in the Appendix. The module was successfully tested as a self-contained unit using the REACT code[1].

6.2 Theory and Model Equations

The \( k-\varepsilon \) turbulence model used is based on the standard two equation \( k-\varepsilon \) model of Launder and Splading [2]. For a steady, incompressible flow the transport equations for the turbulent kinetic energy \( k \) and energy dissipation \( \varepsilon \) can be written in generalized Cartesian coordinates as;

\[
\frac{\partial pU_i k}{\partial x_j} = \frac{1}{r} \frac{\partial}{\partial x_j} \left( \mu + \frac{\mu_t}{\sigma_k} \frac{\partial k}{\partial x_j} \right) + G - \rho \varepsilon \quad (1)
\]

\[
\frac{\partial pU_i \varepsilon}{\partial x_j} = \frac{1}{r} \frac{\partial}{\partial x_j} \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial x_j} \right) + G - \rho \varepsilon \quad (2)
\]

where \( G \) denotes the rate of production of turbulent kinetic energy and is expressed as;

\[
G = \mu_t \frac{\partial U_i}{\partial x_j} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)
\]

The empirical constants, \( \sigma_k, \sigma_\varepsilon, C_1 \) and \( C_2 \) have values 1.0, 1.0, 1.44 and 1.92 respectively.

The above equations are valid only in the fully turbulent region away from the wall. Therefore the wall function method (similar to that described in Chapter 2 for the 2D \( k-\varepsilon \) module) is used to model the damping effects of the thin sublayer region close to the wall.
6.3 Module Evaluation

The 3D $k$-$\varepsilon$ turbulent module was evaluated by interfacing the module with the REACT code as the CFD solver and producing the same results that were generated previously with the full REACT code for a centrifugal impeller calculations (Chen et. al [3]). Figure 1 shows the grid topology of the impeller studied with the shroud removed and Figure 2, shows the reduced pressure plot. In general Chen et al's calculations showed good comparisons with experimental data obtained from laser velocimetry in a water test rig.

REFERENCES


Figure 1. Impeller grid topology
Figure 2. Reduced pressures
APPENDIX E

3D $k$-$\varepsilon$ Turbulence Module Deck

This module consists of two separate programs KEMOD3 and MODIFY, which have to be linked to the main flow solver. A description of each file will be given next.

Program KEMOD3

This is basically the solver for the $k$ and $\varepsilon$ - transport equations. It reads through its argument list different variables from the calling flow solver. These variables are described below.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIM</td>
<td>Number of cell nodes in the I- or $\xi$-coordinate lines. (input from the flow solver)</td>
</tr>
<tr>
<td>NJM</td>
<td>Number of cell nodes in the J- or $\eta$-coordinate lines. (input from the flow solver)</td>
</tr>
<tr>
<td>NJM</td>
<td>Number of cell nodes in the k- or $\zeta$-coordinate lines. (input from the flow solver)</td>
</tr>
<tr>
<td>X</td>
<td>Grid node locations in the x or $\xi$-direction, dimensioned to $X(JXYZ)$ ($JXYZ=NX<em>NY</em>NZ$) (input from flow solver)</td>
</tr>
<tr>
<td>Y</td>
<td>Grid node locations in the y or $\eta$-direction, dimensioned to $Y(JXYZ)$ (input from flow solver)</td>
</tr>
<tr>
<td>Y</td>
<td>Grid node locations in the z or $\zeta$-direction, dimensioned to $Y(JXYZ)$ (input from flow solver)</td>
</tr>
<tr>
<td>U</td>
<td>x-direction velocity ($u$), dimensioned as $U(JXYZ)$ (input from flow solver)</td>
</tr>
<tr>
<td>V</td>
<td>y-direction velocity ($v$), also dimensioned as $V(JXYZ)$ (input from flow solver)</td>
</tr>
<tr>
<td>W</td>
<td>z-direction velocity ($w$), dimensional $W(JXYZ)$ (input from flow solver)</td>
</tr>
<tr>
<td>TE</td>
<td>Turbulence kinetic energy $k$, dimensioned TE(JXYZ) (calculated in the module and returned to the flow solver)</td>
</tr>
</tbody>
</table>
ED Turbulent energy dissipation rate $\varepsilon$, dimensioned $\text{ED(JXYZ)}$ (calculated in the module and returned to the flow solver)

URFK Under-relaxation factor for $k$-equation (input from flow solver)

URFE Under-relaxation factor for $\varepsilon$-equation (input from flow solver)

PRTK Prandtl/Schmidt number for turbulent energy-equation, assumed known (input from flow solver)

PRTE Prandtl/Schmidt number for turbulent energy dissipation equation, assumed known (input from flow solver)

G $= 1.0$ if second order upwinding is desired

$= 0.0$ if first order upwinding is used

(input from flow solver. Usually calculation of $k$ and $\varepsilon$ are not very sensitive to the order of upwinding used)

F1 Mass flux variable at cell faces in x- or $\zeta$-direction, dimensioned $\text{F1(JXYZ)}$ (input from flow solver)

F2 Mass flux variable at cell faces in y or $\eta$-direction, dimensioned $\text{F2(JXYZ)}$ (input from flow solver)

F3 Mass flux variable at cell faces in z or $\zeta$-direction, dimensioned $\text{F3(JXYZ)}$ (input from flow solver)

ITER Iteration number (input from flow solver)

VISCOS Dynamic viscosity (input from flow solver)

VIS Eddy viscosity, dimensioned $\text{VIS(JXYZ)}$ (calculated in the module and returned to the main solver)

C1 Turbulence model constant, $C_1$ (input from flow solver)

C2 Turbulence model constant, $C_2$ (input from flow solver)

CMU Turbulence model constant, $C_\mu$ (input from flow solver)

BCFE Boundary condition flag along east boundary (or y-z plane). It must have one for each boundary node set to: 1-inlet, 2-outlet, 3-symmetry and 4-wall e.g., for an outlet boundary condition on the east boundary set $\text{IBCE}$ to $(\text{NY*NZ})*2$, and similarly for other boundaries, dimensioned $\text{BCFE(JYZ=NY*NZ)}$ (input from flow solver)

BCFW Boundary condition flag along west boundary, dimensioned $\text{BCFW(JYZ)}$ (input from flow solver)

BCFS Boundary condition flag along the south boundary, dimensioned $\text{BCFS(JXZ=NX*NZ)}$ (input from flow solver)
BCFN  Boundary condition flag along north boundary, dimensioned BCFN(JXZ) (input from flow solver)
BCFB  Boundary condition flag along bottom boundary (or x-y plane), dimensioned BCFB(JXY=NX*NY) (input from flow solver)
BCFT  Boundary condition flag along top boundary (or x-y plane), dimensioned BCFT(JXY=NX*NY) (input from flow solver)

The module is interfaced with the main flow solver by a call to KEMOD3 with its arguments. For iterative flow solvers the module is called within the iteration sequence after the solution of the momentum equations where the mean velocities are passed to the module. There are different flow solvers utilizing different schemes from staggered to nonstaggered grid arrangement and for nonorthogonal coordinate system there are at least three alternatives to the choice of the velocity components;

   i. Cartesian velocity components
   ii. Contravariant velocity components
   iii. Covariant velocity components

The Cartesian velocity components are the most widely used and have the advantage of simple formulation of the governing equations. Whatever the arrangement used, mass fluxes at cell faces are required and passed to the module as F1, F2 and F3 in all directions. The location of other variables such as $k$ and $\varepsilon$ are at the cell center or cell nodes.

The module starts by reassigning variable names passed to it from flow solver to names that are shared with the different subroutines of the module in a common statement file "kemod.h". Then a check is made if it is the first iteration in which case the grid file "GRIDG" is called -after passing the grid node locations X, Y and Z- in order to calculate grid related quantities which will be explained later. The need to call GRIDG can be waived if all the grid data are passed to the module. That is all the information about the grid such as interpolation factors FX, FY and FZ, cell volumes (VOL) and normal distances of first grid point from grid boundaries (DNS from south boundary, DNN - from north boundary, DNW - from west boundary, DNE - from east boundary, DNB - from bottom boundary and DNT - from top boundary).

After this a call to subroutine CALCE is made to calculate the turbulent kinetic energy $k$ (with the identifier IPHI=1). The energy dissipation equation is solved next by a call to subroutine CALCE
again with the identifier IPHI=2. The turbulent viscosity is updated next by calling subroutine MODVIS. A brief description of each subroutine is given next.

**Subroutine GRIDG**

Before calling this subroutine, the coordinates of all grid nodes, defined in reference to a fixed Cartesian coordinate frame are read. Figure 3 shows the position of cell and grid nodes. The west-to-east, south-to-north and bottom-to-top directions correspond to the ascending indexing order of i, j and k, respectively, forming a right-handed coordinate system.

This subroutine is called only once to calculate coordinates of grid nodes (intersection of grid lines) and geometrical properties of the grid (cell volumes, interpolation factors, normal distances of near-boundary cell nodes from boundary). All variables including grid node coordinates are converted to one-dimensional arrays. The position of any node in one-dimensional array is therefore defined as:

\[ IJK = (I-1)*NJ + (K-1)*NI*NJ + J \]

where NI, NJ and NK are the maximum number of grid nodes in the i, j and k directions respectively.

The actual number of grid nodes is one row and one column less than for all cell nodes. For I = NI, J = NJ and K = NK fictitious grid nodes are introduced which have the same coordinates as actual nodes on NI-1 in I-direction, NJ-1 in the J-direction and NK-1 in K-direction.

The subroutine then calculates interpolation factors which are associated with cell nodes and are used in the main program to calculate values of dependent variables at locations other than cell nodes (cell centers). Cell volumes are calculated next followed by calculations of normal distances of near-boundary nodes from all the six outer boundaries.
Subroutine CALCE (PHI, IPHI)

This subroutine solves the linearized and discretized transport equations for the turbulent energy \( k \) and the energy dissipation rate \( \varepsilon \). The two dummy parameters in the calling statement, PHI and IPHI, represent arrays containing dependent variables for which the equation is to be solved. The subroutine sets up the convective and diffusive coefficients over the entire field, then it calculates the source terms for either \( k \) or \( \varepsilon \) transport equations. A call is made to MODKE or MODED in order to modify the sources for \( k \) and \( \varepsilon \) equations respectively.

The discretized equations have the form

\[ A_p \Phi_p = \sum_{i=E,W,N,S,T,B} A_i \Phi_i + S \Phi \]

where the coefficients \( A_i \) (i=E,W,N,S,T,B) contain both the convective and diffusive fluxes. These equations are assembled and solved by calling subroutine SOLSIP which is based on Stone's Strongly Implicit Solver [4].

Subroutine SOLSIP

This subroutine solves the system of linear algebraic equations for \( k \) and \( \varepsilon \) using Stone's Implicit Procedure [4]. The array RES (IJK) is used to store the residuals. The sum of absolute residuals "RES1" calculated in the first pass through this part of the routine is used as a measure of convergence of the solution process as a whole and this value is stored in RESOR (IPHI). This variable RESOR (IPHI) is passed to the main solver and if desired can be normalized and compared with the maximum error allowed there. If necessary inner iterations counter L and the sum of absolute residuals RES1 are printed out to monitor the rate of convergence of \( k \) and \( \varepsilon \) solution. If the ratio RSM is greater than the maximum allowed for the variable in question, SOR (IPHI), and the number of inner iterations is smaller than a prescribed maximum, NSWP (IPHI), then the routine repeats the sequence of calculating the residuals, increment vectors and updating the dependent variable.
Subroutine MODVIS

This section calculates the effective viscosity and is called after calculating \( k \) and \( \varepsilon \). At locations where \( \varepsilon \) is close to zero (i.e., \( \leq 10^{-30} \)) viscosity is set to zero. A provision is made for under relaxing changes in effective viscosity which may help to stabilize oscillations and improve convergence rate.

Subroutines MODK and MODED

These subroutines are called from subroutine CALCE and they set the boundary conditions for \( k \) and \( \varepsilon \). For the kinetic energy equation for example, the bottom boundary is checked first for one of the options below;

1. An inflow boundary \( BCFB(IJ) = 1 \) \((IJ = (I-1)*NJ+J)\), where the source term is set to accept the inlet values at the x-y plane (bottom boundary K=1).

2. Outflow boundary \( BCFB(IJ) = 2 \), where zero gradient in the z-direction is employed.

3. Symmetry boundary, \( BCFB(IJ) = 3 \), where gradients normal to symmetry x-y plane are zero.

4. Wall boundary, \( BCFB(IJ) = 4 \), where the turbulent kinetic energy production (per unit volume) term \( GENTB(I) \) calculated from subroutine WALLFN in program MODIFY is added to the rest of the source term \( SU(IJK) \).

Boundary conditions for the \( \varepsilon \)-equation are similar to those of \( k \) except at the wall where they are set to appropriate values for each near wall treatment.

Program MODIFY

This program is compiled separately and is called from the u, v and w momentum solver. It basically updates the flux source term of the discretized momentum equation due to wall shear stresses. If the u-momentum equation for example is discretized in the form
\[ A_p^* u_p = \sum_{i=EWNSTB} A_i^* u_i + S_u^* \]

where \( P, E, W, N, S, T, B \) are cell nodes, and \( A_p^* \) and \( A_i^* \)s contain convective and diffusive coefficients. \( S_u^* \) is the source term containing pressure gradients and cross-derivative diffusion terms and convective terms for second-order upwinding scheme. This source term is usually linearized as \( S_u^* = S_u - B_p u_p \). The term \( B_p \) is usually moved to the left hand side of the equation and modifies the diagonal coefficient \( A_p = A_p^* + B_p \), and the equation can be written as

\[ A_p u_p = \sum_{i=EWNSTB} A_i^* u_i + S_u \]

Then \( S_u \) and \( B_p \) are passed to subroutine MODIFY where they are modified if a wall is present (e.g., BCFB(IJ) = 4 for bottom boundary).
Figure 3. Cell volume and coordinate system
Oct 12 1996 16:42  sskemod_3d  Page 1

1 C
2 C 3D-SINGLE-SCALE K-E TURBULENCE MODULE C
3 C
4 C
5 C Rocketdyne CFD Technology Center
6 C
7 C
8 C
9 C
10 SUBROUTINE KEMOD (NIM, NJM, NKM, C
11 & X, Y, Z, U, V, W, TE, TD, C
12 & URNX, URNF, PRX, PTX, G, F1, F2, F3, ITER, C
13 & VISCOS, VIS, Cl, C2, CMU, C
14 & BCF, BCFW, BCFS, BCFP, BCFS, BCFIT)
15 C
16 C INCLUDE 'kemod.h'
17 C
18 C DIMENSION LI(JX), L(JK), C
19 & X(JKXZ), X(JKXZ), X(JYXZ), VOL(JXJ); C
20 & FY(JXJ), F(JXJ), Z(JYXZ)
21 C
22 DIMENSION U(JXJ), V(JYJ), W(JYJ)
23 C
24 C CALCULATE GRID GEOMETRIC VARIABLES INITIALLY
25 C
26 C IF(ITER LE 1) THEN
27 C CALL GRIDG
28 ENDIF
29 C
30 C
31 C CALL KINETIC ENERGY SOLVER
32 C CALL CALCE(TE,1)
33 C CALL ENERGY DISSIPATION SOLVER
34 C CALL CALCE(ED,2)
35 C
36 C UPDATE AND CALCULATE THE EDDY VISCOSITY C
37 C CALL MODVIS
38 C
39 C
40 C RETURN
41 C END
42 C
43 C
44 C SUBROUTINE GRIDG
45 C INCLUDE 'kemod.h'
46 C
47 C NI = NIM + 1
48 C NJ = NIM + 1
49 C NK = NIM + 1
50 C DO 10 I=1,NI
51 L(I)=1-(I-1)*NI
52 10 CONTINUE
53 C
54 C DO 20 K=1,NK
55 L(K)=1-(K-1)*NK
56 20 CONTINUE
57 C
58 C DO 25 K=1,NK
59 C DO 35 J=1,NJ
60 I=LI(NJ)+LK(K)+J
61 I=M+LI
62 X(I)=X(I)+F
63 Y(I)=Y(I)+G
64 Z(I)=Z(I)+H
65 25 CONTINUE
66 C
67 C DO 40 I=1,NI
68 C DO 50 J=1,NJ
69 C
70 C
71 C

Oct 12 1996 16:42  sskemod_3d  Page 2

72 C
73 C
74 C
75 C CONTINUE
76 C
77 C
78 C
79 C
80 C
81 C
82 C
83 C 3D
84 C
85 C
86 C
87 C
88 C
89 C
90 C
91 C
92 C
93 C
94 C VOL(I1,A2,A3,B1,B2,B3,Q2,Q3) = (A2*B3-B2*A3)*Q1+ (B1*A3-A1*B3)*Q2 C
95 C + (A1*B2-A2*B1)*Q3
96 C
97 C C CALCULATION OF CELL VOLUMES
98 C
99 C
100 C
101 C
102 C
103 C
104 C
105 C
106 C
107 C
108 C
109 C
110 C
111 C
112 C
113 C
114 C
115 C
116 C
117 C
118 C
119 C
120 C
121 C
122 C
123 C
124 C
125 C
126 C
127 C
128 C
129 C
130 C
131 C
132 C
133 C
134 C
135 C
136 C
137 C
138 C
139 C
140 C
141 C
142 C

sskemod_3d
099 C WALL
100 C WALL
101 C WALL
102 C WALL
103 C PROD.
104 C PROD.
105 C PROD.
106 C PROD.
107 C PROD.
108 C PROD.
109 C PROD.
110 C PROD.
111 C PROD.
112 C PROD.
113 C PROD.
114 C PROD.
DO I=1,NM
1267
1268
II = (I-1)*NK
1269
1270
DO K2,NMK
1271
1272
IK = II + K
1273
1274
IK = LK(K) + LI(I) + 2
1275
1276
IDL = LK - NJ
1277
1278
IMK = IJK - NJ
1279
1280
IMK = IJK - NJ
1281
1282
IMK = IJK - NJ
1283
1284
IF(BCF(VK,KK,EQ,4)) THEN
1285
1286
1287
CALL WALLFN (LM, LB, DX1, DX2, DY1, DY2, DX1, DX2, CAT, DEN, GENT, DEN, & CMUX/5, VISC, ELOG, YPLS, SPU, SUV, YSP, SUV, YSP, SPU, SPU, SUW, SUW, TAU)
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
& CMOUS, VISC, ELOG, YPLS, SP, SUJ3, SUV, SPW, SWW, TAU)
1388  BP2(IJK) = BP2(IJK) + SPW
1389  SUI(IJK) = SUI(IJK) + SUW
1390  SVF2(IJK) + SPU
1391  SUV2(IJK) = SUW
1392  SFV(IJK) + SPU
1393  GENT(IJK) = GENT
1394  AR(IJK) = 0.0
1395  ENDF
1396  ENDDO
1397  ENDSW
1398  END
1399
1400 C----------------------------------------
1401 SUBROUTINE WALLFP(IW, LB, DX1, DX2, DY1, DY2, DZ1, DZ2, CAPA,
1402 & DEN, GENT, DELN, CMU35, VISC, ELOG, YPLS, SP, SUW,
1403 & SPW, SUW, SWW, TAU)
1404 C----------------------------------------
1405 INCLUDE 'kemod.h'
1406 C-----WALL CELL FACTOR AREA
1407 XAN=DX1*DX2*DY1*DY2*DZ1
1408 YAN=DX2*DZ1*DX1*DZ2
1409 TAN=DX1*DY2*DX2*DY1
1410 ARN=0.5*SQRT(XAN**2+YAN**2+ZAN**2)
1411 ARNH=0.5/ARN
1412 C-----COMPONENTS OF UNIT NORMAL VECTOR
1413 ALFAN=YN/ARN
1414 BETAN=YAN/ARN
1415 GAMAN=ZN/ARN
1416 C-----CALCULATE Y + AND LAMBDA-WALL COEFF.
1417 CONST=DEM*CMOUS*SQRT(TE(LM))
1418 YPLS=DELN*CONST/(VISC*1.E-30)
1419 TCOEF=VISC/DELN
1420 IF(YPLS.GT.11.63) TCOEF=CONST*CAPA/(ALGOL(ELOG*YPLS))
1421 TAN=TAN+ARN
1422 C-----SOURCE TERMS FOR VELOCITIES
1423 SPU=TAN*1.0-ALFAN*2)
1424 SPV=TAN*1.0-BETAN*2)
1425 SPW=TAN*1.0-GAMAN*2)
1426 SUW=TAN+ALFAN*2)
1427 SUC=TAN+ALFAN*(VAN-ALFAN*W(LM)-GAMAN*W(LM))
1428 SUM=TAN+ALFAN*(ALFAN*V(LM)+GAMAN*W(LM))
1429 C-----VELOCITY PARALLEL TO WALL
1430 UP=SQRT((SPU**2(LM)-SUC)**2+SPV**2(LM)-SUC)**2+2
1431 (* (SPW**2(LM)-SUM)**2)/(|TAR+1.0-30)
1432 UP=ABS(UP-SQRT(U(LM))**2+V(LM)**2+2)
1433 C-----WALL SHEAR STRESS AND GENER. TERM
1434 SUU=SUW*TAR+U(LB)
1435 SUV=SUW+V(LB)
1436 SUM=SUW+TAR+V(LB)
1437 TAU=U+V
1438 GENTE=TAU+DELN)
1439 RETURN
1440 END
1441 C----------------------------------------
1442 C
1443 C
1444 C-- kemod.h
1445 C
1446 PARAMETER (IJX=70)
1447 PARAMETER (IJY=41)
1448 PARAMETER (IJZ=41)
1449 PARAMETER (JXY=JX*JY)
1450 PARAMETER (JXZ=JX*JZ)
1451 PARAMETER (JYZ=JY*JZ)
1452 PARAMETER (JXYZ=JX*JY*JZ)
1453 C
1454 COMMON NJ, NJK, NNM, NNM, NJK, NNM, NJK, L(JXX), L(JXY), IU, IV, IW,
1455 * IP, IJE, IED, IVIS, IEN, IIT, MXIT, ITSTEP, ITEST, ICAL(10).

sskemod_3d

Oct 12 1996 16:42  sskemod_3d  Page 21

1456  * IMON,JMON,KNM, JXKMON, JXKPR, IDIR, NSWP(I), G.SOR(2)
1457  * X(JXYZ),Y(JXYZ),Z(JXYZ),F(JXYZ),Y(JXYZ),F(JXYZ),F(JXYZ),
1458  * VOL(JXYZ),DP1(JXYZ),DP2(JXYZ),DP3(JXYZ),
1459  * AR(JXYZ),AM(JXYZ),AM(JXYZ),AS(JXYZ),AP(JXYZ),AB(JXYZ),
1460  * AP(JXYZ),SU(JXYZ),BE(JXYZ),BP(JXYZ),BRE(JXYZ),
1461  * SWU(JXYZ),DB(JXYZ),DN(JXYZ),
1462  * RESOR(2),SNORIN(2),SCRF(2),URF(2),SAF,CTR,ANT,OMEGA,
1463  * ALFA, DENSIT, SORIN, VISCOS, READI, WRITT, IOBST, GREAT, SMALL,
1464  * C1,C2,CAPP,CMU1,CMU25,CMY75,SWJ(JXYZ),SWU(JXYZ),WBJ(JXYZ),
1465  * WBJ(JXYZ),WBJ(JXYZ),FZ2B(JXYZ),FJXW(JXYZ),
1466  * COMMNVK, VJW(JXYZ),VSN(JXYZ),VJXYZ, DXJXYZ,
1467  * FJXYZ, T(E(JXYZ), W1(JXYZ), T(1), P1(JXYZ), P2(JXYZ), F3(JXYZ),
1468  * FJXYZ)
CHAPTER 7

3D Algebraic Stress Turbulence Model

Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Introduction</td>
<td>91</td>
</tr>
<tr>
<td>7.2 Theory and Model Equations</td>
<td>91</td>
</tr>
<tr>
<td>References</td>
<td>94</td>
</tr>
<tr>
<td>Appendix F</td>
<td>95</td>
</tr>
</tbody>
</table>
7.1 Introduction

In this section a description is given of the three-dimensional Algebraic Stress turbulence Model (ASM) based on the work of Rodi [1]. The model is coded as a self contained computer program to compute turbulent flow quantities when interfaced with a CFD solver. Detailed description of the module structure, variables used and how to interface the module with CFD flow solvers are given in the Appendix.

The module uses as input the mean flow properties, as computed by conventional CFD solvers, and calculates the Reynolds stresses, turbulent kinetic energy and the energy dissipation. It is structured to be self-contained and compatible with many CFD codes. The module has not been tested thoroughly due to the ending of the contract earlier than scheduled. Some testing of the module has been done at UAH but that also has been put on hold. However, the module as assembled is capable of interfacing with a number CFD solvers.

The module computes turbulent flow quantities in three-dimensional body-fitted geometry with or without rotation about any one of the three axis. The standard wall functions is used for the near wall treatment.

7.2 Theory and Model Equations

The Algebraic Stress (ASM) module discussed here is based on the work of Rodi [1]. The idea is to simplify or truncate the Reynolds stress equation by approximating the convective and diffusive transport of the Reynolds stresses $\bar{u}_i\bar{u}_j$ in terms of the corresponding transport of turbulent energy. This allows the transport equation for the stresses to be expressed as a set of algebraic formulae containing the turbulence energy and its rate of dissipation as unknowns in the form:

$$\frac{\bar{u}_i\bar{u}_j}{(P-\epsilon)} = \frac{k}{(P-\epsilon)} \left[ P_{ij} - \frac{2}{3} \delta_{ij} \epsilon + \Phi_{ij} \right]$$

where $P_{ij} =$ Production and $P = \frac{1}{2} P_{kk}$

$$\Phi_{ij} = \text{pressure-strain redistribution}$$

$$\Phi_{ij} = \Phi_{ij,1} + \Phi_{ij,2} + \Phi_{ij,1w} + \Phi_{ij,2w}$$
Rotta's linear return-to-isotropy concept for the non-linear part

\[ \Phi_{ij,1} = -C_1 \frac{e}{k} (\overline{u_i u_j} - \frac{2}{3} k \delta_{ij}) \]

is used and the "isotropization of production" concept for the linear "rapid" part

\[ \Phi_{ij,2} = -C_2 (P_{ij} - \frac{2}{3} P \delta_{ij}) \]

is used. Gibson and Launder [2] concept for the wall reflection terms is used as

\[ \Phi_{ij,1w} = C_{lw} \rho \frac{e}{k} (\overline{u_k u_m} n_k n_m \delta_{ij} - \frac{3}{2} \overline{u_k u_i} n_k n_j - \frac{3}{2} \overline{u_k u_j} n_k n_i) f \]

\[ \Phi_{ij,2w} = C_{2w} (\Phi_{km,2} n_k n_m \delta_{ij} - \frac{3}{2} \Phi_{ik,2} n_k n_j - \frac{3}{2} \Phi_{jk,2} n_k n_i) f \]

where \((n_i)\) is the wall-normal unit vector in the \(i\)-direction. The wall-distance function \((f)\) represents the ratio of the turbulence length scale \((L_e = \frac{k^{3/2}}{\varepsilon})\) and the wall distance and is given as

\[ f = \left( \frac{C_m^{0.75} K^{1.5}}{\varepsilon} \right) \frac{1}{\Delta n} \]

with \(\Delta n\) being the wall-normal distance.

The resulting set of algebraic equations for the Reynolds stresses can be arranged in the form

\[ A_{ij} \overline{u^2} + B_{ij} \overline{v^2} + C_{ij} \overline{w^2} + D_{ij} \overline{u \cdot v} + E_{ij} \overline{v \cdot w} + F_{ij} \overline{u \cdot w} = G_{ij} \]

where \(A_{ij}, B_{ij}, C_{ij}, D_{ij}, E_{ij}, F_{ij},\) and \(G_{ij}\) are functions of the mean and turbulent flow variables.

The above equation can be solved iteratively in the main flow solver. However, the algebraic system of equations is stiff and convergence difficulties are encountered when solved iteratively. Therefore, the set of equations was cast in the general matrix form \(A \mathbf{T} = \mathbf{B}\), where
\[
A = \begin{bmatrix}
\frac{3\epsilon}{2\lambda k} + \frac{2}{\lambda k} \frac{\partial U}{\partial y} & \frac{\partial V}{\partial y} & \frac{\partial W}{\partial y} & \frac{2\partial U}{\partial y} \cdot \frac{\partial V}{\partial x} - 6C_0\Omega_z & \frac{2\partial U}{\partial z} \cdot \frac{\partial W}{\partial x} + 6C_0\Omega_y \\
\frac{\partial U}{\partial x} & \frac{3\epsilon}{2\lambda k} + \frac{2}{\lambda k} \frac{\partial V}{\partial y} & \frac{\partial W}{\partial z} & \frac{2\partial U}{\partial z} \cdot \frac{\partial V}{\partial y} - 6C_0\Omega_z & \frac{2\partial U}{\partial z} \cdot \frac{\partial W}{\partial x} + 6C_0\Omega_y \\
\frac{\partial V}{\partial y} & \frac{\partial W}{\partial z} & \frac{3\epsilon}{2\lambda k} + \frac{2}{\lambda k} \frac{\partial W}{\partial y} & \frac{2\partial V}{\partial z} - \frac{\partial U}{\partial y} + 6C_0\Omega_x & \frac{6C_0\Omega_x}{\partial z} \cdot \frac{\partial W}{\partial z} - \frac{\partial U}{\partial z} + 6C_0\Omega_y \\
0 & \frac{\partial W}{\partial y} + 2C_0\Omega_x & \frac{\partial V}{\partial y} + 2C_0\Omega_y & \frac{\partial W}{\partial y} + 2C_0\Omega_y & \frac{\partial V}{\partial y} + 2C_0\Omega_y \\
\frac{\partial W}{\partial x} & 2C_0\Omega_y & 0 & \frac{\partial U}{\partial z} + 2C_0\Omega_y & \frac{\partial U}{\partial z} + 2C_0\Omega_y \\
\end{bmatrix}
\]

\[
T = [\rho \bar{u} \bar{w}, \rho \bar{v} \bar{w}, \rho \bar{u} \bar{v}, \rho \bar{v} \bar{w}, \rho \bar{u} \bar{w}]^T
\]

\[
B = \begin{bmatrix}
\frac{\rho \epsilon}{\lambda} + \frac{3}{2(1-C^2)} \Phi_{11,1w} + \Phi_{11,2w} \\
\frac{\rho \epsilon}{\lambda} + \frac{3}{2(1-C^2)} \Phi_{22,1w} + \Phi_{22,2w} \\
\frac{\rho \epsilon}{\lambda} + \frac{3}{2(1-C^2)} \Phi_{33,1w} + \Phi_{33,2w} \\
\frac{1}{(1-C^2)} (\Phi_{12,1w} + \Phi_{12,2w}) \\
\frac{1}{(1-C^2)} (\Phi_{23,1w} + \Phi_{23,2w}) \\
\frac{1}{(1-C^2)} (\Phi_{13,1w} + \Phi_{13,2w}) \\
\end{bmatrix}
\]

where \( \lambda = \frac{1-C^2}{C_1} - \frac{P}{\rho \epsilon} \)

The matrix was inverted at each iteration step to obtain a converged solution.
REFERENCES


APPENDIX F

3D Algebraic Stress Module Deck

ASMOD is a FORTRAN source code to solve 2D/Axisymmetric turbulent flow quantities using the algebraic stress model when interfaced with a main flow solver. The module consists of the main routine ASMOD that calls a number of subroutines to perform different functions that will be explained below.

Subroutine ASMMOD

This is basically the main routine that reads through its argument list different variables from the calling flow solver which are described below.

List of Argument Variable Names

<table>
<thead>
<tr>
<th>INITASM</th>
<th>Initialization parameter that writes and sets variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIM</td>
<td>Number of grid nodes in the i (or x) direction</td>
</tr>
<tr>
<td>NJM</td>
<td>Number of grid nodes in the j (or y) direction</td>
</tr>
<tr>
<td>NKM</td>
<td>Number of grid nodes in the k (or z) direction</td>
</tr>
<tr>
<td>LI</td>
<td>LI(I)=(I-1)*NJ, dimensioned to NX. Calculated as in subroutine GRIDG of the 3D ( k - \varepsilon ) module.</td>
</tr>
<tr>
<td>LK</td>
<td>LK(K)=(K-1)<em>NI</em>NJ dimensioned to NZ. Calculated as in subroutine GRIDG of the 3D ( k - \varepsilon ) module.</td>
</tr>
<tr>
<td>FX</td>
<td>grid interpolation factor in the x-direction</td>
</tr>
<tr>
<td>FY</td>
<td>grid interpolation factor in the y-direction</td>
</tr>
<tr>
<td>FZ</td>
<td>grid interpolation factor in the z-direction</td>
</tr>
<tr>
<td>X</td>
<td>Grid node locations in the x or ( \xi )-direction, dimensioned to X(JXYZ=NX<em>NY</em>NZ)</td>
</tr>
<tr>
<td>Y</td>
<td>Grid node locations in the y or ( \eta )-direction, dimensioned to Y(JXYZ)</td>
</tr>
<tr>
<td>Z</td>
<td>Grid node locations in the z or ( \xi )-direction, dimensioned to Z(JXYZ)</td>
</tr>
<tr>
<td>VOL</td>
<td>Control cell volume (similar to that calculated in GRIDG of ( k - \varepsilon ) module)</td>
</tr>
<tr>
<td>U</td>
<td>mean velocity in x or ( \xi )-direction, dimensioned to U(JXYZ) (input from the flow solver)</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>V</td>
<td>Mean velocity in the y or η-direction, dimensioned yo V(JXYZ) (input from the flow solver)</td>
</tr>
<tr>
<td>W</td>
<td>Mean velocity in the z or ζ-direction, dimensioned yo W(JXYZ) (input from the flow solver)</td>
</tr>
<tr>
<td>VIS</td>
<td>Eddy viscosity</td>
</tr>
<tr>
<td>TE</td>
<td>Turbulent kinetic energy, dimensioned to TE(JXYZ) calculated in the module.</td>
</tr>
<tr>
<td>ED</td>
<td>Turbulent energy dissipation, dimensioned to ED(JXYZ)</td>
</tr>
<tr>
<td>U2</td>
<td>Normal Reynolds stress component $u^2$, calculated in the module</td>
</tr>
<tr>
<td>V2</td>
<td>Normal Reynolds stress component $v^2$, calculated in the module</td>
</tr>
<tr>
<td>W2</td>
<td>Normal Reynolds stress component $w^2$, calculated in the module</td>
</tr>
<tr>
<td>UV</td>
<td>Shear stress component $uv$, calculated in the module</td>
</tr>
<tr>
<td>VW</td>
<td>Shear stress component $vw$, calculated in the module</td>
</tr>
<tr>
<td>UW</td>
<td>Shear stress component $uw$, calculated in the module</td>
</tr>
<tr>
<td>GEN</td>
<td>Turbulent energy generation term</td>
</tr>
<tr>
<td>SUASM</td>
<td>Source term for the U-momentum equation due to Reynolds stress gradients. Calculated in the module and passed to the main solver.</td>
</tr>
<tr>
<td>SVASM</td>
<td>Source term for the V-momentum equation due to Reynolds stress gradients. Calculated in the module and passed to the main solver.</td>
</tr>
<tr>
<td>SWASM</td>
<td>Source term for the W-momentum equation due to Reynolds stress gradients. Calculated in the module and passed to the main solver.</td>
</tr>
<tr>
<td>BCFW</td>
<td>Boundary condition flag along the west boundary (or y-z plane). It must have one for each boundary node set to; 1-inlet, 2-outlet, 3-symmetry and 4-wall. For example for an outlet flow condition on the west boundary set BCFW to (NY*NZ)<em>2, and similarly for the other boundaries, dimensioned to BCFW(JYZ=NY</em>NZ) (input from flow solver)</td>
</tr>
<tr>
<td>BCFE</td>
<td>Boundary condition flag for the east boundary dimensioned to BCFE(JYZ)</td>
</tr>
<tr>
<td>BCFS</td>
<td>Boundary condition flag for the south boundary dimensioned to BCFS(JXZ)</td>
</tr>
<tr>
<td>BCFN</td>
<td>Boundary condition flag for the north boundary dimensioned to BCFN(JYZ)</td>
</tr>
<tr>
<td>BCFB</td>
<td>Boundary condition flag for the bottom boundary dimensioned to BCFB(JXY)</td>
</tr>
<tr>
<td>BCFT</td>
<td>Boundary condition flag for the top boundary dimensioned to BCFT(JXY)</td>
</tr>
<tr>
<td>GENTW</td>
<td>Turbulent generation terms calculated from the wall functions close to the wall in the west direction. Similarly for the other GENTE, GENTS....</td>
</tr>
<tr>
<td>OMX</td>
<td>Frame rotation term in the x-direction. Similarly OMY &amp; OMZ in the y and z-directions respectively</td>
</tr>
</tbody>
</table>
DENSIT  Constant density  
VISCOS  Kinematic viscosity  

All dimensions considered are one-dimensional. The position of any node is defined as IJK = (I,J,K) = (I-1)*NJ + (K-1)*NK + J, where NI, NJ and NK are the number of grid nodes in the X, Y and Z-directions respectively. It is assumed that grid related data such as control volumes and interpolation factors be passed to the module from an external grid generator, similar to the one listed in the 3D k-e module (Chapter 6).

Subroutine CALPIJ

This subroutine calculates the production terms of the individual stress components.

Subroutine CALUIUJ

This subroutine calculates the individual stress component from its algebraic equation. It sets the coefficients of the algebraic stress equations which are solved implicitly at each iteration step by inverting a 6x6 matrix.

Subroutine SORUVW

This subroutine calculates the source terms needed in the momentum equation of the main CFD solver due to Reynolds stress gradients.

Subroutine SOLV

This subroutine is a Gaussian elimination solver to invert a 6x6 matrices.

Subroutine WALSTRS

This subroutine calculates the Reynolds stresses near the walls based on wall functions.
INCLUDE 'param.h'

INTEGER LOGICAL IASM, N, N+, N++, N, N++, N, N++, N, N++, N

IF(INITASM.EQ.1) THEN
   WRITE(*,*) 'READING INPUTS TO ASM TURBULENCE MODEL'
   OPEN(42, FILE='ASM.INP', STATUS='OLD', MODE='UNKNOWN')
   READ(42,*) CIASM, C2ASM, C1P, C2P, COMEGA'
   READ(42,*) CIASM, C2ASM, C1P, C2P, COMEGA'
   IF(INITASM.EQ.1) THEN
      WRITE(*,*) 'READING INPUTS TO ASM TURBULENCE MODEL'
      OPEN(42, FILE='ASM.INP', STATUS='OLD', MODE='UNKNOWN')
      READ(42,*) CIASM, C2ASM, C1P, C2P, COMEGA'
      READ(42,*) CIASM, C2ASM, C1P, C2P, COMEGA'
   END IF
END IF

READ(1,*) IJK
DO 20 IJK=1, N
   DUDX(IJK)=0.0
   DUDY(IJK)=0.0
20 CONTINUE
END
•-_

O3
ai

E _.
E

o
O"

n

x_
000_0

o

E
(/1
gO

000_0

..._

+11

0

0

0

O0

0

_

I',-

ii_#Hll_+

_l_ll_*_Zl_l_iZll_lll_O_O_

HO_

o.

t

i

i

_

i

,it-.

61
"13
CO
I
"0
0

E
U_

m
I%

"0
iO

I
"0
0

E
iO

h.
iD

O_
O_
Oi

(J

o


C

C --SET STRESS COMPONENTS AT THE BOUNDARY EDGES.

C --WEST & EAST FACE

DO I=2,NKM

KK=(KK+1)*NJ

DO J=2,NKM

JK=KK+J

IF(BCFP(JK).EQ.4) THEN

LJ=LX(K)+LJ(J)+2

TAUMIN=2.*T(EJK)

ENDIF

VIST=VIST(JK)-VISCOS

U2(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V2(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W2(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+3

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+4

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+5

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+6

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+7

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+8

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)

ENDIF

IF(BCFP(JK).EQ.4) THEN

IJK=LX(K)+LJ(J)+9

TAUMIN=0.0

VIST=VIST(JK)-VISCOS

U(JK)=-2.*VIST+DUXD(JK)+2./3.)*DENSIT*TE(JK)

V(JK)=-2.*VIST+DVYD(JK)+2./3.)*DENSIT*TE(JK)

W(JK)=-2.*VIST+DWDJ(JK)+2./3.)*DENSIT*TE(JK)

U(JK)=VIST(DUDD(JK)+DVDD(JK))

V(JK)=VIST(DVDD(JK)+DVDD(JK))

W(JK)=VIST(DWDJ(JK)+DVDD(JK))

U2(JK)=AMAX1(U2(JK),TAUMAX)

U2(JK)=AMAX1(U2(JK),TAUMAX)

W2(JK)=AMAX1(W2(JK),TAUMAX)
ENDDO
RETURN
END
C*******************************************************************************
SUBROUTINE SOLV(A,BB,N)
C*******************************************************************************
...
CONTINUE

RETURN

--------------------------------------------------------------------------------

-- param.h

PARAMETER (JX=102)
PARAMETER (JY=45)
PARAMETER (JZ=26)
PARAMETER (JXY=JX*JY)
PARAMETER (JXZ=JX*JZ)
PARAMETER (JYZ=JY*JZ)
PARAMETER (GREAT=1.0E+30)
PARAMETER (SMALL=1.0E-30)
PARAMETER (HAF=0.5)
PARAMETER (QTR=0.25)
PARAMETER (AHT=0.125)
PARAMETER (IEF=2)

--------------------------------------------------------------------------------

-- asmmod.h

COMMON BLOCKS FOR VARIABLES INSIDE 3D ASM MODULE

COMMON/ASMCBO/
& NI, NJ, NK, NJI, NIK, NJK
COMMON/ASMCBI/
& P11(JX), P12(JX), P13(JX), P22(JX), P33(JX), P12(JY), P13(JY), P23(JY)
& P12(JZ), P23(JZ)
COMMON/ASMCB2/
& DXDX(JX), DXDY(JX), DXDZ(JX), DXDY(JX), DXDZ(JX), DXDY(JX)
& DXDX(JX), DXDY(JX), DXDZ(JX)
COMMON/ASMCB3/
& FUNX(JX), FUNY(JX), FUNZ(JX)
COMMON/ASMCBO/
& C1ASM, C2ASM, C1P, C2P, WREFON, COMEGA, RELT
During the course of this work, some related turbulence modeling work was published or presented at different meetings. Copies of these papers are listed in this chapter, they include;


This paper was presented at the advanced Earth-to-Orbit propulsion technology conference held at NASA Marshall Space Flight Center in Huntsville, Alabama on May 19-21, 1992. The work tests different correction to the standard $k$-$\varepsilon$ turbulence models that accounts for streamline curvature and rotations using different near wall treatments.


This paper was presented at the eleventh workshop for computational fluid dynamics applications in rocket propulsion held at NASA Marshal Space Flight Center in Huntsville, Alabama on April 20-22, 1993. The paper outlined the progress of the 2D/axisymmetric single and multi-scale $k$-$\varepsilon$ turbulence module deck developments.

(3) A. Hadid, M. Sindir, C. Chen and H. Wei "Computations of confined swirling flows with high order turbulence models in a modular form"

This paper was presented at the twelfth workshop for computational fluid dynamics applications in rocket propulsion held at NASA Marshal Space Flight Center in Huntsville, Alabama April-May 1994. The paper presented the status of the 2D/axisymmetric second order closure models using the algebraic and the full Reynolds stress models.

This paper presents a test for an anisotropic $k$-$\varepsilon$ turbulence model. This model is an improvement on the standard $k$-$\varepsilon$ model since it can predict Reynolds stress anisotropies without the need to solve additional equations for the stresses.


This paper tests a new one-point closure model that incorporates the effects of rotation on the power-law decay exponent of the turbulent kinetic energy. A modification to the $\varepsilon$-equation proposed by Zeman using large eddy simulation results was used. A new definition of the mean rotation was proposed based on critical point theory to generalize the effects of rotation on turbulence to arbitrary mean deformations.
COMPARATIVE STUDY OF ADVANCED TURBULENCE MODELS FOR TURBOMACHINERY

A. H. Hadid and M. M. Sindir
CFD Technology Center
Rocketdyne Division, Rockwell International
Mail Code IB39, 6633 Canoga Avenue
Canoga Park, CA 91303

ABSTRACT

Development and assessment of the standard $k-\varepsilon$ turbulence model for rotating flows with different near-wall treatments is presented. These include the standard wall function (1), Patel’s two-layer model (2), and Lam and Bremhorst’s (3) low-Reynolds number model. Two test cases were chosen to validate these models for rotating flows. The first, from Daily and Nece (4), is for a rotating disk cavity in which recirculation and secondary flows are induced by the rotating element. The second case is that of a confined double concentric jets with a sudden expansion by Roback and Johnson (5).

It is shown that near-wall effects are important close to rotating walls and that the two-layer model behaves better than the other two near-wall models. For confined swirling flows with fixed walls, the near wall effects are of secondary importance to the Reynold’s stress anisotropy.

INTRODUCTION

Accurate predictions of turbulent flows are crucial to the design and analysis of many physical and engineering applications. Increases in available computational capabilities have permitted the development and testing of sophisticated models in the numerical simulation of turbulent flows. Direct numerical simulation, where all essential scales of the turbulent flow are resolved by solving the unsteady Navier-Stokes equations, are possible only at low to moderate Reynolds numbers. Turbulent flow analysis for engineering applications, therefore, can only be achieved by utilizing the time-averaged Navier-Stokes equations coupled with some level of modelling.

The complex structure of swirling flowfields requires careful consideration of the turbulence model derivation and development. The analysis of turbulent transport and modeling evolves from the Reynolds-averaged Navier-Stokes equations and auxiliary equations for velocity and length scales for eddy viscosity specifications. Simple eddy viscosity models based on the Boussinesq hypothesis of linear relationship between turbulent shear stress and rate of strain have been quite successful in predicting a wide variety of turbulent flows.

One of the widely used models is the two-equation $k-\varepsilon$ model. The model developed originally by Launder and Spalding (1) was successful in providing good predictions for a large range of turbulent flows. The equations can be derived from the full transport equations for the Reynolds stresses assuming fully turbulent flow. Effects such as that of rotation which are included in the Reynolds stress equations are cancelled out and the resulting scalar $k-\varepsilon$ equations are invariant to system rotation.

For low-Reynolds number flows close to solid boundaries, adjustments to the model are needed to bridge the viscous dominated sublayer region with the fully turbulent flow region. The success of the wall function method depends on the universality of the turbulent structure near the wall. In many complex flows, however, the flowfield near the wall has to be determined accurately and the traditional wall-function method is not satisfactory. This is because the specification of all turbulence quantities in terms of the friction velocity fail at separation where the flow near the wall is no longer controlled by the wall shear stress. Patel et al. (6) assessed the relative performance of various models which describe the near-wall flows and found that there are still areas of improvements needed to accurately model flow behavior near the wall.

Jones and Launder (7) extended the original $k-\varepsilon$ model to the low-Reynolds number form which allowed the calculation to be performed all the way to the wall. Numerical difficulties of accurately resolving the large gradients close to the wall necessitates resolving the wall region with very fine grid structure. Chen and Patel (8) introduced a method to resolve the near-wall region which combines the standard $k-\varepsilon$ model with the one-equation model of Wolfshtein (9) near the wall. In this “two-layer” model an algebraically prescribed eddy-viscosity for the wall region is coupled to the $k-\varepsilon$ model to describe the details of the flow in the vicinity of the wall. Momentum and continuity equations are solved up to the wall and this reduces the physical uncertainties of near-wall turbulence and the numerical difficulties of resolving the vary large gradients of turbulence parameters.

The purpose of this paper is to discuss the application of the $k-\varepsilon$ turbulence model with various near-wall treatments in the prediction of confined swirling flows. These models include, the standard wall function approach (WF), Chen and Patel’s (2) two-layer model (2L), and Lam and Bremhorst’s (3) low-Reynolds number model (LB).
Evaluation of the various turbulence models was performed by comparison with two selected experimental studies. The first is that of Daily and Nece (4) where rotating disk cavity circulation and secondary flows are induced by a rotating wall. The second is that of Roback and Johnson (5) for a confined double concentric jets with a sudden expansion. Flow swirl in this case is induced by imposing a tangential velocity component at the outer jet.

Numerical predictions for turbulent flows in two-dimensional axisymmetric geometries were obtained using a finite-volume second order upwind differencing scheme on a non-staggered grid with a pressure correction method based on the SIMPLE algorithm (9). The development and evaluation of the turbulence models for rotating flows is part of an ongoing program to assess different models for rotating machinery applications. A discussion on the effects of swirl and streamline curvature on the turbulence structure through the gradient Richardson number formulation is given. Key problem areas will be identified and recommendations for the near-wall treatment as they pertain to rotating flows will be proposed.

MODEL AND EQUATION FORMULATION

Consider an incompressible, statistically steady and axisymmetric turbulent flow, the Reynolds averaged momentum and continuity equations can be expressed in a generalized form as:

$$\frac{\partial(p\Phi)}{\partial t} + \frac{\partial(pu\Phi)}{\partial x} + \frac{\partial(pv\Phi)}{\partial r} = \frac{\partial}{\partial x}(\Gamma \Phi_x \frac{\partial \Phi}{\partial x}) + \frac{1}{r} \frac{\partial}{\partial r}(r \Gamma \Phi_r \frac{\partial \Phi}{\partial r}) + S_\Phi \tag{1}$$

where \( \Phi \) is the dependent variable
\( \Phi = u, v, w \) for the axial, radial, and tangential velocities
p, \( \mu \), and \( S_\Phi \) are the fluid density, viscosity and the source terms for the variable \( \Phi \).

The source terms for the dependent variables are:

Axial direction, \( \Phi = u \), \( \Gamma \Phi_x = 2 \mu_e \), \( \Gamma \Phi_r = \mu_e \)

$$S_u = \frac{\partial}{\partial x}(\mu_e \frac{\partial u}{\partial x}) - \frac{1}{r} \frac{\partial}{\partial r}(\mu_e \frac{\partial v}{\partial r}) \tag{2}$$

Radial direction, \( \Phi = v \), \( \Gamma \Phi_x = \mu_e \), \( \Gamma \Phi_r = 2 \mu_e \)

$$S_v = \frac{\partial}{\partial x}(\mu_e \frac{\partial v}{\partial x}) - 2 \mu_e \frac{v}{r^2} + \frac{\mu_e}{r} \frac{\partial p}{\partial r} \tag{3}$$

Tangential direction, \( \Phi = w \), \( \Gamma \Phi_x = \mu_e \), \( \Gamma \Phi_r = \mu_e \)

$$S_w = \frac{\mu_e}{r} \frac{\partial w}{\partial r} - \frac{w}{r^2} \frac{\partial \mu}{\partial r} \tag{4}$$

The turbulence models in the two-equation k-\( \varepsilon \) model transport equations for the turbulent kinetic energy (k) and energy dissipation (\( \varepsilon \)) can be written in the same general form as equation (1).

Turbulent Kinetic energy equation

$$\Phi = k, \Gamma \Phi_x = \frac{\mu_k}{\sigma_k}, \text{ and } S_\Phi = G - \rho \varepsilon \tag{5}$$

Energy dissipation equation

$$\Phi = \varepsilon, \Gamma \Phi_x = \frac{\mu_\varepsilon}{\sigma_\varepsilon}, \text{ and } S_\Phi = \frac{\varepsilon}{k} (C_1 \varepsilon G - C_2 \varepsilon^2) \tag{6}$$

\( \sigma_k \) and \( \sigma_\varepsilon \) are turbulent Schmidt numbers \( G \) denotes the rate of production of the turbulent kinetic energy and is express as:

$$G = \mu_e \left[ 2 \left( \frac{\partial u}{\partial x} \right)^2 + \frac{\partial v}{\partial r} \right] + \frac{1}{r} \left( \frac{\partial u}{\partial r} \right)^2 + \frac{\partial w}{\partial x} \left( \frac{\partial w}{\partial x} - \frac{\partial w}{\partial r} \right) \tag{7}$$

\( \mu_e = \mu + \mu_t \), and the eddy viscosity is obtained from \( \mu_t = C_1 f_{\mu} \rho k^2 \varepsilon \)

C_\mu, C_1, C_2, \sigma_k and \( \sigma_\varepsilon \) are constants whose values are 0.09, 1.44, 1.92, 1.0, 1.0, respectively.
Near-Wall Treatment. A near-wall turbulent flow can be divided into two regions, the inner viscous sublayer where low turbulence Reynolds number effects are important and the velocities decrease rapidly to zero at the wall, and an outer fully turbulent region. The successful use of the $k$-$\varepsilon$ turbulence model for many complex flows depends on how accurately the flowfield near the wall is determined. Different models are used to treat this thin sublayer region, they include:

Wall function methods, the following equations are assumed to hold

$$u^* = y^*$$ for $y^* \leq 11.6$ \hspace{1cm} (8)

$$u^* = \frac{1}{k} \ln(y^*)$$ for $y^* > 11.6$ \hspace{1cm} (9)

where $u^* = \frac{u}{u_+}$, $y^* = \frac{y u_+}{v}$ and $u_+ = (\tau_w)_{1/2} / \rho$

$\tau_w$ is the wall shear stress which is estimated from

$$\tau_w = \frac{\mu u_+}{\delta}$$ for $y^* \leq 11.6$ \hspace{1cm} (10)

$$\tau_w = \frac{kC_\mu}{\rho u_+ k^{0.5}}$$ for $y^* > 11.6$ \hspace{1cm} (11)

where $u_+$ denotes the velocity component parallel to the wall, and $\delta$ is the normal distance from the wall.

In this approach, $k$ and $\varepsilon$ equations are solved with $f_\mu = f_1 = f_2 = 1$ only in the fully turbulent region beyond some distance from the wall. Boundary conditions, i.e., velocity components and turbulence parameters at that distance are specified in terms of the friction velocity ($u_+$).

In the low-Reynolds number model, the flow is resolved all the way to the wall with a very fine mesh. Many models have been proposed that are based on the $k$-$\varepsilon$ model and differ mainly in the choice of the damping functions $f_\mu, f_1$ and $f_2$ to bridge the gap between the sublayer and the fully turbulent regions. Lam and Bremhorst's model$^{(3)}$ is used in this work, where

$$f_\mu = \left[ 1 - \exp(-0.016R_y) \right]^{1/2} \left( 1 + \frac{20.5}{R_t} \right)$$

$$f_1 = 1 + \frac{0.06}{f_\mu}$$ and $f_2 = 1 - \exp(-R_t^2)$

$$R_y = \frac{k^{1/2} v}{\nu}$$ and $R_t = \frac{k^2}{\nu \varepsilon}$ are turbulent Reynolds numbers

These damping functions tend to unity with increasing distance from the wall. In order to resolve the very large gradients of turbulence parameters a fine mesh is required in the viscous sublayer which increases the computational time and numerical difficulties may be encountered.

In order to alleviate some of the problems encountered in the low-Reynolds number approach and yet accurately resolve the near-wall region, Chen and Patel$^{(2)}$ pursued the two-layer concept. In this model a simple algebraically prescribed eddy-viscosity model for the wall region is coupled to the $k$-$\varepsilon$ model for the outer flow to describe the details of the flow. Unlike the low-Reynolds number model that requires the solution of transport equations of both $k$ and $\varepsilon$ all the way to the wall, the one-equation model requires the solution of only the turbulent kinetic energy equation in the sublayer region while algebraically specifying the eddy-viscosity and energy dissipation.

$$v_t = C_\mu \frac{k^{1/2} L_\mu}{L_\varepsilon}$$ and $\varepsilon = k^{3/2}/L_\varepsilon$

The length scales $L_\mu$ and $L_\varepsilon$ contain the necessary damping effects in the near-wall region in terms of the turbulence Reynolds number $R_y$

$$L_\mu = C_1 y \left[ 1 - \exp(-R_y/A_\mu) \right]$$ \hspace{1cm} (12)

$$L_\varepsilon = C_1 y \left[ 1 - \exp(-R_y/A_\varepsilon) \right]$$ \hspace{1cm} (13)

The length scales $L_\mu$ and $L_\varepsilon$ become linear and approach $C_1 y$ with increasing distance from the wall. $C_1 = kC_\mu^{-0.75}$ with $A_\mu = 2C_\mu$, Chen and Patel$^{(2)}$ gave values for the constant $A_\mu = 70$. The damping effects decay rapidly with distance from the wall.
independent of the magnitude of the wall shear stress. The matching between the one-equation and the standard k-ε models is carried out along prescribed grid lines where \( R_y = 200 \).

**STREAMLINE CURVATURE AND SWIRL CORRECTIONS** Turbulent flows in many engineering applications such as turbomachinery and combustion devices are frequently subjected to complicating influences such as mean strain and body forces due to rotation. In such complex flows streamline curvature and swirl can exert a large influence on the structure of turbulence. Bradshaw\(^{(10)}\) reviewed the effects of streamline curvature and discussed the large effect exerted on shear-flow turbulence by curvature of streamlines in the plane of the main shear. So and Mellor\(^{(11)}\) suggested that the appropriate parameter governing this effect is \( F = \frac{uR}{\partial u/\partial y} \), where \( R \) is the radius of streamline curvature. Militzer et al.\(^{(12)}\) provided a simple generalization of this parameter for a 2-D recirculating flow as

\[
F = \frac{\left( \frac{u^2+v^2}{2} \right)^{1/2} R}{\left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)}
\]

They modified the turbulence production term \( G \) in the turbulent energy equations to include curvature effects and obtained improved predictions. Launder et al.\(^{(13)}\) proposed a simple modification to the constant \( C_2 \) in the \( \varepsilon \)-equation to account for streamline curvature due to swirl in the form

\[
C_2 = 1.0 - 0.2 R_{is}
\]

where \( R_{is} \) is a swirl Richardson number defined by

\[
R_{is} = \frac{w/r^2 \partial (w/v)/\partial r}{(\partial u/\partial r)^2 + (r \partial (w/v)/\partial r)^2}
\]

Another expression of \( R_{is} \) can also be derived as

\[
R_{is} = \frac{k^2 w}{\varepsilon \sqrt{\frac{u^2+v^2}{2}} \partial r}
\]

The basis of the above correction is that the effect of swirl on turbulence can be modelled through an increase in the length scale of the energetic turbulence eddies.

Abujela and Lilley\(^{(14)}\) used a modified \( C_2 \) form (Eq. 15) with both definitions of \( R_{is} \) from Equations (16) and (17) as applied to turbulent swirling flows. They concluded that Eq. (16) Richardson number gave better comparisons with experiment as compared to Eq. (17) Richardson's number. They also found the value of \( C_2 \) obtained from Eq. (15) had to be limited to \( 0.1 \leq C_2 \leq 2.4 \) with \( C_S \) and other constants assigned their conventional values.

Srinivasan and Mongia\(^{(15)}\) further split the Richardson number into two parts - the swirl Richardson number \( R_{is} \) and the curvature Richardson number \( R_{ic} \) and corrected \( C_2 \) in the \( \varepsilon \)-equations as:

\[
C_2 = 1.92 \exp \left( 2\alpha_s R_{is} + 2\alpha_c R_{ic} \right)
\]

where \( R_{is} \) is given by equations (16) or (17) and

\[
R_{ic} = \frac{\left( \frac{u^2+v^2}{2} \right)^{1/2} R}{\left( \frac{\partial u}{\partial r} + \frac{\partial v}{\partial x} \right)}
\]

where \( R \) is the radius of curvature given by \( R = \frac{\left( \frac{u^2+v^2}{2} \right)^{1/2}}{\frac{\partial u}{\partial r} + \frac{\partial v}{\partial x}} \), and \( \alpha_s \) and \( \alpha_c \) are constants with values ranging between 0.1 and 2.4.

Chang et al.\(^{(16)}\) investigated the streamline curvature effects in the k-ε model. They managed to obtain satisfactory results in their hybrid k-ε model where modifications of curvature effects in \( C_2 \) is made only in regions where the streamline curvature is large.

In the present study curvature and swirl modifications are made to \( C_2 \) similar to Eq. (18) of Srinivasan and Mongia with \( R_{is} \) as in Eq. (16) and \( R_{ic} \) as in Eq. (19). The exponential form ensures that \( C_2 \) will never become negative. Numerical testing with several values of \( \alpha_s \) and \( \alpha_c \) reveal that \( C_2 \) may become very large and therefore, had to be limited to \( 0.1 \leq C_2 \leq 2.4 \).
The various near-wall treatment models are analyzed by comparing model predictions with experimental data. Two cases of rotating flow experiments were selected for validation, they include; Daily and Nece\textsuperscript{(4)} for rotating disk cavity experiment and Roback and Johnson\textsuperscript{(5)} for swirling flow in a confined double concentric jets with a sudden expansion. The main criterion for selecting these cases is the different mechanisms used to generate swirling flows. In Daily and Nece experiment flow rotation is induced by the rotating wall, while in Roback and Johnson’s Experiment, swirl is imparted to the flow by an outer swirling jet into a sudden expansion. Different rotation mechanisms affecting turbulence can highlight the differences between various turbulence models and offer certain corrections that would prove useful in accurately analyzing the effects of swirl.

**CASE (1) - DAILY AND NECE\textsuperscript{(4)}** In their experimental and analytic study Daily and Nece\textsuperscript{(4)} analyzed the steady-state turbulent flow in enclosed rotating disk cavities. They characterized the existence of four flow regimes depending on the rotational Reynolds number and cavity aspect ratio. The two-dimensional axisymmetric flow considered is that of an incompressible flow bounded by a disk (rotor) and a stationary end wall (stator) of a chamber as shown in Figure 1. The ratio of the axial clearance between the rotor and the stator (s) to the radius of the disk (a) is 0.0255. The disk rotates with a rotational Reynolds number $R=4.4\times10^6$ defined as $R=\Omega a^2/v$, where $\Omega$ is the disk rotational speed in rad/sec and $v$ is the kinematic viscosity.

Numerical computations were performed on a 33x75 grid with different grid clustering near the walls for the different near-wall models. Figure 2, shows the velocity vectors at the top region of the cavity using the WF model. Centrifugal forces move the fluid radially outward on the disk, axially away from the disk on the wall casing, and radially inwards on the stationary end wall. Figure 3, shows the axial variations of the radial velocity component ($v$) at a radial position $r/a=0.765$. The agreement is fair with some discrepancy for all near-wall models close to the rotating disk. Figure 4, shows the axial variation of the tangential velocity ($w$) component at the radial position. At the rotating disk ($x=0$), the tangential velocity component approach the value ($a\Omega$). The 2L near-wall model seem to offer closer agreement with the data than the other two models.

The presence of corner regions presents a difficulty in defining the normal distances used in the definition of turbulent Reynolds number ($R_B$). In the present analysis, values of the normal distance from a wall were based on the normal distance to the nearest solid boundary. Streamline curvature and swirl corrections have not been used in this case.

**CASE (2) - ROBACK AND JOHNSON\textsuperscript{(5)}** Predictions of the experiments of Roback and Johnson\textsuperscript{(5)} have been presented by several workers, e.g. Sloan et al.\textsuperscript{(17)} and Durst and Wenerger\textsuperscript{(18)}. Unfortunately, inlet profiles were not provided in their experiment. Therefore, calculations were started at the expansion plane using the measured velocity profiles at 5mm downstream of the expansion after some adjustments near the edges of the coaxial jets. Measurements of all three main turbulent intensities were used to calculate inlet values of the turbulent kinetic energy. Energy dissipation rate was estimated from

$$\varepsilon = \frac{C_{\mu}k^{3/2}}{L}$$

where $L$ is a length scale of turbulence at the inlet of the order of $L=10^{-4}$ m.

Figure 5, shows an illustration of the test chamber geometry. The confluence plane of the primary (inner) and secondary (outer) jet streams coincides with the chamber expansion plane. The chamber diameter is about twice the secondary tube diameter. The exit from the 8-bladed, 30°, free vortex swirl generator is located approximately 0.05 m upstream from the confluence plane.

A prevalent phenomenon in axisymmetric swirling flows in such geometries is the "bubble" or vortex breakdown which has been studied extensively\textsuperscript{(19,20,21,22)}. The near axisymmetric breakdown can be partially understood from a simplified analysis of the role of pressure and centrifugal forces. It is identified by a slowly varying vortex core which undergoes an abrupt and rapid deceleration, forming a free stagnation point, followed by a region of flow reversal. It is known that the structure of vortex breakdown is unstable and asymmetric in the azimuthal direction, and displays unsteadiness in the axial direction\textsuperscript{(23,24)}. However, no periodic or nonaxisymmetric behavior attributable to the vortex breakdown was observed in Roback and Johnson’s experiment.

In the numerical simulation of the experiment, a 150x100 grid nodes was used with different clustering on the walls for the different near-wall models used. Figure 6, shows the velocity vectors indicating the presence of a closed recirculation zone at the center with additional zones at the corner downstream and between the inner jet and the outward diverted secondary jet. The figure also shows flow diversion outwards with high gradients characterized by large turbulent shear and fluctuation levels.

Comparisons were made of the radial variations of flow variables at two axial locations, $x=0.025$ m upstream of the vortex bubble and $x=0.102$ m located inside the vortex bubble. Figure 7a, shows the radial variation of the axial velocity...
profile at x=0.025 m using the WF method, 2L model and LB model. Fair agreement is predicted by the different models. They also seem to predict small negative velocities at a radial position r=0.0153 m (the interface between the inner and outer jets), slightly underpredicting its strength and width. Figure 7b, shows the radial variation of the axial velocity profiles at x=0.102 m. The 2L model shows a better agreement with the experimental data. These velocities are slightly underpredicted above the outer jet diameter.

Radial variations of the tangential velocities are shown in Figure 8a and 8b at x=0.025 m and x=0.102 m respectively. Figure 8a shows that the 2L model offers a better agreement with the experiment as compared with the WF and LB models. At x=0.102 m, Figure 8b shows that the swirl velocity is underpredicted. That is because the radial transfer of circumferential velocity is highly dependent on the turbulent diffusion mechanisms which are not accurately modelled in the isotropic eddy-viscosity k-ε model used here.

The turbulent intensity predictions for the k-ε model using the different near-wall treatments seem to follow similar trends as shown in Figures 9(a,b), 10(a,b), and 11(a,b). In general within the approximations of the isotropic k-ε model, the 2L model offer a marginal improvements over the WF and LB near wall models. The peaks in the axial, radial, and tangential turbulence intensities occur around the edges of the inner and outer jets. Figure 13a, shows the axial-azimuthal Reynolds stress profile at x=0.025 m. Figure 12b and 12c, show the axial-radial and radial-azimuthal Reynolds stress profiles at x=0.102 m.

The analysis of the main turbulent intensities and of the Reynolds stress components using the isotropic eddy-viscosity k-ε turbulent model do not reveal exclusively the advantage of one near-wall model over the other. Moreover, Reynolds shear stress profiles are sensitive to the upstream inlet conditions and the developing mean flowfield. Although the mean flow quantities show a general trend of improved predictions using the 2L near-wall model, the main effects of turbulence are due to anisotropy of Reynolds stresses especially around the highly sheared region of the outward diverted outer jet and the vortex bubble.

Streamline curvature and swirl corrections have been attempted in the present analysis with little success. Corrections of C2 using equation (18) with equations (16) and (19) for the swirl and curvature Richardson numbers. Figure 13(a,b) show the radial distribution of the radial velocity at x=0.025 m, and the axial velocity at x=0.102 m. Small improvement is detected with these corrections. The constants αc and αw used are those recommended by Srinivasan and Mungia(15) in their calculations (αc=0.75 and αw=-2.0). These constants were not optimized in the present calculations.

CONCLUSIONS

Flow predictions were performed for the standard k-ε turbulence model with different near-wall treatments to assess their performance when applied to rotating flows. Comparisons of predictions with the experimental data of Daily & Nece, and Roback & Johnson show reasonable agreement for all near-wall models and in general, the two-layer model seem to offer better comparisons compared to the wall function and Lam & Bremhorst low-Reynolds number models. From a computational perspective, the two-layer model require less computer time and relatively few grid points in the wall region than the low-Reynolds number model and is less sensitive to the location of the interface between the sublayer and the fully turbulent regions. Streamline curvature and swirl corrections show small improvements. However, further study is needed to optimize their constants.

REFERENCES

(10) Bradshaw, p. "Effects of Streamline Curvature on Turbulent Flow" AGARDograph No. 169, 1973

Fig. 1 Rotating Closed Cavity

Fig. 2 Velocity Vectors

Fig. 3 Axial Distribution of Radial Velocity (m/s), at r/a=0.765

Fig. 4 Axial Distribution of Tangential Velocity (m/s), at r/a=0.765
Fig. 5 Swirling Coaxial Jets Discharging into an Expanded Duct

Fig. 6 Velocity Vectors

Fig. 7a Radial Distribution of Axial Velocity (m/s) at z=0.025m

Fig. 7b Radial Distribution of Axial Velocity (m/s) at z=0.102m

Fig. 8a Radial Distribution of Tangential Velocity (m/s) at z=0.025m

Fig. 8b Radial Distribution of Tangential Velocity (m/s) at z=0.102m
Fig. 9a Radial Distribution of Axial Turbulence Intensity at x = 0.025m

Fig. 9b Radial Distribution of Axial Turbulence Intensity at x = 0.102m

Fig. 10a Radial Distribution of Radial Turbulence Intensity at x = 0.025m

Fig. 10b Radial Distribution of Radial Turbulence Intensity at x = 0.102m

Fig. 11a Radial Distribution of Tangential Turbulence Intensity at x = 0.025m

Fig. 11b Radial Distribution of Tangential Turbulence Intensity at x = 0.102m
Fig. 12a Radial Distribution of Axial-Azimuthal Reynolds Stress at x = 0.025m

Fig. 12b Radial Distribution of Axial-Radial Reynolds Stress at x = 0.102m

Fig. 12c Radial Distribution of Radial-Azimuthal Reynolds Stress at x = 0.102m

Fig. 13a Radial Distribution of Radial Velocity at x = 0.025m

Fig. 13b Radial Distribution of Axial Velocity at x = 0.102m
ADVANCED TURBULENCE MODELS FOR TURBOMACHINERY

Ali H. Hadid, Michele E. DeCroix, and Munir M. Sindir
Rocketdyne Division, Rockwell International

ABSTRACT

Development and assessment of the single-time-scale $k$-$\varepsilon$ turbulence model with different near-wall treatments and the multi-scale $k$-$\varepsilon$ turbulence model for rotating flows are presented. These turbulence models are coded as self-contained module decks that can be interfaced with a number of CFD main flow solvers. For each model, a stand-alone module deck with its own formulation, discretization scheme, solver and boundary condition implementations is presented. These satellite decks will take as input (from a main flow solver) the velocity field, grid, boundary condition specifications and will deliver turbulent quantities as output. These modules were tested as separate entities and although many logical and programming problems were overcome only wider use and further testing can render the modules sufficiently "fool proof".
DEVELOPMENT OF A MODULAR FORMAT FOR GENERAL USE TURBULENCE MODELS

Ali H. Hadid
Michele E. DeCroix

Rockwell International, Rocketdyne Division

Workshop for Computational Fluid Dynamic Applications in Rocket Propulsion

April 20-22, 1993
NASA Marshall Space Flight Center
STRUCTURE OF FUTURE CODES LEADS TO DEVELOPMENT OF MODULES

- FUTURE CODES COMPOSED AT TIME OF EXECUTION
  - INTELLIGENT DRIVERS
  - MODULES FOR PHYSICAL MODELS

- INTEGRAL PRE- AND POSTPROCESSING TOOLS

- COMMON DATA BASE
TURBULENCE MODELS TO BE ASSESSED

PHENOMENOLOGICAL SINGLE POINT CLOSURE MODELS

SINGLE-SCALE

MULTI-SCALE

2-EQUATION MODELS
K-ε (MKEM)

2-EQUATION MODELS
K-ε (SKEM)

ALGEBRAIC
STRESS MODELS
(ASM)

REYNOLDS
STRESS MODELS
(RSM)

- THE 2-D/AXISYMMETRIC SINGLE-SCALE K-ε MODULE DECK (KEMOD-1) AND THE
  2-D/AXISYMMETRIC MULTI-SCALE K-ε MODULE DECK (KEMOD-2) ARE COMPLETE

- NEAR-WALL TREATMENTS WILL INCLUDE (WHERE APPROPRIATE) WALL FUNCTIONS,
  MULTILAYER MODELS, AND LOW-REYNOLDS NUMBER APPROXIMATIONS
TURBULENCE MODEL DECK STRUCTURE AND INTEGRATION WITH NAVIER-STOKES SOLVER

- Modules are based on the phenomenological single point turbulence closure models.
- They are structured basically to accept as input the mean flow velocities from a N-S solver and to return turbulence quantities to the solver.

---

**Preprocessor**
1. Grid
2. Boundary condition flags
3. Flow properties
4. Initial conditions

**Navier-Stokes Solver**

**Input to Turbulence Deck**
Mean velocity \( U_i \)

**Output from Turbulence Deck**

**Iteration Loop**

---

**Turbulence Model Deck**

**Self-contained deck with built-in solver**

**2 levels of modeling**

- Eddy viscosity models
  1. SKEM
  2. MKEM

- Reynolds stress models
  1. ASM
  2. RSM

---

**User Provided**

**Rocketdyne Provided**
SINGLE-SCALE K-ε MODEL

GENERALIZED TRANSPORT EQUATION IN 2-D/AXISYMMETRIC GEOMETRY

\[ \frac{\partial (\rho u)}{\partial x} + \frac{1}{r} \frac{\partial (\rho u u)}{\partial r} = \frac{\partial}{\partial x} \left( \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial r} \left( \frac{\partial \phi}{\partial r} \right) + S_\phi \]

\[ \phi = u, \quad \Gamma_{\phi_x} = 2\mu e, \quad \Gamma_{\phi_r} = \mu e \]

\[ S_u = -\frac{\partial}{\partial x} \left( \frac{\rho}{r} \frac{\partial u}{\partial r} \right) + \mu e \frac{\partial^2 u}{\partial r^2} + \frac{\mu e}{r^2} \frac{\partial u}{\partial r} \]

U-MOMENTUM

\[ v = v, \quad \Gamma_{\phi_x} = \mu e, \quad \Gamma_{\phi_r} = 2\mu e \]

\[ S_v = -\frac{\partial}{\partial x} \left( \frac{\rho}{r} \frac{\partial v}{\partial r} \right) + \mu e \frac{\partial^2 v}{\partial r^2} - 2\mu e \frac{r}{r^2} \frac{v^2}{r} \]

V-MOMENTUM
SINGLE-SCALE k-ε MODEL (CONT'D)

- W-MOMENTUM
  \[ \phi = w, \quad \Gamma \phi_x = \mu e, \quad \Gamma \phi_r = \mu e \]
  \[ S_w = -\frac{\rho vw}{r} + \frac{w}{r^2} \frac{\partial}{\partial r} (r\mu e) \]

- TURB. KINETIC ENERGY
  \[ \phi = K, \quad \Gamma \phi_x = \mu + \frac{\mu t}{\sigma_k} = \Gamma \phi_r \]
  \[ S_\phi = G - \rho \varepsilon \]

- TURB. ENERGY DISSIPATION
  \[ \phi = \varepsilon, \quad \Gamma \phi_x = \mu + \frac{\mu t}{\sigma_k} = \Gamma \phi_r \]
  \[ S_\phi = \frac{\varepsilon}{k} (c_1 f_1 G - c_2 f_2 \rho \varepsilon) \]
SINGLE-SCALE $k-\varepsilon$ MODEL (CONT'D)

\[ G = \mu_e \left\{ 2 \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial r} \right)^2 + \left( \frac{\partial v}{\partial \theta} \right)^2 \right] + \left( \frac{\partial u}{\partial r} + \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial x} \right)^2 + \left( \frac{\partial w}{\partial r} - \frac{w}{r} \right)^2 \right\} \]

\[ \mu_t = C_\mu f_\mu \rho \frac{k^2}{\varepsilon} \]

\[ \mu_e = \mu + \mu_t \]

$C_\mu = 0.09$, $c_1 = 1.44$, $c_2 = 1.92$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.0$
KEMOD-1 MODULE DECK

- KEMOD-1 IS THE SINGLE-SCALE K-\( \varepsilon \) TURBULENCE MODULE DECK. IT CONSISTS OF TWO SEPARATE Routines KEMOD AND MODIFY WHICH HAVE TO BE LINKED TO THE MAIN FLOW SOLVER

- KEMOD IS CALLED WITHIN THE ITERATION LOOP OF THE MAIN FLOW SOLVER

- THE MEAN VELOCITIES AND OTHER VARIABLES ARE PASSED TO THE MODULE THROUGH ITS ARGUMENT LIST (EXPLAINED IN THE USER'S MANUAL)

- A NONSTAGGERED BODY FITTED GRID ARRANGEMENT IS USED BY THE MODULE. IT USES THE MEAN FLOW VARIABLES (VELOCITIES AND MASS FLUXES) TO CONSTRUCT THE DISCRETIZED ALGEBRAIC EQUATION

- DISCRETIZED ALGEBRAIC EQUATIONS ARE SOLVED BY STONE'S STRONGLY IMPlicit SOLVER
KEMOD-1 MODULE DECK (CONT'D)

- **SUBROUTINE GRIDG**

  READS GRID NODE LOCATIONS PASSED FROM MAIN SOLVER AND FOR THE FIRST ITERATION CALCULATES GRID RELATED QUANTITIES (CELL AREAS AND VOLUME, NORMAL DISTANCES FROM SOLID BOUNDARIES AND INTERPOLATION FACTORS)

- **SUBROUTINE CALCE**

  ASSEMBLES THE COEFFICIENTS AND SOURCE TERMS FOR THE DISCRETIZED K AND ε TRANSPORT EQUATIONS IN THE FORM

  \[ A_p \phi_p = \sum_{i = E, W, N, S} A_i \phi_i + S\phi \]

  THE SUBROUTINE SOLVES THE DISCRETIZED EQUATIONS AFTER MODIFYING THE SOURCES AND BOUNDARY CONDITIONS FOR THE PARTICULAR PROBLEM
KEMOD-1 MODULE DECK (CONT'D)

- SUBROUTINE TWOLAY
  CALLED IF THE TWO-LAYER OR THE LOW-REYNOLDS NUMBER MODELS ARE USED. IT CALCULATES THE COEFFICIENTS NEEDED TO DESCRIBE THE ENERGY DISSIPATION AND EDDY VISCOSITIES CLOSE TO A WALL

- SUBROUTINE SOLSIP
  SOLVES THE SYSTEM OF ALGEBRAIC K AND \( \varepsilon \) EQUATIONS USING STONE'S STRONGLY IMPLICIT METHODS

- SUBROUTINE USERM
  THIS SUBROUTINE HAS DIFFERENT ENTRY SECTIONS WHERE VARIABLES ARE UPDATED AND BOUNDARY CONDITIONS ARE SET

- SUBROUTINE MODIFY
  THIS IS THE ONLY SUBROUTINE THAT HAS TO BE CALLED FROM THE MOMENTUM EQUATION SOLVER OF THE MAIN ROUTINE. IT UPDATES THE FLUX SOURCE TERM OF THE DISCRETIZED MOMENTUM EQUATION DUE TO WALL SHEAR STRESSES
KEMOD-1 MODULE DECK (CONT'D)

START

READ GEOMETRY DATA
GRID NODES

CALL GRIDG CALCULATE
Fx, Fy, ARE, VOL

CALL CALCE (TE, 1)
(TURB KINETIC ENERGY K)

IF(LAY2 = OR ● LRE)

YES

MODPHI
FOR IDIR = 1

SOLSIP

NO

CALL CALCE (ED, 2)
ENERGY DISSIPATION
EQUATION (c)

MMDPHI
FOR IDIR = 2

SOLSIP

MODVIS

END

KEMOD FLOW CHART

Rockwell International
Rocketdyne Division

CFD 93 013-012/D1/AHH

1760
KEMOD-1 INTERFACE WITH A MAIN SOLVER
MULTI-SCALE $k-\varepsilon$ MODEL

- DERIVED BY PARTITIONING THE TURBULENT ENERGY SPECTRUM INTO TWO SETS OF WAVE NUMBER REGIONS (PRODUCTION AND DISSIPATION RANGES) GIVING TWO EVOLUTION EQUATIONS FOR EACH REGION

- PARTITION LOCATION IS DETERMINED AS PART OF THE SOLUTION

- TURBULENT KINETIC ENERGY IN THE PRODUCTION RANGE OF THE SPECTRUM

\[ \phi = k_p, \Gamma_{\phi x} = \Gamma_{\phi r} = \mu + \frac{\mu_t}{\sigma_{k_p}} \]

\[ S_{k_p} = G - \rho \varepsilon_p \]
MULTI-SCALE k-ε MODEL (CONT'D)

- ENERGY TRANSFER RATE IN THE PRODUCTION RANGE OF THE SPECTRUM

\[ \phi = \varepsilon_p, \quad \Gamma_{\phi X} = \Gamma_{\phi r} = \frac{\mu_t}{\sigma_{k_p}} \]

\[ S_{\varepsilon_p} = \frac{1}{\rho} C_{p1} \frac{G^2}{K_p} + C_{p2} \frac{G \varepsilon_p}{K_p} - \rho C_{p3} \frac{\varepsilon_p^2}{K_p} \]

- TURBULENT KINETIC ENERGY IN THE DISSIPATION RANGE OF THE SPECTRUM

\[ \phi = k_t, \quad \Gamma_{\phi X} = \Gamma_{\phi r} = \frac{\mu_t}{\sigma_{k_t}} \]

\[ S_{k_t} = \rho \varepsilon_p - \rho \varepsilon_t \]
MULTI-SCALE $k$-$\varepsilon$ MODEL (CONT'D)

- ENERGY DISSIPATION RATE IN THE DISSIPATION RANGE

\[
\phi = \varepsilon_t, \quad \Gamma \phi_x = \Gamma \phi_r = \mu + \frac{\mu_t}{\sigma_{\varepsilon_t}}
\]

and

\[
S_{\varepsilon_t} = \rho C_{t1} \frac{\varepsilon_p^2}{k_t} + \rho C_{t2} \frac{\varepsilon_t \varepsilon_p}{k_t} - \rho C_{t3} \frac{\varepsilon_t^2}{k_t}
\]

- MODEL IS SIMILAR TO THAT USED BY KIM AND CHEN WITH CONSTANTS

\[
\sigma_{kp} = 0.75, \quad \sigma_{\varepsilon_p} = 1.15, \quad \sigma_{kt} = 0.75, \quad \sigma_{\varepsilon_t} = 1.15
\]

\[
C_{p1} = 0.21, \quad C_{p2} = 1.24, \quad C_{p3} = 1.84, \quad C_{t1} = 0.29
\]

\[
C_{t2} = 1.28, \quad C_{t3} = 1.66 \text{ and } C_{\mu} = 0.09
\]
KEMOD-2 MODULE DECK

• KEMOD-2 is a multi-time scale K-ε turbulence module deck. It consists of two main routines KEMOD and MODIFY.

• KEMOD is called within the iteration loop of the main flow solver.

• Mean velocities and other variables are passed to the module through its argument list (explained in the user's manual).

• The module is structured in a similar way to KEMOD-1 and subroutine CALCE assembles the coefficients and source terms for the discretized K, ε, K_t, ε_t transport equations.
KEMOD-2 MODULE DECK (CONT'D)

START

READ GEOMETRY DATA GRID NODES

CALL GRIDG CALCULATE $F_x, F_y, A_R, V_C$

CALL CALCKE (TE, 1) (TURB KINETIC ENERGY $K_t$)

CALL CALCKE (TET, 2) (TURB KINETIC ENERGY $K_t$)

CALL CALCKE (ED, 3) (ENERGY TRANSFER RATE $\epsilon_p$)

CALL CALCKE (EDT, 4) (ENERGY DISSIPATION RATE $\epsilon_t$)

MODVIS

END

KEMOD-2 FLOW CHART
START

GEOMETRY AND INITIAL CONDITIONS

ITER = 0
ITERATION COUNTER

ITER = ITER + 1

U, V MAIN FLOW SOLVER

CALL OTHER ROUTINES FOR W-EQUATION, PRESSURE-CORRECTION OR OTHERS

CALL KEMOD TO CALCULATE C^ and C^3

CHECK MAXIMUM RESIDUAL ERROR OF EQUATIONS

IF ITER + GE + MAXIMUM ITERATION OR RESIDUAL ERROR IS SMALLER THAN PRESET VALUE

NO

YES

OUTPUT

KEMOD-2 INTERFACE WITH MAIN SOLVER
ROBACK AND JOHNSON – SWIRLING COAXIAL JETS DISCHARGING INTO AN EXPANDED DUCT


GEOMETRY

---

Rockwell International
Rocketdyne Division

CFD 92-032-0019/03/1998
ROBACK AND JOHNSON RESULTS

VELOCITY VECTORS

SINGLE-SCALE K-ε MODEL

MULTI-SCALE K-ε MODEL
ROBACK AND JOHNSON RESULTS
TANGENTIAL VELOCITY PREDICTIONS AT X = 0.025 M

DATA
- WALL FUNCTION
- LOW-REYNOLDS NO. MODEL
- 2-LAYER MODEL

DATA
- SINGLE-SCALE K-\( \varepsilon \) MODEL
- MULTI-TIME SCALE K-\( \varepsilon \) MODEL

LEGEND
- Roback et al.
  - Single scale k-\( \varepsilon \)
  - Multi-scale k-\( \varepsilon \)

SINGLE-SCALE K-\( \varepsilon \)

Rockwell International
Rocketdyne Division

CFD 93 013 021/D1/AHI
SUMMARY

- KEMOD-1 (2-D)
  - SINGLE SCALE k-ε TURBULENCE MODULE COMPLETE
  - TESTED USING REACT AND USA CODES

- KEMOD-2 (2-D)
  - MULTISCALE k-ε TURBULENCE MODULE COMPLETE
  - TESTED USING REACT CODE

- DEVELOPMENT OF MODULES FOR FULL AND ALGEBRAIC REYNOLDS STRESS MODELS IN PROGRESS

- WORK ON 3-D MODULES TO BEGIN AS SCHEDULED (FY '94)
COMPUTATIONS OF CONFINED SWIRLING FLOWS WITH HIGH ORDER TURBULENCE MODELS IN A MODULAR FORM

A. Hadid\(^1\), M. Sindir\(^1\), C. Chen\(^2\), and H. Wei\(^2\)

\(^1\) CFD Technology Center
Rocketdyne Division/Rockwell International
Mail Code IB39, 6633 Canoga Avenue
Canoga Park, CA 91303

\(^2\) Department of Chemical Engineering
University of Alabama at Huntsville
Huntsville, AL 35899

Abstract

A finite-volume procedure is used to compare the performance of different high order turbulence models for confined swirling turbulent flows. Eddy-viscosity single and multi-scale \(k\)-\(\varepsilon\) turbulence models together with second-moment algebraic and Reynolds stress models are tested for a two-dimensional, axisymmetric swirling flow case. The ability of second-moment closure models to capture the interaction between swirl and the turbulent stress field is crucial to the predictive performance of the computational scheme.

To enhance the predictive capability of CFD tools for engineering applications, advanced turbulence models are coded as self-contained module decks that can be interfaced with a number of CFD solvers. Three of these modules, namely the single and the multi-scale models and the Algebraic stress model (ASM) have been successfully interfaced and tested with the code MAST of the University of Alabama at Huntsville in a relatively short time. These modules are independently tested and evaluated with the data of Roback and Johnson for swirling turbulent flow in a confined double concentric jets with a sudden expansion.

Modularization of a general purpose CFD code structure in terms of different aspects of physical models is necessary for computational efficiency. Further, individual modular routines are transportable and can be easily modified to include extra physical effects. This would allow many users using different CFD codes to concentrate their talents on developing and improving physical hypothesis for specific engineering problems.

Introduction

Computational Fluid Dynamics (CFD) has been used extensively for the last decade or so in analyzing complex flow phenomena for many industrial applications, such as, turbomachinery and combustion devices. Most flows of technological interest are turbulent and for many of them, relatively simple prediction methods are sufficient to produce results of engineering accuracy. For others, mainly in high technology applications, accurate predictions using high order turbulence models are required. Increases in available computational capabilities have permitted the development and testing of sophisticated models in the numerical simulation of turbulent flows. Direct numerical simulation, where all essential scales of the turbulent flow are resolved by solving the unsteady Navier-Stokes equations, is possible only at low to moderate Reynolds numbers. Turbulent flow analysis for engineering applications, therefore, can only be achieved by utilizing the time-averaged Navier-Stokes equations coupled with some level of modelling. The analysis of turbulent transport and modelling evolves from the Reynolds-averaged Navier-Stokes equations and auxiliary equations for velocity and length scales for eddy viscosity specifications towards a more sophisticated modeling strategy - one offering greater width of applicability, particularly in complex shear flows or where external force fields modify the turbulence structure.

One of the widely used models is the two-equation single-time-scale \(k\)-\(\varepsilon\) model\(^{(1)}\). In this model transport equations for the turbulence energy \(k\) and the energy dissipation \(\varepsilon\) are solved to determine the turbulent eddy viscosity. An improvement to the single scale \(k\)-\(\varepsilon\) model is the multi-time-scale \(k\)-\(\varepsilon\) model where the energy spectrum of a turbulent flow is split into a production range and a dissipation range\(^{(2)}\). Improved predictions using the multi-scale over the single-scale \(k\)-\(\varepsilon\) model have been demonstrated\(^{(3,4)}\).

Other complicated single-scale models offering greater width of applicability, particularly in complex shear flows or where external force fields modify the turbulence structure are based on second-moment closures. These take the exact equations for the transport of the Reynolds stresses \((\overline{u_iu_j})\) as their starting point and devise approximations for the
unknown turbulent correlations appearing in them. In a three-dimensional flow, or even in an axisymmetric flow, all six components of the Reynolds stress tensor are nonzero. With a full second-moment closure model (RSM'), therefore, differential transport equations need to be solved over the solution domain for each of these components. This represents an increase in the task of numerical solution compared with the situation where the k-ε eddy-viscosity model is adopted. An intermediate level of modeling has evolved known as Algebraic second-moment closure (ASM), with the aim of retaining the greater physical realism of second-moment treatments while achieving computational times closer to that of an eddy-viscosity model. The simplification is achieved by approximating the convective and diffusive transport of the Reynolds stresses in terms of the corresponding transport of turbulent energy. This allows the transport equations for the stresses to be expressed as a set of algebraic formulae containing the turbulence energy and its rate of dissipation as unknowns. Second moment schemes have been extensively and successfully applied to a wide range of flows, as reviewed for example by Leschiziner. Few applications, however, have considered axisymmetric swirling flows where the external forces due to swirl exert damping effects on the turbulent transport.

Progress in turbulence modelling have been paralleled by improvements in numerical techniques, essentially, combining second moment closure with non-orthogonal, co-located grids using finite-volume methods. However, the implementation of RSM into non-orthogonal finite-volume codes poses difficulties: the co-located variable arrangement can cause decoupling of the mean velocity and Reynolds stress fields leading to oscillating solutions or even divergence. Using a special interpolation procedure in the context of Rhie, Obi and Peric calculated the two-dimensional turbulent flow on a co-located grid arrangement using the Reynolds stress turbulence model.

In the present paper, we present predictions of two dimensional/axisymmetric swirling flow using various models based on eddy-viscosity single and multi-scale k-ε and on second moment closure. These models are cast in a modular form enabling them to be used with a number of flow solvers based on the finite-volume and finite-difference methods. A discussion of the different models used and their assessment is presented. The modular structure of the different turbulence models will also be presented and discussed.

**Theory and Model Equations**

The turbulent flow considered is two-dimensional and steady which can be described by the

Reynolds averaged continuity and momentum equations which may, respectively, be written as

\[ \frac{\partial \bar{u}}{\partial x} + \frac{1}{r} \frac{\partial \bar{v}}{\partial r} = 0 \]  \hspace{1cm} (1)

\[ \frac{\partial \bar{u} \phi}{\partial x} + \frac{\partial \bar{v} \phi}{\partial r} = \frac{1}{r} \left( \frac{\partial (\mu \frac{\partial \phi}{\partial x})}{\partial x} + \frac{\partial (\mu \frac{\partial \phi}{\partial r})}{\partial r} \right) + r \Phi \]  \hspace{1cm} (2)

Where \( \Phi \) stands for any of the momentum components \( U, V, \) and \( rW \) and the corresponding sources \( S_{\Phi} \) are

\[ S_{U} = \frac{\partial \bar{P}}{\partial x} \frac{\partial \bar{u}^{2}}{\partial x} - \frac{1}{r} \frac{\partial \bar{u} \bar{v}}{\partial r} \]

\[ S_{V} = \frac{\partial \bar{P}}{\partial r} \frac{\partial \bar{v}^{2}}{\partial r} - \frac{2 \mu}{r^{2}} \frac{\partial \bar{v}^{2}}{\partial r} + \frac{\partial \bar{u} \bar{v}}{\partial r} + \frac{\bar{r} \bar{w}^{2}}{r} \]

\[ S_{W} = - \frac{2 \mu}{r} \frac{\partial \bar{w}}{\partial r} + \frac{\partial (\mu \bar{w} \bar{w})}{\partial x} - \frac{\partial (\mu \bar{v} \bar{w})}{\partial x} - 2 \frac{\partial \bar{w} \bar{v}}{\partial x} \]

where \( \rho, \mu \) are the fluid density and viscosity respectively.

The appearance of the Reynolds stresses \( \bar{u}_{ij} \) represents an unknown correlation and different turbulence models provide the means of relating these unknowns to known determinable quantities.

**Single-Scale Eddy-Viscosity Turbulence Models**

Here it is assumed that a single-time-scale (proportional to \( k/\varepsilon \)) can be used to describe the turbulent flow. Turbulence is simulated through transport equations for the turbulent kinetic energy (\( k \)), and its rate of dissipation (\( \varepsilon \)). The stress tensor is modelled using a gradient transport model of the form

\[ \bar{u}_{ij} = \nu_{t} \left( \frac{\partial U_{i}}{\partial x_{j}} + \frac{\partial U_{j}}{\partial x_{i}} \right) - \frac{2}{3} k \delta_{ij} \]  \hspace{1cm} (3)

The general form of the two-equation eddy-viscosity turbulence model can be written as

Kinetic Energy (\( k \)) equation:

\[ C_{k} = D_{k} + P - \varepsilon \]  \hspace{1cm} (4)

where

\[ C_{k} = \frac{\partial U_{i} \frac{k}{\partial x_{j}}}{\partial x_{j}} \quad \text{Convection of } k \]

\[ D_{k} = \frac{\partial}{\partial x_{j}} \left( \nu_{t} + \frac{\nu_{t}}{\sigma_{k}} \frac{k}{\partial x_{j}} \right) \quad \text{Diffusion of } k \]

\[ P = -2 \frac{\partial U_{i}}{\partial x_{j}} = -\nu_{t} \left( \frac{\partial U_{i}}{\partial x_{j}} + \frac{\partial U_{j}}{\partial x_{i}} \right) \quad \text{Production of } k \]
v_t = eddy viscosity = C_μ \frac{k^2}{ε}

Energy Dissipation (ε) equation:

C_ε = D_ε + \frac{ε}{k} (C_ε1P - C_ε2ε) \tag{5}

where

C_ε = \frac{∂U_{ij}}{∂x_j} \quad \text{Convection of ε}

D_ε = \frac{∂}{∂x_j} \left[ (v_i + v_j) \frac{ε}{σ_ε} \frac{∂ε}{∂x_j} \right] \quad \text{Diffusion of ε}

In the present study, the standard two-equation model was used with the wall function\(^1\) and the two-layer model\(^11\) to bridge the gap between the near-wall log-layer region and the fully turbulent region away from the wall. In the standard model the numerical values of the constants are C_μ=0.09, C_ε1=1.44, C_ε2=1.92, σ_ε=1.0 and σ_ε=1.3. Details of the implementation of the wall function and the two-layer models can be found in Hadid and Sindir\(^12\).

Multi-Time-Scale k-ε Turbulence Model

The Multi-time-scale turbulence model used here is based on the variable energy partitioning of the turbulent energy spectrum proposed by Kim and Chen\(^3\). In this model the turbulent kinetic energy spectrum is divided into two sets of wave number regions giving two evolution equations for each region. These equations represent the kinetic energy (k_p) and the energy dissipation (ε_p) in the production range of the spectrum and the kinetic energy (k_t) and the energy dissipation (ε_t) in the dissipation range of the spectrum. This model allows the partition to move toward the high wave number region when production is high and toward the low wave number region when production vanishes.

The equations for the turbulent kinetic energy (k_p) and the energy transfer rate (ε_p) for the production range are

C_{kp} = D_{kp} + P - ε_p \tag{6}

C_{ep} = D_{ep} + \frac{P}{ρ k_p} (\frac{1}{C_p1P + C_p2ε_p} \cdot \frac{ε_p^2}{k_p}) \tag{7}

The equations for the turbulent kinetic energy (k_t) and the dissipation rate (ε_t) for the high wave number transfer region are

C_{kt} = D_{kt} + ε_p \cdot ε_t \tag{8}

C_{et} = D_{et} + \frac{ε_p}{k_t} (C_ε1ε_p + C_ε2ε_t) - C_ε3 \frac{ε_t^2}{k_t} \tag{9}

where

C_{kp} = ρ \frac{∂U_{ij}k_p}{∂x_j} \quad \text{and} \quad C_{ep} = ρ \frac{∂U_{ij}ε_p}{∂x_j}

D_{kp} = \frac{∂}{∂x_j} \left[ (μ + \frac{μ_t}{C_p}) \frac{∂k_p}{∂x_j} \right] \quad \text{and}

D_{ep} = \frac{∂}{∂x_j} \left[ (μ + \frac{μ_t}{C_p}) \frac{∂ε_p}{∂x_j} \right]

Similarly for C_{kt}, C_{et}, D_{kt}, and D_{et} equations and the model constants used are those of Kim and Chen\(^3\).

The terms \( \frac{1}{ρ} C_p1 \frac{P^2}{k_p} \) and \( ρ C_t1 \frac{ε_p^2}{k_t} \) represent variable energy transfer functions. The former increases the energy transfer rate when production is high and the latter increases the dissipation rate when the energy transfer rate is high. The turbulent viscosity µ_t is given as µ_t = pC_μk^2/ε_t = ρC_μk^2/ε_t, where k=k_p+k_t is the total turbulent kinetic energy.

Second Moment Closure Models

The exact form of the Reynolds stress equations can be derived from the time-averaged form of the Navier-Stokes equations and can be written as:

\[ \frac{D}{Dt} (ρu_{ij}) = P_{ij} + φ_{ij} + D_{ij} + ρε_{ij} \tag{10} \]

where

\[ P_{ij} = -ρ \left( \frac{∂u_i}{∂x_k}u_j - \frac{∂u_j}{∂x_k}u_i \right) \quad \text{Production} \]

\[ D_{ij} = \frac{∂}{∂x_k} \left( -ρ u_{ijk} - δ_{ik} u_{ij} - δ_{jk} u_{ij} \right) \quad \text{Diffusion} \]

\[ φ_{ij} = ρ \left( \frac{∂u_i}{∂x_j} + \frac{∂u_j}{∂x_i} \right) \quad \text{Pressure-strain redistribution} \]

\[ ε_{ij} = \frac{2}{3} δ_{ij} ε \quad \text{Dissipation} \]

Due to the introduction of correlations of higher orders, modelling of these terms is required to close the set of equations.
Algebraic Stress Model (ASM)

The ASM model used is based on the work of Rodi (5). The idea is to simplify the stress equation (eq. 10) by approximating the convective and diffusive transport of the Reynolds stresses ($\overline{uu}$j) in terms of the corresponding transport of turbulent energy. This simplification allows the transport equation of the stresses to be expressed as a set of algebraic formulae containing the turbulent energy and its rate of dissipation as unknowns. This set of algebraic equations can be written as:

$$\overline{uu}j = \frac{k}{p} \left( \Phi_{ij} - \frac{2}{3} \delta_{ij} \sigma + \Phi_{ij} \right)$$  (11)

The pressure-strain term $\Phi_{ij}$ is decomposed into a fluctuating part ($\Phi_{ij,1}$), a part due to the mean rate of strain ($\Phi_{ij,2}$), and a part due to reflected wall-influence ($\Phi_{ij,w}$), i.e., $\Phi_{ij} = \Phi_{ij,1} + \Phi_{ij,2} + \Phi_{ij,w}$

Rotta's return to isotropy concept is used to model the non-linear part ($\Phi_{ij,1}$) as

$$\Phi_{ij,1} = -C_{1k} \frac{\overline{u_i u_j}}{k} \left( \overline{u_i u_j} - \frac{2}{3} \delta_{ij} \sigma \right)$$

$\Phi_{ij,2}$ is modelled using the isotropization of production concept as

$$\Phi_{ij,2} = -C_2 \left( \Phi_{ij} - \frac{2}{3} \delta_{ij} \sigma \right)$$

The wall reflection term $\Phi_{ij,w}$ is modelled following Shih (13) and Gibson and Launder (14) as

$$\Phi_{ij,w} = \Phi_{ij,1w} + \Phi_{ij,2w}$$

where

$$\Phi_{ij,1w} = C_1 \frac{\overline{u_i u_m n_k u_n \delta_{ij} - \frac{3}{2} \overline{u_i u_j n_k u_n} \frac{3}{2} \overline{u_i u_j n_k u_n} \frac{1}{2} \delta_{ij} \sigma} \right) f$$  (12)

$$\Phi_{ij,2w} = C_2 \left( \Phi_{km,2} \overline{u_k u_m \delta_{ij}} - \frac{3}{2} \overline{u_i u_j n_k u_n} \frac{3}{2} \overline{u_i u_j n_k u_n} \frac{1}{2} \delta_{ij} \sigma \right) f$$  (13)

where $n_i$ is the wall-normal unit vector in the $i$-direction. The wall distance function ($f$) represents the ratio of turbulence length scale and the wall distance

$$C_m \left( \frac{k^{1.5}}{\epsilon} \right) \frac{1}{\Delta n}$$

where $\Delta n$ is the wall-normal distance. The above wall-correction terms are written in a tensorially invariant form and their effect is to transfer energy from the wall-normal normal stress component to the tangential stresses i.e it is redistributive.

For axisymmetric swirling flows the set of algebraic stress equations can be written in a general matrix form as $\mathbf{A} \mathbf{T} = \mathbf{B}$ where

$$\begin{bmatrix}
\frac{3e}{2k} + \frac{2}{\partial x} \frac{\partial U}{\partial x} \\
\frac{3e}{2k} + \frac{2}{\partial y} \frac{\partial V}{\partial y} \\
\frac{3e}{2k} + \frac{2}{\partial z} \frac{\partial W}{\partial z}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial U}{\partial x} \\
\frac{\partial V}{\partial y} \\
\frac{\partial W}{\partial z}
\end{bmatrix}
$$

Reynolds Stress Model (RSM)

In the RSM model the full transport equation for the Reynolds stresses (eq. 10) are solved for each stress component ($\overline{uu}j$) after modelling the diffusion and the pressure strain terms similar to Launder et al. (15). The diffusion term is modelled as

$$\mathbf{D}_{ij} = \frac{\partial}{\partial x} \left[ \rho \frac{C_k}{\epsilon} \frac{\partial u_i}{\partial x} \right]$$

The pressure-strain redistribution term $\Phi_{ij}$ is modelled in a similar way to that used in the ASM model discussed earlier. Special consideration is given to the problem of mean velocity-Reynolds stress...
decoupling which appear when using a collocated grid arrangement which is a source of numerical instability. This is done by invoking a special interpolation procedure for the cell-face stresses in the context of Rhie(9). This practice results in the addition of normal stresses to the pressure term where the cell-face velocity is sensitized to the pressure differences as well as to normal stress differences at the nodes surrounding the face.

Turbulence Model Decks (Modules)

As the state-of-the-art of computers has advanced, so has the range, size and complexity of flow models being applied. Users have become more sophisticated and there is a constant demand for improvement. CFD codes have adapted to this demand and many general-purpose computer codes have been developed and used. As general-purpose codes become larger, their code structure becomes sophisticated. In general codes can be divided into three main areas, they include; 1) Numerical algorithms (which can be subdivided into discretization methods and solution techniques). 2) Methods of dealing with complex geometries. 3) Physical models (which include turbulence models, porosity, combustion kinetics, two-phase flow...). It seems, therefore, that the practicing engineer must have the knowledge of all these elements of the CFD program in order to successfully utilize this code. To obtain the maximum benefits from these general-purpose CFD codes, modularization of the code structure may be necessary. That is developing individual modular routines for the solver and for different physical models for example. If such modules are successful it would allow users to concentrate their talents on developing and improving physical hypothesis such as turbulent models for example that can easily be tested using such modules.

In the present work, turbulent modules are being developed to meet this need. Figure 1, shows a flowchart of a turbulence module interfaced with a typical main flow solver. The module is called by the flow solver passing to it the mean flow velocities, mass fluxes at cell faces and grid information among others. The turbulence differential equations are discretized and the matrix coefficients are setup and solved using Stones strongly implicit method(16). In the ASM module, the set of algebraic stress equations are solved simultaneously using Gauss-Seidel method at each step or iteration. In the eddy-viscosity models the values of k, e, and eddy viscosity (μ) are passed to the main flow solver, while, in the second moment closure models the Reynolds stresses \( \overline{u' \overline{v'}} \) are passed to the main solver. The solver then calls subroutine MODIFY of the module where the momentum sources are modified to account for the near-wall shear stresses in the eddy-viscosity models or to calculate Reynolds stress gradients in the second moment models.

These modules are structured to be self-contained and transportable to a number of general purpose CFD solvers to maximize computational efficiency. They have been tested independently at the University of Alabama at Huntsville using the MAST code.

Results

The various turbulence models are analyzed by comparing model predictions with the experimental data of Roback and Johnson(17) for swirling flow in confined double concentric jets with a sudden expansion.

Figure 2, shows the decay of the mean axial centerline velocity using both the single and multi-scale k-e models. Figure 2a, shows the comparison using the wall-function near wall approach and Figure 2b shows the results using the two-layer near wall model. The single-scale k-e model seems to underpredict the extent of the central recirculation zone as compared with the multi-scale k-e model. Moreover, improved comparisons with the data are obtained using the two-layer near wall model. Figure 3, shows the radial profile of the mean axial velocity at two distances downstream of the jet exit. Again, the two-layer model predicts better comparisons with the data than the wall function approach. The radial profiles of the mean tangential velocity are shown in Figure 4.

Figure 5, shows the radial profile of the mean axial velocity at three axial locations using the algebraic stress model (ASM) with wall function and two-layer near wall models. The radial profiles of the rms axial turbulent intensity are shown in Figure 6. Streamline contours are shown in Figure 7 using the single-scale k-e and the ASM models with the two-layer near wall model. The extent of the central vortex is better predicted using the ASM model. Preliminary results were also obtained using the full Reynolds stress model (RSM). Comparisons with the backward facing step data of Driver and Seegmiller(18) shows improved predictions over the single scale k-e model as shown in Figure 8 where the radial profiles of the axial normal stress and shear stress are plotted at four step heights downstream. Further testing of the RSM model for swirling flows are planned.
Conclusions

Different turbulence models for industrial applications have been formatted in a modular form and successfully interfaced and tested independently using two different main flow solvers. The turbulence models include the single and multi-scale k-ε models both with wall functions and two-layer near wall models. Second moment models that include the algebraic (ASM) and full Reynolds stress model (RSM) have been tested. It was shown that the two-layer near-wall model improves predictions as compared to the wall function approach. Convergence of the stiff ASM model equations was obtained by solving the 6x6 stress equations (for axisymmetric/swirling flows) at each iteration. The wall-reflection terms in the pressure-strain model showed little or no improvements in the ASM model predictions. Elaborate pressure-strain models that require no wall-damping are needed e.g. Speziale et al. The full Reynolds stress model (RSM) promises to be the next model to be used for engineering applications.

References

(5) Rodi W., Z. Ang. Math. und Mech., vol 56, 1976, pp.219-
(9) Rhie C., Univ. of Ill., Urbana-Champaign, Ph.D. Thesis, 1981
(16) Stone H., SIAM J. num. anal. vol.5, 1968, pp.530-
Decay of mean axial centerline velocity

Figure 2. Mean axial centerline velocity
--- single-scale k-ε model
--- multi-scale k-ε model

Figure 3. Radial profile of mean axial velocity

Figure 4. Radial profiles of mean tangential velocity

Figure 5. Radial profiles of mean axial velocity
--- ASM turbulence model
--- wall function model
--- two-layer model
Figure 6. Radial profiles of the axial turbulent intensity
ASM turbulence model
--- wall function model
--- two-layer model

Figure 7. Streamline contours
Figure 8. Backward facing step (Driver & Seegmiller)

--- single-scale k-ε model

--- RSM model
A NUMERICAL STUDY OF TWO-DIMENSIONAL VORTEX SHEDDING FROM RECTANGULAR CYLINDERS

A. H. HADID†  M. M. SINDIR†  R. I. ISSA†

Abstract

An efficient time-marching, noniterative calculation method is used to analyze time-dependent flows around rectangular cylinders. The turbulent flow in the wake region of a square section cylinder is analyzed using an anisotropic k-ε model. Initiation and subsequent development of the vortex shedding phenomenon is naturally captured once a perturbation is introduced in the flow. Transient calculations using standard eddy-viscosity and anisotropic k-ε models, averaged over an integral number of cycles to get the fluctuating energy (organized and turbulent), are compared with experimental data. It is shown that the anisotropic k-ε model resolves the anisotropy of the Reynolds stresses and gives mean energy distribution closer to the experiment than the standard k-ε model.

1. INTRODUCTION

Vortex shedding is a periodic unsteady flow phenomenon that occurs frequently behind bluff bodies and is therefore of great practical importance. Many attempts to calculate the two-dimensional (2-D) vortex shedding motion past square and circular cylinders by solving the unsteady Navier-Stokes equations were successful at low Reynolds numbers where the flow is laminar and the fluctuations are periodic, e.g., [1] and [2]. At higher Reynolds numbers which are more relevant in practice, turbulent fluctuations are superimposed on the periodic unsteady motion. The problem then concerns the decomposition of the flow into organized motion that is resolved in the calculation and a remaining turbulent motion to be represented by a turbulence model. Previous analysis of vortex shedding calculations at high Reynolds numbers have not been successful due to the inadequacy of the standard k-ε model and the lack of affordable higher order models that take into account the anisotropy of the turbulent intensities.

Franke et al. [3] analyzed the unsteady turbulent flow for a square cylinder using the standard k-ε model. They showed that the model tends to damp the periodic shedding motion underpredicting the Strouhal number. They also analyzed the detailed experimental results of Cantwell and Coles [4] for vortex shedding in the 2-D wake behind a circular cylinder. They additionally point out the need for improved models that account for the history and transport effects of the individual stresses. MacInnes et al. [5] used the standard k-ε model to simulate the periodically forced turbulent mixing layer investigated experimentally by Weisbrot and Wygnanski [6]. They managed to capture the main features of the mixing layer development where there is a clear distinction between the organized and the random turbulent motion.

Presented at the 4th International Symposium on Computational Fluid Dynamics - Davis on September 9 - 12, 1991. Received on March 21, 1992. Received in final form on May 12, 1992.
† CFD Technology Center, Rockwell Int./Rocketdyne Div., 6633 Canoga Ave., Canoga Pk., CA 91303, USA
‡ Dept. Mineral Resources Engineering, Imperial College of Science, Technology and Medicine, London, SW7, 2BP, England
Majumdar and Rodi [7] have shown that the separated turbulent flow past circular cylinders cannot be predicted realistically without a time-accurate numerical procedure to account for the periodic shedding of vortices.

Experimental investigations are needed to judge the different numerical and turbulent schemes. Durão et al. [8] conducted an experimental study of transient turbulent flow behind a square cylinder. They used spectral analysis and digital filtering of the LDV data in order to separate and quantify the turbulent and periodic, nonturbulent motions. They show for example that in the zone of highest velocity fluctuations the energy associated with the turbulent fluctuations is about 40% of the total energy. Therefore, for a successful simulation of transient turbulent flows, a reliable time-accurate numerical procedure and a good turbulence model are needed.

The purpose of the present paper is to model turbulent vortex shedding flows using an efficient time-accurate numerical procedure based on the PISO [9] methodology. Calculations of the turbulent vortex shedding are performed using the two-equation k-ε model with isotropic eddy-viscosity and with a modified two-equation model using an anisotropic eddy-viscosity. In the anisotropic model, nonlinear corrections are added to improve the eddy-viscosity representation of the Reynolds stresses as developed by Yoshizawa [10] with the aid of a two-scale direct interaction approximation. A similar anisotropic eddy-viscosity model was also developed by Speziale [11]. The adequacy of the models to simulate transient turbulent flows is assessed with the aid of the experimental results of Durão et al. [8] for vortex shedding in the 2-D wake behind a square cylinder at \( \text{Re} = 14,000 \).

2. MODEL EQUATIONS

The basic equations of motion in transient periodic flows can be written after separating the flow into an organized (phase averaged) component

\[
U_i(x_i, t) = \frac{1}{N} \sum_{n=0}^{N} u_i(x_i, t + nT) \tag{1}
\]

where \( U_i(x_i, t) \) is the resolvable portion of the instantaneous velocity \( u_i \), and \( T \) is the period of the oscillation, and a random turbulent component \( u'_i(x_i, t) \). The instantaneous velocity \( u_i(x_i, t) \) is then given by

\[
u_i = U_i + u'_i = \overline{u}_i + \overline{u}'_i + u'_i \tag{2}
\]

where \( \overline{u}_i \) is the time-mean component of the velocity, and \( \overline{u}'_i \) is the periodic fluctuating component. Assuming an incompressible flow, the momentum equations can be written after applying phase averaging as;

\[
\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \nu \frac{\partial U_i}{\partial x_j} + R_{ij} \right) \tag{3}
\]

where \( R_{ij} = -< u'_i u'_j > \) is the phase-averaged Reynolds stress tensor and \( \nu \) is the kinematic viscosity.

**Standard Isotropic k-ε Model**

In the standard isotropic k-ε model [12], \( R_{ij} \) is approximated by using the eddy-viscosity \( \nu_e \) as;

\[
R_{ij} = -\frac{2}{3} k \delta_{ij} + \nu_e \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \tag{4}
\]
where $k$ is the phase-averaged turbulent kinetic energy and $\nu_t = C_k(k^2/\epsilon)$, $\epsilon$ is the phase-averaged energy dissipation rate and $C_k$ is a model constant. The spatial and temporal distribution of $k$ and $\epsilon$ are determined from differential transport equations of these quantities

$$\frac{\partial k}{\partial t} + U_k \frac{\partial k}{\partial z_i} = \left( \nu + \nu_t \frac{\partial k}{\partial z_i} \right) + G - \epsilon$$

(5)

$$\frac{\partial \epsilon}{\partial t} + U_k \frac{\partial \epsilon}{\partial z_i} = \left( \nu + \nu_t \right) \frac{\partial \epsilon}{\partial z_i} + \frac{\epsilon}{k} (C_1 G - C_2 \epsilon)$$

(6)

where $G = R_j \frac{\partial U_j}{\partial z_i}$ is the turbulent generation term. The constants $C_k$, $C_1$, $C_2$, $\sigma_k$, and $\sigma_\epsilon$ have values of 0.09, 1.44, 1.92, 1.0, and 1.3, respectively.

Anisotropic k-$\epsilon$ Model

In the anisotropic model the Reynolds stresses can be expressed as;

$$R_{ij} = -\frac{2}{3} k \delta_{ij} + \nu_t \left( \frac{\partial U_i}{\partial z_j} + \frac{\partial U_j}{\partial z_i} \right) + \frac{1}{3} \left( \sum_{m=1}^{3} \tau_m S_{mk} \right) \frac{\partial U_i}{\partial z_m} - \sum_{m=1}^{3} \tau_m S_{mj}$$

(7)

$$\tau_m = C_{\tau m} \frac{k^3}{\epsilon^2}$$

(8)

$$S_{1ij} = \frac{\partial U_i}{\partial z_j} \frac{\partial U_j}{\partial z_k}$$

(9)

$$S_{2ij} = \frac{1}{2} \left( \frac{\partial U_i}{\partial z_k} \frac{\partial U_k}{\partial z_j} + \frac{\partial U_k}{\partial z_i} \frac{\partial U_i}{\partial z_k} \right)$$

(10)

$$S_{3ij} = \frac{\partial U_k}{\partial z_i} \frac{\partial U_k}{\partial z_j}$$

(11)

and $C_{\tau m}$ ($m = 1, 2, 3$) are model constants. The first two terms on the right hand side of (7) give the familiar isotropic eddy-viscosity representation, while the third and fourth terms express the anisotropy of $R_{ij}$. These additional nonlinear quadratic terms of the mean velocity gradients seem to be a simple way to resolve the individual normal stresses with the k-$\epsilon$ model. The anisotropy is reflected especially in the k-$\epsilon$ equation where both the diffusion and production terms are quadratic forms of the mean velocity gradients and turbulent kinetic energy gradients.

The anisotropic eddy-viscosity model has been successfully used by Nisizima and Yoshizawa [13] and Myong and Kasagi [14] for fully developed turbulent channel flows. In their calculations only $C_{\tau 1}$ and $C_{\tau 2}$ were optimized to reproduce the anisotropy of the turbulent intensities since $C_{\tau 3}$ does not appear in their equations. In the present study the flow is shear dominated with little departure from isotropy. Therefore, the model constants $C_{\tau 1}$, $C_{\tau 2}$, and $C_{\tau 3}$ were optimized to 0.01, 0.01, and 0.001, respectively, to satisfy the realizability constraint. (Note: zero constants reduce to the isotropic eddy-viscosity model.)

Applications of the k-$\epsilon$ isotropic and anisotropic eddy-viscosity models were made using wall functions to bridge the viscosity affected near the obstacle wall region. It is assumed that inadequacies in near-wall modelling play a minor role to the inaccuracy of normal Reynolds
stress differences arising from use of an isotropic eddy viscosity. Improvements can be made by integrating all the way to the wall [15] or by using the two-layer model of Chen and Patel [16].

3. NUMERICAL METHOD

The PISO methodology [9], in conjunction with a finite-volume technique, is used to solve the implicitly discretized, time-dependent flow equations. The method is essentially noniterative, where the solution process is split into a series of steps whereby operations on pressure are decoupled from those on velocity at each time-step. The avoidance of iterations substantially reduces the computational effort compared with that required by iterative methods. This is possible since the splitting error of PISO is negligibly small at the level of time-step required to eliminate the temporal truncation error. A backward temporal difference scheme is used, while the convective terms are discretized using a second-order upwind difference scheme. The method can also be used for steady-state flows, e.g., Hadid et al. [17].

Calculations are performed for the turbulent flow around a square cylinder (step height, \( H = 20 \text{ mm} \)) in a domain extending about 16 \( H \) downstream and 2.5 \( H \) upstream of the obstacle.
2-D Vortex Shedding from Rectangular Cylinder

The calculations captured the vortex shedding phenomenon after perturbing the flow at the inlet. A reference velocity of 0.68 m/s and turbulence intensity of 6\% (i.e., $k = \langle u'^2 \rangle = 3.6 \times 10^{-3} \text{ m}^2/\text{s}^2$) were used as the inlet conditions. The length scale $L$ of turbulence at the inlet was not measured in the experiment but an order of $L \sim 0.1 \text{ mm}$ was assumed from which the energy dissipation rate $\varepsilon = k^{3/2}/L$ was estimated. It is expected that the calculated results are not sensitive to the precise value of $\varepsilon$ used at the inlet. The upper and lower boundaries are treated as symmetry planes, at the exit, a zero-gradient outflow boundary condition is applied to each variable. The computational domain is resolved by 75x40 grid cells with clustering at the obstacle walls. An optimized time step of 0.001 sec. was chosen for the calculations.

4. RESULTS AND DISCUSSION

Figure 1(a) shows the normal velocity history at the centerline of the wake for $Re = 14000$ at five step heights downstream. The power spectrum of the normal velocity fluctuations (Fig.1(b)) confirms the oscillatory nature of the flow with a single predominant frequency of about 4.7 Hz, which is in agreement with experimental results [8]. Figure 1(c) shows a marker particle trace at time=3 sec., which illustrates the shedding pattern. In order to calculate the time-mean kinetic energy of the velocity fluctuations, the fluctuating velocity component (organized + turbulent) is $\bar{u} = u - \overline{U}$. For the 2-D plane geometry considered, the kinetic energy of the velocity fluctuations can be written as;

$$E = \frac{3}{4} \left( \bar{u}_1^2 + \bar{u}_2^2 \right)$$

(12)

where $\bar{u}_1^2 = \bar{u}_1^2 - 2\bar{u}_1\bar{U}_1 + \overline{U_1^2}$, and the time-mean value of the kinetic energy of the velocity fluctuations is

$$\overline{E} = \frac{3}{4} \left( \overline{\bar{u}_1^2} + \overline{\bar{u}_2^2} \right)$$

(13)
where \( \overline{u_i}^2 = \left( \overline{u_i^2} - 2\overline{u_i'U_i} + \overline{U_i^2} \right) = \left[ \overline{U_i'}^2 - 2(\overline{U_i'} \overline{U_i}) - 2\overline{u_{i}'U_i'} \right] \)

and from the definition of time averaging \( \overline{u_i'U_i} = 0 \), we get,

\[ \overline{u_i}^2 = \overline{U_i}^2 - \overline{U_i'}^2 + \overline{u_i'}^2 \quad (i = 1, 2) \quad (14) \]

The first two terms on the right hand side of (14) represent the organized periodic energy contribution, while the last term represents the turbulent energy contribution.

Figure 2 shows the distribution of the mean axial velocity at the centerline. The anisotropic model gives better distribution downstream of the obstacle. Figure 3 compares the calculated distribution of the time-mean kinetic energy of the fluctuating motion (periodic + turbulent) along the centerline of the flow. The figure shows a better trend exhibited by the anisotropic k-\( \epsilon \) model due to the improved resolution of the normal stresses. The standard k-\( \epsilon \) model acts to damp the periodic fluctuations by producing too much eddy viscosity, which underestimates the time-averaged momentum transfer. Hence, the length of the separation region behind the obstacle is overpredicted. Also, the maximum of the kinetic energy at the centerline lies further downstream. The length of the recirculation zone and the location of the maximum fluctuating energy are improved by using the anisotropic model. The figure also shows some fluctuating energy in front of the obstacle, whereas measurements indicated that the flow remained virtually laminar there. This is because in the k-\( \epsilon \) model the large velocity gradients at the stagnation region produce large turbulent kinetic energy. Results are also obtained from calculations in which the production of \( k \) in front of the obstacle was suppressed. Figure 4 shows the mean axial velocity distribution indicating better comparison with the experiment downstream of the obstacle. Figure 5 shows the distribution of the mean kinetic energy along the centerline. It can be seen that suppressing the production of the kinetic energy in front of the obstacle causes an...
increase in the fluctuating energy. Also, the peak of the energy fluctuations is shifted slightly upstream closer to the experimental data. The figure also shows smaller residual fluctuating energy in front of the obstacle. Figure 6(a) and (b) show the contour plots of the normal turbulent stress term $< v' v' >$ at an instant $T = 3$ sec. It can be seen that the anisotropic k-ε model produces higher $< v' v' >$ values, which act to increase the total fluctuating energy.

5. CONCLUSIONS
The turbulent vortex shedding flow behind a square cylinder was analyzed using an efficient time-accurate numerical method based on the PISO methodology. Turbulence was modeled using an anisotropic k-ε model which resolves the anisotropy of the Reynolds stresses reasonably well. Comparisons with the experimental data show the advantages of the model as compared with the standard isotropic k-ε model. Accurate predictions, however, can only be made by accounting for the history and transport effects of the individual Reynolds stresses. The anisotropic k-ε model seems to offer a compromise between the computationally intensive
Reynolds stress model and the standard isotropic k-ε model.

REFERENCES

Single point modeling of rotating turbulent flows

By A. H. Hadid\textsuperscript{1}, N. N. Mansour\textsuperscript{2} AND O. Zeman\textsuperscript{3}

A model for the effects of rotation on turbulence is proposed and tested. These effects which influence mainly the rate of turbulence decay are modeled in a modified turbulent energy dissipation rate equation that has explicit dependence on the mean rotation rate. An appropriate definition of the rotation rate derived from critical point theory and based on the invariants of the deformation tensor is proposed. The modeled dissipation rate equation is numerically well behaved and can be used in conjunction with any level of turbulence closure. The model is applied to the two-equation $k$-$\varepsilon$ turbulence model and is used to compute separated flows in a backward-facing step and an axisymmetric swirling coaxial jets into a sudden expansion. In general, the rotation modified dissipation rate model show some improvements over the standard $k$-$\varepsilon$ model.

1. Motivation and objectives

The ability to accurately model the effects of rotation on turbulence has a wide variety of important applications in rotating machinery and combustion devices. Many turbulent flows of engineering importance involve combinations of rotational and irrotational strains. However, turbulence models of the eddy viscosity type are oblivious to the presence of rotational strains since they depend only on the mean velocity gradients through their symmetric part (i.e. the mean rate of strain tensor). The rotation rate, for example, does not explicitly enter the equations for the turbulent kinetic energy and its dissipation rate, yet evidence from experiments (Wigeland and Nagib 1978, Jacquin et al. 1990) and from direct numerical simulation (Bardina et al. 1985, Speziale et al. 1987, Mansour et al. 1991) show that the decay rate of turbulence is reduced by the presence of rotation.

The effects of rotation on turbulence are known to be subtle. They are manifested through changes in the spectrum of the turbulence caused by nonlinear interactions. For initially isotropic turbulence, rotation inhibits the cascade of energy from large to small scales. Zeman (1994) proposed a modified energy spectrum that takes into account the effects of rotation at high Reynolds number by introducing a rotation wavenumber, $k_\Omega = \sqrt{13}/\varepsilon$, below which rotation effects on spectral transfer are important. Much of the application work in simulating rotating flows have been conducted using varieties of eddy viscosity models ($k$-$\varepsilon$ or $k$-$l$) and second order closure models with modified dissipation rate transport equation to account for

\textsuperscript{1} Rocketdyne Division/Rockwell International
\textsuperscript{2} NASA/Ames Research Center
\textsuperscript{3} Center for Turbulence Research
rotational effects. However, most of these models fail to predict the asymptotic behavior of the turbulence decay rate in the limits of large rotation rate. The objectives of this work are to model the effects of rotation using single-point two equation models and to offer an appropriate definition of the mean rotation rate that is consistent with the fact that spin is the main cause of reduction in the dissipation rate.

2. Accomplishments

For incompressible viscous flow with constant properties, the modeled transport equations for the turbulent kinetic energy, $k$, and its dissipation rate, $\varepsilon$, that are widely used for engineering applications take the form;

\begin{align*}
k_{t} + U_j k_{,j} &= D_k + P_k - \varepsilon \\
\varepsilon_{t} + U_j \varepsilon_{,j} &= D_\varepsilon + P_\varepsilon - \Phi_\varepsilon
\end{align*}

where $D_k$ and $D_\varepsilon$ are the diffusion terms for $k$ and $\varepsilon$ respectively and are modeled as

\begin{align*}
D_k &= \left[ \left( \nu + \nu_t \right) k_{,j} \right]_{,j} \\
D_\varepsilon &= \left[ \left( \nu + \nu_t \right) \varepsilon_{,j} \right]_{,j}
\end{align*}

where $\nu$ is the laminar viscosity and $\nu_t$ is the eddy viscosity $= C_\mu k^2 / \varepsilon$. $\sigma_k$ and $\sigma_\varepsilon$ are the ratio of Prandtl to Schmidt numbers and are taken as constants. $P_k$ is the production term for $k$ given as $P_k = -\overline{u'_i u'_j} U_{i,j}$, where $u'_i u'_j$ is the Reynolds stress term and $U_i$ is the mean velocity in the $i$-direction.

Assuming that the production of the dissipation rate $P_\varepsilon$ is proportional to the production of turbulent kinetic energy $P_k$, i.e. $P_\varepsilon \sim P_k / T$ where $T$ is the turbulent time scale given by $T = k / \varepsilon$. Similarly assume that the destruction rate of dissipation rate $\Phi_\varepsilon$ is proportional to the turbulent energy dissipation rate term, i.e. $\Phi_\varepsilon \sim \varepsilon / T$. The modeled form of the dissipation rate equation becomes

\begin{equation}
\varepsilon_{t} + U_j \varepsilon_{,j} = D_\varepsilon + C_1 \frac{\varepsilon}{k} P_k - C_2 \frac{\varepsilon^2}{k}
\end{equation}

Due to the symmetry of the Reynolds stress tensor $u'_i u'_j$, the kinetic energy production term can be written as $P_k = -\overline{u'_i u'_j S_{ij}}$, where $S_{ij} = (U_{i,j} + U_{j,i})/2$ is the mean rate of strain tensor. Therefore it can be seen that the standard dissipation rate, eq. (3), has no explicit dependence on the mean rotation tensor $\Omega_{ij} = (U_{i,j} - U_{j,i})/2$. It follows that the commonly used modeled dissipation rate equation can only be affected indirectly by rotational strains through the changes that they induce in the Reynolds stress tensor.

In order to sensitize the dissipation rate equation to rotational effects, consider the simple case of isotropic turbulence in a rotating frame. In this case, an initially decaying isotropic turbulence is described by:

\begin{equation}
k_{,t} = -\varepsilon
\end{equation}
Equations (4) and (5) do not distinguish the difference between isotropic turbulence in a rotating frame and in an inertial frame. Models that have a non-zero rotational correction have been proposed by Bardina et al. (1985), for example, for rotating isotropic turbulence where eq. (5) takes the form

$$\varepsilon, = -C_2 \frac{\varepsilon^2}{k} - C_3 \Omega \varepsilon$$

with $C_2 = 1.83$ and $C_3 = 0.15$.

The above model is able only to accurately predict the reduction in the decay rate of the turbulent kinetic energy in rotating isotropic turbulence for weak to moderate rotation rates where the effects are small. However, for sufficiently high rotation rates and long enough time, the model drastically underpredicts the decay rate of the turbulent kinetic energy.

Hanjalic and Launder (1980) proposed a model for which the $\varepsilon$-transport equation in rotating isotropic turbulence takes the form

$$\varepsilon, = -C_2 \frac{\varepsilon^2}{k} - C_3 \Omega^2 k$$

where $C_2 = 1.92$ and $C_3 = 0.27$.

This model predicts unphysical behavior of negative dissipation rate at high rotation rates, thus violating the realizability constraint. Other modifications to the dissipation rate transport equation have been proposed to account for rotational strains, e.g. Raj (1975) and Pope (1978). Again they fail in one way or another to account accurately for the rotational effects.

3. Proposed model

In the present work, a new model is proposed that accounts for rotational effects and correctly predicts the asymptotic behavior at zero to infinite rotation rates. Consider the dissipation rate equation in rotating isotropic turbulence

$$\varepsilon, = - \left(1.7 + \frac{5}{6} \frac{\alpha^2}{\alpha^2 + 1}\right) \frac{\varepsilon^2}{k}$$

with

$$\alpha = 0.35 Ro^{-1}$$

where $Ro$ is the Rossby number defined as $Ro^{-1} = \Omega k / \varepsilon$. For $\Omega \gg 1$, $C_2 = 2.5$, which gives a power law exponent $n = 0.6$ (in $k \sim t^{-n}$) matching the power law proposed by Squires et al. (1993) for the asymptotic state of rotating homogeneous turbulence at high Reynolds numbers.

The experimental data of Jacquin et al. (1990) are used to test the proposed model. Their experiments consisted of measuring the velocity field and characteristic quantities characterizing the fluctuating field downstream of a rotating cylinder.
containing a honeycomb structure and a turbulence producing grid. The coupled
differential equations for \(k\) and \(\varepsilon\) describing the effects of rotation on an initially
isotropic turbulence can be written as

\[
k_{t} = -\varepsilon \\
\varepsilon_{t} = -\left(C_{2} + C_{3} \frac{\alpha^{2}}{\alpha^{2} + 1}\right) \frac{\varepsilon^{2}}{k}
\]

These equations were solved numerically using a fourth-order Runge-Kutta inte-
gration scheme. The model predictions (with \(C_{2} = 1.7\) and \(C_{3} = 5/6\)) are compared
with the experimental data of Jacquin et al. (1990) as shown in Fig. 1a. The model
predicts well the evolution of turbulent kinetic energy and its decay rate for a wide
range of rotation rates. We have also tested the model for the three Reynolds
numbers measured by Jacquin et al. (1990), and found similar agreement of the
model predictions with the data. We should point out at this point that the value
\(C_{2} = 1.7\), proposed here for zero rotation rate, is lower than the value convention-
ally used in \(k-\varepsilon\) modeling. We find that with the conventional value of \(C_{2} = 1.92\)
(and \(C_{3} = 3/5\)) the model fails to predict the experimental data (see Fig. 1b)

![Figure 1](image)

**Figure 1.** Decay of turbulent kinetic energy. Symbols are the data of Jacquin
et al. (1990), lines are the model predictions. \(\circ \& --- \Omega = 62.8 \text{ (rad/s)}\), \(\square \& ---- \Omega = 31.4 \text{ (rad/s)}\), \(\triangle \& \ldots \ldots \Omega = 15.7\). (a) Model predictions with \(C_{2} = 1.7\)
and \(C_{3} = 5/6\); (b) Model predictions with \(C_{2} = 1.92\) and \(C_{3} = 3/5\).

4. Rotation Rate For General Flows

In order to test the rotational correction proposed in eq. (8) to the dissipation
rate equation for general flows where the rotation rate is a function of position and
in the presence of mean strains, the question arises as to what is the appropriate
definition of the rotation rate, $\Omega$?

In most previous studies, the rotation rate or the mean vorticity $\Omega$ was replaced
by $\sqrt{\Omega_{ij}\Omega_{ij}/2}$, where $\Omega_{ij} = (U_{i,j} - U_{j,i})/2$ is the rotation rate tensor of the mean
flow. However, such definition does not distinguish between a vortex sheet and
a vortex. A definition of a vortex or a region of vorticity (with spin) was given
by Chong et al. (1990) -using the arguments of the critical point theory and the
invariants of the deformation tensor- as a region in space where the vorticity is
sufficiently strong to cause the rate of strain tensor to be dominated by the rotation
tensor, i.e. the rate of deformation tensor has complex eigenvalues. This definition
satisfies the principle of frame invariance since it depends only on the properties
of the deformation tensor. We shall use it because the reduction in the dissipation
rate is due mainly to the spin that the mean imposes on the turbulence. Consider
the matrix $D_{ij}$ of the elements of the deformation tensor,

$$D_{ij} = U_{i,j}$$  \hspace{1cm} (12)

which can be split to

$$D_{ij} = S_{ij} + \Omega_{ij}$$  \hspace{1cm} (13)

The complex eigenvalues of $D_{ij}$ are found by solving the characteristic equation

$$|D_{ij} - \lambda \delta_{ij}| = 0$$

where $\lambda$'s are the eigenvalues of $D_{ij}$. For a $3 \times 3$ matrix, $\lambda$ can
be found from the solution of

$$\lambda^3 + P\lambda^2 + Q\lambda + R = 0$$  \hspace{1cm} (14)

where $P$, $Q$ and $R$ are the matrix invariants and are given by

$$P = -U_{i,i}$$  \hspace{1cm} (15)

$$Q = \frac{1}{2}(P^2 - S_{ij}S_{ji} - \Omega_{ij}\Omega_{ji})$$  \hspace{1cm} (16)

$$R = \frac{1}{3}(-P^3 + 3PQ - S_{ij}S_{jk}S_{ki} - 3\Omega_{ij}\Omega_{jk}\Omega_{ki})$$  \hspace{1cm} (17)

For an incompressible flow $P = 0$ from continuity and the characteristic equation
becomes

$$\lambda^3 + Q\lambda + R = 0$$  \hspace{1cm} (18)

Now if

$$A = \left[ -\frac{R}{2} + \sqrt{\left(\frac{R^2}{4} + \frac{Q^3}{27}\right)} \right]^{1/3}$$

and,

$$B = \left[ -\frac{R}{2} - \sqrt{\left(\frac{R^2}{4} + \frac{Q^3}{27}\right)} \right]^{1/3}$$
then the three roots of $\lambda$ are:

$$A + B, -\frac{A + B}{2} + i\frac{A - B}{2}\sqrt{3}, -\frac{A + B}{2} - i\frac{A - B}{2}\sqrt{3}$$

That is $\lambda$ can have:

(i) all real roots which are distinct when

$$[(Q/3)^3 + (R/2)^2] < 0,$$

or

(ii) all real roots where at least two roots are equal when

$$[(Q/3)^3 + (R/2)^2] = 0,$$

or

(iii) one real root and a pair of complex conjugate roots when

$$[(Q/3)^3 + (R/2)^2] > 0.$$

We shall follow Chong et al. (1990) and define the rotation rate

$$\Omega = \Im(\lambda) = \frac{\sqrt{3}}{2}(A - B), \quad \text{when } [(Q/3)^3 + (R/2)^2] > 0,$$

$$\Omega = 0 \text{ otherwise.}$$

It is important to note that for two dimensional Cartesian flows, the rotation rate defined by Eq. (19) reduces to $\Omega = \sqrt{|Q|}$, when $Q$, the determinant of the deformation tensor matrix, is negative. For pure shear the definition, eq. (19) yields $\Omega = 0$. Conventional models that are calibrated for shear flows, need not be recalibrated when corrections based on $\Omega$ are added to the model.

5. Numerical Procedure

For a two-dimensional, incompressible and steady turbulent flow, the Reynolds averaged momentum, continuity, turbulent kinetic energy and dissipation rate equations can be written in the generalized form;

$$\frac{\partial}{\partial x}(\rho U \Phi) + \frac{1}{r} \frac{\partial}{\partial y}(\rho r V \Phi) = \frac{\partial}{\partial x} \left( \Gamma_\Phi \frac{\partial \Phi}{\partial x} \right) + \frac{1}{r} \frac{\partial}{\partial y} \left( r \Gamma_\Phi \frac{\partial \Phi}{\partial y} \right) + S_\Phi$$

Where $r = 1$ for Cartesian two-dimensional flow, and $y = r$ for two-dimensional axisymmetric flow. Table 1 gives a summary of the terms in eq. (20) for the dependent variables solved in the code.
Single point modeling of rotating turbulent flows

\[ \begin{array}{|c|c|c|}
\hline
\Phi & \Gamma_{\Phi_x} & \Gamma_{\Phi_r} & S_{\Phi} \\
\hline
1 & 0. & 0. & 0. \\
U & 2\mu_e & \mu_e & -\partial P/\partial x + 1/r \partial (\mu_x \partial V/\partial x)/\partial y \\
V & \mu_e & 2\mu_e & -\partial P/\partial y + \partial (\mu_x \partial U/\partial y)/\partial y \\
W & \mu_e & \mu_e & -\rho VW/r - W/r^2 \partial (r \mu_x )/\partial r \\
k & \mu + \mu_t/\sigma_k & \mu + \mu_t/\sigma_k & P_k - \rho e \\
\epsilon & \mu + \mu_t/\sigma_k & \mu + \mu_t/\sigma_k & C_1 P_k \epsilon /k - C_2 P \epsilon^2 /k \\
\hline
\end{array} \]

Table 1. Summary of the governing equations. \( \rho \) is the density, \( \Gamma_{\Phi_x} \) and \( \Gamma_{\Phi_r} \) are the exchange coefficients in the axial and radial directions respectively, \( S_{\Phi} \) is the source term for the variable \( \Phi \). In the table, \( \mu_e \) is the effective viscosity given as \( \mu_e = \mu + \mu_t \), where \( \mu \) is the laminar viscosity and \( \mu_t \) is the turbulent viscosity, \( \mu_t = C_\mu \rho k^2 / \epsilon \).

In the standard \( k-\epsilon \) turbulence model the constants \( C_\mu, C_1, C_2, \sigma_k \) and \( \sigma_t \) have the values 0.09, 1.44, 1.92, 1.0 and 1.0 respectively.

In the rotation modified \( k-\epsilon \) turbulence model, only \( C_2 \) takes the form given by eq. (11) i.e, \( C_2 = 1.7 + (5/6) \alpha^2 / (\alpha^2 + 1) \).

The governing transport eq. (20) is solved using the primitive variables on a nonstaggered mesh and converted into a system of algebraic equations by integrating over control volumes defined around each grid point. The SIMPLE pressure-correction scheme (Patankar 1980) is used to couple the pressure and velocities and the resulting algebraic equations are solved iteratively. The convective terms are differenced using a second-order upwind scheme while the diffusion terms are approximated by a central differencing scheme. The physical domain is discretized using a non-uniform mesh where grid points are clustered close to the walls.

6. Model Application

The performance of the present model for complicated recirculating flows is demonstrated through calculations and comparisons with the experimental data of Driver & Seegmiller (1985) for backward-facing step flows and with the experiments of Roback & Johnson (1983) for a confined swirling coaxial jets into a sudden expansion.

Figure 2, shows the streamlines for the backward-facing step using the rotation modified \( k-\epsilon \) turbulence model. The calculations were performed on a 100x40 grid points. The computational domain had a length of 50H (H is the step height) and a width of 9H. The experimental data were used to specify the inflow conditions for a channel flow calculation where the fully developed profiles at the channel exit were used as the inlet conditions for the backward-facing step calculations. Fully developed flow conditions were imposed at the outflow boundary. The standard wall function approach (Launder & Spalding 1974) was used to bridge the viscous sublayer near the wall.
A. H. Hadid, N. N. Mansour & O. Zeman

The computed reattachment lengths were 5.50H using the standard k-ε turbulence model and 6.22H for the rotation modified k-ε turbulence model. The modified k-ε model prediction is closer to the experimental value of 6.10H. While these results are encouraging, they are mainly due to the fact that we have changed the value of $C_2$ for the non-rotating case. In general, a change in the value of $C_2$ will result in poor predictions of the mean profiles. The mean velocity profile at three locations downstream are shown on Fig. 3, while the turbulent stress profiles at $X/H = 4$ are shown on Fig. 4. All the quantities were normalized by the step height ($H$) and the experimental reference free-stream velocity ($U_{ref}$). It can be seen that the overall performance of the rotation modified dissipation rate equation is better than the standard k-ε model especially in the recirculation region (Figs. 3a, and 4). Some improvements are also obtained in the recovery region using the modified k-ε model. Figure 5 shows the contours of the effective rotation rate used as defined by Eq. (19).

For the 2D/axisymmetric swirling flow computations, the expressions for the invariants $Q$ and $R$ (Eqs. (16) & (17) respectively) are expanded and Eq. (19) is used to obtain the values of $\Omega$. The model was used to predict the mean profiles for a confined double concentric jets with a swirling outer jet flow into a sudden expansion (Roback & Johnson, 1983, see Fig. 6). Measurements are available for the mean...
Single point modeling of rotating turbulent flows

FIGURE 3. Mean axial velocity profiles at different axial locations. o data (Driver & Seegmiller, 1985); --- modified k-ε model; ---- standard k-ε model. (a) $X/H = 4$, (b) $X/H = 8$, (c) $X/H = 12$.

FIGURE 4. Turbulent stress profiles at $X/H = 4$. o data (Driver & Seegmiller, 1985); --- modified k-ε model; ---- standard k-ε model. (a) $u_1'u_1'/U_{ref}^2$, (b) $u_2'u_2'/U_{ref}^2$, (c) $u_1'u_2'/U_{ref}^2$.

velocity profiles and velocity fluctuations downstream of the expansion. Simulations with a coarse nonuniform grid of 30×20 mesh points were made. However, there is some uncertainty about the inlet conditions to be used since the first velocity
FIGURE 5. Contours of the effective rotation rate, $\Omega$. Contour levels were set between (0.1,1.0) with an increment level = .01. are

FIGURE 6. Roback & Johnson’s swirling coaxial jets discharging into an expanded duct.

profiles measured were 5mm downstream of the expansion.

To predict this flow, the measured profiles at 5mm were adjusted near the edges and were used as inlet conditions at the expansion plane. Preliminary results obtained with the coarse mesh indicate similar trends as the experiment. Figure 7 shows the streamline contours using the standard and the modified $k$-$\varepsilon$ turbulence models. The figure shows that a closed internal recirculation zone forms at the center with an additional zone at the corners downstream of the step. This causes a flow diversion outwards with high gradients between these regions. Figure 8 shows the axial and tangential velocity profiles at 25 mm downstream of the expansion using the standard and the modified $k$-$\varepsilon$ turbulence models. Results in this case indicate little or no improvements offered using the modified $k$-$\varepsilon$ model over the standard $k$-$\varepsilon$ model. Finer mesh may improve the results but the uncertainties in the inlet boundary conditions raise the question about the adequacy of using this experiment for validation purposes.
Figure 7. Swirling coaxial jets discharging into an expanded duct. Streamfunction contour. \(-\) levels were set between (-0.15,0.) with an increment level = 0.01, \(-\) levels were set between (0.,0.7) with an increment level = 0.05. (a) Standard \(k\)-\(\varepsilon\) model, (b) Modified \(k\)-\(\varepsilon\) model.

Figure 8. Velocity profiles at \(X = 25\) mm. o data (Roback & Johnson, 1983); \(-\) modified \(k\)-\(\varepsilon\) model; \(-\) standard \(k\)-\(\varepsilon\) model. (a) Axial Velocity, (b) Tangential velocity.

7. Conclusions

A new simple model for the turbulent energy dissipation rate equation has been proposed to account for the rotational effects on turbulence. A frame invariant
The definition of the rotation rate proposed by Chong et al. (1990) based on the critical point theory was used. The model can be used in conjunction with any level of turbulence closure. It was applied to the two-equation $k$-$\varepsilon$ turbulence model and was tested for separated flows in a backward-facing step and for axisymmetric swirling jet into a sudden expansion. The model is numerically stable and showed improvements over the standard $k$-$\varepsilon$ turbulence model. It is important to point out that the present study was carried out to roughly evaluate the model, but that a systematic recalibration of the constants in the $k$-$\varepsilon$ model is needed before going any further with the proposed model.

The authors would like to acknowledge many discussions with Dr. K. Shariff regarding proper definition of the rotation rate.

REFERENCES


### Title and Subtitle

**Comparative Study of Advanced Turbulence Models for Turbomachinery**

### Author(s)

Ali H. Hadid and Munir M. Sindir

### Performing Organization Name and Address

Rocketdyne Division/Rockwell International  
6633 Canoga Avenue, P. O. Box 7922  
Canoga Park, CA 91303-7922

### Sponsoring Agency Name and Address

National Aeronautics and Space Administration  
George C. Marshall Space Flight Center  
Huntsville, AL 35812

### Abstract

A computational study has been undertaken to study the performance of advanced phenomenological turbulence models coded in a modular form to describe incompressible turbulent flow behavior in two-dimensional/axisymmetric and three-dimensional complex geometry. The models include a variety of two equation models (single and multi-scale k-ε models with different near wall treatments) and second moment algebraic and full Reynolds stress closure models. These models were systematically assessed to evaluate their performance in complex flows with rotation, curvature, and separation. The models are coded as self-contained modules that can be interfaced with a number of flow solvers. These modules are stand-alone satellite programs that come with their own formulation, finite-volume discretization scheme, solver and boundary condition implementation. They will take as input (from any generic Navier-Stokes solver) the velocity field, grid (structured H-type grid) and computational domain specification (boundary conditions), and will deliver, depending on the model used, turbulent viscosity, or the components of the Reynolds stress tensor. There are separate 2D/axisymmetric and/or 3D decks for each module considered.

The modules are tested using Rocketdyne's proprietary code REACT. The code utilizes an efficient solution procedure to solve Navier-Stokes equations in a non-orthogonal body-fitted coordinate system. The differential equations are discretized over a finite-volume grid using a non-staggered variable arrangement and an efficient solution procedure based on the SIMPLE algorithm for the velocity-pressure coupling is used. The modules developed have been interfaced and tested using finite-volume, pressure-correction CFD solvers which are widely used in the CFD community. Other solvers can also be used to test these modules since they are independently structured with their own discretization scheme and solver methodology. Many of these modules have been independently tested by Professor C.P. Chen and his group at the University of Alabama at Huntsville (UAH) by interfacing them with own flow solver (MAST).