Implementation of a Semi-Implicit $k - \varepsilon$ Turbulence Model

L. Davidson
CERFACS
42, Av. Gustave Coriolis
31057 Toulouse
FRANCE

Report TR/RF/90/25, 1990

1 Introduction

This report treats implementation of a $k - \varepsilon$ turbulence model into a two-dimensional finite volume code [12], [9]. The code solves the time averaged continuity equation ρ, the momentum equations ρu, ρv, and the equation for total energy ρe_o; the pressure is calculated using the gas law. The main features of the code are:

- explicit, cell-centered finite volume, central differencing, local time stepping
- four stage Runge-Kutta scheme for the mean flow equations
- the k and ε equations are solved using a semi-implicit solver (hybrid central/upwind scheme, ADI)
- fourth-order numerical non-homogenous dissipation term in all mean flow equations [14]

Initially attempts were carried out to solve the k and ε equations explicitly using the existing Runge-Kutta solver. However, no stable convergent solutions were obtained. The main cause for these problems was probably the large source terms. These terms contain the ratio ε/k which in regions of weak turbulence (laminar regions) causes problems since both k and ε tend to zero. In order to remedy these stability problems a semi-implicit solver has been implemented for solving k and ε.

2 The $k - \varepsilon$ Turbulence model

The $k - \varepsilon$ model has the following form [13]:

$$\frac{\partial}{\partial x_j}(\rho U_j k) = \frac{\partial}{\partial x_j}\{(\mu + \frac{\mu_t}{\sigma_k})\frac{\partial k}{\partial x_j}\} + P_k - \rho\varepsilon - S_k \tag{1}$$

$$\frac{\partial}{\partial x_j}(\rho U_j \varepsilon) = \frac{\partial}{\partial x_j}\{(\mu + \frac{\mu_t}{\sigma_\varepsilon})\frac{\partial \varepsilon}{\partial x_j}\} + \frac{\varepsilon}{k}(c_{1\varepsilon} P_k - c_{2\varepsilon} f_2 \rho\varepsilon) - S_\varepsilon \tag{2}$$

where

$$P_k = -\rho\overline{u_i u_j}\frac{\partial U_i}{\partial x_j}$$

The Reynolds stress tensor is given by

$$\rho\overline{u_i u_j} = -\mu_t(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial U_m}{\partial x_m}) + \frac{2}{3}\delta_{ij}\rho k$$

and turbulent viscosity is calculated as

$$\mu_t = \frac{c_\mu f_\mu \rho k^2}{\varepsilon}$$

The constant c_μ, and the turbulent Prandtl numbers are set to their standard values: $c_\mu = 0.09$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$ (see below for the remaining constants and damping functions).

Note that Eqs. 1, and 2 are in steady form. At each time step the steady equations are solved using an under-relaxation factor of 0.5 (see [10]).

2.1 Walls

There are two main types of wall boundary conditions that can be used for k and ε.

- Wall-functions (assumptions of local equilibrium in the logarithmic law region) in which k and ε are fixed at the points adjacent to the walls can be used. This way of treating the boundary conditions is simple and computationally cheap as very few grid lines are needed in the boundary layer. However, it is normally less accurate than using low Reynolds number models.

- In low Reynolds number $k - \varepsilon$ models [11] the whole boundary layer, including the viscous sublayer, is resolved. In order to account for the viscous effects, damping functions are introduced and additional sources are, in some models, also added. The advantage of this type of treatment of the near wall flow is that it is fairly accurate, but, of course, more expensive than wall-functions.

2.1.1 Wall-functions

In the wall functions implemented in the code k and ε-equations are not solved at the node adjacent to the wall. Instead k and ε are fixed according to the hypothesis of the logarithmic law of the wall and equilibrium between the production and dissipation terms in the k-equation. The following expressions are obtained [13]:

$$\begin{aligned} k &= c_\mu^{-1/2} u_\tau^2 \\ \varepsilon &= \frac{u_\tau^3}{\kappa n} \end{aligned}$$

where u_τ denotes the friction velocity which is obtained from the logarithmic law of the wall, n denotes the normal distance from the wall, and κ denotes the von Kàrman constant $(= 0.41)$.

Standard values on the constants have been used $c_1 = 1.44$, $c_2 = 1.9$, and the damping functions are set to one, $f_\mu \equiv f_2 \equiv 1$.

2.1.2 Low Reynolds number $k - \varepsilon$ model of Chien

The model of Chien [2] is presented below. The damping functions and the additional source terms in Eqs. 1 and 2 have the form:

$$\begin{aligned} f_\mu &= 1 - exp(-0.0115 n^+) \\ f_2 &= 1 - 0.22 exp\{-(R_T/6)^2\} \\ S_k &= 2\mu \frac{k}{n^2} \\ S_\varepsilon &= 2\mu \frac{\varepsilon}{n^2} exp(-0.5 n^+) \end{aligned}$$

where

$$\begin{aligned} R_T &= \frac{\rho k^2}{\mu \varepsilon} \\ n^+ &= \frac{\rho u_\tau n}{\mu} \end{aligned}$$

The constants are set according to [2]: $c_1 = 1.35$, $c_2 = 1.8$.

The boundary conditions at the walls are $k = \varepsilon = 0$.

2.1.3 Low Reynolds number $k - \varepsilon$ model of Jones & Launder

The model of Jones & Launder [7] is presented below. The damping functions and the additional source terms in Eqs. 1 and 2 have the form:

$$\begin{aligned} f_\mu &= exp(-\frac{3.4}{(1+0.02R_T)^2}) \\ f_2 &= 1 - 0.3exp(-R_T^2) \\ S_k &= 2\mu\{[\frac{\partial\sqrt{k}}{\partial x}]^2 + [\frac{\partial\sqrt{k}}{\partial y}]^2\} \\ S_\varepsilon &= -2\nu\mu_t\delta_{ij}[\frac{\partial^2 U_i}{\partial x_k \partial x_k}]^2 \end{aligned}$$

where

$$R_T = \frac{\rho k^2}{\mu\varepsilon}$$

The constants are set according to [7]: $c_1 = 1.44$, $c_2 = 1.92$.

The boundary conditions at the walls are $k = \varepsilon = 0$.

3 The discretization

Let Φ denote a general variable ($\Phi = k, \varepsilon$). Equations 1 and 2 can then both be written as

$$\frac{\partial}{\partial x_m}(\rho U_m \Phi) = \frac{\partial}{\partial x_m}(\Gamma_\Phi \frac{\partial \Phi}{\partial x_m}) + \overline{S}^\Phi \qquad (3)$$

where $\overline{S}^\Phi$ denotes source per unit volume. If a flux vector J_m containing convection and diffusion is defined as

$$J_m = \rho u_m \Phi - \Gamma_\Phi \frac{\partial \Phi}{\partial x_m} \qquad (4)$$

Eq. 3 can be written as:

$$\frac{\partial J_m}{\partial x_m} = \overline{S}^\Phi$$

In vector notation the equation reads:

$$\nabla \cdot \mathbf{J} \;=\; \overline{S}^{\Phi}$$

Integrating this equation over a volume (with volume V and bounding surface A) using Gauss' law, gives:

$$\int_A \mathbf{J} \cdot \mathbf{dA} \;=\; \int_V \overline{S}^{\Phi} dV$$

which for a control volume gives (in, for simplicity, one dimension)

$$\{\mathbf{J} \cdot \mathbf{A}\}_{i+1/2} + \{\mathbf{J} \cdot \mathbf{A}\}_{i-1/2} \;=\; S^{\Phi} \qquad (5)$$

where S^{Φ} is the total source in the control volume.

Note that for Cartesian coordinates (where $u \geq 0$) the positive sign in front of the contribution for the low-I ($i - 1/2$) is turned into a negative one because the scalar product between the two vectors ($\mathbf{J} \cdot \mathbf{A}$) is negative.

3.1 Convection

The convection, which is the first part of the flux vector $\mathbf{J}$ in Eq. 4, is the scalar product of the velocity vector and the area vector multiplied by the density and the variable Φ. For the high-I face it gives (see Fig. 1):

$$\{\dot{m}\Phi\}_{i+1/2} \;=\; \{\rho \mathbf{u} \cdot \mathbf{A}\Phi\}_{i+1/2} \qquad (6)$$

3.2 Diffusion

Diffusion is the second part of the flux vector $\mathbf{J}$ in Eq. 4, and it has the form:

$$\mathcal{D} \;=\; (\mathbf{J} \cdot \mathbf{A})_{diff} = -\Gamma_{\Phi} \mathbf{A} \cdot \nabla\Phi$$

For the high-I face, for example, it gives in Cartesian coordinates (x,y)

$$-\{\Gamma_{\Phi} \mathbf{A} \cdot \nabla\Phi\}_{i+1/2} \;=\; -\{\Gamma_{\Phi}(A_x \frac{\partial\Phi}{\partial x} + A_y \frac{\partial\Phi}{\partial y})\}_{i+1/2}$$

and in curvilinear coordinates (ξ_1, ξ_2) [4]

$$-\{\mathbf{A} \cdot \nabla\Phi\}_{i+1/2} = -\{\mathbf{A} \cdot \mathbf{g}_i g^{ij} \frac{\partial\Phi}{\partial\xi_j}\}_{i+1/2} = -\{\mid \mathbf{A} \mid \mathbf{n} \cdot \mathbf{g}_i g^{ij} \frac{\partial\Phi}{\partial\xi_j}\}_{i+1/2} \qquad (7)$$

The covariant (=tangential) base vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ correspond to the I and J-grid lines, respectively. The metric tensor appears because the components of the product $\mathbf{A} \cdot \mathbf{g_i}$ and the derivative $\partial\Phi/\partial\xi_j$ are both covariant, and the product of their (contravariant) base vectors is not equal to zero or one (as in Cartesian coordinate systems) since they are non-orthogonal to each other [6]. In order to further illustrate this take two arbitrary vectors, $\mathbf{r}$ and $\mathbf{t}$, in a non-orthogonal coordinate system. The product of these two vectors is

$$\mathbf{r} \cdot \mathbf{t} = (r_i \mathbf{g}^i)(t_j \mathbf{g}^j) = r_i t_j (\mathbf{g}^i \cdot \mathbf{g}^j) = r_i t_j g^{ij}$$

where $g^{11} \neq 1$, $g^{22} \neq 1$, $g^{21} = g^{12} \neq 0$. The components of g^{ij} in Eq. 7 have the form (see p. 196 in [5]), [4]

$$\begin{aligned} g^{11} &= \frac{1}{g} \\ g^{12} &= \frac{cos\alpha}{g} \\ g^{21} &= g^{12} \\ g &= 1 - cos^2\alpha \end{aligned}$$

where α denotes the angle between $\mathbf{g}_1$ and $\mathbf{g}_2$. The normal vector in Eq. 7 is equal to the cross product of $\mathbf{g}_2$ and $\mathbf{g}_3$, i.e.

$$\mathbf{n} = \mathbf{g}_2 \times \mathbf{g}_3 \equiv \mathbf{g}^1$$

which means that

$$\mathbf{n} \cdot \mathbf{g}_2 \;=\; 0$$

Equation 7 now gives

$$-\{\mathbf{A} \cdot \nabla\Phi\}_{i+1/2} = -\{\mid \mathbf{A} \mid \mathbf{n} \cdot \mathbf{g}_1 g^{1j} \frac{\partial\Phi}{\partial\xi_j}\}_{i+1/2} = -\{\mid \mathbf{A} \mid \mathbf{n} \cdot \mathbf{g}_1 (g^{11}\frac{\partial\Phi}{\partial\xi_1} + g^{12}\frac{\partial\Phi}{\partial\xi_2})\}_{i+1/2} \quad (8)$$

The equation can be rewritten so that

$$\{\Gamma_\Phi \mathbf{A} \cdot \nabla\Phi\}_{i+1/2} = D_{I,i+1/2}(\Phi_{i+1} - \Phi_i) + D_{J,i+1/2}(\Phi_{i+1/2,j+1/2} - \Phi_{1+1/2,j-1/2}) \quad (9)$$

where the D's contain all geometrical terms and the diffusion coefficient, and subscript I denotes coordinate direction I. The term associated with $D_{I,1+1/2}$ in Eq. 9 represents the orthogonal diffusion, and the last term in Eq. 9 represents the non-orthogonal diffusion (which is zero in Cartesian coordinate systems).

3.3 Source terms

The source terms are taken as constant in the control volume, i.e.

$$\int_V S^\phi dV = \overline{S}^\Phi \delta V = S^\Phi$$

3.4 The discretized equation

Combining Eqs. 5, 6 and 9 gives

$$\{\dot{m}_{i+1/2}\Phi_{i+1/2} - D_{I,i+1/2}(\Phi_{i+1} - \Phi_i)\} - \{\dot{m}_{i-1/2}\Phi_{i-1/2} - D_{I,i-1/2}(\Phi_i - \Phi_{i-1})\} = S^\Phi \tag{10}$$

where the non-orthogonal diffusion terms have been included in the source term. The problem now arises for the convection terms: how to estimate $\Phi_{i\pm1/2}$? It is well known that second order central differencing causes unphysical occilations in Euler calculations where no viscous damping effetcs are present. However, when the diffusive (viscous) transport is of the same order as the convective transport the use of central differencing poses no problems. The upwind scheme ($\Phi_{i+1/2} = \Phi_i$ if $\dot{m}_{i+1/2} \geq 0$ and $\Phi_{i+1/2} = \Phi_{i+1}$ otherwise) is always stable but only first order accurate. The compromise chosen in this work is the, in code based on SIMPLE-methods commonly used, hybrid central/upwind differencing [10]. In this scheme central differencing is used when diffusion is large enough (Peclet number smaller than two), and upwind differencing otherwise (the Peclet number P_e is the local Reynolds number based on the cell size, $P_e = U\delta x/(\nu_t + \nu)$). The switch from central differencing to upwind differencing is chosen at $P_e = 2$ in order to ensure the coefficients in the linearized equation (Eq. 13) to be positive. For the east face, for example, one obtains

$$\begin{aligned}&\{convection - diffusion\}_{i+1/2} = \{\dot{m}\Phi\}_{i+1/2} - D_{I,i+1/2}(\Phi_{i+1} - \Phi_i) = \\ &max\{\frac{1}{2}\dot{m}_{i+1/2} + D_{I,i+1/2}, 0, \dot{m}_{i+1/2}\}\Phi_i - max\{-\frac{1}{2}\dot{m}_{i+1/2} + D_{I,i+1/2}, 0, -\dot{m}_{i+1/2}\}\Phi_{i+1}\end{aligned} \tag{11}$$

Equation 11 into 10 gives

$$\begin{aligned}&max\{\frac{1}{2}\dot{m}_{i+1/2} + D_{I,i+1/2}, 0, \dot{m}_{i+1/2}\}\Phi_i - max\{-\frac{1}{2}\dot{m}_{i+1/2} + D_{I,i+1/2}, 0, -\dot{m}_{i+1/2}\}\Phi_{i+1} \\ &-max\{\frac{1}{2}\dot{m}_{i-1/2} + D_{I,i-1/2}, 0, \dot{m}_{i-1/2}\}\Phi_{i-1} + max\{-\frac{1}{2}\dot{m}_{i-1/2} + D_{I,i-1/2}, 0, -\dot{m}_{i-1/2}\}\Phi_i = S^\Phi\end{aligned} \tag{12}$$

This can be rewritten, using the continuity equation ($\dot{m}_{i+1/2} - \dot{m}_{i-1/2} = 0$) and linearizing the source terms as $S^\Phi = S_P^\Phi \Phi - S_C$ (subscript P denotes Proportional, C denotes Constant), so that

$$a_{P,i}\Phi_i = a_{E,i}\Phi_{i+1} + a_{W,i}\Phi_{i-1} + S_{C,i}^\Phi \tag{13}$$

where

$$\begin{aligned}a_E &= max\{-\frac{1}{2}\dot{m}_{i+1/2} + D_{I,i+1/2}, 0, -\dot{m}_{i+1/2}\}\\ a_W &= max\{\frac{1}{2}\dot{m}_{i-1/2} + D_{I,i-1/2}, 0, \dot{m}_{i-1/2}\}\\ a_P &= a_W + a_E - S_P\end{aligned}$$

Index P, E, W denote Point, East and West, respectively. The source terms for, for example, $\Phi = k$ contain the generation and dissipation terms (see Eq. 1) as well as the non-orthogonal diffusion. The first and the last contributions are included into the constant part S_C whereas the dissipation term is linearized using the expression for the turbulent viscosity

$$-\rho\varepsilon = c_\mu f_\mu \rho^2 k^2/\mu_t$$

and included into S_P so that

$$\begin{aligned}S_P &= -c_\mu f_\mu \rho^2 k/\mu_t \delta V\\ S_C &= D_{J,i+1/2}(k_{i+1/2,j+1/2} - k_{i+1/2,j-1/2}) - D_{J,i-1/2}(k_{i-1/2,j+1/2} - k_{i-1/2,j-1/2}) + P_k\delta V\end{aligned}$$

The extension to two dimensions is straight-forward and the resulting form of the discretized equation reads

$$\begin{aligned}a_{P,i,j}\Phi_{i,j} &= a_{E,i,j}\Phi_{i+1,j} + a_{W,i,j}\Phi_{i-1,j} + a_{N,i,j}\Phi_{i,j+1} + a_{S,i,j}\Phi_{i,j-1} + S^{\Phi}_{C,i,j}\\ a_{E,i,j} &= max\{-\frac{1}{2}\dot{m}_{i+1/2,j} + D_{I,i+1/2,j}, 0, -\dot{m}_{i+1/2,j}\}\\ a_{W,i,j} &= max\{\frac{1}{2}\dot{m}_{i-1/2,j} + D_{I,i-1/2,j}, 0, \dot{m}_{i-1/2,j}\}\\ a_{N,i,j} &= max\{-\frac{1}{2}\dot{m}_{i,j+1/2} + D_{J,i,j+1/2}, 0, -\dot{m}_{i,j+1/2}\}\\ a_{S,i,j} &= max\{\frac{1}{2}\dot{m}_{i,j-1/2} + D_{J,i,j-1/2}, 0, \dot{m}_{i,j-1/2}\}\\ a_{P,i,j} &= a_{W,i,j} + a_{E,i,j} + a_{S,i,j} + a_{N,i,j} - S_{P,i,j} \qquad (14)\end{aligned}$$

where index i, j has been added as to remind that the a-coeffcients and the source terms are matrices.

4 Solver

A tri-diagonal solver is used, and since it solves linear equations along *lines* (say I-line), Eq. 14 is rewritten so that the contributions to point (i, j) from

$(i, j-1)$ and $(i, j+1)$ are transferred to the rigth-hand-side, which in matrix form gives

$$\begin{pmatrix} a_{P,1} & -a_{E,1} & & & & & \\ -a_{W,2} & a_{P,2} & -a_{E,2} & & & & \\ & & \ldots\ldots\ldots\ldots & & & & \\ & & & -a_{W,i} & a_{P,i} & -a_{E,i} & \\ & & & \ldots\ldots\ldots\ldots & & & \\ & & & & -a_{W,ni} & a_{P,ni} & \end{pmatrix} \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \ldots \\ \Phi_i \\ \ldots \\ \Phi_{ni} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \ldots \\ b_i \\ \ldots \\ b_{ni} \end{pmatrix}$$

where

$$b_i = S_{C,i,j} + a_{N,i,j}\Phi_{i,j+1} + a_{S,i,j}\Phi_{i,j-1}$$

Eliminating the $a_{W,2}$-element by adding to row 2 row 1 multiplied with $a_{W,2}/a_{P,1}$ gives

$$\begin{pmatrix} a_{P,1} & -a_{E,1} & & & & & \\ & A_{P,2} & -a_{E,2} & & & & \\ & & \ldots\ldots\ldots\ldots & & & & \\ & & & -a_{W,i} & a_{P,i} & -a_{E,i} & \\ & & & \ldots\ldots\ldots\ldots & & & \\ & & & & -a_{W,ni} & a_{P,ni} & \end{pmatrix} \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \ldots \\ \Phi_i \\ \ldots \\ \Phi_{ni} \end{pmatrix} = \begin{pmatrix} b_1 \\ B_2 \\ \ldots \\ b_i \\ \ldots \\ b_{ni} \end{pmatrix}$$

where

$$\begin{aligned} B_2 &= b_2 + \frac{a_{W,2}b_1}{a_{P,1}} \\ A_{P,2} &= a_{P,2} - \frac{a_{W,2}a_{E,1}}{a_{P,1}} \end{aligned}$$

Doing this for each row, and thereby eliminating the lower diagonal, gives

$$\begin{pmatrix} a_{P,1} & -a_{E,1} & & & & \\ & A_{P,2} & -a_{E,2} & & & \\ & & \ldots\ldots\ldots\ldots & & & \\ & & & & A_{P,i} & -a_{E,i} \\ & & & \ldots\ldots\ldots\ldots & & \\ & & & & & A_{P,ni} \end{pmatrix} \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \ldots \\ \Phi_i \\ \ldots \\ \Phi_{ni} \end{pmatrix} = \begin{pmatrix} b_1 \\ B_2 \\ \ldots \\ B_i \\ \ldots \\ B_{ni} \end{pmatrix} \tag{15}$$

where

$$\begin{aligned} B_i &= b_i + \frac{a_{W,i}b_{i-1}}{A_{P,i-1}} \\ A_{P,i} &= a_{P,i} - \frac{a_{W,i}a_{E,i-1}}{A_{P,i-1}} \end{aligned} \tag{16}$$

Equation 15 can be written as

$$A_{P,i}\Phi_i - a_{E,i}\Phi_{i+1} = B_i \tag{17}$$

where $A_{P,i}$ and B_i are defined in Eq. 16. The linear equation system in Eq. 14 can now be solved along each i-line by first calculating all $A_{P,i}$ and B_i defined in Eq. 16, and then calculating Φ_{i+1} from Eq. 17.

Above in this section expressions have been derived to solve the linear equation system in Eq. 14 along I-lines. For solving Eq. 14 along J-lines , Eq. 14 is rewritten so that the contributions to point (i,j) from $(i-1,j)$ and $(i+1,j)$ are transferred to the rigth-hand-side, and the subsequent expressions are derived in exactly the same way as above. In this way Eq. 14 can be solved, alternating, along I-lines and J-lines.

At each time step the k-equation is solved twice (once along I-lines and once along J-lines), and then the ε-equation is solved twice, and, finally, the turbulent viscosity is calculated.

The current practice is to first calculate the case with the Baldwin-Lomax model. Then a restart is made, using the results obtained with Balwin-Lomax model; the first 500-1000 time steps only the k and ε equations are solved. If a low Reynolds number model is used, the high Reynolds number model (with wall-functions) is used during the first few (say 10) time steps in order to create turbulence in the whole field (k and ε will otherwise stay frozen at the initial values, close to zero). When this somewhat complicated procedure is used, k and ε need not to be initialized to any "sensible" values, but one can initialize k and ε to some small value (say $1 \cdot 10^{-10}$).

The additional CPU time required for the $k-\varepsilon$ model is rather modest: for the test case below one Runge-Kutta time step requires, on a FX/80 Alliant (with parallel and vector options), $3.08s$, while the $k-\varepsilon$ model required $0.51s$. Furthermore, the $k-\varepsilon$ model was solved only every second time step, which means that the $k-\varepsilon$ model only requires 9 % additional CPU time.

5 Test case

The low Reynolds number model of Chien [2] has been used for calculating the transonic flow in a plane channel with a bump [3]. In this report the flow around a single profile is calculated using the the same model.

A C-mesh is used (see Fig. 2), which was generated at Aerospatiale [1]. There are 64 nodes in the direction normal to the profile, 256 nodes around the profile, and 2x50 in the wake. The y^+ values for the first grid line close to the wall is close to one, and approximately seven grid lines are situtated in the region where $y^+ \leq 20$, except near the trailing edge close to the upper surface where the boundary layer is thicker.

The Reynolds number is $2.1 \cdot 10^6$, the Mach number is 0.15 and the angle of attack is 13.1^o; experiments have been carried out by Aerospatiale in two different windtunnels (F1 and F2).

Pressure iso-contours are presented in Fig. 3, and as can be seen the pressure is fairly regular except near the trailing edge where some pressure occilations appear, which can be due to that the grid is too coarse there. There is a rapid change in the calculated viscosity in the region of the trailing edge: in the boundary layer close to the surface it is small since the f_μ-function is small, but once the flow enters the wake the viscosity increases rapidly, which maybe explains the pressure occilations here.

There are also some irregularities in the pressure contours close to the upper surface at $x/L \approx 0.12$ (which even causes separation!); this is where the transition is imposed. The transition is imposed by setting, for $x/L \geq 0.12$, the turbulent viscosity to 1 % of the laminar one. It seems that this way of imposing transition is too crude and too abrupt. It should be noted that the transition at the lower surface is imposed at $x/L = 0.3$, but that the production of turbulence is not suffient here (the velocity gradients are smaller than at the upper surface), and the flow does not become turbulent until $x/L \simeq 0.7$.

It can also be seen small "bumps" in the pressure contours in the boundary layer on the upper surface near the trailing edge. These are due to the $-2/3\rho k$-term in the stress tensor, i.e. the last term in the expression below

$$\rho\overline{u_i u_j} \quad = \quad -\mu_t(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial U_m}{\partial x_m}) + \frac{2}{3}\delta_{ij}\rho k$$

If the last term is removed the "bumps" in the pressure contours dissappear. This means that the turbulent kinetic energy is large compared to the other diffusion term in the momentum equations in this region, which also maybe indicates that the grid should be finer here in order to damp the turbulent kinetic energy by the presence of the wall.

The stream-lines are shown in Fig. 4; no separation occurs which should be compared with the experimental value $x/L = 0.94$. The iso-contours of the turbulent kinetic energy in Fig. 5 show how the turbulent boundary layer grows, and, after the profile, is goes over in a turbulent wake. The c_P and c_F curves are compared with experimental data in Figs. 6-7. The predicted c_F values are, however, too high compared to the experimental values. In Fig. 7 the predicted negative c_F values show the small separation zone which was discussed above. The calculated C_{lift} is 1.60 which should be compared with the experimental values 1.55 and 1.49 in the F1 and F2 windtunnel, respectively.

6 Conclusions

Two versions of low Reynolds number $k-\varepsilon$ models have been implemented (the model of Chien and the model of Jones & Launder). Since no stable, convergent solutions could be obtained using the explicit Runge-Kutta solver, a semi-implicit solver has been implemented for solving the k and ε equations.

The model of Chien is used for calculating the flow around a single airfoil ($\mathcal{R}_e = 2.1 \cdot 10^6$, $\mathcal{M} = 0.15$, $\alpha = 13^o$). The predicted c_p values agree rather well with experiments, whereas the predicted c_F's are too high compared with experiments.

Some problems were encountered when imposing transition on the upper surface which caused separation at the position where transtion was imposed. Further work will be carried out to investigate this problem.

Work is currently going on calculating the same flow using the model of Jones & Launder. In the future work on the $k-\varepsilon$ models will be undertaken in order to test streamline curvature correction [8], and corrections of the near-wall turbulent length scales [15].

References

[1] **Ph. CHANEZ, and L. PALICOT**, Évaluation d'un code Navier-Stokes bidimensionnnel pour le calcul de l'écoulement autour d'un profile d'aile, Note interne Aerospatiale 443.548/90, 1990.

[2] **K.Y. CHIEN**, Predictions of Channel and Boundary Layer Flows with a Low-Reynolds-Number Turbulence Model, *AIAA Journal*, **20**, 33-38, 1982.

[3] **L. DAVIDSON**, Calculation of the Transonic Flow in a Plane Channel Using a Low Reynolds $k-\varepsilon$ Turbulence Model and the Baldwin-Lomax Turbulence Model, Rept. TR/RF/90/26, CERFACS, 1990.

[4] **L. DAVIDSON and P. HEDBERG**, Mathematical Derivation of a Finite-Volume Formulation for Laminar Flow in Complex Geometris, *J. Numer. Meth. Fluids*, **9**, 531-540, 1989.

[5] **W. FLÜGGE**, Tensor Analysis and Continuum Mechanics, Springer-Verlag, Berlin, 1972.

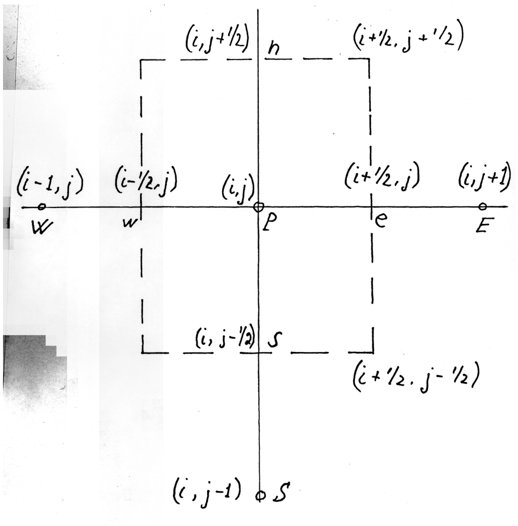

Figure 1: A control volume in, for simplicity, Cartesian coordinates. P (i,j) denotes an arbitrary node with its west neighbour W $(i-1,j)$, east E $(i+1,j)$, south S $(i,j-1)$, and north N $(i,j+1)$ neighbours. The faces of the control volume are located at e $(i+1/2,j)$, w $(i-1/2,j)$, n $(i,j+1/2)$, and s $(i,j-1/2)$.

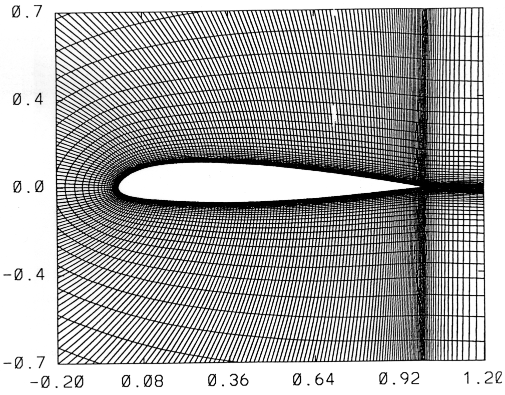

Figure 2: The C-grid close to the profile.

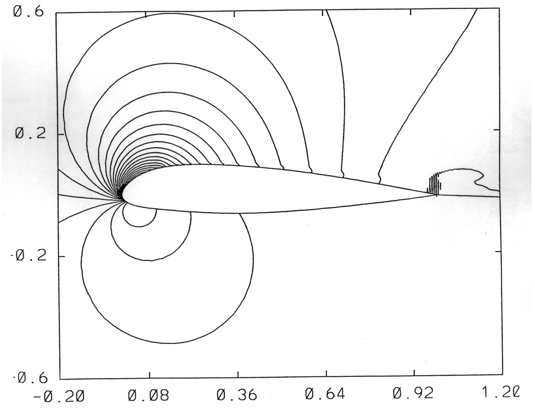

Figure 3: Calculated pressure iso-contours.

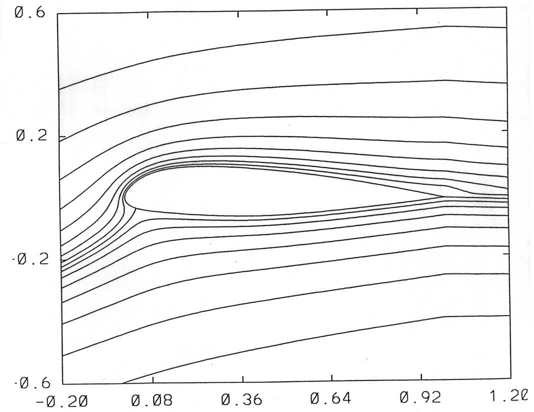

Figure 4: Calculated stream lines.

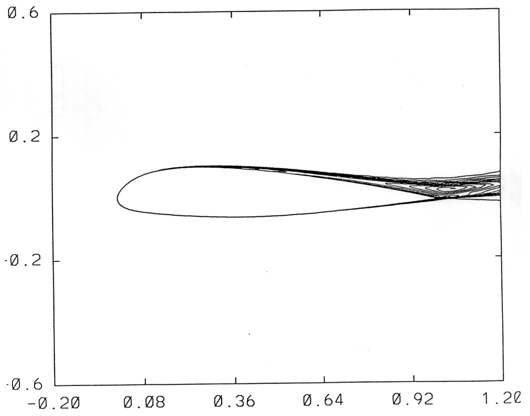

Figure 5: Calculated kinetic energy iso-contours.

[6] **F. IRGENS**, Tensoranalyse og kontinuumsmekanikk, (in Norwegian), del III, Institutt for mekanikk, Norges Tekniska Høgskole, Trondheim, 1966.

[7] **W.P. JONES and B.E. LAUNDER**, The Prediction of Laminarization With a Two-Equation Model of Turbulence, *Int. J. of Mass and Heat Transfer*, **15**, 301-314, 1972.

[8] **M.A. LESCHZINER and W. RODI**, Calculation of Annular and Twin Parallel Jets Using Various Discretization Schemes and Turbulence-Model Variations, *ASME J. Fluids Engng*, **103**, 352-360, 1981.

[9] **B. MÜLLER and A. RIZZI**, Runge-Kutta Finite Volume Simulation of Transonic Flow Over a NACA0012 Airfoil using the Navier-Stokes Equations, FFA TN 1986-60, 1986

[10] **S.V. PATANKAR**, Numerical Heat Transfer and Fluid Flow, McGraw-Hill, New York, 1980.

[11] **V.C. PATEL, W. RODI, and G. SCHEURER**, Turbulence Models for Near-Wall and Low Reynolds Number Flows: A Review, *AIAA Journal*, **9**, 1308-1319, 1986.

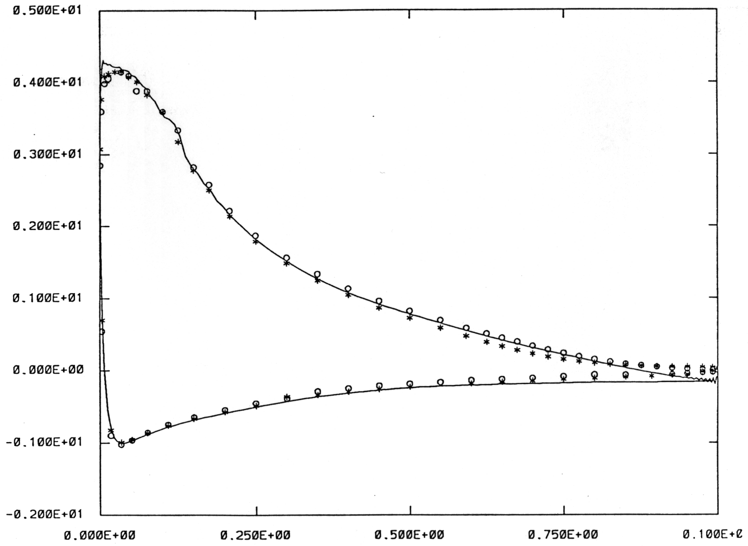

Figure 6: c_P. Lines: calculations; markers: experiments (o: F1 windtunnel; $*$: F2 windtunnel).

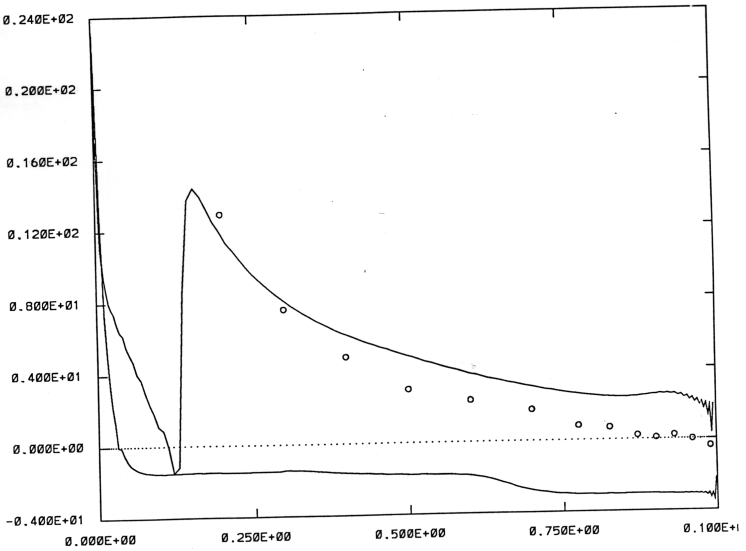

Figure 7: c_F. Lines: calculations; markers: experiments (F1 windtunnel).

[12] **A. RIZZI and B. MÜLLER**, Large-Scale Viscous Simulation of Laminar Vortex Flow Over a Delta Wing, *AIAA Journal*, **27**, 833-840, 1989.

[13] **W. RODI**, Turbulence Models and Their Application in Hydraulics, International Association of Hydraulic Research, Monograph, Delft, 1980.

[14] **R.C. SWANSON and E. TURKEL**, Artificial Dissipation and Central Difference Schemes for the EULER and Navier-Stokes Equations, AIAA Paper No 87-1107, 1987.

[15] **C. YAP**, Prediction of Convective Heat Transfer in Impinging and Separated Flows, 4th Biennial Colloquium on Computational Fluid Dynamics, 1990.