SSSU 130
ISSN 0140 3818

# An Evaulation of the RANS Method for the Prediction of Steady Ship Rudder Performance Compared to Wind Tunnel Measurements

Arthur Stuck, Stephen Turnock, Neil Bressloff

Ship Science Report No 130

January 2004

SSSU 130
ISSN 0140 3818

University
of Southampton

# School of Engineering Sciences

## *Ship Science*

**An Evaulation of the RANS Method for the Prediction of Steady Ship Rudder Performance Compared to Wind Tunnel Measurements**

**Arthur Stuck, Stephen Turnock, Neil Bressloff**

**Ship Science Report No 130**

**January 2004**

**Abstract**

Using Reynolds-Averaged Navier-Stokes (RANS) methods detailed investigations are carried out, focussing on different cases of steady rudder flow. During the introducing two-dimensional cases the hybrid meshing scheme is optimized for rudder use, and grid parameters are studied. Also the effect of solver settings and the capabilities of implemented turbulence models are investigated. Two different verification strategies are applied and compared – eventually concentrating on parametric grid independence studies, rather than using global systematic grid refinement.

The three-dimensional investigation concentrates on the all-movable square tipped NACA0020 rudder tested within the large wind tunnel of the University of Southampton ($3.5 \cdot 2.5m$). This case is modeled in free stream as well as within the wind tunnel, using the Spalart-Allmaras and the $k$-$\epsilon$ RNG turbulence models. The numerical prediction of the tunnel blocking effect is investigated. In particular, the tip vortex flow with its effect on the rudder performance is studied and how this is influenced by the numerical dicretization. Here the span-wise distribution of the normal force coefficient is an informative tool. The three-dimensional grids consist of between $2 \cdot 10^6$ to $4.5 \cdot 10^6$ cells.

Verification studies are carried out and results are validated against experimental data as far as available. Generally, the lift prediction turned out to show closer agreement with experimental benchmark (within 10%) than the drag, which used to deviate more (within 30%). The prediction of the tip vortex shows the expected characteristics; The tip peak in the span-wise normal force distribution is found to be captured well.

# Contents

# Nomenclature

| | |
|---|---|
| $\mathbf{A}$ | transformation matrix (from global to rudder fixed coordinate system) |
| $A_R$ | rudder aspect ratio |
| $C_d$, $c_d$ | drag coefficient (3D or 2D) |
| $C_{d_{di}}$ | induced drag coefficient (only 3D) |
| $C_{d_f}$, $c_{d_f}$ | frictional drag coefficient (3D or 2D) |
| $C_{d_{pv}}$, $c_{d_{pv}}$ | viscous pressure drag coefficient (3D or 2D) |
| $C_G$ | correction factor for grid extrapolation |
| $C_l$, $c_l$ | lift coefficient (3D or 2D) |
| $c$ | rudder chord length |
| $c_n$ | normal force coefficient (2D) |
| $c_p$ | pressure coefficient |
| $D$ | experimental data |
| $D$, $d$ | drag force (3D or 2D) |
| $D_{di}$ | induced drag force (3D) |
| $D_f$, $d_f$ | frictional drag force (3D or 2D) |
| $D_{pv}$, $d_{pv}$ | viscous pressure drag force (3D or 2D) |
| $G_\nu$ | production of turbulent viscosity (Spalart-Allmaras turbulence model) |
| $\mathbf{g}$ | acceleration vector of free fall |
| $k$ | kinetic energy ($k$-$\epsilon$ turbulence model) |
| $L$, $l$ | lift force (3D or 2D) |
| $Ma$ | Mach number |
| $n$ | normal force (2D) |
| $n_{bndr}$ | number of nodes perpendicular to rudder surface within boundary mesh |
| $n_{chord}$ | chord-wise number of nodes |
| $n_{wake}$ | number of nodes along the wake extension of boundary mesh |
| $P_G$ | estimated order of approximations |
| $P_k$ | production of turbulence ($k$-$\epsilon$ turbulence model) |
| $P_{th}$ | theoretical order of approximations |
| $p$ | pressure |
| $p_0$ | reference pressure |
| $Re$ | Reynolds number |
| $R_G$ | ratio of changes in grid dependent solutions |
| $r_G$ | global grid refinement factor |
| $S$ | computational results (simulation) |
| $s$ | rudder span |
| $t$ | time |
| $U$ | uncertainty value |
| $U_{G1}$ | estimated grid dependent uncertainty |

| | |
|---|---|
| $U_\infty$ | free stream velocity |
| $u_i$ | velocity components |
| $u_n$, $u_t$ | velocity components (tangential and normal) |
| $u_p$ | velocity within boundary layer |
| $u^+$ | non-dimensional velocity within boundary layer |
| $\overline{u_i' v_j'}$ | turbulent stress tensor |
| $\mathbf{V}$ | velocity vector |
| $v_0$ | inlet velocity |
| $x_i$ | global coordinates |
| $\mathbf{x_0}$ | location vector of rudder pivot |
| $(x, y, z)$ | global coordinate system (in CFD models) |
| $(x', y', z')$ | rudder fixed coordinate system |
| $Y_\nu$ | destruction of turbulent viscosity (Spalart-Allmaras turbulence model) |
| $y_p$ | wall distance within boundary layer |
| $y^+$ | non-dimensional wall distance within boundary layer |
| $z_{floor}$ | z-coordinate of tunnel floor |
| $z_{hi}$, $z_{lo}$ | z-bounds of sectional rudder slices |
| $z_{roof}$ | z-coordinate of wind tunnel roof |
| $z_{tip}$ | z-coordinate of rudder tip |
| $\alpha$ | angle of incidence |
| $\alpha_{UR}$ | under-relaxation factor |
| $\Gamma$ | circulation |
| $\Delta X$ | grid line distance |
| $\delta$ | error value |
| $\delta_{BL}$ | boundary layer thickness |
| $\delta_{G1}$ | estimated grid dependent error |
| $\delta_{RE,G1}$ | first order grid dependent error (Richardson extrapolation) |
| $\epsilon$ | destruction of kinetic energy ($k$-$\epsilon$ turbulence model) |
| $\epsilon_{ij}$ | difference between solution i and j |
| $\zeta_x$ | x-vorticity |
| $\mu$ | molecular viscosity |
| $\mu_t$ | turbulent viscosity (eddy viscosity) |
| $\nu$ | kinematic viscosity |
| $\tilde{\nu}$ | turbulent kinematic viscosity (Spalart-Allmaras turbulence model) |
| $\rho$ | fluid density |
| $\tau_{ij}$ | viscous tension tensor |
| $\tau_w$ | wall friction |
| $\phi_P$, $\phi_{nb}$ | scalar variable (own or neighbouring cell) |

# Glossary

| | |
|---|---|
| CFD | computational fluid dynamics |
| CPU | central processing unit |
| ITTC | International Towing Tank Conference |
| NACA | National Advisory Committee for Aeronautics |
| quad | quadrilateral |
| RANS | Reynolds Averaged Navier Stokes |
| RNG | Renormalization Group ($k$-$\epsilon$ turbulence model variant) |
| SIMPLE | Semi-Implicit Method for Pressure-Linked Equations |
| SMR | scaled mass residuals |
| tri | triangular |
| 2D | two-dimensional |
| 3D | three-dimensional |

# Chapter 1

# Introduction

## 1.1 Background

Airfoil and rudder flows are a traditionally well examined application of computational fluid dynamics (CFD) – so a lot of different approaches have been applied in order to predict these flows. Also many experiments have been conducted so that a large pool of experimental data is available to validate the computational results.

Several experimental series have been carried out by Molland and Turnock, testing a wall mounted NACA0020 spade rudder in the large $3.5m \cdot 2.5m$ wind tunnel of the University of Southampton [8]. This arrangement refers to rudders which are typically used for large commercial vessels.

## 1.2 Goal and Procedure

The NACA0020 spade rudder, tested within the wind tunnel of the University of Southampton, is the eventual 3D flow case selected for the RANS investigation. To obtain reliable solutions rather than random results it is important to understand the influence of decisive CFD parameters. Hence two-dimensional investigations are carried out in advance, focussing on two NACA0012 cases taken from literature. Then, preparing the final three-dimensional computations, 2D slices of the considered 3D setup are investigated.

## 1.3 Rudder Performance

The steering moment, induced by the rudder force, traditionally provides the ability to control the movements of a vessel. Hence it is one of the most important contributions to ship manoeuvering. In reality the rudder interacts with propeller, hull and possibly other hydrodynamically influential parts – altogether forming a complex interaction system, which describes the manoeuvering characteristics of a vessel. In this work only the rudder is investigated, operating under so-called free stream conditions without any influences from hull or propulsor. As experimental studies cannot reproduce ideal free stream conditions the blocking effects, resulting from the limited wind tunnel extensions, are considered here.

Rudder performance is mainly valued on the base of the integral forces. The resulting vector force acts at the centre of pressure. It is traditionally subdivided in two scalar components: the drag, acting in free flow direction, and the lift, acting perpendicular to it. The following

non-dimensional coefficients are standardized according to ITTC:

$$C_d = \frac{D}{\rho/2\, A_R U_\infty^2} \tag{1.1}$$

$$C_l = \frac{L}{\rho/2\, A_R U_\infty^2} \tag{1.2}$$

using the fluid density $\rho$, the projected rudder area $A_R = s \cdot c$ and the angle of incidence $\alpha$. The two-dimensional coefficients are described by lower case letters, made non-dimensional by the rudder chord length $c$:

$$c_d = \frac{d}{\rho/2\, c\, U_\infty^2} \tag{1.3}$$

$$c_l = \frac{l}{\rho/2\, c\, U_\infty^2} \tag{1.4}$$

The lift is mainly pressure dominated. The contribution originating from viscous skin friction is mostly negligible.

The normal force is defined perpendicular to the rudder chord. Here only the 2D coefficient is used – either in real two-dimensional cases or for thin sectional slices of the 3D spade rudder. The normal force coefficient is

$$c_n = \frac{n}{\rho/2\, c\, U_\infty^2} \tag{1.5}$$

The drag force can be subdivided into three components: the frictional drag, the viscous pressure drag, and the induced drag. The frictional drag results from the shear force acting on the rudder surface. It is mainly a phenomenon of the diffusive boundary layer flow – its non-dimensional coefficients are

$$C_{d_f} = \frac{D_f}{\rho/2\, A_R U_\infty^2} \tag{1.6}$$

$$c_{d_f} = \frac{d_f}{\rho/2\, c\, U_\infty^2} \tag{1.7}$$

The viscous pressure drag results from boundary layer effects and viscous losses in the flow. In contrast to ideal inviscid fluids, the pressure is unable to recover on the rear part of the rudder resulting in pressure differences between leading and trailing edge regions. This effect is growing at rising angles of incidence, finally ending up in extended detachments. Coefficients are defined as

$$C_{d_{pv}} = \frac{D_{pv}}{\rho/2\, A_R U_\infty^2} \tag{1.8}$$

$$c_{d_{pv}} = \frac{d_{pv}}{\rho/2\, c\, U_\infty^2} \tag{1.9}$$

The induced drag is restricted to three-dimensional cases and results from a limited spanwise rudder extension. According to classical theory of the lifting line the infinite vortex leaves the rudder at its free tip, and is washed downstream with the flow. This free vortex induces an additional drag component, which also exists in ideal inviscid flow. Its non-dimensional form is

$$C_{d_{di}} = \frac{D_{di}}{\rho/2\, A_R U_\infty^2} \tag{1.10}$$

# Chapter 2

# Applied CFD Methods

## 2.1 Governing Equations

Fluid flow in general is described by the Navier-Stokes equations, consisting of the equations of conservation for mass and momentum. According to Peric and Ferzinger [4] the differential form for the conservation of momentum is

$$\frac{D\mathbf{V}}{Dt} = \mathbf{g} - \frac{1}{\rho}\nabla p + \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \tag{2.1}$$

using the transport theorem for the substantive derivative:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + (\mathbf{V}\cdot\nabla) \tag{2.2}$$

The scalar conservation of mass is described by

$$\frac{\partial \rho}{\partial t} + \frac{\partial u_j}{\partial x_j} = 0 \tag{2.3}$$

The velocity vector $\mathbf{V}$ consists of the components $u_i$, $p$ represents the pressure and $\rho$ the density of the fluid. The kinematic viscosity is $\nu$, and $\mathbf{g}$ describes the acceleration vector of gravity. In this work the flow is assumed to be steady and incompressible; The density is set constant – therefore all time-dependent derivatives in the equations for momentum and continuity can be neglected.

These equations form a set of coupled nonlinear vector equations which cannot be solved analytically for practical applications – only a few solutions are known for academic problems. For that reason numerical approaches are used as described in the following sections.

## 2.2 RANS Equations

By averaging the Navier-Stokes equations the instantaneous values are subdivided into a mean and a fluctuating component: These are the Reynolds Averaged Navier-Stokes (RANS) equations which are in their differential form for incompressible flow:

$$\frac{\partial\left(\rho\overline{u}_i\right)}{\partial x_i} = 0 \tag{2.4}$$

$$\frac{\left(\partial\rho\overline{u}_i\right)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho\overline{u}_i\overline{u}_j + \rho\overline{u_i'u_j'}\right) = -\frac{\partial\overline{p}}{\partial x_i} + \frac{\partial\overline{\tau}_{ij}}{\partial x_j} \tag{2.5}$$

They contain the viscous tensions

$$\overline{\tau}_{ij} = \mu \left( \frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i} \right) \tag{2.6}$$

In addition to the laminar stress these averaged equations contain turbulent stresses, described by the turbulent stress tensor $\overline{u_i' u_j'}$. Thus, six new unknown variables are introduced, making turbulence modeling necessary as they cannot be solved directly.

## 2.3 Turbulence Modeling

To solve the averaged Navier-Stokes equations the assumption of the eddy viscosity hypothesis provides an additional relationship between the mean variables, some additional turbulence parameters and the entries of the Reynolds tensor:

$$-\overline{u_i' u_j'} = \mu_t \left( \frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i} \right) - \frac{2}{3} \rho \, \delta_{ij} \, k \tag{2.7}$$

It contains the turbulent kinetic energy $k = (u_i u_i)/2$. The eddy viscosity $\mu_t$ has a similar effect to the diffusion of viscosity, but it is a feature of local turbulence, and assumes isentropic turbulence. So the six entries of the Reynolds tensor are reduced to one scalar value $\mu_t$ – however, further modeling is required to determine its value. For that reason zero, one and two equation turbulence models are used. Here the one equation Spalart-Allmaras turbulence model and the two-equation $k$-$\epsilon$ models are applied as they are described in [1].

### 2.3.1 Standard $k$-$\epsilon$ Turbulence Model

The standard $k$-$\epsilon$ model is one of the most popular turbulence models. Based on the assumption of the eddy viscosity, the additional turbulent viscosity is determined as

$$\mu_t \approx \rho c_\mu \frac{k^2}{\epsilon} \tag{2.8}$$

Two equations, one for the kinetic energy k, the other for its dissipation rate $\epsilon$ have to be solved:

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho \overline{u}_j k) = \frac{\partial}{\partial x_j} \left( \mu \frac{\partial k}{\partial x_j} \right) - \frac{\mu_t}{\sigma_k} \frac{\partial k}{\partial x_j} + P_k - \mu \overline{\frac{\partial u_i'}{\partial x_k} \frac{\partial u_i'}{\partial x_k}} \tag{2.9}$$

The second right-hand term of equation 2.9, describing the transport of $k$ by fluctuation, has already been modeled. Also the production term of $k$ requires modeling:

$$P_k \approx \mu \left( \frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i} \right) \frac{\partial \overline{u}_i}{\partial x_j} \tag{2.10}$$

The dissipation rate $\epsilon$ can be obtained, solving the following transport equation:

$$\frac{\partial}{\partial t}(\rho \epsilon) + \frac{\partial}{\partial x_j}(\rho \overline{u}_j \epsilon) = \frac{\partial}{\partial x_j} \left( \frac{\mu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x_j} \right) + c_{\epsilon 1} P_k \frac{\epsilon}{k} - c_{\epsilon 2} \rho \frac{\epsilon^2}{k} \tag{2.11}$$

$c_\mu$, $c_{\epsilon 1}$, $c_{\epsilon 2}$, $\sigma_k$ and $\sigma_\epsilon$ are empirical coefficients defined within the solver. According to Date [3] some of the major weaknesses of the $k$-$\epsilon$ turbulence model for rudder and airfoil applications are:

- $k$-$\epsilon$ is known for over-predicting the turbulent kinetic energy, particulary in terms of boundary layer flows in areas of stagnation, detachment and reattachment. In rudder and airfoil application this affects the flow around leading and trailing edges.
- Flow separation is often under-predicted heavily by standard $k$-$\epsilon$. In terms of rudder flow the prediction of viscous pressure drag $c_{d_{pv}}$ and stall is often far from reality.
- If combined with wall functions (section 2.3.4), $k$-$\epsilon$ automatically inherits those weaknesses. Results are often made worse, particularly concerning stagnation, separation and reattachment phenomena.

### 2.3.2 RNG $k$-$\epsilon$ Turbulence Model

The RNG derivative of the standard $k$-$\epsilon$ turbulence model includes additional statistical approaches to improve some of its weaknesses. Mainly small scale turbulence statistics are implemented to modify the eddy viscosity hypothesis. The transport equations for $k$ and $\epsilon$ contain additional coefficients, as described in equation 2.12 and 2.13:

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho \bar{u}_j k) = \frac{\partial}{\partial x_j}\left(\alpha_k \mu \frac{\partial k}{\partial x_j}\right) - \frac{\alpha_k \mu_t}{\sigma_k}\frac{\partial k}{\partial x_j} + P_k - \mu \overline{\frac{\partial u_i'}{\partial x_k}\frac{\partial u_i'}{\partial x_k}} \tag{2.12}$$

$$\frac{\partial}{\partial t}(\rho \epsilon) + \frac{\partial}{\partial x_j}(\rho \bar{u}_j \epsilon) = \frac{\partial}{\partial x_j}\left(\frac{\alpha_\epsilon \mu_t}{\sigma_\epsilon}\frac{\partial \epsilon}{\partial x_j}\right) + c_{\epsilon 1}^* P_k \frac{\epsilon}{k} - c_{\epsilon 2}\rho\frac{\epsilon^2}{k} \tag{2.13}$$

Anyway, the RNG modification still suffers from the general assumptions of the eddy viscosity hypothesis (equation 2.7) that it is based on.

### 2.3.3 Spalart-Allmaras Turbulence Model

Recently, the turbulence model proposed by Spalart and Allmaras has enjoyed a growing popularity – particularly for turbulent airfoil applications. According to the solver documentation [1] it calculates the eddy viscosity solving a modified transport equation for the turbulent kinematic viscosity:

$$\frac{\partial}{\partial t}(\rho \tilde{\nu}) + \frac{\partial}{\partial x_i}(\rho \tilde{\nu} \bar{u}_i) = G_\nu + \frac{1}{\sigma_{\tilde{\nu}}}\left[\frac{\partial}{\partial x_j}\left((\mu + \rho\tilde{\nu})\frac{\partial \tilde{\nu}}{\partial x_j}\right) + C_{b2}\rho\left(\frac{\partial \tilde{\nu}}{\partial x_j}\right)^2\right] - Y_\nu \tag{2.14}$$

$G_\nu$ is the production of turbulent viscosity, and $Y_\nu$ is its destruction occurring in the near-wall region. $\sigma_{\tilde{\nu}}$ and $c_{b1}$ are default constants defined within the solver. Rather than using equation 2.8 with the classical length scales, the turbulent viscosity is obtained by $\mu_t = \rho\tilde{\nu} f_{v1}$ and a viscous damping function $f_{v1}$. Turbulent production and destruction are determined by applying further modeling, based on the magnitude of vorticity.

Wall boundary conditions are simulated directly by calculating the wall shear stress from equation 2.17 while the mesh is fine enough to resolve the laminar sub-layer. If this is not the case wall functions are applied as described in section 2.3.4.

### 2.3.4 Wall-functions

The no-slip wall boundary layer itself is multi-layered, consisting of a laminar sub-layer, a viscous-turbulent layer and an outer layer dominated by turbulence. Resolving this complex structure directly requires a very fine mesh and high computational resources. Instead, on the base of experiments the following velocity profile is assumed close to the wall:

$$u^+ = \frac{1}{\kappa} \ln y^+ + C \tag{2.15}$$

where typically $\kappa = 0.4$ and $C = 5.5$. The non-dimensional wall distance and wall shear stress are given by equation 2.16 and 2.17, respectively.

$$y^+ = \frac{\Delta y_p}{\nu} \sqrt{\frac{\tau_w}{\rho}} \tag{2.16}$$

$$u^+ = \frac{u_p}{\sqrt{\tau_w / \rho}} \tag{2.17}$$

For $y^+ \geq 11.63$ the boundary layer is considered as turbulent, consequently equation 2.15 is applied – however, it is recommended by Date [3] to keep the non-dimensional wall distance between $30 < y^+ < 500$. The finer the grid resolution within this interval, the better flow gradients are resolved. Hence, he suggests to set the optimal $y^+$ parameter close to the lower bound of the proposed interval.

The assumptions made for the derivation of wall functions are not valid for stagnation and separation points and in areas of detached flow – nevertheless they are applied here, making a potentially major contribution to the modeling error.

## 2.4 Boundary Conditions

In order to solve the governing equations a complete description of the boundary conditions of the solution domain is necessary. Those can be divided into Dirichlet and von Neumann conditions. In the following descriptions are provided as far as they are of interest in this work.

#### Velocity Inlet

For the velocity inlet the velocity components $u_i$ and the turbulence variables $k$ and $\epsilon$ or the turbulent kinematic viscosity $\tilde{\nu}$ are defined explicitly as Dirichlet conditions. To minimize their disturbing influence on the flow field the inlet conditions are positioned distant from the rudder itself. The inlet definition of the turbulence variables is less influential; Assuming free stream inlets, those are defined by the small non-negative value $10^{-4}$.

#### No-slip Wall

The no-slip wall boundary condition assumes a zero wall-tangential velocity $u_t = 0$, additionally setting the normal gradient of the normal velocity to zero:

$$\left( \frac{\partial u_n}{\partial n} \right)_{wall} = 0$$

The applied turbulence models define the flow to be fully turbulent. An economic approach to simulate the multi-layered no-slip walls are wall functions, described in section 2.3.4. Alternatively it is possible to resolve these boundary layers directly by a non-dimensional wall distance $y^+ \leq 1$.

#### Slip Wall

The slip wall boundary condition only regards the convective displacement effect of a wall, neglecting the wall friction. This boundary condition is realized, defining a zero shear stress on the concerned wall. Thus, there are no viscous boundary layers, and there is no need for high grid resolutions next to the walls.

#### Pressure Outlet

For the pressure outlet boundary condition the pressure itself is defined explicitly (Dirichlet condition), while the other variables are described by a gradient (von Neumann conditions). The pressure value is set relative to the operating pressure – in this case a zero difference. Having defined the pressure, mass conservation is provided by the velocity-pressure correction.

## 2.5 Obtaining a Solution

In this work computations are carried out using the commercial flow solver *Fluent* ver. 6.0.20. The following sections describe some of the crucial features of the numerical solution process, referring to the solver documentation [1].

### 2.5.1 Remarks on Discretization and Linearization

A *finite volume* approach is chosen to discretize the governing equations into algebraic ones. This system of equations has to be linearized to be solved numerically. Throughout this work only *second order schemes* are applied for the governing equations. According to Peric and Ferzinger [4] they represent a good compromise between numerical effort and precision as long as convergence can be obtained on the used grid.

### 2.5.2 Iteration Process

#### Elementary Solver Settings

Depending on the case the *2D* or *3D* solver is selected in *Fluent*. As only steady cases are investigated all time dependent derivatives in the governing equations become zero. The *segregated* solver is chosen, solving the equations sequentially. The solution procedure is based on the following scheme:

1. Fluid properties are updated, based on the current solution.
2. The $u_x$, $u_y$, and $u_z$ momentum equations are each solved in turn using current values for pressure and face mass fluxes, in order to update the velocity field.
3. Since the velocities obtained in Step 2 may not satisfy the continuity equation locally, a "Poisson-type" equation for the pressure correction is derived from the continuity equation and the linearized momentum equations. This pressure correction equation is then solved to obtain the necessary corrections to the pressure and velocity fields and the face mass fluxes such that continuity is satisfied.
4. Transport-equations for scalar turbulence variables are solved using the previously updated values of the other variables.
5. A check for convergence of the equation set is made.

Throughout this work the *SIMPLE* pressure-velocity correction is applied. The cycle is repeated until convergence is reached.

### Scaled Residuals

The applied convergence criterion is based on the scaled mass residuals (SMR):

$$R^{\phi} = \frac{\sum_{\text{cells } P} |\sum_{\text{nb}} a_{\text{nb}}\phi_{\text{nb}} + b - a_P\phi_P|}{\sum_{\text{cells } P} |a_P\phi_P|} \tag{2.18}$$

$\phi$ describes the mass flow rate for each cell and $a_{P,nb}$ are the centre or neighbour coefficients. Scaling is taken care of by the denominator – in case of the mass residual the largest absolute value of the mass residual during the first five iterations is chosen. The solver documentation [1] demands residuals to drop for at least three orders of magnitude – here, according to El Moctar [7], they are desired to drop for five orders to minimize the convergence error. For some of the more complicated 3D cases also four orders are accepted. But as this is just an automatic stopping criterion depending on the quality of the initial guess (which it is scaled by), convergence is checked additionally for every single case on the base of force and residual convergence behaviour.

### Under-relaxation

The computed change $\Delta\phi$ from iteration to iteration of the flow variable is reduced by the under-relaxation factor $\alpha_{UR}$:

$$\phi = \phi_{\text{old}} + \alpha_{UR}\,\Delta\phi \tag{2.19}$$

By under-relaxation the iteration process can often be stabilized for the trade-off of a higher computing time. By default $\alpha_{UR_{mass}} = \alpha_{UR_{mom}} = 0.3$ is used for the equations of mass and momentum. As only second order schemes are applied they are reduced further in case of poor convergence.

### Multi-grid Methods

In order to reduce the required number of iterations and CPU time *multi-grid* methods are used. Computations are carried out on different grid resolution levels. The method is based upon the idea that the global (low-frequency) error, existing on a fine mesh, can be represented on a coarse mesh – there it again becomes accessible as local (high-frequency) error and can be corrected by the solver. The default coarsening and refinement cycles are applied: *V* for the pressure and *Flexible* for the momentum equations. In *Flexible* the convergence behaviour on the current grid level is recognized by an implemented logic that controls the further multi-grid cycles. So it is determined flexibly when and how often each level is visited, whereas in the *V* and *W* cycles the traversal pattern is defined explicitly.

### Parallel Computation

As particularly the 3D cases require huge computational resources, the calculations are carried out in parallel on the *Iridis Beowulf* cluster of the University of Southampton. It consists of 404 processors, has 192Gb of memory and 8.5Tb of local disk storage – in practice, resources are limited as many users are working simultaneously. For parallel computation the solution domain has to be divided into several subdomains. This is taken care of by an automatic partitioning algorithm, implemented in *Fluent*: The *Bisect* scheme is applied recursively to the

parent domain and then to the child subdomains – every time splitting one domain into two. Also uneven numbers of subdomains can be created if an uneven number of partitions is split one more time than the rest. Due to a lack of time this work does not contain any parallelization studies focussing on parallel efficiencies.

## 2.6 Verification and Validation Procedure

Estimating the quality of a numerical solution requires determination of the uncertainties of the results, generally called verification study. Then, during the following validation process, results are compared against any kind of benchmark, mostly experimental data.

In this work two different procedures are applied: Firstly, the verification method proposed by Stern [10] is introduced, requiring systematical grid refinement in all dimensions. Secondly, grid-independence studies focussing on single grid parameters are described. The latter approach can be used more easily on unstructured hybrid grids – the first method is only applicable under certain restrictions as described below.

### 2.6.1 Stern's Methodology

#### Errors and Uncertainties

In order to estimate accuracy and quantify deviations a comparison between numerical results and experimental data is necessary. As different sources of errors are involved a direct comparison is not possible – a method of verification and validation is necessary to figure out and identify the different types of errors. Both, experimental data $D$ and computational results $S$ suffer from errors $\delta$, which are statistically encapsuled by the uncertainty interval $\pm U$ in 95 cases out of 100. Generally, components of uncertainties are added by the $||L||_2$ norm:

$$U_{tot}^2 = U_1^2 + U_2^2 + \cdots + U_n^2$$

Error components, which can be estimated by sign and magnitude, simply are to be added:

$$\delta_{tot} = \delta_1 + \delta_2 + \cdots + \delta_n$$

Stern suggests dividing the simulation error $\delta_S$ and uncertainty $U_S$ into a modeling part $\delta_{SM}$ ($U_{SM}$) and a numerical part $\delta_{SN}$ ($U_{SN}$). The first consists of insufficiencies originating from the mathematical description of the physical flow, simplified implementation and turbulence modeling. The second originates from discretization, interpolation, rounding off and incomplete iteration. The decisive idea is to declare the numerical error as grid dependent (so that it can be accessed by grid convergence studies), whereas the modeling error is treated as grid independent. Ideally, having estimated the numerical error, the modeling error can be obtained by comparing the experimental benchmark, the numerical solution and its grid dependent numerical error. Stern's procedure assumes some essential preconditions to determine the modeling error – some are mentioned below:

- Reliable experimental data must be available, ideally providing own errors and uncertainties.
- Only in case of asymptotic grid convergence the method can be carried out efficiently based on three grids.

- Here the iterative error, which is part of the numeric error, is neglected as it is demanded that the SMR has dropped by at least four orders of magnitude. This is necessary since the iterative error does not converge over grid refinement, though it is declared to be part of the numerical error.

### Grid Extrapolation

As described above, a grid convergence study is required to extrapolate a grid independent solution. At least three grids are necessary to estimate the convergence behaviour. Those have to be refined for every coordinate direction, each time using the refinement factor

$$r_G^{i,j,k} = \frac{\Delta X_A}{\Delta X_B} \tag{2.20}$$

$\Delta X_A$ and $\Delta X_B$ are the grid line distances for the base mesh and the refined one, respectively. Grid convergence is estimated on the base of the ratio of changes in the solution:

$$R_G = \frac{\varepsilon_{21}}{\varepsilon_{32}} = \frac{S_2 - S_1}{S_3 - S_2} \tag{2.21}$$

$S_{1,2,3}$ describes the solutions on the fine (1), medium (2) and coarse (3) grid. The following cases can be subdivided:

- Converging condition: $0 < R_G < 1$
- Oscillatory condition: $R_G < 0$
- Diverging condition: $1 < R_G$

According to the Richardson extrapolation described by Peric and Ferzinger [4] the first order error of the fine grid

$$\delta_{RE,G1} = \frac{\varepsilon_{21}}{r_G^{P_G} - 1} \tag{2.22}$$

describes the difference between the computation at high resolution and the grid independent solution, neglecting all terms of higher order. $r_G$ is the grid refinement factor. The estimated order of accuracy is defined as

$$P_G = \frac{\ln(\varepsilon_{31}/\varepsilon_{21})}{\ln r_G} \tag{2.23}$$

A correction factor

$$C_G = \frac{r_G^{P_G} - 1}{r_G^{P_{th}} - 1} \tag{2.24}$$

is introduced, using $P_{th}$ for the theoretical order of the applied approximations. For $C_G \approx 1$ the obtained solutions are within asymptotic range – in this case sign and magnitude of the error

$$\delta_{G1} = C_G \, \delta_{RE,G1} \tag{2.25}$$

and the uncertainty

$$U_{G1} = |(1 - C_G) \, \delta_{RE,G1}| \tag{2.26}$$

can be estimated. For $C_G > 1$ equation 2.22 under-predicts the error, $C_G < 1$ indicates its over-prediction. If $C_G$ turns out to be far from one, a lack of confidence in equation 2.24 and 2.25 is justified. Then only the uncertainty should by calculated from

$$U_{G1} = |C_G \, \delta_{RE,G1}| + |(1 - C_G) \, \delta_{RE,G1}| \tag{2.27}$$

In practice, this procedure can only predict a grid independent solution in case of asymptotic grid convergence. For oscillatory behaviour more results are necessary to resolve the grid dependent oscillations and proof possible oscillatory convergence. This requires enormous grid generation efforts, and eventually it is not clear if the plotted oscillations can be extrapolated reliably to an grid independent solution.

### 2.6.2 Parametric Grid Independence Studies

An alternative to the procedure proposed by Stern are grid-independence studies, focussing on single grid parameters. This approach is more flexible, especially, when it comes to complex grid configurations. With respect to unstructured hybrid grids it is expected to be more practicable and easier to apply. Furthermore, it does not suffer from any kind of restrictions, as those necessary to apply Stern's procedure.

The variation of single grid parameters is carried out in order to find a parameter independent solution. In contrast to Stern's procedure, here no grid extrapolation is carried out. The procedure is rather regarded as a way of efficient grid optimization. Stern's systematic refinement approach often worsens the grid in certain areas where the global refinement factors are unsuitable. By parametric studies the mesh can be accustomed flexibly, addressing the right parameters. So it is possible to generate and optimize a mesh, avoiding useless wastage of cells.

## 2.7 Numerical Diffusion

Additionally to the physical diffusion the computation is prone to the so-called numerical diffusion, manifesting by an increase of diffusive transport in the Navier-Stokes equations. Numerical diffusion is caused by the finite approximations used for discretization of the Navier-Stokes equations. Especially low order schemes like first order upwind differences amplify this effect. It is also increased by coarse grid resolutions – so areas of high gradients need appropriately refined meshes.

The over-prediction of diffusive transport results in smeared flow fields of smoothed variable distributions and flattened gradients. According to El Moctar [7], in case of airfoil and rudder applications, the lift tends to be under-predicted by rising numerical diffusion. The drag, which usually is one order of magnitude smaller, is affected strongly in most cases: Its viscous pressure drag component $c_{d_{pv}}$ is known to suffer from heavy over-prediction. The frictional drag $c_{d_f}$ normally is under-predicted at an increasing numerical diffusion.

## 2.8 Remarks on Grid Generation and Modeling

The hybrid grids are generated in *Gambit* ver. 2.0.4. In 2D the outer mesh consists of either triangles or unstructured quadrilaterals, the inner boundary mesh uses only quadrilateral cells to resolve the boundary layer flow. The 3D grids are generated by the so-called *Cooper* tool. This extrudes the 2D base grids into the third dimension, defining the 3D node distribution by meshing the volume edges.

All rudder and airfoil cases work under steady state conditions and are regarded as incompressible at $Ma < 0.3$. They are set up according to the Reynolds number provided by the experimental cases. The 3D spade rudder (NACA0020 at $Re = 8.0 \cdot 10^5$) is modeled $s = 1.0m$ in span and $c = 0.667m$ in chord. This refers to the original dimensions of the experimental case. On behalf of comparability all cases are scaled to the same chord length. With the default

fluid properties $\rho = 1.225kg/m^3$, $\mu = 1.4607kg/ms$ and $\nu = \mu/\rho$ the free stream velocity is determined as

$$v_0 = \frac{Re \cdot \nu}{c} \tag{2.28}$$

The rudder geometry is defined by a curve following the points of the analytic NACA thickness distribution. The trailing edge is modeled as a sharp edge.

In the free stream cases the coordinate system $(x, y, z)$ is defined on the trailing edge with the x-axis pointing downstream in the direction of the chord (figure B.11(a)). For the tunnel cases the rudder is turned within the solution domain around the pivot at $0.3c$ from the leading edge. The origin is defined on the point of the trailing edge at $\alpha = 0°$, the x-axis pointing downstream along the tunnel symmetry line. The $(x, y, z)$ coordinate system is independent from the actual angle of incidence, while the $(x', y', z)$ system is rudder fixed (figure B.11(b)). The z-axis points in span-wise direction towards the free tip, assuming the root of the rudder as the $z = 0$ plane.

# Chapter 3

# Two-dimensional Rudder Flows

For the initial investigations three flow cases of 2D NACA sections are selected from literature: Firstly, the NACA0012 at $Re = 7.6 \cdot 10^5$ which has been investigated experimentally by Michos [5]. This case is chosen as it provides 2D experimental data at a similar Reynolds number to the 3D spade rudder, which is concentrated on later. Secondly, the NACA0012 at $Re = 6.0 \cdot 10^6$ is selected from Abbott and v. Doenhoff [2] as it is one of the most reliable and accepted two-dimensional cases in literature. Finally, a 2D slice of the 3D spade rudder (NACA0020, $Re = 8.0 \cdot 10^5$) is investigated in a brief parametric study: It focusses on single grid parameters to prepare the following 3D cases which are based on this 2D mesh.

## 3.1 NACA0012, $Re = 7.6 \cdot 10^5$

At the beginning of the two-dimensional analysis it has been decided to investigate the capabilities of the hybrid grids created in *Gambit*. A grid extrapolation is carried out following Stern's procedure (section 2.6.1). These convergence studies are normally conducted on structured grids, using systematic all-dimensional refinement. However, the extrapolation is applied to hybrid grids here, under certain restrictions as described below. These studies are based on the free stream NACA0012 at $Re = 7.6 \cdot 10^5$, which has been experimentally investigated by Michos [5]. The Spalart-Allmaras turbulence model is chosen due to its special capabilities of resolving the boundary layers directly, without wall functions. Grid convergence studies are carried out for the angles of incidence $\alpha = 0°$ and $\alpha = 10°$.

### 3.1.1 Grid Generation and Boundaries

In a hybrid grid topology an "inner" C-grid of quadrilaterals enwraps the rudder, while the "outer" mesh consists of unstructured triangles. In *Gambit* the boundary layer tool allows to create a boundary mesh within an existing block topology. No extra block definition is necessary in advance. The "inner" mesh is defined by the chord-wise node distribution, the number of nodes perpendicular to the surface, a first spacing next to the wall and an expansion ratio.

A grid convergence study is carried out for this case, refining the mesh successively by the global refinement factor $\sqrt{2}$ (section 2.6.1). Strictly speaking, this grid extrapolation is only declared for structured grids which are to be refined systematically in every coordinate direction. Here this procedure is applied to hybrid grids. It is directly applicable to the "inner" boundary layer mesh (C-grid) – the outer mesh of triangles has to be adjusted by the *Size Function* tool, which is implemented in *Gambit*: It allows the definition of a first spacing and an expansion ratio going out from any source geometry it is defined on. So the cell size can be controlled

reasonably, whereas aspect ratio, cell alignment and the exact node positions are up to the automatic meshing tool. For that reason "systematic" grid refinement can only be realized in a restrictive way outside of the boundary layer mesh. Therefore, the enwraping C-grid has to be set up wide enough to cover the area of highest flow gradients. The thickness of the boundary mesh is declared at least three times the corresponding thickness of a flat plate boundary layer. This is necessary as the flow around a rudder is much more complex – deviating further, the higher the angle of incidence. According to Munson and Young [9], the local thickness of the $0.99v_0$ boundary layer in a turbulent flat plate flow is estimated by

$$\delta_{BL} = \frac{0.37x}{(Re(x))^{1/5}} \tag{3.1}$$

The non-dimensional wall distance is set to $y^+ \leq 1$ for the coarse grid, adapting this parameter according to the grid refinement procedure. The use of wall functions would have dominated the convergence studies as results would have been way too sensitive to changes in $y^+$.

The block-topology is shown in figure B.1(a) and B.1(b). Different from the other 2D cases, here two extra blocks are necessary to get additional control over the wake mesh. The automatic extension of the boundary mesh into the wake does not work appropriately in this case: By default it extends with the same first spacing and expansion ratio in x-direction which are chosen perpendicular to the rudder surface. This works quite well in the other cases using wall functions – here it has to be modified as, without wall functions, the grid resolution perpendicular to the rudder surface is unsuitable for the wake extension in x-direction. Therefore the extra blocks on either side of the wake allow an independent meshing of the boundary mesh extension.

Except from not using wall functions for the no-slip walls, the boundary conditions are the same as for the NACA0012 at $Re = 6.0 \cdot 10^6$ (section 3.2.1). Again the angle of incidence is defined by the inlet boundary conditions, instead of turning the rudder within the domain.

### 3.1.2 Verification and Validation

The grid refinement study is carried out in accordance with the procedure recommended by Stern (section 2.6.1). Convergence is estimated on the base of the integral rudder forces. Investigated are the zero-lift case at $\alpha = 0°$ and the pressure dominated case at $\alpha = 10°$. The three grids are refined successively by $r_G = \sqrt{2}$. Some grid parameters are mentioned in table 3.1. As the

| $n_{cells}$ | coarse | medium | fine |
|---|---|---|---|
| $n_{chord}$ | 100 | 141 | 200 |
| $n_{wake}$ | 20 | 28 | 40 |
| $n_{bndr}$ | 45 | 64 | 90 |
| $\sum bndr$ | 10,800 | 21,623 | 43,200 |
| $\sum total$ | 44,012 | 65,266 | 102,888 |

Table 3.1: Grid refinement, NACA0012, $Re = 7.6 \cdot 10^5$

geometry is symmetric only the values for one side of the rudder are mentioned – the summations are given for the whole domain.

In the $\alpha = 0°$ case convergence is estimated on the base of drag prediction. For $\alpha = 10°$ lift and drag predictions are taken into consideration (table 3.2). As the scaled residuals have dropped between four to five orders of magnitude the numerical error is neglected. Forces tend to converge earlier as observed during the calculations. According to Stern grid convergence is reached for $R_G < 1$, which is obtained for both drag cases at $\alpha = 0°$ and $\alpha = 10°$. Concerning

| | | coarse | medium | fine |
|---|---|---|---|---|
| $\alpha = 0°$ | $c_d \cdot 100$ | 1.22 | 1.27 | 1.33 |
| $\alpha = 10°$ | $c_d \cdot 100$ | 2.53 | 2.68 | 2.72 |
| | $c_l$ | 1.013 | 1.019 | 0.996 |

Table 3.2: Rudder performance over grid refinement, NACA0012, $Re = 7.6 \cdot 10^5$

the lift prediction at $\alpha = 10°$ the convergence behaviour is oscillating. More grids would be necessary to get a closer view of the oscillation and determine its bounds. This is regarded as rather uneconomic, exceeding the limits of this work. However, it is possible to go through the first steps of Stern's procedure as described in section 2.6.1. The theoretical order of the method is $P_{th} = 2$ as second order schemes are applied for all variables. Results are presented in table 3.3.

| | $\alpha = 0°$ | $\alpha = 10°$ | |
|---|---|---|---|
| | $c_d$ | $c_d$ | $c_l$ |
| $R_G$ | 0.938 | 0.267 | -3.492 |
| $P_G$ | 0.184 | 3.814 | - |
| $\delta_{RE,G1} \cdot 10^4$ | -0.937 | -1.454 | - |
| $C_G$ | 0.066 | 2.750 | - |
| $U_G \cdot 100$ | - | 0.065 | - |

Table 3.3: Grid extrapolation, NACA0012, $Re = 7.6 \cdot 10^5$

At this point the procedure hardly seems to be able to give reliable information about the convergence process and the quality of the numerical solution. For $\alpha = 0°$ the correction factor $C_G = 0.066$ turns out to be that far from one that it does not seem to make sense to calculate the uncertainty with it. The lift prediction at $\alpha = 10°$ does not even converge over grid refinement. For the drag the correction factor $C_G = 2.75$ is also far from one – anyway, it is decided to continue the procedure and calculate an uncertainty value $U_G$ for that case. But the very small value hardly seems to be realistic, considering the level of confidence corresponding to $C_G = 2.75$.

In terms of lift prediction results are not that far from experimental benchmark (table 3.4), which comes with an experimental uncertainty of 10%. Here it seems to be more comprehensive to take a look at the development of the chord-wise pressure distribution, rather than judging on the base of force convergence according to Stern's procedure. Figure B.2 shows the chord-

| | $\alpha = 0°$ | $\alpha = 10°$ | |
|---|---|---|---|
| | $c_d$ | $c_d$ | $c_l$ |
| CFD | 0.0133 | 0.0272 | 0.996 |
| exp. | 0.0080 | 0.0310 | 0.950 |
| $\Delta$ [%] | 66.0 | -12.2 | 4.9 |

Table 3.4: Comparison against experimental benchmark, NACA0012, $Re = 7.6 \cdot 10^5$, fine mesh

wise pressure distribution for the different grid resolutions. Astonishingly, there are hardly any differences around the pressure and suction peak. But it can be observed that the fine grid slightly predicts less pressure along the pressure side and more along the suction side, resulting in a smaller integral lift force. The highest local deviations between the individual resolutions are observed around the trailing edge, where the finer grids predict a small suction peak. This

has not been expected and does hardly seem to be realistic. Potentially the high aspect ratio of the first cell layers on the rudder surface due to the fine boundary resolution ($y^+ < 1$) causes numerical problems. Cells of high aspect ratio tend to decouple the conservation equations, falsifying the results, and causing convergence problems. Particularly if the flow is not one-dimensional anymore, as assumed along boundary layers, cell alignment relative to the direction of the flow becomes more important. E.g. areas of detachment and stagnation do not have this dominating direction of the boundary layer flow anymore, and calculations can become more sensitive to cell aspect ratios in these areas.

### 3.1.3 Discussion of Procedure

In this test case Stern's grid extrapolation procedure is applied to hybrid grids – which does not directly correspond to the refinement approach demanded by the theory. However, the area of highest gradients around the rudder is systematically refinable and also the outer area can be adjusted. But potentially the poor results of the grid extrapolations are also caused by other reasons. At first, it seems to be rather crude to verify a convergence process on the base of three values. Easily a so-called convergence could be interpreted, though in reality the three values are part of an oscillation process, or they are just a sort of random behaviour pretending to be convergent. The other way round an oscillatory behaviour which is convergent could be titled as divergent, if the wrong values were selected. So the higher the complexity of the case, and the more single effects (which could be affected differently by the refinement) are superposed, the harder it becomes to recognize convergence behaviour by this procedure. Individual effects, e.g. the local suction peak at the trailing edge, can easily spoil the convergence, interpreted on the base of the integral forces. Here the procedure which might have worked in case of a flat plate flow does not. In terms of the integral lift, which is pressure dominated, it seems to be more comprehensive to decide on the base of the pressure distribution whether or not grid convergence is satisfactory.

In this work the global grid refinement procedure does not prove a proper tool as it already fails in 2D application. Furthermore, the effort of resolving the boundary layers without wall functions is immense and increases the grid-size heavily. It also worsens a lot of problems connected to the hybrid meshing scheme: The high aspect ratio of cells in the boundary mesh makes it harder to connect the outer mesh, particularly in the wake. With respect to the following 3D investigation at a very similar Reynolds number this scheme brings even more disadvantages: High aspect ratios of the boundary cells are expected to be even more critical in 3D application. The free tip would cause further difficulties as its boundary layer could hardly be meshed without wall functions by the applied *Cooper* scheme (compare 3D meshing scheme, section 4.2.2). A mixture of different boundary treatments would be necessary – then again cell geometry around the tip would become rather inhomogeneous and the computation of the 3D flow around the tip would suffer from high cell aspect ratios.

Due to these problems wall functions are applied in the following cases. Grids are optimized focussing on individual parameters, rather than by one global grid refinement study. Here the combination of meshing scheme and verification approach has turned out to be insufficient for this special application.

## 3.2 NACA0012, $Re = 6.0 \cdot 10^6$

This case, experimentally investigated by Abbott and v. Doenhoff [2], is one of the most comprehensive two-dimensional airfoil studies in literature. Experiments have been conducted at

a relatively high Reynolds number, compared to the other cases, about one order of magnitude higher. Hence, this case is less critical in terms of laminar-turbulent transition and provides good conditions to estimate the quality of the numerical results.

### 3.2.1 Grid Generation and Boundaries

Initially, hybrid grids are generated using triangular cells (*tri*) for the outer mesh and quadrilateral cells (*quad*) for the boundary mesh next to the rudder (figure B.3, left). It is observed that the number of cells can be reduced significantly by changing to an outer mesh, consisting of unstructured quads (right). So the transition from the boundary mesh to the unstructured outer mesh can be realized in a smoother way. Firstly, cell topology in and outside the boundary layer is the same, and secondly, the quad meshing scheme seems to be able to adapt the aspect ratio of the outer cells gradually to that of their neighbours, located within the structured boundary layers. The latter capability has a crucial advantage over the triangular cells: Especially triangular cells suffer from skewness, the more their shape deviates from the equilateral optimum. Connecting triangular cells to a boundary mesh of quads often requires extensive cell refinement to reduce skewness – though such grid resolution has not been intended originally. An example is given in figure B.3. At the end of the extension of the boundary layer mesh, cell aspect ratios grow as the expansion ratio is greater than one. The tri scheme tries to connect low-skewed cells at the intersection, which automatically ends up in very fine grid resolutions. Similar problems occur every time the quadrilateral boundary layer cells have high aspect ratios at the intersection of inner and outer mesh. The unstructured quad scheme can cope with that problem more easily, saving lots of cells in those areas.

An advantage of unstructured over structured meshing becomes clear in the outer mesh, distant from the rudder: The numerous useless cells, which are a consequence of matching structured meshing, can be avoided while cell aspect-ratios remain small at the same time. Additionally, solver convergence and the quality of the finite discretization benefit from the almost square shape of the majority of the unstructured quads.

Both, tri and quad grids, use identical boundary layer meshes and are controlled by mostly the same *Size Functions*. A survey of the main grid parameters is given in table 3.5.

| | bndr. mesh | | | outer mesh | total mesh |
|---|---|---|---|---|---|
| | $n_{chord}$ | $n_{bndr}$ | $n_{cells}$ | $n_{cells}$ | $n_{cells}$ |
| tri | 200 | 50 | 25,000 | 69,278 | 94,278 |
| quad | 200 | 50 | 25,000 | 39,925 | 64,925 |

Table 3.5: Hybrid grid, NACA0012, $Re = 6.0 \cdot 10^5$

Following Date's recommendations [3], the outer boundaries are located 15c away from the profile in both dimensions. As many angles of incidence are to be investigated, it is decided to vary the boundary conditions, rather than turning the rudder within the solution domain – this way only one grid has to be created. A velocity inlet is modeled on both upwind edges, while downstream pressure outlets are assumed. For $\alpha = 0°$ pressure outlets are declared on the edges either side of the rudder. Its surfaces are defined as no-slip walls. A more detailed description of the individual boundary conditions is given in section 2.4.

According to Date's investigations, calculations are carried out using the $k$-$\epsilon$ RNG turbulence model. His RNG calculations of the same case have shown better results than the standard $k$-$\epsilon$ calculations did, especially, with respect to lift prediction at higher angles of incidence –

though stall prediction was still poor. Thus the RNG model is chosen, not at least on behalf of comparability.

### 3.2.2 Results and Discussion

This discussion is based on the results of the *quad* mesh, which is believed to be superior in this case. With respect to the later 3D computations its ability of economical meshing is highly appreciated.

#### Convergence

At first, convergence behaviour is investigated to ensure numerically converged solutions. The automatic stopping criterion is defined by $SMR < 10^{-5}$ while computations are limited to 2000 outer iterations. A reliable, almost monotonous convergence behaviour is observed up to an angle of incidence of $\alpha \leq 15°$. For $\alpha = 10°$ the convergence history is plotted in figure B.4. The computation is carried out on a single node, using the solver settings described in section 2.5.2. In case of the *tri* grids the convergence process takes little longer, as more cells are involved. The fact that each triangular cell consists of one forth less faces can obviously not compensate for the bigger grid size.

For higher angles of incidence convergence behaviour worsens and tends to become oscillatory – but as long as the SMR drops for four orders of magnitude and lift and drag converge reasonably, the results are accepted. An example is given for the $\alpha = 15°$ case (figure B.11): According to the history of $SMR$ and forces , convergence is still judged to be sufficient, though it is not that straight-forward as for $\alpha = 10°$.

#### Turbulence Modeling and Wall Functions

Calculations are carried out using the $k$-$\epsilon$ RNG turbulence model and wall-functions for the near-wall boundary layers. Though the considered range of incidence is restricted, the turbulence dependent contribution to the modeling error rises, the higher the angle of incidence. As only one grid has to cover all calculations, a compromise is to be found for the non-dimensional wall distance $y^+$. Its behaviour over the angle of incidence $\alpha$ is shown in figure B.6. Due to stagnation and separation it can be observed that the wall-functions fail, as the flow on the rear suction side decelerates; $y^+$ tends to zero, leaving the interval it has originally been restricted to. Anyway, the assumptions made for wall functions and the $k$-$\epsilon$ turbulence model are invalid for points of separation and stagnation, and for areas of detachment (Peric and Ferzinger [4]), partly explaining the deviations at higher angles of incidence.

#### Integral Forces

Generally, the lift prediction shows a good correlation with experimental and numerical data it is compared against. As described above, only converged data is taken into consideration, so results are plotted for $\alpha \leq 15°$. The lift curve shows the same development as Date's results do; But, compared against experimental benchmark, it shows higher over-prediction than Date's results do at small angles ($\alpha < 10°$), while for $\alpha > 10°$ it comes closer to the measurements, showing less under-prediction.

The drag shows constant over-prediction of the same quality but of higher quantity than Date's results. For $\alpha < 6°$ it comes close to experimental benchmark. Here results can probably profit from the effect that due to numerical diffusion (section 2.7) the viscous pressure drag

$c_{d_{pv}}$ increases about one order of magnitude more than the frictional drag $c_{d_f}$ decreases (El Moctar [7]). For small angles the latter dominates, and both effects compensate each other. At higher angles both components are of almost the same value, and from that point on the total drag is over-predicted heavily. For higher angles $c_{d_f}$ is almost constant, while $c_{d_{pv}}$ grows rapidly. The effect of laminar-turbulent transition is believed to be of low influence here, as for $Re = 6 \cdot 10^6$ the transition length is shorter than for the cases of lower Reynolds numbers. Selected results are given in table 3.6, comparing the prediction of both meshing schemes and their deviation from experimental data.

| | *tri* | | | | *quad* | | | |
|---|---|---|---|---|---|---|---|---|
| $\alpha$ | $c_d$ | $c_l$ | $\Delta c_d$ | $\Delta c_l$ | $c_d$ | $c_l$ | $\Delta c_d$ | $\Delta c_l$ |
| [°] | [-] | [-] | [%] | [%] | [-] | [-] | [%] | [%] |
| 0 | 0.0086 | 0.0000 | 11.1 | - | 0.0086 | 0.0000 | 11.1 | - |
| 6 | - | - | - | - | 0.0121 | 0.6408 | 10.3 | 7.3 |
| 10 | 0.0190 | 1.0527 | 17.8 | 5.27 | 0.0201 | 1.0284 | 24.6 | 2.8 |

Table 3.6: Grid dependent deviation from measurements, NACA0012, $Re = 6.0 \cdot 10^6$

**Pressure Distribution**

The chord-wise pressure distribution is examined at $\alpha = 8°$ and compared against Date's results (figure B.8). Most sensitive are the areas of highest gradients, especially around the leading edge, where the grid has to be resolved sufficiently. Generally, the distribution matches well, only the chord-wise cell distribution around the leading edge could have been refined little further ($n_{chord} = 200$). But as the suction peak is very sharp, its impact on the integral forces, which are matching well for the regarded angle of incidence, is limited. The grid-resolution around the trailing edge seems to be adequate.

## 3.3 NACA0020, $Re = 8.0 \cdot 10^5$

Preparing the three-dimensional investigation, in 2D some parametric grid independence studies are conducted in advance. Focussing on a 2D slice of the 3D spade rudder, the base grid for the later 3D *Cooper* mesh is to be optimized. This kind of parametric study is labour intensive – but particularly with respect to the following 3D cases it is important to mesh economically.

Again a hybrid meshing scheme is applied. Due to the experiences of the former cases, the unstructured *quad* scheme is used. This investigation is carried out for the $\alpha = 10°$ angle of incidence. The chord-wise and layer-wise parametric grid studies are conducted in a kind of iteration loop, always assuming the best value for the other parameter, which has been determined before.

### 3.3.1 Chord-wise Grid Study

The first parameter to be investigated is the chord-wise cell distribution. Parameter convergence is valued by the integral lift and drag coefficients. The chord-wise number of nodes $n_{chord}$ is counted only for one side, so the overall number around the whole rudder has twice the value. In figure B.14(a) the lift shows monotonous convergence, while the drag seems to be fluctuating – however, the four calculated values are not able to resolve the drag convergence process fine enough, which might be oscillating. As the lift gradient decreases distinctly and the drag matches

the value which is eventually obtained by $n_{chord} = 200$, $n_{chord} = 125$ is chosen for the following investigations. Anyway, the differences in lift and drag are less than 1%, so it is believed to have resolved the flow sufficiently by this parameter.

### 3.3.2 Layer-wise Study of Boundary Mesh

The layer-wise study of the boundary layer mesh focusses on the node distribution perpendicular to the rudder surface. It has an impact on the solution in two ways: The boundary layer mesh uses the same expansion ratio perpendicular to the rudder surface as for its extension into the wake, going out from the trailing edge. Firstly, this reduces one degree of freedom – but secondly, it ensures that cells at the outer corners of the wake-extension are almost square (figure B.12(c)). This eases the unstructured meshing around these corners effectively. As mentioned earlier, large cell aspect ratios of the structured boundary layers are to be avoided. The adjoining unstructured mesh tries to connect cells of small aspect ratio, which automatically causes an unevenness in the mesh close to the rudder. It also limits the thickness of the boundary mesh, as a wide structured boundary mesh automatically means higher aspect ratios in its outer layers. However, the thickness of the boundary layer mesh is defined three times the thickness of the corresponding flat plate boundary layer $\delta_{BL}$ (equation 3.1). The first cell size is optimized according to the non-dimensional wall distance $y^+$. A brief study has been conducted to find a proper value to cover the whole range of incidence, similarly as explained in section 3.2.2. Having defined the first spacing, the layer-wise number of nodes $n_{bndr}$ is varied perpendicular to the surface, adapting the expansion factor. As shown in figure B.15 lift and drag show asymptotic convergence over such refinement. $n_{bndr} = 25$ is regarded the best compromise between accuracy and grid size – however, as 3D calculations have already been undertaken at $n_{bndr} = 20$, the latter value is chosen. This convergence level also seems to be acceptable, keeping in mind that variation over the plotted interval is very small: approx. 1.5% in drag and 0.5% in lift, so that the flow seems to be resolved properly by this parameter.

### 3.3.3 Comparison of Boundary Conditions

The 3D NACA0020 has been tested within the large wind-tunnel of the University of Southampton, which is $2.5m$ in height and $3.5m$ in width. The rudder is $c = 0.667m$ in chord and $s = 1m$ in span, so that a certain blocking effect, increasing at higher angles of incidence, is expected: The actual wind tunnel velocity (dynamic pressure) is measured upstream of the rudder to control the required velocity. To round the rudder geometry within the wind-tunnel the flow is perturbed, channeled and accelerated. Additionally the boundary layers, growing out from the tunnel walls, cut the cross-section, and the flow is accelerated further. Compared against the 3D case, the 2D slice will over-predict the blocking effect (the free tip vortex decreases pressure differences between pressure and suction side and decreases the 3D lift). Thus the blocking effect can be studied efficiently in 2D as tendencies are expected to be over-predicted – but generally the 3D case should show the same behaviour.

For this investigation a grid of $n_{chord} = 125$ and $n_{bndr} = 20$ has been chosen, based on the earlier results. Firstly, the rudder is modeled in free stream conditions, using the same boundary conditions as described in section 3.2.2.

In the next case the rudder profile is located within a horizontal section of the 3D tunnel. To change the angle of incidence this time the rudder has to be turned, instead of adjusting the inlet boundary conditions. Defining a slip wall on the tunnel walls the skin friction is neglected, while only the convective displacement effect of these walls is taken into consideration.

Finally, these walls are set to no-slip walls, which is expected to accelerate the flow even further. The mesh is shown in figure B.9. The thickness of the boundary layers in CFD has to match the real thickness in the model-tests. As no measurements are available, the thickness is estimated by the lift reduction recognisable on the span-wise lift distribution of the 3D measurements: This curve decreases at the root of the rudder, where it works in reduced velocity within the boundary layer of the tunnel floor. This way a boundary layer thickness on the tunnel walls $\delta \approx 100mm$ is estimated at the position of the rudder. Equation 3.1 associates a flat plate length of $x = 9.67c \approx 10c$ with that thickness. This upstream length seems to provide a realistic boundary layer thickness at the x-coordinate of trailing edge, and is regarded as to be long enough to minimize the influence of the inlet boundary condition. Figure B.10 shows the development of the $0.99v_0$ contour line on the tunnel walls. On the rudder surfaces this line does not represent the thickness of the local boundary layer, but it is recognizable that the pressure field of the rudder influences the development of the contour lines on the tunnel walls. On the base of the contour plot B.10 the thickness of the $0.99v_0$ boundary layer is estimated $\delta \approx 90mm$ on the tunnel walls at the x-coordinate of trailing edge. So the CFD boundary layer seems to match the thickness of the physical boundary layer.

As presented in figure B.16(a), the lift increases slightly when the tunnel is modeled. Introducing the no-slip conditions on the side-walls of the tunnel this effect becomes little stronger. Differences are growing the higher the angle of incidence – at $\alpha = 20°$ calculations predict the lift in the tunnel to be relatively lower than in free stream, which seems to be non-physical. This is traced back to worsening convergence and the failure of wall functions.

The drag dependence on the boundary conditions is not predicted in the expected way. The relationship between the slip and the no-slip tunnel cases seems to be realistic but, compared to free stream, the drag within the tunnel is predicted relatively lower, which seems to be nonphysical. This observation might be explained by the different ways of modeling the angle of incidence. Altering the angle by changing the boundary conditions means to apply only one grid – consequently the wake mesh is always the same, independently from the angle of incidence. Turning the rudder within the tunnel means using different meshes for every angle of incidence. The wake refinement is done by *Size Functions*, attached to the tunnel symmetry-line. For $\alpha \neq 0°$ this is different from the direction of the chord, carrying the refining *Size Functions* in free stream. Though the expansion ratio is close to one, and the wake is refined generously this is expected to cause the problem. Details of the tunnel mesh are presented in figure B.12 – the different strategies in grid generation and the corresponding coordinate systems for the tunnel and free stream cases are shown in figure B.11(a) and B.11(b). Except from this difference both cases are set up identically, using the same boundary mesh and *Size Functions*. But as the drag prediction is very sensitive, slight differences in the wake mesh might be decisive. This guess is backed by the fact that the differences mainly result from pressure-drag $c_{d_{pv}}$. The viscous part $c_{d_f}$ only shows little differences and reacts on tunnel blocking in the expected way (figure B.16(b)). This issue is worth further investigation, e.g. on a constant mesh – but would exceed the available time. Instead, the pressure distributions of free stream and tunnel case are plotted and compared in figure B.18. Differences can mainly be observed along the first half of the chord – particularly the suction peak is more developed in the accelerated tunnel flow. Along the aft part (in the area of the main grid differences) pressure differences caused by tunnel blocking and the different grid generation approaches seem to be negligible.

### 3.3.4 Comparison of Turbulence Models

Here the standard $k$-$\epsilon$, the $k$-$\epsilon$ RNG and the Spalart-Allmaras turbulence models are compared and results are plotted in figure B.17(a). For lower angles of incidence $k$-$\epsilon$ predicts the highest lift, followed by $k$-$\epsilon$ RNG and Spalart-Allmaras. At higher angles Spalart-Allmaras is the only one capable of predicting a major decrease in lift due to detachments. This is also confirmed by the prediction of flow fields at $\alpha = 15°$ (figure B.13): Spalart-Allmaras predicts detachments of negative velocity gradients at $0.61c$ from the leading edge. For the same case $k$-$\epsilon$ and its RNG modification do hardly show any negative velocity gradients on the rear suction side – the flow only stagnates without detaching.

The drag prediction at small angles is the highest for $k$-$\epsilon$, followed by Spalart-Allmaras and RNG, lying relatively close to each other. Then again, at higher angles of incidence, Spalart-Allmaras is the only model predicting detachments of larger extends, resulting in a rapid increase of drag.

### 3.3.5 Results

This 2D study is carried out in order to find an economic way of meshing, and to give a better understanding of the two-dimensional case before the 3D spade rudder is investigated. As no experimental benchmark is available for the 2D slice it is only possible to compare the results of the parametric studies against each other. Eventually the following conclusions are drawn:

- In chord-wise direction $n_{chord} = 125$ nodes (on each rudder side) seem to be sufficient to resolve the flow appropriately. Lift and drag show a good level of convergence.
- The study of node distribution in the boundary layer mesh, perpendicular to the rudder surface and along the wake extension, shows asymptotic convergence. The optimum is assumed at $n_{bndr} = 25$ nodes.
- A tunnel blocking effect can be observed in the 2D results. Lift prediction is affected by convective displacement and diffusive skin friction of the tunnel walls – both increasing the lift. The drag behaviour, which turned out to be highly sensitive, cannot be interpreted that easily. Further investigation is recommended to figure out its dependencies in detail.
- At small angles of incidence predictions of the different turbulence models are very similar. For higher angles the Spalart-Allmaras turbulence model is the only one predicting detachment of larger extents.

# Chapter 4

# Three-dimensional Rudder Flows

## 4.1 Description of Experimental Flow Case

This flow case is one of the wind-tunnel setups investigated by Molland and Turnock [8]. Here the all-movable spade-rudder No. 2 (described by Turnock [11]), operating in free stream (i.e. without propeller), is selected. It consists of a NACA0020 profile, is rectangular in planform and has a square tip. Its root is mounted to the floor of the wind tunnel. The model has a chord length of $c = 0.667m$ and extends $s = 1.0m$ in span-wise direction. To trigger turbulent flow, a roughness strip of $0.15mm$ is glued on either side of the rudder, $0.05c$ behind the leading edge.

The experiments have been conducted within the large wind tunnel of the University of Southampton, which has a cross section of $3.5m \cdot 2.5m$. Horizontally the boundaries are located $1.75m$ from the chord, while the free tip has a vertical clearance of $1.5m$. The free stream velocity within the tunnel of $v_0 = 20m/s$ refers to $Re = 8.0 \cdot 10^5$.

## 4.2 3D Grid Generation

### 4.2.1 Development of 3D Grid Generation Strategy

At the beginning it is concentrated on conventional hybrid meshing schemes – i.e. encapsulating the rudder in a boundary layer mesh of hexahedra which is connected to an outer grid of tetrahedra. It is observed that the two-dimensional difficulties of connecting the structured boundary layers to the outer unstructured mesh even become worse in 3D. Additionally, new difficulties arise since the quadrilateral base faces of the cells within the structured boundary layers tend to have large aspect ratios, particularly along the rudder edges.

Instead, to make use of the quasi-two-dimensional span-wise geometry and flow, it is decided to set up a prismatic cell structure by expanding the 2D base grid from the root towards the tip of the rudder. Along the span-wise z-direction an expansion ratio is defined. Cells are clustered near the tip to catch the tip vortex and next to the root (tunnel floor) to resolve the viscous boundary layer. Beyond the tip, for $z > z_{tip} = 1.0m$, the prisms are extruded a little further into the solution domain. Thus a change of meshing schemes in areas of difficult geometry, like the emerging tip edges, can be avoided. The rest of the domain is meshed by an unstructured, tetrahedral scheme.

Alternatively, the prismatic cell topology is applied throughout the whole solution domain. This brings a loss of freedom for local cell-clustering on the one hand – on the other hand, the difficult transition from prisms to tetrahedra can be avoided altogether. During the grid generation process it is observed that the fully prismatic grids can manage with even less cells

than grids of changing structure can: The latter is strongly affected by cell skewness in the transition area, which locally requires finer grid resolution again. Mostly the saved cells in the outer areas are more than compensated. Hence, in case of this relatively simple geometry, it is regarded as most efficient to apply the *Cooper* meshing scheme throughout the whole domain. Not at least grids are simplified essentially, making preprocessing easier and faster.

### 4.2.2 Applied 3D Grids

As described before, the *Cooper* scheme is applied throughout the whole solution domain. The 2D grid, optimized in section 3.3, is used as a base mesh to be extruded. In order to investigate the 3D wind tunnel blocking, the rudder is modeled within the tunnel as well as in the free stream. This also matches with the 2D procedure so that comparisons can be drawn. The final grid topologies are shown in figure C.1 for the tunnel and free stream boundaries. The rudder is enwraped in one block, extending from the base $z = z_{floor} = 0$ to the tip $z = z_{tip} = 1m$. Beyond the tip plane, for $z > z_{tip}$, the adjacent block leaves a hollow, which is filled up by two inner blocks, replacing the rudder. The subdivision of this inlay is provided on behalf of meshing as described later. Every block is meshed by prisms using the *Cooper* tool. All block interfaces are matched.

The parameters for the two-dimensional base grids are already available, as determined in section 3.3. For the three-dimensional case it is necessary to refine the grid around the tip with respect to the tip vortex and next to root, which is mounted to a no-slip wall. This requires expansion ratios going out from the tip in high and low z-direction, and from the root of the rudder (tunnel floor). Some grid parameters are mentioned in table 4.1.

| $n_{nodes}$ | free stream | wind tunnel |
|---|---|---|
| base mesh | 20,420 | 18,970 |
| main tip mesh | 1,830 | 1,785 |
| aft tip mesh | 186 | 193 |
| z-distribution $(z_{floor} \dots z_{tip})$ | 80 | 80 |
| z-distribution $(z_{tip} \dots z_{roof})$ | 60 | 50 |
| $\sum total$ | 2,979,760 | 2,565,000 |

Table 4.1: 3D grids, NACA0020, $Re = 8.0 \cdot 10^5$

In 3D also the tip face of the rudder has to be meshed. It is subdivided into two faces, serving as base faces for the inner blocks. Here the advantage of unstructured meshing can be exploited as shown in figure C.3. Along the larger front face of the tip a boundary layer mesh is created, adjacent to the outer boundary layer mesh. This provides a smooth transition from the outer mesh, across the outer boundary layer mesh, across the inner boundary layer mesh, to the interior area. The inside is filled up by unstructured quadrilaterals. The aft triangle is defined as an extra block and meshed separately. Here cell aspect ratios of the adjacent outer mesh are relatively small so that a smooth transition, without an inner boundary layer mesh, can be realized. Within structured C-grids especially this aft part of the tip face causes problems of high cell skewness.

## 4.3 Boundary Conditions

**Free Stream**

In the free stream case any disturbing influences of the boundaries are to be minimized. Following recommendations of Chau and El Moctar [6], the boundaries are located ten chord-length horizontally and six chord-length vertically away from the rudder. A velocity inlet is applied, defining the angle of incidence by the velocity components of the flow. As in 2D, only one mesh covers the whole range of incidence. Downstream a pressure outlet is defined. The side faces are set to velocity inlet or pressure outlet, assuming the same definitions of flow variables as for the upstream inlet and the downstream outlet, respectively. For the zero angle of incidence both side faces are defined as pressure outlets. The ground face ($z = 0$), the rudder is attached to, is set to a no-slip wall. Its viscous boundary layer is modeled by wall functions, using a non-dimensional wall distance $y^+ \approx 40$. The top of the domain is modeled as a pressure outlet. The rudder surfaces are no-slip walls.

**Wind Tunnel**

For the tunnel case the solution domain extends $10c$ up- and downstream, going out from the trailing edge. The upstream length of the domain also affects the thickness of the boundary layers on the tunnel walls. As described during the 2D study in section 3.3.3, it is tuned according to equation 3.1. The dimensions of the tunnel cross-section is described in section 4.1. Its floor $z = 0$ and side-walls are modeled as no-slip walls with $y^+ \approx 40$. The tunnel roof is defined as a slip wall – this saves resources as the viscous boundary layer on the wall is neglected and does not need resolution. Upstream a velocity inlet is defined, downstream a pressure outlet. The angle of incidence is realized by turning the rudder – hence, the direction of the inlet velocity is constant for all tunnel cases. Figure C.2 shows a plot of the 3D wind tunnel mesh.

## 4.4 Convergence

For the three-dimensional investigation the automatic convergence criterion is defined by the SMR, dropping for four orders of magnitude. Here again the SMR criterion causes a numerical overhead as forces converge much earlier – figure C.4 shows the convergence histories for the free stream case at $\alpha = 10°$, using the $k$-$\epsilon$ RNG turbulence model. Generally, 3D results are judged to be converged as long as the SMR drops for about four orders of magnitude and forces show reasonable convergence, which is proved for every individual case. As this work does not particularly focus on optimization of computational efforts no comparisons of parallel efficiencies are drawn.

## 4.5 Verification and Validation

### 4.5.1 3D Grid Refinement Study

The *Cooper* scheme, applied to extrude the 2D base mesh, allows an independent refinement study of node distribution in z-direction. Parameter convergence is estimated on the base of the integral rudder forces as well as on the span-wise distribution of the normal force coefficient. Especially the tip flow, which is particularly three-dimensional, is expected to be sensitive to the span-wise node distribution. This short investigation is based on three free stream grids, successively increasing the number of nodes in z-direction by $\sqrt{2}$, while the first cell size for the

non-dimensional wall distance $y^+$ is kept constant at $z = z_{floor} = 0$ and $z = z_{tip} = 1m$. The base grids are identical to the standard 2D free stream cases.

The z-refinement study is carried out using the $k$-$\epsilon$ RNG turbulence model at $\alpha = 10°$. The prediction of lift and drag seems to be quite independent from the variation in span-wise node distribution, the way it is carried out here (table 4.2).

| | coarse | medium | fine | exp. |
|---|---|---|---|---|
| $n_{nodes,z}$ | 99 | 140 | 198 | - |
| $n_{nodes,tot}$ | 2,106,252 | 2,979,760 | 4,241,520 | - |
| $C_d$ | 0.0545 | 0.0540 | 0.0542 | 0.0440 |
| $C_l$ | 0.5280 | 0.5304 | 0.5288 | 0.5107 |
| $\Delta_{C_d}[\%]$ | +23.8 | +22.8 | +23.2 | - |
| $\Delta_{C_l}[\%]$ | +3.4 | +3.9 | +3.6 | - |

Table 4.2: Grid parameters and results of z-refinement study, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$

Also the span-wise plot of the normal force coefficient (figure C.6) hardly shows any differences. While the first cell sizes are kept constant, only the expansion ratios in z-direction profit from the increasing number of nodes. Apparently these ratios are sufficiently small, as the results are almost constant. Except from the non-dimensional wall distance $y^+$, the first cell size is also kept constant on behalf of cell geometry around the tip. The constant cell size provides a homogeneous, almost cubic cell shape around the tip, and ensures numerical coupling of the governing equations. With respect to computations at higher angles of incidence the medium grid is selected for the following calculations.

### 4.5.2 Integral Forces

Calculations are carried out for free stream and tunnel boundary conditions. Both cases are investigated using the $k$-$\epsilon$ RNG and the Spalart-Allmaras turbulence model. Results are plotted in figure C.5.

#### Lift

The lift prediction (figure C.5) qualitatively captures the blocking effect of the modeled wind tunnel – hence, the lift increases due to an increasing effective Reynolds number. This behaviour can be observed for both turbulence models. Generally, the $k$-$\epsilon$ RNG computations predict a higher lift than the corresponding Spalart-Allmaras cases do. One more time it is noticed that the latter turbulence model tends to predict a decrease in lift earlier. Over the considered range of incidence the tunnel case, which geometrically comes closest to the real experimental case, over-predicts the lift. Therefore, the closer agreement between the free stream and the experimental curves is explained by the superposition of two countereffects: The lower free stream velocity decreases the lift relatively, which is compensated by the tendency of over-predicting the lift. This over-estimation is observed for both turbulence models – it has already been noticed during the 2D investigations of the NACA0012 at $Re = 6.0 \cdot 10^6$ (particularly at small angles) in section 3.2, and is also confirmed by Date's studies of the same case [3]. This stands in contrast to the general observation of lift under-prediction, influenced by numerical diffusion.

### Drag

It is noticed (figure C.5) that the tunnel drag is predicted higher than in free stream. In 3D the increase in frictional and viscous pressure drag due to blocking is expected to be partly compensated by a reduced induced drag: Compared to the free stream case the closer tunnel roof restricts the tip flow, and with it the drag component induced by the tip vortex.

For low angles of incidence Spalart-Allmaras calculates the higher drag, at $\alpha = 10°$ both turbulence models almost calculate the same values and for $\alpha = 20°$ Spalart-Allmaras predicts the higher drag again. At $\alpha = 20°$ results are dominated by the different predictions of the viscous pressure drag as Spalart-Allmaras can cope with detached flow much better than $k$-$\epsilon$ does. Generally, the over-prediction of drag is relatively higher and less sensitive to changes in boundaries and turbulence modeling than it is the case for the lift. It must be kept in mind that the pressure drag is caused by small differences in the pressure-field – additionally to modeling errors it is highly sensitive to numerical errors.

## 4.5.3 Normal Force Distribution

The span-wise distribution of the normal force coefficient $c_{n(z)}$ is determined using an integration code, which is briefly introduced in appendix A. The coefficient is determined for thin sectional rudder slices, plotted over the rudder span (figure C.7). At the root $c_{n(z)}$ is reduced as the rudder acts within the boundary layer of the tunnel floor – next to the tip it is affected by the tip vortex. Closer to the tip behind the local minimum $c_{n(z)}$ rises again, almost reaching the maximum normal force. Beyond this peak it drops rapidly. For growing circulation at higher angles of incidence the $c_{n(z)}$ peak becomes wider and rather blunt.

Figure C.6 and C.8 demonstrate how the $c_{n(z)}$ prediction depends on turbulence modeling. Generally, the lift is predicted smaller by Spalart-Allmaras, but particularly the tip peak only reaches half the $k$-$\epsilon$ RNG height.

The influence of boundary modeling on the tip peak (figure C.8) shows the expected behaviour. While the normal force distribution over most of the span is higher for the tunnel case, its tip peak is predicted lower than in free stream. This shows that moving the tunnel roof closer to the tip slightly decreases the development of the tip vortex. Moving the roof closer and closer towards the tip would ideally end up in a 2D case, which should not show any free tip vortex.

## 4.5.4 Fields of Pressure and Vorticity

Confirming the results of the previous sections, plots of the pressure fields and tip-vortices help giving a more detailed view of the solutions. The vortex-plots (figure C.11) display a selected iso-surface of x-vorticity (here $\zeta_x = \frac{\partial u_y}{\partial y} - \frac{\partial u_z}{\partial z} = -500$) as well as the geometrical rudder surfaces. The plots are coloured according to the distribution of the pressure coefficient $c_p$.

At $\alpha = 0°$ the vortices separate from the trailing edge and on either side of the tip. They are simply caused by the displacement effect of the rudder. The plot for $\alpha = 10°$ shows the reunion of the suction and pressure side vortices to one tip vortex behind the rudder. At $\alpha = 20°$ the vortices grow large due to the higher circulation $\Gamma \sim \alpha$. They are also predicted to unite earlier in the wake.

In figure C.9 slices at characteristic span-wise positions are selected to plot the chord-wise distribution of the pressure coefficient $c_p$: the maximum lift position, the local minimum before the tip and the tip maximum itself; Additionally, the 2D distribution is provided. Compared to the 2D case, the tip flow reduces pressure differences the closer it comes to the tip. Up to the z-position of the local minimum the curves are simply flattening. Beyond this minimum the tip

vortex smoothes the chord-wise pressure distribution, equalizing particularly the suction side. At the tip maximum the suction side reaches two maxima: the leading edge peak and another one at $x/c \approx 0.5$. The second rather flat maximum, caused by the induced velocity of the tip vortex, makes up most of the $c_{n(z)}$ peak in the span-wise plot. Comparing the pressure distributions on the rudder surfaces (figure C.10) and the development of the tip vortices (figure C.11), it becomes clear how the pressure field on the suction side is influenced by the tip vortex, converging in negative z-direction. It finally leaves the rudder, uniting with the vortex of the pressure side, which is of very limited influence on the tip peak in the $c_{n(z)}$ distribution. This can also be observed in the chord-wise pressure plot for the maximum $c_n$ position (figure C.9).

## 4.6 Results and Discussion

### Grid Independence

Having optimized the base grid during the 2D investigation, grid independence of the *Cooper* mesh in z-direction is proved in an additional 3D independence study. Here again it must be kept in mind that the free stream and the tunnel grids are generated by different strategies – so the angle of incidence is modeled either turning the rudder or the inlet boundary conditions. As in 2D this causes slight grid differences, additionally to the intended variation of boundary conditions. This has already been investigated in 2D and described in section 3.3.3.

### Boundary Modeling

In 3D, boundary modeling of free stream and wind tunnel cases shows the expected effect. The tunnel blocking increases lift and drag prediction, compared against free stream. But for both turbulence models the total deviations of rudder performances from the experimental measurements grow, if the tunnel is modeled. That indicates that the close correlation between the free stream calculation and experimental data are just a superposition of multiple effects, compensating each other – the computations can only be regarded as exact as the prediction of the tunnel case. As demonstrated by the span-wise $c_{n(z)}$ distribution, the CFD blocking prediction also affects the tip vortex.

### Turbulence Modeling

Also for the 3D investigations turbulence modeling plays quite an important role. Particularly the Spalart-Allmaras lift prediction comes much closer to experimental benchmark than $k$-$\epsilon$ RNG does. Drag prediction does not match that well – here both turbulence models predict similar values of relatively high deviation. The tip vortex prediction is quite sensitive to turbulence modeling, so $k$-$\epsilon$ RNG predicts it more developed than Spalart-Allmaras as observed in the vortex plots as well as in the $c_{nz}$ distribution. Generally, the 3D cases are not as sensitive to the numerical capability of predicting separation and detachment as the 2D calculations are – the tip flow compensates the pressure differences on either side of the rudder, so that the 3D stall is reduced relatively compared against 2D.

# Chapter 5

# Conclusions and Summary

This documentation tries to give a detailed description of the conducted RANS simulations – at the same time it tries to show the development of the project in a way of finding an appropriate approach to treat the 3D rudder-tunnel geometry most efficiently. This optimization process has been continuing throughout the whole work. At this place the conclusion is drawing a line where some of the most interesting results should be presented and particularly emphasized – rather than giving a complete summary of the provisional results. So it is tried to make some final statements and comparisons, taking into consideration exemplary features and results, observed during the whole project.

## 5.1 Grid Generation Procedure

In terms of grid generation the 2D calculations are a sort of preparation with the goal of finding a proper method to handle the 3D cases. At the beginning the NACA0012 case at $Re = 7.6 \cdot 10^5$ is meshed, using traditional hybrid grids. No wall functions are applied, requiring very fine boundary layer resolutions. This case revealed the difficulties of this approach, consequently the later cases used boundary layers. The NACA0012 case at $Re = 6.0 \cdot 10^6$ applies unstructured triangular and quadrilateral schemes for the outer mesh, finding the latter to be the more efficient one.

The two-dimensional NACA0020 case at $Re = 8.0 \cdot 10^5$ is set up to optimize the 2D grid with respect to the final 3D calculations: The unstructured *Cooper* meshing tool is found to be an appropriate grid generation approach: It provides reasonable control over the mesh by its parameters, creating good-natured prismatic cell geometries. The *Cooper* scheme can be regarded as a compromise between unstructured and structured grids, extruding the unstructured base grids into the third dimension in a "structured" way. Strictly speaking, of course, it must be classified as unstructured. It is emphasized one more time that the base grids, consisting of quadrilateral cells, are regarded as more suitable to create the 3D *Cooper* meshes from. So the only-hexahedral hybrid grids are more homogeneous than the volume grids consisting of a triangular based outer mesh.

## 5.2 Verification Procedure

Not only the meshing scheme has been adapted and optimized to the special demands – also the verification procedure has changed with it. The procedure proposed by Stern, based on three systematically refined grid resolutions, turned out to be inappropriate in this case – particularly

without wall functions. Instead, the verification based on parametric grid independence studies, carried out independently from each other, proved to be more flexible. It allows not only to investigate parameter independence but also to develop an economical way of meshing. This approach is applied for the later two- and three-dimensional calculations.

## 5.3 Influences on Validation

### 5.3.1 Turbulence Modeling and Numerical Diffusion

Turbulence modeling plays quite an important role in this work, in some cases its influence even dominates the solutions. The $k$-$\epsilon$ RNG and Spalart-Allmaras models are applied throughout this work, some of the earlier calculations also use the standard $k$-$\epsilon$ model.

#### Lift

For the 2D NACA0020 at $Re = 8 \cdot 10^5$ $k$-$\epsilon$ RNG tends to predict a greater lift than Spalart-Allmaras does. Particularly at $\alpha \geq 15°$ only the latter one calculates negative lift gradients and detachments of larger extents. Similar results are obtained in 3D, only that detachments are of lesser extents – hence, differences at higher angles of incidence are smaller.

For the 2D NACA0012 at $Re = 6 \cdot 10^6$ only $k$-$\epsilon$ RNG is applied. Particularly for small angles the lift tends to slight over-prediction as in Date's results [3]. It is assumed that the $k$-$\epsilon$ RNG turbulence model over-estimates the lift here, as in case of the 3D NACA0020, while generally it is under-predicted with a growing numerical diffusion. So it is expected that both effects compensate each other, bringing the computations closer to the experimental results.

#### Drag

For small angles of incidence the 3D drag prediction by $k$-$\epsilon$ RNG turned out to be greater than by Spalart-Allmaras. At $\alpha \geq 15°$ this changes due to larger separation predicted by Spalart-Allmaras.

For the 2D NACA0020 it can be observed that both turbulence models almost calculate the same frictional drag component $c_{d_f}$, the greater value is determined by $k$-$\epsilon$ RNG. Basically this drag component tends to be under-predicted by numerical diffusion.

The viscous pressure drag $c_{d_{pv}}$ grows with the angle of incidence due to separation, and therefore is predicted higher by Spalart-Allmaras. Throughout most of this work the drag generally tends to be over-predicted. For rather blunt profiles (like the NACA0020) the dominating viscous pressure drag is generally over-predicted by numerical diffusion. For the slender NACA0012 at $Re = 6 \cdot 10^6$ this can be observed at $\alpha \neq 0°$, when the influence of viscous pressure drag increases. This trend is additionally supported by the influence of $k$-$\epsilon$ RNG, also over-estimating the drag (over the concerned range of incidence).

#### Tip Vortex

The prediction of the tip vortex is observed to be very sensitive to turbulence modeling. So Spalart-Allmaras predicts it to be smaller than $k$-$\epsilon$ RNG does. This can be observed in the vortex and pressure plots as well as in the span-wise distribution of the normal force coefficient. It also affects the integral lift and drag forces, and assimilates the turbulence dependent differences in 2D and 3D performances: The $k$-$\epsilon$ RNG lift decreases relatively in 3D as the tip vortex and flow are predicted in larger extends than by Spalart-Allmaras. In terms of drag prediction

the bigger tip vortex, predicted by $k$-$\epsilon$ RNG, causes an increasing induced drag, compared to Spalart-Allmaras. These 3D trends act against the turbulence dependent differences observed in 2D. Thus in 3D differences, depending on turbulence modeling, appear smaller than they do in 2D.

### 5.3.2 Boundary Conditions

The simulation is able to capture most of the blocking effects caused by the limited extension of the wind tunnel solution domain. Results have already been discussed during the two- and three-dimensional investigations, showing an increasing integral lift due to tunnel blocking. Also the drag is affected, but it must be kept in mind that the results are influenced by different grid generation approaches, used to model the angle of incidence. Particularly the prediction of the pressure drag component $c_{d_{pv}}$ is affected, which is very sensitive to slight changes in the pressure field. It is believed to affect especially the 2D cases, as dynamic pressure differences are more distinct than in 3D. Ideally, if the grid resolution is sufficient in all areas, results should show independence from these differences. This issue is not investigated further than done in section 3.3.3 due to a lack of time, but it is worth a closer look as certain grid imperfections are indicated.

It is also possible to identify a 3D blocking effect on the tip vortex. Compared to free stream, the tip peak in the span-wise normal force distribution is reduced for the tunnel case, though the rest of the normal force curve $c_{n(z)}$ increases. This can be explained by the tunnel sidewalls and roof restricting the development of the tip vortex.

### 5.3.3 Further Imperfections in the Computational Solutions

The CFD models can only represent the physics to a limited extent. Besides the mentioned numerical and modeling errors there are other deficiencies in describing reality, some of them are mentioned here:

Laminar-turbulent transition is neglected during this work. Instead the flow is assumed to be turbulent throughout the whole solution domain. This is expected to increase drag prediction slightly, as skin-friction along the laminar onset is over-estimated. Also possible effects of laminar detachment, which particularly model tests are prone to, are neglected.

To minimize the disturbing influence of the boundaries in the free stream cases, the recommendations from literature are followed, but a certain level of disturbance will surely remain. Also for the tunnel cases the boundary conditions are just a simplified model. E.g. the velocity profile of the tunnel inlet is simply implemented as a homogeneous velocity field of constant turbulence, which is put far enough to minimize its impact again. Also the fan will have a disturbing effect in reality, so that the real boundary conditions differ from the simplified CFD model. But additionally to the difficulties in modeling, there is a lack of information about the flow which is to be modeled: No real measurements of boundary layers or inlet velocity profiles are available, so the CFD model would have to be based on assumptions and simplifications, anyway.

Like the flow case also the geometry is simplified. The trailing edge of the rudder is assumed to be sharp. Additionally the gap at the root of the rudder and its turbulence triggers are neglected.

## 5.4 Critics and possible Improvements

To understand the unexpected prediction of the blocking effect on the viscous pressure drag $c_{d_{pv}}$ (section 3.3.3) it is recommended to investigate the free stream case by the same grid generation approach used for the tunnel – i.e. turning the rudder within the domain. Comparison of these results could prove grid dependence.

For future investigations of the NACA0020 at $Re = 8.0 \cdot 10^5$ it is also recommended to spend little more cells besides the rear boundary layer mesh of the rudder. Such refinement especially seems to be of interest with respect to perturbations and detachments at higher angles of incidence. This has not been discovered by the grid refinement studies as the angle of incidence has been set constant to $\alpha = 10°$. Possibly this could also affect the prediction of viscous pressure drag $c_{d_{pv}}$. Also the 2D grids of the NACA0012 cases could need some more cells on both sides of the central wake. Again, this could improve the results for higher angles of incidence, as the direction of the free stream does not match with the direction of the wake mesh for $\alpha \neq 0°$.

In the 3D grid refinement study it has become obvious that a finer resolution in z-direction is of negligible influence using this grid generation approach (wall functions on the whole surface of the rudder, including the tip). For further investigations it is estimated that the changes, recommended for the base grid, would finally result in a 3D mesh of about $4 \ldots 5 \cdot 10^6$ cells.

Furthermore, it is recommended to model the laminar-turbulent transition around the leading edge. Within *Fluent* it is possible to define the flow to be either laminar or turbulent on individual blocks. So the laminar onset could be modeled on an extra block around the leading edge, ending at the position of the turbulence trigger.

# Appendix A

# Determination of Span-wise Pressure Distribution

## A.1 Sectional Pressure Integration

A custom-made code is necessary to determine the span-wise distribution of the sectional normal force coefficient $c_{n\,(z)}$. To calculate the resultant sectional pressure force the pressure coefficient $c_p = (p - p_0)/(\rho/2\, v_0^2)$ is integrated over thin rudder slices of the thickness $\Delta z$. Its projection, perpendicular to the chord, is the normal force coefficient $c_{n\,(z)}$, which is calculated and plotted over the span. The integration is carried out for very thin slices of $z_{lo} < z < z_{hi}$, rather than for real 2D sections, as the code must cope with unstructured grids.

A rudder fixed coordinate system is chosen for the pressure integration. In the free stream cases the rudder chord always runs parallel to the x-axis of the CFD coordinate system $(x, y, z)$ (figure B.11(a)) – hence, the integration can be carried out directly. For the tunnel cases a coordinate transformation becomes necessary as for $\alpha \neq 0°$ the CFD coordinate system $(x, y, z)$ deviates from the rudder fixed system $(x', y', z)$ (figure B.11(b)). The pivot is the point of maximal thickness, $0.3c$ distant from the leading edge. The transformation reads

$$\mathbf{x}' = \mathbf{A}\,(\mathbf{x} - \mathbf{x_0}) - \mathbf{x}'_0 \tag{A.1}$$

with

$$\boldsymbol{A} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Both translation vectors are

$$\mathbf{x_0} = \mathbf{x}'_0 = \begin{pmatrix} 0.7c \\ 0 \\ 0 \end{pmatrix}$$

The trapezium integration-scheme, which is applied here, is of second order like the discretization and interpolation schemes used in the solver.

$$c'_l \approx \sum_{i=1}^{n} \left[ \frac{c_{p_{i-1}} + c_{p_{i-1}}}{2} \left(x'_i - x'_{i-1}\right) \right] \tag{A.2}$$

$$c'_d \approx \sum_{i=1}^{n} \left[ \frac{c_{p_{i-1}} + c_{p_{i-1}}}{2} \left(y'_i - y'_{i-1}\right) \right] \tag{A.3}$$

During the integration all nodes, allocated on one slice, are projected to a rudder section of constant z-coordinate. This z-location of the quasi-section is determined by

$$\overline{z} \approx \frac{\sum_{i=1}^{n} z_i}{n} \tag{A.4}$$

Having carried out the integration, the span-wise coefficients are re-transformed to the original coordinate system, perpendicular to the free stream direction:

$$\begin{pmatrix} c_d \\ c_l \\ c_z \end{pmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} c'_d \\ c'_l \\ c'_z \end{pmatrix} \tag{A.5}$$

The normal force coefficient is determined by

$$c_n = c_l \cos\alpha + c_d \sin\alpha \tag{A.6}$$

## A.2 *F90* Integration Code

The Fortran 90 code requires an input file of the $c_p$ distribution on the rudder surfaces. Additionally, the thin rudder slices need to be defined by their z-bounds. The following scheme tries to give a brief description of the integration code:

1. Pick out the nodes (coordinates and $c_p$ values) for individual rudder slices.
2. Size nodes for each slice in chord-wise direction.
3. Calculate $c_l$ and $c_n$ for each slice, integrating $c_p$ in the rudder fixed coordinate system.
4. Calculate the integral lift coefficient $c_l$ by span-wise integration of $c_{l(z)}$. This value is compared against the 3D rudder lift coefficient to prove correctness and accuracy of the code.

```
      program sortout
c Integration of pressure-coeff around slices of NACA0020
      implicit none
      integer i,j,k,l,nmtch1,nmtch2,nvar1,nvar2,nspan
      doubleprecision alpha,alphabm,fl,fd,fx,fx1,fx2,fy,fy1,fy2,
     1  zloact,zhiact,sumz,fdint,flint,cn
      doubleprecision, dimension(1:100) :: zhi,zlo,fdk,flk,zcl
      doubleprecision, dimension(1:500) :: xls1,yls1,zls1,pls1,
     1  xls2,yls2,zls2,pls2
      doubleprecision, dimension(1:25000) :: x1,y1,z1,p1,x2,y2,z2,p2
c
      open(30,file='geom_in.txt')
      open(40,file='press_in1.txt')
      open(50,file='press_out.txt')
      open(60,file='clz_out.txt')
      open(70,file='cnz_out.txt')
c
      write(*,*) 'nspan, alpha, nvar1, nvar2?'
      read(*,*) nspan,alpha,nvar1,nvar2
      read(30,*) ((zlo(i),zhi(i)),i=1,nspan)
      alphabm=alpha/180.*3.14159265
c
c Reading data file
c first pressure side (1), second suction side (2)
      call subread(nvar1,x1,y1,z1,p1)
      call subread(nvar2,x2,y2,z2,p2)
c
c Outer loop for each spanwise position
      do 5 l=1,nspan
      zhiact=zhi(l)
      zloact=zlo(l)
      fx1=0.              ! reset for integration
      fx2=0.
      fy1=0.
      fy2=0.
      sumz=0.
c
c Sorting values by rising x-cooridinate
      call subsort(nvar1,nmtch1,x1,y1,z1,p1,xls1,yls1,zls1,pls1,
     1  zhiact,zloact,alphabm)
      call subsort(nvar2,nmtch2,x2,y2,z2,p2,xls2,yls2,zls2,pls2,
     1  zhiact,zloact,alphabm)
```

Figure A.1: Fortran 90: Main program *sortout*, Page 1

```
c
c Sectional integration of pressure-coeffitients
      do 120 i=2,nmtch1
      fy1=(pls1(i)+pls1(i-1))/2*(xls1(i)-xls1(i-1))+fy1 ! no correction
      fx1=-(pls1(i)+pls1(i-1))/2*(yls1(i)-yls1(i-1))+fx1  ! neg. correction
  120 continue
      do 122 i=1,nmtch1
  122   sumz=zls1(i)+sumz
      do 125 i=2,nmtch2
      fy2=-(pls2(i)+pls2(i-1))/2*(xls2(i)-xls2(i-1))+fy2  ! neg. correction
      fx2=(pls2(i)+pls2(i-1))/2*(yls2(i)-yls2(i-1))+fx2   ! no correction
  125 continue
      do 127 i=1,nmtch2
  127 sumz=zls2(i)+sumz
      zcl(l)=sumz/(nmtch1+nmtch2)
c
      fx=fx1+fx2
      fy=fy1+fy2
c
      fd=+fx*cos(alphabm)+fy*sin(alphabm)      ! Transf. in lift and drag-coo's
      fl=-fx*sin(alphabm)+fy*cos(alphabm)
      fl=fl/0.667                               ! Dimensional coefficients
      fd=fd/0.667
      cn=fl*cos(alphabm)+fd*sin(alphabm)
      fdk(l)=fd
      flk(l)=fl
c
c Output
      write(50,160) 'pressure side,
     1   z = ',zlo(l),'...',zhi(l),'  zcl=',zcl(l)
      write(50,*) '    z,          x,          p'
      write(50,150) ((zls1(i),xls1(i),pls1(i)),i=1,nmtch1)
      write(50,160) 'suction side, z = ',zlo(l),'...',zhi(l)
      write(50,*) '    z,          x,          p'
      write(50,150) ((zls2(i),xls2(i),pls2(i)),i=1,nmtch2)
c
      write(50,170) 'Lift= ',fl,'      Drag= ',fd
      write(60,*) zls1(1), fl
      write(70,*) zls1(1), cn
      write(*,170) 'Lift= ',fl,'  Drag= ',fd,'  nx/side=',nmtch1
    5 continue
c
c Final Integration Control
      do 140 l=2,nspan
      fdint=(fdk(l)+fdk(l-1))/2*(zcl(l)-zcl(l-1))+fdint
      flint=(flk(l)+flk(l-1))/2*(zcl(l)-zcl(l-1))+flint
  140 continue
c
      write(50,*) 'cd_ctr=',fdint,'   cl_ctr=',flint
  150 format(3(f7.4,"   "))
  160 format(2(a,f6.4))
  170 format(2(a,f8.5),a,i4)
      close(30)
      close(40)
      close(50)
      close(60)
      close(70)
      end
```

Figure A.2: Fortran 90: Main program *sortout*, Page 2

```
      subroutine subread(nvar,x,y,z,p)
      implicit none
      integer i,nvar
      doubleprecision, dimension(1:25000) :: x,y,z,p
      doubleprecision, dimension(1:500) :: xmtch,ymtch,zmtch,pmtch,xls,yls,
     1  zls,pls
      character(len=1) var1,var2,var3,var4
c
c read coordinate and pressure data
      read(40,*) var1
      read(40,*) ((x(i)),i=1,nvar)  ! x-coordinates
      read(40,*) var2
      read(40,*) ((y(i)),i=1,nvar)  ! y-coordinates
      read(40,*) var3
      read(40,*) ((z(i)),i=1,nvar)  ! z-coordinates
      read(40,*) var4
      read(40,*) ((p(i)),i=1,nvar)  ! pressure values
c
      end subroutine subread
```

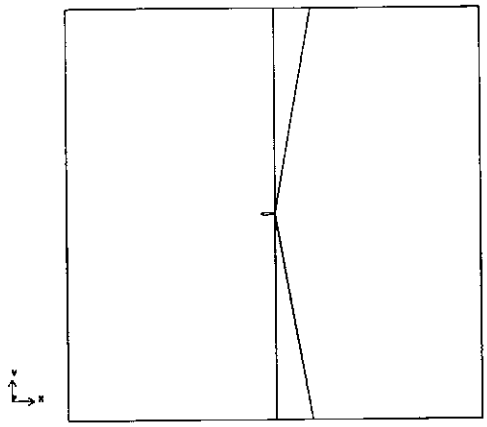
Figure A.3: Fortran 90: Subroutine *subread*

```
      subroutine subsort(nvar,nmtch,x,y,z,p,xls,yls,zls,pls,
     1  zhi,zlo,alphabm)
      implicit none
      integer i,j,nvar,nmtch,idxsor
      doubleprecision zhi,zlo,sortx,alphabm
      doubleprecision, dimension(1:25000) :: x,y,z,p
      doubleprecision, dimension(1:500) :: xmtch,ymtch,zmtch,pmtch,xls,
     1  yls,zls,pls
c
c Search for matching z-coordinates
      j=0
      do 50 i=1,nvar
      if((z(i).ge.zlo).and.(z(i).le.zhi)) then
      j=j+1
!     xmtch(j)=x(i)
!     ymtch(j)=y(i)
c
c Coordinate transformation into wing-fixed system
      xmtch(j)=(cos(alphabm)*(x(i)+0.467)+sin(alphabm)*y(i))-0.467
      ymtch(j)=-sin(alphabm)*(x(i)+0.467)+cos(alphabm)*y(i)
      zmtch(j)=z(i)
      pmtch(j)=p(i)
      endif
   50 continue
      nmtch=j              ! Number of matching data sets found
c
c Sorting the quadrils by rising x
      do 100 j=1,nmtch
      sortx=xmtch(j)
      idxsor=j
      do 110 i=1,nmtch
      if(xmtch(i).lt.sortx) then
      sortx=xmtch(i)
      idxsor=i
      endif
  110 continue
      xls(j)=xmtch(idxsor)
      yls(j)=ymtch(idxsor)
      zls(j)=zmtch(idxsor)
      pls(j)=pmtch(idxsor)
      xmtch(idxsor)=1.e9
  100 continue
      end subroutine subsort
```

Figure A.4: Fortran 90: Subroutine *subsort*

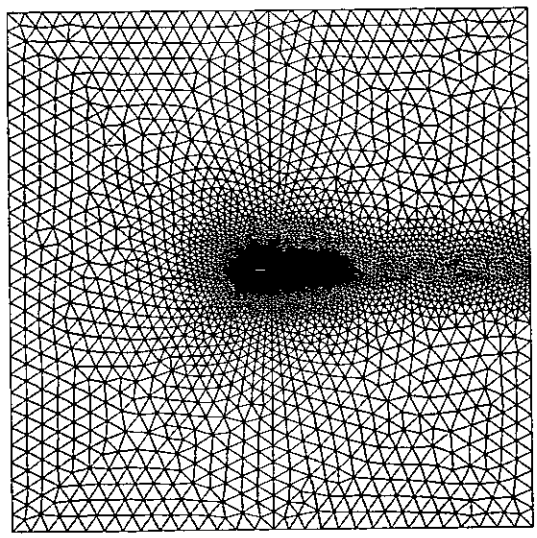
# Appendix B

# Figures of Two-dimensional Flow



(a) Block topology, total domain

(b) Block topology and boundary layer mesh, medium



(c) Hybrid mesh, medium, total domain

Figure B.1: Hybrid mesh schemes for NACA0012, $Re = 7.6 \cdot 10^5$

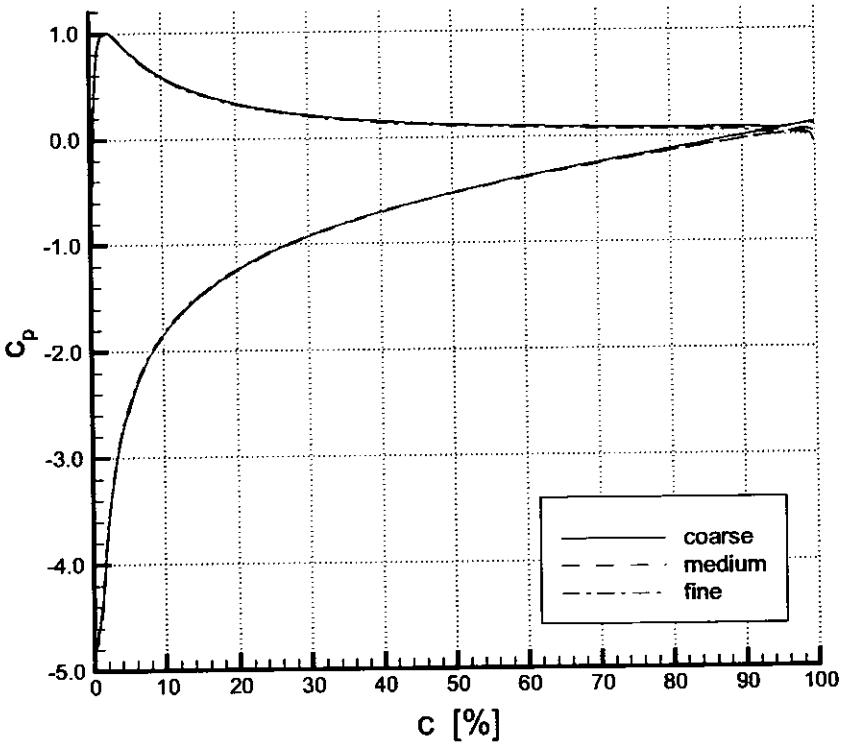
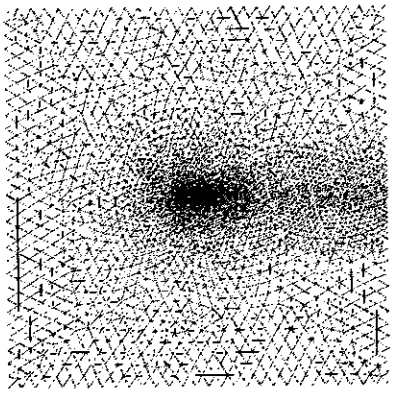
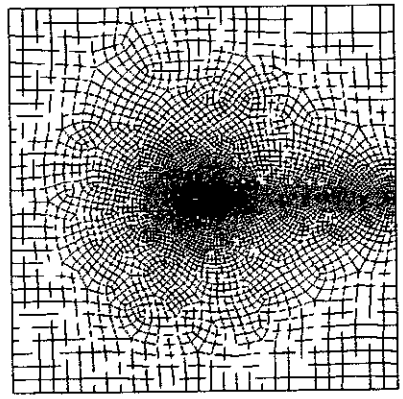


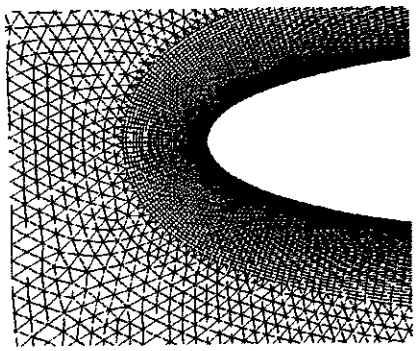

Figure B.2: $c_p$ distribution, NACA0012, $Re = 7.6 \cdot 10^5$, $\alpha = 10°$



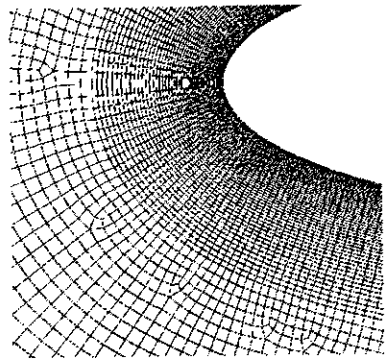
(a) Hybrid tri, total domain



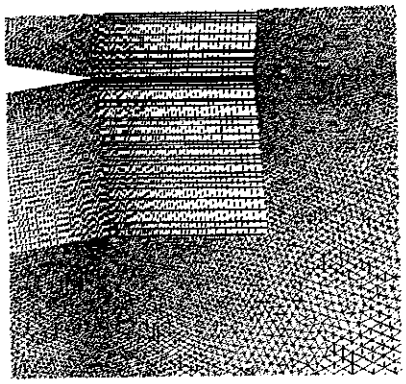
(b) Hybrid quad, total domain



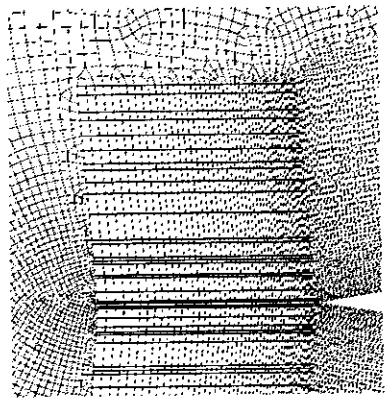
(c) Hybrid tri, leading edge



(d) Hybrid quad, leading edge

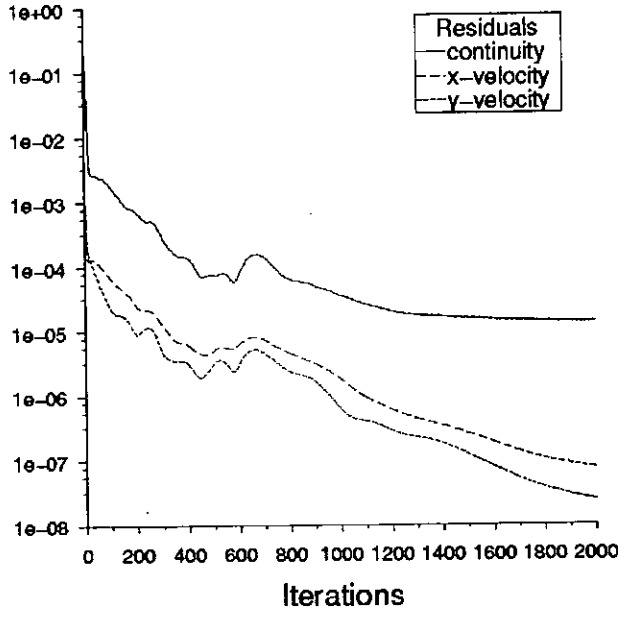


(e) Hybrid tri, trailing edge

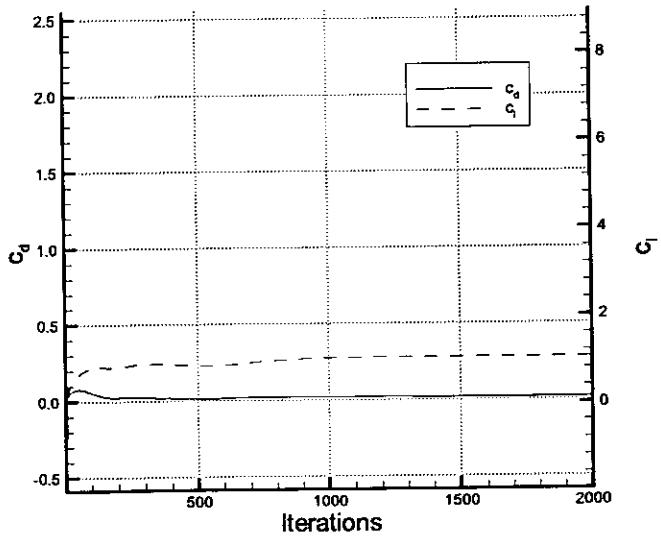


(f) Hybrid quad, trailing edge

Figure B.3: Hybrid grids, NACA0012, $Re = 6.0 \cdot 10^6$

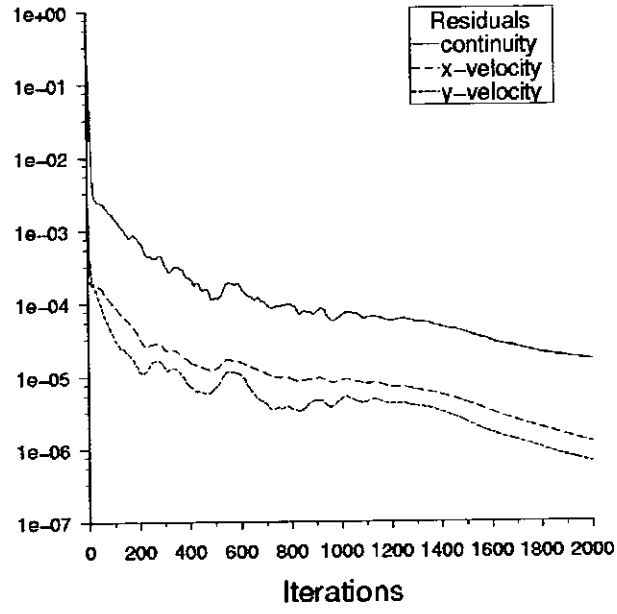



(a) Convergence of scaled residuals

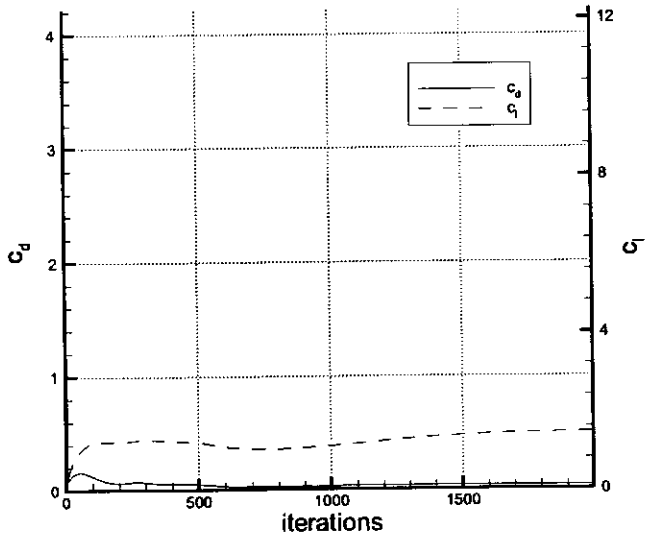



(b) Convergence of forces

Figure B.4: Convergence history, NACA0012, $Re = 6.0 \cdot 10^6$, $\alpha = 10°$, hybrid quad grids




(a) Convergence of scaled residuals




(b) Convergence of forces

Figure B.5: Convergence history, NACA0012, $Re = 6.0 \cdot 10^6$, $\alpha = 15°$, hybrid quad grids

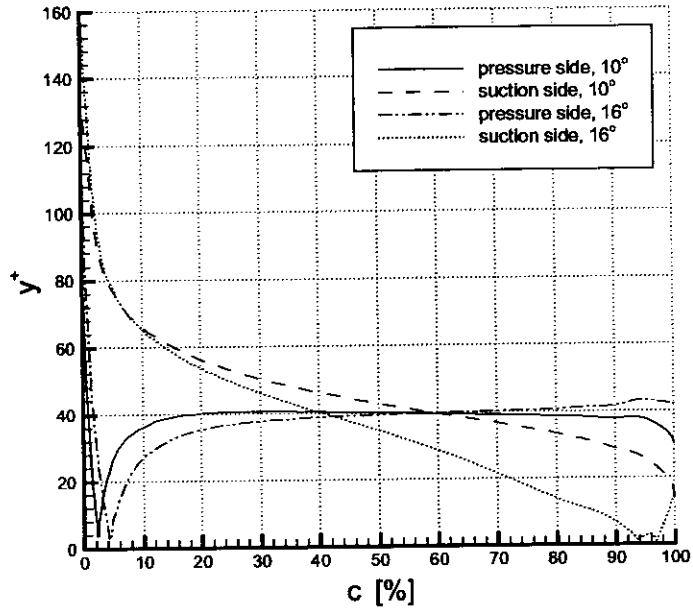



Figure B.6: $y^+$ distribution at $\alpha = 10°$ and $\alpha = 16°$, NACA0012, $Re = 6.0 \cdot 10^6$

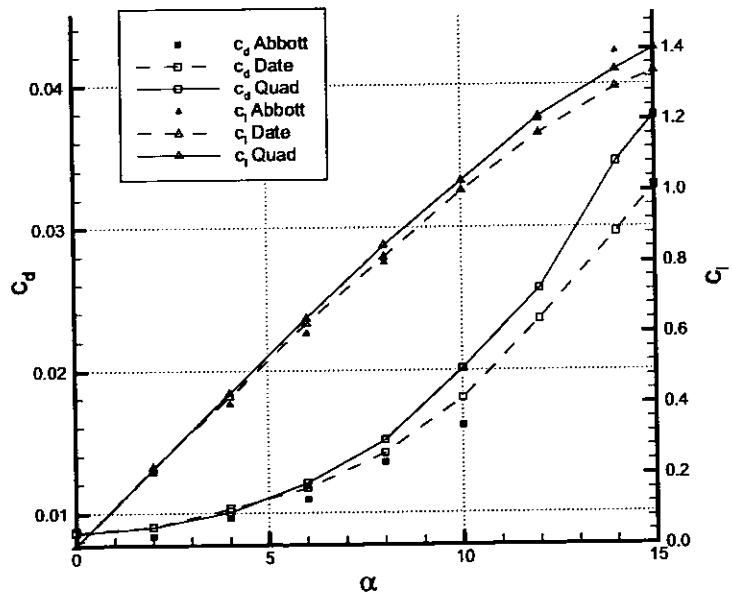



Figure B.7: NACA0012 performance, $Re = 6.0 \cdot 10^6$

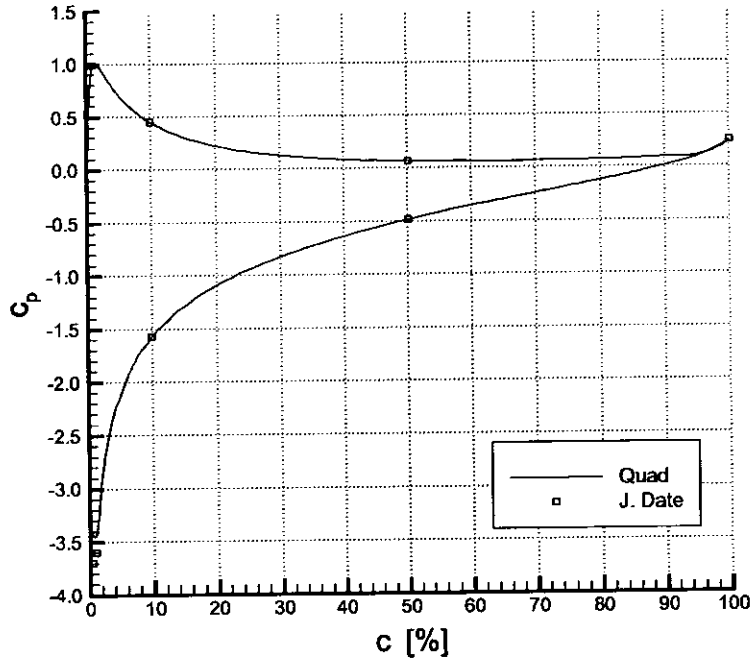



Figure B.8: $c_p$ distribution, NACA0012, $Re = 6.0 \cdot 10^6$, $\alpha = 10°$

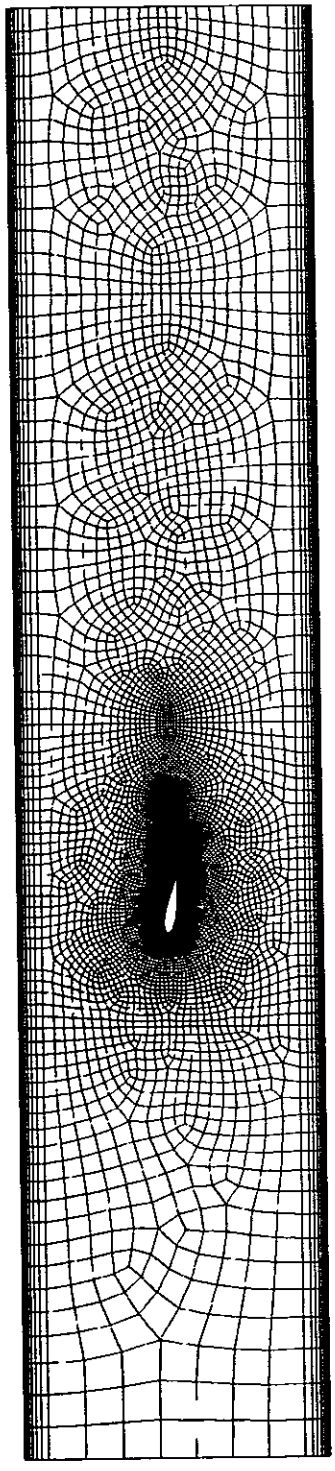


Figure B.9: Tunnel mesh, NACA0020, $Re = 8.0 \cdot 10^5$

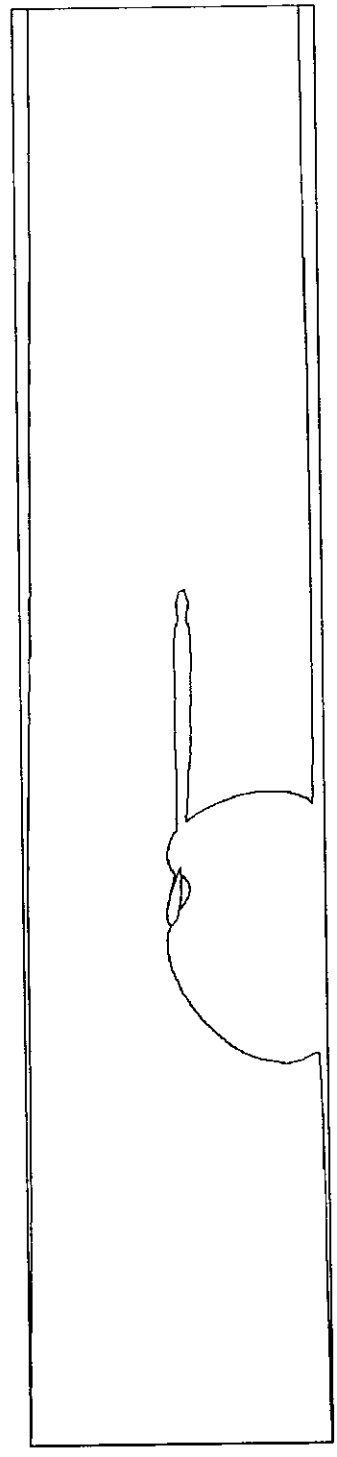
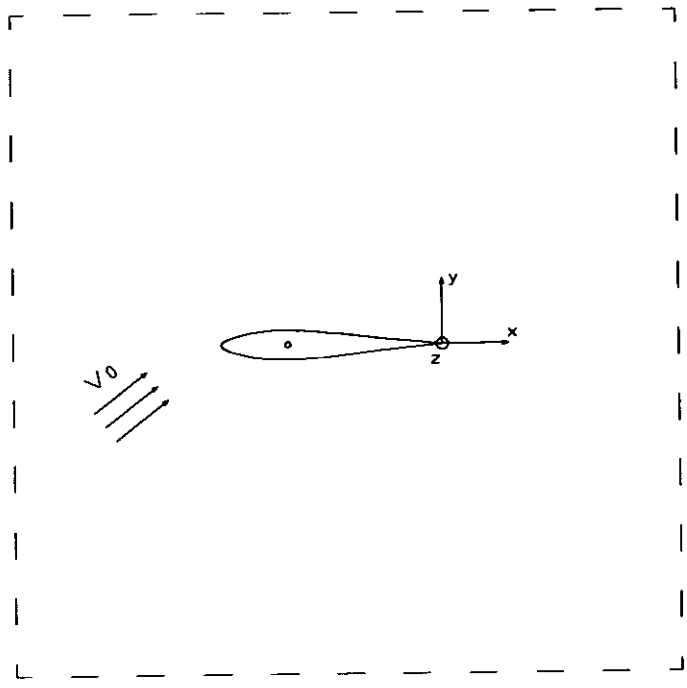
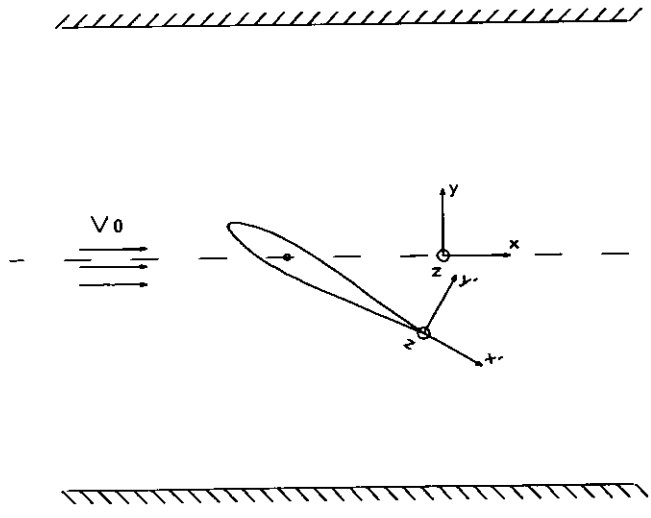


Figure B.10: $0.99v_0$ contour line, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$

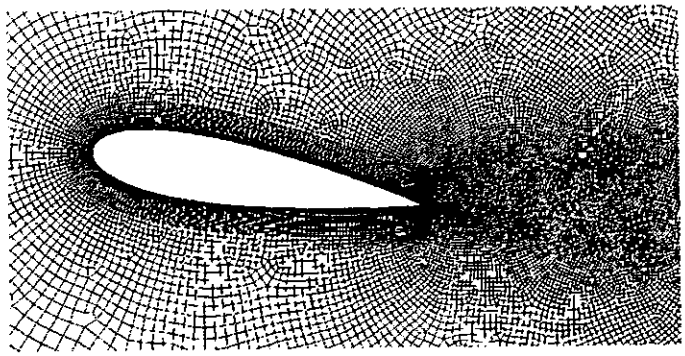



(a) Free stream

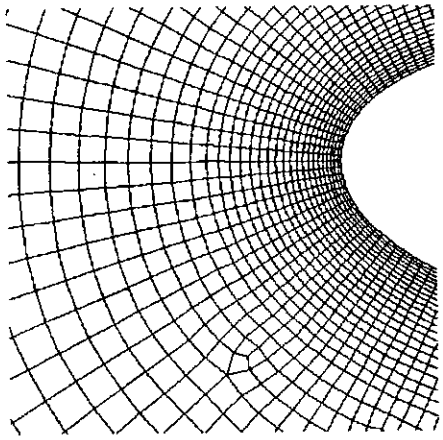



(b) Wind tunnel

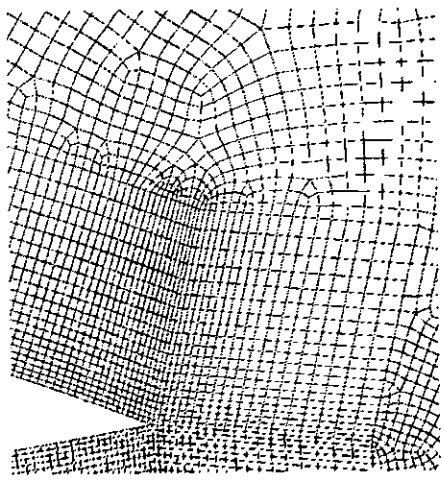
Figure B.11: Coordinate systems for NACA0020, $Re = 8.0 \cdot 10^5$



(a) Mesh around whole profile

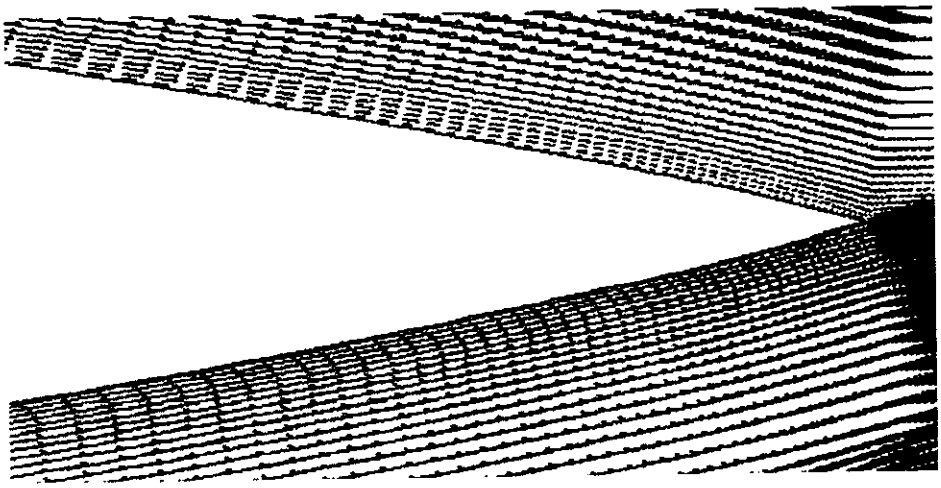


(b) Leading edge mesh

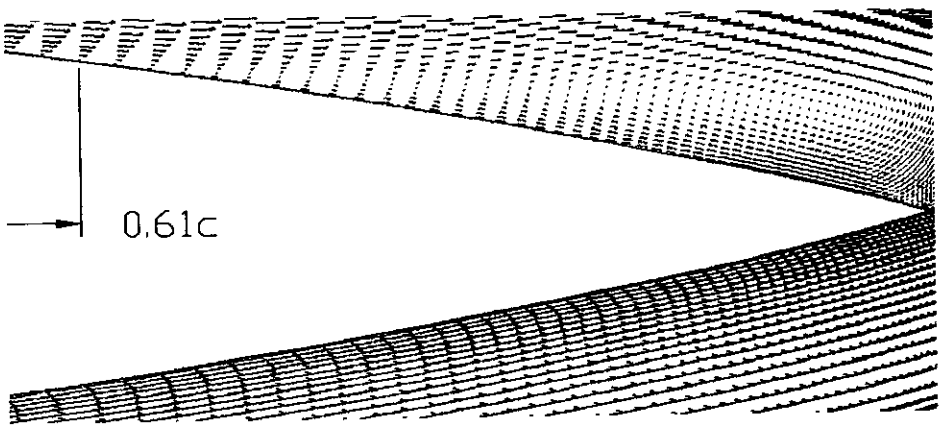


(c) Trailing edge mesh

Figure B.12: Hybrid quad grids, NACA0020, $Re = 8.0 \cdot 10^5$

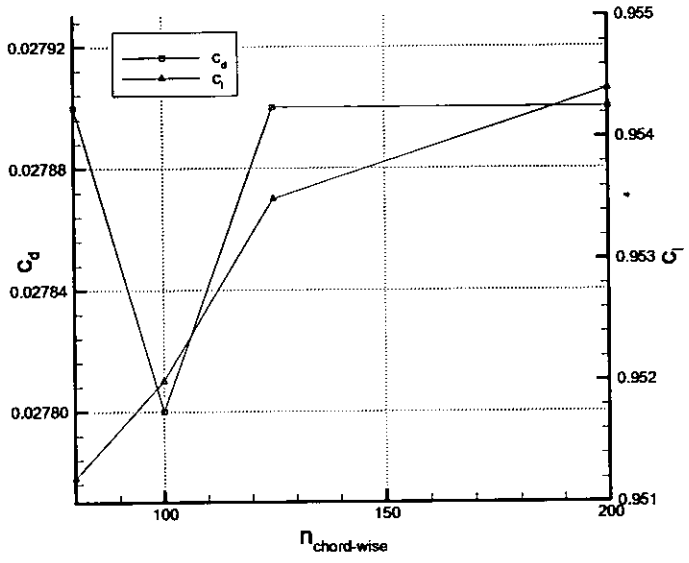


(a) $k$-$\epsilon$ RNG

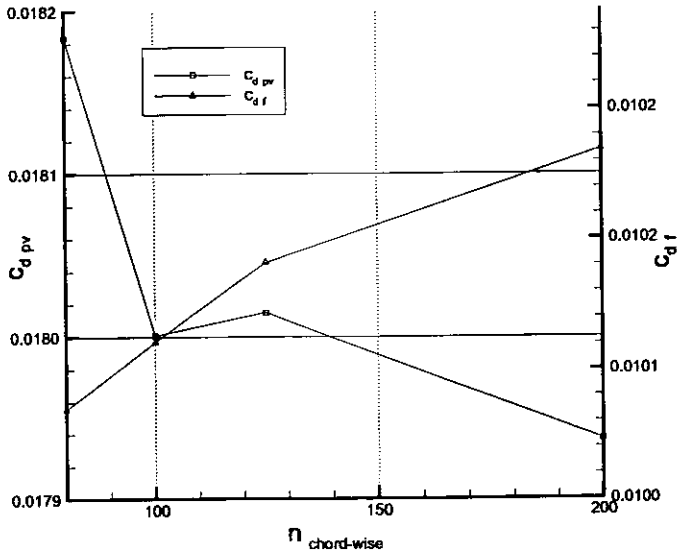



(b) Spalart-Allmaras

Figure B.13: Prediction of velocity field depending on turbulence modeling, trailing edge of NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 15°$

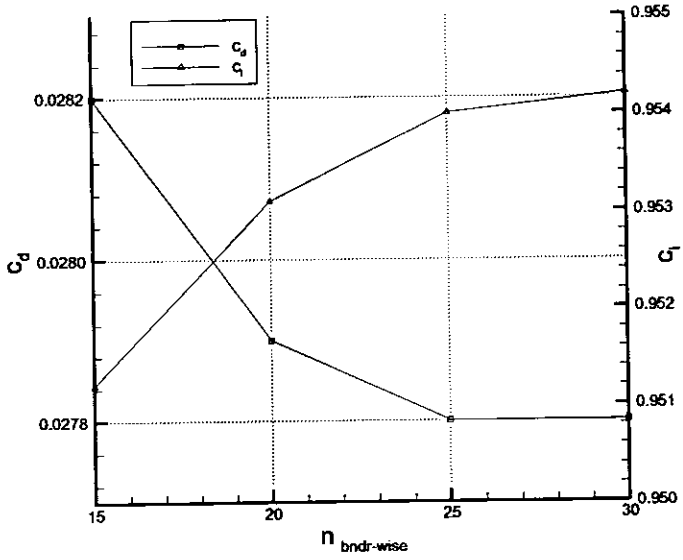



(a) $c_d$, $c_l$ grid convergence

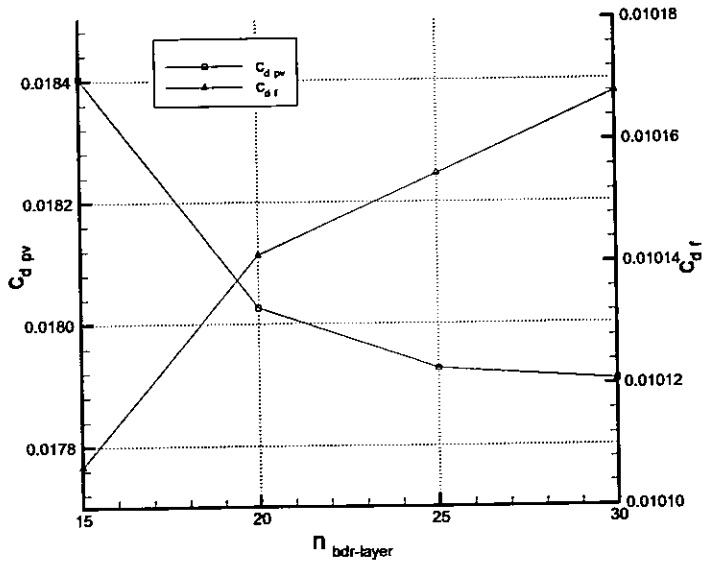



(b) $c_{d_{pv}}$, $c_{d_f}$ grid convergence

Figure B.14: Chord-wise grid convergence study, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG

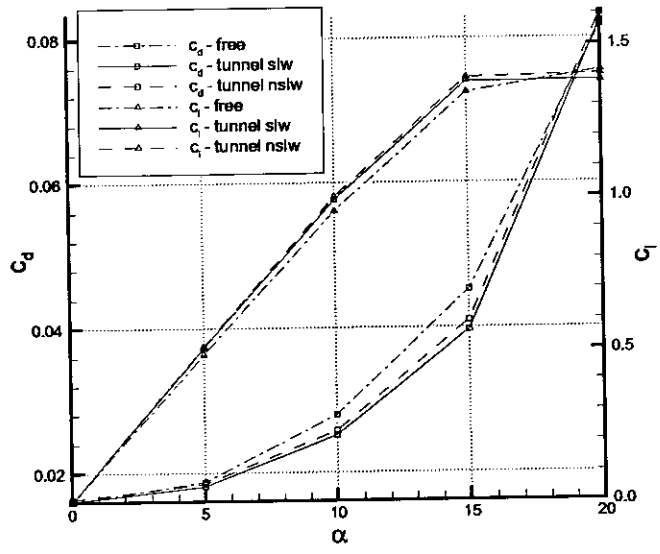



(a) $c_d$, $c_l$ grid convergence

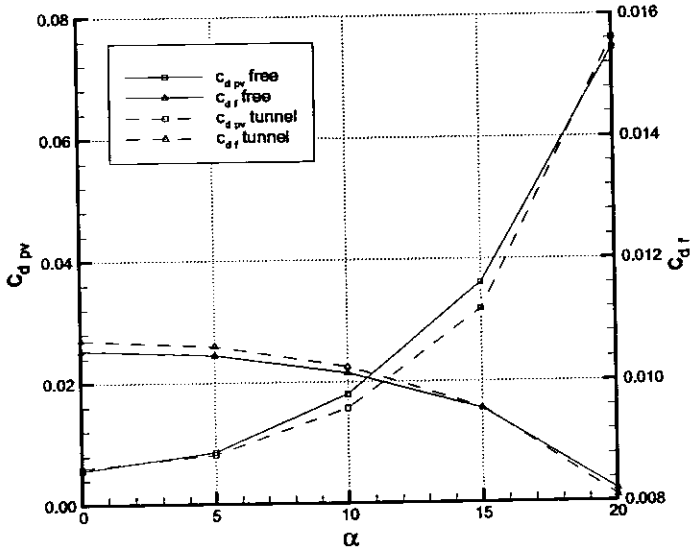



(b) $c_{d_{pv}}$, $c_{d_f}$ grid convergence

Figure B.15: Layer-wise grid convergence study, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG

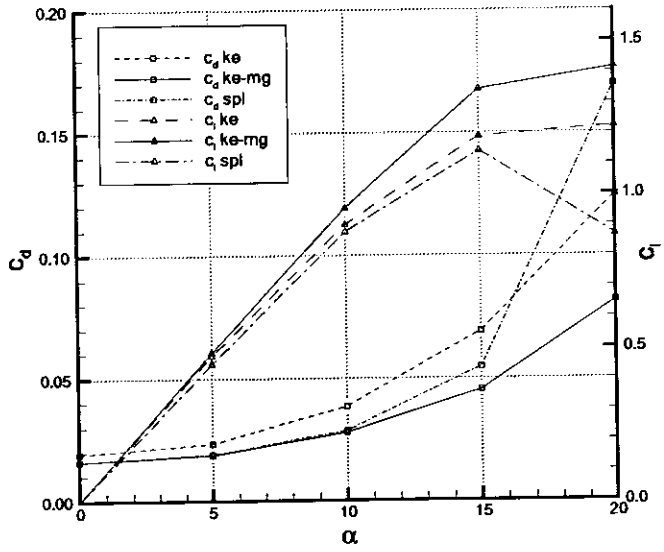



(a) $c_d$, $c_l$ performance

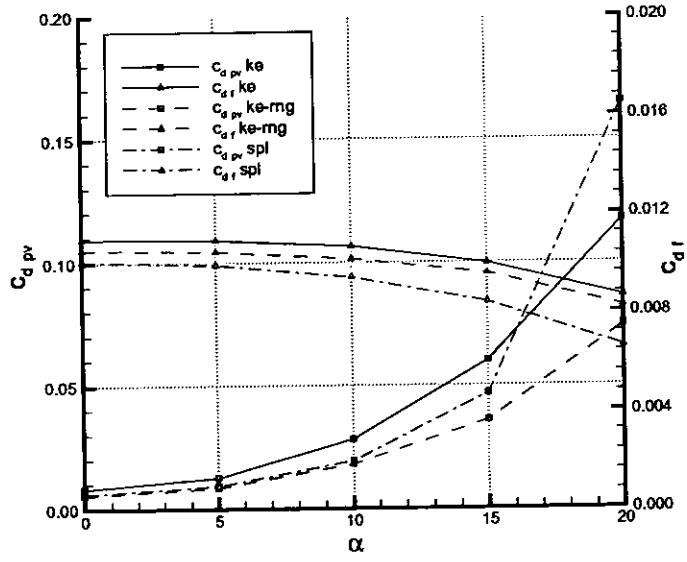



(b) $c_{d_{pv}}$, $c_{d_f}$ performance

Figure B.16: Study of tunnel blocking, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG




(a) $c_d$, $c_l$ performance




(b) $c_{d_{pv}}$, $c_{d_f}$ performance

Figure B.17: Study of turbulence models, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, free stream

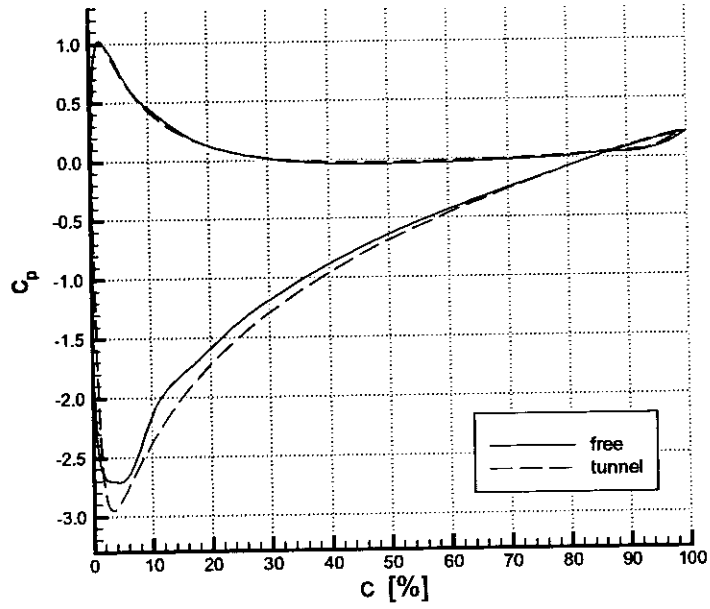
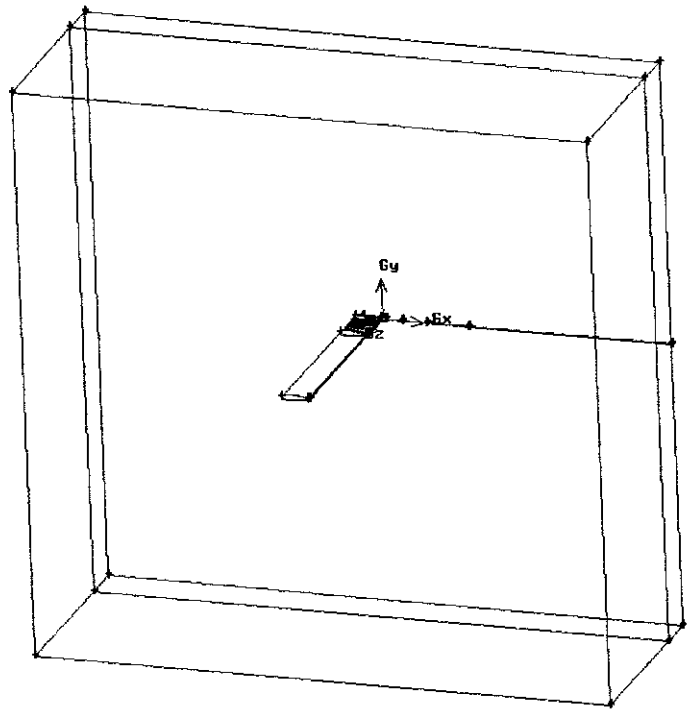


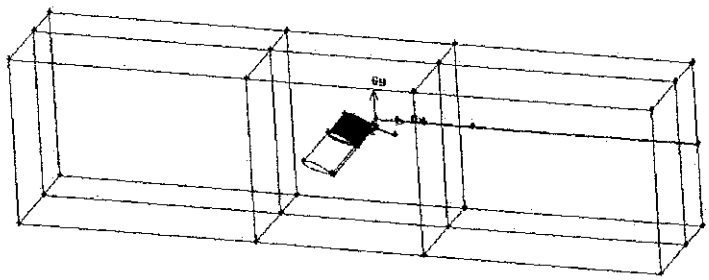

Figure B.18: $c_p$ distribution in wind tunnel and free stream boundaries, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG

# Appendix C

# Figures of Three-dimensional Flow




(a) Free stream




(b) Wind tunnel

Figure C.1: Block topology, NACA0020, $Re = 8.0 \cdot 10^5$

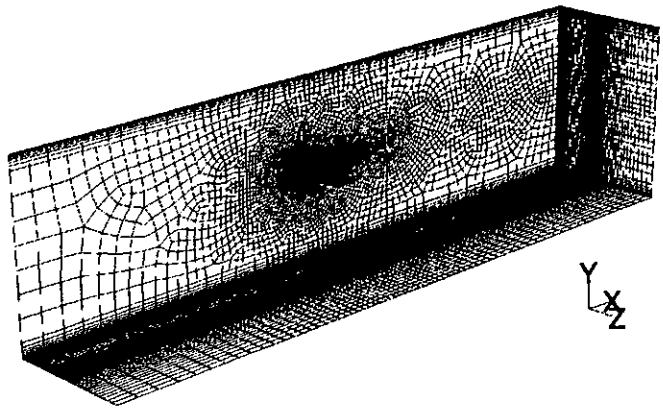
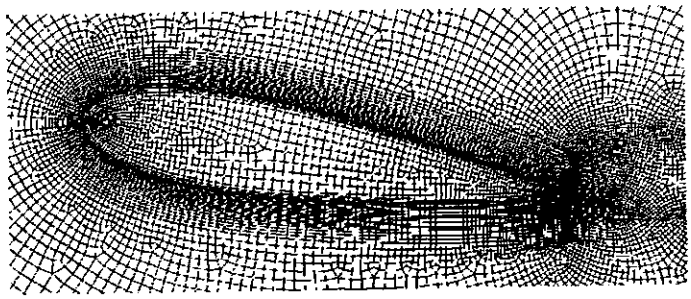
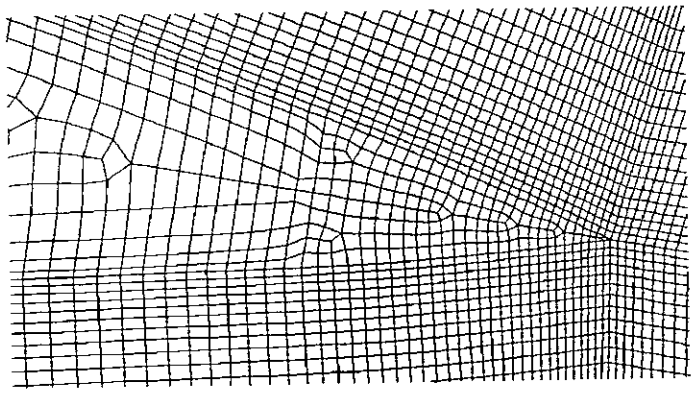



Figure C.2: Wind tunnel mesh, NACA0020, $Re = 8.0 \cdot 10^5$

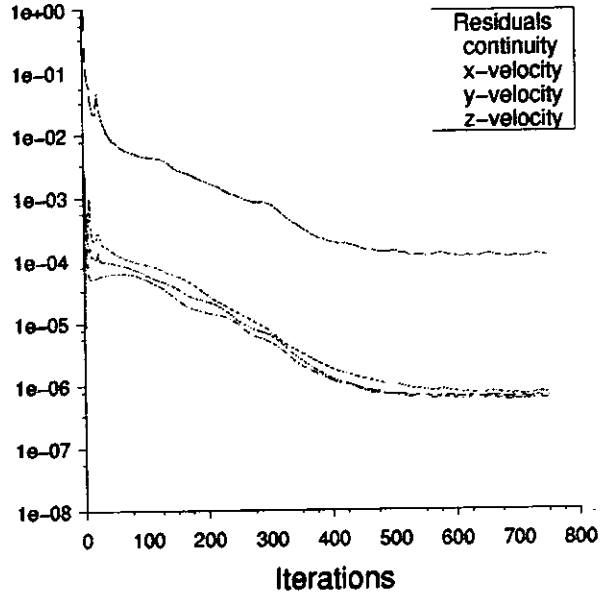


(a) Tip mesh, whole rudder

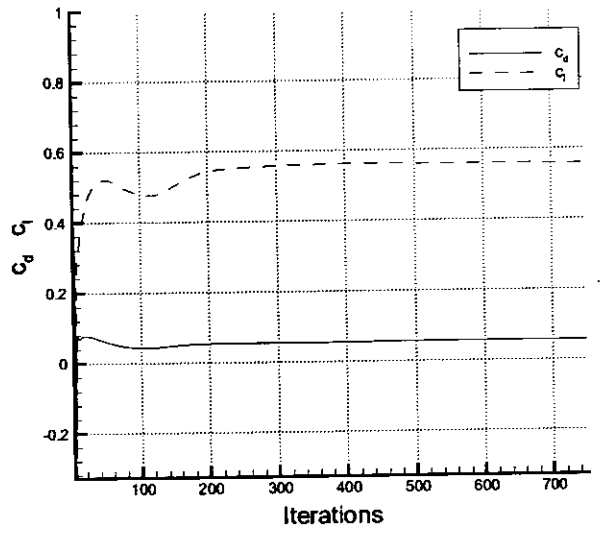


(b) Tip mesh, trailing edge

Figure C.3: Slices of 3D grid at $z = z_{tip}$, NACA0020, $Re = 8.0 \cdot 10^5$




(a) Convergence of scaled residuals




(b) Convergence of forces

Figure C.4: Convergence history, NACA0012, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG

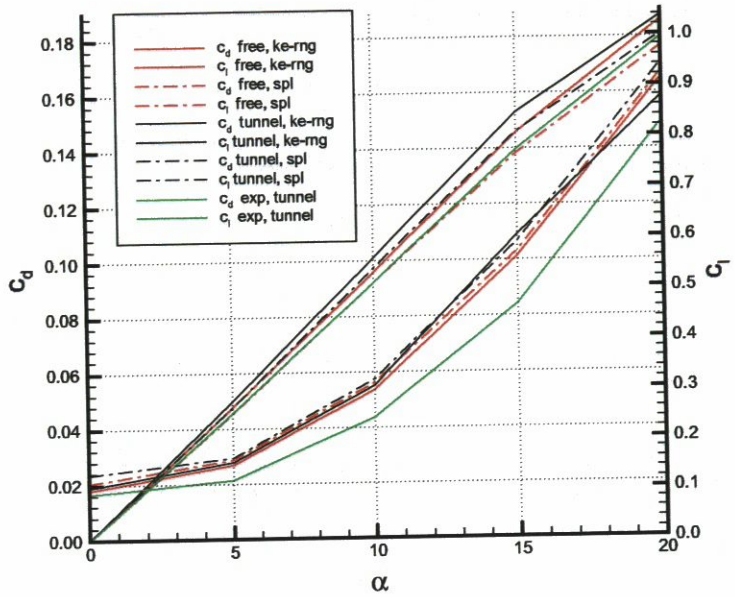



Figure C.5: 3D NACA0020 performance, $Re = 8.0 \cdot 10^5$

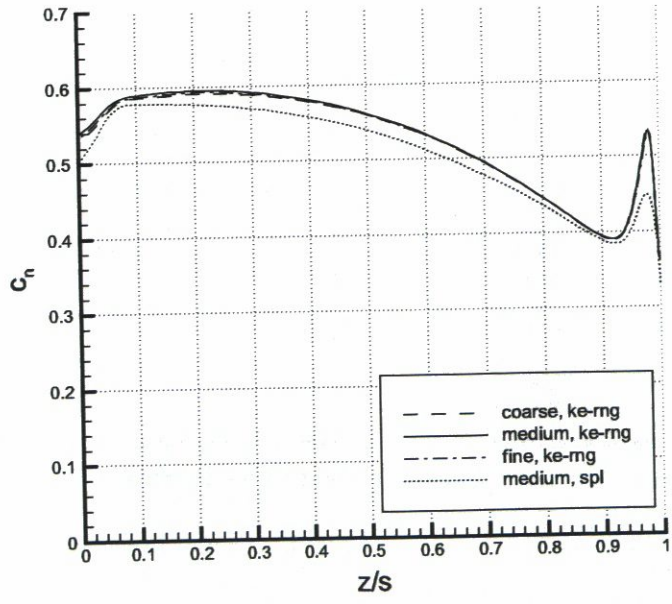



Figure C.6: Span-wise $c_n$ distribution on different grid resolutions, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, free stream

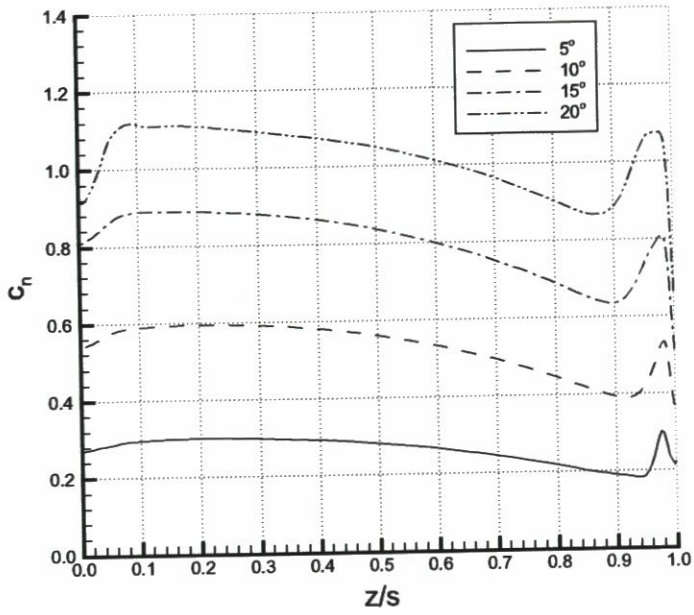



Figure C.7: Span-wise $c_n$ distribution, NACA0020, $Re = 8.0 \cdot 10^5$, $k$-$\epsilon$ RNG

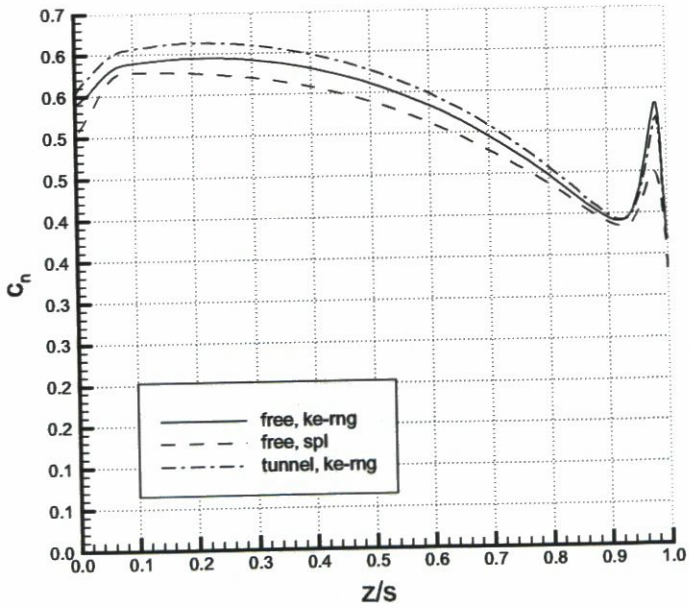



Figure C.8: Span-wise $c_n$ distribution, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$

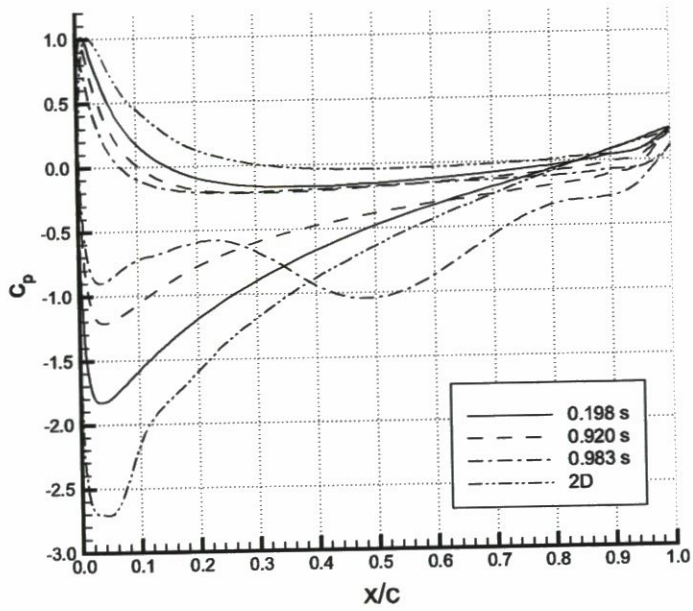
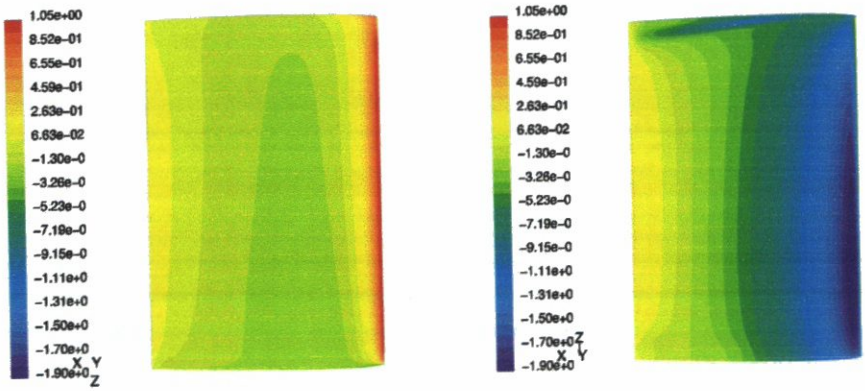
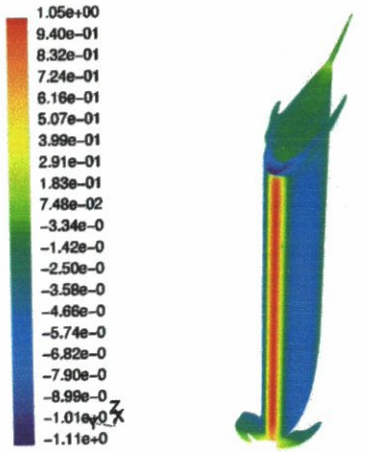



Figure C.9: Chord-wise $c_p$ distribution at selected positions of span, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, free stream

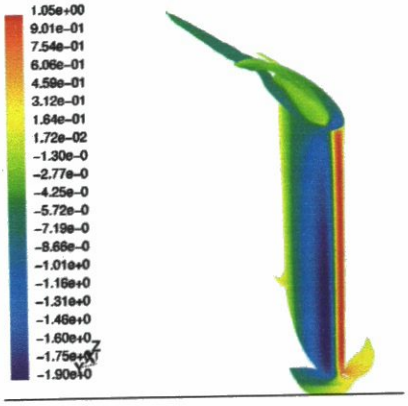



(a) Pressure side (b) Suction side

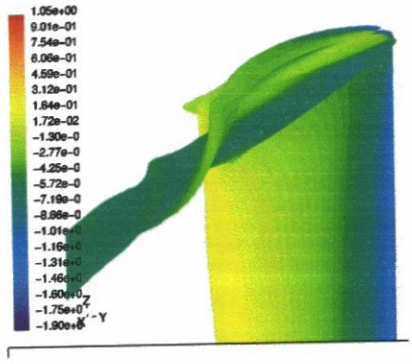
Figure C.10: $c_p$ distribution on rudder surface, NACA0020, $Re = 8.0 \cdot 10^5$, $\alpha = 10°$, $k$-$\epsilon$ RNG



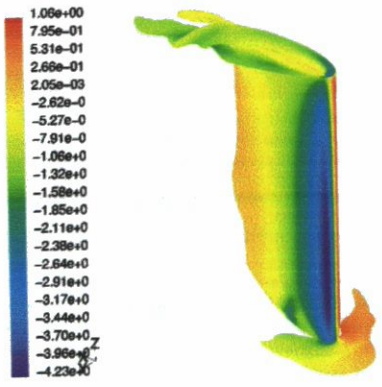

(a) $\alpha = 0°$



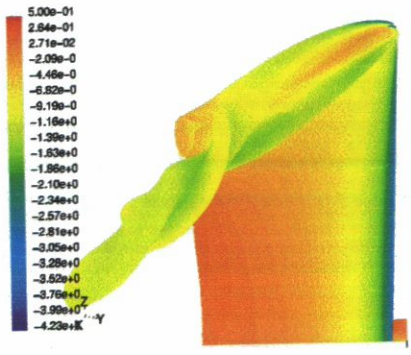

(b) $\alpha = 10°$, front view




(c) $\alpha = 10°$, rear view




(d) $\alpha = 20°$, front view




(e) $\alpha = 20°$, rear view

Figure C.11: Tip vortex development, NACA0020, $Re = 8.0 \cdot 10^5$, $k$-$\epsilon$ RNG

# Bibliography


[1] *Fluent 6.0 User's Guide.* Fluent Inc. November 2001.

[2] I. H. Abbott and A. E. von Doenhoff. *Theory of Wing Sections.* Dover Publications Inc. 1959.

[3] J. C. Date. *Performance Prediction of High Lift Rudders operating under Steady and Periodic Conditions.* PhD thesis, University of Southampton, July 2001.

[4] J. H. Ferzinger and M. Peric. *Computational Methods for Fluid Dynamics.* Springer-Verlag Berlin and Heidelberg, 1997.

[5] A. Michos, G. Bergeles, and N. Athanassisiadis. Aerodynamic characteristics of NACA0012 airfoil in relation to wind generators. *Wind Engineering, Vol. 7, No. 4,* 1983.

[6] O. A. M. El Moctar. *Berechung von Ruderkräften.* Institut für Schiffbau der Universität Hamburg, Bericht 582, 1997.

[7] O. A. M. El Moctar. *Numerische Berechung von Strömungskräften beim Manöverieren von Schiffen.* PhD thesis, TU Hamburg-Harburg, Schriftenreihe Schiffbau, Bericht 611, July 2001.

[8] A. F. Molland and S. R. Turnock. *Wind Tunnel Tests on the Influence of Propeller loading on ship rudder Performance: Four Quadrant Operation, Low and Zero Speed Operation.* Ship Science Report No. 64, University of Southampton, October 1993.

[9] B. R. Munson, D. F. Young, and T. H. Okiishi. *Fundamentals of Fluid Mechanics.* John Wiley & Sons, Canada, 1990.

[10] F. Stern, H. W. Coleman, R. V. Wilson, and E. G. Paterson. Verification and validation of CFD simulations. *Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference, San Francisco, California,* July 1999.

[11] S. R. Turnock. *Prediction of Ship Rudder-Propeller Interaction using Parallel Computations and Wind Tunnel Measurements.* PhD thesis, University of Southampton, April 1993.