

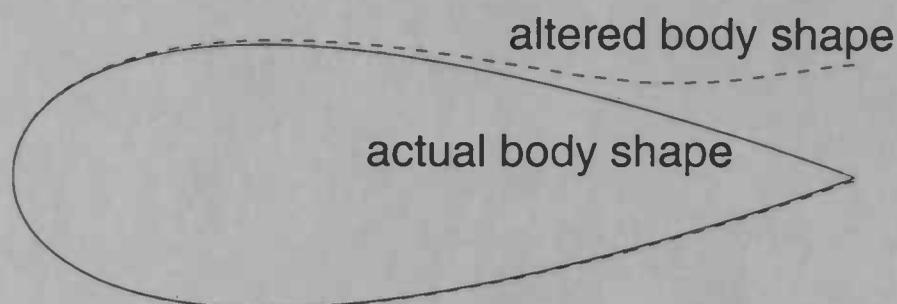
WORDT
NIET UITGELEEND

Bibliothek



Computational Techniques for Viscous-Inviscid Boundary Layer Interaction

E.G.M. Coenen



Rijksuniversiteit Groningen
Bibliotheek Informatica / Rekencentrum
Landleven 5
Postbus 800
9700 AV Groningen



KUNGL
TEKNISKA
HÖGSKOLAN

Department of
Mathematics

RUG

Computational Techniques for Viscous-Inviscid Boundary Layer Interaction

by E.G.M. Coenen



Royal Institute of Technology
Department of Aeronautical
Engineering
Stockholm, Sweden

&

University of Groningen
Department of Mathematics
Groningen, The Netherlands



Rijksuniversiteit Groningen
Bibliotheek informatica / Rekencentrum
Landleven 5
Postbus 800
9700 AV Groningen

Abstract

This study presents fast computational techniques for the solution of incompressible, laminar and turbulent flow past an aerofoil. Different methods are used for the coupling of a viscous momentum-integral boundary layer to an inviscid potential flow, solved with a panel method. These interaction methods have been compared and the (quasi-)simultaneous method, for which different interaction laws have been constructed, is shown to be the fastest and the most robust. Results from a commercial Navier-Stokes code, Rampant, are used as a benchmark for the code validation.

Preface

The research described in this project has been carried out as a Master's Thesis in the Division of Aeronautical Engineering at the Royal Institute of Technology (KTH) in Stockholm, Sweden, under the shared leadership of Professor A. Rizzi (KTH) and Professor A.E.P. Veldman (University of Groningen), and the supervision of Dr. J. Ooppelstrup (KTH). It fulfills the requirements for the Engineering Degree in Mathematics (ingenieursdiploma Technische Mechanica) at the University of Groningen, the Netherlands, where the author is a student.

Financial support for living and travel expenses was provided for by the Groninger Universiteits Fonds (GUF), the Faculty of Mathematics and Natural Science (Groningen), the Department of Mathematics and Computer Science (Groningen), the Foundation of the Vrijvrouwe van Renswoude of Delft, the Simeon Korijn Fonds, and the Royal Institute for Engineers (KIvI), for which I am very grateful.

I would like to thank everybody from the department C2M2/NADA for their help and making my stay in Stockholm very agreeable. Also I would like to thank Kristian Petterson for his help in supplying me with the results from Rampant, and Florian Wagener for his encouragement and helpful comments.

Edith Coenen
Stockholm, August 1996

Contents

Introduction	5
1 Description of the model problem: Flow around an aerofoil	7
2 Theory	10
2.1 The Navier-Stokes equations	10
2.2 The boundary layer concept	12
2.3 The outer flow	13
2.3.1 Boundary conditions	13
2.3.2 Solving the potential flow equations	14
2.4 The boundary layer	15
2.4.1 Laminar flow	16
2.4.2 Turbulent flow	18
3 Numerical analysis	23
3.1 Discretisation of the outer flow	23
3.2 Discretisation of the boundary layer equations	31
3.2.1 Laminar flow	31
3.2.2 Turbulent flow	31
4 Viscous-inviscid interaction	32
5 Results and discussion	37
5.1 Aerodynamic coefficients	37
5.2 Laminar results compared with turbulent results	39
5.3 Turbulent results	43
5.4 Validation	46
5.5 Comparison of different panel matrices and different interaction laws for the (quasi-) simultaneous method	52
5.6 Comparison of different coupling methods	56
6 Conclusions	59

A Manipulations	61
A.1 The outer flow velocity	61
A.2 The panel matrix	63
A.3 Thin-aerofoil theory	65
B Triple-deck theory	69
C Results (quasi-)simultaneous method	71
D User's guide for VIBLI	75

Introduction

The Navier-Stokes equations describe completely all possible fluid flow situations. However, since their complexity makes solution difficult, expensive and time-consuming, they have to be simplified. In the case of the flow of air past an aerofoil, viscosity effects are important only in a small region near the surface of the aerofoil. In this region the Navier-Stokes equations can be approximated by the so-called boundary layer equations, as was shown by Prandtl in 1904 [1]. Outside the boundary layer, viscous effects can be neglected, and instead of the Navier-Stokes equations one may use inviscid Euler equations.

Solution of the simplified equations for realistic geometries requires the use of numerical methods. The best known method for solving the boundary layer equations is the direct method. This method follows from the classical theory of matched asymptotic expansions [1], [2].

As the flow is divided in two parts, the flow variables of each part can be developed in asymptotic series with decreasing powers of the Reynolds number. Initially, the zero order solution of the outer expansion is calculated, which gives the solution of the Euler equations. Then, the zero order solution of the inner expansion is calculated, which gives exactly the solution of Prandtl's boundary layer equations. The effect of the boundary layer is that it tends to change the actual body surface shape, the change in profile being expressed in terms of the displacement thickness. This change in body surface shape is felt by the external flow, which now sees a different effective body from the actual profile. The outer expansion in first order is also affected, and is calculated next.

It is seen that these expansions are calculated in a sequential or a so-called hierarchical manner, as first the external flow is calculated and second the boundary layer. The above way of calculating is therefore based on a direct hierarchy.

However, the above direct method results in singularities [3] of the solution occurring near a point of separation (caused by severe adverse pressure gradient), and can therefore not be used in regions where reversed flow is present. This deficiency led to the development of the inverse method [4], in which the hierarchy is changed: the displacement thickness is prescribed instead of the pressure distribution. A related faster version named the semi-inverse method was developed by Le Balleur in 1978 [5].

These inverse methods require a reasonable choice for the displacement thick-

ness to permit stable integration through separation and into the region of reversed flow, this being achieved by way of an iterative process between the boundary layer and the outer flow. The convergence speed of these iterative methods is very slow and this will be examined later.

As explained above, both the direct and inverse methods make use of a hierarchy between the boundary layer and the outer flow. However, in regions with irregular geometry, such as a sharp trailing edge of an aerofoil, or regions with irregular flow, like reversed flow, this hierarchy is doubtful. An explanation for this is given in triple-deck theory [6], [7]. In order to solve the equations without any hierarchy, but having strong interaction between the boundary layer and the outer flow, Veldman developed the (quasi-)simultaneous method [8]. This method has a higher convergence speed and is more robust than the iterative methods.

In this study the (quasi-)simultaneous method has been adapted to find the solution for two-dimensional steady incompressible flow, both laminar and turbulent, past an aerofoil. Furthermore, this method is compared with the iterative methods. The problem posed with the (quasi-)simultaneous method is finding a good relation for the external flow, in which also the influence of the boundary layer is present. Next, a good law describing the interaction of the viscous and inviscid regions is necessary for the quasi-simultaneous method.

Two relations for the external flow and three interaction laws have been tested and will be discussed.

The simultaneous method is faster than the quasi-simultaneous method. The last method however, is more robust and can be easily used with a (compressible) Euler solver, which was originally the goal of this project. Because of this reason a lot of time has been spent on the quasi-simultaneous method. However, a lot of other interesting problems have been met and been looked at, therefore the Euler code has not been implanted.

The first part of the report deals with the governing equations. The second part outlines the numerics required to solve the equations and the different viscous-inviscid interaction methods are explained in greater detail. In the final part the computed results are presented and discussed.

Chapter 1

Description of the model problem: Flow around an aerofoil

The geometry utilised in this study is a two-dimensional aerofoil section. Therefore some aerofoil nomenclature will be given here as the definitions will return throughout this report.

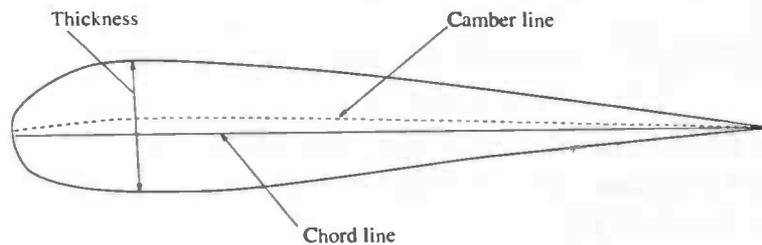


Figure 1.1: Aerofoil geometry

Considering figure 1.1, the *camber line* is defined as the set of points situated halfway between the upper and lower surfaces of the section, with these distances being measured normal to the camber line. The *chord line* is defined as the straight line running from the *leading edge* of the aerofoil to its *trailing edge*, specifying the leading and trailing edges as the forward and rearward extremities, respectively, of the camber line. The *thickness* may be defined as the dimension of the aerofoil in the direction perpendicular to the chord line and the distance between the camber line and the chord line is defined as the *camber*.

Furthermore we introduce c as the chord length, ϵc as the maximum camber, pc as the distance along the chord line from the leading edge to the point of maximum camber, and τ as the maximum thickness of the aerofoil.

By combining now a chord line and a given thickness distribution, the NACA four-digit series aerofoil sections can be obtained from formulas [9], [10].

The expressions for the coordinates on the upper and lower surface are, see also figure 1.2:

$$\begin{aligned}x_{upper} &= x_c - y_t \sin \theta, \\y_{upper} &= y_c + y_t \cos \theta, \\x_{lower} &= x_c + y_t \sin \theta, \\y_{lower} &= y_c - y_t \cos \theta,\end{aligned}$$

with (x_c, y_c) a point on the camber line. Coordinate x_c has a value between $x_c = 0$ at the leading edge and $x_c = c$ at the trailing edge. The y_c coordinate is given by two parabolas joined at the maximum camber point:

$$y_c = \begin{cases} \frac{\epsilon x_c}{p^2} \left(2p - \frac{x_c}{c} \right) & \text{for } 0 < \frac{x_c}{c} < p, \\ \frac{\epsilon(c-x_c)}{(1-p^2)} \left(1 + \frac{x_c}{c} - 2p \right) & \text{for } p < \frac{x_c}{c} < 1. \end{cases}$$

The angle between the chord line and the slope of the camber line in point (x_c, y_c) is θ . The thickness distribution y_t is given by the relation:

$$\pm y_t = 10\tau c \left[0.2960 \sqrt{\frac{x_c}{c}} - 0.1260 \frac{x_c}{c} - 0.3516 \left(\frac{x_c}{c} \right)^2 + 0.2843 \left(\frac{x_c}{c} \right)^3 - 0.1015 \left(\frac{x_c}{c} \right)^4 \right].$$

In this study the symmetric NACA 0004 and NACA 0012 wing sections have been used for test purposes. For a NACA 0012 the coordinates can be found in Abbott and Von Doenhoff [9].

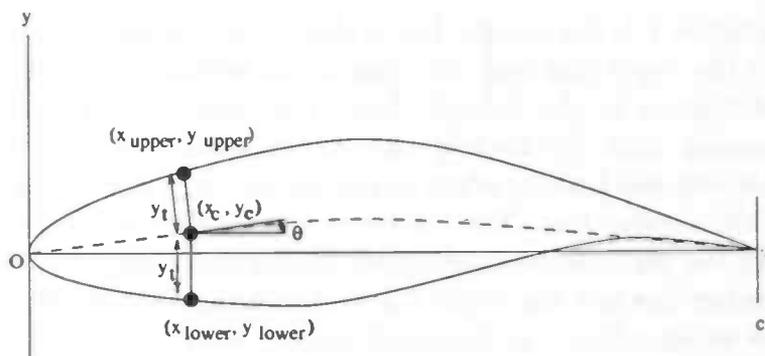


Figure 1.2: Aerofoil coordinates

The four digits of the aerofoil's designation 0012 can be explained as follows: the first digit gives $\epsilon \times 100$, the second digit gives $p \times 10$ and the last two digits give $\tau \times 100$. For the NACA 0012 this means a maximum camber of 0.0, a distance between the leading edge and the maximum camber point of 0.0 and a maximum thickness of 0.12, as can be seen in figure 1.3.

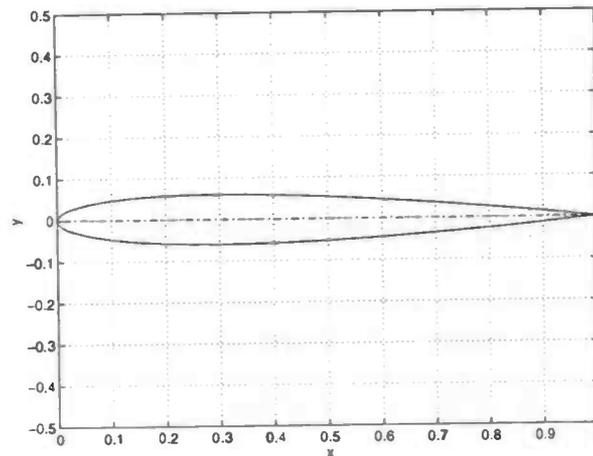


Figure 1.3: NACA 0012

Chapter 2

Theory

2.1 The Navier-Stokes equations

The compressible Navier-Stokes equations are based on the conservation laws for mass, momentum and energy. In conservative form for a two-dimensional Cartesian coordinate system (x, y) , the time-dependent Navier-Stokes equations, neglecting body forces, can be expressed as follows:

$$\frac{\partial U}{\partial t} + \frac{\partial F_c}{\partial x} + \frac{\partial G_c}{\partial y} = \frac{\partial F_v}{\partial x} + \frac{\partial G_v}{\partial y}, \quad (2.1)$$

where U , the conservative variables vector, is:

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad (2.2)$$

with ρ the density, u the horizontal and v the vertical velocity component, p the pressure and E the total energy:

$$E = e + \frac{1}{2}(u^2 + v^2),$$

where e is the internal energy.

F_c and G_c are the convective flux vectors in the x and y directions, respectively:

$$F_c = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(\rho E + p) \end{pmatrix}, \quad G_c = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + pv(\rho E + p) \end{pmatrix},$$

with the viscous fluxes defined as:

$$F_v = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ u\tau_{xx} + u\tau_{yx} - q_x \end{pmatrix}, \quad G_v = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ v\tau_{yy} + v\tau_{xy} + q_y \end{pmatrix}.$$

The viscous stress tensor is represented by τ , which for a Newtonian medium is linear in the velocity gradient. For a medium in local thermodynamical equilibrium τ can be expressed as follows:

$$\tau_{xx} = 2\mu \frac{\partial u}{\partial x} - \frac{2}{3}\mu \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad (2.3)$$

$$\tau_{yy} = 2\mu \frac{\partial v}{\partial y} - \frac{2}{3}\mu \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right), \quad (2.4)$$

$$\tau_{xy} = \tau_{yx} = \mu \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right), \quad (2.5)$$

where μ is the viscosity coefficient.

The heat flux q in each direction is expressed by Fourier's law, with T the temperature and k the coefficient of thermal conductivity:

$$q_x = -k \frac{\partial T}{\partial x}, \quad (2.6)$$

$$q_y = -k \frac{\partial T}{\partial y}. \quad (2.7)$$

To close the system of Navier-Stokes equations the state equations for a perfect gas are used:

$$p = \rho RT, \quad \epsilon = c_v T,$$

where $R = c_p - c_v$ the universal gas constant, with c_p and c_v the specific heats at constant pressure and constant volume. These specific heats are taken to be constant.

2.2 The boundary layer concept

As mentioned in the introduction, even with today's powerful computers, the complexity of the three-dimensional Navier-Stokes equations makes solution a difficult, lengthy and expensive process. In order that solutions might be found for most problems quickly and cheaply, the Navier-Stokes equations are simplified by approximating certain terms.

In 1904 Prandtl introduced the *boundary layer* concept, based on mathematical analysis [1]. He showed that the influence of viscosity is confined to a very thin layer in the immediate neighbourhood of the surface. In this layer the motion of the flow is retarded by frictional forces. The velocity $u(x, y)$ increases from zero at the wall (no-slip), to its full value at the edge of the layer, this corresponding to the velocity of a frictionless flow, U_∞ . Outside this thin layer the forces due to friction are very small and can be neglected. The flow can therefore be divided into two parts: a viscous flow in the thin boundary layer with characteristic thickness $\delta(x)$, and an inviscid flow outside this region, see figure 2.1.

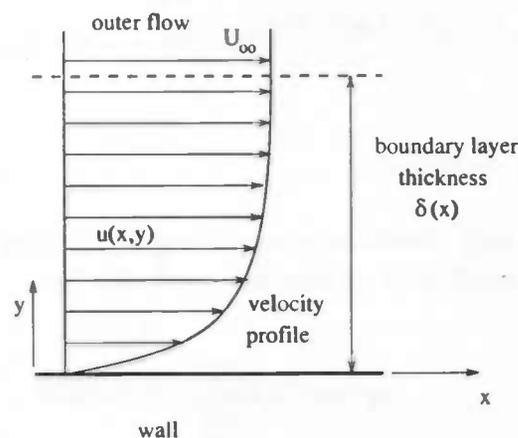


Figure 2.1: *Boundary layer velocity profile*

In the boundary layer the Navier-Stokes equations can be simplified to the boundary layer equations (section 2.4). Further away from the surface the flow is governed by the inviscid Euler equations. The complete flow can now be found by coupling this boundary layer to the outer flow.

2.3 The outer flow

If there is no viscosity ($\mu = 0$) and no heat conduction ($k = 0$), the Navier-Stokes equations are reduced to the Euler equations. For an incompressible steady flow the Euler equations are the following:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (2.8)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x}, \quad (2.9)$$

$$u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y}. \quad (2.10)$$

2.3.1 Boundary conditions

The geometry for this project is an aerofoil. Therefore instead of talking about the velocities u and v , the terms u_n and u_t , the velocities in normal and tangential direction to the curved surface, have to be used.

As the diffusion term disappears, the Euler equations require fewer boundary conditions than the Navier-Stokes equations. The boundary condition connected to the viscosity, the no-slip condition, is no longer needed. The flow-tangency condition $u_n = 0$ at the surface remains.

Near the surface the boundary layer equations are taken as the governing equations, rather than the Euler equations. However, the introduced boundary layer has the effect of disturbing the boundary condition of the outer flow: u_n is no longer equal to zero. Thus, a new boundary condition, taking the influence of the boundary layer into account, has to be found.

The aerofoil used for this project has a sharp trailing edge. For this reason the outer flow has to satisfy the *Kutta condition*, as to have no recirculation with infinite velocities at the trailing edge. The velocity should be finite at the trailing edge so as to leave the aerofoil smoothly. The Kutta condition can be described as equalizing the flow velocities on the upper and lower surface of the aerofoil at the trailing edge.

The conditions at infinity are:

$$u_t = U_\infty = 1, \quad u_n = 0, \quad p = p_\infty = 0.$$

2.3.2 Solving the potential flow equations

For an irrotational flow the velocity can be written as:

$$u = \frac{\partial \phi}{\partial x} \quad \text{and} \quad v = \frac{\partial \phi}{\partial y}. \quad (2.11)$$

Together with the continuity equation (2.8) this gives the Laplace equation:

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0. \quad (2.12)$$

Any function, ϕ , that satisfies the Laplace equation in a simply connected domain is a velocity potential of a possible flow. In the incompressible irrotational case, solving the Euler equations is equal to solving the potential equation.

Here the potential equation is solved with source and vortex distributions. The potential may be decomposed as follows:

$$\phi = \phi_{\infty} + \phi_s + \phi_v, \quad (2.13)$$

with ϕ_{∞} the potential of the uniform onset flow, ϕ_s the potential of a source distribution of strength $q(s)$ per unit length and ϕ_v the potential of a vortex distribution of strength $\gamma(s)$ per unit length.

Hence,

$$\phi = U_{\infty}(x \cos \alpha + y \sin \alpha) + \int \frac{q(s)}{2\pi} \ln r ds - \int \frac{\gamma(s)}{2\pi} \theta ds. \quad (2.14)$$

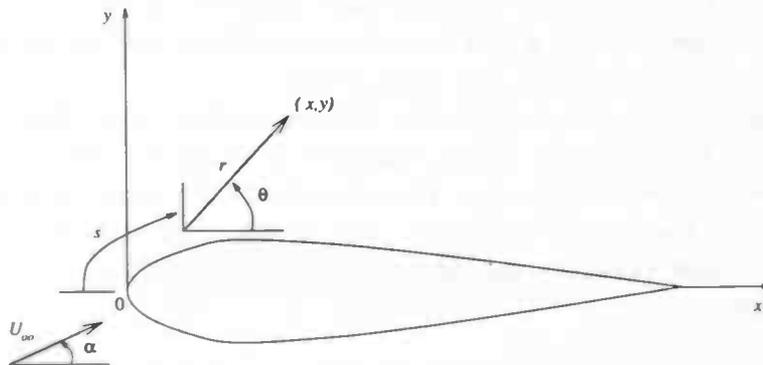


Figure 2.2: Nomenclature analysis

The velocity at infinity is denoted by U_∞ and the angle of attack by α . The integrals are taken over the body surface, with s the distance measured along the surface and (r, θ) the polar coordinates of the point (x, y) , see figure 2.2. The source and vortex strength are determined so as to meet the boundary condition of flow tangency and the Kutta condition.

The source strength $q(s)$ varies over the surface and the vortex strength $\gamma(s)$ is taken to be constant over the whole aerofoil. The source strength can be seen as being related to the flow-tangency condition, which must be satisfied for all the points on the surface. The vortex strength can be seen to be related to the Kutta condition, which has only to be satisfied at the trailing edge and so $\gamma(s)$ can be taken to be constant.

2.4 The boundary layer

The boundary layer hypothesis is that the viscous terms only play an important role close to the solid surface. At the surface the motion of the fluid is retarded, so in a thin layer the velocity increases from zero at the surface to a value equal to the value of the inviscid outer flow. This thin layer has been called the boundary layer.

The boundary layer has the effect of increasing the apparent aerofoil thickness profile for the outer flow, see figure 2.3. The shape of the new body surface shape can be expressed by the *displacement thickness*:

$$\delta^*(x) = \int_0^\infty \frac{1}{u_e} (u_e - u) dy, \quad (2.15)$$

where u_e is the tangential velocity at the edge of the boundary layer. The boundary condition for the outer flow can now be corrected. Using these new definitions in the continuity equation the *transpiration velocity*, the normal velocity at the undisturbed solid surface, can be obtained:

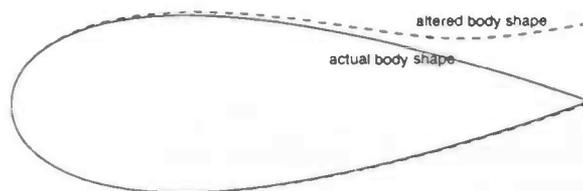


Figure 2.3: Actual body shape plus the effect of the boundary layer, $\alpha > 0$

$$v(x, 0) = \frac{d}{dx}(u_e \delta^*). \quad (2.16)$$

This transpiration velocity can be seen as a disturbance caused by the boundary layer.

2.4.1 Laminar flow

The boundary layer equations can be derived from the Navier-Stokes equations. In the case of a flat plate the incompressible steady boundary layer equations are written as [1]:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (2.17)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \frac{\partial^2 u}{\partial y^2}, \quad (2.18)$$

$$\frac{\partial p}{\partial y} = 0. \quad (2.19)$$

The boundary conditions are $u = v = 0$ at the surface, and $u \rightarrow u_e$ and $p \rightarrow p_e$ at the edge of the boundary layer. The values u_e and p_e come from the Euler equation at the edge of the boundary layer.

We introduce now some new definitions:

- momentum thickness:

$$\theta = \frac{1}{u_e^2} \int_0^\infty (u_e^2 - u^2) dy - \delta^* = \int_0^\infty \frac{u}{u_e} \left(1 - \frac{u}{u_e}\right) dy, \quad (2.20)$$

- shape factor:

$$H = \frac{\delta^*}{\theta}, \quad (2.21)$$

- skin friction:

$$\tau_w = \left(\mu \frac{\partial u}{\partial y} \right) \Big|_{y=0}, \quad (2.22)$$

- skin friction coefficient:

$$C_f = \frac{\tau_w}{\frac{1}{2} \rho u_e^2}. \quad (2.23)$$

With the use of these new definitions the momentum-integral equation, also called the *von Kàrmàn equation*, can be derived from the boundary layer equations [1]:

$$\frac{C_f}{2} = \frac{d\theta}{dx} + \theta \frac{(H+2)}{u_e} \frac{du_e}{dx}. \quad (2.24)$$

The von Kàrmàn equation can also be written in terms of δ^* , using definition (2.21):

$$\frac{C_f}{2} = \frac{d}{dx} \left(\frac{\delta^*}{H} \right) + \left(\frac{2}{H} + 1 \right) \frac{\delta^*}{u_e} \frac{du_e}{dx}. \quad (2.25)$$

This equation contains three unknowns: δ^* , H and C_f . Hence, it can be seen that two further relations are needed to close the system. In this project the closure functions based on solutions of the Falkner-Skan equation are used [11], [12]:

$$H = 2.591 \exp\left(-\frac{\partial u_e}{\partial x} \delta^{*2} Re/6\right),$$

$$f_2 = 0.3149 \exp(2.5541 - H) - 0.08, \quad \text{for } H \leq 7$$

$$f_2 = (0.3149 \exp(2.5541 - 7) - 0.08)7/H, \quad \text{for } H > 7$$

where Re is the Reynolds number:

$$Re = \frac{\rho U_\infty L}{\mu}, \quad (2.26)$$

and

$$f_2 = \frac{C_f \delta^* Re}{2 H}. \quad (2.27)$$

If the pressure gradient becomes positive, and therefore the velocity gradient becomes negative, the flow at the edge of the boundary layer is slowed down and if the adverse pressure gradient is strong enough, it may cause flow reversal.

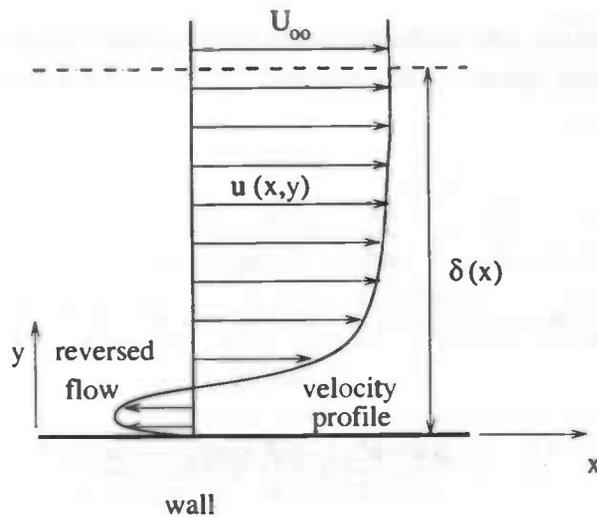


Figure 2.4: *Flow reversal*

The point at which the streamline separates from the surface is called the *point of separation*. It can be seen that if the term $\frac{du_e}{dx}$ becomes negative, δ^* will become large and in the end the solution will break down. Closure relations, able to produce a negative skin friction coefficient are necessary, in order to handle at least some separation.

2.4.2 Turbulent flow

When the Reynolds number is sufficiently small, small disturbances are damped out by the viscosity, hence the flow is stable. Such flows are called laminar. For larger Reynolds numbers, even a small disturbance can be strongly amplified leading to a unstable and time-dependent flow. This turbulent flow however, is still described by the Navier-Stokes equations.

For steady turbulent flows, the flow properties can be averaged over time. The variables are separated into an average value and a fluctuating part, for instance:

$$u(x, y, t) = \bar{u}(x, y) + u'(x, y, t),$$

with

$$\bar{u}(x, y) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t_0}^{t_0+T} u(x, y, t) dt,$$

the time-average of $u(x, y, t)$.

Together with the so-called 'Reynolds rules of averaging' the incompressible time-averaged Navier-Stokes equations become [2], [10]:

$$\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial y} = 0, \quad (2.28)$$

$$\rho(\bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y}) = -\frac{\partial \bar{p}}{\partial x} + \frac{\partial}{\partial x}(\mu \frac{\partial \bar{u}}{\partial x} - \rho \overline{u'u'}) + \frac{\partial}{\partial y}(\mu \frac{\partial \bar{u}}{\partial y} - \rho \overline{u'v'}), \quad (2.29)$$

$$\rho(\bar{u} \frac{\partial \bar{v}}{\partial x} + \bar{v} \frac{\partial \bar{v}}{\partial y}) = -\frac{\partial \bar{p}}{\partial y} + \frac{\partial}{\partial x}(\mu \frac{\partial \bar{v}}{\partial x} - \rho \overline{u'v'}) + \frac{\partial}{\partial y}(\mu \frac{\partial \bar{v}}{\partial y} - \rho \overline{v'v'}). \quad (2.30)$$

There are more unknowns than there are equations and so the system is not yet closed. From these equations the turbulent boundary layer equations can now be derived.

The fluctuations u' and v' are of the same order, in contrast to the mean flow terms u and v , hence also the Reynolds stresses $\overline{u'v'}$, $\overline{u'u'}$ and $\overline{v'v'}$ are of the same order. As the characteristic thickness of the boundary layer is assumed to be small compared to the characteristic length of the profile, the y -derivatives still dominate the x -derivatives. With this the system can be approximated by the following incompressible turbulent boundary layer equations:

$$\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial y} = 0, \quad (2.31)$$

$$\bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} + \frac{\partial}{\partial y}(\nu \frac{\partial \bar{u}}{\partial y} - \overline{u'v'}), \quad (2.32)$$

$$\frac{1}{\rho} \frac{\partial \bar{p}}{\partial y} = \frac{\partial}{\partial y}(\nu \frac{\partial \bar{v}}{\partial y} - \overline{v'v'}), \quad (2.33)$$

with $\nu = \frac{\mu}{\rho}$. As the boundary layer is so thin, the pressure in the x -momentum equation can be taken from the inviscid pressure distribution. The Euler equation at the edge of the boundary layer can therefore still be used to eliminate the pressure. Hence,

$$\bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} = u_e \frac{\partial u_e}{\partial x} + \frac{\partial}{\partial y}(\nu \frac{\partial \bar{u}}{\partial y} - \overline{u'v'}). \quad (2.34)$$

As the Reynolds stress terms are of the same order in (2.32) and (2.33), $\frac{\partial \bar{p}}{\partial x}$ and $\frac{\partial \bar{p}}{\partial y}$ should be of the same order as well. However, as the y -derivatives dominate the x -derivatives, $\frac{\partial \bar{p}}{\partial y}$ can also be taken equal to zero again.

Then the only difference between equations (2.31), (2.33) and (2.34), with u and v replaced by their steady values, and the former local laminar boundary layer equations (2.17), (2.18) and (2.19), is the incorporation of the turbulent

Reynolds stresses $-\overline{\rho u'v'}$ in the x -momentum equation. This turbulent stress forms together with the wall shear stress the total stress τ :

$$\tau = \mu \left(\frac{\partial \bar{u}}{\partial y} \right) \Big|_{y=0} - \overline{\rho u'v'}. \quad (2.35)$$

As before, the momentum equation can be transformed into the integral-momentum equation, only this time with a different τ . So the von Kàrmàn equation is still valid. However, new closure relations are needed in order to determine again the unknowns δ^* , H and C_f .

In the same way there is a global momentum equation there is also a global continuity equation. This equation can be found by assuming that the mass transport, using the definition of δ^* , is given by:

$$Q(x) = \int_0^{\delta(x)} u dy = \int_{\delta^*}^{\delta} u_e dy = u_e(\delta - \delta^*), \quad (2.36)$$

The mass transport Q does not have to be constant, but it can vary between adjacent positions x and $x + \Delta x$. The rate at which Q increases with x is called the entrainment velocity E and is defined as:

$$E = \frac{dQ}{dx} = \frac{d}{dx}(u_e(\delta - \delta^*)) = \frac{d}{dx}(u_e \theta H_1), \quad (2.37)$$

with

$$H_1 = \frac{\delta - \delta^*}{\theta}. \quad (2.38)$$

The equation for E is called the *entrainment equation*. Head [10] assumed that the entrainment velocity E after being non-dimensionalised by dividing by u_e , is only dependent on H_1 :

$$\frac{E}{u_e} = 0.0306(H_1 - 3.0)^{-0.6169}. \quad (2.39)$$

H_1 in turn is a function of the shape factor H . The function with which Cebeci and Bradshaw fitted their experimental data is [10]:

$$H_1 = 3.3 + 0.8234(H - 1.1)^{-1.287}, \quad \text{for } H \leq 1.6$$

$$H_1 = 3.3 + 1.5501(H - 0.6778)^{-3.064}, \quad \text{for } H > 1.6$$

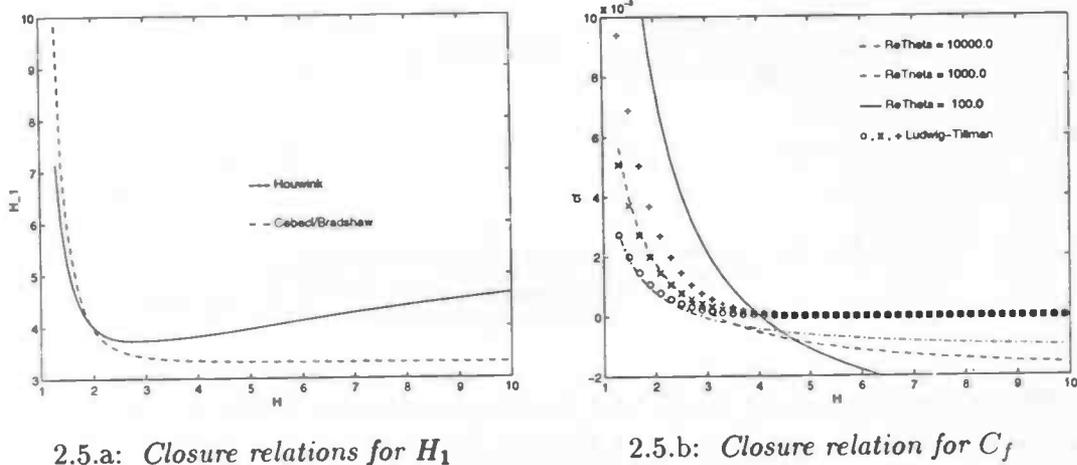


Figure 2.5: Turbulent closure relations

A closure function used by Houwink at the NLR is [13]:

$$H_1 = (0.5H + 1)H/(H - 1), \quad \text{for } H \leq 2.732$$

$$H_1 = (0.5ht + 1)ht/(ht - 1), \quad \text{for } 2.732 < H \leq 4$$

$$H_1 = 1.75 + 5.52273ht/(ht + 5.818181), \quad \text{for } H > 4$$

with

$$ht = 0.5(H - 2.732) + 2.732.$$

The closure problem is completed with a formula for the skin friction. Ludwig-Tillman skin friction law gives:

$$C_f = 0.246(10^{-0.678H})Re_\theta^{-0.268}, \quad (2.40)$$

where

$$Re_\theta = \frac{u_e \theta}{\nu}, \quad (2.41)$$

As this formula can not predict a negative C_f it is unable to handle separation. A better relation, able to handle some separation comes from Houwink [13]:

$$C_f = \left(\frac{0.01013}{\log Re_\theta - 1.02} - 0.00075 \right) \left(\frac{0.9}{H/h_0} - 0.5 \right), \quad (2.42)$$

with

$$h_0 = \frac{1}{(1 - 6.55 \sqrt{0.5 \left(\frac{0.01013}{\log Re_\theta - 1.02} - 0.00075 \right)})}. \quad (2.43)$$

These closure relations for H_1 and the skin friction are shown in figures 2.5. It can be seen that for small values of Re_θ the Houwink relation does not resemble the Ludwig-Tillman relation at all.

The Houwink C_f formula is not functioning for small values of Re_θ , as is the case at the nose, therefore the two given closure relations for the skin friction have been combined to one. At the nose the formula of Ludwig and Tillman is taken. Away from the nose, where both formulas predict the same value for the skin friction coefficient, we switch over to the Houwink formula.

The diffusivity of turbulence will prevent the boundary layer separating.

Chapter 3

Numerical analysis

3.1 Discretisation of the outer flow

The incompressible flow along an aerofoil can be described by the solution of the potential equation. This can be done using the *panel method* devised by Smith and Hess [10]. Thin-aerofoil theory is not appropriate here: its ignoring of the aerofoil's thickness results in a significant loss in accuracy for the prediction of the pressure distribution. This is undesirable: the pressure distribution has a marked effect on the behaviour of the fluid at the edge of the boundary layer. Therefore the prediction has to be as accurate as possible.

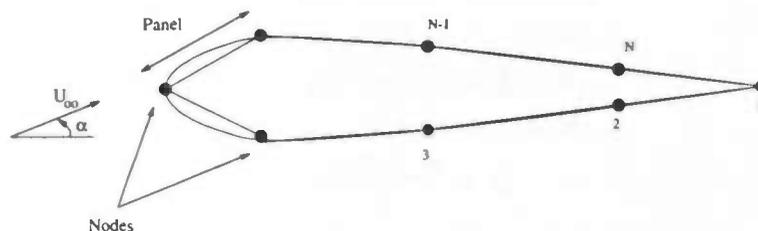


Figure 3.1: *Definition of nodes and panels*

On the body contour a certain number of points, referred to as *nodes*, are selected. A node i has the coordinates (x_i, y_i) . Connecting two adjacent nodes together with a straight line forms a *panel*, see figure 3.1. The flow is produced by a source and vortex distribution along the panels. If there are N nodes, which means also N panels, the potential (2.14) becomes:

$$\phi = U_{\infty}(x \cos \alpha + y \sin \alpha) + \sum_{j=1}^N \int_{\text{panel } j} \left[\frac{q(s)}{2\pi} \ln r - \frac{\gamma}{2\pi} \theta \right] ds. \quad (3.1)$$

The source strength $q(s)$ is taken to be constant on each panel. Therefore, $q(s) = q_i$ on panel i , with panel i defined as being between the i th and the

$(i + 1)$ th nodes. The values of the q_i 's and the vortex strength γ can be found by imposing the flow-tangency condition at N control points together with the Kutta condition.

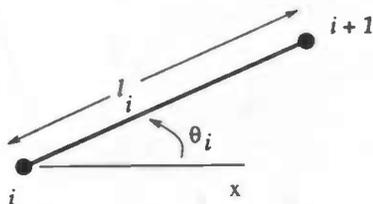


Figure 3.2: The i th panel

We define θ_i to be the angle between panel i and the x -axis and l_i to be the length of panel i . The unit vectors normal respectively tangential to the i th panel are:

$$\vec{n}_i = (-\sin \theta_i, \cos \theta_i), \quad (3.2)$$

$$\vec{t}_i = (\cos \theta_i, \sin \theta_i). \quad (3.3)$$

The velocity is infinite at the end of each panel, because the source strength is discontinuous. Therefore the control points are chosen to be in the midpoints of the panels. The coordinates of the control point at the midpoint of panel i are taken as:

$$\bar{x}_i = \frac{x_i + x_{i+1}}{2},$$

$$\bar{y}_i = \frac{y_i + y_{i+1}}{2}.$$

Hence, the velocity components at the control points are:

$$u_i = u(\bar{x}_i, \bar{y}_i),$$

$$v_i = v(\bar{x}_i, \bar{y}_i).$$

The flow-tangency condition, with the disturbance of the boundary layer taken into account, can now be written in the following way:

$$-u_i \sin \theta_i + v_i \cos \theta_i = \frac{d}{dx}(u_e \delta^*) \Big|_i, \quad (3.4)$$

and the Kutta condition as:

$$u_1 \cos \theta_1 + v_1 \sin \theta_1 = -u_N \cos \theta_N - v_N \sin \theta_N. \quad (3.5)$$

The minus sign in the above equation is due to the definition of the tangential direction. The velocity components in the midpoint of panel i , made up of source and vortex distributions, can be written as:

$$u_i = U_\infty \cos \alpha + \sum_{j=1}^N u_{s_{ij}} + \sum_{j=1}^N u_{v_{ij}}, \quad (3.6)$$

$$v_i = U_\infty \sin \alpha + \sum_{j=1}^N v_{s_{ij}} + \sum_{j=1}^N v_{v_{ij}}, \quad (3.7)$$

where $u_{s_{ij}}$, $u_{v_{ij}}$, $v_{s_{ij}}$ and $v_{v_{ij}}$ are respectively the vertical or horizontal velocities at the midpoint of panel i due to the source or vortex distribution on panel j . To find these velocities, a new local coordinate system is introduced, $(\tilde{x}_i, \tilde{y}_i)$, oriented with panel j , see figure 3.3.

$$u_{s_{ij}} = \tilde{u}_{s_{ij}} \cos \theta_j - \tilde{v}_{s_{ij}} \sin \theta_j, \quad (3.8)$$

$$v_{s_{ij}} = \tilde{u}_{s_{ij}} \sin \theta_j + \tilde{v}_{s_{ij}} \cos \theta_j, \quad (3.9)$$

$$u_{v_{ij}} = \tilde{u}_{v_{ij}} \cos \theta_j - \tilde{v}_{v_{ij}} \sin \theta_j, \quad (3.10)$$

$$v_{v_{ij}} = \tilde{u}_{v_{ij}} \sin \theta_j + \tilde{v}_{v_{ij}} \cos \theta_j, \quad (3.11)$$

where the local velocity components are written as:

$$\tilde{u}_{s_{ij}} = \frac{1}{2\pi} \int_0^{l_j} \frac{q(t)(\tilde{x}_i - t)}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt,$$

$$\tilde{v}_{s_{ij}} = \frac{1}{2\pi} \int_0^{l_j} \frac{q(t)\tilde{y}_i}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt,$$

$$\tilde{u}_{v_{ij}} = -\frac{1}{2\pi} \int_0^{l_j} \frac{\gamma\tilde{y}_i}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt = -\frac{\gamma}{2\pi} \int_0^{l_j} \frac{\tilde{y}_i}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt,$$

$$\tilde{v}_{v_{ij}} = -\frac{1}{2\pi} \int_0^{l_j} \frac{\gamma(\tilde{x}_i - t)}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt = -\frac{\gamma}{2\pi} \int_0^{l_j} \frac{(\tilde{x}_i - t)}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt.$$

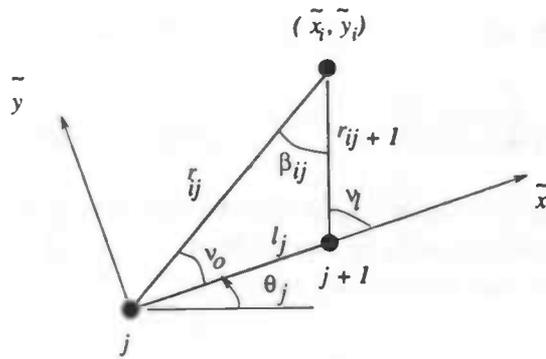


Figure 3.3: Geometric interpretation

In figure 3.3 it is shown that r_{ij} is the distance from the j th node to the middle of the i th panel and that β_{ij} is the angle subtended at the middle of the i th panel by the j th panel. The following integrals can now be calculated and given a geometric interpretation:

$$\int_0^{l_j} \frac{\tilde{y}_i}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt = \tan^{-1} \frac{\tilde{y}_i}{\tilde{x}_i - t} \Big|_{t=0}^{t=l_j} = \nu_l - \nu_0 = \beta_{ij},$$

$$\int_0^{l_j} \frac{(\tilde{x}_i - t)}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt = -\ln \sqrt{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} \Big|_{t=0}^{t=l_j} = -\ln \frac{r_{ij+1}}{r_{ij}}.$$

If $i = j$ the second integral is equal to zero, as then $r_{ij} = r_{ij+1}$. For the first integral it is not so obvious what the value of the angle β_{ii} should be. If the point $(\tilde{x}_i, \tilde{y}_i)$ approaches the panel from outside the geometry, $\beta_{ii} = \pi$. If the point would approach the panel from the other side, β_{ii} would be $-\pi$. However, as the flow is outside the aerofoil geometry β_{ii} is taken to be π .

Now, consider more carefully $\tilde{u}_{s_{ij}}$ and $\tilde{v}_{s_{ij}}$. For $i \neq j$,

$$\tilde{u}_{s_{ij}} = \frac{-q_j}{2\pi} \ln \frac{r_{ij+1}}{r_{ij}}. \quad (3.12)$$

As to be more accurate, Taylor series can be used for q if $i = j$. Taking then the Cauchy principal value of the integral gives:

$$\begin{aligned} \tilde{u}_{s_{ii}} &= \frac{1}{2\pi} \int_0^{l_i} \frac{q(t)}{(\tilde{x}_i - t)} dt, \\ &= \frac{1}{2\pi} \int_0^{l_i} \left\{ \sum_{p=0}^{\infty} \frac{-q^{(p)}(\tilde{x}_i)}{p!} (t - \tilde{x}_i)^{p-1} \right\} dt, \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{2\pi} \int_0^{l_i} \left\{ \frac{q^{(0)}(\tilde{x}_i)}{(\tilde{x}_i - t)} - q^{(1)}(\tilde{x}_i) + \frac{q^{(2)}(\tilde{x}_i)}{2!} (t - \tilde{x}_i) + \dots \right\} dt, \\
&= \frac{1}{2\pi} \left\{ q_i \ln \frac{|\tilde{x}_i - 0|}{|\tilde{x}_i - l_i|} - \frac{dq_i}{dx} l_i + \dots \right\}, \\
&= 0 - \frac{l_i}{2\pi} \frac{dq_i}{dx} + \dots
\end{aligned}$$

Taking only the first term gives:

$$\tilde{u}_{s_{ii}} = \frac{-l_i}{2\pi} \frac{dq_i}{dx}. \quad (3.13)$$

If only a constant value is used for q on each panel then:

$$\tilde{u}_{s_{ii}} = 0.$$

For $i \neq j$, $\tilde{v}_{s_{ij}}$ will be:

$$\tilde{v}_{s_{ij}} = \frac{q_j}{2\pi} \beta_{ij}, \quad (3.14)$$

and if $i = j$:

$$\tilde{v}_{s_{ii}} = \frac{q_i}{2\pi} \beta_{ii} + \frac{1}{2\pi} \int_0^{l_i} \frac{q^{(1)}(\tilde{x}_i)(t - \tilde{x}_i)\tilde{y}_i}{(\tilde{x}_i - t)^2 + \tilde{y}_i^2} dt, \quad (3.15)$$

$$= \frac{q_i}{2\pi} \beta_{ii} + \frac{\tilde{y}_i}{4\pi} \frac{dq_i}{dx} \ln \frac{(\tilde{x}_i - 0)^2 + \tilde{y}_i^2}{(\tilde{x}_i - l_i)^2 + \tilde{y}_i^2}, \quad (3.16)$$

$$= \frac{q_i}{2\pi} \beta_{ii} + 0. \quad (3.17)$$

If we introduce now

$$\bar{u}_{s_{ij}} = \frac{-1}{2\pi} \ln \frac{r_{ij+1}}{r_{ij}} \cos \theta_j - \frac{\beta_{ij}}{2\pi} \sin \theta_j,$$

$$\bar{v}_{s_{ij}} = \frac{-1}{2\pi} \ln \frac{r_{ij+1}}{r_{ij}} \sin \theta_j + \frac{\beta_{ij}}{2\pi} \cos \theta_j,$$

$$\bar{u}_{v_{ij}} = \frac{\beta_{ij}}{2\pi} \cos \theta_j - \frac{1}{2\pi} \ln \frac{r_{ij+1}}{r_{ij}} \sin \theta_j,$$

$$\bar{v}_{v_{ij}} = \frac{\beta_{ij}}{2\pi} \sin \theta_j + \frac{1}{2\pi} \ln \frac{r_{ij+1}}{r_{ij}} \cos \theta_j,$$

then (3.8)-(3.11) can be rewritten:

$$u_{s_{ij}} = q_j \bar{u}_{s_{ij}} + \Delta_{ij} \left(\frac{-l_i}{2\pi} \frac{dq_i}{dx} \cos \theta_i \right), \quad (3.18)$$

$$v_{s_{ij}} = q_j \bar{v}_{s_{ij}} + \Delta_{ij} \left(\frac{-l_i}{2\pi} \frac{dq_i}{dx} \sin \theta_i \right), \quad (3.19)$$

$$u_{v_{ij}} = \gamma \bar{u}_{v_{ij}}, \quad (3.20)$$

$$v_{v_{ij}} = \gamma \bar{v}_{v_{ij}}, \quad (3.21)$$

with $\Delta_{ij} = 0$ if $i \neq j$ and $\Delta_{ij} = 1$ if $i = j$. The flow-tangency condition and the Kutta condition can now be put together in the following form:

$$C \frac{d\vec{q}}{dx} + A\vec{q} = \vec{b}, \quad (3.22)$$

with

$$A_{ij} = -\bar{u}_{s_{ij}} \sin \theta_i + \bar{v}_{s_{ij}} \cos \theta_i, \quad i = 1, \dots, N \quad \text{and} \quad j = 1, \dots, N$$

$$A_{i,N+1} = \sum_{j=1}^N (-\bar{u}_{v_{ij}} \sin \theta_i + \bar{v}_{v_{ij}} \cos \theta_i), \quad i = 1, \dots, N$$

$$A_{N+1,j} = \sum_{k=1, N} (\bar{u}_{s_{kj}} \cos \theta_k + \bar{v}_{s_{kj}} \sin \theta_k), \quad j = 1, \dots, N$$

$$A_{N+1,N+1} = \sum_{k=1, N} \sum_{j=1}^N (\bar{u}_{v_{kj}} \cos \theta_k + \bar{v}_{v_{kj}} \sin \theta_k),$$

$$b_i = -U_\infty \sin(\alpha - \theta_i) + \frac{d}{dx}(u_e \delta^*)|_i, \quad i = 1, \dots, N$$

$$b_{N+1} = -U_\infty \cos(\theta_1 - \alpha) - U_\infty \cos(\theta_N - \alpha),$$

$$\vec{q} = (q_1, q_2, \dots, q_N, \gamma)^T,$$

$$C_{ij} = 0, \quad i \neq j \quad \text{and} \quad j = 1, N$$

$$C_{11} = \frac{-1}{2\pi} l_1,$$

$$C_{NN} = \frac{-1}{2\pi} l_N.$$

Equation (3.22) can be simplified by equating matrix C to zero.

The term $\frac{\partial q}{\partial x}$ will be discretized downwind, as will be explained later in this section. However, since there is no wake, downwind discretisation is not possible for the two trailing edge points, since two undefined points would be needed.

Even so, the coupling with the boundary layer will not take place in these last two points. This is done because the upper and lower surfaces will suddenly meet each other at the trailing edge if there is no wake region present. Hence, we see reason for the simplification. The two trailing edge points are obtained by extrapolation as will be explained in more detail at the end of chapter 4. The following equation remains:

$$\boxed{A\vec{q} = \vec{b}.} \quad (3.23)$$

The matrix A contains the real geometry of the profile. The influence of the boundary layer is included in vectors \vec{b} and \vec{q} . Therefore the vectors \vec{q} and \vec{b} can be divided up into two parts, an undisturbed part and the disturbance:

$$\vec{q} = \vec{q}_0 + \vec{q}^*, \quad (3.24)$$

$$\vec{b} = \vec{b}_0 + \vec{b}^*, \quad (3.25)$$

with

$$\begin{aligned} b_{0i} &= -U_\infty \sin(\alpha - \theta_i), & i = 1, \dots, N \\ b_{0N+1} &= -U_\infty \cos(\theta_1 - \alpha) - U_\infty \cos(\theta_N - \alpha), \\ b_i^* &= \frac{d}{dx}(u_e \delta^*)|_i, & i = 1, \dots, N \\ b_{N+1}^* &= 0. \end{aligned}$$

The goal is now to find a relation between the tangential velocity at the edge of the boundary layer, u_e coming from the outer flow, and the displacement thickness δ^* .

After some manipulation it can be shown that u_e is also divided into two parts, namely the constant value u_{e0} , which is the undisturbed part of u_e due to the source and vortex distributions q_0 and γ_0 , and u_e^* , the disturbance due to the source and vortex distributions q^* and γ^* :

$$u_e = u_{e0} + u_e^*. \quad (3.26)$$

In appendix A.1 it can be seen how expressions for u_{e0} and u_e^* are derived. The velocity u_e^* is dependent on the displacement thickness:

$$u_{e_i}^* = \sum_{k=1}^N X(i, k) \frac{\partial}{\partial x}(u_e \delta^*) \Big|_k - \frac{l_i}{2\pi} \frac{dq_i}{dx}. \quad (3.27)$$

The last term in equation (3.27) will disappear if constant values for q are used. Then,

$$u_{e_i}^* = \sum_{k=1}^N X(i, k) \frac{\partial}{\partial x} (u_e \delta^*) \Big|_k. \quad (3.28)$$

The discretisation of these formulas, (3.27) and (3.28) is as follows: the term $\frac{d}{dx}(u_e \delta^*)$ is discretized upwind and the term $\frac{dq}{dx}$ is discretized downwind (see appendix A.2). The discretisation of $\frac{dq}{dx}$ is done downwind as to increase the value of the diagonal elements of the resulting panel matrix E . After the discretisation, the general formulation for u_e in point i is:

$$u_{e_i} = u_{e0_i} + \sum_{k=1}^N E(i, k) u_{e_k} \delta_k^*. \quad (3.29)$$

As two different equations have been found, (3.27) and (3.28), also two panel matrices E have been constructed. Panel matrix E_0 , coming from equation (3.28) and panel matrix E_1 , derived from equation (3.27).

For a point i on the upper surface away from the leading and trailing edge matrices E have the following properties:

- positive diagonal elements: $E_{ii} > 0$
- positive lower diagonal elements: $E_{ii-1} > 0$
- all other elements are negative: $E_{ij} < 0$ for $j \neq i, i-1$
- not symmetric: $E_{ij} \neq E_{ji}$
- the diagonal and lower diagonal elements are almost equal: $E_{ii} \approx E_{ii-1}$
- $|E_{ii} + E_{ii-1}| \geq \sum_{j \neq i, i-1} |E_{ij}|$

It is clear that the panel matrices that have been found are not positive definite. If the control points would have been taken in the endpoints, instead of in the midpoints of the panels, the panel matrices might have been more symmetric, like the Hilbert matrix derived with thin-aerofoil theory (appendix A.3).

The irregular structure of the coefficients of E may give rise to problems if separation occurs, as the system is less stable.

3.2 Discretisation of the boundary layer equations

3.2.1 Laminar flow

The laminar boundary layer equations can be reduced to the non-linear momentum-integral equation:

$$\frac{C_f}{2} = \frac{\partial}{\partial x} \left(\frac{\delta^*}{H} \right) + \left(\frac{2}{H} + 1 \right) \frac{\delta^*}{u_e} \frac{\partial u_e}{\partial x}. \quad (3.30)$$

Discretisation of the x -derivatives in this equation is performed upwind. Hence, for a boundary station i on the upper surface the momentum-integral equation can be discretized as follows:

$$\frac{1}{h_i} \left(\frac{\delta_i^*}{H_i} - \frac{\delta_{i-1}^*}{H_{i-1}} \right) + \left(\frac{2}{H_i} + 1 \right) \frac{\delta_i^*}{u_{e_i}} \frac{u_{e_i} - u_{e_{i-1}}}{h_i} = \frac{C_{f_i}}{2}, \quad (3.31)$$

with $h_i = 0.5(l_i + l_{i-1})$, where l_i is the length of panel i . The equation is closed with the algebraic relations for C_{f_i} and H_i given previously in section 2.4.1.

3.2.2 Turbulent flow

The turbulent case requires one equation more than the laminar case. This is the entrainment equation, which reads:

$$E = \frac{\partial}{\partial x} (u_e (\delta - \delta^*)) = \frac{\partial}{\partial x} (u_e \theta H_1). \quad (3.32)$$

Once again, the x -derivatives are approximated with first order upwind finite difference quotients. At a boundary layer station i on the upper surface the following discretized formulation applies:

$$E_i = \frac{1}{h_i} \left(\frac{u_{e_i} \delta_i^* H_{1_i}}{H_i} - \frac{u_{e_{i-1}} \delta_{i-1}^* H_{1_{i-1}}}{H_{i-1}} \right). \quad (3.33)$$

As before, $h_i = 0.5(l_i + l_{i-1})$ and for H_{1_i} , E_i and C_{f_i} the algebraic closure relations from section 2.4.2 will be used. The momentum-integral equation is discretized in the same way as is done in the laminar case.

Chapter 4

Viscous-inviscid interaction

Two relations between the velocity at the edge of the boundary layer u_e and the displacement thickness δ^* have been found in chapter 3. One relation comes from the outer flow equations and the other one comes from the laminar or turbulent boundary layer equations. These two relations can be written symbolically as:

outer flow:

$$u_e = E[\delta^*], \quad (4.1)$$

boundary layer:

$$u_e = B[\delta^*]. \quad (4.2)$$

The classical way to solve these two equations is first to calculate the u_e coming from the outer flow and, then, use the new u_e to calculate a new δ^* . This method of calculation is in agreement with the theory of matched asymptotic expansions. This method is referred to as the *direct method* as it makes use of a sequential way of calculating based on a direct hierarchy between the outer flow and the boundary layer [1]:

$$\begin{aligned} u_e^{(n)} &= E[\delta^{*(n-1)}], \\ \delta^{*(n)} &= B^{-1}[u_e^{(n)}]. \end{aligned}$$

However, this direct method, is unable to handle separation. The inverse B^{-1} does not have to exist. This problem can be avoided by using a so-called *inverse method* [4]:

$$\begin{aligned} \delta^{*(n)} &= E^{-1}[u_e^{(n-1)}], \\ u_e^{(n)} &= B[\delta^{*(n)}]. \end{aligned}$$

The hierarchy has changed. Now u_e is prescribed for the outer flow equations instead of the displacement thickness. This time, inverse E^{-1} is well behaved.

The inverse method is very slow to converge, therefore a *semi-inverse method* was invented by Le Balleur [5]. The outer flow is calculated in a direct way and the boundary layer is calculated in an inverse way. Both regions have δ^* prescribed. The new δ^* is calculated by a special relaxation formula in which λ is the relaxation parameter.

$$\begin{aligned} u_{e_{OF}}^{(n)} &= E[\delta^{*(n-1)}], \\ u_{e_{BL}}^{(n)} &= B[\delta^{*(n-1)}], \\ \delta^{*(n)} &= \delta^{*(n-1)} + \lambda(u_{e_{BL}}^{(n)} - u_{e_{OF}}^{(n)}). \end{aligned}$$

The indicator *OF* stands for outer flow and the indicator *BL* for boundary layer. In the laminar case the relaxation parameter is chosen to be $Re^{-\frac{1}{2}}$ since δ^* is then of order $Re^{-\frac{1}{2}}$.

The iterations are continued until the global error is small enough:

$$\sqrt{\sum_i (u_{e_{BL_i}} - u_{e_{OF_i}})^2} < 1 \times 10^{-2}.$$

An even more faster method has been developed by Veldman, termed the *quasi-simultaneous method* [8]. This method avoids an iterative way of computing as much as possible. An interaction law *I* is introduced, which is an approximation of the interaction between the inviscid and viscous flow. The choice of *I* is based on operator *E* and should be made in a way that it can be used as a boundary condition for the boundary layer equation: $u_e = I[\delta^*]$. The interaction law *I* will allow the most important part of the inviscid interaction to be considered simultaneously with the development of the viscous solution. Symbolically this is written as:

$$\begin{cases} u_e^{(n)} - I[\delta^{*(n)}] = E[\delta^{*(n-1)}] - I[\delta^{*(n-1)}], \\ u_e^{(n)} - B[\delta^{*(n)}] = 0, \end{cases}$$

$$u_e^{(n)} - E[\delta^{*(n)}] = 0 \quad (\text{direct}).$$

The choice of *I* does not have any influence on the final solution. After convergence the operator *I* disappears as $I[\delta^{*(n)}] \approx I[\delta^{*(n-1)}]$, so equation (4.1) is satisfied. The operator *I* only has an influence on the speed of convergence. The better *I* is approximated by *E*, the faster the system converges.

In chapter 3 the relation for the outer flow has been derived:

$$u_{e_i} = u_{e0_i} + \sum_{j=1}^N E(i, j) u_{e_j} \delta_j^*.$$

Notice that in this equation E is the constructed panel matrix. With this, the interaction law $u_e = I[\delta^*]$ for a point i is chosen to be:

$$\begin{aligned} u_{e_i} &= u_{e_{0i}} + E(i, i-1)u_{e_{i-1}}\delta_{i-1}^* + E(i, i)u_{e_i}\delta_i^* + E(i, i+1)u_{e_{i+1}}\delta_{i+1}^*, \\ &= u_{e_{0i}} + \sum_{j=i-1}^{i+1} I(i, j)u_{e_j}\delta_j^*, \end{aligned}$$

where $I(i, j)$ is a tridiagonal matrix with on the diagonal the diagonal values of $E(i, j)$ and on the upper and lower diagonal, the upper and lower diagonal values of $E(i, j)$. In more detail the system during the n th downstream march can be written for a point i as:

$$\begin{cases} u_{e_i}^{(n)} = u_{e_{0i}} + \sum_{j=1}^N E(i, j)u_{e_j}^{(n-1)}\delta_j^{*(n-1)} \\ \quad + I(i, i-1)[u_{e_{i-1}}^{(n)}\delta_{i-1}^{*(n)} - u_{e_{i-1}}^{(n-1)}\delta_{i-1}^{*(n-1)}] + I(i, i)[u_{e_i}^{(n)}\delta_i^{*(n)} - u_{e_i}^{(n-1)}\delta_i^{*(n-1)}] \\ \quad + I(i, i+1)[u_{e_{i+1}}^{(n)}\delta_{i+1}^{*(n)} - u_{e_{i+1}}^{(n-1)}\delta_{i+1}^{*(n-1)}], \\ u_{e_i}^{(n)} = B[\delta_i^{*(n)}]. \end{cases}$$

However, the interaction law is not really efficient in this way. It can be seen that the term $I(i, i+1)$ is not used at all: the new values of $u_{e_{i+1}}^{(n)}$ and $\delta_{i+1}^{*(n)}$ have not yet been calculated, so the old values of the $(n-1)$ th downstream march have to be used. This has as a result that the whole last part is equal to zero. In order to take full advantage of the interaction law an extra loop is used where pointwise local Gauss-Seidel iterations are performed. The system can now be written, the unmodified values being indicated by the index \bar{n} , as:

$$\begin{cases} u_{e_i}^{(n)} = E[\delta_i^{*(\bar{n}-1)}] - I[\delta_i^{*(\bar{n}-1)}] \\ \quad + I(i, i-1)u_{e_{i-1}}^{(n)}\delta_{i-1}^{*(n)} + I(i, i)u_{e_i}^{(n-1)}\delta_i^{*(n)} + I(i, i+1)u_{e_{i+1}}^{(n-1)}\delta_{i+1}^{*(n-1)}, \\ u_{e_i}^{(n)} = B[\delta_i^{*(n)}]. \end{cases}$$

By putting both equations together $u_{e_i}^{(n)}$ can be eliminated. The resulting equation is left to be solved with Newton's method in order to obtain $\delta_i^{*(n)}$. The new $u_{e_i}^{(n)}$ is finally calculated with the outer flow equation in a direct way.

At high angles of attack underrelaxation can be necessary, as to make the system converge. This is done by multiplying the tridiagonal matrix $I(i, j)$ with the relaxation parameter $\omega \geq 1$:

$$u_{e_i} = u_{e_{0i}} + \sum_{j=i-1}^{i+1} \omega I(i, j)u_{e_j}\delta_j^*.$$

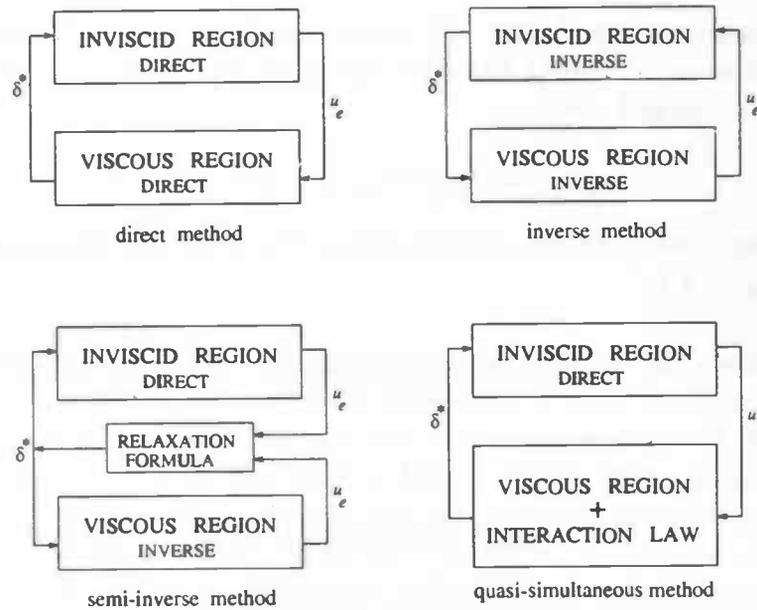


Figure 4.1: Coupling methods

The quasi-simultaneous process can be described as follows:

- calculation of $E[\delta^{*(n-1)}] - I[\delta^{*(n-1)}]$ for every station i
 - do k -times extra loop, using Gauss-Seidel
 - * calculation of $\delta_i^{*(n)}$ and $u_{e_i}^{(n)}$ using Newton's method until local error, $|u_{e_i}^{(n)} - u_{e_i}^{(n-1)}| < \varepsilon$
 - until k -times done
- continue until the global error:

$$\max_i |u_{e_i}^{(n)} - u_{e_i}^{(n-1)}| < 1 \times 10^{-5}$$

If I is equal to E the highest rate of convergence is achieved. The method can then be called the *simultaneous method*:

$$\begin{cases} u_e^{(n)} - E[\delta^{*(n)}] = 0, \\ u_e^{(n)} - B[\delta^{*(n)}] = 0. \end{cases}$$

Again, $u_e^{(n)}$ can be eliminated by putting both equations together so as to calculate $\delta^{*(n)}$.

The simultaneous method does not make use of a k -loop. Because of this a smaller global error is needed than for the quasi-simultaneous method as to obtain the real converged solution:

$$\max_i |u_{e_i}^{(n)} - u_{e_i}^{(n-1)}| < 1 \times 10^{-7}.$$

Again, underrelaxation can be used if necessary. The discussed coupling methods are shown in figure 4.1.

For all methods the coupling near the stagnation point at the nose is done with the direct method. This is done as to avoid possible convergence problems which may occur at the nose because of the irregular structure of E .

In the two trailing edge points N and 1 , there will be no coupling at all. As there is not a wake region programmed, the upper surface and the lower surface suddenly would meet each other there. This causes problems for the calculation of the downwind derivatives in the trailing edge points: points in the wake region would be necessary, but they do not exist. Without a wake region, the velocity u_e will decrease strongly near the trailing edge, as a second stagnation point is seen there by the undisturbed outer flow. In reality there is no second stagnation point anymore as a result of the influence of the boundary layer.

To avoid the trailing edge problems, the values in the endpoints are obtained by extrapolation. For u_e , H and C_f the values of the station before are taken. For δ^* the following extrapolation is used as to change the sharp trailing edge in a infinite trailing edge. The second stagnation point will no longer occur, see figure 4.2 below. The displacement thickness in the trailing edge points becomes:

$$\delta_1^* = \delta_2^* + \bar{y}_2 - \bar{y}_1, \quad (4.3)$$

$$\delta_N^* = \delta_{N-1}^* + \bar{y}_{N-1} - \bar{y}_N, \quad (4.4)$$

with \bar{y}_i , as before defined as the y -coordinate of the midpoint of panel i . With these relations the upper and lower displacement bodies become parallel and the flow is able to leave the trailing edge smoothly. This is called the smoothing effect of the boundary layer.

The displacement thickness in the above relation is chosen to be parallel to the camberline. Another alternative would have been to take δ^* parallel to the freestream.

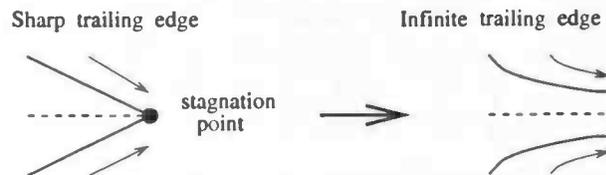


Figure 4.2: *Smoothing effect boundary layer*

Chapter 5

Results and discussion

5.1 Aerodynamic coefficients

First some standard non-dimensional coefficients are introduced which will be used to display the results. These aerodynamic coefficients express the behaviour of the wing:

pressure coefficient:

$$C_p \equiv \frac{p - p_\infty}{\frac{1}{2}\rho U_\infty^2}, \quad (5.1)$$

lift coefficient:

$$C_l \equiv \frac{L'}{\frac{1}{2}\rho U_\infty^2 c}, \quad (5.2)$$

drag coefficient:

$$C_d \equiv \frac{D'}{\frac{1}{2}\rho U_\infty^2 c}, \quad (5.3)$$

in which L' , the lift, is a force acting in the direction perpendicular to the freestream direction. The force acting in the direction of the freestream is the drag D' , see figure 5.1. Assuming C_p to be constant over each panel i , it can be derived, using Bernoulli's equation that in this case:

$$C_{p_i} = 1 - u_{\epsilon_i}^2, \quad (5.4)$$

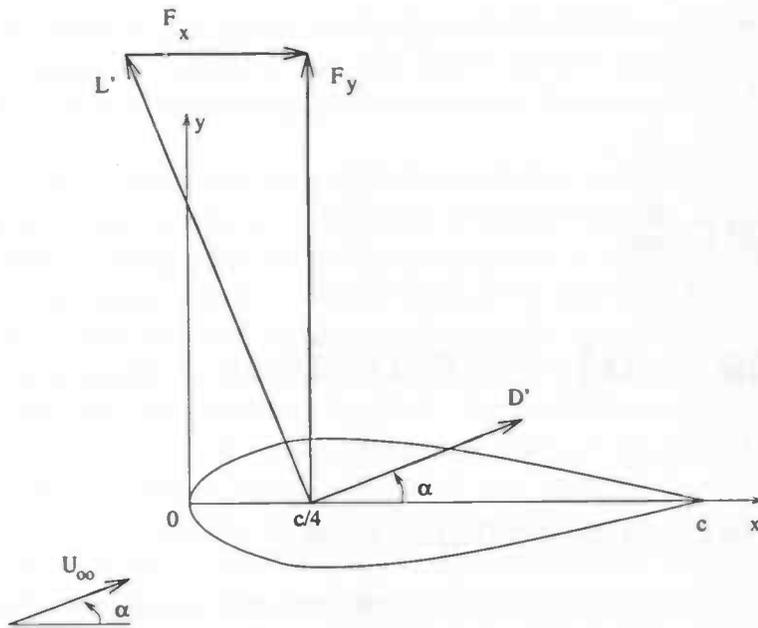


Figure 5.1: Forces on an aerofoil

As the skin friction and pressure coefficients are known for each panel, the total forces in the x and y direction can be calculated. As before θ_i is the angle between panel i and the x -axis and l_i is the length of panel i . Then,

$$F_x = \sum_i \{C_{p_i} l_i \sin \theta_i + \text{sign} \frac{1}{2} C_{f_i} l_i \cos \theta_i\}, \quad (5.5)$$

$$F_y = \sum_i \{-C_{p_i} l_i \cos \theta_i + \text{sign} \frac{1}{2} C_{f_i} l_i \sin \theta_i\}, \quad (5.6)$$

where $\text{sign} = +1$, if panel i is on the upper surface and $\text{sign} = -1$, if panel i lies on the lower surface. At an angle of attack α , the lift L' and the drag D' are:

$$L' = F_y \cos \alpha - F_x \sin \alpha, \quad (5.7)$$

$$D' = F_x \cos \alpha + F_y \sin \alpha. \quad (5.8)$$

The lift and drag act on a fixed point of the aerofoil, called the aerodynamic center, which is for a two-dimensional aerofoil in incompressible potential flow located at the quarter-chord point.

5.2 Laminar results compared with turbulent results

To have a laminar flow over an aerofoil a thin profile is needed together with an angle of attack close to zero. If these two conditions are not satisfied transition will take place and the flow will become turbulent. The programmed code can only handle fully laminar flow or fully turbulent flow.

Use of the laminar code for a thin NACA 0004 profile produces the following results. For $Re = 1 \times 10^6$, $\alpha = 0^\circ$, the flow is fully laminar. If the angle of attack is increased to 1° the stagnation point has moved slightly forward on the lower surface and a small separation region becomes apparent at the trailing edge. In figure 5.2a it can be seen that the displacement thickness δ^* grows smoothly and is at an angle of attack of 1° no longer symmetric. The displacement thickness is taken negative right from the stagnation point, which is on the lower surface if $\alpha = 0^\circ$. This is done as to make a clear difference between the values left and right from the stagnation point. The pressure distribution displays a very slight adverse gradient, see figure 5.2b.

Increasing the angle of attack further makes the value of the shape factor at the leading edge increase towards 3.2, and the value of the skin friction at the leading edge goes down towards zero. This can be seen in figures 5.3a and 5.3b.

For an angle larger than 2.09° the program diverges and crashes suddenly. As the profile is very thin, the curve the flow has to make to go around the nose at already a small angle of attack with the stagnation point no longer on the nose but on the lower surface, is very sharp. The streamline therefore wants to separate from the surface as there is not enough viscosity to keep it attached. This phenomenon is called leading edge separation. At the point of separation the skin friction coefficient will be equal to zero.

Physically, transition will start taking place just after separation to go over in a turbulent flow. The extra viscosity of the turbulence would then make the streamline reattach to the surface.

As the program however does not have the transition and makes use of the direct method at the nose, there is nothing that can bring the streamline back to the surface.

Because of this, as soon as the leading edge separation takes place, the program crashes as it is unable to handle it.

If the Re number is increased the program crashes earlier, for $Re = 3 \times 10^6$ at $\alpha = 1^\circ$. This happens because the speed with which the flow has to make the curve around the nose is larger, what will make it more difficult for the streamline to follow the surface. Therefore it separates earlier.

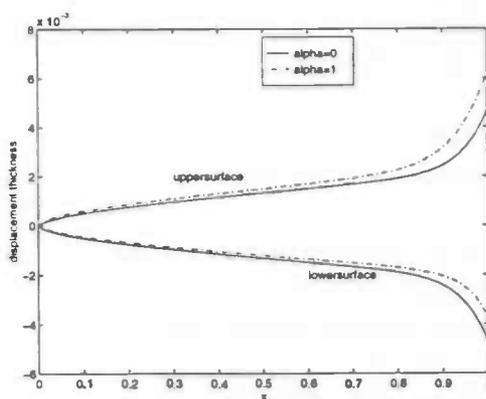
If the laminar code is used for a thick aerofoil, like a NACA 0012 at $Re = 1 \times 10^6$, it can be seen that already at an angle of attack of zero there is a large region of trailing edge separation, started halfway the upper and lower surface,

see figure 5.4a. Also δ^* has grown much, figure 5.4b. If this angle is only slightly increased the point of separation on the upper surface moves forward and the separation has become too strong for the boundary layer model and the program crashes.

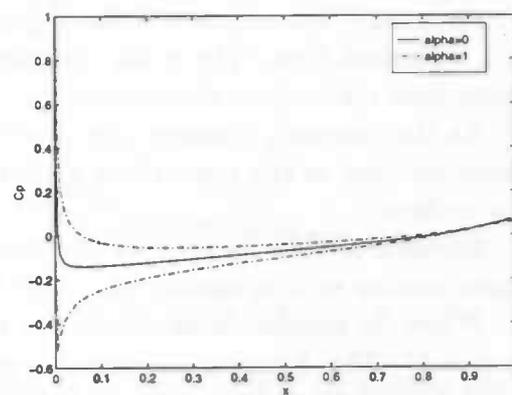
If the same calculations are done with the turbulent code, the results should be quite different, because of the extra viscosity of the turbulence. For the case of the thin NACA 0004 leading edge separation starts now at a higher angle of attack, $\alpha \approx 3^\circ$. The flow stays attached longer. It is to be expected that reattachment should take place too, however as the coupling at the nose is done with the direct method, which is unable to handle separation, the program crashes as soon as the point of leading edge separation is reached. For turbulent separation the shape factor is around 2.4. For the calculated cases there is no sign of trailing edge separation. The turbulence keeps it attached. See figures 5.6a and 5.6b.

This can also be perfectly seen if the turbulent code is used for the NACA 0012. At an angle of attack zero there is no separation at all. The streamline stays nicely attached, whereas in the laminar case there is a large region of separation, see again figure 5.4a. It can be noticed that at the nose of a thick aerofoil the laminar and the turbulent codes do not differ much. This can be seen in figures 5.4 and 5.5, in which the displacement thickness, skin friction and pressure coefficient are displayed. Therefore assuming fully turbulent flow will not effect the results.

If the obtained results for the NACA 0012 at zero angle of attack are compared with experimental results [9], it can be seen that the results from the turbulent code are in good comparison with reality, as was to be expected. Therefore, as experimentally the flow is turbulent, the turbulent code will be used for the other computations.

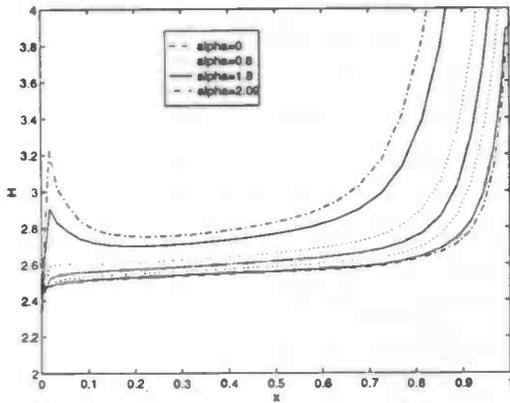


5.2.a: Displacement thickness

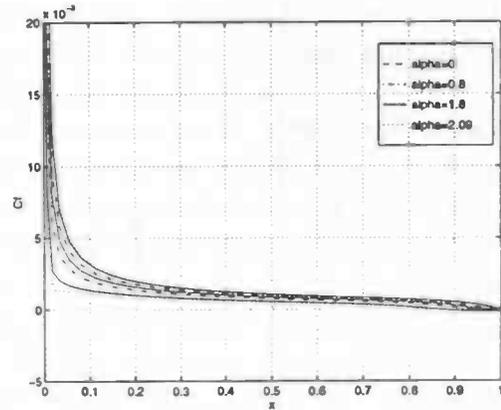


5.2.b: Pressure distribution

Figure 5.2: Laminar results NACA 0004 at $Re = 1 \times 10^6$

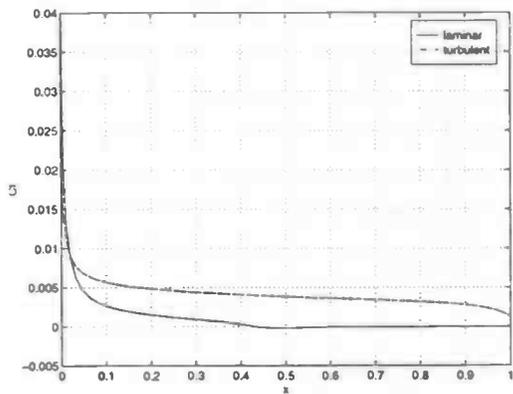


5.3.a: H increasing at the leading edge

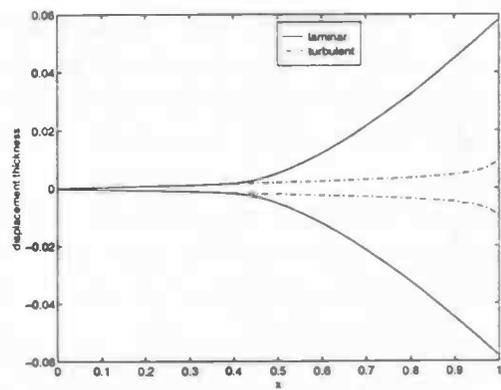


5.3.b: C_f decreasing at the leading edge

Figure 5.3: Laminar case NACA 0004 approaching leading edge separation



5.4.a: Skin friction



5.4.b: Displacement thickness

Figure 5.4: NACA 0012, laminar compared with turbulent

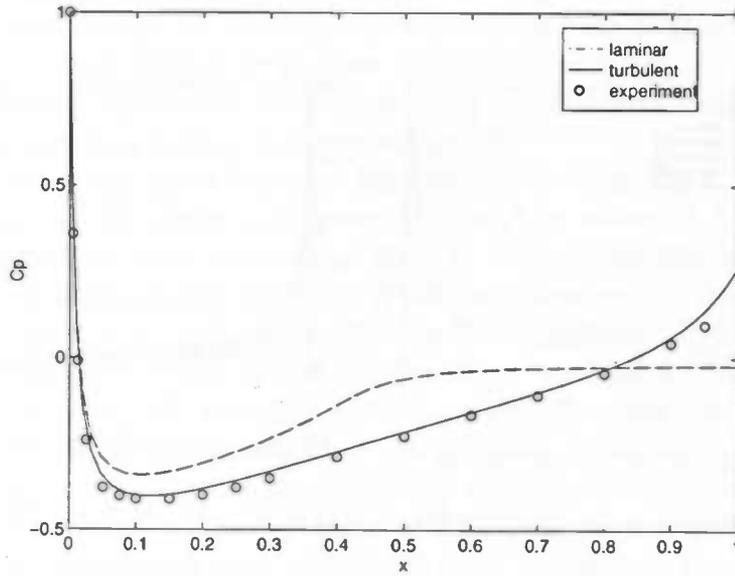
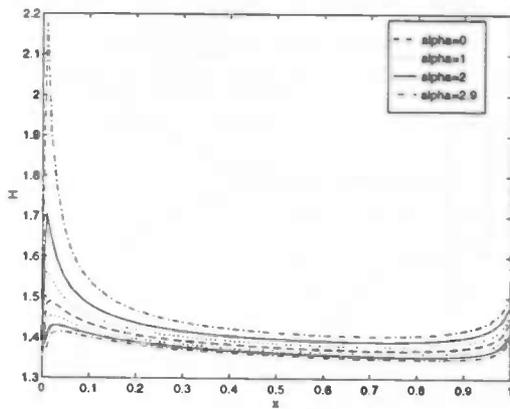
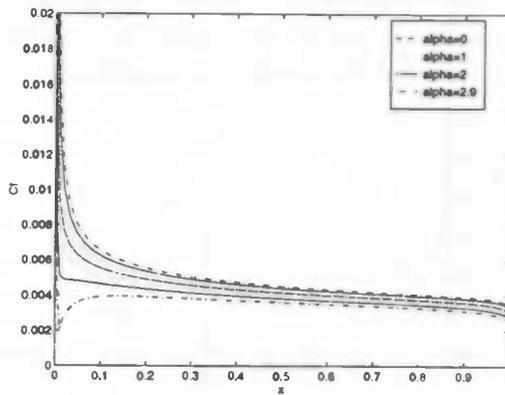


Figure 5.5: C_p for NACA 0012, laminar, turbulent and experiment



5.6.a: H increasing at the leading edge



5.6.b: C_f decreasing at the leading edge

Figure 5.6: Turbulent case NACA 0012, approaching leading edge separation

5.3 Turbulent results

In the test case discussed next a turbulent flow past a NACA 0012 is calculated at $Re = 2267000$ and at different angles of attack. The quasi-simultaneous coupling method is used and the total number of points is 120. The calculations start at an angle of zero and increase to the angle at which the trailing edge separation has become too strong for the code to handle.

The initial values used for u_e and δ^* in all the stations at an angle of attack $\alpha = 0^\circ$, are the theoretical values of a laminar flow past a flat plate found by Blasius. For the shape factor H the initial value 1.35 is taken in all the stations, except for the stations at the nose, as it is known from prior results that the initial value should be around 1.3 to 1.4. With this 'laminar flat plate' initialisation the program takes a lot of time to converge. For angles of attack higher than zero, the results for u_e , δ^* and H , obtained at a smaller angle are used as the initial values. The initial values that are used often have to come from a test case with an angle 1 or 2 degrees smaller than the case that is to be calculated. With an initialisation coming from prior obtained results the program converges a lot faster.

For test cases at a high angle of attack, like angles approaching the angle at which the maximum lift coefficient is obtained, $\alpha_{Cl_{max}}$, or angles even higher than this $\alpha_{Cl_{max}}$, a lot more time is required to converge. The calculations have to be done into the region of separation. At these high angles often underrelaxation is necessary to make the code converge. The convergence rate and used time will be discussed in more detail in section 5.5

In figure 5.8a the behaviour of the displacement thickness can be seen. At $\alpha = 0^\circ$ the flow is symmetric: the values obtained on the upper and lower surface for δ^* are the same. If the aerofoil is put under an angle of attack the stagnation point moves forward on the lower surface and the flow has to make a little curve around the nose. Because of this, the displacement thickness on the upper surface grows much more, as δ^* starts growing from the stagnation point, where it has its lowest value. The displacement thickness on the lower surface on the other hand grows less. At high angles of attack the displacement thickness on the upper surface grows very steeply near the trailing edge and reaches a very high value. For α higher than 11.8° the model is unable to handle the problem and the program crashes.

In figure 5.8b the skin friction is shown at different angles of attack. It can be seen that at $\alpha = 9^\circ$ a small separation region has formed near the trailing edge. For higher angles the separation point moves towards the leading edge and the region of separation grows. In figure 5.8a it can be seen that it is at the separation point that the displacement thickness starts growing steeper than before. The skin friction obtains its highest value at the stagnation point, as there the velocity gradient is very large. Furthermore, the skin friction graph shows the moving of the stagnation point clearly. The peak of the graph moves

to the right at increasing angle of attack.

Figure 5.9a shows the shape factor H . Small wiggles occur at the nose. They are caused by the change of coupling method: around the stagnation point the direct method is used, slightly away from the stagnation point the code switches over to the (quasi-) simultaneous method. The other figures display no visible wiggles, as the coupling is done between u_e and δ^* .

It can be seen that, apart from the leading and trailing edges, H almost has a constant value on the upper and lower surface. The value for H on the lower surface is around 1.35 (this observation has led to choosing this initial value at $\alpha = 0^\circ$). For increasing α the value for H on the lower surface does not change much. On the upper surface however, the value for H increases at each increasing angle with a maximum value at the trailing edge. At high angles of attack H starts growing very rapidly already before the point of separation. At the point of separation H has a value around 3.

The velocity at the edge of the boundary layer u_e is shown in figure 5.9b. It can be seen that at increasing angle of attack there is a so-called suction peak growing at the leading edge. After this peak the flow on the upper surface decelerates. At high angles of attack the system does not function very well and gives a value for u_e of 1 near the trailing edge on the upper surface.

In the end points on the upper and lower surface the velocities have the same value. The Kutta condition is satisfied. The values at the lower surface, right from the stagnation point are again taken negative as to make the difference between upper and lower surface.

In figure 5.7 the effect of the displacement thickness at an angle of attack of 5° is shown. The profile has really changed for the outer flow. A remarkable change is the change of trailing edge. It is no longer sharp but it has become infinite. The second stagnation point at the trailing edge has disappeared and the flow can leave the aerofoil smoothly. This demonstrates the smoothing effect of the boundary layer.

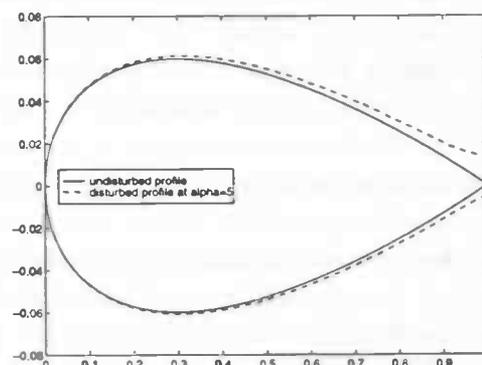
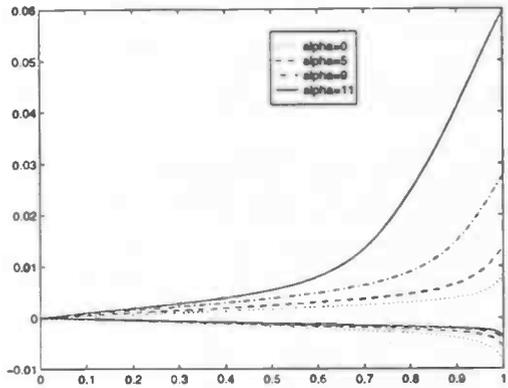
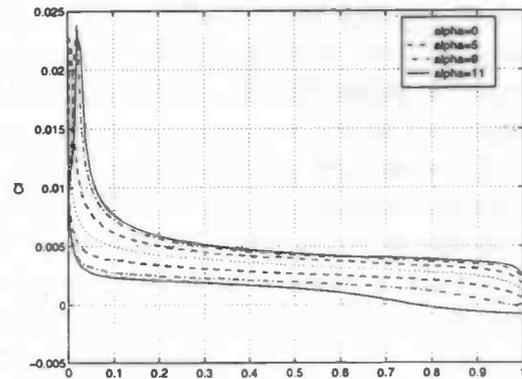


Figure 5.7: *Effect boundary layer*

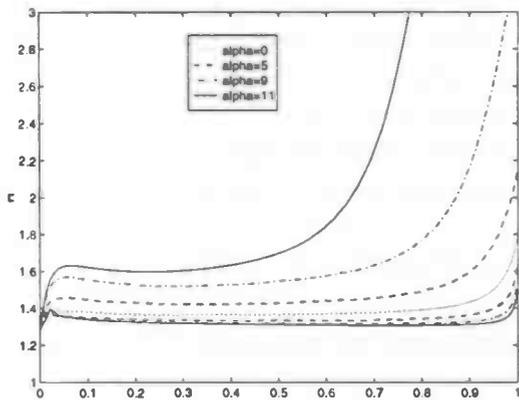


5.8.a: *Displacement thickness*

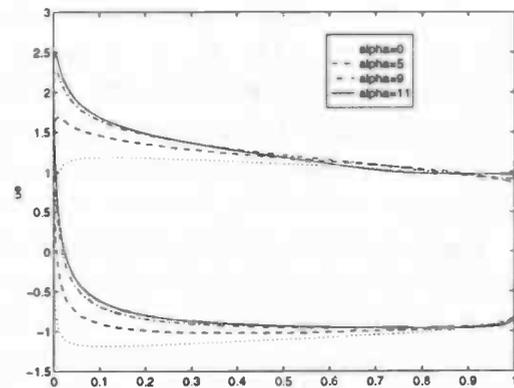


5.8.b: *Skin friction*

Figure 5.8: *Turbulent results NACA 0012 at $Re = 2267000$*



5.9.a: *Shape factor*



5.9.b: *Tangential velocity at the edge of the BL*

Figure 5.9: *Turbulent results NACA 0012 at $Re = 2267000$*

5.4 Validation

The results obtained by the programmed code Viscous Inviscid Boundary Layer Interaction (VIBLI) have been compared with those obtained from a commercial Navier-Stokes code Rampant. One of the CFD software packages of Fluent Inc., Rampant can employ either a two-equation $k - \epsilon$ turbulence model or a full Reynolds stress Model. For the aerofoil case computed here, the $k - \epsilon$ model was used.

To be able to resolve viscous effects, such as boundary layer growth, high cell concentrations are needed normal to the surface close to the boundary. This is necessary so as to resolve the gradients in the flow normal to the surface. The streamwise gradients in the flow are not as critical or as large as the normal gradients, hence not so many cells are required in the streamwise direction. Therefore the ideal grid for this case would be one where cells would have a high aspect ratio close to the surface. The aspect ratio is defined here as the length of the cell divided by its height.

However, for complex geometries it is much easier to use an unstructured grid instead. The triangle-shaped cells of an unstructured grid easily map around the surface. To get satisfactory convergence with an unstructured grid the cells have to be close to an aspect ratio of one, otherwise the numerical scheme may be unstable. This means that to be able to solve the boundary layer adequately with an unstructured grid and to have a stable convergence, very small cells are needed on the boundary. This implies that to solve the Navier-Stokes equations massive numbers of cells are needed on the boundary, which in turn results in high computational time.

The code Rampant has been written to enable solution of the Navier-Stokes equations for complex geometries. It therefore makes use of an unstructured grid, see figure 5.17. So as to avoid high computation times only 1000 cells have been used on the boundary, which means the first cell on the boundary is approximately 1mm high. The cells are not small enough for an accurate prediction of the viscous effects, as can be clearly seen in the results given by Rampant.

Rampant was run for a NACA 0012 aerofoil at $Re = 2267000$ and at a Mach number $M = 0.1$, whereas VIBLI deals with an incompressible flow $M = 0$. The effect of this small difference in Mach number is however negligible. The Rampant program has calculated cases at different angles of attack and each calculation took at least 24 hours.

As can be seen in figures 5.10 to 5.13, the C_p results at different angles of attack obtained by VIBLI are in reasonably good agreement with the results obtained by Rampant. At the nose the Rampant results show some oscillations. These may be caused by the fact that the clustering at the nose was not fine enough. However, it may also be a result of a lack of accuracy in the original aerofoil definition, when the initial grid was generated. Near the trailing edge

the C_p results produced by Rampant have lower values than the results obtained with VIBLI. This difference is an effect of the influence of the wake, which is not modelled in VIBLI. In the displayed results it seems that the Kutta condition is not satisfied. However, the C_p values in the two trailing edge points are equal. The upper and lower C_p graphs crossing each other just before the trailing edge is a result of the used extrapolation for δ^* in those two points.

The C_f results obtained by both programs differ significantly, see figure 5.14. At the nose the Rampant code predicts leading edge separation, $C_f = 0$, whereas VIBLI does not show any sign of possible leading edge separation. This difference is caused by the fact that the Rampant code starts off with a laminar flow at the nose which will not be able to stay attached. However, at the point of separation, transition will take place and the flow will become turbulent. The VIBLI code starts with a turbulent flow immediately from the nose. Compared to laminar flow, a turbulent flow is more energetic and effectively has higher viscosity and so may thus stay attached. The C_f results away from the leading edge are in better agreement.

Trailing edge separation is predicted by Rampant as starting at around $\alpha = 13^\circ$, see figure 5.16. However, the graphs are unable to show this correctly, as with the employed grid the viscous effects cannot be predicted very well. Small oscillations are visible. VIBLI predicts separation to start at $\alpha = 9^\circ$. That Rampant is predicting separation to occur later can also be seen in the obtained C_l results.

C_l	$\alpha = 1^\circ$	$\alpha = 5^\circ$	$\alpha = 10^\circ$	$\alpha = 11^\circ$
VIBLI	0.110	0.542	0.967	0.953
Rampant	0.103	0.510	0.925	0.977

In the table above the C_l results of VIBLI are compared with the results obtained by Rampant. For $\alpha = 1^\circ$ and $\alpha = 5^\circ$ the results correlate fairly well. The VIBLI and Rampant results are still quite close at $\alpha = 10^\circ$ and $\alpha = 11^\circ$ but whereas VIBLI predicts maximum lift, Rampant keeps on predicting increasing lift. Rampant obtains maximum lift around $\alpha = 16^\circ$ or 17° , see figure 5.15.

That VIBLI is able to calculate a maximum lift coefficient is due to the fact that good closure relations have been used for H_1 and C_f . Furthermore it is essential to use a small global error. If the global error is not small enough cases at high angles of attack will predict a lift coefficient that is too high. Maximum lift will as a result also not be predicted. Instead the lift coefficient will continue to increase.

In figure 5.15 it can be seen that up to $\alpha = 9^\circ$ the C_l graph for the VIBLI results is almost linear. After $\alpha = 9^\circ$, where the flow starts to separate, the graph starts making a turn and reaches maximum lift at $\alpha = 10^\circ$, whereafter C_l decreases until the program crashes.

C_d	$\alpha = 1^\circ$	$\alpha = 5^\circ$	$\alpha = 10^\circ$	$\alpha = 11^\circ$
VIBLI	3.72×10^{-3}	4.36×10^{-3}	6.38×10^{-3}	6.69×10^{-3}
Rampant	1.65×10^{-2}	2.60×10^{-2}	6.03×10^{-2}	7.25×10^{-2}

In the above table the drag coefficients are shown. A big difference is seen between the drag coefficients obtained by Rampant and the ones obtained by VIBLI. This difference is a result of the difference in C_p near the trailing edge. The difference in C_f may also have a small influence.

As explained before the differences in result are mainly due to the fact that the VIBLI code has no wake and no transition region, and Rampant using not enough cells to predict the viscous effects accurately on an unstructured grid. However, despite all this it can be said that the results obtained by VIBLI, compared with the Rampant results, are in reasonably good agreement. Furthermore it has to be remarked that a test case run at VIBLI only takes a few minutes to converge for cases where there is no trailing edge separation.

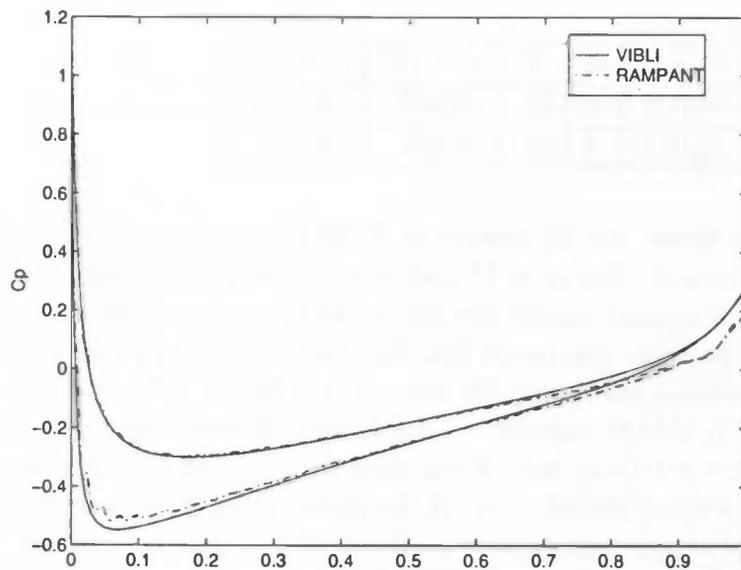


Figure 5.10: Pressure distribution at $\alpha = 1^\circ$

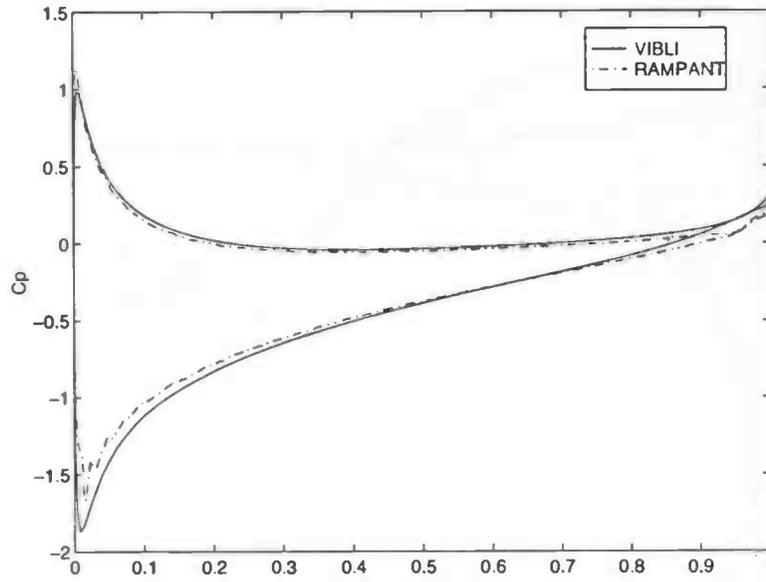


Figure 5.11: *Pressure distribution at $\alpha = 5^\circ$*

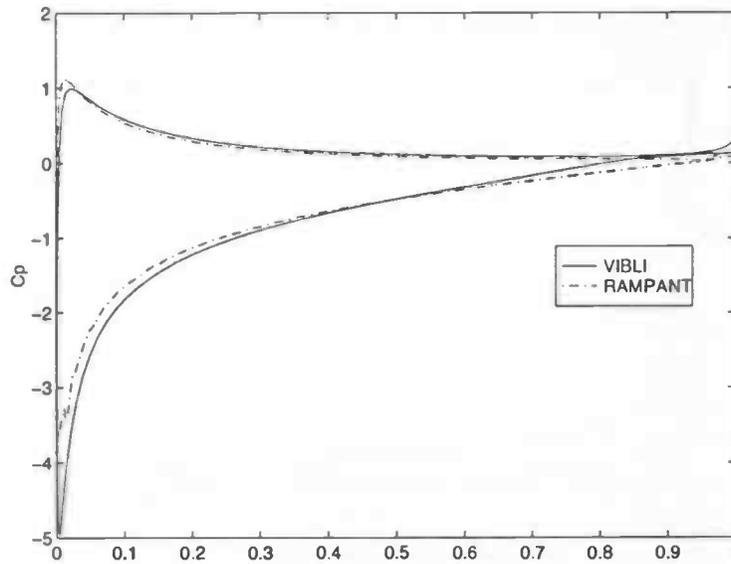


Figure 5.12: *Pressure distribution at $\alpha = 10^\circ$*

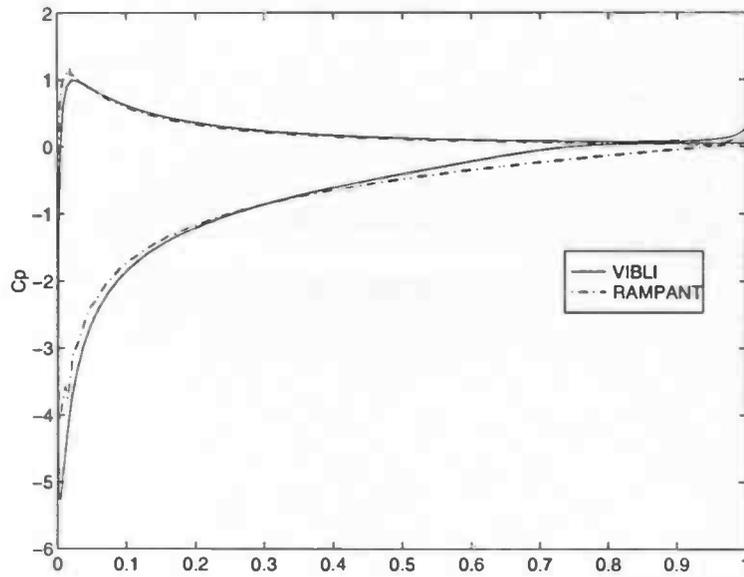


Figure 5.13: *Pressure distribution at $\alpha = 11^\circ$*

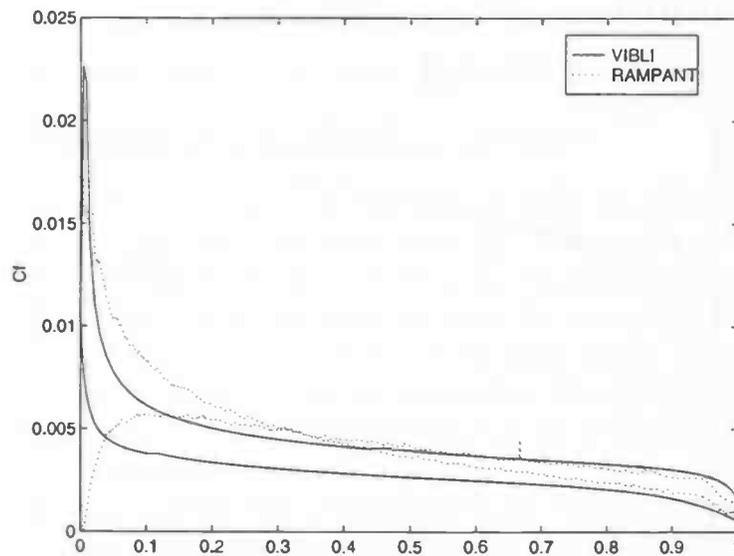


Figure 5.14: *Skin friction at $\alpha = 5^\circ$*

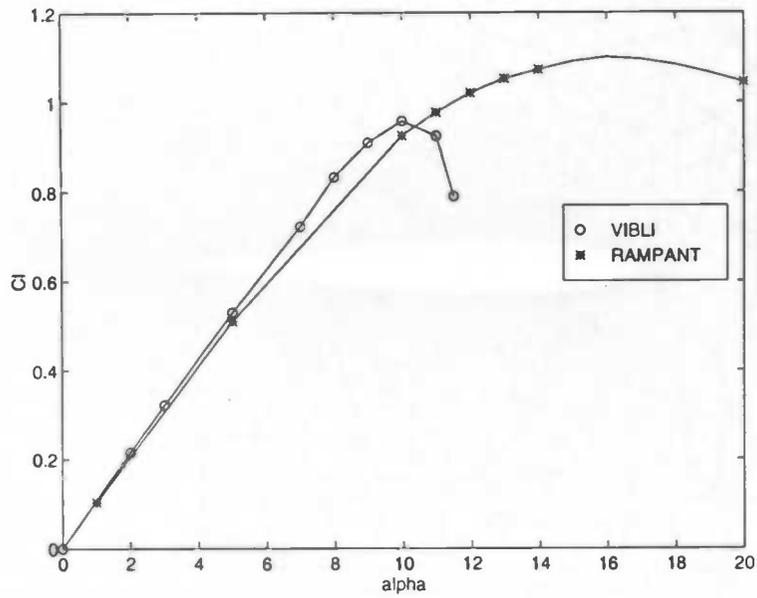


Figure 5.15: *Lift coefficient*

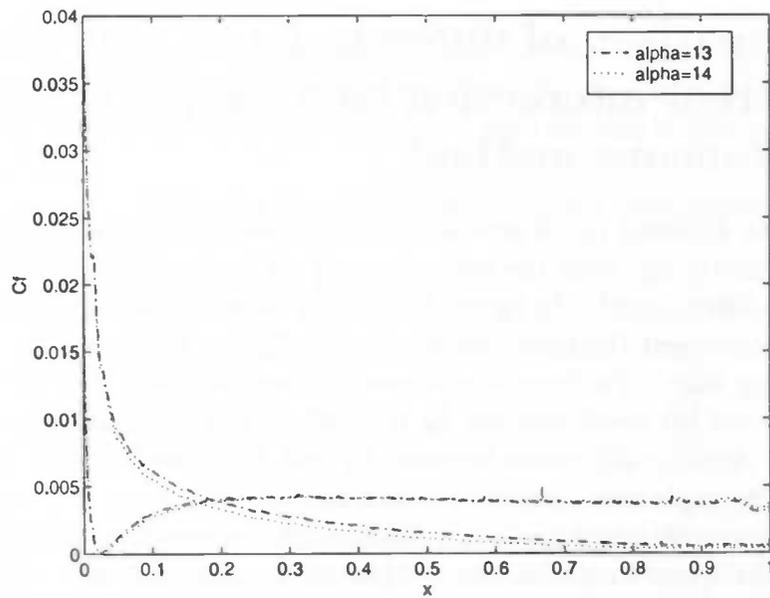


Figure 5.16: *C_f results produced by Rampant*

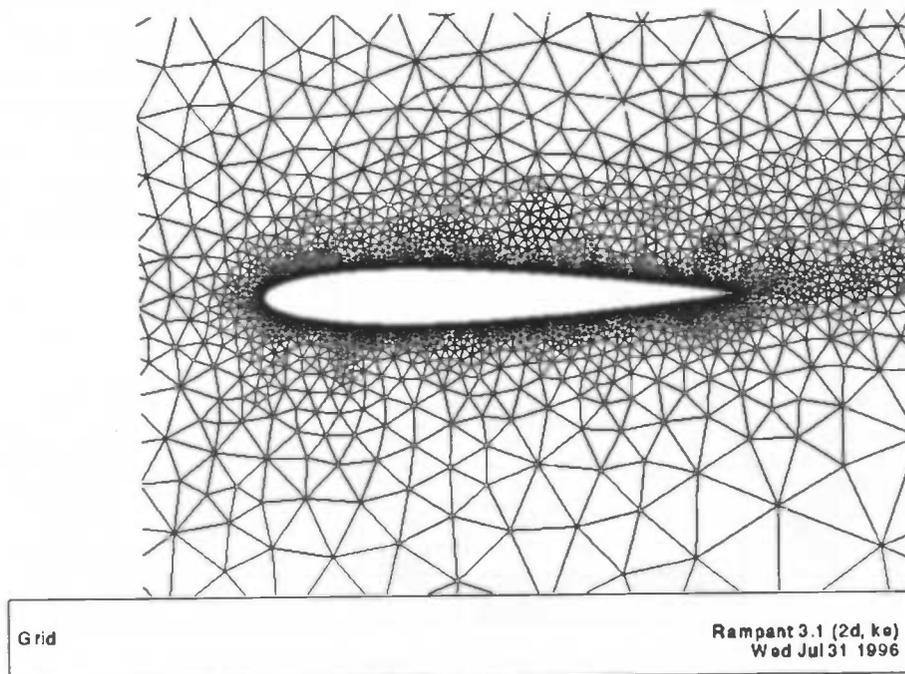


Figure 5.17: Grid used by Rampant at $\alpha = 13^\circ$

5.5 Comparison of different panel matrices and different interaction laws for the (quasi-) simultaneous method

In chapter 3 two different panel matrices E have been introduced, panel matrix E_0 and panel matrix E_1 , with the extra term $\frac{dq}{dx}$. The results obtained by these panel matrices differ slightly. In figure 5.18 it can be seen that at a high angle of attack the displacement thickness obtained with E_0 has grown a little bit more near the trailing edge. Furthermore it can be seen in the given tables below that the calculated lift coefficient for E_0 is smaller. The differences in result are however small. Another difference between E_0 and E_1 is the fact that E_1 is more robust, as will be explained later in this section.

The simultaneous method uses as interaction law the constructed panel matrix $I = E$. For the quasi-simultaneous method this is not the case. The quasi-simultaneous method uses a simpler I , which is only based on E . Because of this the simultaneous method is calculating $u_e = E[\delta^*]$ each time for every local iteration on a point i whereas the quasi-simultaneous method only has to calculate $E[\delta^*]$ for every i before all the local iterations start.

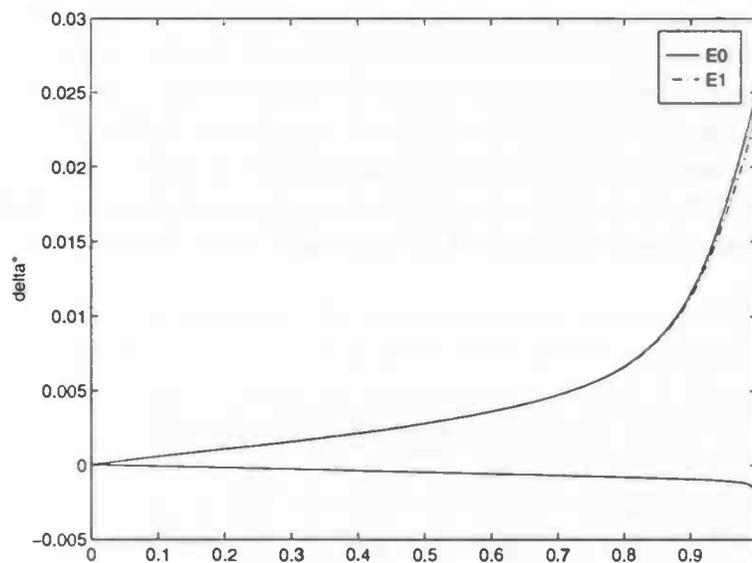


Figure 5.18: δ^* for different panel matrices at $Re = 1 \times 10^8$, $\alpha = 12^\circ$

For this reason the quasi-simultaneous method is slower but more robust than the simultaneous method. The quasi-simultaneous method can be used in combination with any outer flow, for instance it can be used with a compressible Euler solver.

For the quasi-simultaneous method three interaction laws have been constructed. I_0 is the interaction law derived from E_0 , I_1 is derived from E_1 . The third interaction law I_2 has been constructed with the use of thin-aerofoil theory, see appendix A.3.

In this report the following three combinations have been tested for the quasi-simultaneous method:

- $E_0 + I_0$
- $E_1 + I_1$
- $E_0 + I_2$

For the simultaneous method the two variations with E_0 and E_1 have been tested. The methods have run for a NACA 0012 profile at $Re = 1 \times 10^8$ and 120 points. The quasi-simultaneous method used a k -loop of $k = 20$. The variation $E_1 + I_1$ for the quasi-simultaneous method has problems with the initialisation at $\alpha = 0^\circ$. It was necessary to re-initialise the shape factor H during the first couple of iterations. If this is not done the system will diverge locally and crash. Underrelaxation will not solve this problem.

In the code this re-initialisation has been programmed and has been used for every variation of the simultaneous and quasi-simultaneous method. The other variations take less time to converge if this re-initialisation of H is not used.

Some results for different angles of attack are shown in the following tables. A more complete set of the obtained results is given in appendix C. The tables display the lift coefficient, the reduction factor, the total number of global iterations and the usertime in seconds. The test cases have run on a SUN sparc 10 station.

$\alpha = 0^\circ$	C_l	red.fact.	no.iter.	time(sec)
qsim($E_0 + I_0$)	0.000	0.738	31	170.2
qsim($E_1 + I_1$)	0.000	0.882	58	254.6
qsim($E_0 + I_2$)	0.000	0.828	39	201.7
sim(E_0)	0.000	0.836	71	31.7
sim(E_1)	0.000	0.903	116	45.6

$\alpha = 5^\circ$	C_l	red.fact.	no.iter.	time(sec)
qsim($E_0 + I_0$)	0.563	0.844	35	108.5
qsim($E_1 + I_1$)	0.569	0.914	55	171.6
qsim($E_0 + I_2$)	0.563	0.864	34	114.3
sim(E_0)	0.563	0.851	55	24.7
sim(E_1)	0.569	0.930	86	35.2

$\alpha = 11^\circ$	C_l	red.fact.	no.iter.	time(sec)
qsim($E_0 + I_0$)	1.171	0.977	197	593.6
qsim($E_1 + I_1$)	1.194	0.985	271	842.3
qsim($E_0 + I_2$)	1.171	0.972	173	576.7
sim(E_0)	1.170	0.948	166	59.7
sim(E_1)	1.193	0.975	268	92.6

$\alpha = 13^\circ$	C_l	red.fact.	no.iter.	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)	1.322	0.996	836	2695.7
qsim($E_0 + I_2$)	1.279	0.993	533	1862.5
sim(E_0)	1.278	0.986	553	185.9
sim(E_1)	1.320	0.989	615	209.3

$\alpha = 14^\circ$	C_l	red.fact.	no.iter.	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)	1.300	0.998	1801	5641.5
qsim($E_0 + I_2$)	1.234	0.995	865	2902.2
sim(E_0)	1.232	0.993	1038	333.7
sim(E_1)	1.295	0.995	1333	438.9

From the tables it is clear that the interaction law does not influence the end result. Furthermore it can be seen that if the same panel matrix is used the same lift coefficients are obtained. Only at high angles of attack is there sometimes a slight difference in C_l although the same panel matrix is used. This difference is a result of numerical errors.

At an angle of attack of zero all methods require almost twice as much time to converge than at an angle of 1° . For the cases with angles of attack between 1° and 7° , less time is required to converge than for $\alpha = 0^\circ$. The initial values of Blasius, not resembling the end result, used at an angle of 0° , is the reason for this slow convergence, together with the fact that H has to be re-initialised during the first 5 iterations. For angles greater than zero a prior result has been used for the initialisation, as the program has difficulties or does not start to work at all with the initial values of Blasius. The prior solution resembles the end result much better. The re-initialisation of H is then not necessary any more. The calculations go therefore much faster.

If the three quasi-simultaneous methods are compared it is obvious that, up to an angle of 11° , the quasi-simultaneous method using $E_0 + I_0$ is the fastest method. However, the variation using $E_0 + I_2$ takes, except for $\alpha = 0^\circ$, fewer iterations and has a lower reduction factor for $\alpha \geq 8^\circ$.

After $\alpha = 11.5^\circ$ the quasi-simultaneous method using $E_0 + I_0$ does not converge any more. This is exactly at the angle at which the streamline should start to separate from the surface, according to other VIBLI results. The system with $E_0 + I_0$ therefore is not very robust. This is in all probability a result of having the position of the boundary layer stations in the midpoints of the panels, which caused the irregular structure of the coefficients in E_0 .

The systems $E_1 + I_1$ and $E_0 + I_2$ do not have that problem, they continue working fine till even after $\alpha_{Cl_{max}}$. A difference is that for $E_0 + I_2$ an extra calculation is required as to calculate $\alpha = 14^\circ$. At $\alpha = 13^\circ$ the position of the stagnation point predicted by $E_0 + I_2$ is different from the position predicted by $E_1 + I_1$. This is a result of the difference in the panel matrix. Because of this the undisturbed outer flow of $E_0 + I_2$ predicts at $\alpha = 14^\circ$ the stagnation point to be two points away from the initial stagnation point of the prior calculation at $\alpha = 13^\circ$ instead of one point. An extra calculation is necessary so as to come one point nearer.

If the systems $E_1 + I_1$ and $E_0 + I_2$ are compared in time, it is visible that

$E_0 + I_2$ is faster and uses less iterations. From these results it is therefore clear that $E_0 + I_2$ is the best option. That $E_0 + I_2$ works so well is a result of the fact that for the construction of the interaction law I_2 the boundary layer stations have been taken in the endpoints of the panels. The thin-aerofoil matrix T from which I_2 is derived is because of this reason positive definite (appendix A.3). To make method $E_0 + I_2$ converge regularly, underrelaxation is required. This is necessary already for small angles of attack. For $E_1 + I_1$ this is only necessary at high angles of attack and can therefore be seen as the most robust method of the two.

If the two simultaneous methods are compared it is clear that the simultaneous method using E_0 is the best option. This method is faster, requires fewer iterations and has a smaller reduction factor than the simultaneous method using I_1 . Again, before $\alpha = 14^\circ$ could be calculated an extra calculation was required.

For all the tested methods it often happens that at high angles of attack in the beginning of the iteration process there is some trouble with convergence. For a few iterations the reduction factor is higher than one and so the solution diverges. However after a few iterations it recovers and a regular pattern of convergence is seen. Of course, a correct relaxation parameter is required. If during the iterations the position of the stagnation point changes, which happens only at high angles of attack, the program has again some troubles with converging. Again the reduction factor becomes greater than one, but after a few steps it is back to a value below one and has a regular convergence pattern afterwards.

5.6 Comparison of different coupling methods

In the previous section it has been concluded that the simultaneous method with panel matrix E_0 and the quasi-simultaneous method with panel matrix E_0 and interaction law I_0 are the fastest methods for small angles of attack. In this section these two methods will be compared with the direct and the semi-inverse methods. The quasi-simultaneous method uses a k -loop with $k = 5$, which can even be taken smaller as to reduce the time used per iteration a little bit.

For the test case again a turbulent flow past a NACA 0012 aerofoil is taken at $Re = 2267000$, $\alpha = 2^\circ$ and 120 stations. The subject of the workings of the direct method requires little discussion. It does not work at all. Even at $\alpha = 0^\circ$ the code crashes at the first global iteration, after diverging on some local iterations in which separation is predicted. The separation is predicted as a result of lack of interaction.

The semi-inverse method, with panel matrix E_0 , converges very slowly. This can clearly be seen in figure 5.19. The convergence graph of the semi-inverse method is almost horizontal. In the table below it can furthermore be seen that 1182 iterations were needed for the semi-inverse method to converge, with one iteration taking approximately 0.314 seconds.

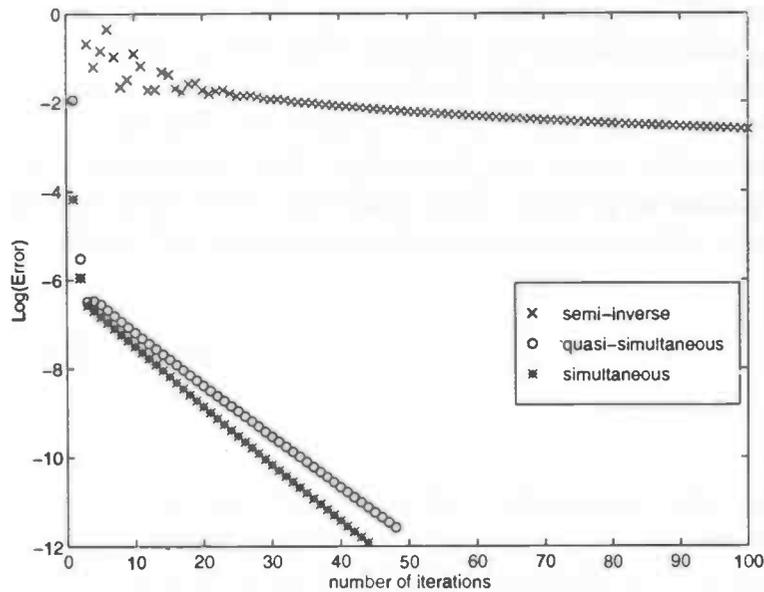


Figure 5.19: *Convergence of semi-inverse and (quasi-) simultaneous methods*

If the semi-inverse method is compared with the simultaneous and quasi-simultaneous methods, it is clear that although the semi-inverse method may use less time per iteration, it requires a greater number of iterations. It is therefore a much slower method.

$\alpha = 2^\circ$	no.iter.	total time(sec)	time per iter.	glob.error
sem-inv(E_0)	1182	371.15s	0.314s	1×10^{-2}
sim(E_0)	80	32.09s	0.401s	1×10^{-7}
qsim($E_0 + I_0$)	48	61.58s	1.283 s	1×10^{-5}

The semi-inverse method is also less robust. It is very sensitive to the choice of the relaxation parameter λ . If λ is slightly too large or too small the code may start to diverge.

For an angle of attack of zero the maximum Reynolds number that the semi-inverse code is able to handle is around 5×10^6 , whereas the (quasi-) simultaneous method simply calculates $Re = 1 \times 10^8$ without any problem. It can also be expected that for the test case with $Re = 2267000$ the semi-inverse method is unable to calculate up to an angle of 14.8° as can the (quasi-) simultaneous methods. However, this has not been tested, as it would take a lot of time

An explanation for the behaviour of the four discussed coupling methods, can be found in triple-deck theory, see appendix B. Both the direct and the semi-inverse method make use of a hierarchy between the boundary layer and the outer

flow. In regions with irregular flow or geometry, like regions with flow reversal or a sharp trailing edge, the hierarchy changes. This change of hierarchy may cause problems if it does not agree with the hierarchy of the numerical method used, as is the case for the direct and semi-inverse methods. The (quasi-) simultaneous methods do not make use of any hierarchy. The calculations are done in as simultaneous manner as possible. They avoid the problems that may arise if the hierarchy changes and are therefore much more robust and faster.

Chapter 6

Conclusions

In this study a viscous-inviscid boundary layer interaction code was developed for the incompressible, steady, laminar and turbulent flow past an aerofoil.

The laminar code is only to be used for thin aerofoils at very small angles of attack. The turbulent code on the other hand can handle both thin and thick aerofoils at angles of attack that can be increased till even after the angle at which maximum lift is predicted. Furthermore, the turbulent code is able to calculate cases at high Reynolds numbers.

The obtained turbulent results have been compared with a commercial Navier-Stokes code Rampant and were in reasonable good agreement. The differences are mainly due to the fact that Rampant contains as well a transition as a wake region, which influences the results.

For the viscous-inviscid interaction the (quasi-) simultaneous method was used, and several variations have been tested. Two panel matrices have been constructed, of which one uses Taylor series for the source strength, whereas the other one uses constant values on each panel. The latter has proven to have the highest convergence rate, whereas the other one has shown to be the most robust of the two.

Three interaction laws have been constructed, of which two are based on the used panel matrix whereas the third one is based on thin-aerofoil theory. The quasi-simultaneous method using this third mentioned thin-aerofoil interaction law, in combination with the panel matrix constructed with constant values for the source strength, is both fast and robust. The thin-aerofoil matrix is positive definite, as the boundary layer stations have been taken in the endpoints of the intervals in which the surface has been divided. For the construction of the panel matrix on the other hand, the boundary layer stations have been taken in the midpoints of the panels, which resulted in an irregular structure of the matrix.

The variations of the (quasi-) simultaneous method have been compared with other coupling methods and they proved to be much faster and more robust.

The next step that can now be taken is to implement a compressible Euler code, for which purpose this quasi-simultaneous interaction code has been devel-

oped. Other interesting improvements would be the implementation of a wake region and the construction of a panel matrix, based on boundary layer stations in the endpoints of the panels.

Appendix A

Manipulations

A.1 The outer flow velocity

The expression for the tangential velocity at the edge of the boundary layer coming from the outer flow can be found by manipulating equations (3.3), (3.6), (3.7), (3.18)-(3.21),(3.24) and (3.25):

$$\begin{aligned}u_{e_i} &= u_i \cos \theta_i + v_i \sin \theta_i, \\&= U_\infty \cos(\alpha - \theta_i) + \sum_{j=1}^N (u_{s_{ij}} + u_{v_{ij}}) \cos \theta_i + \sum_{j=1}^N (v_{s_{ij}} + v_{v_{ij}}) \sin \theta_i, \\&= U_\infty \cos(\alpha - \theta_i) + \sum_{j=1}^N q_j (\bar{u}_{s_{ij}} \cos \theta_i + \bar{v}_{s_{ij}} \sin \theta_i) \\&\quad + \gamma \sum_{j=1}^N (\bar{u}_{v_{ij}} \cos \theta_i + \bar{v}_{v_{ij}} \sin \theta_i) - \frac{l_i}{2\pi} \frac{dq_i}{dx}, \\&= U_\infty \cos(\alpha - \theta_i) + \sum_{j=1}^N (q_{0j} + q_j^*) c1_{ij} + (\gamma_0 + \gamma^*) c2_i - \frac{l_i}{2\pi} \frac{dq_i^*}{dx},\end{aligned}$$

with

$$c1_{ij} = \bar{u}_{s_{ij}} \cos \theta_i + \bar{v}_{s_{ij}} \sin \theta_i, \quad (\text{A.1})$$

$$c2_i = \sum_{j=1}^N (\bar{u}_{v_{ij}} \cos \theta_i + \bar{v}_{v_{ij}} \sin \theta_i). \quad (\text{A.2})$$

Considering only the undisturbed part gives an expression for u_{e0} :

$$u_{e0_i} = U_\infty \cos(\alpha - \theta_i) + \sum_{j=1}^N q_{0j} c_{1ij} + \gamma_0 c_{2i}. \quad (\text{A.3})$$

Using these new definitions, equation (3.23) and the relations found on page 28, it can be derived that:

$$\begin{aligned} u_{e_i} &= u_{e0_i} + \sum_{j=1}^N q_j^* c_{1ij} + \gamma^* c_{2i} - \frac{l_i}{2\pi} \frac{dq_i^*}{dx}, \\ &= u_{e0_i} + \sum_{j=1}^N \left\{ \sum_{k=1}^N A^{-1}(j, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\} c_{1ij} \\ &\quad + \left\{ \sum_{k=1}^N A^{-1}(N+1, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\} c_{2i} - \frac{l_i}{2\pi} \frac{d}{dx} \left\{ \sum_{k=1}^N A^{-1}(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\}, \\ &= u_{e0_i} + \sum_{k=1}^N \left\{ \sum_{j=1}^N A^{-1}(j, k) c_{1ij} + A^{-1}(N+1, k) c_{2i} \right\} \frac{d}{dx} (u_e \delta^*) \Big|_k \\ &\quad - \frac{l_i}{2\pi} \frac{d}{dx} \left\{ \sum_{k=1}^N A^{-1}(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\}, \\ &= u_{e0_i} + \sum_{k=1}^N X(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k - \frac{l_i}{2\pi} \frac{d}{dx} \left\{ \sum_{k=1}^N A^{-1}(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\}, \\ &= u_{e0_i} + u_{e_i}^*, \end{aligned}$$

and

$$X(i, k) = \sum_{j=1}^N A^{-1}(j, k) c_{1ij} + A^{-1}(N+1, k) c_{2i}, \quad (\text{A.4})$$

$$u_{e_i}^* = \sum_{k=1}^N X(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k - \frac{l_i}{2\pi} \frac{d}{dx} \left\{ \sum_{k=1}^N A^{-1}(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k \right\}. \quad (\text{A.5})$$

The equation above gives the relation between u_e^* and δ^* , which is needed for the interaction between the boundary layer and the outer flow.

If only constant values have been used for q then the last term of equation (A.5) is dropped:

$$u_{e_i}^* = \sum_{k=1}^N X(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k. \quad (\text{A.6})$$

A.2 The panel matrix

The panel matrix E gives a direct relation between the tangential velocity at the edge of the boundary layer, coming from the outer flow, and the displacement thickness. For the case where Taylor series for q have been used, the panel matrix E_1 is produced after the discretisation of $\frac{dq}{dx}$, done downwind and $\frac{d}{dx}(u_e \delta^*)$, done upwind.

In (3.27) it had been found that:

$$u_{e_i}^* = \sum_{k=1}^N X(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k - \frac{l_i}{2\pi} \frac{dq_i}{dx}.$$

For a boundary layer station k on the upper surface both derivatives are discretized as follows:

$$\frac{d}{dx} (u_e \delta^*) \Big|_k = \frac{u_{e_k} \delta_k^* - u_{e_{k-1}} \delta_{k-1}^*}{h_{u_k}}, \quad (\text{A.7})$$

$$\frac{dq_i}{dx} = \frac{q_{i+1} - q_i}{h_{u_{i+1}}} = \sum_{k=1}^N (A^{-1}(i+1, k) - A^{-1}(i, k)) \frac{d}{dx} (u_e \delta^*) \Big|_k, \quad (\text{A.8})$$

with $h_{u_k} = 0.5(l_k + l_{k-1})$. For the lower surface this is:

$$\frac{d}{dx} (u_e \delta^*) \Big|_k = \frac{u_{e_k} \delta_k^* - u_{e_{k+1}} \delta_{k+1}^*}{h_{l_k}}, \quad (\text{A.9})$$

$$\frac{dq_i}{dx} = \frac{q_{i-1} - q_i}{h_{l_{i-1}}} = \sum_{k=1}^N (A^{-1}(i-1, k) - A^{-1}(i, k)) \frac{d}{dx} (u_e \delta^*) \Big|_k, \quad (\text{A.10})$$

where $h_{l_k} = 0.5(l_{k+1} + l_k)$.

In the points on either side of the stagnation point the gradient of $u_e \delta^*$ is taken over half the distance so as to use the value of the stagnation point, $u_{e, \text{stagnation}} = 0$.

The gradients of q in the two trailing edge points are not necessary. The coupling of the outer flow to the boundary layer will not take place in these points, as to avoid problems that may arise from the fact that there is no wake region present. The problem of defining q in the wake is therefore solved. With these

assumptions matrix E_1 can be constructed. Taking a boundary layer station i on the upper surface, matrix \tilde{X} can be constructed:

$$\tilde{X}(i, k) = X(i, k) - \frac{l_i}{2\pi} \frac{2(A^{-1}(i+1, k) - A^{-1}(i, k))}{l_{i+1} + l_i}, \quad \text{for } k \in \text{uppersurface}$$

$$\tilde{X}(i, k) = X(i, k) - \frac{l_i}{2\pi} \frac{2(A^{-1}(i-1, k) - A^{-1}(i, k))}{l_i + l_{i-1}}, \quad \text{for } k \in \text{lowsurface}$$

which gives:

$$u_{e_i}^* = \sum_{k=1}^N \tilde{X}(i, k) \frac{d}{dx} (u_e \delta^*) \Big|_k. \quad (\text{A.11})$$

We define the points $k = k_{\text{lower}}$, the point next to the stagnation point on the lower surface, and $k = k_{\text{upper}}$, the point next to the stagnation point on the upper surface. Matrix E_1 is given by:

$$E_1(i, k) = \frac{2}{l_k + l_{k+1}} \tilde{X}(i, k), \quad \text{for } k = 1$$

$$E_1(i, k) = \frac{2}{l_k + l_{k+1}} \tilde{X}(i, k) - \frac{2}{l_{k-1} + l_k} \tilde{X}(i, k-1), \quad \text{for } k = 2, \dots, k_{\text{lower}} - 1$$

$$E_1(i, k) = \frac{2}{l_k} \tilde{X}(i, k) - \frac{2}{l_{k-1} + l_k} \tilde{X}(i, k-1), \quad \text{for } k = k_{\text{lower}}$$

$$E_1(i, k) = \frac{2}{l_k} \tilde{X}(i, k) - \frac{2}{l_{k+1} + l_k} \tilde{X}(i, k+1), \quad \text{for } k = k_{\text{upper}}$$

$$E_1(i, k) = \frac{2}{l_k + l_{k-1}} \tilde{X}(i, k) - \frac{2}{l_{k+1} + l_k} \tilde{X}(i, k+1), \quad \text{for } k = k_{\text{upper}} + 1, \dots, N$$

$$E_1(i, k) = \frac{2}{l_k + l_{k-1}} \tilde{X}(i, k). \quad \text{for } k = N$$

Similar manipulations can be performed for stations i on the lower surface. Thus, the expression for $u_{e_i}^*$ becomes:

$$u_{e_i}^* = \sum_{k=1}^N E_1(i, k) (u_e \delta^*) \Big|_k. \quad (\text{A.12})$$

For the case, where q has been taken a constant value on every panel, the panel matrix E_0 can be found by doing the similar discretisation for $\frac{d}{dx} (u_e \delta^*)$ only. The above formulation for E_1 is still correct for E_0 if instead of \tilde{X} , the unmodified X is used.

Consider now the diagonal, and upper and lower diagonal elements of matrix E_1 for a point i halfway along the upper surface. It is assumed that panels around this point i are, for reasons of convenience, of equal length h . Using the values calculated by the program, it can be found that:

$$\begin{aligned}
E_1(i, i) &\approx \frac{1}{h}(X(i, i) - X(i, i + 1)) \\
&+ \frac{1}{h} \left(\frac{h}{2\pi} \frac{A^{-1}(i + 1, i + 1) - A^{-1}(i, i + 1)}{h} - \frac{h}{2\pi} \frac{A^{-1}(i + 1, i) - A^{-1}(i, i)}{h} \right), \\
&\approx \frac{1}{h}(X(i, i) - X(i, i + 1)) + \frac{1}{h} \left(\frac{2}{2\pi} - \frac{-2}{2\pi} \right), \\
&\approx \frac{1}{h}(X(i, i) - X(i, i + 1)) + \frac{2}{\pi h}, \\
&\approx E_0(i, i) + \frac{2}{\pi h}.
\end{aligned}$$

In the same way it can be found that:

$$E_1(i, i - 1) \approx \frac{1}{h}(X(i, i - 1) - X(i, i)) - \frac{1}{\pi h}, \quad (\text{A.13})$$

$$\approx E_0(i, i - 1) - \frac{1}{\pi h}, \quad (\text{A.14})$$

$$E_1(i, i + 1) \approx \frac{1}{h}(X(i, i + 1) - X(i, i + 2)) - \frac{1}{\pi h} \quad (\text{A.15})$$

$$\approx E_0(i, i + 1) - \frac{1}{\pi h}. \quad (\text{A.16})$$

It is interesting to compare these matrix elements with the elements of matrix T constructed with thin-aerofoil theory. This is done in appendix A.3.

A.3 Thin-aerofoil theory

With thin-aerofoil theory the velocity field of the outer flow is found due to a source distribution along the chord line of the aerofoil. The approximation for the velocity u on the body surface shape is:

$$u(x, 0) = \int_0^c \frac{q(t)}{2\pi} \frac{dt}{x - t}, \quad (\text{A.17})$$

with $q(t)$ the source strength per unit length.

For the upper surface $\frac{1}{2}q(t)$ equals the vertical velocity at the surface $v(x, 0)$ [10]. In (2.16) it had been found that the boundary condition for the outer flow satisfies $v(x, 0) = \frac{\partial}{\partial x}(u_e \delta^*)$, which for reasons of convenience is simplified to $v(x, 0) = \frac{\partial}{\partial x}(\delta^*)$.

If the outer flow is divided into an undisturbed part and a disturbance part, $u_e = u_{e0} + u_e^*$, then the following model can be found:

$$u_e = u_{e_0} + \frac{1}{\pi} \int_0^c \frac{\frac{\partial \delta^*}{\partial \xi}}{x - \xi} d\xi. \quad (\text{A.18})$$

The chord is divided up into $N + 1$ parts $[x_j, x_{j+1}]$ for $j = 0$ to N . It is seen that here x_i is defined as a begin- or endpoint of an interval, instead of a midpoint. To evaluate the integral in the interval $[x_{i-1}, x_{i+1}]$ around x_i , a quadratic function for δ^* is used. The discretisation of the integral in the point $x = x_i$ is as follows:

$$\begin{aligned} u_e(x_i) &= u_{e_0}(x_i) + \frac{1}{\pi} \int_{x_0}^{x_{N+1}} \frac{\frac{\partial \delta^*}{\partial \xi}}{x_i - \xi} d\xi, \\ &= u_{e_0}(x_i) + \frac{1}{\pi} \sum_{j=0, j \neq i-1, i}^N \int_{x_j}^{x_{j+1}} \frac{\partial \delta^*}{\partial \xi} \Big|_{j+\frac{1}{2}} \frac{d\xi}{x_i - \xi} \\ &\quad + \frac{1}{\pi} \int_{x_{i-1}}^{x_{i+1}} \left\{ \frac{\partial \delta^*}{\partial \xi} \Big|_i + (\xi - x_i) \frac{\partial^2 \delta^*}{\partial \xi^2} \Big|_i \right\} \frac{d\xi}{x_i - \xi}, \\ &= u_{e_0}(x_i) + \frac{1}{\pi} \sum_{j=0, j \neq i-1, i}^N \frac{\partial \delta^*}{\partial \xi} \Big|_{j+\frac{1}{2}} \ln \frac{|x_i - x_j|}{|x_i - x_{j+1}|} \\ &\quad + \frac{\partial \delta^*}{\partial \xi} \Big|_i \ln \frac{|x_i - x_{i-1}|}{|x_i - x_{i+1}|} - \frac{\partial^2 \delta^*}{\partial \xi^2} \Big|_i (x_{i+1} - x_{i-1}). \end{aligned}$$

The derivatives of δ^* are discretized using a central discretisation scheme as to obtain:

$$\begin{aligned} u_e(x_i) &= u_{e_0}(x_i) + \frac{1}{\pi} \sum_{j=0, j \neq i-1, i}^N \left\{ \frac{1}{h} (\delta_{j+1}^* - \delta_j^*) \ln \frac{|i-j|}{|i-j-1|} \right\} \\ &\quad - \frac{2}{h} (\delta_{i+1}^* - 2\delta_i^* + \delta_{i-1}^*), \end{aligned}$$

for $i = 1, \dots, N$. The formula above can be rewritten as:

$$u_e(x_i) = u_{e_0}(x_i) + \sum_{j=0}^{N+1} T_{ij} \delta_j^*, \quad (\text{A.19})$$

with

$$\begin{aligned}
 T_{ij} &= \frac{1}{\pi h} \ln \left| 1 - \frac{1}{(i-j)^2} \right|, & j \neq 0, i-1, i, i+1, N \\
 T_{i,i} &= \frac{4}{\pi h}, & i = 1, \dots, N \\
 T_{i,i-1} &= -\frac{1}{\pi h} (2 - \ln 2), & i \neq 1 \\
 T_{i,i+1} &= -\frac{1}{\pi h} (2 - \ln 2), & i \neq N \\
 T_{1,0} &= -\frac{2}{\pi h}, \\
 T_{i,0} &= -\frac{1}{\pi h} \ln \frac{i}{i-1}, & i \neq 1 \\
 T_{i,N+1} &= -\frac{1}{\pi h} \ln \frac{N+1-i}{N-i}, & i \neq N \\
 T_{N,N+1} &= -\frac{2}{\pi h}.
 \end{aligned}$$

The constructed matrix T is positive definite.

The interaction law I_2 , describing only the local behaviour, that has been constructed for the quasi-simultaneous method with the use of this T -matrix is for a point i on the upper surface:

$$I_2(i, i) = \frac{4}{\pi h}, \quad (\text{A.20})$$

$$I_2(i, i-1) = \frac{-2}{\pi h}, \quad (\text{A.21})$$

$$I_2(i, i+1) = \frac{-2}{\pi h}. \quad (\text{A.22})$$

For a point i on the lower surface the sign changes.

It is interesting to observe what happens when the displacement thickness is defined halfway between the points where u_e is defined. Thus, let $d_{j+\frac{1}{2}}^*$ be defined halfway the j th and $(j+1)$ th grid points. To convert expression (A.19) we simply substitute:

$$\delta_j^* = \frac{1}{2} (d_{j-\frac{1}{2}}^* + d_{j+\frac{1}{2}}^*). \quad (\text{A.23})$$

This yields:

$$u_e(x_i) = u_{e0}(x_i) + \frac{1}{2} \sum_j T_{ij} (d_{j-\frac{1}{2}}^* + d_{j+\frac{1}{2}}^*), \quad (\text{A.24})$$

$$= u_{e0}(x_i) + \frac{1}{2} \sum_j (T_{i,j} + T_{i,j+1}) d_{j+\frac{1}{2}}^*. \quad (\text{A.25})$$

The coefficients $\hat{T}_{i,j+\frac{1}{2}} \equiv \frac{1}{2}(T_{i,j} + T_{i,j+1})$ no longer make a symmetric definite M -matrix. Instead $\hat{T}_{i,i-\frac{1}{2}}$ and $\hat{T}_{i,i+\frac{1}{2}}$ are both positive and equal to:

$$\hat{T}_{i,i-\frac{1}{2}} = \hat{T}_{i,i+\frac{1}{2}} = \frac{1}{\pi h} + \frac{1}{2\pi h} \ln 2. \quad (\text{A.26})$$

All other elements of this matrix are negative as before.

The matrix \hat{T} can be compared with panel matrix E in (3.29). In particular E_{ik} should resemble $\hat{T}_{i,k+\frac{1}{2}}$. Indeed, the structure of E_{ik} as described in section 3.1 is identical to that of \hat{T} .

Going into more detail, in the calculations with 120 points the coefficients of E_{ik} in a station near the middle of the profile are found to be:

$$E_{i,i-2} = -7.1, \quad E_{i,i-1} = 13.4, \quad E_{i,i} = 13.3, \quad E_{i,i+1} = -7.1.$$

The local gridsize being $h = 0.026$ leads to the following values for the entries of \hat{T} :

$$\hat{T}_{i,i-\frac{3}{2}} = -9.8, \quad \hat{T}_{i,i-\frac{1}{2}} = 16.5, \quad \hat{T}_{i,i+\frac{1}{2}} = 16.5, \quad \hat{T}_{i,i+\frac{3}{2}} = -9.8.$$

These values do resemble their corresponding E entries.

Appendix B

Triple-deck theory

A triple deck layer is a three-layered region around a singularity point x_s , with in the x -direction an extent of $O(Re^{-3/8}L)$ for a laminar incompressible flow [6]. In the y -direction the area is divided into three parts, see also figure B.1:

1. In the sublayer with thickness $O(Re^{-5/8}L)$ the flow is viscous and is governed by the boundary layer equations. In this layer the boundary layer reacts to the singularity.
2. The middle layer with thickness $O(Re^{-1/2}L)$ is an inviscid continuation of the oncoming boundary layer.
3. In the top layer with thickness $O(Re^{-3/8}L)$ the flow is inviscid and can be described with a potential flow.

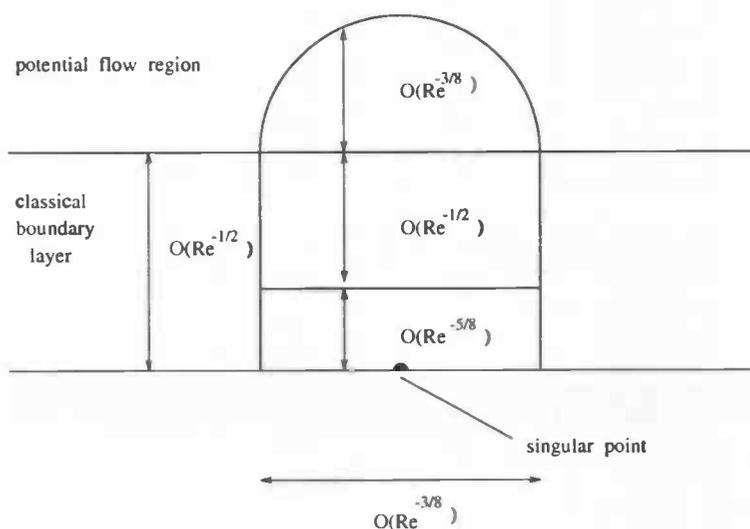


Figure B.1: *Triple-deck*

As the triple-deck has only a small extent $O(Re^{-\frac{3}{8}}L)$, the interaction with the outer flow has a local character. A local linearized description of the outer flow will therefore be sufficient to give an approximation of the interaction.

As $|x - x_s| = O(Re^{-\frac{3}{8}}L)$ there is no definite hierarchy between the viscous region and the inviscid region, as can be found by asymptotic expansions. Such a region is defined as a region with strong interaction [7].

For $|x - x_s| > O(Re^{-\frac{3}{8}}L)$ the hierarchy of the flow tends to the classical direct boundary-layer hierarchy, whereas as $|x - x_s| < O(Re^{-\frac{3}{8}}L)$ the hierarchy changes into the inverse type. The change of hierarchy of the triple-deck gives an explanation for the numerical difficulties of the direct and inverse methods. Problems arise when the hierarchy of the numerical method differs from the asymptotic hierarchy. In the (quasi-)simultaneous method there is no hierarchy and it avoids therefore these problems.

Although the description given here is only valid for incompressible laminar flow, the same type of conclusions concerning locality and hierarchy, can be drawn from the appropriate asymptotic theory for turbulent flow.

Appendix C

Results (quasi-)simultaneous method

In this section some results are given of the simultaneous and quasi-simultaneous methods used with different panel matrices and interaction laws. The tables display the lift coefficient, the reduction factor, the total number of iterations and the user time in seconds. The tested case is a NACA 0012 aerofoil in turbulent flow, for $Re = 1 \times 10^8$, 120 points and at different angles of attack. The quasi-simultaneous method used a k -loop of $k = 20$. If a smaller k would have been used less time would have been required for the solution to converge.

$\alpha=0$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.000	0.738	31	170.2
qsim($E_1 + I_1$)	0.000	0.882	58	254.6
qsim($E_0 + I_2$)	0.000	0.828	39	201.7
sim(E_0)	0.000	0.836	71	31.7
sim(E_1)	0.000	0.903	116	45.6

$\alpha=1$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.114	0.776	25	78.8
qsim($E_1 + I_1$)	0.115	0.878	40	124.3
qsim($E_0 + I_2$)	0.114	0.815	25	86.7
sim(E_0)	0.114	0.798	39	19.5
sim(E_1)	0.115	0.910	81	31.7

$\alpha=2$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.227	0.792	27	84.9
qsim($E_1 + I_1$)	0.229	0.884	41	130.3
qsim($E_0 + I_2$)	0.227	0.825	26	91.1
sim(E_0)	0.227	0.796	43	20.7
sim(E_1)	0.229	0.817	70	29.9

$\alpha=3$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.340	0.809	29	89.9
qsim($E_1 + I_1$)	0.343	0.897	47	146.7
qsim($E_0 + I_2$)	0.340	0.842	29	98.6
sim(E_0)	0.340	0.822	45	21.5
sim(E_1)	0.343	0.916	84	33.1

$\alpha=4$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.452	0.826	32	98.3
qsim($E_1 + I_1$)	0.457	0.905	49	153.6
qsim($E_0 + I_2$)	0.452	0.853	30	102.7
sim(E_0)	0.452	0.835	50	23.1
sim(E_1)	0.457	0.493	55	26.5

$\alpha=5$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.563	0.844	35	108.5
qsim($E_1 + I_1$)	0.569	0.914	55	171.6
qsim($E_0 + I_2$)	0.563	0.864	34	114.3
sim(E_0)	0.563	0.851	55	24.7
sim(E_1)	0.569	0.930	86	35.2

$\alpha=6$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.673	0.861	40	122.9
qsim($E_1 + I_1$)	0.681	0.922	61	190.1
qsim($E_0 + I_2$)	0.673	0.875	39	132.4
sim(E_0)	0.673	0.866	62	27.0
sim(E_1)	0.681	0.931	79	33.7

$\alpha=7$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.781	0.880	45	137.2
qsim($E_1 + I_1$)	0.791	0.932	67	210.4
qsim($E_0 + I_2$)	0.781	0.887	44	146.9
sim(E_0)	0.781	0.882	71	29.8
sim(E_1)	0.791	0.946	84	36.7

$\alpha=8$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.887	0.898	54	162.4
qsim($E_1 + I_1$)	0.898	0.941	81	254.7
qsim($E_0 + I_2$)	0.887	0.894	52	174.1
sim(E_0)	0.887	0.897	83	33.3
sim(E_1)	0.898	0.953	121	47.9

$\alpha=9$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	0.989	0.957	112	327.0
qsim($E_1 + I_1$)	1.003	0.975	164	505.0
qsim($E_0 + I_2$)	0.989	0.956	106	351.0
sim(E_0)	0.988	0.913	99	38.7
sim(E_1)	1.002	0.960	155	57.8

$\alpha=10$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	1.085	0.967	143	419.7
qsim($E_1 + I_1$)	1.102	0.980	205	630.8
qsim($E_0 + I_2$)	1.085	0.964	131	435.2
sim(E_0)	1.084	0.930	227	46.7
sim(E_1)	1.102	0.967	199	72.0

$\alpha=11$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)	1.171	0.977	197	593.6
qsim($E_1 + I_1$)	1.194	0.985	271	842.3
qsim($E_0 + I_2$)	1.171	0.972	173	576.7
sim(E_0)	1.170	0.948	166	59.7
sim(E_1)	1.193	0.975	268	92.6

$\alpha=12$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)	1.272	0.993	536	1697.4
qsim($E_0 + I_2$)	1.240	0.980	244	820.5
sim(E_0)	1.240	0.975	337	113.8
sim(E_1)	1.270	0.983	395	134.2

$\alpha=13$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)	1.322	0.996	836	2695.7
qsim($E_0 + I_2$)	1.279	0.993	533	1862.5
sim(E_0)	1.278	0.986	553	185.9
sim(E_1)	1.320	0.989	615	209.3

$\alpha=13.4$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)				
qsim($E_0 + I_2$)	1.275	0.994	605	2095.1
sim(E_0)	1.274	0.989	667	218.2
sim(E_1)				

$\alpha=14$	Cl	red.fact.	# iterations	time(sec)
qsim($E_0 + I_0$)				
qsim($E_1 + I_1$)	1.300	0.998	1801	5641.5
qsim($E_0 + I_2$)	1.234	0.995	865	2902.2
sim(E_0)	1.232	0.993	1038	333.7
sim(E_1)	1.295	0.995	1333	438.9

Appendix D

User's guide for VIBLI

To have the program VIBLI calculate the solution of an air flow past an aerofoil the following points have to be taken care of.

1. The file named 'input' has to be modified for the case that is to be calculated. The input has to contain:
 - (a) Reynolds number
 - (b) total number of points on the aerofoil surface. This number should be an even number.
 - (c) NACA-number. See for the explanation of this number chapter 1.
 - (d) angle of attack $\alpha \geq 0$
 - (e) the coupling method to be used:
 - i. simultaneous(0)
 - ii. quasi-simultaneous(1)
 - iii. semi-inverse(2)
 - iv. direct(3)
 - (f) global error epsilon, $\varepsilon \ll 1$
 - (g) relaxation parameter: $\omega \geq 1$ for the (quasi-)simultaneous method or $\lambda \ll 1$ for the semi-inverse method
 - (h) nature of the flow: laminar(0) or turbulent(1)
 - (i) variation of the coupling:
 - i. panel matrix E_0 (+ interaction law I_0) (0)
 - ii. panel matrix E_1 (+ interaction law I_1) (1)
 - iii. panel matrix E_0 + interaction law I_2 (2)
 - (j) initialisation: Blasius(0), which is used at $\alpha = 0^\circ$ or an inputfile(1). This inputfile should contain the results of a prior calculation at an

angle of attack one to two degrees smaller than the case that is to be calculated. The file should be termed 'RES.ref' and should contain the same number of points as is indicated at (b).

2. The program VIBLI has to be compiled with the command `f77 -r8 *.f` (or the command `make`, if the makefile is used). The flag `-r8` indicates that the program is compiled for double precision.
3. The program is started with the command `a.out` (or `aout`, if the make command is used). With the extra command `time` before the command `a.out` the user time will be given.
4. If an inputfile is used for the initialisation the program will ask to be given the 'initmove'. The stagnation point seen by the undisturbed outer flow is already predicted for the case to be calculated. The parameter `initmove` now is the difference in position of the stagnation point of the just given position of the stagnation point and the position of the stagnation point of the inputfile.
5. The output given after each global iteration is:
 - (a) number of total global iterations done
 - (b) maximum number of local iterations used
 - (c) logarithm of the global error
 - (d) the reduction factor
6. If the program has converged it gives the lift and drag coefficient, together with the final position of the stagnation point. The program has finished as it has said 'normal ending'.
7. The most important outputfile is named 'RESULT.res' and the 8 columns contain:
 - (a) the x -coordinate
 - (b) tangential velocity u_e
 - (c) displacement thickness δ^*
 - (d) shape factor H
 - (e) skin friction coefficient C_f
 - (f) the y -coordinate
 - (g) y -coordinate plus the displacement thickness
 - (h) pressure coefficient C_p

Other outputfiles are:

- (a) GEO.res, containing the geometry of the profile
 - (b) INVMAT.res, containing diagonals of the inverse of matrix A , see (3.23)
 - (c) COEFF1.res, containing the diagonals of matrix X , see (3.27) or (3.28)
 - (d) COEFF2.res, containing the diagonals of panel matrix E , see (3.29)
 - (e) INIT.res, containing the initialisation of δ^* and u_e . Furthermore, it contains the coordinates of the midpoints of the panels
 - (f) TEST.res, containing temporary results
 - (g) CLOSURE.res, containing temporary values of the closure relations
8. Example inputfile: $Re = 2267000$, $N = 120$, $NACA = 12$, $\alpha = 0^\circ$, coupling method = 0, $\varepsilon = 0.00001$, $\varepsilon_{semi-inverse} = ..$, $\omega = 1$, $\lambda = ..$, case turbulent= 1, variation= 1, initialisation= 0, no. iterations = 1000.

Bibliography

- [1] H. Schlichting, *Boundary layer theory*, 7th ed., Mc Graw-Hill (1979).
- [2] A.E.P. Veldman, *Dictaat Grenslaagstromingen*, (1992).
- [3] S. Goldstein, *On laminar boundary-layer flow near a position of separation*, Quart. J. Mech. Appl. Math. 1 (1948), 48-69.
- [4] D. Catherall and K.W. Mangler, *The integration of the two-dimensional laminar boundary-layer equations past the point of vanishing skin friction*, J. Fluid Mech. (1966), 163-182.
- [5] J.C. Le Balleur, *Couplage visqueux non visqueux: méthode numérique et applications aux écoulements bidimensionnels transsoniques et supersoniques*, La Recherche Aéronautique 183 (1978), 65-76.
- [6] K. Stewartson, *Multistructured boundary layers on flat plates and related bodies*, Adv. appl. Mech. 14 (1974), 145-239.
- [7] A.E.P. Veldman, *Boundary layers with strong interaction: from asymptotic theory to calculation method*, Proc. BAIL 1 Conf. (J.J.H. Miller, Ed.), Dublin (1980), 149-163.
- [8] A.E.P. Veldman, *New, Quasi-simultaneous Method to Calculate Interacting Boundary Layers*, AIAA J. vol. 19 (1981), 79-85.
- [9] I.H. Abbott and A.E. von Doenhoff, *Theory of wing sections*, Dover, New York (1959).
- [10] J. Moran, *An introduction to theoretical and computational aerodynamics*, John Wiley, New York (1984).
- [11] J. Cousteix, *Couche Limite laminaire*, CEPADUES, Toulouse (1988).
- [12] E.G.M. Coenen, study *Introduction à la couche limite interactive*, (1995).
- [13] R. Houwink and A.E.P. Veldman, *Steady and unsteady separated flow computations for transonic airfoils*, AIAA paper 84-1618, 17th Fluid Dyn., Plasma Dyn. and Lasers Conf., Colorado (1984).