COMPUTATIONAL MECHANICS
New Trends and Applications
E. Oñate and S.R. Idelsohn (Eds.)

A PSEUDO-SPECTRAL APPROACH FOR THE INCOMPRESSIBLE BOUNDARY LAYER EQUATIONS WITH AUTOMATIC NORMAL SCALING

Mario A. Storti*

*Centro Internacional de Métodos Computacionales en Ingeniería
Güemes 3450
3000 Santa Fe, Argentina
e-mail: mstorti@minerva.unl.edu.ar
web page: http://venus.unl.edu.ar/gtm-eng.html

Key words: boundary layer equations, meshless methods, spectral methods, viscous flow, incompressible flow, laminar flow, parabolized Navier-Stokes equations,

Abstract. *A pseudo-spectral numerical method for the solution of the incompressible 3D boundary layer equations is presented. The method is based on Fourier expansion in the lateral transformed coordinate, similarly to the transformation that leads to the polynomial Tchebyschev expansion in finite intervals, but appropriated to semi-infinite intervals, so that no extra parameter is needed for the outer boundary of the layer. A scaling is applied to the normal coordinate but with the innovation that it is based on the computed boundary layer thickness, i.e. not assuming a priori a variation for it. Spectral decay of the expansion coefficients is shown for the similar solution to the family of wedge flows. Also, spectral convergence of the error is shown for the case of a convergent channel (one of the similar "wedge flows"), for which an analytical solution is available. The method pretends to have a good performance also when using very few parameters, so that results with four terms in the Fourier series (it amounts to two independent parameters) are compared with the well known method from von Kàrmàn and Pohlhausen. Several 2D numerical examples show the precision of the method. For 3D problems, the boundary layer equations are solved in a completely general mesh and coordinate system on the surface using the tensorial form of the equations*[1]*. In order to advance the solution in the streamwise coordinate a mesh-less approximation is used in the coordinates on the surface. This feature allows the treatment of very general geometries. 3D numerical examples include the yawed cylinder and flat plate, and 3D axisymmetric flows like the cone, the sphere and a rotating sphere.*

1 INTRODUCTION

Boundary layer equations are a simplification of the Navier-Stokes equations where viscous terms parallel to the surface are neglected. Thus, the equations are *"parabolized"* so that they can be integrated as if the longitudinal direction were a time-like coordinate. This results, from the point of view of numerical solutions, in a great reduction in the computational resources required. Coupled potential flow / boundary layer solutions have been the first approximate solutions of practical interest to the Navier-Stokes equations for rather general geometries, but mainly in the context of exterior flows around geometries where separation does not happen or it is of secondary interest like, for instance rather streamlined bodies as airfoils, wing-fuselage configurations, ships, propellers. Today, this technique continues to be used even if direct solutions of the Navier-Stokes equations are available, due to the comparatively low computational effort required. This ranges from complete 3D geometries on small computers, but even for supercomputers when iteration is required, for instance optimization or free-surface problems. Also, accurate solutions of the boundary layer equations provide the base flow for linear an non-linear stability analysis[2].

The first approximate methods where of the *integral* type. The velocity profile in the layer were characterized by the layer thickness δ and a *"shape parameter"*, and their evolution were governed by some integral form of the continuity and momentum equations. It is not evident how to extend this method in order to allow a better representation of the profile including more parameters and deriving accordingly more integral equations. In short, it is not evident how to extend the method in order to be *convergent*. On the other hand, convergent methods have been developed via standard numerical methods as finite differences and, in few cases, finite elements[3,16,10], and also via series expansions as in the *Multimoment or Weighted-Residual* (MWR) method[4]. As mentioned before, the longitudinal coordinate is treated as time-like, so that a standard ODE solver (Runge-Kutta, multi-step, for instance) may be applied, and only the normal coordinate y remains to be discretized. The simplicity of the domain ($0 \leq y < \infty$) and the fact that the solution is expected to be highly regular for $y > 0$, suggest the use of a higher order difference method.

Spectral methods[5–6] are based on series expansions in a basis such that, under certain conditions, *infinite order accuracy* (also *termed spectral convergence*) is achieved, i.e. the error of the numerical solution with respect to the exact solution decays faster than any finite power of the degrees of freedom (and then of the computational effort). In this and many order aspects, spectral methods behave as finite difference infinite order methods. Fourier series are the standard choice for periodic problems, whereas Tchebyschev polynomials are used for bounded intervals[5]. In fact, both are closely related, since Tchebyschev polynomials can be viewed as a particular mapping from the bounded interval on the real axis in order to make the problem periodic while retaining the degree of regularity of the solution. One-dimensional stretchings may be used in

order to increase the resolution where required (shocks, boundaries). For semi-infinite regions $0 \leq y < \infty$ (as the normal coordinate in the boundary layer equations) it is common use to truncate to a bounded interval $0 \leq y < y_{\max}$ and use Tchebyschev polynomials.

Since the boundary layer thickness may grow or decrease by several orders according to the variation of the pressure gradient, it is imperative to add or remove points from the computational grid (in a finite difference context) if a constant step size will be adopted in the normal direction. Alternatively, it is adviceable to scale the normal coordinate by a length scale that should be roughly proportional to the layer thickness (either the displacement, momentum or 99% thicknesses). In this way, a uniform resolution is achieved with the same number of points at each longitudinal station. However, as the thickness is a result of the computation it is not evident how to couple such scaling with the time-marching algorithm.

Pruett and Streett[7] and Pruett[2] applied the spectral method to the boundary layer equations mainly in the context of obtaining highly accurate solutions to be used as a base flow for a subsequent stability analysis. In one of the papers[7] the normal coordinate is scaled with a scale length $\propto \sqrt{x}$ which is appropriated for flat potential velocities, whereas in the other[2] the scale length is more general. It is based in the so called Levy-Lees transformation and includes compressibility effects and is appropriated for more general potential velocity profiles. For incompressible 2D flows it amounts to a scale length $\propto {U_{\text{ext}}}^{-1}(\int U_{\text{ext}}(x)\ dx)^{1/2}$, where x is the arc length and U_{ext} the potential velocity. For similar flows $U_{\text{ext}} \propto x^m$, it reduces to a scaling length $\propto x^{1/2(1-m)}$ which is effectively proportional to the thickness of the layer for such flows, but it is not guaranteed that it will cover more general situations. A more representative estimate could be taken from the approximation to the momentum thickness in the well known Thwaites method $\theta \propto U_{\text{ext}}^{-3}(\int U_{\text{ext}}^5\ dx)^{1/2}$. However, we will show that it is relatively easy to scale with the computed displacement thickness. This is one of the main goals of this work. On the other hand, the spectral approximation in the previous mentioned papers[7,2] is based on truncating the computational domain and using a Tchebyschev polynomial expansion, as described previously, whereas we propose here to map directly the semi-infinite domain to the whole real axis and using a Fourier series expansion, thus avoiding the truncation and the subsequent mapping to the bounded interval suitable for the Tchebyschev polynomial expansion. Off course, this mapping involves some care in order to obtain a periodic problem and preserve the regularity of the solution.

1 BOUNDARY LAYER GOVERNING EQUATIONS

The incompressible laminar 2D boundary layer equations are:

$$\begin{aligned} u u_{,x} + v u_{,y} &= \nu\, u_{,yy} + U_{\text{ext}} U_{\text{ext},x} \\ u_{,x} + v_{,y} &= 0 \end{aligned} \qquad \text{(1.a,b)}$$

where x is the streamwise coordinate, y the distance to the wall, u, v the corresponding components of the velocity vector and $(\ )_{,x}$ denotes partial differentiation with respect to x.

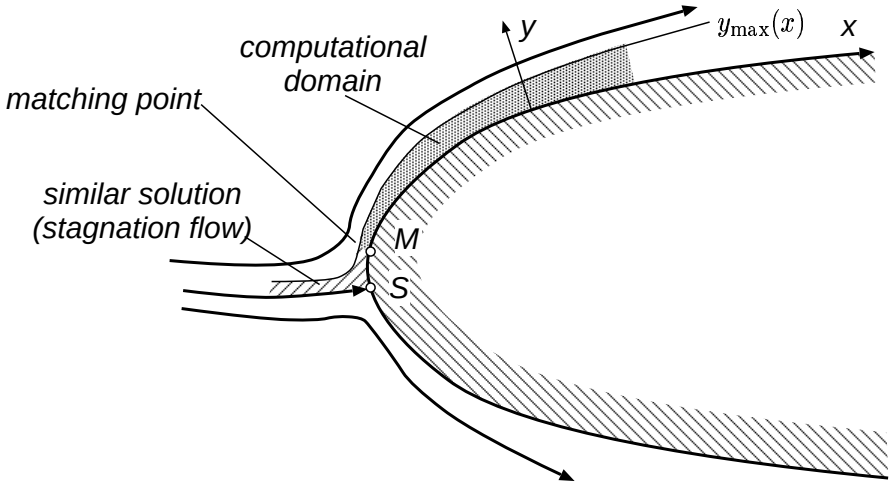

Figure 1: Coordinate system and computational domain.

When using finite differences (or finite elements) to solve this equations we have to restrict the computational domain to some region $y < y_{\max}(x)$. It is evident that in order to have a uniform approximation to the velocity field at all stages, $y_{\max}(x)$ should be chosen as a fixed multiple of a measure of the local thickness as, for instance, $y_{\max}(x) = M_\delta \delta^*(x)$, with M_δ large (say $M_\delta \sim 5$), and δ^* the local displacement thickness. As δ^* is not known a priori, one must guess a certain behavior for δ^*, say:

$$\delta^*(x) \sim \delta_{\text{scal}}(x) \tag{2}$$

We call δ_{scal} the normal scaling length and then use:

$$y_{\max}(x) = M_\delta \delta_{\text{scal}}(x) \tag{3}$$

For instance, the so called Levy-Lees transformation used by Pruett[2] implies a scaling of the normal coordinate of the form:

$$\delta_{\text{scal}} \propto {U_{\text{ext}}}^{-1} (\int U_{\text{ext}}(x)\ dx)^{1/2} \tag{4}$$

For the wedge flows ($U_{\text{ext}} \propto x^m$) it gives a scale length $\propto x^{1/2(1-m)}$, which is correct since it is the same dependence as the displacement thickness. However, it is not guaranteed that it will give a good estimate in more general situations. A more approximate

expression could be based on the momentum thickness estimate in the Thwaites method, since it can be computed from a simple quadrature:

$$\delta_{\text{scal}} = \theta_{\text{Thwaites}} \propto \left(\frac{1}{U_{\text{ext}}^6} \int U_{\text{ext}}^5 \, dx \right)^{1/2} \tag{5}$$

However, large discrepancy exists in the location of the separation point for the simple Thwaites method with respect to the spectrally convergent method proposed here, and since the layer thickness has strong variations near the separation point it is evident that this variations will be not properly followed by the Thwaites estimate (5). For the moment, we will derive discrete equations as if δ_{scal} were a completely independent quantity chosen *a priori*. Afterwards, we will explain how to set δ_{scal} automatically at the same time that the boundary layer equations are being solved in the time-like direction.

It is natural for numerical methods like finite differences to map the computational domain to a rectangle with the auxiliary curvilinear transformation $(x, y) \to (\xi, \eta)$:

$$\begin{aligned} \xi &= x \\ \eta &= \frac{y}{\delta_{\text{scal}}(x)} \end{aligned} \tag{6}$$

so that the computational domain is now the semi-infinite strip $\xi > 0$, $0 < \eta < M_\delta$. This transformation is somewhat related to the well known similarity transformation that leads to similar solutions for wedge flows. The extensions to compressible and axisymmetriccompressible flows are called Illingworth and Levy-Lees transformations[4] respectively. However, the fact that these transformations lead to similar solutions is irrelevant for numerical computations and the only interesting property which is the proper scaling of the normal coordinate is retained by (6). Also, those transformations not only scale the normal coordinate, but also remaps the longitudinal coordinate in such a way that a uniform grid is appropriate. This is not necessary if an algorithm with variable step-size automatically adjusted is used. In addition, it is relatively easy to invert the transformation law (8) for the contravariant components of velocity induced by (6) in order to recover the normal velocity component v, whereas special procedures are needed for the Levy-Lees (and similar) transformations[2].

The transformed equations in the new coordinate system are:

$$\begin{aligned} UU_{,\xi} + VU_{,\eta} &= \frac{\nu}{{\delta_{\text{scal}}}^2} U_{,\eta} + U_{\text{ext}} U_{\text{ext},\xi} \\ \frac{1}{\delta_{\text{scal}}} \left(\delta_{\text{scal}} U \right)_{,\xi} + V_{,\eta} &= 0 \end{aligned} \tag{7.a,b}$$

where U, V are the contravariant components of the velocity vector:

$$\begin{aligned} U &= u \\ V &= \eta_{,x} \, u + \eta_{,y} v = -\frac{y \, \delta_{\text{scal},x}}{{\delta_{\text{scal}}}^2} \, u + \frac{1}{\delta_{\text{scal}}} \, v \end{aligned} \tag{8.a,b}$$

These equations are valid for an arbitrary choice of $\delta_{\text{scal}}(x)$. However, as mentioned, it seems that an optimal choice will be $\delta_{\text{scal}} \sim \delta^*$ or other thickness criterion as the momentum thickness, for instance. One of the innovative contributions of this paper is that the scaling thickness is set plainly to $\delta_{\text{scal}} = \delta^*$, so that the computational domain is coupled with the numerical solution. This will be explained in detail in section §2.5, and will impose a restriction on the coefficients and some care has to be taken in the solution of the resulting system of DAE's (*"Differential-Algebraic Equations"*[8]).

2 PSEUDO-SPECTRAL DISCRETIZATION

As usual, we will take benefit of the parabolic character of the boundary layer equations and solve in the ξ coordinate as a *"time-like"* coordinate, whereas η remains as the *"spatial coordinate"*. As $\xi \equiv x$ in transformation (6) we will use x or ξ indistinctly in the sequel. As the computational domain is the semi-infinite interval $0 \leq \eta < \infty$ and the solution has a high degree of regularity, it is appealing the use of high order or spectral methods. Our method will be based in a pseudo-spectral approximation for η whereas a standard finite difference method will be applied in the time-like coordinate.

2.1 Basics of spectral approximations

Spectral methods are based on the approximation by a set of non-local functions (usually coming from the "spectra" of a differential operator) such that the approximation error converges faster than any finite power of the number of terms involved[6,5,9]. The simplest example is the Fourier series for infinite differentiable periodic functions. Let:

$$\phi(\theta) \approx \hat{\phi}(\theta) = \sum_{k=-M}^{M} c_k \, e^{ik\theta} \tag{9}$$

It can be shown that, if ϕ is regular enough for all θ and periodic $\phi(\theta + 2\pi) = \phi(\theta)$, then series (9) converges faster than any power of M:

$$\|\phi - \hat{\phi}\| < CM^{-p}, \quad \text{for any} \quad p, M > M^*(p) \tag{10}$$

This is termed *"spectral convergence"*. In this as in many other aspects spectral methods can be assimilated as *"infinite order methods"*.

2.2. Finite interval Mapping

For finite intervals one can extend the solution to the real axis periodically and apply the previously described method. For instance, given ϕ defined on $0 < x < 1$ one can extend to $-\infty < \theta < \infty$ with the following mapping:

$$x = |g(\theta)| \tag{11}$$

with:

$$g(\theta) = 2\left[\left(\frac{\theta}{2\pi}\right) - \text{round}\left(\frac{\theta}{2\pi}\right)\right] \tag{12}$$

and round(x) is the nearest integer to x. This is equivalent to map linearly $0 \leq x \leq 1$ to $0 \leq \theta \leq \pi$ and then extend symmetrically with respect to all the points of the form $\theta = n\pi$ with n integer. Note that the mapping is linear in each interval of the form $n\pi \leq \theta \leq (n+1)\pi$. Assuming that ϕ has continuous derivatives of all orders as function of θ, then the extended function is continuous, but its first derivative is discontinuous at $2n\pi$ or $(2n+1)\pi$, unless $\phi_{,x} = 0$ at $x = 0$ or 1, respectively, and similarly for all the odd derivatives. Thus, the rate of convergence is affected by the mapping, unless all the odd derivatives are null at $x = 0, 1$. Consider now the mapping:

$$x = \tfrac{1}{2}(1 - \cos\theta) \tag{13}$$

It is simple to see that $\phi(\theta)$ preserves the same degree of continuity in the interior points as the original function. So that if ϕ has infinite continuous derivatives with respect to x, the same is valid with respect to θ, and a Fourier series is spectrally convergent. This is the basis of spectral approximations for finite intervals.

As ϕ is even, coefficients c_k satisfy $c_k = c_{-k}$ and series (9) can be put in the form:

$$\hat{\phi}(\theta) = \sum_{k=0}^{M} \epsilon_k c_k \cos k\theta = \sum_{k=0}^{M} a_k \cos k\theta \tag{14}$$

with $\epsilon_k = 1$ for $k = 1$ and 2 otherwise. Replacing θ in terms of x from (13) we arrive to the classical expansion in Tchebyschev polynomials. One point that is often used in practice is that given the values of ϕ on a equally spaced grid of points in the θ coordinate coefficients c_k can be computed very efficiently by means of a *Fast Fourier Transform*.

2.3 Semi-infinite interval

The domain of interest for the spectral approximation is the semiinfinite axis $0 < \eta < \infty$. A first attempt is to restrain the computational domain to $0 < \eta < M_\delta$, as is usual in finite differences, apply mapping (13) combined with a linear mapping from $0 \leq x \leq 1$ to $0 \leq \eta \leq M_\delta$, and assume a Tchebyschev polynomial expansion. However, this is not efficient in the sense that the resolution is higher near the outer edge.

Consider now the following mapping:

$$\tanh\left(\frac{\eta}{\eta_{\text{scal}}}\right) = [g(\theta)]^2 \tag{15}$$

It has the same regularity properties that mapping (13) at the wall ($\eta = 0$), and it has a uniformly decreasing resolution from the wall to the outer edge ($\eta = \infty$). The point

with such mappings that include infinity is the regularity of the transformed function at $\theta = \pi$ (the image of $\eta = \infty$). As discussed previously, the resulting function $\phi(\theta)$ should have all the odd derivatives null and the even derivatives even at $\theta = \pi$, in order to have all derivatives continuous there. The condition on the even derivatives is satisfied automatically. Consider the first derivative:

$$U_{,\theta} = \frac{d\eta}{d\theta} U_{,\eta} = 2\theta \, \cosh^2(\eta/\eta_{\text{scal}}) \, U_{,\eta} \tag{16}$$

This is null if $U_{,\eta}$ decays faster to zero than $\cosh^2(\eta/\eta_{\text{scal}}) \sim e^{2\eta/\eta_{\text{scal}}}$. Similar criteria can be developed for the higher derivatives, but in order to make a rigorous analysis we would require also a detailed analysis of how U behaves at $\eta = \infty$. The important point is that $(d\eta/d\theta)$ should not got to infinity faster than the rate at which U_η decreases. If necessary, we may choose a different η_{scal} or modify the left hand side in the mapping (15), for instance:

$$\begin{aligned} 1 - \frac{\eta_{\text{scal}}}{\eta} &= [g(\theta)]^2 \\ \text{or also:} \qquad \tanh\left[\left(\frac{\eta}{\eta_{\text{scal}}}\right)^2\right] &= [g(\theta)]^2 \end{aligned} \tag{17.a,b}$$

where g is defined by (17.a) goes much slowly to infinity than (15) whereas (17.b) goes faster. In addition, the degree of regularity dictaminates the ultimate rate of convergence, i.e. the rate of convergence *when the number of terms M in series (9) goes to infinity*. This means that sometimes the fact that a given mapping goes too fast to infinity and some degree of regularity is lost doesn't mean necessarily that the approximation is worse than that one for a more regular mapping *for a given finite number of terms M*. In conclusion, the choice of an appropriate mapping involves also some degree of experience. In practice, we used in all cases mapping (15) with $\eta_{\text{scal}} = 6$.

2.4 Weighted residual formulation

Let $\bar{U} = U/U_{\text{ext}}$, then it satisfies: $\bar{U}(\eta = 0) = 0$, $\bar{U}(\eta = \infty) = 1$. We assume the following expansion for $\bar{U}$:

$$\bar{U} \approx \sum_{k=0}^{M} a_k \cos k\theta \tag{18}$$

The restrictions mentioned above result in 2 linear restrictions on the a_k coefficients:

$$\begin{aligned} \bar{U}(\eta = 0) &= \sum_{k=0}^{M} a_k = 0 \\ \bar{U}(\eta = \infty) &= \sum_{k=0}^{M} (-1)^k \, a_k = 0 \end{aligned} \tag{19}$$

So that only $N_{\text{dof}} = M - 1$ of the $M + 1$ coefficients are independent. Expansion (18) can be put as:

$$\begin{aligned} \bar{U}(x,\eta) &= \boldsymbol{\phi}^T(\eta) \cdot \mathbf{a}(x) \\ \boldsymbol{\phi}^T &= [\, \phi_0(\eta) \quad \ldots \quad \phi_M(\eta) \,] \\ \phi_k(\eta) &= \cos(k\theta) \end{aligned} \tag{20}$$

where $\mathbf{a} = [\, a_0 \quad a_1 \quad \ldots \quad a_M \,]^T$ is a vector of length $M + 1$ with the coefficients in the series expansions, Replacing in the continuity equation (7.b) and solving for $V_{,\eta}$:

$$\begin{aligned} V_{,\eta} &= -\frac{1}{\delta_{\text{scal}}}(\delta_{\text{scal}} U_{\text{ext}})_{,x} \bar{U} - U \bar{U}_{,x} \\ &= -\frac{1}{\delta_{\text{scal}}}(\delta_{\text{scal}} U_{\text{ext}})_{,x} \bar{U} - U\, \boldsymbol{\phi}^T\, \dot{\mathbf{a}} \end{aligned} \tag{21}$$

Where the dot stands for partial derivative with respect to x (the time-like coordinate). Integrating in the normal direction:

$$\begin{aligned} V &= -\frac{1}{\delta_{\text{scal}}}(\delta_{\text{scal}} U_{\text{ext}})_{,x} \int_0^\eta \bar{U}(\eta')\, d\eta' - U \boldsymbol{\psi}^T \dot{\mathbf{a}} \\ &= -\frac{1}{\delta_{\text{scal}}}(\delta_{\text{scal}} U_{\text{ext}})_{,x}\, V_0 - U \boldsymbol{\psi}^T \dot{\mathbf{a}} \end{aligned} \tag{22}$$

where:

$$\begin{aligned} V_0 &= \int_0^\eta \bar{U}(\eta')\, d\eta' \\ \psi_k &= \int_0^\eta \phi_k(\eta')\, d\eta' \end{aligned} \tag{23}$$

Replacing in the momentum equation (7.a):

$$\begin{aligned} &U_{\text{ext}} \dot{U}_{\text{ext}} (\bar{U}^2 - 1) + U_{\text{ext}}^2 \bar{U} \boldsymbol{\phi}^T \dot{\mathbf{a}} \\ &\qquad - \left[\frac{1}{\delta_{\text{scal}}}(\delta_{\text{scal}} U_{\text{ext}})_{,x} V_0 + U_{\text{ext}} \bar{U}\, \boldsymbol{\psi}^T \dot{\mathbf{a}} \right] U_{,\eta} - \frac{\nu}{{\delta_{\text{scal}}}^2} U_{\text{ext}} \bar{U}_{,\eta\eta} = 0 \end{aligned} \tag{24}$$

In order to obtain a system of ODE's we weight these equations with a series of weight functions $\{w_j\}_{j=1}^{N_{\text{dof}}}$ and obtain a system of the form:

$$\mathbf{F}(\mathbf{a}, \delta_{\text{scal}}, \dot{\mathbf{a}}, \dot{\delta}_{\text{scal}}, x) = 0 \tag{25}$$

Let us assume for the moment that $\delta_{\text{scal}}(x)$ is known a priori, for instance $\delta_{\text{scal}} = c\sqrt{x}$. Then, these system of equations together with restrictions (19) is a system of $M + 1$ DAE's ($M-1$ ODE's and 2 linear restrictions) with $M+1$ unknowns. It can be integrated

by straightforwardly eliminating two of the a parameters (say a_0 and a_1) and obtaining a system of $N_{\text{dof}} = M - 1$ ODE's for the *"state vector"* $\mathbf{u} = [\, a_2 \quad a_3 \quad \ldots \quad a_M \,]^T$. A standard method like Runge-Kutta or any other high order method can be used to solve it numerically. Another possibility is to solve the full system of DAE's with a package like DASSL[8]. This last doesn't seem to be justified since the restrictions are linear. On the other hand, *what certainly should not be done* is to differentiate restrictions (19) in order to obtain a system of $M + 1$ ODE's, since this would lead to a reduction in convergence order.

System (25) is linear in $\dot{\mathbf{a}}$ and $\dot{\delta}_{\text{scal}}$ and can be put in the form:

$$\mathbf{A}\,\dot{\mathbf{a}} + \mathbf{c}\,\dot{\delta}_{\text{scal}} = \mathbf{b} \tag{26}$$

where all $\mathbf{A}, \mathbf{c}$ and $\mathbf{b}$ are functions of $\mathbf{a}, \delta_{\text{scal}}$ and x. Their expressions are:*

← verificar bien estas formulas

$$\begin{aligned}
A_{jk} &= U_{\text{ext}}^2 \left[\int_0^\infty w_j\, \bar{U}\, \phi_k \; d\eta - \int_0^\infty w_j\, \bar{U}\bar{U}_{,\eta}\, \psi_k \; d\eta \right] \\
c_j &= -\frac{{U_{\text{ext}}}^2}{\delta_{\text{scal}}} \int_0^\infty w_j\, \bar{U}_{,\eta}\, V_0 \; d\eta \\
b_j &= \int_0^\infty w_j \left[-U_{\text{ext}}\dot{U}_{\text{ext}}(\bar{U}^2 - 1) + \dot{U}_{\text{ext}}\, V_0 - \frac{\nu}{{\delta_{\text{scal}}}^2}\, U_{\text{ext}}\bar{U}_{,\eta\eta} \right] \; d\eta
\end{aligned} \tag{27}$$

2.5 Automatic normal scaling

As mentioned before, we propose to obtain the normal length scale δ_{scal} at the same time that the system is integrated. Suppose for instance that we want to set $\delta_{\text{scal}}(x) = \delta^*(x)$ the displacement thickness. That would mean that:

$$\begin{aligned}
U_{\text{ext}}\delta^* &= \int_0^\infty (U_{\text{ext}} - u) \; dy \\
&= U_{\text{ext}}\delta_{\text{scal}} \int_0^\infty (1 - \bar{U}) \; d\eta
\end{aligned} \tag{28}$$

so that, canceling out $U_{\text{ext}}\delta^* = U_{\text{ext}}\delta_{\text{scal}}$, we arrive to an additional restriction on the coefficients $\{a_k\}$:

$$1 = \int_0^\infty (1 - \bar{U}) \; d\eta = \sum_{k=0}^{M} \beta_k\, a_k \tag{29}$$

where:

$$\beta_k = \int_0^\infty [\phi_k(\eta = \infty) - \phi_k(\eta)] \; d\eta \tag{30}$$

Auto-scaling consists in solving (25) with δ_{scal} as an additional unknown together with restrictions (19,29). The same discussion as at the end of the previous section §2.4 on

how to reduce the system of DAE's to a system of ODE's or solving the system of DAE's with an appropriate algorithm applies. Again, we choose the first alternative. Note that, as the restriction (29) does not involve δ_{scal}, we can not eliminate δ_{scal}, so that we are obliged to eliminate a further a_k parameter, say a_2, and the *"state vector"* is now $\mathbf{u} = [\, a_3 \quad a_4 \quad \ldots \quad a_M \quad \delta_{\text{scal}} \,]^T$.

2.6 EXTENSION TO 3D

In the 3D case two main problems arise:

- There are two intrinsic coordinates on the surface so that the "time-like" analogy for the streamwise coordinate breaks down. We have to analyze the data dependency domain for the governing equations and choose an advance strategy.
- The statement of the boundary layer equations themselves is far more complex in 3D than in 2D. It's clear that for general surfaces one can not provide easily an orthogonal curvilinear system of intrinsic coordinates (ξ, ζ) on the surface, so that the use of tensor algebra concepts like *contravariant* and *covariant components*, *metric tensors*, *Christoffel symbols* and *covariant derivatives*[10].

2.7 Mesh-less discretization on the surface

In 3D there is no more a direct analogy to problems of temporal evolution. in this respect the governing equations resemble more the eqs. for pure advection without longitudinal and transverse diffusion where the advected quantities are the shape parameters in the normal direction. If (ξ, ζ) is a system of intrinsic coordinates on the surface, then the pseudo=-spectral discretization in the normal direction leads to a system of the form

$$\mathbf{F}(\frac{\partial \mathbf{u}}{\partial \xi}, \frac{\partial \mathbf{u}}{\partial \eta}, \mathbf{u}, \boldsymbol{\xi}) = 0, \tag{31}$$

where $\mathbf{u}$ is the vector of shape parameters (the coefficients in the Fourier expansion of both components of the velocity vector u and w). The *domain data dependency* is given locally by the cone including all the directions of the hodograph of the velocity at the point at the different normal positions. As this dependency cone has an aperture $< 180^\circ$ (save in particular cases as we will see later), there is some degree of freedom in the way the solution is advanced spatially.

Look at figure 2, the velocity at point O given by the intrinsic coordinates (ξ, ζ) at the outer edge of the boundary layer $\eta = \infty$ is the inviscid external velocity $\mathbf{U}$, whereas for $\eta = 0$ (at the wall) is $\mathbf{u} = 0$. For $0 < \eta < \infty$ the locus of velocities $\mathbf{u}(\xi, \eta, \zeta)$ describes a curve called hodograph as shown in the figure. This velocity vectors are enclosed in a cone AOB and the data dependency domain is obtained locally continuing downstream the lines AO and BO. The numerical solution is obtained advancing the solution by layers. For instance, known the unknowns in the layers

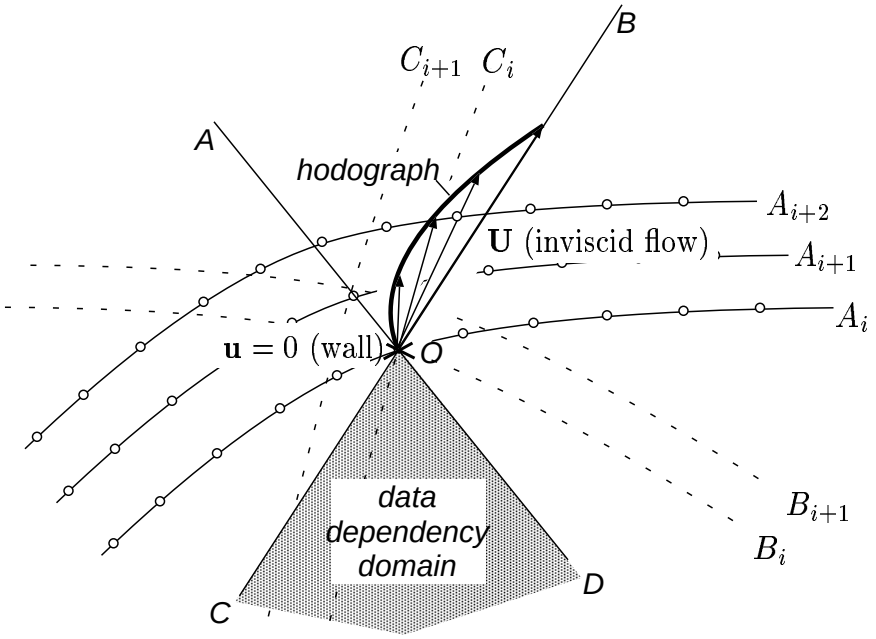

Figure 2: Data dependency domain

$A_{i-1}, A_{i-2}, \ldots$ we can obtain the unknowns at layer A_i, then on A_{i+1}, and so on. The only restriction is that when computing a given layer A_i is that all the points in the data dependency domain of A_i should be already computed, i.e. they should belong to layers $A_{i-1}, A_{i-2}, \ldots$. So, an advancing grid with layers arranged as $\ldots, B_i, B_{i+1}, \ldots$ is admissible, but $\ldots, C_i, C_{i+1}, \ldots$ *is not*. The constraint is stronger as the aperture angle of the cone is wider. Consider a separation bubble like that one depicted in figure 3. Assume that the inviscid field is almost parallel to the x-direction, then at a point A on the separation line the cone starts from the inviscid direction parallel to the x axis. While approaching the wall, the direction of the velocity vectors depart from this direction until that, for the layers of fluid very close to the wall, they align with the separation streamline. As the point A moves closer to the singular separation point S (point A') the angle comprised in the cone approaches 90°, ranging from the x direction to the positive z direction. But for a point A'' immediately on the other side of S the cone ranges from the x direction to the negative z direction, so that near S the dependency cone is 180° wide. This is not surprising, since separation involves a change in the governing equations from predominantly hyperbolic to predominantly elliptic.

Most boundary layer codes use a predetermined computational grid or intrinsic coordinate system. Most of them are based on the either free streamlines, constant x planes (where x is a global Cartesian coordinate parallel to the axis of the body, this is mainly used for fuselages), or constant percentage of chord for wings. It is clear that

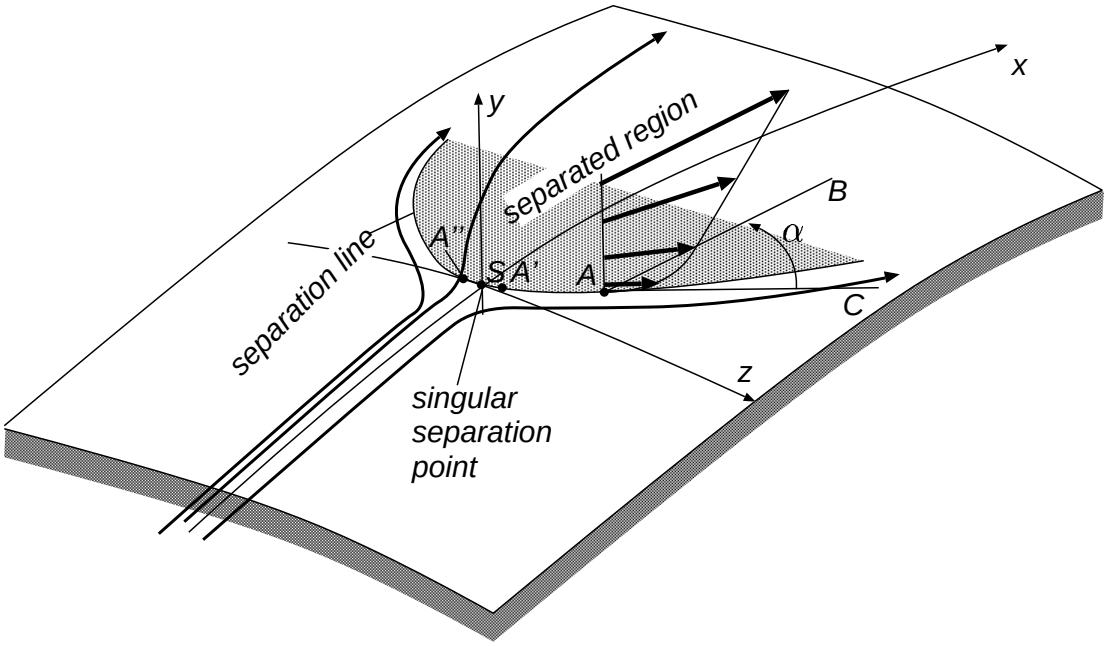

Figure 3: behavior of the data dependency domain near the separation line.

these codes will break well before the separation point, unless the grid point arrives (by mere coincidence) parallel to the separation line at the singular separation point. We propose then to adopt a scheme that could solve system (1) for a point P_1 given the data on a set of points $P_2, \ldots, P_n$ on its data dependency domain (see figure 4). This is equivalent do dynamically adapt the advancing front as the calculations proceed. A suitable numerical technique is the so called *mesh-less method*[11,13]. Its basis is to obtain a polynomial least-squares approximation to the discrete data on points $P_1, \ldots, P_N$ and approximate the derivatives appearing in the governing equation by the derivatives of the fitting polynomial. This strategy may be used in order to interpolate grid values at other points, not pertaining to the original discrete set. However, the most simple choice for the weights of interpolation leads to discontinuous interpolations. Much research is being done currently in order to obtain a continuous approximation, but for simplicity we will stick to the discontinuous one (which is called Fixed Least Squares Method).

Let us start with a scalar field $\phi(x)$ and let

$$\phi(\boldsymbol{\xi}) \approx \hat{\phi}(\boldsymbol{\xi}) = \sum_{k=1}^{M} a_k \, P_k(\boldsymbol{\xi}), \tag{32}$$

be the least squares approximation. Then forming a weighted least squares error functional

$$E(\mathbf{a}) = \sum_{j=1}^{N} w_j \, \left[\phi_j - \hat{\phi}(\boldsymbol{\xi}_j)\right]^2 \tag{33}$$

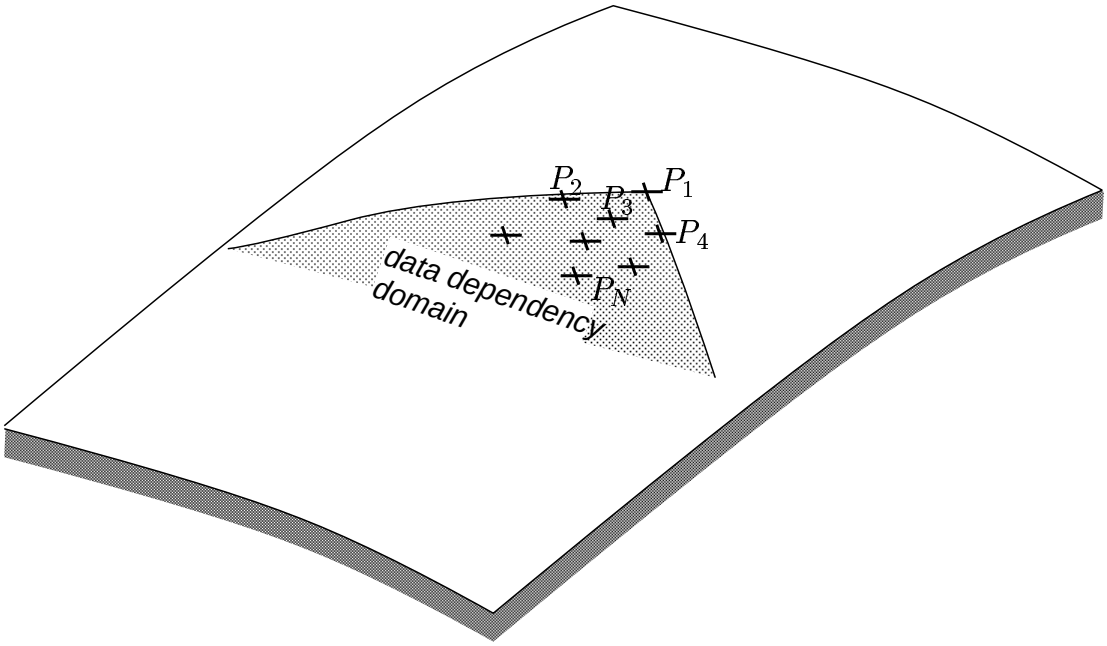

Figure 4: Stencil and data dependency domain for the boundary layer eqs.

and minimizing with respect to the free parameters $\mathbf{a}$ we obtain a system of the form

$$\mathbf{A}\mathbf{a} = \mathbf{b}, \tag{34}$$

with

$$\begin{aligned} A_{kl} &= \sum_j w_j \, P_k(\boldsymbol{\xi}_j) \, P_l(\boldsymbol{\xi}_j), \\ b_k &= \sum_j w_j \, P_k(\boldsymbol{\xi}_j) \, \phi_j. \end{aligned} \tag{35}$$

Then, an approximation to, say, $(\partial\phi/\partial\xi)$ is,

$$\begin{aligned} \frac{\partial\phi}{\partial\xi}(\boldsymbol{\xi}_i) &\approx \frac{\partial\hat{\phi}}{\partial\xi}(\boldsymbol{\xi}_i) \\ &= \sum_k a_k \, \frac{\partial P_k}{\partial\xi}(\boldsymbol{\xi}_i) \\ &= \left(\frac{\partial\mathbf{P}}{\partial\xi} \, \mathbf{A}^{-1} \, \mathbf{W} \, \mathbf{P} \right) \, \boldsymbol{\phi} \\ &= \mathbf{c}^T \, \boldsymbol{\phi}. \end{aligned} \tag{36}$$

$\mathbf{c}$ is a *"numerical stencil"* for the approximation to $\frac{\partial\phi}{\partial\xi}$. Now, let us describe how the system (1) is advanced. First the numerical stencil for the approximation to $\frac{\partial\phi}{\partial\xi}$ and $\frac{\partial\phi}{\partial\eta}$

is obtained in the form

$$\begin{aligned}\frac{\partial\phi}{\partial\xi} &\approx \sum_{j=1}^{N} c_j^{\xi}\,\phi_j,\\ \frac{\partial\phi}{\partial\eta} &\approx \sum_{j=1}^{N} c_j^{\eta}\,\phi_j.\end{aligned} \tag{37}$$

Then, replacing in (1) a non linear system of eqs. in $\mathbf{u}_j$ is obtained,

$$\mathbf{F}\left(\sum_{j=1}^{N} c_j^{\xi}\,\phi_j, \sum_{j=1}^{N} c_j^{\eta}\,\phi_j, \mathbf{u}_j, \boldsymbol{\xi}_j\right)\mathbf{0} \tag{38}$$

This system is solved iteratively with Newton-Raphson's method. In order to reduce the number of iterations, the scheme is started with a "predicted value" $\mathbf{u}_j^{\text{pred}}$ obtained by the same kind of meshless approximation but not including $j = 0$ in the cloud,

$$\phi_j^{\text{pred}} = \sum_{j=2}^{N} c_j^0\,\phi_j \tag{39}$$

Note that the sum starts at $j = 2$. Usually, 2 or 3 iterations are enough for a reduction in the residual by a factor 10^{-6} or more. However, the rate of convergence rapidly deteriorates near the separation point.

2.8 Boundary layer equations in tensorial form

A more detailed presentation of the tensorial form of the boundary layer equations can be found in a technical report[1]. We will give here only a brief summary of the resulting equations. Let (ξ, ζ) be a curvilinear system of coordinates on the surface, and η a (locally) orthogonal coordinate. We will also denote $x^1 = \xi$, $x^2 = \zeta$. The following definitions are standard in tensor calculus:

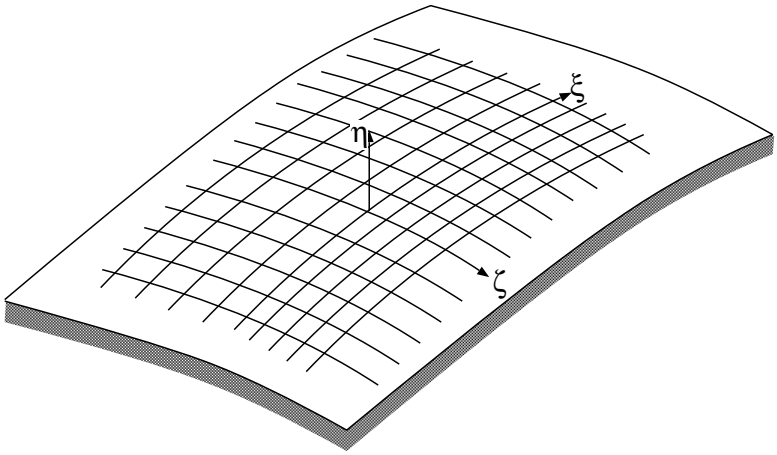

Figure 5: Intrinsic curvilinear coordinates

- The *covariant metric tensor* $g_{\alpha\beta}$ is defined as

$$g_{\alpha\beta} = \sum_{p=1}^{3} \frac{\partial y^p}{\partial x^\alpha} \frac{\partial y^p}{\partial x^\beta}, \tag{40}$$

  where Greek indices run on the surface coordinates $\alpha, \beta = 1, 2$. $\{y^p\}, p = 1, 2, 3$ is a set of Cartesian coordinates. It is easy to show that the resulting metric tensor is independent of the particular Cartesian system used in the definition.
- The *contravariant metric tensor* is defined as the inverse (in the usual matrix sense),

$$g_{\alpha\beta}\, g^{\beta\gamma} = \delta_\alpha{}^\gamma, \tag{41}$$

  where $\delta_\alpha{}^\gamma$ is the identity matrix,

$$\delta_\alpha{}^\gamma = \begin{cases} 1; & \text{if } \alpha = \gamma \\ 0; & \text{otherwise} \end{cases} \tag{42}$$

  We also assume in such expressions that repeated indices (like β in the previous eq.) are summed implicitly. Such indices should appear always once as contravariant (supraindex) and the other as covariant (subindex).
- The *Christoffel symbols of the first kind* are,

$$[\beta\gamma, \alpha] = \sum_p \frac{\partial y^p}{\partial x^\alpha} \frac{\partial^2 y^p}{\partial x^\beta\, x^\gamma}. \tag{43}$$

  They are also independent of the Cartesian system $\{y^p\}$.
- The *Christoffel symbols of second kind* are defined as,

$$\left\{ \begin{matrix} \alpha \\ \beta\ \gamma \end{matrix} \right\} = g^{\alpha\mu}\ [\beta\gamma, \mu] \tag{44}$$

- The *covariant derivative* is defined as,
  - ▷ for a scalar ϕ as simply the partial derivative,

$$\phi_{,\alpha} = \frac{\partial \phi}{\partial x^\alpha} \tag{45}$$

  - ▷ for a *contravariant vector* u^α as,

$$u^\alpha_{,\beta} = \frac{\partial u^\alpha}{\partial x^\beta} + \left\{ \begin{array}{c} \alpha \\ \beta\ \gamma \end{array} \right\} u^\alpha \tag{46}$$

  - ▷ for a *covariant vector* u_α as,

$$u_{\alpha,\beta} = \frac{\partial u_\alpha}{\partial x^\beta} - \left\{ \begin{array}{c} \gamma \\ \alpha\ \beta \end{array} \right\} u_\gamma \tag{47}$$

  - ▷ for a tensor of arbitrary order, as a combination of the previous expressions on each index (covariant or contravariant) of the tensor, independently,

$$(\tau^\alpha_{\ \gamma})_{,\beta} = \frac{\partial \tau^\alpha_{\ \gamma}}{\partial x^\beta} + \left\{ \begin{array}{c} \alpha \\ \beta\ \mu \end{array} \right\} \tau^\mu_{\ \gamma} - \left\{ \begin{array}{c} \mu \\ \gamma\ \beta \end{array} \right\} \tau^\alpha_{\ \mu} \tag{48}$$

The point about the covariant derivative is that it transforms as a covariant vector, whereas the standard partial derivative has not a transformation law, so that when a particular expression involves derivatives in a Cartesian system, they will result likely in covariant derivatives when formulated in a general curvilinear system.

The tensorial form of the boundary layer equations in tensorial form are:

$$\begin{aligned} u^\alpha u^\beta_{\ ,\alpha} + v \frac{\partial u^\beta}{\partial \eta} &= -\frac{\nu}{\delta^2} \frac{\partial^2 u^\beta}{\partial \eta^2} + g^{\beta\gamma} p_{,\gamma} \\ \frac{1}{\delta} \left(\delta u^\alpha\right)_{,\alpha} + \frac{\partial v}{\partial \eta} &= 0 \end{aligned} \tag{49}$$

This system is invariant under mappings of the intrinsic coordinates, i.e. $x^1, x^2 \to \bar{x}^1, \bar{x}^2$. Also $\delta(\boldsymbol{\xi})$ is an arbitrary length scale for the normal coordinate. The system of eqs. is valid for whatever we choose for δ, but the idea is to choose it as close as possible to the real (displacement or momentum) thickness of the boundary layer profile. In this respect we follow with the idea of automatically scaling with δ as it is being computed.

As p is a scalar $p_{,\alpha} = (\partial p / \partial \alpha)$, but usually it is more convenient to put the pressure gradient in the form,

$$g^{\beta\alpha}\, p_{,\alpha} = U^\alpha\, U^\beta_{\ ,\alpha} \tag{50}$$

where U^α are the contravariant component of the outer (inviscid) velocity field.

3 NUMERICAL RESULTS

There is a first set of 2D examples. Due to the abundance of numerical and experimental data, they allow an assessment of the precision of the method.

- ***Similar solutions. Wedge-flows***
- ***Spectra for wedge flows***
- ***Similar flow in a convergent channel***
- ***Howarth's flow***
- ***Circular cylinder***
- ***Ellipses***

With respect to the 3D examples there is a first set which is rather a verification of the code. It includes,

- ***Yawed flat plate***
- ***Yawed circular cylinder***
- ***Invariance under change of mapping***
- ***Cones***

The essentially 3D flows considered are,

- ***Sphere (potential outer flow)***
- ***Sphere (actual outer flow)***
- ***Sphere with rotation***

3.1. Similar solutions Wedge-flows

Since the paper is focused on the spectral approximation of the boundary layer equations we consider only numerical experiments with laminar flow. We start with the similar solutions of the Falkner-Skan equations for several values of the exponent m in the external velocity law $U \propto x^m$. Longitudinal velocity profiles U are shown in figure 6. The set of values for m's have been chosen as in the book of Schlichting[14] for ease of comparison. Also the abscissa has been scaled as in that reference. In order to make a qualitative comparison, we computed integral quantities (skin friction C_f, thickness δ^* and momentum displacement θ) for several number of parameters $M+1$ and compared with values computed by Hartree and found in (16) (see Tables I and II). (Note: the set of m values is different from those shown in figure 6).

It is interesting to see how the method converges as M is increased. For $m = 1$ the solution with four terms ($M = 3$) has a precision of 1% in the displacement thickness, 4% for the momentum thickness and 0.5% for the wall friction with respect to the values indicated by Schetz. As a reference, the method from von Kármán and Pohlhausen[14] gives errors of 1%, 5% and 3.5%. This errors are comparable except for the last one which is seven times greater for the von Kármán and Pohlhausen's method. This comparison is very interesting since it shows that the method not only has a high

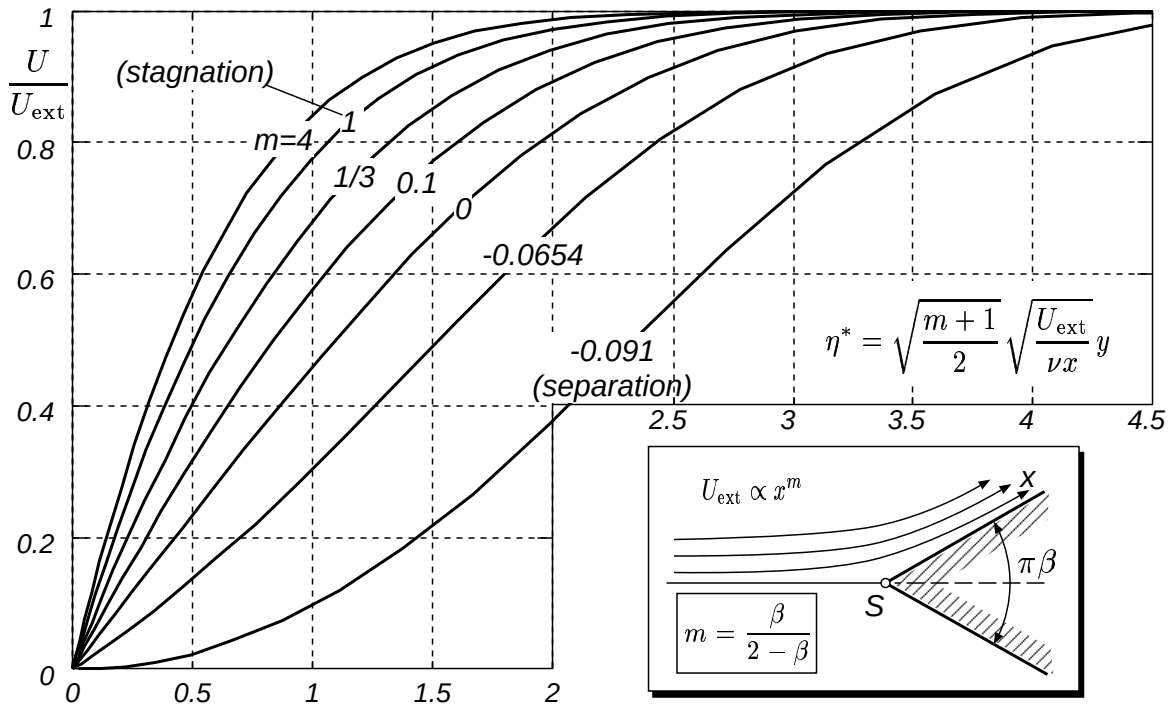

Figure 6: Longitudinal velocity profiles for (similar) wedge flows.

rate of convergence, but also is reasonably accurate for engineering computations with very few terms. Indeed, both methods are equivalent in the sense that with $M = 3$ our method has two degrees of freedom: the layer thickness and the remaining plays the role of the *"shape parameter"* in the von Kármán and Pohlhausen's method. On the other hand, it is enough to use $M = 7$ to have all the three integral quantities accurate to three digits. It is interesting also to see how the values converge while increasing the number of terms. For instance for $M + 1$ =8, 12, 18 and 24 terms the displacement thickness changes only in a 0.01%.

Convergence is slower for decelerated flows. In particular, for the flat plate $m = 0$ always with $M + 1 = 4$ terms the errors are 8%, 16% and 5% respectively, whereas they are 2%, 4% and 3% for the von Kármán and Pohlhausen's method. In this case the comparison is favorable to the second one. On the other hand we need 12 terms to find the solution accurate to three digits. Finally, for $m = -0.0904$, which is the value for which the flows reach separation according to Hartree, our method predicts a momentum thickness accurate to three digits, a difference of 0.5% for the momentum thickness and a wall friction of 0.01 (whereas it is null for the Hartree calculations).

3.2 Spectra for wedge flows

Spectral methods provide a simple form to estimate qualitatively the convergence of the method. As the approximation is *"hierarchical"*, in the sense that best resolution is achieved by the addition of new terms to the series, it is expected that if convergence

Table I

m	$M+1$	$C_f \operatorname{Re}_x^{1/2}$	$(\delta^*/x) \operatorname{Re}_x^{1/2}$	$(\theta/x) \operatorname{Re}_x^{1/2}$
1	4	2.45466	0.64101	0.28004
	6	2.46209	0.64613	0.29071
	8	2.46501	0.64787	0.29232
	12	2.46519	0.64792	0.29237
	18	2.46518	0.64790	0.29234
	24	2.46506	0.64793	0.29236
	Hartree	2.465	0.648	0.292
1/3	4	1.51736	0.96434	0.40234
	6	1.51074	0.98127	0.42543
	8	1.51511	0.98644	0.43004
	12	1.51490	0.98536	0.42898
	24	1.51490	0.98537	0.42899
	Hartree	1.515	0.985	0.429
0.1	4	1.00924	1.28990	0.49960
	6	0.98779	1.34479	0.55509
	8	0.99401	1.35255	0.56099
	12	0.99312	1.34765	0.55640
	24	0.99314	1.34786	0.55659
	Hartree	0.993	1.348	0.557
0	4	0.69913	1.58478	0.55410
	6	0.65832	1.74522	0.68966
	8	0.66563	1.72735	0.66970
	12	0.66408	1.72145	0.66472
	24	0.66411	1.72079	0.66411
	Hartree	0.664	1.721	0.664

is good the contribution from the terms with high indices would be negligible. To see this, we plot the absolute value of the coefficients a_k versus k for several m values in figure 7. It can be seen that coefficients decay strongly. For instance, for all values except $m = -0.0904$ the last coefficients ($k = 13, 14$) are below 10^{-6}, whereas the first ones are above 0.01. This convergence tends to deteriorate for decelerated flows, but even at separation ($m = -0.0904$) the last coefficient represents 10^{-3} of the first one.

Table II

m	$M+1$	$C_f \operatorname{Re}_x^{1/2}$	$(\delta^*/x) \operatorname{Re}_x^{1/2}$	$(\theta/x) \operatorname{Re}_x^{1/2}$
-0.01	4	0.66170	1.62682	0.55854
	6	0.61716	1.81453	0.71368
	8	0.62456	1.78590	0.68373
	12	0.62291	1.78105	0.67983
	24	0.62294	1.78004	0.67891
	Hartree	0.623	1.780	0.679
-0.05	4	0.49200	1.83731	0.56571
	6	0.42137	2.26311	0.88137
	8	0.42962	2.10911	0.74329
	12	0.42693	2.12121	0.75439
	24	0.42697	2.11775	0.75146
	Hartree	0.427	2.117	0.751
-0.0904	4	0.26503	2.17629	0.51471
	6	0.06094	4.10997	1.60755
	8	0.08120	2.95492	0.72862
	12	0.03321	3.29244	0.83852
	24	0.01031	3.44190	0.86797
	Hartree	0	3.428	0.868

3.3 Similar flow in a convergent channel

A related family of similar flows correspond to outer velocity profiles of the form $U_{\text{ext}} \propto |x|^m$ for $x < 0$. In this case, $m < 0$ corresponds to accelerated flow and $m > 0$ to decelerated flow. It is particularly interesting the case $m = -1$ since in this case an exact solution is available. The solution is[14]:

$$\frac{u}{U} = 3 \tanh^2 \left(\frac{\eta^*}{\sqrt{2}} + 1.146 \right) - 2 \tag{51}$$

where $\tanh^{-1} 2/3 = 1.146$ and $\eta^* = (y/x)/\operatorname{Re}_x^{1/2}$. In figure 8 we see the computed profile with four terms and the exact profile. The displacement thickness is $\delta^* = 0.744$, whereas the exact one is 0.778 and von Kármán and Pohlhausen's method predicts $\delta^* = 0.5952$. The error is 0.5% for the proposed method, whereas it is as high as 24% for the von Kármán and Pohlhausen's one.*In figure 9 we can see the maximum error (on a fixed grid of $N = 2^8$ points) of the solution for different number of terms. Note that the ← verificar esto!!

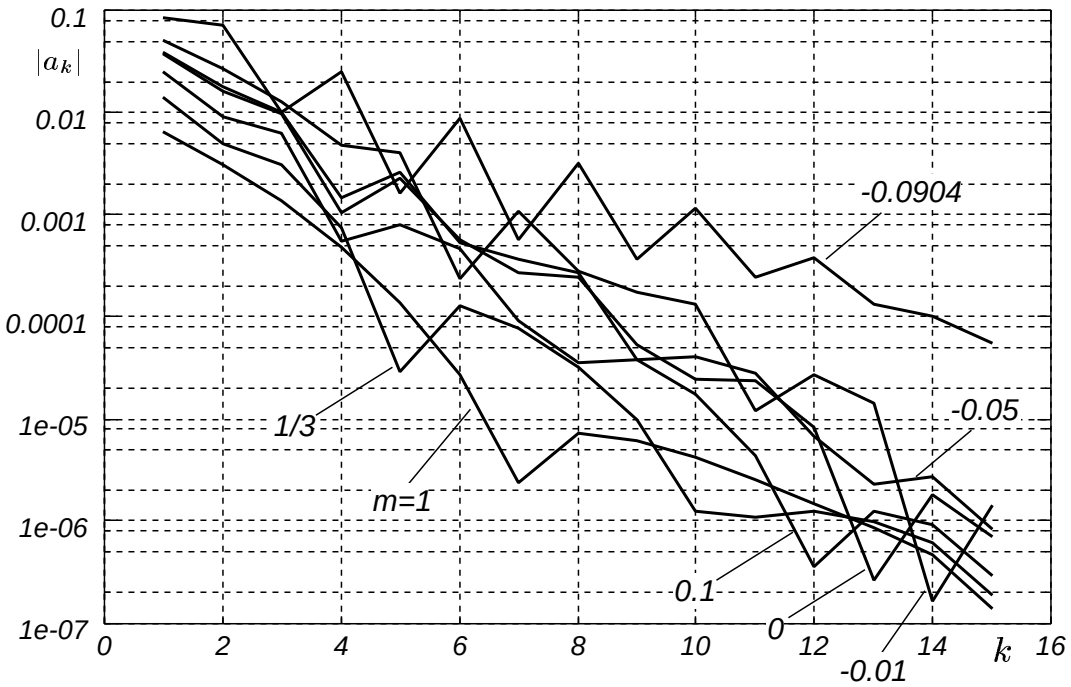

Figure 7: Spectra of coefficients in the Fourier series for similar flows.

vertical axis is logarithmic and the horizontal one is linear, so that a straight line will represent an error $\sim e^{-\gamma M}$, which fits within the concept of *"spectral convergence"*. We see that, even if a certain deceleration in the rate of convergence is observed, an ultimate rate of convergence of 1 order of magnitude each 20 terms is achieved. Even if asserting "spectral convergence" is rather subjective, the rate of convergence is quite high with respect to finite precision methods.

3.4 Howarth flow

This corresponds to a decelerated external flow given by:

$$U(x) = U_0 - ax \tag{52}$$

The time-like coordinate is made non-dimensional by defining $x^* = ax/U_0$. L. Howarth obtained accurate solutions by series expansions whose results can be found in Schlichting's book[14]. Numerical results have been obtained with a number of methods as described in the book of White[4] and recently using finite elements in the normal coordinate by Schetz[3]. In figure 10 we can see the variation of the displacement thickness and wall friction and longitudinal velocity profiles in figure 11. We detected separation at $x^* = 0.01195$ which is in very good coincidence with both the value of $x^* = 0.0119863$ reported by Wippermann using finite differences and taken by White as the reference value, and also with the numerical results from Schetz.

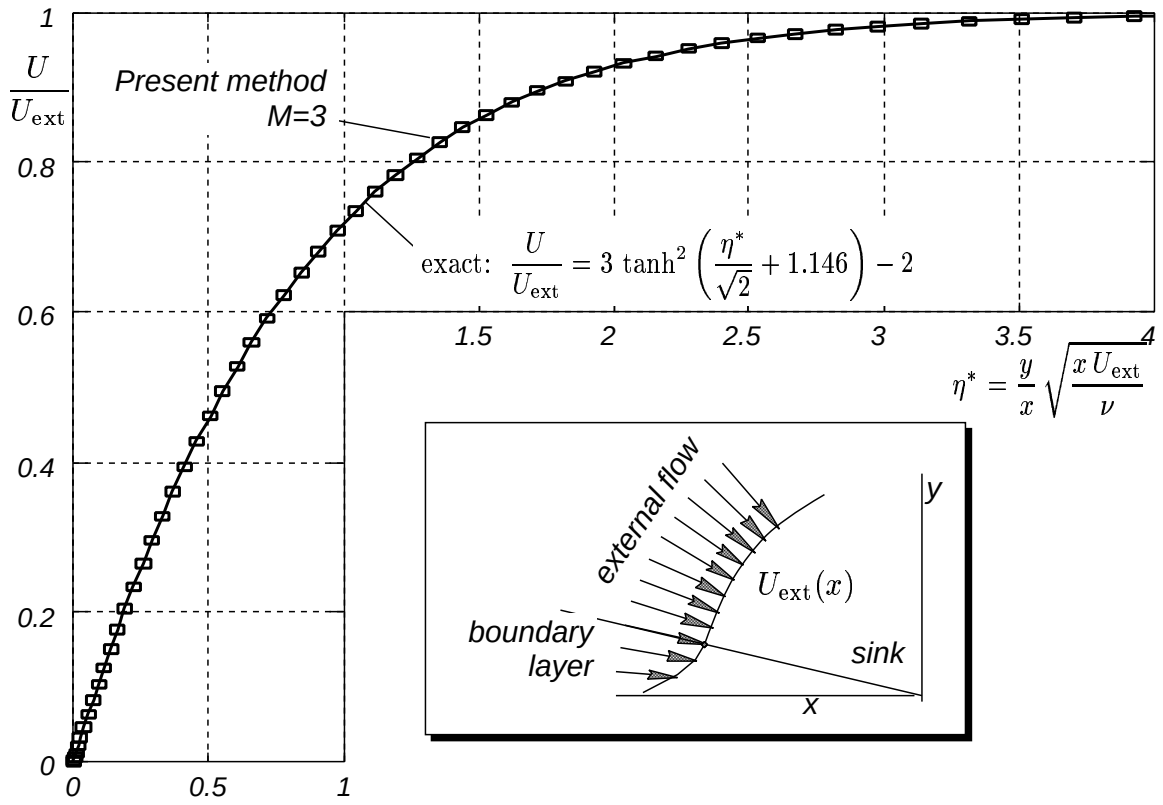

Figure 8: Flow in a convergent channel $U_{\text{ext}} \propto |x|^{-1}$, $(x < 0)$

3.5 Yawed circular cylinder

The inviscid potential around a cylinder is $U_{\text{ext}} = 2U_\infty \sin(\phi)$, where $\phi = x/R$ and x is the arc length on the cylinder. It is well known that the flow separates somewhere slightly after the point of maximum pressure, so that the boundary layer calculations for this expression are of relative practical importance. However, there are accurate calculations via series expansions for this flow, so that it serves as a test for the numerical method. On the other hand, for flows of practical interest over aerodynamic profiles where the flow do not separate, the series do not converge, specially due to the large variation of U_{ext} near the nose. In figure 12 we see the variation of displacement thickness and wall friction whereas longitudinal velocity profiles at several stages are shown in figure 13. There is a rather large discrepancy with respect to the location of the separation point. Whereas both the Blasius series and the von Kármán and Pohlhausen's method predicts separation near $\phi_S = 109.5^\circ$, our method gives $\phi_S = 103.8^\circ$. This seems to be in accordance with more precise calculations using finite differences and MWR as described in the book of White[4], which report $\phi_S = 1 - 4.5^\circ$.

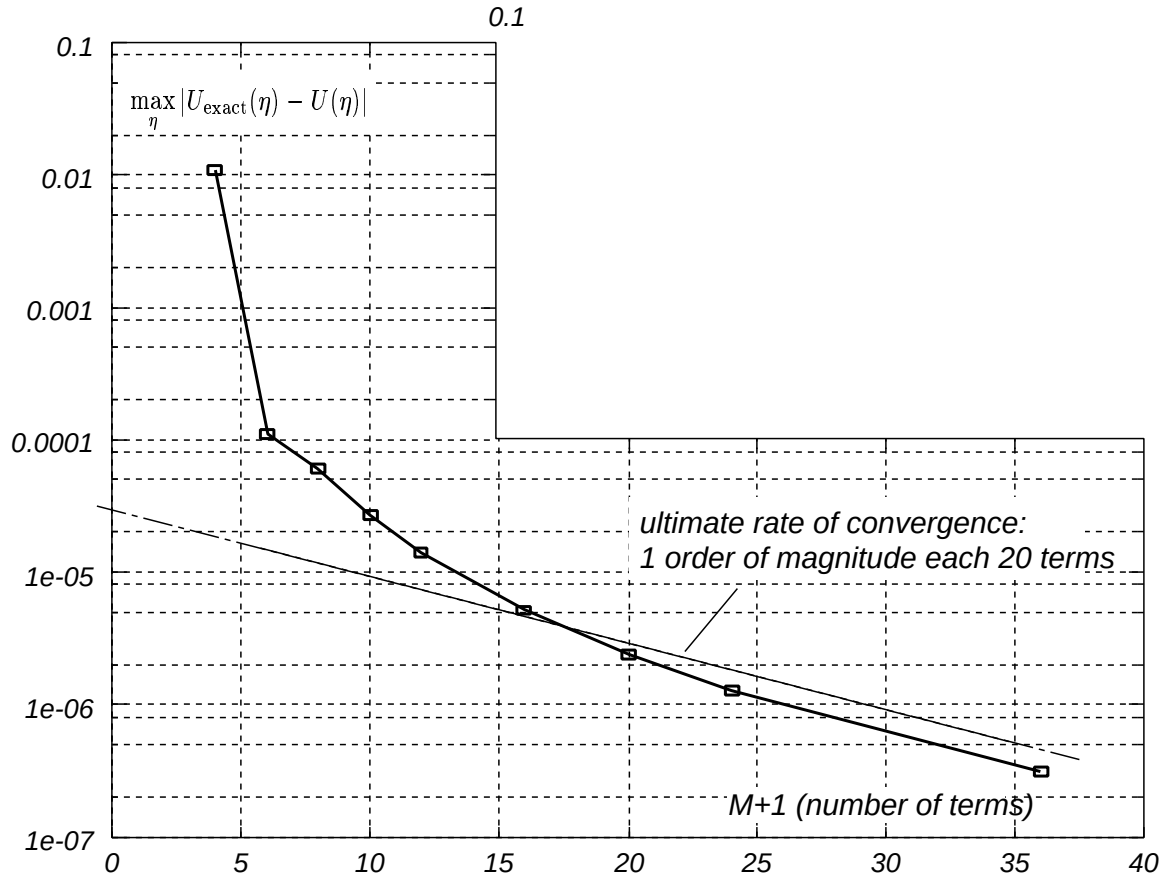

Figure 9: Maximum error versus number of terms for the convergent channel flow ($U_{\text{ext}} \propto -|x|^{-1}$, for $x < 0$).

3.6 Ellipses

Results for flow around ellipses, when the undisturbed flow is oriented with the major axis of the ellipse are shown in figures 14, 15 and 16. Similar results using the approximate method from von Kármán and Pohlhausen are shown in the book from Schlichting[14]. The circular cylinder discussed in the previous section corresponds to the particular case: $a/b = 1$. Also, the flat plate can be included if considered as an ellipse with $a/b \to \infty$.

3.7 Yawed flat plate

Consider a uniform flow impinging on a flat plate at 45° with respect to its leading edge. It can be shown that the boundary layer solution is the same (along each streamline) as if the plate were non-yawed. We verified that the displacement thickness evolves as $\delta^* = 1.721\sqrt{\sqrt{2}\xi}$, where the factor $\sqrt{2}$ takes into account the inclination of the streamline with respect to the ξ axis. Also the u and v velocity components are

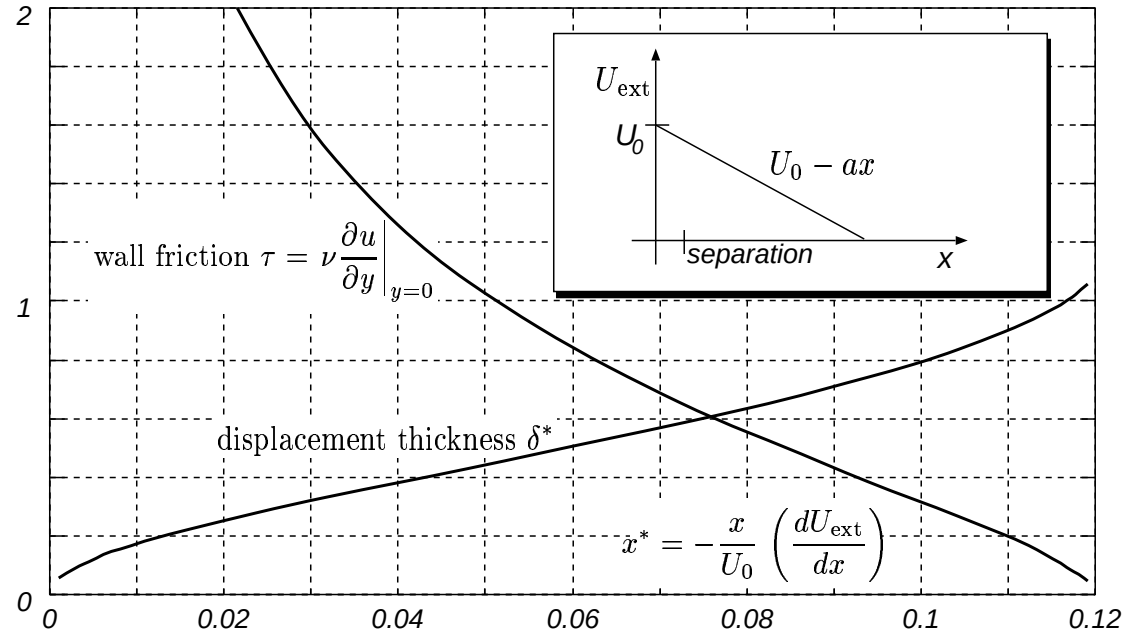

Figure 10: Displacement thickness and wall friction for Howarth's flow $U_{\text{ext}}(x) = U_0 - ax$

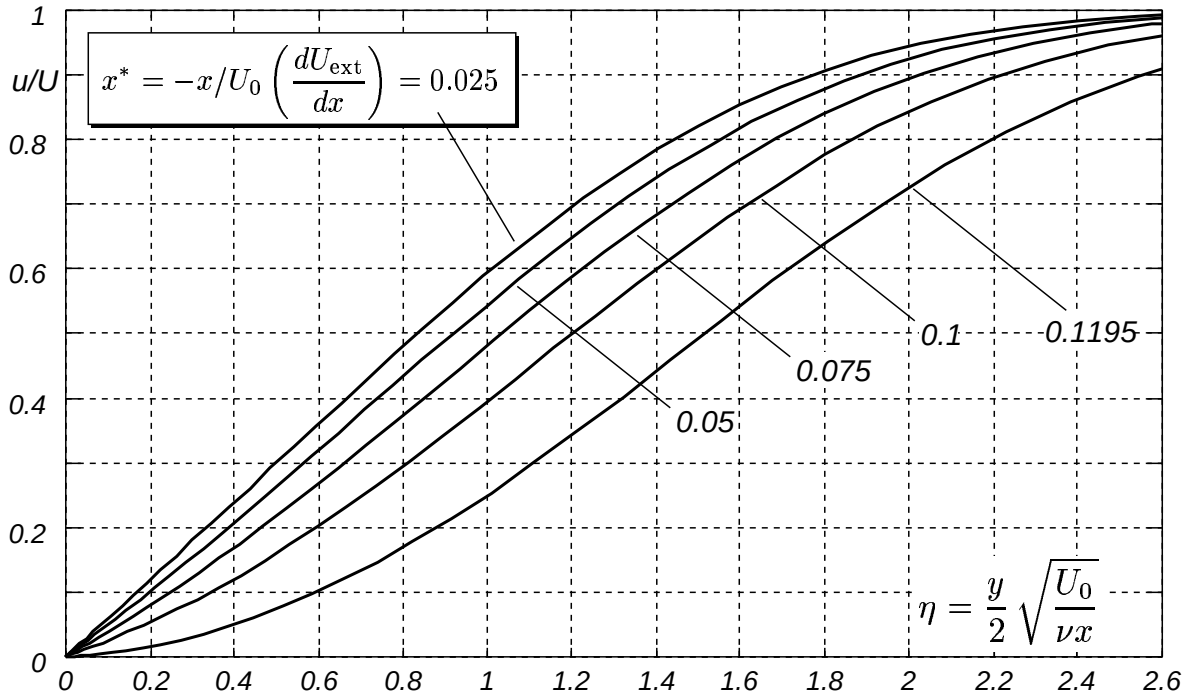

Figure 11: Longitudinal velocity profiles for Howarth's flow.

proportional (equal, in fact, for incidence a 45°).

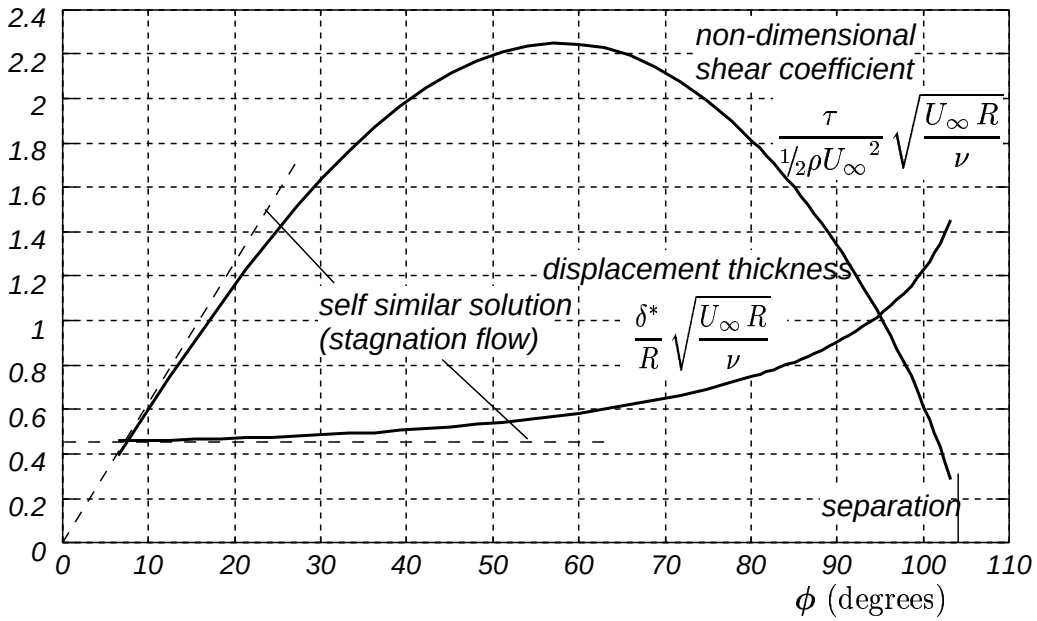

Figure 12: Displacement thickness and wall friction for flow around a circular cylinder $U_{\text{ext}} = 2U_\infty \sin(x/R)$

3.8 Yawed cylinder

This case corresponds to a yawed cylinder such that the non-perturbed flow impinges at 45° to its axis (see figures 17 and 18). As indicates the theory, the velocity component normal to the axis of the cylinder is the same as for the non-yawed cylinder. However the limit streamlines, i.e. the streamlines for the fluid immediately over the wall) tend to curve near the separation point and to align with the separation streamline, which for cylinders is always an x =cnst curve. The computations is with the potential velocity $(u = 2 * sin(\phi))$ and separation occurs at the same point as for the non-yawed cylinder, i.e. at 104.45° which is in very good agreement with computations reported in the literature.

3.9 Invariance under change of mapping

Here we solve the standard Blasius flow but including an artificial mapping of the form;

$$\begin{aligned} \xi &= x \\ \zeta &= z + a\, f(x)\, \sin(2\pi z) \end{aligned} \tag{53}$$

with,

$$f(x) = \begin{cases} [(x-0.5)(1-x)]^2/0.25^4 & ;\ \text{for } 0.5 < x < 1; \\ 0 & ;\ \text{otherwise} \end{cases} \tag{54}$$

and $a = 0.05$. For this value of a the maximum and minimum values of relative stretch-

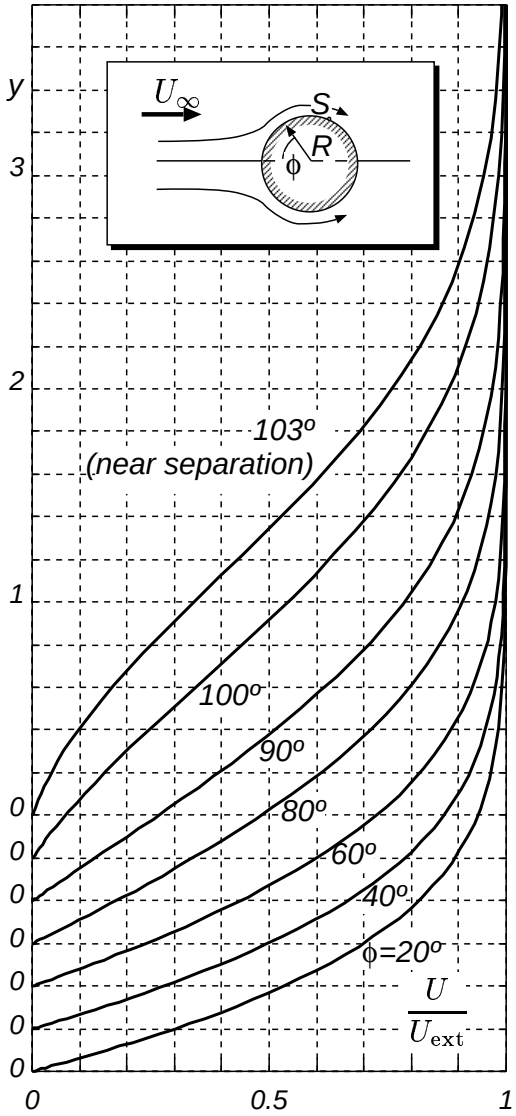

Figure 13: Longitudinal velocity profiles for flow around a circular cylinder.

ing of the iso-ζ lines is in a ratio of 4:1. Here x and z are Cartesian coordinates on the plate, such hat z is parallel to the leading edge and x perpendicular to it, and parallel to the flow. Solving the problem in the distorted coordinates should give the correct result, which will show that the 3D equations are correct. Off course discretization coupled with mesh distortion introduces an error, and the results in the distorted coordinates tend to approach the correct ones only with refinement. We checked that the maximum relative variation in the displacement thickness at $x = 1.2$ was 2×10^{-4}.

3.10 Cones

In 3D, there are two kind of similar flows that have been used for checking the code:

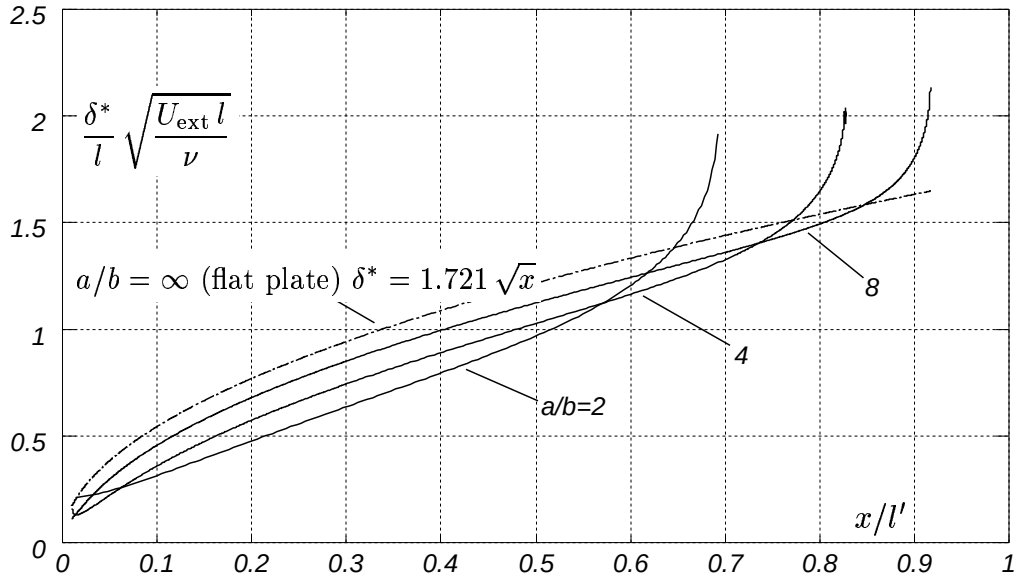

Figure 14: Displacement thickness for flow around ellipses of various slenderness.

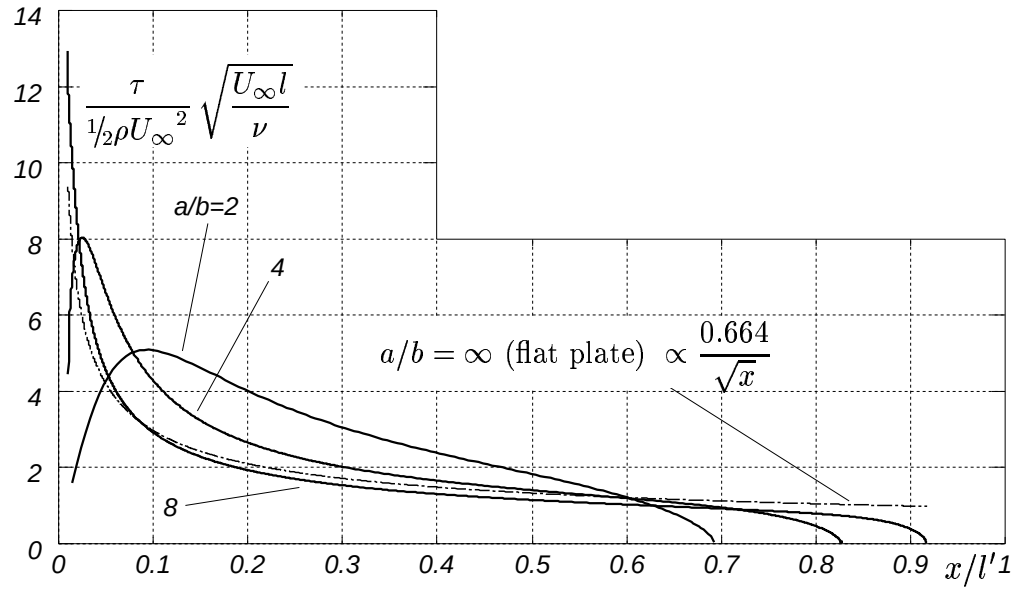

Figure 15: Wall friction or flow around ellipses of various slenderness

yawed wedge flows and *cone flow*. In both cases the external velocity potential is $\propto x^m$. For yawed wedge flow, the normal flow is the same (as usual for cylinders) to the non yawed flow (already shown in the 2D paper[15]). In the case of cone flow, it can be shown that it is equivalent to a wedge 2D flow with a external velocity $\propto x^{3m}$, whereas the displacement thickness is $\sqrt{3}$ to the equivalent wedge flow. This has been checked numerically but will not be reported in detail here.

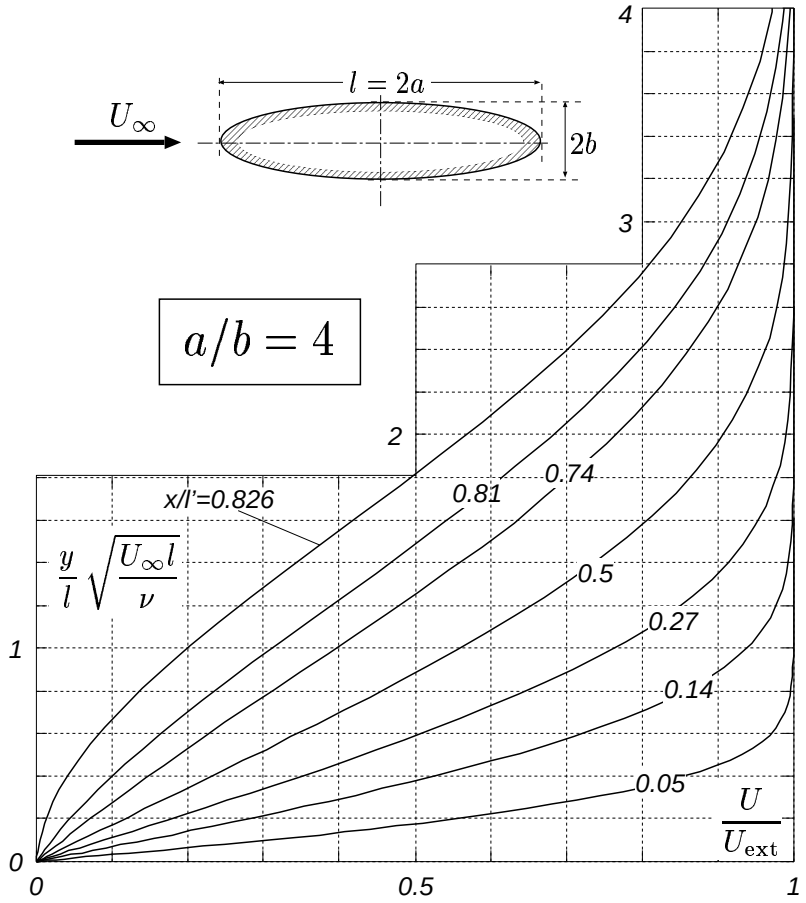

Figure 16: Longitudinal velocity profiles for flow around slenderness $a/b = 4$.

3.11 Sphere (potential velocity distribution)

This is an axysymmetric flow without lateral (azimuthal) component of velocity. Using the velocity distribution given by the potential flow $U = 1.5\sin(\phi)$ we obtain the results for wall friction and displacement Thickness shown in figure 19. The separation point is detected at $\phi = 105.45^\circ$ which is in very good agreement with the value recommended by White as being between 104° and 106°.

3.12 Sphere (actual velocity distribution)

The actual velocity distribution can be approximated with very good precision as

$$\frac{U}{U_0} = 1.5(x/a) - 0.4371\,(x/a)^3 + 0.1481\,(x/a)^5 - 0.0423\,(x/a)^7 \tag{55}$$

With this velocity distribution the code predicts separation at $\phi = 81.6^\circ$ which is again in very good agreement with other computations and with the experimental value of 83° reported by Fage.

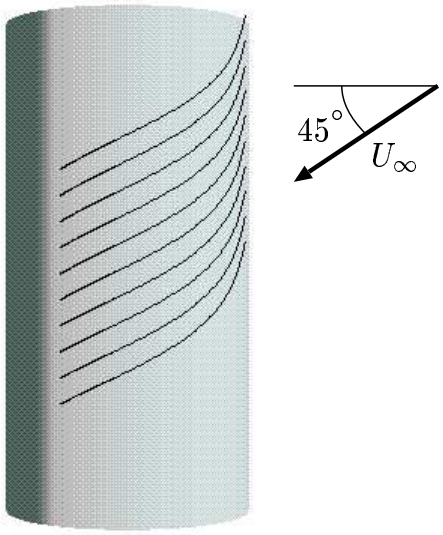

Figure 17: Inviscid streamlines.

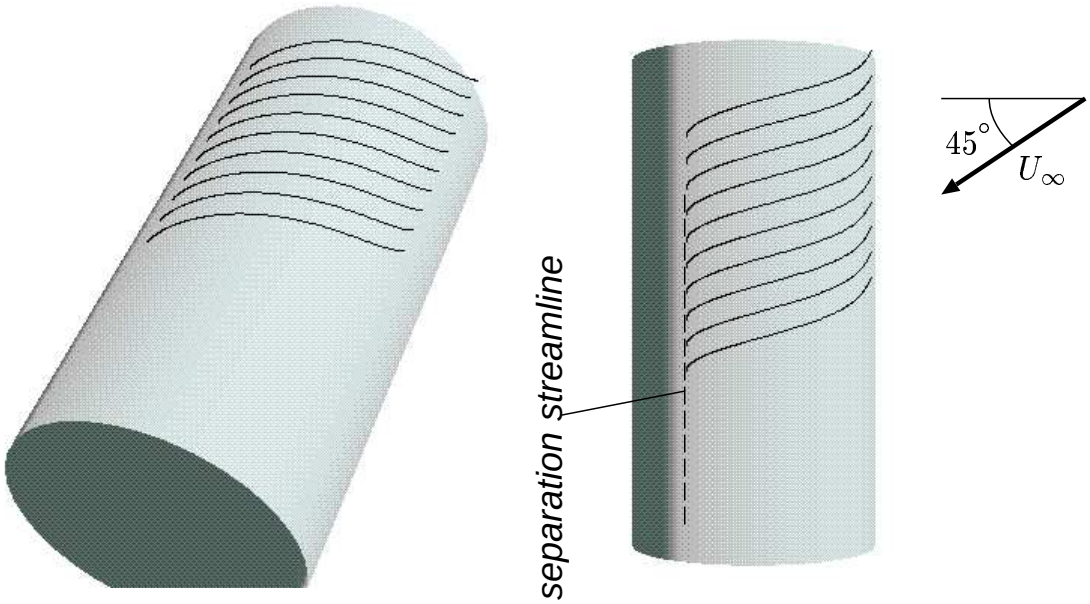

Figure 18: Computed viscous streamlines.

3.13 Rotating sphere

In this case take the the actual distribution of velocity (55) but the sphere is rotating with an angular velocity $\omega R/U_\infty = 1$, about an axis parallel to the free stream velocity. Whereas the inviscid streamlines are simply meridians, the limit viscous streamlines have a tendency to rotate with the sphere, until they align with the separation streamline that is a parallel at 84.2° with respect to the pole facing the fluid. Note that rotation tends to stabilize the boundary layer against separation, resulting in a delay of almost

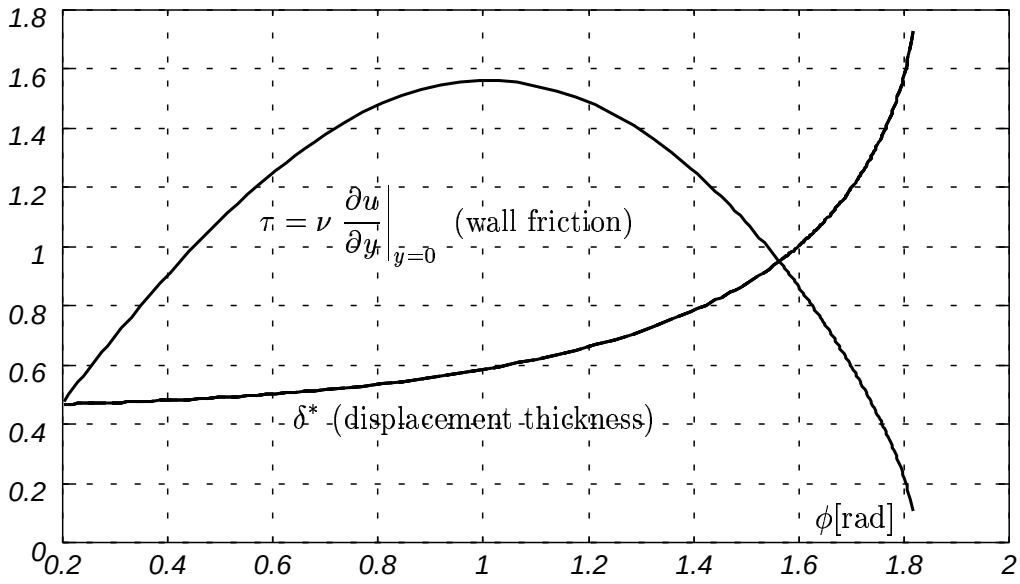

Figure 19: Results for the sphere with velocity distribution given by potential flow.

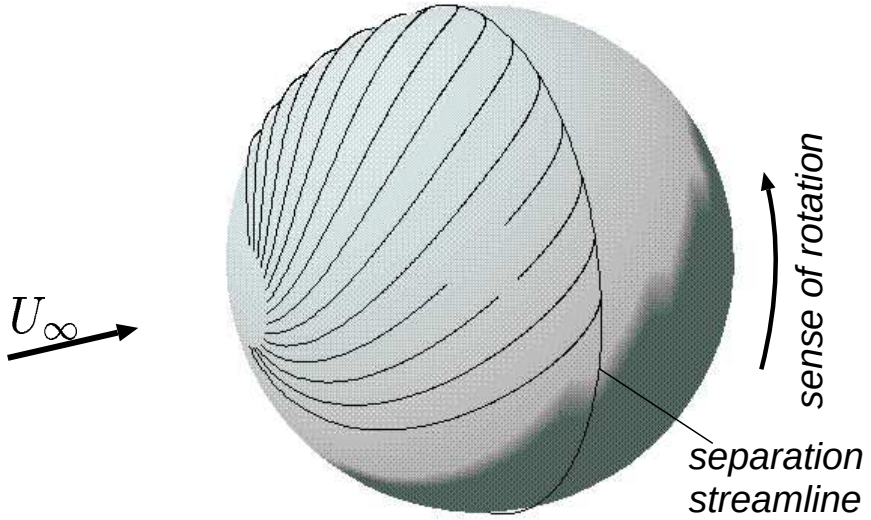

Figure 20: Limit streamlines for the rotating sphere.

3 degrees. This is due to the centrifugal force that can be assimilated to a pressure gradient directed to the equator. Since the separation for the sphere happens before the equator this is equivalent to a favorable pressure gradient. This has a significant incidence in the drag also.

4 CONCLUSIONS

A spectrally accurate algorithm for the solution of the boundary layer equations with automatic scaling of the normal coordinate is presented. The scaling amounts to assume

the scaling length as a further unknown and adding the corresponding restriction as a constraint. The resulting system of Differential-Algebraic Equations (DAE) is solved by eliminating one parameter in the expansion and solving the resulting system of ODE's. The spectral approximation is based on a direct mapping from the semiinfinite domain to a periodic problem and using a Fourier expansion, instead of truncating the semiinfinite domain and mapping to a bounded interval and using a Tchebyschev expansion. Several numerical results are presented, and spectral convergence is demonstrated by analysis of the decay rate of the coefficients in the expansion for similar flows, and by straightforward computation of the maximum error in the longitudinal velocity profile for the special case of flow towards a sink, for which an analytic expression is available. Also, the accuracy of the method when very few parameters are used was analyzed, by comparison with von Kármán and Pohlhausen's method.

REFERENCES

[1] M. Storti, "The incompressible Boundary Layer Equations in Tensor Form", CIMEC internal report. (1997, in spanish)

[2] C.Pruett "On the Wall-Normal Velocity of the Compressible Boundary Layer Equations", NASA-CR 4419 (1991)

[3] Schetz, J.A., E Hytopoulos and M. Gunzburger, "Numerical solution of the boundary layer equations using the finite element method", *FED-vol 123, Advances in Finite Element Analysis in Fluid Dynamics*, M.N. Dhaubhadel, et.al. (edts), ASME (1991)

[4] FM. White, "Viscous Fluid Flow", McGraw-Hill, New York (1974)

[5]. Gottlieb, D., Orszag, S.A., "Numerical Analysis Of Spectral Methods: Theory And Applications", SIAM (1977)

[6] T.A. Zang, C. Streett, and M.Y. Hussaini *"Spectral methods for* CFD*"*, ICASE Report No. 89-13, (1989)

[7] C.D. Pruett and C.L. Streett, "A Spectral Collocation Method for Compressible, Non-Similar Boundary Layers", *Int. J. Num. Meth. Fluids* , **13**, pp. 713-737 (1991)

[8] K. E. Brenan, et al., *"Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations"*, North-Holland (1989)

[9] Voigt, R.G., Gottlieb, D., Hussaini, M.Y., Eds. *"Spectral Methods For Partial Differential Equations"*, SIAM (1984) Houssay

[10] E Hytopoulos, J.A. Schetz and M. Gunzburger "Numerical Solution of the Compressible Boundary Layer Using the Finite Element Method", *AIAA J.*-92-0666 (1992)

[11] J. Batina, A Gridless Euler/Navier-Stokes SOlution Algorithm for Compressible Aircraft Applications, AIAA 93-0333, (Reno NV, Jan. 11-14, 1993)

[12] "A Finite Point Method in Computational Mechanics. Applications to Convective Trnsport and Fluid Flow", by E. Oñate, S.R. Idelsohn, O.C. Zienkiewicz and R.L. Taylor, *International Journal for Numerical Methods in Engineering*, **39**, Nr, 22, pp 3839-3886 (1996)

[13] "A Stabilized Finite Point Method for Analysis of Fluid Mechanics Problems", by E. Oñate, S.R. Idelsohn, O.C. Zienkiewicz, R.K. Taylor, and C. Sacco. To appear in *Comp. Meth. App. Mech. Engng.* (1996)

[14] Schlichting, *"Boundary Layer Theory"*, Pergamon Press (1955)

[15] M. Storti, "A Pseudo-Spectral Approach for the Incompressible Boundary Layer Equations with Automatic Normal Scaling", submitted to *Int. J. Num. Meth. Fluids* . (1997)

[16] Schetz, J.A. *"Foundation of Boundary-Layer Theory for Momentum, Heat and mass Transfer"*, Prentice Hall, Englewood Cliffs, (1984)

[17] Y.S. Wie and J.E. Harris "Numerical Solution of the Boundary layer Equations for a General Aviation Fuselage", *J. Aircraft* , vol 28, No 12, 861-868 (1991)