The computation of fluid flow in a curvilinear non-orthogonal coordinate system.

Honglin. Ye  
*University of Windsor*

Follow this and additional works at: https://scholar.uwindsor.ca/etd

**Recommended Citation**  
https://scholar.uwindsor.ca/etd/3799

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.
THE COMPUTATION OF FLUID FLOW IN A
CURVILINEAR NON-ORTHOGONAL COORDINATE SYSTEM

by

Honglin Ye

A Thesis
Submitted to the Faculty of Graduate Studies and Research
through the Department of Mathematics and Statistics in
Partial Fulfillment of the Requirements for the
Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada
1991

Typeset by AMS-TEX
The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

THE COMPUTATION OF FLUID FLOW IN A CURVILINEAR NON-ORTHOGONAL COORDINATE SYSTEM

by
Honglin Ye

ABSTRACT

A curvilinear coordinate system $z^i$ has been introduced to study $2 - D$ flow in complex geometry. Some flow problems of current interest have been formulated in the $z^i$ coordinate system. The methods for numerically generating the $z^i$ coordinate system are discussed. A new design and analysis algorithm is proposed for inviscid incompressible flow. The solution procedure for Navier-Stokes equations is presented for an expanding channel flow. Numerical results from the present method show comparable accuracy to existing procedures.
ACKNOWLEDGEMENTS

The author thanks Dr. R. M. Barron for his encouragement and valuable advice. The author also thanks Mrs. J. Moore for her kindly helping type this nice copy.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iv</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>v</td>
</tr>
<tr>
<td>NOMENCLATURE</td>
<td>vi</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>CHAPTER I. THE CURVILINEAR COORDINATE</td>
<td>3</td>
</tr>
<tr>
<td>DESCRIPTION OF 2 - D FLOW</td>
<td></td>
</tr>
<tr>
<td>CHAPTER II. Z-GRID GENERATION</td>
<td>21</td>
</tr>
<tr>
<td>CHAPTER III. THE DESIGN AND ANALYSIS</td>
<td>30</td>
</tr>
<tr>
<td>ALGORITHM FOR CASCADE FLOW</td>
<td></td>
</tr>
<tr>
<td>CHAPTER IV. LAMINAR INCOMPRESSIBLE</td>
<td>52</td>
</tr>
<tr>
<td>FLOW IN A CHANNEL</td>
<td></td>
</tr>
<tr>
<td>CHAPTER V. CONCLUSIONS</td>
<td>71</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>75</td>
</tr>
<tr>
<td>VITA AUCTORIS</td>
<td>77</td>
</tr>
</tbody>
</table>

Typeset by AMS-TEX
NOMENCLATURE.

\( C_p : \) pressure coefficient = \( 2(P - P_1)/\rho_1q^2 \)

\( C : \) chord length, constant

\( H : \) cascade blade spacing, enthalpy

\( \rho : \) density

\( P : \) pressure

\( u, v : \) Cartesian components of velocity

\( \vec{q} : \) velocity vector

\( q : \) magnitude of velocity, speed

\( x, y : \) coordinates in Cartesian system \( y^i \)

\( x, z : \) coordinates in curvilinear system \( x^i \)

\( \xi, \eta : \) coordinates in general curvilinear system

\( \alpha : \) stagger angle measured from axial direction

\( \theta : \) flow angle measured from axial direction

\( \psi : \) streamfunction

\( \phi : \) velocity potential

\( \omega, \Omega : \) vorticity

\( e : \) total specific energy

\( \tilde{g}(i) : \) base vectors

\( \tilde{g}^{(i)} : \) reciprocal basis vectors
$A^i, Q^i :$ contravariant components of vector $\vec{A}, \vec{q}$

$\Gamma, \Omega_i :$ covariant components of vector $\vec{A}, \vec{q}$

$\nabla :$ gradient operator

$\nabla^2 :$ Laplace operator

$L :$ a differential operator

$\Gamma :$ circulation

$\delta :$ standard central difference operator

$\epsilon, \kappa :$ constants

Subscripts.

$x, y, z, \xi, \eta :$ partial differentiation

$,:$ covariant differentiation

$I :$ upstream properties

$O :$ downstream properties

$+:$ blade upper surface properties, lower boundary properties

$- :$ blade lower surface properties, upper boundary properties

$LE :$ leading edge properties

$TE :$ trailing edge properties
INTRODUCTION

Most flows of practical interest involve 'complex geometry', namely domains whose boundaries do not coincide with coordinate lines of a Cartesian or any other simple coordinate system. The most realistic method for solving the problem of flow in complex geometry is the numerical method. Of the two most commonly used numerical methodologies in computational fluid dynamics, the finite element method, owing to its intrinsic geometric flexibility, is regarded as the most natural tool for the complex geometry problem. However, numerical approaches which can take full advantage of the superior simplicity and efficiency of the finite difference method and are convenient for complex geometrical boundaries, have received a proportionately large share of the research and development effort.

On the other hand, the improvements in the accuracy and efficiency of algorithms for solving partial differential equations, as well as the accumulation of experiences in numerical modelling of the analysis problem, have stimulated interest in the inverse problem (or design problem). The inverse problem is actually to discover parts of boundaries of a complex geometry that satisfy the governing flow field equations and specified boundary conditions.

The needs for both analysis and design have made the research on the complex geometry problem a very important and fruitful area in computational fluid dynamics. However, there is much scope for improvement in the existing methods since many of them are either inefficient or difficult to use.
for design problems.

The objectives of this study are to describe a kind of curvilinear coordinate system and the transformation between it and the Cartesian system, discuss some geometric aspects of the coordinate system and the relevant flow equations and present a grid generation process that can be easily used for design as well as for analysis. The capabilities and efficiency of the proposed method will be demonstrated by application to an inviscid cascade flow problem and a viscous channel flow problem.
CHAPTER I. THE CURVILINEAR COORDINATE
DESCRIPTION OF 2 – D FLOW

1.1 A Non-orthogonal Curvilinear Coordinate Geometry

The motion of a continuous medium results from the imbalance of the density, pressure, temperature, etc. with respect to space and time. The equations which govern the motion are the principles of mechanics and thermodynamics for the conservation of mass, momentum and energy. If the properties of the medium are continuous and sufficiently differentiable in some domain of space and time, the physical quantities are field functions, and the essences of the imbalance are described by differential operators \( \frac{\partial}{\partial t}, \nabla \) and the characteristic properties of the medium, such as \( \rho, \overrightarrow{q}, e \), etc.

The curvilinear coordinate description of the motion is particularly interesting and important for problems involving curved boundaries. For two dimensional flow problems in a domain \( D \) with the geometric boundary \( \partial D \), \( \partial D \) can be considered as consisting of several single-valued curves

\[
\begin{align*}
   x = \text{constant} \\
   z(x, y) = \text{constant}
\end{align*}
\]

\[ (x, y) \in \partial D \]
and a curvilinear coordinate system \( x^i = (x, z) \) defined by

\[
\begin{align*}
  x &= x \\
  (x, y) &= (x, z(x, y)) \\
  z &= z(x, y)
\end{align*}
\]

(1.2)

\( (x, y) \in D \cup \partial D \)

can be introduced in replace of Cartesian coordinates \( y^i = (x, y) \).

In order to formulate the equations describing the flow in the domain \( D \), it is only necessary to know the metric tensors for the coordinate systems used. Since the base vectors in the coordinate system \( x^i \), written in Cartesian components, are

\[
\begin{align*}
  \tilde{g}_{(1)} &= (1, y_x) \\
  \tilde{g}_{(2)} &= (0, y_z)
\end{align*}
\]

(1.3)

the components of the metric tensor are thus

\[
\begin{align*}
  g_{11} &= 1 + y_x^2 \\
  g_{12} &= g_{21} = y_x y_z \\
  g_{22} &= y_z^2
\end{align*}
\]

(1.4)

All the essential metric properties of the coordinate system \( x^i \) are completely determined by the metric tensor \( g_{ij} \). In particular, the distance between two adjacent points, denoted by \( ds \), is related to the infinitesimal coordinate increments \( dx, dz \) through

\[
ds^2 = g_{11} dx^2 + 2g_{12} dx dz + g_{22} dz^2
\]

(1.5)

\[
= (1 + y_x^2) dx^2 + 2 y_x y_z dx dz + y_z^2 dz^2.
\]

4
and the angle $\theta$ between the coordinate lines at any point is given by

$$\cos \theta = \frac{g_{12}}{\sqrt{g_{11} g_{22}}} = \frac{y_x}{\sqrt{1 + y_x^2}}.$$  

Let $g$ denote the determinant of the matrix whose typical element is $g_{ij}$. Then $g$ is not zero for it is the square of the Jacobian $\left| \frac{\partial y^i}{\partial x^j} \right| = y_z$ of the transformation to Cartesian coordinates $y^i$. Thus, the conjugate metric tensor $g^{ij}$ can be defined

$$\begin{align*}
g^{11} &= 1 \\
g^{12} &= g^{21} = \frac{-y_x}{y_z} \\
g^{22} &= \frac{1 + y_x^2}{y_z^2}
\end{align*}$$

(1.7)

and the reciprocal basis vectors are

$$\begin{align*}
\tilde{g}^{(1)} &= g^{1p} \tilde{g}_{(p)} = (1, 0) \\
\tilde{g}^{(2)} &= g^{2p} \tilde{g}_{(p)} = \left( -\frac{y_x}{y_z}, \frac{1}{y_z} \right).
\end{align*}$$

(1.8)

Any Cartesian vector $\vec{A}$ can be represented by using either base vectors or associated reciprocal base vectors, namely $\vec{A} = A^i \tilde{g}^{(i)} = A_i \tilde{g}^i$, where $A^i, A_i$ are the contravariant and covariant components of $\vec{A}$ respectively.

It is apparent that coordinates $x^i$ are curvilinear and non-orthogonal in general. The basis vectors depend on the point under consideration, and the rate of change of the $j^{th}$ base vector with respect to $k^{th}$ coordinate is given by

$$\frac{\partial \tilde{g}^{(j)}}{\partial x^k} = [jk, i] \tilde{g}^{(i)}$$

(1.9)
or

\[
\frac{\partial \tilde{g}_{ij}}{\partial x^k} = \left\{ \begin{array}{c} m \\ jk \end{array} \right\} \tilde{g}_{(m)}
\]

where \([jk,i]\) and \(\left\{ \begin{array}{c} m \\ jk \end{array} \right\}\) are the Christoffel symbols of the first and the second kind and are determined from the metric tensor

\[
[jk,i] = \frac{1}{2} \left( \frac{\partial g_{ij}}{\partial x^k} + \frac{\partial g_{ik}}{\partial x^j} - \frac{\partial g_{ik}}{\partial x^j} \right)
\]

and

\[
\left\{ \begin{array}{c} m \\ jk \end{array} \right\} = g^{ip}[jk,p].
\]

The derivatives of \(g_{ij}\) in the \(x^i\) coordinate system have the simple form

\[
\frac{\partial g_{ij}}{\partial x^k} = \frac{\partial^2 y}{\partial x^k \partial x^i} \frac{\partial y}{\partial x^j} + \frac{\partial^2 y}{\partial x^k \partial x^j} \frac{\partial y}{\partial x^i}
\]

whence,

\[
[jk,i] = \frac{\partial^2 y}{\partial x^k \partial x^i} \frac{\partial y}{\partial x^j}.
\]

Also, since

\[
g^{ij} \frac{\partial y}{\partial x^j} = \frac{1}{g} \frac{\partial y}{\partial x^i},
\]

the Christoffel symbol of the second kind is given by

\[
\left\{ \begin{array}{c} i \\ jk \end{array} \right\} = \frac{1}{g} \frac{\partial^2 y}{\partial x^k \partial x^j} \frac{\partial y}{\partial x^i}.
\]
A direct calculation yields

\[ [11, 1] = y_x y_{xz}, \quad [11, 2] = y_z y_{xz}, \]

\[ [12, 1] = y_x y_{zz} = [21, 1], \quad [21, 2] = y_z y_{zz} = [12, 2], \]

\[ [22, 1] = y_x y_{zz}, \quad [22, 2] = y_z y_{zz}, \]

\[
\begin{cases}
2 & 11 = \frac{1}{y_z} y_{xz}, \\
2 & 12 = \frac{1}{y_z} y_{xz}, \\
2 & 22 = \frac{1}{y_z} y_{zz}
\end{cases}
\]

and all others are zeros.

Considering the variability of the base vectors, the derivatives of functions (tensors) with respect to space coordinates \( x^i \) are quantified by the covariant differentiation. This is of fundamental importance in the study of flow in a domain using a curvilinear coordinate system.

By definition, the covariant derivative of a contravariant vector \( A^i \) in the \( x^i \) coordinate system is

\[
(1.17) \quad A^i_{;j} = \frac{\partial A^i}{\partial x^j} + \left\{ \begin{array}{c} i \\
j k \end{array} \right\} A^k.
\]

In the present case, this mixed second order tensor has the following compo-
\[
\begin{align*}
A^1 &= \frac{\partial A^1}{\partial z} \\
A^1 &= \frac{\partial A^1}{\partial z} \\
A^2 &= \frac{\partial A^2}{\partial z} + \frac{1}{y_z} (y_{zz} A^1 + y_{z} A^2) \\
A^2 &= \frac{\partial A^2}{\partial z} + \frac{1}{y_z} (y_{zz} A^1 + y_{z} A^2)
\end{align*}
\]

(1.18)

The covariant derivative of a scalar function \( \phi \) is the same as the conventional partial derivative

\[
\phi, _1 = \frac{\partial \phi}{\partial x}, \quad \phi, _z = \frac{\partial \phi}{\partial z}.
\]

(1.19)

It is evident that \( \phi, _i \) are the components of the covariant vector, gradient of \( \phi \). The covariant derivative of a contravariant vector \( A^i \) with respect to \( x^i \), and then summed on \( i \), is called the divergence of the vector \( \vec{A} \), i.e.

\[
\text{div} \vec{A} = \frac{\partial A^i}{\partial x^i} + \left\{ \begin{array}{c} i \\ ik \end{array} \right\} A^k.
\]

(1.20)

Using

\[
\left\{ \begin{array}{c} i \\ ik \end{array} \right\} = \frac{1}{g^{\frac{1}{2}}} \frac{\partial g^{\frac{1}{2}}}{\partial x^k}
\]

(1.21)

we have

\[
\text{div} \vec{A} = \frac{1}{g^{\frac{1}{2}}} \frac{\partial}{\partial x^i} \left( g^{\frac{1}{2}} A^i \right)
\]

(1.22)

\[
= \frac{1}{y_z} \left( \frac{\partial}{\partial x} (y_z A^1) + \frac{\partial}{\partial z} (y_z A^2) \right).
\]
Since $\text{div} \vec{A}$ is a scalar, it is independent of the coordinate system.

The curl of the vector $\vec{A}$ is defined by

$$(1.23) \quad \text{curl} \vec{A} = \epsilon^{ijk} g^{-\frac{1}{2}} g_{kr} A_{rj} = \epsilon^{ijk} g^{-\frac{1}{2}} A_{k,j}$$

where $\epsilon^{ijk}$ is the permutation symbol. Using the formula for covariant derivative of a covariant vector, the curl of the vector $\vec{A}$ is

$$(1.24) \quad \text{curl} \vec{A} = g^{-\frac{1}{2}} \left( \frac{\partial A_2}{\partial x^1} - \frac{\partial A_1}{\partial x^2} \right),$$

since the terms involving Christoffel symbols cancel.

The Laplacian of a scalar $\phi$, defined as the divergence of the gradient of $\phi$, takes the form

$$(1.25) \quad \nabla^2 \phi = \frac{1}{g^{\frac{1}{2}}} \frac{\partial}{\partial x^i} \left( g^{\frac{1}{2}} (\nabla \phi)^i \right)$$

$$= \frac{1}{g^{\frac{1}{2}}} \frac{\partial}{\partial x^i} \left( g^{\frac{1}{2}} g^{ip} \phi, p \right)$$

$$= \frac{1}{y_z} \frac{\partial}{\partial x} \left( y_z \frac{\partial \phi}{\partial x} + y_z \left( -\frac{y_x}{y_z} \right) \frac{\partial \phi}{\partial z} \right)$$

$$+ \frac{1}{y_z} \frac{\partial}{\partial z} \left( y_z \left( -\frac{y_x}{y_z} \right) \frac{\partial \phi}{\partial x} + y_z \left( -\frac{1 + y_z^2}{y_z^2} \right) \frac{\partial \phi}{\partial z} \right)$$

$$= \frac{1}{y_z^3} \left( y_z L(\phi) - \phi_z L(y) \right)$$

where $L$ is a differential operator defined by

$$(1.26) \quad L = y_z^2 \frac{\partial^2}{\partial x^2} - 2y_z y_x \frac{\partial^2}{\partial x \partial z} + (1 + y_z^2) \frac{\partial^2}{\partial z^2}.$$
The Laplacian of a contravariant vector $A^i$ has the form

(1.27)

$$\nabla^2 \tilde{A} = A_{i,j,k} = (A_{i,j})_{,k}$$

$$= \frac{\partial A^i_{,j}}{\partial x^k} + \left\{ \begin{array}{c} i \\ k \end{array} \right\} A^m_{,j} - \left\{ \begin{array}{c} m \\ k \end{array} \right\} A^i_{,m}$$

$$= \frac{\partial}{\partial x^k} \left( \frac{\partial A^i}{\partial x^j} + \left\{ \begin{array}{c} i \\ j \end{array} \right\} A^m \right) + \left\{ \begin{array}{c} i \\ k \end{array} \right\} \left( \frac{\partial A^m}{\partial x^i} + \left\{ \begin{array}{c} m \\ i \end{array} \right\} A^l \right)$$

$$- \left\{ \begin{array}{c} m \\ k \end{array} \right\} \left( \frac{\partial A^i}{\partial x^m} + \left\{ \begin{array}{c} i \\ m \end{array} \right\} A^l \right)$$

$$= \frac{\partial^2 A^i}{\partial x^i \partial x^k} - \left\{ \begin{array}{c} m \\ k \end{array} \right\} \frac{\partial A^i}{\partial x^m} + \left\{ \begin{array}{c} i \\ k \end{array} \right\} \frac{\partial A^m}{\partial x^i} + \left\{ \begin{array}{c} i \\ j \end{array} \right\} \frac{\partial A^m}{\partial x^k}$$

$$+ A^m \frac{\partial}{\partial x^k} \left\{ \begin{array}{c} i \\ j \end{array} \right\} + \left( \left\{ \begin{array}{c} i \\ k \end{array} \right\} \left\{ \begin{array}{c} m \\ j \end{array} \right\} - \left\{ \begin{array}{c} m \\ k \end{array} \right\} \left\{ \begin{array}{c} i \\ l \end{array} \right\} \right) A^l.$$  

For $i = 1$, this gives

(1.28)

$$\nabla^2 A^1 = (A^1_{zz} + 2A^1_{xz} + A^1_{zz}) - \frac{1}{y_z} A^1_{z}(y_{zz} + 2y_{xz} + y_{zz}).$$

However, $\nabla^2 A^2$ is more lengthy,

(1.29)

$$\nabla^2 A^2 = (A^2_{zz} + 2A^2_{xz} + A^2_{zz}) - \left\{ \begin{array}{c} 2 \\ k \end{array} \right\} \frac{\partial A^2}{\partial x^m}$$

$$+ 2 \left\{ \begin{array}{c} 2 \\ k \end{array} \right\} \frac{\partial A^m}{\partial x^j} + + A^m \frac{\partial}{\partial x^k} \left\{ \begin{array}{c} 2 \\ j \end{array} \right\}$$

$$+ \left( \left\{ \begin{array}{c} 2 \\ k \end{array} \right\} \left\{ \begin{array}{c} 2 \\ j \end{array} \right\} - \left\{ \begin{array}{c} 2 \\ k \end{array} \right\} \left\{ \begin{array}{c} 2 \\ l \end{array} \right\} \right) A^l.$$  

With the basic operations so defined, the formulation of flow problems in the $\bar{x}^i$ system can be obtained in tensor form.
The tensor method, which deals with entities and properties that are independent of the reference frame, has the attractive feature that it can inherently simplify complex geometric configurations. It is possible that the tensor method may involve tedious and complex symbolic manipulations which cause difficulties in the analysis. With the development of symbolic computation in computer science, this shortcoming can be minimized and the use of the invariant tensor equations in computational fluid dynamics may prove to be a considerable convenience.
1.2 Fluid Dynamics Equations in the $x^i$ Coordinate System

Before going on to discuss special forms of the flow equations, it is worthwhile to say a few words on the mutual transform between $x^i$ and $y^i$ coordinate systems. The transformation defined in equation (1.2) can always be inverted in the neighbourhood of a point to give $y = y(x, z)$ provided that the Jacobian $y_z$ exists and does not vanish. This guarantees that no infinitesimal region in one coordinate system is collapsed into a single point in the other, because geometrically the Jacobian represents the ratio of area elements in the two coordinate systems. Moreover, if neither $y_z$ nor its reciprocal is zero, the derivatives of a function in one coordinate system can be inverted to give the derivatives in the other. Since the Cartesian base vectors are invariant, the covariant derivatives of a vector on the Cartesian base with respect to any coordinate reduce to the partial derivatives of the Cartesian components. These derivatives by definition are the components of a covariant vector. Therefore, the derivatives transform according to the equation

$$
\frac{\partial}{\partial x^i} = \frac{\partial y^i}{\partial x^j} \frac{\partial}{\partial y^j}
$$

(1.30)

The explicit form for this is the following

$$
\begin{pmatrix}
\frac{\partial}{\partial x} \\
\frac{\partial}{\partial y}
\end{pmatrix}
= \begin{pmatrix}
\frac{\partial x}{\partial x} & \frac{\partial x}{\partial y} \\
\frac{\partial y}{\partial x} & \frac{\partial y}{\partial y}
\end{pmatrix}
\begin{pmatrix}
\frac{\partial}{\partial x} \\
\frac{\partial}{\partial y}
\end{pmatrix}
= \begin{pmatrix} 1 & y_z \\
0 & y_z \end{pmatrix}
\begin{pmatrix}
\frac{\partial}{\partial x} \\
\frac{\partial}{\partial y}
\end{pmatrix}.
$$

(1.31)
The inverse transformation exists and we get

\[
(1.32) \quad \left( \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} 
\end{array} \right) = \left( \begin{array}{cc} 1 & y_x \\ 0 & y_z 
\end{array} \right)^{-1} \left( \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} 
\end{array} \right) = \left( \begin{array}{cc} 1 & -\frac{y_x}{y_z} \\ 0 & \frac{1}{y_z} 
\end{array} \right) \left( \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} 
\end{array} \right).
\]

Thus, the components of \( \nabla \) on the Cartesian basis vectors are expressed in terms of the combination of the derivatives with respect to \( x^i \) coordinates.

The dot product of \( \nabla \) with itself yields the Laplacian for a scalar

\[
(1.33) \quad \nabla^2 = \left( \begin{array}{c} \frac{\partial}{\partial x} - \frac{y_z}{y_z} \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} - \frac{y_z}{y_z} \frac{\partial}{\partial z} 
\end{array} \right) \left( \begin{array}{c} \frac{\partial}{\partial x} - \frac{y_z}{y_z} \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} - \frac{y_z}{y_z} \frac{\partial}{\partial z} 
\end{array} \right) + \left( \begin{array}{c} \frac{1}{y_z} \frac{\partial}{\partial x} \\ \frac{1}{y_z} \frac{\partial}{\partial y} 
\end{array} \right) \left( \begin{array}{c} \frac{1}{y_z} \frac{\partial}{\partial x} \\ \frac{1}{y_z} \frac{\partial}{\partial y} 
\end{array} \right)
\]

\[
= \frac{1}{y_z^3} \left[ y_z L(\quad) - L(y) \frac{\partial}{\partial z} \right]
\]

The flow equations using \( x^i \) coordinates as independent variables can be formulated using either local base vectors \( \vec{g}_{(i)} \) (or equivalent reciprocal base vectors \( \vec{g}^{(i)} \)) or Cartesian base vectors. In cases where difficulties arise from expressing vectors in terms of local base vectors \( \vec{g}_{(i)}, \vec{g}^{(i)} \), the simple expedient of expressing vectors in terms of Cartesian base vectors while conserving the equations in the \( x^i \) coordinate can remove the difficulties.

(1) Euler equations for planar flow

The equations which govern inviscid plane flow are Euler equations. The system of the four conservation equations without external forces can be written in the following form

\[
(1.34) \quad \frac{\partial f}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0
\]
where \( f, F, G \) are four component vectors

\[
(f) = \begin{pmatrix}
\rho \\
\rho u \\
\rho v \\
\rho c
\end{pmatrix} \quad (F) = \begin{pmatrix}
\rho u \\
\rho u^2 + P \\
\rho uv \\
(\rho c + P)u
\end{pmatrix} \quad (G) = \begin{pmatrix}
\rho v \\
\rho uv \\
\rho v^2 + P \\
(\rho c + P)v
\end{pmatrix}
\]

In these equations, \( \rho \) is the density, \( P \) the pressure, \( e \) the total specific energy and \( u, v \) the Cartesian components of velocity. Using the transformation formula for the partial derivatives, the system in \( x^i \) coordinates takes the form

\[
\frac{\partial \bar{f}}{\partial t} + \frac{\partial \bar{F}}{\partial x} + \frac{\partial \bar{G}}{\partial y} = 0
\]

where

\[
\bar{f} = y_x f \quad \bar{F} = y_x F \quad \bar{G} = G - y_x F.
\]

This system is of hyperbolic type with respect to time. If suitable initial and boundary conditions are assigned, the problem with given geometric constraints could be defined.

(2) Equations for steady, compressible, irrotational flow

For steady flow, \( \frac{\partial}{\partial t} = 0 \), and the conservation form of the Euler equation, as embedded in equations (1.34) and (1.35), read explicitly

\[
\begin{cases}
\nabla \cdot (\rho \vec{q}) = 0 \\
(\vec{q} \cdot \nabla)\vec{q} = \frac{1}{2} \nabla q^2 + (\nabla \times \vec{q}) \times \vec{q} = -\frac{1}{\rho} \nabla P \\
\vec{q} \cdot \nabla H_0 = 0
\end{cases}
\]
where \( H_o \) is the total specific enthalpy defined by

\[
\rho H_o = \rho c + P = \rho (H + \frac{1}{2} q^2). \tag{1.39}
\]

The last equation in (1.38) indicates that the total specific enthalpy does not change along the streamlines. From the first law of thermodynamics,

\[
T \nabla S = \nabla H - \frac{1}{\rho} \nabla P \tag{1.40}
\]

\[
= \nabla H + \frac{1}{2} \nabla q^2 + (\nabla \times \bar{q}) \times \bar{q}
\]

\[
= \nabla H_o + (\nabla \times \bar{q}) \times \bar{q}.
\]

This equation relates the entropy gradient to the vorticity in the flow field along with the gradient of the total specific enthalpy. If \( H_o \) is uniform in the whole flow field and the flow is irrotational, entropy is also uniform in the flow field. Under this circumstance, an equivalent system for (1.38) is

\[
\begin{align*}
\nabla \cdot (\rho \bar{q}) &= 0 \\
\nabla \times \bar{q} &= 0 \\
H_o &= \frac{\gamma}{\gamma - 1} \frac{P}{\rho} + \frac{1}{2} q^2
\end{align*} \tag{1.41}
\]

where \( \gamma \) designates the ratio of specific heats. This system can be simplified further by introducing streamfunction or velocity potential.

The velocity potential formulation: Using the definition of the \textit{curl}, the condition of irrotationality can be written as

\[
\frac{\partial Q_2}{\partial x} - \frac{\partial Q_1}{\partial z} = 0 \tag{1.42}
\]
where \( Q_1, Q_2 \) are the covariant components of velocity vector \( \vec{q} \). Thus, a velocity potential function \( \phi \) can be introduced as

\[
\phi_x = Q_1, \quad \phi_z = Q_2
\]

and the speed is given by

\[
q^2 = (Q_1 - \frac{y_x Q_2}{y_z})^2 + \left( \frac{1}{y_z} Q_2 \right)^2
\]

\[
= \frac{1}{y_z^2} \left( y_z^2 \phi_z^2 - 2 y_x y_z \phi_x \phi_z + (1 + y_x^2) \phi_z^2 \right).
\]

The continuity equation, according to equation (1.22), is

\[
\frac{1}{y_z} \left( \frac{\partial}{\partial x} (y_z g^{11} \rho Q_1 + y_z g^{12} \rho Q_2) + \frac{\partial}{\partial z} (y_z g^{21} \rho Q_1 + y_z g^{22} \rho Q_2) \right) = 0,
\]

i.e.

\[
(y_z \rho \phi_x - y_x \rho \phi_z)_x + \left( -y_x \rho \phi_x + \frac{1 + y_x^2}{y_z} \phi_z \right)_z = 0.
\]

letting

\[
\begin{cases}
C_1 = y_z \\
C_2 = -y_x \\
C_3 = \frac{1 + y_x^2}{y_z},
\end{cases}
\]

the conservative form of the potential equation in the \( x^i \) system is

\[
(C_1 \rho \phi_x + C_2 \rho \phi_z)_x + (C_2 \rho \phi_x + C_3 \rho \phi_z)_z = 0
\]
The streamfunction formulation: The continuity equation in the $x^i$ system, according to equation (1.22), can be written as

\begin{equation}
\frac{\partial}{\partial x}(y_z \rho Q^1) + \frac{\partial}{\partial z}(y_z \rho Q^2) = 0
\end{equation}

where $Q^1$, $Q^2$ are the contravariant components of the velocity vector $q$. A streamfunction $\psi$ can be defined such that

\begin{equation}
y_z \rho Q^1 = \psi_z, \quad y_z \rho Q^2 = -\psi_x
\end{equation}

Thus

\begin{equation}
Q^1 = \frac{1}{\rho y_z} \psi_z, \quad Q^2 = -\frac{1}{\rho y_z} \psi_x
\end{equation}

and

\begin{equation}
q^2 = (Q^1)^2 + (y_z Q^1 + y_z Q^2)^2
\end{equation}

\begin{equation}
= \frac{1}{\rho^2 y_z^2} (y_z^2 \psi_x^2 - 2 y_z y_z \psi_x \psi_z + (1 + y_z^2) \psi_z^2).
\end{equation}

The associated covariant components are

\begin{equation}
\begin{cases}
Q_1 = \frac{1 + y_z^2}{\rho y_z} \psi_z - \frac{y_z y_z}{\rho y_z} \psi_x \\
Q_2 = \frac{y_z y_z}{\rho y_z} \psi_z - \frac{y_z^2}{\rho y_z} \psi_x.
\end{cases}
\end{equation}

Substituting these into the irrotationality equation yields

\[-\left(\frac{1}{\rho} y_z \psi_z - \frac{1}{\rho} y_z \psi_z\right)_x - \left(-\frac{1}{\rho} y_z \psi_x + \frac{1 + y_z^2}{\rho y_z} \psi_z\right)_x = 0.
\]

Hence, the streamfunction equation in conservative form reads

\begin{equation}
\left(\frac{C_1}{\rho} \psi_x + \frac{C_2}{\rho} \psi_z\right)_x + \left(\frac{C_2}{\rho} \psi_z + \frac{C_3}{\rho} \psi_z\right)_x = 0
\end{equation}
The streamfunction equation or velocity potential equation are to be solved along with the Bernoulli equation. Since these systems describe flows with uniform entropy and uniform enthalpy, the use of these equations should be limited to flows without shocks or with weak shocks.

(3) Equations for steady, incompressible flow

For steady, incompressible, inviscid flow, Euler equations reduce to the simpler form

\[
\begin{align*}
\nabla \cdot \left( \vec{q} \right) &= 0 \\
(\vec{q} \cdot \nabla) \vec{q} &= \frac{1}{2} \nabla q^2 + (\nabla \times \vec{q}) \times \vec{q} = -\frac{1}{\rho} \nabla P
\end{align*}
\]

(1.54)

This system, if written in \( y' \) coordinates, is

\[
Af_x + Bf_y = 0
\]

(1.55)

where

\[
f = \begin{pmatrix}
    u \\
    v \\
    P
\end{pmatrix}
A = \begin{pmatrix}
    1 & 0 & 0 \\
    u & 0 & \frac{1}{\rho} \\
    0 & u & 0
\end{pmatrix}
B = \begin{pmatrix}
    1 & 0 & 0 \\
    v & 0 & 0 \\
    0 & v & \frac{1}{\rho}
\end{pmatrix}
\]

(1.56)

The eigenvalues are \( \lambda = \frac{v}{u} \) and \( \lambda = \pm i \), and the equations are neither hyperbolic nor elliptic. The proper supplementary conditions are very difficult to assign. It is well known that the system can be simplified to give the streamfunction-vorticity formulation

\[
\begin{align*}
\nabla^2 \psi &= \omega \\
\psi_y \omega_x - \psi_x \omega_y &= 0
\end{align*}
\]

(1.57)
Thus, the Poisson equation for streamfunction can be solved subject to boundary conditions given on the whole boundary. The vorticity equation is of hyperbolic type with space so that the solution for it exists only for the Cauchy problem. Since the solution for the Poisson equation completely determines vorticity on the boundary, the problem is ill-posed.

In fact, incompressible, inviscid flow problems can only be solved for the irrotational case. Under that circumstance, in addition to the Bernoulli equation which states the conservation of the energy, the control equation can be chosen from any of the Laplace equations for streamfunction $\psi$, velocity potential $\phi$ and Cartesian velocity components $u$ and $v$. These equations in the $x^i$ coordinate system are

\[
(1.58) \quad y_z L(\phi) - \phi_z L(y) = 0
\]

\[
(1.59) \quad y_z L(\psi) - \psi_z L(y) = 0
\]

\[
(1.60) \quad \begin{align*}
 & y_z L(u) - u_z L(y) = 0 \\
 & y_z L(v) - v_z L(y) = 0
\end{align*}
\]

where (1.58) and (1.59) can be obtained from the corresponding equations (1.47) and (1.53) for compressible flow by letting $\rho = constant$.

( 4 ) Navier-Stokes equations for incompressible flow

The equations which describe viscous flow are Navier-Stokes equations. The momentum equation for incompressible viscous flow is

\[
(1.51) \quad \rho \frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot (\mathbf{q} \mathbf{q}) + \nabla P - \mu \nabla^2 \mathbf{q} = \mathbf{f}_e
\]

where $\mu$ is the coefficient of viscosity and $\mathbf{f}_e$ is the external force.
Since the presence of the term $\nabla^2 \frac{1}{q}$ can cause difficulty, the equations for Cartesian components are used instead of ones for the components on the local base vectors. These two equations together with the continuity equation can be used to determine the velocity and the pressure. Because boundary conditions for the pressure usually do not exist, the momentum equations can serve as boundary conditions for pressure. The control equation for the pressure, a Poisson equation, is obtained by raising the order of the momentum equation by one and combining with the continuity equation. This equation written in the $y^i$ system is

$$ (1.62) \quad \nabla^2 P = 2(u_x v_y - v_x u_y) $$

Using the transform formulae, the equations in the $x^i$ system are the following

$$ \left\{ \begin{align*}
\rho u_t + u \left( u_x - \frac{y_x}{y_z} u_z \right) + v \frac{1}{y_z} u_z + \left( P_x - \frac{y_x}{y_z} P_z \right) \\
-\mu \frac{1}{y_z^2} \left( y_x L(u) - u_x L(y) \right) = 0
\end{align*} \right. $$

$$ (1.63) \quad \left\{ \begin{align*}
\rho v_t + u \left( v_x - \frac{y_x}{y_z} v_z \right) + v \frac{1}{y_z} v_z + \frac{1}{y_z} P_z \\
-\mu \frac{1}{y_z^2} \left( y_x L(v) - v_x L(y) \right) = 0
\end{align*} \right. $$

$$ \nabla^2 P = \frac{2}{y_z} (u_x v_z - v_x u_z). $$

Thus, momentum equations are used to solve for $u$, $v$ and the Poisson equation is solved subject to the momentum equations applied at each boundary point for the boundary conditions on $\frac{\partial P}{\partial x}$ and $\frac{\partial P}{\partial z}$.

A parallel streamfunction-vorticity formulation will be given in Chapter IV.
CHAPTER II. z-GRID GENERATION

The first step in a numerical technique is the discretization. In this process, the flow equations and the supplementary conditions of the continuum-model formulation of the problem are replaced by a large system of algebraic equations (i.e. difference equations). Each of these difference equations is a mathematical approximation for the physical property at a point. The discretization, depending on the way the domain is split into small elementary cells, has a decisive effect on the numerical approach of the method. This is particularly true for problems with curved boundaries. For the last 20 years, methods for dealing with the curved boundary problem have been given much attention. It is the success in this area that has greatly advanced computational fluid dynamics. Faster, simpler and more sophisticated numerical methods for complex geometry problems are still very interesting research topics from the theoretical standpoint, as well as from engineering needs.
2.1 A Review of the Grid Generation Methods

In order to keep the simple form of the finite difference formulae, the discretized grid points are taken to be located at the intersections of two families of orthogonal straight lines. At the early stage of computational fluid dynamics, this kind of straight line discretization was accomplished on the physical domain. Since curved boundaries are not conformal to straight grid lines, the numerical calculations have to be performed under various approximations for the boundary conditions. For example, (i) directly impose boundary conditions at the nearest boundary grid points; (ii) specify boundary conditions for a point by some sort of interpolation. These practices, due to the reason that the solution of a flow problem is very sensitive to the boundary conditions, have failed to give results with reasonable accuracy.

Using exact boundary conditions requires the boundary grid points to coincide with the boundary shape. This can be achieved either by using varied space steps $\Delta x$, $\Delta y$, or by adopting an irregular shaped grid cell along the boundary. As a consequence, the order of the truncation errors formally decreases [1], the rate of the convergence slows down [2], [3], and the difficulties of writing the computer code increase. To overcome these shortcomings, the transformation theory is used to construct a regular shaped domain which is the image of the domain being considered, such that a uniform orthogonal discretization in the image domain corresponds one-to-one to a discretization in the physical domain. In addition to being conformal to the boundary shape,
the grid point distribution should be such that adequate resolution is obtained, i.e., high resolution where the flow has large gradients, low resolution in regions with small flow gradients. A well ordered data base is also desired to facilitate the numerical process. The main burden of grid generation is the design of a mapping from some rectangular box in \((\xi, \eta)\) space onto the flow domain \((x, y)\) with its curved boundaries.

The existing mapping methods can be grouped into three main types:

\(1\) Geometrical method

The theoretical foundation of this method is the implicit function theorem. If two functions

\[
\begin{align*}
F(x, y, \alpha, \beta) &= 0 \\
G(x, y, \alpha, \beta) &= 0
\end{align*}
\]

are defined in the physical plane \((x, y)\), where \(\alpha, \beta\) are two independent parameters, then

\[
\begin{vmatrix}
\frac{\partial F}{\partial x} & \frac{\partial F}{\partial y} \\
\frac{\partial G}{\partial x} & \frac{\partial G}{\partial y}
\end{vmatrix} \neq 0
\]

(2.2)

confirms the existence of two families of functions

\[
\begin{align*}
x &= f(\alpha, \beta) \\
y &= g(\alpha, \beta).
\end{align*}
\]

(2.3)

The requirement of being conformal to a boundary point \((x_0, y_0)\) can be realized by choosing \(\alpha, \beta\) such that

\[
\begin{align*}
x_0 &= f(\alpha_0, \beta_0) \\
y_0 &= g(\alpha_0, \beta_0)
\end{align*}
\]

(2.4)
and a set of distinct points \((x_o, y_o), (x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\) can be obtained from (2.3) for constant \(\alpha_o\) and varied \(\beta\), (or for constant \(\beta_o\) and varied \(\alpha\)). Assume that this set of points is the images of \(N + 1\) points in the \((\xi, \eta)\) plane for \(\xi = 0\) and \(\eta = 0, 1, 2, \ldots, N\). Repeating this procedure for \(\xi = 1, 2, \ldots, M\), a pointwise mapping from \((\xi, \eta)\) to \((x, y)\) will be defined. Once the transformation quantities \(x_\xi, x_\eta, y_\xi, \) and \(y_\eta\) are evaluated, the equations of flow can be solved on the \((\xi, \eta)\) plane for \(0 \leq \xi \leq M, 0 \leq \eta \leq N\).

This method has been used to generate \(O-\) type, or \(C-\) type grids for isolated airfoil calculation. The advantage of this method is that the grid size in the physical plane can be easily controlled and adjusted. The disadvantage is that highly accurate transform quantities are difficult to obtain from simple difference calculations.

(2 ) Conformal mapping method

Conformal mapping deals with the complex function transformation. If two families of curves \(\phi(x, y) = \text{constant}\) and \(\psi(x, y) = \text{constant}\) in a complex plane \(z = x + iy\) satisfy Cauchy-Riemman conditions, then \(\phi\) and \(\psi\) are orthogonal and define a complex analytic function \(w(z) = \phi + i\psi\). Under the mapping \(z = f(\zeta)\), where \(\zeta = \xi + i\eta\) is another complex plane,

\[
(2.5) \quad x + iy = f(\xi + i\eta) = f_1(\xi, \eta) + if_2(\xi, \eta)
\]

and

\[
(2.6) \quad \phi + i\psi = w[f(\xi + i\eta)] = \Phi(\xi, \eta) + i\Psi(\xi, \eta).
\]

Equation (2.5) indicates the pointwise relation for two planes \(z\) and \(\zeta\), and equation (2.6) indicates the linewise relation. Thus the intersections of \(\phi\) and \(\psi\) in the \(z\) plane transform one-to-one to intersections of \(\Phi\) and \(\Psi\) in the
$\zeta$ plane. So, if a mapping is defined to transform a physical domain into a regularly shaped computational domain, then a uniform orthogonal mesh can be constructed in the computational domain which corresponds to an orthogonal mesh in the physical plane.

The general form of the transform is written as

$z = \zeta + \frac{c_1}{\zeta} + \frac{c_2}{\zeta^2} + \frac{c_3}{\zeta^3} + \cdots$

where $c_i$'s are complex numbers. For complex geometry, it is impossible to determine those $c_i$'s with which a regular shaped computational domain can be reached in a single mapping. The usual practice is to use a series of simple mappings to improve the shape of the boundary step by step. This leads to a considerable increase in the work to compute the transform quantities. The control of the grid size in using conformal mappings is not easy either.

( 3 ) Differential equation method

The concept to define the transformation from the physical plane to the computational plane by differential equations was proposed by Winslow [4], and Sackett and Healley [5]. This method determines the grid point positions in the physical plane by solving elliptic equations in the uniform computational mesh. Godunov and Prokopenov [6] and Thompson [7], [8], [9] extended this method to generate curvilinear non-orthogonal meshes. Walkden [10] and Moretti [11] proposed to use the transformation which maps the physical domain into streamline potential coordinates. Since the streamfunction $\psi$ and potential $\phi$ are the real and imaginary parts of a complex analytic function, $\psi$ and $\phi$ are harmonic functions and conjugate with each other. In other words, the solutions of two Laplace equations in the physical domain can form a coordinate system.
If a transformation from the \((\xi, \eta)\) plane to the \((x, y)\) plane is defined by two Laplace equations for \(\xi\) and \(\eta\) with \(x\) and \(y\) as independent variables, then the maximum and the minimum of \(\xi\) and \(\eta\) are attained only on the boundaries of the domain. By the uniqueness theorem, the transformation is one-to-one. These properties satisfy the requirements for \(\xi(x, y) = \text{constant}\) and \(\eta(x, y) = \text{constant}\) to be grid lines. Using the transformation relations

\[ y_\eta = J\xi_x, \quad x_\eta = -J\xi_y, \]

\[ y_\xi = -J\eta_x, \quad x_\xi = J\eta_y \]

\[ J = x_\xi y_\eta - x_\eta y_\xi \]

the two Laplace equations are converted into nonlinear elliptic differential equations for \(x, y\) in the \((\xi, \eta)\) plane. The grid positions \((x_{ij}, y_{ij})\) are obtained by solving this pair of equations subject to constant values of \(\xi, \eta\) at the boundary. A modification to the above procedure which defines the transformation by Poisson equations makes the control of the grid size an easy matter.

The differential equation method has been widely used owing to its simplicity in dealing with complex geometry boundary problems.
2.2 \( z \) Coordinate Generation

In Chapter I, some of the most commonly encountered flow equations are written in the \( z^i \) coordinate system. These systems of equations are not closed in the sense that introducing the \( z \) coordinate shifts \( y \) to be a new unknown. The solving of these equations is pending for the specific form of function \( z = z(x, y) \). In other words, unless the transformation between \( x^i \) and \( y^i \) is defined explicitly, these equations cannot be solved. It is well known that the analytical solution for the fluid flow equation is rare. The numerical method is essential, especially for the flow with curved boundaries. In order to solve an equation in a domain \( D \) by the finite difference method, subject to various physical constraints on the curved boundaries of \( D \), it is ideal to transform to, and carry out the calculations in a rectangular computational plane. Since the function \( z \) is independent of the flow, it can be arbitrarily chosen. This fact can be taken advantage of to construct a rectangular domain

\[
D' \cup \partial D' = \{(x, z(x, y))|x_I \leq x \leq x_O, z_+ \leq z \leq z_-\}
\]

where \( x_I, x_O, z_+, z_- \) are chosen constants.

Once the object domain is mapped into a rectangular domain, a uniform orthogonal mesh can be constructed on \( D' \cup \partial D' \). Since the \( z \) coordinate has been kept unchanged, the discretization in \( z \) direction is identical in both \( D \) and \( D' \). However, the discretization in the \( y \) direction in \( D \) depends on the mapping \( z = z(x, y) \).
Geometrical methods have long been used to generate the \( z \) coordinate lines. For the simple channel type flow, a widely used transformation takes the form

\[
(2.8) \quad z = \frac{y(x) - y_+(x)}{y_-(x) - y_+(x)}
\]

where \( y_+(x) \) and \( y_-(x) \) are the \( y \) coordinate of the lower and the upper boundaries of the physical domain. This mapping transforms the curved upper and lower boundaries into straight \( z = 0 \) and \( z = 1 \) lines.

The transformation quantities associated with this mapping are

\[
(2.9) \quad \begin{aligned}
  y_z &= y_-(x) - y_+(x) \\
  y_x &= z \frac{\partial y_-}{\partial x} + (1 - z) \frac{\partial y_+}{\partial x}.
\end{aligned}
\]

For the analysis problem where \( y_-(x) \) and \( y_+(x) \) are known functions, the transformation quantities at each grid point can be determined with no difficulty. As indicated earlier, it is not very convenient to use the geometrical method to generate grids when the multi-element problem is considered.

The construction of \( z \) coordinate lines can be accomplished by using a differential equation method. If a function \( \eta(x,y) \) defined in the physical domain is the solution of the Laplace equation, i.e.,

\[
(2.10) \quad y_z L(\eta) - \eta_x L(y) = 0,
\]

then according to the previous discussion, \( \eta(x,y) = \text{constant} \) is good for a family of grid lines. Suppose the \( z \) coordinate lines coincide with this family of grid lines. Then \( z = \eta, \eta_x = 0, \eta_z = 1 \), and equation (2.10) reduce to

\[
(2.11) \quad L(y) = 0
\]
since \( L(\eta) = 0 \).

Equation (2.11) is nonlinear, elliptic and involves only one unknown \( y \). Once the proper boundary conditions are given, the solution of equation (2.11) yields the corresponding \( y \) coordinate in the physical domain. The transformation quantities \( y_x \) and \( y_z \) can be found by numerical differencing. Hence, equation (2.11) can play the roll of defining a transformation between \( y \) and \( z \) and to generating the \( z = \text{constant} \) lines in the physical domain which accomplish the discretization in the \( y \) direction.

The grid lines generated by this procedure can be interpreted as a family of streamlines confined between \( y_+ \) and \( y_- \) if \( \eta(x,y) \) is considered as the streamfunction. This can also be justified by realizing that \( L(y) = 0 \) is nothing but \( z_{xx} + z_{yy} = 0 \), and \( y_+ \) and \( y_- \) are streamlines since \( z_+ \) and \( z_- \) are constants.

The grid generation procedure proposed here has a remarkable advantage in that it is suitable for both analysis and design problems. The difference in solving these two kinds of problems is to assign a Dirichlet boundary conditions for analysis and a Neumann boundary condition is used for design. The detailed discussion will be given in Chapter III.
CHAPTER III. THE DESIGN AND ANALYSIS ALGORITHM FOR CASCADE FLOW

A cascade is a periodic distribution of $2 - D$ blade cross sections, which serves as a basis for the fully $3 - D$ calculations required for turbomachinery flows [12], [13]. The periodicity of the flow and the restrictions on various geometric parameters result in many difficulties in cascade flow calculations. The development of an efficient and accurate cascade flow calculation procedure, which is suitable for both design and analysis problems, is undertaken in this chapter.
3.1 The Formulation in \((x, z)\)-plane

2-D steady incompressible potential flow can be described by the Laplace equation for streamfunction \(\psi\) and the Bernoulli equation for pressure, i.e.,

\begin{equation}
\nabla^2 \psi = 0
\end{equation}

\begin{equation}
\nabla (P + \frac{q^2}{2}) = 0.
\end{equation}

The physical domain \(D \cup \partial D\) can be chosen to consist of only one period with the airfoils lying on the upper and lower boundaries. The coordinate system \(y^i\) is constructed such that the \(y\)-direction is parallel to the cascade (called the “tangential direction”) and the \(z\)-direction normal to \(y\) through the cascade (called the “axial direction”). Thus the domain is irregularly-shaped. The velocity components are given in terms of the streamfunction by

\begin{equation}
u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}
\end{equation}

and the speed is

\begin{equation}
q^2 = u^2 + v^2 = \left(\frac{\partial \psi}{\partial x}\right)^2 + \left(\frac{\partial \psi}{\partial y}\right)^2.
\end{equation}

The periodicity of the flow can be characterized as

\begin{equation}
\nabla \psi(x, y + H) = \nabla \psi(x, y).
\end{equation}
where $H$ is the cascade blade spacing. Instead of studying this problem in the physical plane with irregularly-shaped geometric boundaries, it is convenient to choose the $x^i$ coordinate system with the airfoils lying on the straight $z = constant$ lines. This judicious choice of coordinate curves makes the study of the problem possible in a rectangular computational domain.

The equation for the streamfunction $\psi$ in the $x^i$ coordinate system, $y_z L(\psi) - \psi_z L(y) = 0$, consists of two symmetric parts for $\psi$ and $y$. Moreover, since $L(y) = 0$ can be used to generate a system of grids, the streamfunction equation in such a coordinate system takes the form $L(\psi) = 0$. Thus the incompressible, inviscid cascade problem can be formulated in the $x^i$ coordinate system as the following

(3.6) \hspace{1cm} L(y) = 0

(3.7) \hspace{1cm} L(\psi) = 0

(3.8) \hspace{1cm} P + \frac{q^2}{2} = constant

where

(3.9) \hspace{1cm} q^2 = \left( \frac{\partial \psi}{\partial x} - \frac{y_z}{y_z \partial z} \right)^2 + \left( \frac{1}{y_z} \frac{\partial \psi}{\partial z} \right)^2.

Now, the streamfunction equation has been split into two simpler parts. Solving these two interactive parts completes the grid generation and flow determination process. Moreover, this approach has remarkable advantages for the design problem. In this case, some parts of the boundary are to be determined to match certain given flow quantities. This can be done easily and efficiently by generating a system of grids which satisfy $L(y) = 0$ and $L(\psi) = 0$.  

32
Two sets of boundary conditions, for \( y \) and \( \psi \) in terms of \( x, z \) are needed for the purpose of solving equations (3.6) and (3.7). Both equations (3.6) and (3.7) are elliptic. Equation (3.6) is nonlinear for \( y \) and serves basically as a grid generation equation. Equation (3.7) is linear for \( \psi \). The computational domain is chosen such that

\[
D' \cup \partial D' = \{(x, z)|x_I \leq x \leq x_O, 0 \leq z \leq 1\}
\]

with the blade surfaces lying on \( z = 0 \) and \( z = 1 \). The position of the leading edge and trailing edge can be prescribed as

\[
\begin{align*}
y(x_{LE}, 0) &= 0, & y(x_{TE}, 0) &= C \cos \alpha \\
y(x_{LE}, 1) &= H, & y(x_{TE}, 1) &= H + C \cos \alpha.
\end{align*}
\]

(3.10)

The periodic condition on \( y \) is

\[
y(x, 1) = H + y(x, 0), \quad x_I < x < x_{LE} \text{ or } x_{TE} < x < x_O.
\]

(3.11)

The geometric boundary conditions at upstream and downstream can be given as

\[
\begin{align*}
y_e(x_I, z) &= \kappa_I, & 0 \leq z \leq 1 \\
y_e(x_O, z) &= \kappa_O, & 0 \leq z \leq 1
\end{align*}
\]

(3.12)

where \( \kappa_I, \kappa_O \) are constants that can be arbitrarily chosen. The boundary conditions for \( \psi \) are dependent upon the grid given by equation (3.6), the grid boundaries and the upstream velocity. If \( q_I \) is chosen to be unity, then

\[
\begin{align*}
u_I &= \cos \theta_I, & v_I &= \sin \theta_I.
\end{align*}
\]

(3.13)
where $\theta_I$ is the upstream flow angle measured from the axial direction.

The rate of flow across any $x = constant$ line is

$$
(3.14) \quad \psi(x,1) - \psi(x,0) = H \cos \theta_I, \quad x_I \leq x \leq x_O.
$$

Since the airfoils are a portion of the streamlines $z = 0$ and $z = 1$, the boundary conditions for $\psi$ on the airfoil can be taken as

$$
(3.15) \quad \begin{cases} 
\psi(x,0) = 0, & x_{LE} \leq x \leq x_{TE} \\
\psi(x,1) = H \cos \theta_I, & x_{LE} \leq x \leq x_{TE}.
\end{cases}
$$

However, on the periodic boundaries, $\psi(x,0)$ is unknown and equation (3.14) gives

$$
(3.16) \quad \psi(x,1) = H \cos \theta_I + \psi(x,0) \quad \text{for} \quad x_I < x < x_{LE}
$$

$$
\text{or} \quad x_{TE} < x < x_O.
$$

According to equations (1.32), (3.3) and (3.13)

$$
(3.17) \quad -v_I = \psi_x(x_I,z) - y_z(x_I,z)u_I
$$

or, using equation (3.12)

$$
(3.18) \quad \psi_x(x_I,z) = \kappa_I \cos \theta_I - \sin \theta_I, \quad 0 \leq z \leq 1.
$$

Similarly,

$$
(3.19) \quad \psi_z(x_O,z) = \kappa_O \cos \theta_I - \tan \theta_O \cos \theta_I, \quad 0 \leq z \leq 1,
$$

since conservation of mass flux requires that $q_O \cos \theta_O = q_I \cos \theta_I$.

The boundary conditions for $y$ on the airfoil portions are different in the cases of the design and analysis problems. The basic relation for the surface speed is

$$
(3.20) \quad q^2 = \left( \frac{1 + y_z^2}{y_z^2} \right) \psi_z^2.
$$

34
For the analysis problem, the airfoil shape is given and the solution to the flow through the cascade is required (i.e., \( q \) or \( C_p \)). Boundary conditions in this case are of Dirichlet type, i.e.,

\[
\begin{align*}
  y(x, 0) &= f_+(x), & x_{LE} &\leq x \leq x_{TE} \\
  y(x, 1) &= H + f_-(x), & x_{LE} &\leq x \leq x_{TE}
\end{align*}
\]

(3.21)

where \( f_+(x) \) and \( f_-(x) \) are airfoil ordinates under a rotation of angle \( \alpha \). After equations (3.6) and (3.7) have been solved, the pressure coefficient can be calculated using equation (3.20) and

\[
(3.22) \quad C_p = 1 - q^2.
\]

For the design problem, the airfoil shape will be generated which corresponds to specified \( C_p \) (or \( q^2 \)) distribution. In this case, boundary conditions for \( y \) are of nonlinear Neumann type, from equation (3.20). This condition is written as

\[
\begin{align*}
  y_z(x, 0) &= \sqrt{1 + y_z^2(x, 0)} \frac{\psi_z(x, 0)}{q_+}, & x_{LE} &\leq x \leq x_{TE} \\
  y_z(x, 1) &= \sqrt{1 + y_z^2(x, 1)} \frac{\psi_z(x, 1)}{q_-}, & x_{LE} &\leq x \leq x_{TE}
\end{align*}
\]

(3.23)

where \( q_+ \) and \( q_- \) are upper and lower surface speeds, respectively. All these boundary conditions are illustrated in Figs. 1 and 2.
\[ D : \quad y(x, 1) = \frac{\sqrt{1 + y_0^2(x, 1)}}{q} \psi_0(x, 1) \]

\[ y(x, 1) = H + y(x, 0) \]

\[ A : \quad y(x, 1) = H + f_-(x) \]

\[ y(x, 1) = H + y(x, 0) \]

\[ y = y(x, 1) \]

\[ y_{x_1}(x) = f(x) \]

\[ y(x, 0) = f_+(x) \]

\[ x_{lE} \quad A : \quad y(x, 0) = f_+(x) \quad x_{TE} \]

\[ D : \quad y_0(x, 0) = \frac{\sqrt{1 + y_0^2(x, 0)}}{q_0} \psi_0(x, 0) \]

**Fig. 1**  Boundary conditions for \( y \) in \((x, z)\)–plane.
\[ \psi(x, 1) = H \cos \theta_1 + \psi(x, 0) \]

\[ \psi(x, 1) = H \cos \theta_1 \]

\[ \psi(x, 1) = H \cos \theta_1 + \psi(x, 0) \]

\[ \psi(z, 2) = m \cos \theta_1 - \sin \theta_1 \]

\[ y_2 \frac{\partial^2 \psi}{\partial x^2} - 2y_2y_s \frac{\partial^2 \psi}{\partial x \partial z} + (1 + y_2^2) \frac{\partial^2 \psi}{\partial z^2} = 0 \]

Fig. 2  Boundary conditions for \( \psi \) in \((x, z)\)-plane.
3.2 Numerical Procedure

Standard finite differences are used to discretize the equations \( L(y) = 0 \) and \( L(\psi) = 0 \), namely central differencing everywhere, leading to linear algebraic equations for \( \psi \) and nonlinear algebraic equations for \( y \). \textit{SLOR} can be easily applied to solve these two sets of difference equations. In the uniform discretized grid system, a difference equation corresponding to \( L(w) = 0 \) for each interior point \((i, j)\) can be written as

\begin{equation}
-w_{i,j+1}^{(n+1)} + B_{i,j}w_{i,j}^{(n+1)} - w_{i,j-1}^{(n+1)} = D_{i,j}
\end{equation}

where \( B_{i,j} \) and \( D_{i,j} \) are determined by

\begin{equation}
B_{i,j} = 2 \left[ 1 + \left( \frac{\Delta z}{\Delta x} \right)^2 \frac{(\delta_{xy})_{i,j}^2}{1 + (\delta_{xy})_{i,j}^2} \right]
\end{equation}

\begin{equation}
D_{i,j} = (B_{i,j} - 2) \left[ \frac{1}{2} \left( w_{i+1,j}^{(n)} + w_{i-1,j}^{(n+1)} \right) (\Delta x)^2 \frac{(\delta_{xy})_{i,j} (\delta_{xx} w)_{i,j}}{(\delta_{xy})_{i,j}} \right],
\end{equation}

and

\begin{equation}
(\delta_{xx} w)_{i,j} = \frac{1}{4 \Delta x \Delta z} \left( w_{i+1,j+1}^{(n)} - w_{i+1,j-1}^{(n)} + w_{i-1,j+1}^{(n+1)} - w_{i-1,j-1}^{(n+1)} \right)
\end{equation}

\begin{equation}
(\delta_{xy})_{i,j} = \frac{1}{2 \Delta x} \left( y_{i+1,j}^{(n)} - y_{i-1,j}^{(n+1)} \right)
\end{equation}

\begin{equation}
(\delta_{xy})_{i,j} = \frac{1}{2 \Delta z} \left( y_{i,j+1}^{(n)} - y_{i,j-1}^{(n)} \right).
\end{equation}
Denoting \( z = 0 \) and \( z = 1 \) lines by \( j = 0 \) and \( j = J \) respectively, equation (3.24) should be solved for \( j = 1, 2, \cdots, J - 1 \) if \( i_{LE} < i < i_{TE} \), and boundary conditions are applied at \( j = 0 \) and \( j = J \). For \( i < i_{LE} \) or \( i > i_{TE} \), either \( j = 0 \) or \( j = J \) should be taken as an interior point. If equation (3.24) is to be solved for \( j = 1, 2, \cdots, J \), then a dummy grid line \( j = J + 1 \) can be introduced, and the associated periodic boundary conditions are

\[
(3.30) \quad w_{i,0}^{(n+1)} + G = w_{i,J}^{(n+1)}
\]

\[
(3.31) \quad w_{i,1}^{(n+1)} + G = w_{i,J+1}^{(n+1)}
\]

where \( G \) is the period of \( w \).

System (3.24), (3.30), (3.31) can be divided into two subsystems

\[
(3.32) \quad \begin{cases} 
-w_{i,j+1}^{(n+1)} + B_{i,j} \tilde{w}_{i,j}^{(n+1)} - \tilde{w}_{i,j-1}^{(n+1)} = D_{i,j} & (j = 1, 2, \cdots, J) \\
\tilde{w}_{i,0}^{(n+1)} = 0 \\
\tilde{w}_{i,J}^{(n+1)} = G 
\end{cases}
\]

\[
(3.33) \quad \begin{cases} 
-w_{i,j+1}^{(n+1)} + B_{i,j} \tilde{w}_{i,j}^{(n+1)} - \tilde{w}_{i,j-1}^{(n+1)} = 0 & (j = 1, 2, \cdots, J) \\
\tilde{w}_{i,0}^{(n+1)} = 1 \\
\tilde{w}_{i,J}^{(n+1)} = 1 
\end{cases}
\]

such that

\[
(3.34) \quad w_{i,j}^{(n+1)} = \tilde{w}_{i,j}^{(n+1)} + \kappa \tilde{w}_{i,j}^{(n+1)} \quad (j = 0, 1, \cdots, J)
\]
satisfies equations (3.24) and (3.30) automatically for any constant $\kappa$. By requiring $\kappa$ to be consistent with equation (3.31), which gives

\[
\kappa = \frac{\bar{w}_{i,J+1}^{(n+1)} - \bar{w}_{i,1}^{(n+1)} - G}{\bar{w}_{i,1}^{(n+1)} - \bar{w}_{i,J+1}^{(n+1)}},
\]

the solutions are obtained from equation (3.34) for $j = 0, 1, \ldots, J + 1$.

The circulation around a cascade blade is given by

\[
\Gamma = H(\sin \theta_I - \tan \theta_O \cos \theta_I).
\]

Once the blade spacing and the upstream velocity is given, $\Gamma$ is only a function of outlet flow angle. This simple property is used in this work to enforce the Kutta condition through a simple successive correction relation

\[
(\tan \theta_O)^{(n+1)} = (\tan \theta_O)^{(n)} + (q_{iTE,J}^{(n)} - q_{iTE,0}^{(n)}).
\]

A flowchart depicting the complete sequence of calculations is shown in Fig. 3. For the analysis problem ($M = 1$), the computational procedure contains two steps: first, generating the grid by solving $L(y) = 0$; second, solving $L(\psi) = 0$ on the generated grid and adjusting the outlet flow angle in order that the Kutta condition is satisfied. For the design problem ($M = 2$), the above two steps are performed repeatedly until the correction to the airfoil shape in two successive cycles becomes sufficiently small.
Fig. 3 Flowchart

1. Start
2. Input data, specify B.C.s
3. If $|y^{(n+1)} - y^{(n)}| < \varepsilon_1$, then $L(y) = 0$; otherwise, go to 4.
4. Specify new $y_s$ on blade surface
5. If $|y_s^{(n+1)} - y_s^{(n)}| < \varepsilon_3$, then $y_s = \sqrt{\frac{1 + v_s^2}{1 - C_p v_s^2}} \psi_s$; otherwise, go to 6.
6. If $|q_+(x_{TB}) - q_-(x_{TB})| < \varepsilon_4$, then $C_p = 1 - \frac{1 + v_s^2}{v_s^2} \psi_s^2$; otherwise, go to 7.
7. Adjust outlet flow angle $\theta_0$
8. Output data
9. Stop
3.3 Sample Calculations

In order to check the reliability of determining the flow field or designing the airfoil geometry using the above described algorithm, a cascade and an isolated airfoil for which analytical results exist have been calculated and the predicted airfoil surface properties are presented and compared with analytical solutions.

The calculations are performed on the uniformly discretized mesh with \( \kappa_I, \kappa_O \) taken to be zero. The convergence criteria are that the maximum changes in \( y \) and \( \psi \), from one iterative step to the next, both be less than \( \epsilon_1 = \epsilon_2 = 5.0 \times 10^{-4} \). However, in the design case, the tolerable correction for \( y_s \) on the blade surface is chosen to be \( \epsilon_3 = 5 \times 10^{-3} \). This also serves as the minimum correction \( \epsilon_4 \) for the circulation \( \Gamma \).

The first test example of the present method is the computation of a Gostelow cascade, carried out on an 81 \( \times \) 21 uniform mesh with 21 grid points on the blade surface. Analytical results for the Gostelow airfoil geometry and pressure distribution have previously been obtained through the application of Merchant and Collar's transformation to a set of ovals [14]. This cascade has a spacing to chord ratio \( H/C = 0.9901573 \), and a stagger angle of 37.5 degrees. Flow inlet angle is 53.5 degrees, and the corelated outlet flow angle is 30.025 degrees (\( \tan \theta_C = 0.57793012 \)). These reported data, together with the exact \( C_p \), are used to design the blade geometry. The predicted blade surface ordinates compare very well with the exact values, except for a few
points near the leading edge on the upper surface. The designed Gostelow airfoil ordinates are tabulated in Table 1 and comparison of designed with exact values is shown in Fig. 4.
Table 1  Designed coordinates for the Gostelow cascade

<table>
<thead>
<tr>
<th>x</th>
<th>y−</th>
<th>y+</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000000</td>
<td>0.018134</td>
<td>−0.008990</td>
</tr>
<tr>
<td>0.039700</td>
<td>0.098911</td>
<td>−0.002107</td>
</tr>
<tr>
<td>0.079300</td>
<td>0.155251</td>
<td>0.027645</td>
</tr>
<tr>
<td>0.119000</td>
<td>0.202396</td>
<td>0.064579</td>
</tr>
<tr>
<td>0.158700</td>
<td>0.243859</td>
<td>0.104498</td>
</tr>
<tr>
<td>0.198300</td>
<td>0.281155</td>
<td>0.145406</td>
</tr>
<tr>
<td>0.238000</td>
<td>0.315021</td>
<td>0.186316</td>
</tr>
<tr>
<td>0.277700</td>
<td>0.346171</td>
<td>0.226531</td>
</tr>
<tr>
<td>0.317300</td>
<td>0.374925</td>
<td>0.265757</td>
</tr>
<tr>
<td>0.357000</td>
<td>0.401713</td>
<td>0.303504</td>
</tr>
<tr>
<td>0.396700</td>
<td>0.426654</td>
<td>0.339976</td>
</tr>
<tr>
<td>0.436300</td>
<td>0.449968</td>
<td>0.374880</td>
</tr>
<tr>
<td>0.476000</td>
<td>0.471881</td>
<td>0.408024</td>
</tr>
<tr>
<td>0.515700</td>
<td>0.492415</td>
<td>0.439514</td>
</tr>
<tr>
<td>0.555300</td>
<td>0.511690</td>
<td>0.469357</td>
</tr>
<tr>
<td>0.595000</td>
<td>0.529931</td>
<td>0.497359</td>
</tr>
<tr>
<td>0.634700</td>
<td>0.547060</td>
<td>0.523528</td>
</tr>
<tr>
<td>0.674400</td>
<td>0.563299</td>
<td>0.547870</td>
</tr>
<tr>
<td>0.714000</td>
<td>0.578870</td>
<td>0.570391</td>
</tr>
<tr>
<td>0.753700</td>
<td>0.593796</td>
<td>0.590799</td>
</tr>
<tr>
<td>0.793400</td>
<td>0.608800</td>
<td>0.608800</td>
</tr>
</tbody>
</table>
Fig. 4  Designed Gostelow Cascade

\[ \alpha = 37.5^\circ \quad \theta_I = 53.5^\circ \quad \theta_O = 30.025^\circ \]
Fig. 5  Pressure coefficient for the Gostelow cascade

\[ \alpha = 37.5^\circ \quad \theta_I = 53.5^\circ \quad \theta_O = 30.025^\circ \]
Pressure distribution and outlet flow angle are chosen as the basis for comparison in the analysis problem. The computed pressure distribution, for specified inlet and outlet flow angle reported by Gostelow, is shown in Fig. 5. This result is in close agreement with the exact values, and the trailing edge pressure coefficient difference between upper and lower surface is minor. When the outlet flow angle is unspecified and the Kutta condition is employed, the derived outlet flow angle is 28.47 degree (\(\tan \theta_O = 0.54230\)). However, the pressure coefficient just slightly changes near the trailing edge with agreement still quite good.

The second application of the present method is to compute an isolated airfoil as a cascade with large blade spacing and zero stagger angle. The flow over a Karman-Trefftz airfoil is calculated on a 81 \(\times\) 31 uniform mesh with 21 grid points on the airfoil surface. Blade spacing is chosen to be 10 times the chord length. The data used in the computation are taken from [14]. The flow is assumed to be at zero incidence \(\theta_I = 0\). The Kutta condition is used in both the design and analysis computation. The derived outlet flow angles are 1.53 degrees (\(\tan \theta_O = 0.02675\)) in design and 1.98 degrees (\(\tan \theta_O = 0.03454\)) in analysis. Computed \(C_p\) and designed airfoil are shown in Figs. 6 and 7. The comparison between computed and exact values shows good agreement.
Fig. 6  Pressure coefficient for the Karman-Trefftz airfoil
Fig. 7  Designed Karman-Trefftz airfoil

- EXACT
0 0 0 - PRESENT

---

49
3.4 A Comparison of the Methods

The classical conformal transformation method of Merchant and Collar [14] and Gostelow [15], [16] has long provided test cases with which numerical solutions for cascade analysis may be compared. The method has difficulties in dealing with arbitrary airfoils and it cannot be extended to compressible flow calculations. More sophisticated singularity methods have been successfully used for both incompressible and compressible calculations, however, the design of airfoils using the singularity method is typically a time consuming iterative analysis process, e.g., Wilkinson [17], Korn [18] and Bauer, et al. [19]. Many transformation methods that map the blade-to-blade plane into a rectangular domain are available for cascade calculation. These methods, e.g., [20] [21], need an extra mapping step and their geometric flexibility is limited. Stanitz [22] provided a design model which performs calculation on the $(\phi, \psi)$-plane, where $\phi$ is velocity potential and $\psi$ is streamfunction. The periodic boundary conditions in the cascade problem cannot be easily applied in this formulation. The von Mises transformation leads to a very simple calculation procedure, see Barron [23], Dulikravich [24]. However, the assumption which must be made about the location of the forward stagnation point and the breakdown for the case where local reversed flow occurs are known to obstruct the use of the method.

This transformation method differs from the von Mises method, on the basis that it does not require the coordinate lines to be streamlines except
on the solid wall boundaries. This makes the present transformation more applicable and versatile.

The present method is based on the representation of the Laplace operator in a type of curvilinear coordinate system. The dynamical quantity $\psi$ and the geometrical quantity $y$ are related through a partial differential equation which governs the flow. Grid generation and determination of the flow field can be accomplished by solving two separated parts of this equation. For the cascade problem, the periodic boundary conditions for $y$ and $\psi$ are easy to apply in this formulation. The difference equations, which have periodic solutions with known period, can be solved efficiently by a modified tridiagonal solver. Applied to test cases, the present method yields very good results for both design and analysis problems for incompressible flow.

Since the Laplace operator for a scalar under the $(z, y) \rightarrow (x, z)$ transformation always contains the operator $L$ and the expression $L(y)$, the grid generation process which uses equation (3.6) leads to a simplification of any original equation which involves the Laplacian. So, the present grid generation equation can be applied without modification to more complicated $2-D$ flows, such as compressible flow, viscous flow and multi-element flow, etc.
CHAPTER IV. LAMINAR INCOMPRESSIBLE FLOW IN A CHANNEL

4.1 A Test Problem

In 1982, the International Association for Hydraulic Research (IAHR) Working Group on Refined Modelling of Flows, devoted the Fifth IAHR Meeting to a specific subject — to assess the capabilities of various numerical methods to deal with laminar flows in complex geometry. A single well defined test problem, namely the laminar flow through a smooth expansion channel proposed by Roache [25], was chosen for testing various numerical methods. More than fifteen groups of CFD specialists participated in the investigation of the test case. The comparison and discussion of the solution along with the description of the methods used by participants are reported by Napolitano and Orlandi [26].
The domain of interest is enclosed by 4 pieces of curves, i.e.,
\[
\begin{align*}
y_+ &= \frac{1}{2} \left( \tanh \left( 2 - \frac{30x}{Re} \right) - \tanh(2) \right) \\
y_- &= 1 \\
x_I &= 0 \\
x_O &= \frac{Re}{3}
\end{align*}
\]
(4.1)
where \( y_- \) is a symmetry plane, \( y_+ \) a solid wall and \( x_I, x_O \) the inlet and outlet respectively. The channel shape depends on the Reynolds number \( Re \); the channel becomes longer and straighter as \( Re \) increases and the solution exhibits a scaling behaviour for \( Re >> 1 \) [25]. (The solutions at the same finite difference grid points become identical for \( Re >> 1 \)).

The inflow boundary conditions are prescribed as fully developed Poiseuille flow and its Cartesian velocity components are given by
\[
\begin{align*}
u &= 3(y - y^2/3) \\
\text{on} \quad x &= x_I \\
v &= 0
\end{align*}
\]
(4.2)
The outflow boundary conditions are left for arbitrary choice by the specific investigator.

The wall vorticity and pressure at 21 equally spaced \( \frac{x}{x_O} \) locations were chosen as the unknowns for test cases \( Re = 10 \) and \( Re = 100 \). The accuracies of the solutions are judged by the average percentage error \( \epsilon_N \) defined by
\[
\epsilon_N = \frac{100}{19} \sum_{i=1}^{19} \left| \frac{\Omega_i - \Omega_i^C}{\Omega_i^C} \right|
\]
(4.3)
where \( \Omega_i \) are the vorticity at the aforementioned equally spaced points along the wall, and \( \Omega_{iC} \) refers to the benchmark solution by Cliffe et al. [27]. The wall vorticity values at the inlet and outlet were excluded from the definition of \( \varepsilon_\Omega \) to reduce the influence of the singularity at \( x = x_I, y = y_+ \) and of the differently chosen outlet boundary conditions.

It was shown that many solutions are characterized by large values of \( \varepsilon_\Omega \), in spite of the presumed simplicity of the flow cases to be computed.

With reference to the discretization methods of the participant groups who used the finite difference method, it can be seen that:

(1) Wada and Adachi used the Cartesian coordinate system to discretize the physical domain and to solve for the primary variables. The wall boundary conditions are imposed by interpolation. The solution reveals very large errors for vorticity with \( \varepsilon_\Omega = 135.48 \) for \( Re = 10 \) and \( \varepsilon_\Omega = 111.92 \) for \( Re = 100 \). This corroborates the conclusion that Cartesian discretization in physical domain is inadequate to compute the complex geometry problem.

(2) Alfrink used curvilinear finite differences [28] in the physical space to solve for primary variables. The CPU time required by this method is significantly long due to its complicated representation for derivatives.

(3) Two groups (Demirovic and Gosman, Latrobe and Delapierre) used the finite volume method. Since the finite volume method is based on the integral form on the element cell, the conservation property is preserved and the complicated domain can be arbitrarily discretized. However, the results by both groups are not satisfactory.

(4) Guj and Favini generated a curvilinear mesh by solving a quasi linear elliptic system and used an extension of the marker-and-cell (MAC) scheme. It is well known that the MAC method is well adapted for the pressure but
presents disadvantages concerning the velocity. This may accounts for the poor accuracy for the wall vorticity.

(5) Transformation methods were used by many groups. Various grid generation techniques such as the algebraic method, conformal mapping method and differential equation method were employed. For this comparatively simple geometry, the methods of transformation and grid generation do not appear to have much effect on the solution.
4.2 Flow Equations and Boundary Conditions

Channel flow can be easily formulated in the $x^i$ coordinate system. For an incompressible viscous flow, the dilatation field is everywhere zero and only the vorticity needs to be defined. Since vorticity inside the flow region is conserved while the solid wall can generate the vorticity, using the exact boundary condition is obviously of great importance.

According to the continuity equation, contravariant components of velocity $\vec{q}$ are defined, i.e.

$$ Q^1 = \frac{1}{y_z} \psi_z, \quad Q^2 = -\frac{1}{y_z} \psi_x $$

and consequently, covariant components are

$$ Q_1 = \left( \frac{1 + y_x^2}{y_z} \psi_z - y_x \psi_x \right) $$

$$ Q_2 = (-y_z \psi_x + y_x \psi_z). $$

Thus, vorticity is

$$ \Omega = \frac{1}{y_z} \left( (-y_z \psi_x + y_x \psi_z)_x - \left( \frac{1 + y_x^2}{y_z} - y_z \psi_x \right)_z \right). $$

This can also be written as

$$ \Omega = -\frac{1}{y_z} \left( y_z L(\psi) - \psi_z L(y) \right). $$
The vorticity transport equation describes the vorticity convection and diffusion in the flow field. Written in nondimensional form, the vorticity transport equation reads

\[
\frac{\partial \Omega}{\partial t} + \nabla \cdot (\vec{q} \Omega) = \frac{1}{Re} \nabla^2 \Omega.
\]  

This equation is obtained by taking the curl of the Navier-Stokes equation.

Noticing that the convection term

\[
\nabla \cdot (\vec{q} \Omega) = \frac{1}{y} \frac{\partial}{\partial x^i} \left( y^{\frac{1}{2}} Q^i \Omega \right)
\]

\[
= \frac{1}{y_z} \left( \frac{\partial}{\partial x} (\psi_z \Omega) + \frac{\partial}{\partial z} (-\psi_x \Omega) \right)
\]

\[
= \frac{1}{y_z} (\psi_z \Omega_z - \psi_x \Omega_x)
\]

the explicit form of the vorticity equation in \( x^i \) system is

\[
\frac{\partial \Omega}{\partial t} + \frac{1}{y_z} (\psi_z \Omega_z - \psi_x \Omega_x) - \frac{1}{y_z^2 Re} (y_z L(\Omega) - \Omega_z L(y)) = 0.
\]

If \( z \) coordinates are used, then the problem is governed by the following system

\[
\begin{aligned}
L(y) &= 0 \\
L(\psi) &= -y_z^2 \Omega \\
\frac{\partial \Omega}{\partial t} + \frac{1}{y_z} (\psi_z \Omega_z - \psi_x \Omega_x) - \frac{1}{y_z^2 Re} L(\Omega) &= 0
\end{aligned}
\]  

The boundary conditions for streamfunction and for vorticity are connected.

In the present study, the boundary conditions are systematized as follow: Any
specification for velocity must be considered as a condition for streamfunction, while conditions for vorticity are deduced directly from the definition (4.7).

Following this rule, the streamfunction can be decided first. On the solid wall \( y_+ \), the standard no-slip, no-injection conditions are imposed. Since \( \vec{q} = 0 \), i.e. \( Q^1 = Q^2 = 0 \), then \( \psi_x = \psi_z = 0 \), thus the solid wall is a streamline with \( \psi_+ = constant \).

On the symmetry line \( y_- \), \( Q_2 = 0 \). Taking into account that \( y_z = 0 \) on \( y_- \), the boundary condition for streamfunction is \( \psi_x = 0 \), namely \( y_- \) is also a streamline, \( \psi_- = constant \). The value of streamfunction at the solid wall and the symmetry line must be determined in accordance with the inlet boundary condition.

The outlet flow condition chosen herein is \( Q_2 = 0 \), so that the outlet flow is also fully developed Poiseuille flow. This yields a relation

\[(4.12) \quad y_z \psi_x = y_x \psi_z \quad \text{on} \quad x = x_0.\]

The inlet flow is similar to the outlet. In addition to \( Q_2 = 0 \), at the inlet

\[(4.13) \quad Q_1 = u \]

\[= -y_z \psi_x + \frac{1 + y_z^2}{y_z} \psi_x = \frac{\psi_x}{y_z} \]

where equation (4.12) has been used to eliminate \( \psi_x \) in \( Q_1 \). Whence, by integrating \( \psi_x = y_z u \) from \( z = 0 \) to \( z = 1 \), the streamfunction at the inlet is

\[(4.14) \quad \psi = \frac{3}{2} y^2 - \frac{1}{2} y^3 \quad \text{on} \quad x = x_1.\]

The constant values for streamfunction at \( y_+ \) and \( y_- \) are thus determined, \( \psi_- = 1 \) and \( \psi_+ = 0 \). The numerical integration for streamfunction at \( x = x_0 \) should also use \( \psi_- = 1 \) and \( \psi_+ = 0 \) as the boundary condition.
The vorticity at the symmetry line is obviously zero since \( Q_2 \) is identically zero and \( Q_1 \) does not change with \( z \).

In contrast to the symmetry line, \( \frac{\partial Q_1}{\partial z} \neq 0 \) on the solid wall. Thus

\[
\Omega = \frac{1 + \frac{y^2_z}{y^2}}{y^2} \psi_{zz}
\]

since \( Q^1 \) and \( Q^2 \) are both zeros.

Noticing that

\[
\psi(x, 0) = 0, \quad \frac{\partial \psi}{\partial z}(x, 0) = 0,
\]

the Taylor expansion for \( \psi(x, \Delta z) \) about \( z = 0 \) is,

\[
\psi(x, \Delta z) = \frac{1}{2} \Delta z^2 \frac{\partial^2 \psi}{\partial x^2}(x, 0) + \frac{1}{6} \Delta z^3 \frac{\partial^3 \psi}{\partial x^3}(x, 0) + \cdots
\]

Thus, a first order approximation for \( \psi_{zz} \) is

\[
\psi_{zz} = \frac{2\psi(x, \Delta z)}{\Delta z^2}.
\]

Substituting this into (4.15) yields

\[
\Omega(x, 0) = \left(1 + \frac{y^2_z}{y^2}\right) \frac{2\psi(x, \Delta z)}{\Delta z^2}.
\]

This formula for wall vorticity is of first order accuracy.

A second order approximation for \( \psi_{zz} \) can be obtained by eliminating the third derivative terms in the Taylor expansion for \( \psi(x, \Delta z) \) and \( \psi(x, 2\Delta z) \) about \( z = 0 \). Since

\[
\psi(x, 2\Delta z) = 2\Delta z^2 \frac{\partial^2 \psi}{\partial x^2}(x, 0) + \frac{4}{3} \Delta z^3 \frac{\partial^3 \psi}{\partial x^3}(x, 0) + \cdots
\]

a combination of (4.17) and (4.20) yields

\[
\psi_{zz} = \frac{8\psi(x, \Delta z) - \psi(x, 2\Delta z)}{2\Delta z^2}.
\]
Thus, the wall vorticity can be calculated by

\[(4.22) \quad \Omega(x, 0) = \left(1 + \frac{y^2}{y_z^2}\right) \frac{8\psi(x, \Delta z) - \psi(x, 2\Delta z)}{2\Delta z^2} (x, 0)\]

Equation (4.22) is of second order accuracy. Inlet vorticity, by definition, is

\[(4.23) \quad \Omega(0, z) = \left(\frac{1}{y_z} \frac{\partial Q_2}{\partial x} - \frac{1}{y_z} \frac{\partial Q_1}{\partial z}\right)_{(0, z)}\]

\[= \left(\frac{1}{y_z} \frac{\partial Q_2}{\partial x} + \frac{y_z}{y_z^2} \frac{\partial Q_2}{\partial z} - \frac{1}{y_z} \frac{\partial Q_1}{\partial z}\right)_{(0, z)}\]

Since \(Q_2\) is zero,

\[(4.24) \quad \Omega(0, z) = -\left(\frac{\partial^2 \psi}{\partial y^2}\right)_{(0, z)} - \frac{1}{\Delta z} \left(\psi_x - \frac{y_z}{y_z^2} \psi_z\right)_{(\Delta x, z)}\]

i.e.,

\[(4.25) \quad \Omega(0, z) = -3(1 - y) + \frac{v(\Delta x, y)}{\Delta x}\]

Of course, higher order approximations for \(\frac{\partial Q_2}{\partial x}\) are also possible.

This discussion shows that, at the inlet, the commonly used condition \(\Omega = \frac{\partial u}{\partial y}\) is insufficient. The change of the vertical velocity component also contributes to the inflow vorticity.

The outlet vorticity can be discussed in the same manner, which gives

\[(4.26) \quad \Omega(x_f, z) = -\left(1 + \frac{y^2}{y_z^2} \psi_z\right)_{(x_f, z)} + \frac{1}{\Delta z} \left(\psi_x - \frac{y_z}{y_z^2} \psi_z\right)_{(x_f - \Delta x, z)}\]

An alternate determination of outlet vorticity is based on physical consideration. Since change in the height over the last 20% of channel length is of the order \(10^{-5}\), and \(Q_2\) is zero at the outlet, the vorticity convection is virtually zero, and can be neglected. That is to say,

\[(4.27) \quad \psi_x \Omega_z = \psi_z \Omega_x\]

which can also be written as

\[(4.28) \quad y_z \Omega_z = y_z \Omega_x\]
Channel geometry and grids for $Re = 10$

Fig. 8
4.3 Method of Solution

The numerical procedure used herein is: (1) to generate the grid system by solving \( L(y) = 0 \); (2) to initialize the flow field and to specify boundary conditions; (3) to solve \( L(\psi) = -y_z^2 \Omega \) subject to known vorticity, yielding the streamfunction field; (4) to advance the vorticity transport equation for one time step for the new vorticity field; (5) to update the vorticity boundary values at the inlet, the outlet and the solid wall; (6) to repeat (3), (4), (5) if the steady state is not reached.

The method for grid generation and for solving the streamfunction equation is essentially the same as that described in Chapter III. In this problem, a \( 21 \times 21 \) uniformly spaced grid system in the \( x \) coordinate system corresponding to the curvilinear discretization in the \( y \) system for which the grid points at the inlet and outlet are uniformly spaced, i.e.,

\[(4.29) \quad \Delta y = \frac{1}{20} (y_- - y_+).\]

The grid point is denoted by \( x_i = i \Delta x, z_j = j \Delta z \) for \( i, j = 0, 1, \cdots, 20 \). The geometry and the grid system generated are depicted in Fig. 8.

The system of difference equations for the streamfunction on each \( i \) line has constant boundary values \( \psi_{i,0} = 0, \psi_{i,20} = 1 \). \textit{SLOR} can be used to solve the system for all the inner points by sweeping from \( i = 1 \) to \( i = 19 \). The outlet streamfunction values are updated using (4.12). If central differencing is applied to \( \psi \) and \( y_z \) and one-sided three-point differencing is used for \( \psi_x \).
and $y_x$, the equation for the outlet streamfunction correction is

\begin{equation}
\psi_{20,j} = \frac{1}{3}(4\psi_{19,j} - \psi_{18,j} + k_j(\psi_{20,j+1} - \psi_{20,j-1}))
\end{equation}

where

\begin{equation}
k_j = \frac{3y_{20,j} - 4y_{19,j} + y_{18,j}}{y_{20,j+1} - y_{20,j-1}}
\end{equation}

and $j = 1, 2, \ldots , 19$.

Various techniques can be used to solve the vorticity transport equation. A Split Implicit Method is modified for the present problem. The split vorticity equations are written as

\begin{equation}
\frac{1}{2} \frac{\partial \Omega}{\partial t} = -\frac{1}{y_z} \psi_x \Omega_x \\
+ \frac{1}{Re} \left( \frac{y_{zz} \Omega_{x}}{y_z} - \frac{y_x \Omega_{xz}}{y_z} \right)
\end{equation}

\begin{equation}
\frac{1}{2} \frac{\partial \Omega}{\partial t} = \frac{1}{y_z} \psi_z \Omega_z \\
+ \frac{1}{Re} \left( \frac{1 + y_z^2 \Omega_{zz} z}{y_z^2} - \frac{y_x \Omega_{xz}}{y_z} \right)
\end{equation}

This splitting has significant physical meaning. In (4.32), the first term on the right hand side represents the vorticity convection along the vertical coordinate lines $x = \text{constant}$ and the second term represents the vorticity diffusion in the direction normal to the $x = \text{constant}$ lines. In (4.33), the first right hand side term represents vorticity convection along $z = \text{constant}$ coordinate lines and the second term represents the diffusion normal to it. Diffusion terms in these equations consist of cross derivative terms which represent the effect of the curvilinear non-orthogonal coordinate description.
These two equations are of parabolic type. The steady state solution is obtained for sufficiently small time derivative \( \frac{\partial \Omega}{\partial t} \). At each time level, the coupled system is solved by first marching a half time step \( \frac{1}{2} \Delta t \) for equation (4.32) and then marching another half time step for (4.33) to reach the next time level.

The convective terms are treated explicitly for simplicity. The diffusion terms are treated partially implicitly, namely only implicitly expressing the second order derivative terms \( \Omega_{xx} \) and \( \Omega_{zz} \), while explicitly expressing the cross derivative terms \( \Omega_{xz} \). In this way, the tridiagonal nature of the matrix is preserved, and **SLOR** can be applied to sweep (4.32) for \( j = 1, 2, \cdots, 19 \) and to sweep (4.33) for \( i = 1, 2, \cdots, 19 \).

The complete numerical procedure for vorticity equation is

\[
\begin{align*}
\frac{\Omega_{i,j}^{n+\frac{1}{2}} - \Omega_{i,j}^{n}}{\Delta t} &= - \left( \frac{\psi_z}{y_z} \right)_{i,j}^{n} \frac{\Omega_{i+1,j}^{n} - \Omega_{i-1,j}^{n}}{2\Delta x} \\
&\quad + \frac{1}{Re} \frac{\Omega_{i+1,j}^{n+\frac{1}{2}} - 2\Omega_{i,j}^{n+\frac{1}{2}} + \Omega_{i-1,j}^{n+\frac{1}{2}}}{\Delta x^2} \\
&\quad - \frac{1}{Re} \left( \frac{y_z}{y_z} \right)_{i,j}^{n} \frac{\Omega_{i+1,j+1}^{n} - \Omega_{i-1,j+1}^{n} - \Omega_{i+1,j-1}^{n+\frac{1}{2}} - \Omega_{i-1,j-1}^{n+\frac{1}{2}}}{4\Delta x \Delta z}
\end{align*}
\]
\[
\frac{\Omega_{i,j}^{n+1} - \Omega_{i,j}^{n+\frac{1}{2}}}{\Delta t} = \left(\psi_z\right)_{i,j} \frac{\Omega_{i,j+1}^{n+\frac{1}{2}} - \Omega_{i,j-1}^{n+\frac{1}{2}}}{2\Delta z} + \frac{1}{Re} \left(1 + \frac{y_z^2}{y_z^2}\right)_{i,j} \frac{\Omega_{i,j+1}^{n+1} - 2\Omega_{i,j}^{n+1} + \Omega_{i,i-1}^{n+1}}{\Delta z^2} - \frac{1}{Re} \left(\frac{y_z}{y_z}\right)_{i,j} \frac{\Omega_{i+1,j+1}^{n+\frac{1}{2}} - \Omega_{i-1,j+1}^{n+\frac{1}{2}} - \Omega_{i+1,j-1}^{n+\frac{1}{2}} + \Omega_{i-1,j-1}^{n+\frac{1}{2}}}{4\Delta x \Delta z}.
\]

The boundary conditions for vorticity are updated by applying (4.24) at the inlet for \(i = 0, j = 1, 2, \cdots, 19\); and (4.22) (or (4.19)) at the wall for \(i = 1, 2, \cdots, 19, j = 0\).

The outlet vorticity is calculated by using (4.28), which leads to an expression for \(\Omega_{20,j}\) with \(j = 1, 2, \cdots, 19\). Equation (4.23) is also applied at the corner grid \(i = 20, j = 0\), where all the derivatives are approximated by three-point, one-sided differences. Thus, the formula for calculating the outlet wall vorticity is

\[
\Omega_{20,0} = \frac{1}{3 \left(1 + \frac{\Delta x y_z}{\Delta z y_z}\right)} \left((4\Omega_{19,0} - \Omega_{18,0}) + \frac{\Delta x y_z}{\Delta z y_z} (4\Omega_{20,1} - \Omega_{20,2})\right) .
\]
4.4 Results and Discussion

Results of the calculation procedure described in section 4.3 are compared with the benchmark solution by Cliffe et al. [27] in Table 2 for flow at \( Re = 10 \) and in Table 3 for flow at \( Re = 100 \). The convergence criteria used is

\[
R_{20,0} = |\Omega_{20,0}^n - \Omega_{20,0}^{n-1}| \leq \epsilon_{\Omega}
\]

(4.37)

where \( \epsilon_{\Omega} = 10^{-5} \). This indicates the maximum correction for outlet wall vorticity from one time step to the next. Experience with this method shows that this \( \epsilon_{\Omega} \) is a faithful indicator of ultimate convergence. That is, if (4.37) is satisfied, then the corrections for all boundary point vorticities are less than \( \epsilon_{\Omega} \). The maximum norm of residues of the vorticity transport equations calculated at all inner points by

\[
R_{\Omega} = \max_{i,j} \left| \left( \bar{\mathbf{g}} \cdot \nabla - \frac{\nabla^2}{Re} \right) \Omega_{i,j} \right|
\]

(4.38)

is \( 10^{-3} \) for \( Re = 10 \), and \( 10^{-4} \) for \( Re = 100 \). Thus the steady state solution has been achieved.

The wall vorticity formulae of both first order accuracy and second order accuracy were tested. The average percentage errors \( \epsilon_{\Omega} \) are essentially the same whether using the first order formula or the second order formula. For \( Re = 10 \) case, \( \epsilon_{\Omega} \) is 9.15 if the first order wall vorticity formula is used, and is 8.38 if the second order formula is used. However, it can be seen by point to point comparison from Table 2 that the first order formula works well for
the separation region, but introduces larger errors to the regions near the inlet and outlet. The second order formula significantly improves the accuracy near inlet and outlet, but at the cost of reducing the accuracy near the separation bubble. In particular, the position of the minimum wall vorticity is not correct when using the second order formula. This shows that, in the case of a coarse grid and 'weakly separated' flow, the one-sided three-point difference formula may not be appropriate for approximating the separation region. To obtain high resolution in the weakly separated flow region, a finer grid near the wall is preferred.

The minimum values of wall vorticity obtained by this method are less than that of Cliffe et al, namely stronger separations are predicated. This is partially because the modified inlet vorticity was used in this method. The other possible reason may due to using the central difference scheme for the advection term even though the diffusion is dominant in the present problem. On the other hand, the positions of the separation point and the reattachment point compare favourably to the benchmark solution.

The results for the \( Re = 100 \) case display a similar trend as that of the \( Re = 10 \) case. The average percentage error is 6.83. The wall vorticity value at the outlet is almost the same as that of Cliffe et al. and the minimum value is slightly lower.

Comparing the results from this method with those of the IAHR workshop, it can be concluded that the present method is reliable for computation of the flow in complex geometry. The reasons that influence the accuracy may include the inlet vorticity modification, time dependent formulation and central difference for convective terms. By using proper 'windward' finite differences, the present time dependent method has the potential to be extended
to deal with higher Reynolds number flows.
Table 2  Wall vorticity values for $Re = 10$

<p>| $x/x_0$ | Present | | Cliffe et al [27] |
|---------|---------|---------|
|         | 1st order | 2nd order |         |
| 0.00    | 3.0030   | 3.0000   | 3.0000 |
| 0.05    | 2.6430   | 2.5830   | 2.5750 |
| 0.10    | 1.7604   | 1.6480   | 1.7060 |
| 0.15    | 0.4124   | 0.3563   | 0.3990 |
| 0.20    | -0.1551  | -0.1921  | -0.1000|
| 0.25    | -0.1565  | -0.1739  | -0.1350|
| 0.30    | -0.1159  | -0.1269  | -0.1080|
| 0.35    | -0.1081  | -0.1165  | -0.1060|
| 0.40    | -0.1063  | -0.1115  | -0.1030|
| 0.45    | -0.0842  | -0.0857  | -0.0790|
| 0.50    | -0.0383  | -0.0375  | -0.0330|
| 0.55    | 0.0228   | 0.0241   | 0.0270 |
| 0.60    | 0.0902   | 0.0907   | 0.0920 |
| 0.65    | 0.1582   | 0.1565   | 0.1570 |
| 0.70    | 0.2233   | 0.2182   | 0.2170 |
| 0.75    | 0.2837   | 0.2736   | 0.2720 |
| 0.80    | 0.3382   | 0.3212   | 0.3190 |
| 0.85    | 0.3861   | 0.3597   | 0.3570 |
| 0.90    | 0.4263   | 0.3881   | 0.3850 |
| 0.95    | 0.4572   | 0.4053   | 0.4020 |
| 1.00    | 0.4572   | 0.4053   | 0.4080 |</p>
<table>
<thead>
<tr>
<th>$x/z_o$</th>
<th>Present</th>
<th>Cliffe et al [27]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>3.0000</td>
<td>3.0000</td>
</tr>
<tr>
<td>0.05</td>
<td>2.6945</td>
<td>2.5016</td>
</tr>
<tr>
<td>0.10</td>
<td>1.7041</td>
<td>1.7216</td>
</tr>
<tr>
<td>0.15</td>
<td>0.7336</td>
<td>0.7315</td>
</tr>
<tr>
<td>0.20</td>
<td>0.1189</td>
<td>0.1018</td>
</tr>
<tr>
<td>0.25</td>
<td>-0.0757</td>
<td>-0.0954</td>
</tr>
<tr>
<td>0.30</td>
<td>-0.1150</td>
<td>-0.1280</td>
</tr>
<tr>
<td>0.35</td>
<td>-0.0843</td>
<td>-0.0981</td>
</tr>
<tr>
<td>0.40</td>
<td>-0.0223</td>
<td>-0.0320</td>
</tr>
<tr>
<td>0.45</td>
<td>0.0577</td>
<td>0.0513</td>
</tr>
<tr>
<td>0.50</td>
<td>0.1444</td>
<td>0.1386</td>
</tr>
<tr>
<td>0.55</td>
<td>0.2302</td>
<td>0.2220</td>
</tr>
<tr>
<td>0.60</td>
<td>0.3021</td>
<td>0.2973</td>
</tr>
<tr>
<td>0.65</td>
<td>0.3719</td>
<td>0.3638</td>
</tr>
<tr>
<td>0.70</td>
<td>0.4252</td>
<td>0.4223</td>
</tr>
<tr>
<td>0.75</td>
<td>0.4792</td>
<td>0.4729</td>
</tr>
<tr>
<td>0.80</td>
<td>0.5165</td>
<td>0.5156</td>
</tr>
<tr>
<td>0.85</td>
<td>0.5588</td>
<td>0.5524</td>
</tr>
<tr>
<td>0.90</td>
<td>0.5828</td>
<td>0.5843</td>
</tr>
<tr>
<td>0.95</td>
<td>0.6162</td>
<td>0.6101</td>
</tr>
<tr>
<td>1.00</td>
<td>0.6162</td>
<td>0.6210</td>
</tr>
</tbody>
</table>
CHAPTER V. CONCLUSIONS

A curvilinear coordinate system $x^i$ has been introduced to study $2 - D$ flow in complex geometry. Some flow problems of current interest have been formulated in the $x^i$ coordinate system. The methods for numerically generating the $x^i$ coordinate system are discussed. A new design and analysis algorithm is proposed for inviscid incompressible flow. The solution procedure for Navier-Stokes equations is presented for an expanding channel flow. Numerical results from the present method show comparable accuracy to existing procedures.

The $x^i$ coordinate system consists of one family of straight lines and one family of general curves. Geometrical boundaries for arbitrarily shaped physical domains can be assigned to be the coordinate lines of the $x^i$ system, such that a rectangular computational domain can be easily constructed. The transformation from $y^i$ to $x^i$ is the simplest one that can transform arbitrary domains into a rectangle. Consequently, the flow equations in the $x^i$ system possess relatively simple forms compared to those written in a general curvilinear coordinate system. The variations of vectors and tensors along the coordinate lines and normal to the coordinate lines are distinguished by the changes of corresponding contravariant and covariant components. The relationship between contravariant and covariant components in the $x^i$ system and the Cartesian components can be easily determined, and either of them can be used to study the flow field.
The numerical construction of the coordinate system or the grid generation can be accomplished very efficiently. Since only one unknown coordinate in the physical domain is to be determined, the solving of two coupled differential equations is averted. This not only reduces the grid generation cost, but also makes the design by differential equations easy to handle. For many practical problems, the grid system generated by \( L(y) = 0 \) is adequate to be used to study the flow field. Although the \( z^i \) coordinate system can be generated by the algebraic method, using one single differential equation to design a geometric boundary, subject to some given surface property and/or field property, is of interest.

The coordinate system which uses streamlines as coordinate lines has the advantage for imposing boundary conditions. Because streamlines \( \psi(x, y) = \text{constant} \) is a special case of the family \( z(x, y) = \text{constant} \), the formulations obtained in Chapter I can be easily transferred to the \((x, \psi)\) coordinate system by simply changing \( z \) to \( \psi \). In particular, equations formulated for streamfunction reduce to the forms that are usually referred to as the von Mises coordinate equation since \( z = \psi, \psi_x = 0 \) and \( \psi_z = 1 \).

The von Mises method requires that \( y_\psi \) neither vanish nor becomes infinite. This is too restrictive for many practical problems. For incompressible inviscid flow, streamfunction \( \psi \) is the solution of the Laplace equation. Therefore, \( \psi \) attains its maximum and minimum on the boundary of the physical domain and for each \((x, y)\) point there exists one and only one \( \psi \). However, it will happen that more than one \( y \) exists for some \((x, \psi)\) if \( y_\psi \) is zero or infinite at some point. In this case, \( \psi \) loses its qualification to serve as a coordinate. For incompressible viscous flow, the velocity field has sources and the streamfunction is a solution of the Poisson equation. Since \( \psi \) does not necessarily
attain its extreme values on the boundary, it cannot serve as a coordinate either. If local reversed flow occurs, the von Mises coordinates cannot be used. For branch flow (including multielement flow), the von Mises method has to assume the position of the branch point and assign the rate of flow for each branch. This is at best difficult, and usually impossible. The $z^i$ coordinate system specifies only the solid wall to be the coordinate line and off the the wall the position of the coordinate line can be arbitrarily chosen. Thus the transformation requirement that $y_z$ is nonzero and finite can always be met.

The von Mises transformation, which uses the dynamic quantity $\psi$ as one of the coordinates and keeps the other coordinates unchanged, does not need the extra grid generation process. However, the governing equation in the $(x, \psi)$ plane is a nonlinear equation for the geometric quantity $y$. In case of compressible flow, this nonlinear equation is coupled with unknown density, and the iterative convergence is difficult. In the present method, the nonlinear grid generation equation is easy to solve, and the dynamic equation possesses a simple form, reducing the convergence difficulty.

The method proposed in this work is a combination of that obtained by full grid generation and that of the von Mises coordinate method. Some of the disadvantages of both of the above methods can be overcome while their main advantages are retained. At the same time, the numerical procedure is efficient.

The $z^i$ coordinate system is very suitable for channel type flow. For external flow problems, the ratio of the number of grid points on the body surface to the number of all grid points taken is comparatively low. Since the former relates to the accuracy and the latter reflects the computational cost, it is necessary to attempt to increase this ratio. In order to obtain adequate
resolution on a globally coarse grid, a stretching technique can be used to improve the grid distribution. This can also be done by solving a Poisson equation $L(y) = f(x, y)$ to generate the grid, where the source $f(x, y)$ can be designed to get a fine grid near the solid wall and a coarse grid in the far field.
REFERENCES


[5 ] Sackett, S. and Healey, R., JASON- A Digital Computer Program for the Numerical Solution of the Linear Poisson Equation \( \nabla \cdot (\eta \nabla \phi) + \rho = 0 \), UCRL-18721 Lawrence Radiation Laboratory, University of California, Berkeley, California (Feb. 1969).


VITA AUCTORIS

Honglin Ye was born in 1957 in Shandan, Gansu, China. He graduated from high school in Jan. 1974. In March 1978, he went to Northwest Normal University, Lanzhou, China where he obtained a B.Sc. in Physics in 1982. He served as a lecturer in Physics at Northwest Normal University for 7 years. He is now a student of the University of Windsor and hopes to obtain a Master's degree in Mathematics in Fall 1991.