Numerical solution of Euler equations on streamline-aligned meshes.

Azat M. Latypov

University of Windsor

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

Recommended Citation
https://scholar.uwindsor.ca/etd/2890

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 519-253-3000 ext. 3208.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI
NUMERICAL SOLUTION OF EULER EQUATIONS ON STREAMLINE-ALIGNED MESHES

by

Azat Latypov

A Dissertation
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 Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada

1997

©Azat Latypov
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/films, de reproduction sur papier ou sur format électronique.

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

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

0-612-52411-6
ABSTRACT

By formulating the governing equations of fluid motion in streamline-aligned coordinates one can avoid difficulties associated with grid generation and can solve problems of inverse design or problems with free boundaries.

In this dissertation, previous approaches utilizing streamline-aligned coordinates have been extended for both steady and unsteady flow motions.

The governing equations for steady inviscid two-dimensional gas motion have been formulated in an orthogonal system of independent coordinates consisting of the streamfunction and its orthogonal complimentary function. The resulting system of differential conservation laws expresses conservation of mass, momentum and energy. The conservative finite volume approximation of these equations can be used to calculate flows with strong shocks. For the case of potential velocity vector field, two different simplified formulations of the governing equations are derived. In order to compute purely supersonic flows, a conservative hybrid grid-characteristic scheme has been developed. To calculate transonic potential flows, two iterative algorithms have been implemented.

For the case of compressible unsteady flow, the streamline-aligned coordinates are introduced through a consideration of metric coefficients of the coordinate transformation. It is demonstrated that, if one family of coordinate lines is aligned along the velocity vector field at any time and certain compatibility conditions for this coordinate transformation are satisfied, it is possible to use the remaining degrees of freedom to ensure that the resulting coordinate system is orthogonal, or to specify the local value of the jacobian of the transformation. A central–difference Lax–Wendroff numerical scheme with additional explicitly
added dissipation is used to solve the resulting system of transformed equations.

The developed approaches and numerical algorithms are tested on calculations of sub-sonic, transonic and supersonic internal flows.
Contents

ABSTRACT ii

LIST OF FIGURES vii

NOMENCLATURE ix

1 Introduction 1

2 Formulation of Steady Euler Equations in Streamline–Aligned Co-
ordinates 8

2.1 Introduction .......................................... 8

2.2 Governing Equations in Streamline Coordinates .......... 10

2.2.1 Streamline–Aligned Coordinates ...................... 10

2.2.2 Derivation of the Governing Equations in Streamline–Aligned
Orthogonal Coordinates .................................. 12

2.2.3 Proof of the Governing System Formulation in Streamline Co-
ordinates .................................................. 17
2.2.4 The Governing System for Potential Flows .......... 20
2.2.5 Characteristics and Compatibility Conditions .......... 25
2.2.6 Boundary Conditions .................................. 30

3 Numerical Solution of Steady Euler Equations in Streamline–Aligned Coordinates

3.1 Introduction ............................................. 35
3.2 A Conservative Hybrid Grid–Characteristics Scheme .......... 38
  3.2.1 Conservative Approximation of the Governing System .... 38
  3.2.2 Linearized System of Equations ....................... 40
  3.2.3 Generalization to the Nonlinear Case ................. 46
3.3 An Iterative Algorithm for the Full–Potential Equivalent Equation ... 47
3.4 An Iterative Algorithm for the System of Two Equations ........ 50
3.5 Numerical Results .................................... 55

4 A Streamline–Adaptive Formulation for Unsteady Euler Equations 63

4.1 Introduction ............................................. 63
4.2 Streamline–Aligned Coordinate System .................... 66
4.3 Governing Equations in Streamline-Aligned Coordinate System .... 69
4.4 Characteristic Directions of the Extended Governing System .... 72
4.5 Appendix: Some Derivations from Sections 4.3 and 4.4 ........ 77
  4.5.1 Derivation of Eqns. (4.14) – (4.16) .................... 77
5 Numerical Solution of Unsteady Euler Equations in Streamline–

Aligned Coordinates

5.1 Introduction ................................................. 79

5.2 Coordinate Transformation Equations ....................... 80

5.2.1 Boundary Conditions and Formulation of the Problem .... 80

5.2.2 Numerical Solution of Coordinate Transformation Equations . 81

5.3 Transformed Euler Equations .................................. 83

5.3.1 Internal Nodes ............................................. 83

5.3.2 Boundary Nodes ........................................... 86

5.3.3 Artificial Dissipation Terms ............................. 92

5.4 Numerical Results ............................................ 92

6 Conclusions ..................................................... 110

References ....................................................... 114
List of Figures

2.1 Original and mapped domains. ........................................ 13
2.2 The characteristics directions. ..................................... 30

3.1 Finite volume and computational stencil. ......................... 39
3.2 Approximation coefficients’ space. ................................. 41
3.3 Mach number vs psi coordinate. .................................... 57
3.4 Jet emerging from supersonic nozzle. ............................. 58
3.5 A transonic flow in a nozzle. ....................................... 59
3.6 Pressure and Mach number distributions along the wall. ........ 60
3.7 Influence of the number of iterations on each grid level on the convergence rate. ............................................. 61
3.8 Subsonic flowfield in the axisymmetric bumpy channel. ....... 62

4.1 Local geometry of the grid in terms of the metric coefficients. 67

5.1 Subsonic channel with 20% bump. ................................. 96
5.2 Subsonic channel with 20% bump. ................................. 97
5.3 Subsonic channel with 20% bump. Pressure contours. .................. 98
5.4 Subsonic channel with 20% bump. Computational grid. .................. 99
5.5 Transonic channel with 10% bump. ........................................ 100
5.6 Transonic channel with 10% bump. ........................................ 101
5.7 Transonic channel with 10% bump. Pressure contours. .................. 102
5.8 Transonic channel with 10% bump. Computational grid. .................. 103
5.9 Flow about an oscillating membrane. $t = 0.05$ sec. ..................... 104
5.10 Flow about an oscillating membrane. $t = 0.10$ sec. ..................... 105
5.11 Flow about an oscillating membrane. $t = 0.15$ sec. ..................... 106
5.12 Flow about an oscillating membrane. $t = 0.20$ sec. ..................... 107
5.13 Flow about an oscillating membrane. $t = 0.25$ sec. ..................... 108
5.14 Flow about an oscillating membrane. $t = 0.30$ sec. ..................... 109
**NOMENCLATURE**

\[ A, B, C \] flux vectors, introduced in (4.13)

\[ A^{(e)}, A^{(p)} \] matrices, defined in the relations following (2.26)

\[ A_{0}^{(e)}, A_{0}^{(p)} \] local linearization matrices, introduced in (3.4)

\[ a, b, c \] flux vectors, introduced in (4.10)

\[ [a] \] jump of the quantity \( a \) across the shock

\[ B(\psi) \] positive function used to control the mesh, introduced in (2.3)

\[ C \] integrating factor used to introduce \( \varphi \) in (2.2)

\[ C_{0}, C_{n}, C_{\pm} \] characteristic lines

\[ D \] full enthalpy

\[ D_{\pm} \] operator of differentiation along \( C_{\pm} \)

\[ D_{n+1/2,m} \] a finite volume

\[ E, G, F \] metric coefficients, introduced in (4.3)

\[ e \] internal energy

\[ f = \frac{J_{1}}{J_{2}} \], introduced in the relations preceding (3.10)

\[ g(z) = \frac{1}{z^{(2+\gamma)}} \], introduced in the relations following (2.26)

\[ h_{\varphi}, h_{\psi}, h_{\tau} \] stepsizes of the grid in \( \varphi, \psi \) and \( \tau \) directions

\[ J \] Jacobian of the coordinate transformation

\[ J_{1}, J_{2} \] factors introduced in (2.7), (2.8) and (2.9)

\[ p \] pressure

\[ q \] velocity vector magnitude
\( R_{m \pm 1/2} \) interpolation matrices, introduced in (3.2) and (3.3)

\( \tilde{R} \) diagonal matrix, introduced in (3.7)

\( \tilde{R}_{ii} \) diagonal entries of the matrix \( \tilde{R} \)

\( R^+_{n}, u_i \) Riemann invariants

\( S = p/\rho^*, \) entropy function

\( s = \sqrt{z-1} \)

\( t, \tau \) time

\( U = (z, \theta, C, S, D)^T, \) vector of primitive variables

\( u, v \) velocity vector components

\( v \) velocity vector

\( w, f \) steady Euler flux vectors, introduced in (2.1). Also vectors of geometric variables introduced in (4.4) and (4.5)

\( W, F, H \) flux vectors, defined in the relations following (2.12)

\( (x, y) \) Cartesian coordinates

\( z = M^2, \) squared Mach number

\( \alpha^* \) scalar parameter, introduced in (3.5)

\( \gamma = \kappa - 1 \)

\( \varepsilon^{(n)} \) correction vector, introduced in (3.11)

\( \kappa \) ratio of specific heats

\( \xi_{\alpha}, \xi_{\gamma} \) eigenvalues of the characteristic equation

\( \nu \) geometry factor (\( \nu = 0 \) for plane flows and \( \nu = 1 \) for axisymmetric flows)
\( \tau^* \) iterative parameter, introduced in (3.11)

\( \Omega(x,y) \) an arbitrary domain in \( x-y \) plane

\( \Omega(\phi, \psi) \) image of \( \Omega(x, y) \) in \( \phi-\psi \) plane

\( \Omega \) matrix, introduced in (3.7)

\( \omega_i \) eigenvectors of the characteristic equation

\( \rho \) density

\( \sigma_i \) CFL number, introduced in (3.5)

\( \theta \) = \( \arctan(v/u) \), velocity vector argument

(\( \phi, \psi \)) streamline-aligned coordinates
Chapter 1

Introduction

There are several advantages in formulating the governing equations of steady gas motion in a system of independent coordinates aligned with streamlines.

Von Mises [1] used the streamfunction as one of the independent coordinates in boundary layer studies. The second independent coordinate was chosen to be the transverse Cartesian coordinate to the flow direction. Stanitz [2,3] developed a modification of the von Mises approach in order to solve inverse design problems of potential gas dynamics. The essence of Stanitz's modification was that a velocity vector potential was used as the second independent coordinate instead of the transverse Cartesian coordinate.

A related and more general approach was suggested by Martin in inviscid gas dynamics [4] as well as in viscous incompressible fluid dynamics [5].

During the last 15 years a number of techniques, based upon the "streamfunction-
as-a-coordinate" (SFC) ideology, have been used in Computational Fluid Dynamics (CFD). The latest review of these works as well as a bibliography can be found in [6].

One of the attractive advantages provided by the SFC methodology in CFD applications is that it permits computation of the parameters of the flow without prior grid generation in the computational domain. More exactly, the governing equations play a double role of both the equations describing the motion of the media and the grid generation equations. As a result of this, both the computational time and the memory requirements can be reduced. Furthermore, the resulting streamline-aligned computational grid naturally conforms to the boundaries of the physical domain. These features are utilized in most of the works dealing with the SFC method in CFD (see [6]–[43]).

Other applications of CFD, in which the SFC concept has been found to be useful, include the class of problems with initially unknown shape of a boundary segment. Typical problems belonging to this class are ones with free or elastic boundaries, problems involving inverse design of aerodynamic shapes for given distribution of some physical quantities (e.g., pressure) along a segment of the boundary, or problems involving the optimal design of the shape, which provides the maximum or minimum to some integral functional, calculated from the solution (e.g., minimization of a drag force, or maximization of a lift force acting on an airfoil).

The common feature of problems falling into this category is that the shape of at least a part of the boundary is not given, but rather needs to be determined as a
result of the solution procedure. As a consequence, the use of conventional iterative approaches requires adjusting the computational grid to the geometry of the domain at each iteration. Since the typical domain in this class of problems is bounded by streamlines, the SFC technique makes it possible to avoid the costly procedure of grid recalculation.

This advantage has been exploited in a number of works. Applications of the SFC concept to airfoil, turbomachinery and nozzle inverse design problems can be found in [2, 3, 15–17, 24, 25, 29]. A problem in which the shape of the nozzle wall is prescribed everywhere but in the vicinity of the nozzle’s throat is considered in [42]. The shape of the wall near the throat is designed so that the distribution of pressure along this part of the wall remains smooth.

The SFC technique has been successfully applied to the numerical solution of free boundary problems involving jets in [26, 32–34].

Among the areas of application mentioned above, problems with elastic boundaries and problems of optimal design of aerodynamic shapes remain the areas in which the SFC technique’s advantages have not been widely exploited thus far.

Although the SFC concept has mainly been used in theoretical fluid dynamics and CFD due to its advantages in the fields mentioned above, there exist a number of works extending this methodology to other applications. In [35–37] streamline independent coordinates have been applied to the study of turbulence and chaos occurring in a fluid flow. In [18, 19] properties of a flow in porous media were in-
vestigated using the von Mises coordinates. This coordinate transformation has also been applied to two-phase fluid flow in [10]. In the latter case, one needs to consider two streamfunctions, each corresponding to one of the phases existing in the flow. In [38] a streamline-aligned coordinate transformation was used in order to obtain a simplified form of the Navier-Stokes equations. This simplification (called the "parabolized" Navier-Stokes equations) is based on neglecting some of the second order spatial derivatives in order to make the steady-state equations elliptic only within the boundary layer and parabolic elsewhere. This feature simplifies the numerical solution of the resulting system, while giving better approximation to the original Navier-Stokes equations than obtained from the boundary layer theory. As an order of magnitude comparison shows, the terms to be neglected are to be chosen from those containing the second order derivatives in the direction of the flow. Writing the governing equations in a streamline-aligned coordinate system simplifies the selection of terms to be neglected.

An essential feature of the SFC technique is the choice of the independent coordinate or coordinates which compliment the streamfunction to produce the nondegenerate system of independent coordinates. It is well known that one of the major limitations of the original von Mises approach [1] is that the resulting system of coordinates degenerates at the locations where the velocity vector is normal to the axis of the transverse Cartesian coordinate. This situation is typical for the flow in the vicinity of the leading edge of an airfoil. This limits applicability of the von Mises
approach to cases where one can expect in advance that the flow direction will not
change significantly over the flow domain.

A number of modifications to the von Mises coordinates have been tried. A dis-
cussion and references on this issue can be found in [6]. Out of the most recent
approaches, not covered in the above-cited review, one should mention the technique
developed in [21–23], in which it is advocated to use the Lagrangian time as a comple-
mentary independent coordinate to two streamfunctions in three-dimensional steady
gas flow.

In the current work we consider a streamline–based transformation which gener-
ates an orthogonal streamwise coordinate system. Such a coordinate system, gen-
erated by streamfunction and velocity vector potential for the solution of design
problems in inviscid compressible potential flows, was suggested in [2, 3]. Later, this
approach was generalized to nonpotential flows by a number of authors (eg, [30–
32, 43]).

The orthogonality of this system ensures that, even in the case of a flow region
with complex geometry, the coordinate transformation does not degenerate. A further
advantage of the formulation used in the current work is that it is expressed as a
system of “physically consistent” conservation laws and, as a consequence, it may be
used to calculate flows with strong shocks.

Use of a streamfunction as one of the independent coordinates in the case of
steady flows essentially utilizes the steady form of the continuity equation. If com-
pressible unsteady flow is being considered, a streamfunction cannot be introduced. Due to this limitation, most of the works dealing with compressible Euler equations in streamline-aligned coordinates tackled only steady flows. An important step in generalizing the streamline-aligned coordinates to unsteady flows was made in [44], where a Lagrangian approach to the description of unsteady fluid motion has been extended. The essence of this extension is that the system of Euler equations is initially written in an arbitrary time-dependent system of coordinates. Following that, instead of associating each point in the new coordinates with a fluid particle (as in the case of classical Lagrangian formulation), the author of [44] suggests imposing a grid constraint condition in the finite-volume discretization of the governing equations. This constraint condition ensures that the boundary of the finite-element cell is parallel to the local fluid velocity vector. As the flow evolves in time, the extended Lagrangian grid moves in such a way that two out of four cell boundaries of each computational cell are aligned along the velocity vector field. In this approach, the coordinate transformation is introduced on the stage of discretization.

In this dissertation, an alternative approach has been proposed. Namely, the requirement that one family of coordinate lines is aligned along the velocity vector field is ensured in the system of the governing equations, prior to considering its discretization. By satisfying this requirement at the “continuous” stage of formulation, correct boundary conditions can be imposed for the geometric unknown functions which are introduced into the governing system.
The dissertation is organized as follows. In Chapter 2, the streamline-aligned orthogonal coordinates are introduced and steady Euler equations are formulated in this coordinate system. The case of potential flow is also considered and two different full-potential-equivalent formulations are proposed. Then, the characteristics and compatibility conditions for the governing system are derived, and the boundary conditions are formulated and discussed.

In Chapter 3, a conservative hybrid grid-characteristics scheme for steady supersonic flow calculations is constructed and numerical algorithms to solve the transonic full-potential-equivalent equations are given. Sample calculations are performed for subsonic, transonic and supersonic flows.

In Chapter 4, unsteady streamline-aligned coordinates are introduced and the system of Euler equations is written in these coordinates.

Chapter 5 describes the numerical algorithm used to solve the unsteady Euler equations using the formulation developed in the previous chapter. This algorithm is applied to the unsteady flow of a gas through a channel with a flexible membrane wall.

Finally, in the last chapter, conclusions are given and advantages and limitations of the present approach are discussed.
Chapter 2

Formulation of Steady Euler
Equations in Streamline–Aligned
Coordinates

2.1 Introduction

The Euler equations are conservation laws expressed as a first-order system of non-linear partial differential equations. The physical significance of these equations stems from the fact that they express conservation of mass, momentum and energy in continuous compressible media in the absence of viscous dissipation.

Historically, the first works dealing with numerical modeling of compressible gas motion were based on the transonic small disturbance (TSD) equation [50] and later
the full potential (FP) equation [51, 52]. Both of these equations are simplifications of the original system of Euler equations. TSD equation is obtained from the system of Euler equations under the assumption that the Mach number is close to one ($M - 1 \ll 1$) everywhere in the flowfield, while the FP equation results from the assumption that the velocity vector field is irrotational and can be represented through a scalar potential function.

Although both of these reduced forms of Euler equations were used in the development of early numerical algorithms for transonic flows calculations, the underlying simplifying assumptions used in their derivations do not allow them to be used to describe flowfields which contain subsonic, transonic and supersonic zones and/or flows with shocks, which result in vorticity production in the flowfield.

Starting from the late 70's and early 80's, a number of researchers (eg. [53, 54, 56–58]) developed numerical algorithms for the full system of Euler equations. A survey of some of these works can be found in Section 3.1.

In the following sections, the steady form of Euler equations is formulated using the concept of streamline-aligned orthogonal coordinates. For the case of potential flow, the analogue of a FP equation is derived in these coordinates. Also, characteristic and compatibility condition for this new formulation are derived and used to formulate the appropriate boundary–value problem.
2.2 Governing Equations in Streamline Coordinates

2.2.1 Streamline–Aligned Coordinates

Consider the system of steady two-dimensional Euler equations in Cartesian coordinates \((x, y)\):

\[
\frac{\partial w}{\partial x} + \frac{\partial h}{\partial y} = \nu h, \tag{2.1}
\]

where

\[
w = \begin{bmatrix}
y'' \rho u \\
y''(p + \rho u^2) \\
y'' \rho uv \\
y'' \rho u D \\
C u
\end{bmatrix}, \quad f = \begin{bmatrix}
y'' \rho v \\
y'' \rho uv \\
y''(p + \rho u^2) \\
y'' \rho v D \\
- C u
\end{bmatrix}, \quad h = \begin{bmatrix}
0 \\
0 \\
p \\
0 \\
0
\end{bmatrix},
\]

and \(D = q^2/2 + e + p/\rho\) is full enthalpy, \(q^2 = u^2 + v^2\), \(e = p/(\kappa - 1)\rho\) is the internal energy of a perfect gas and \(C\) is the unknown function which will be discussed below.
In equation (2.1)

\[
\nu = \begin{cases} 
0, & \text{for plane flows}, \\
1, & \text{for axisymmetric flows}.
\end{cases}
\]

In the following we use the notation (2.1.i) to refer to the \(i\)-th equation of the system (2.1). The equations (2.1.1)–(2.1.4) express the conservation of mass, two components of momentum and energy respectively.

Consider the two differential relations

\[
d\varphi = C(udx + vdy),
\]
\[
d\psi = B(\psi)y^{\nu}\rho(udy - vdx),
\]

where \(B(\psi)\) is a given positive function, which will be used to refine the mesh. It should be noted that the right-hand sides of both (2.2) and (2.3) are full differentials because of the equations (2.1.1) and (2.1.5) respectively. It should also be emphasized that (2.1.5) implies that the new unknown function \(C\) is an integrating factor, which makes it possible to introduce \(\varphi\) in the general case of vortical flow. Therefore, (2.2) and (2.3) define a pair of functions

\[
\varphi = \varphi(x, y)
\]
\[
\psi = \psi(x, y).
\]
The value of each of these functions can be determined up to one arbitrary constant.

It follows from (2.2) and (2.3) that

\[ \mathbf{v} \cdot \nabla \psi = 0, \quad (2.4) \]
\[ \nabla \psi \cdot \nabla \varphi = 0, \quad (2.5) \]

where \( \mathbf{v} \) is the velocity vector. In other words, contours of \( \psi \) are aligned along the direction of the velocity vector field, while contours of \( \varphi \) are perpendicular to this direction. The pair of functions \( \varphi \) and \( \psi \) will be referred to as \emph{streamline coordinates} or \emph{streamline variables}.

### 2.2.2 Derivation of the Governing Equations in Streamline–Aligned Orthogonal Coordinates

The change of independent variables from \( x \) and \( y \) to \( \varphi \) and \( \psi \) in (2.1) can be done as in [43], using (2.2) and (2.3) and the chain rule

\[
\frac{\partial}{\partial x} = \frac{\partial \varphi}{\partial x} \frac{\partial}{\partial \varphi} + \frac{\partial \psi}{\partial x} \frac{\partial}{\partial \psi} = C u \frac{\partial}{\partial \varphi} - B(\psi)y'' \rho_v \frac{\partial}{\partial \psi} \]

\[
\frac{\partial}{\partial y} = \frac{\partial \varphi}{\partial y} \frac{\partial}{\partial \varphi} + \frac{\partial \psi}{\partial y} \frac{\partial}{\partial \psi} = C v \frac{\partial}{\partial \varphi} + B(\psi)y'' \rho_u \frac{\partial}{\partial \psi}
\]
Figure 2.1: Original and mapped domains.

In the current work, however, the transformation is done using the change of independent variables applied directly to the system of integral conservation laws corresponding to (2.1). Apart from simplifying the algebra, this enables us to obtain the governing system in \( \varphi - \psi \) coordinates, written as a system of conservation laws.

Consider a gas flow in some domain \( ABCD \) (Fig. 2.1). Assume that the boundary \( AB \) is a solid wall, \( CD \) is either a solid wall or a plane/axis of symmetry, \( AD \) is the "inlet" and \( BC \) is the "outlet" and both are orthogonal to the local direction of \( \mathbf{v} \).

It is well known that the system of Euler equations may not have a smooth solution in a classical sense even if the boundary data are smooth enough. In order to take into account non-smooth flows (that is, those with shocks or contact discontinuities), one must consider the concept of a weak solution.

Consider an arbitrary subdomain \( \Omega(x, y) \) of \( ABCD \). Integrating (2.1) over \( \Omega(x, y) \)
and applying Gauss’ theorem, one obtains

\[
\oint_{\partial \Omega(x,y)} w \, dy - f \, dx = \nu \iint_{\Omega(x,y)} h \, dx \, dy \quad (2.6)
\]

Equation (2.1) is said to have a weak solution in \(ABCD\) if (2.6) is valid for any subdomain \(\Omega(x,y)\).

After transformation to the coordinates \(\varphi - \psi\), \(ABCD\) becomes a rectangle \(A'B'C'D'\), while \(\Omega(x,y)\) maps into \(\Omega(\varphi,\psi)\) (Fig. 2.1).

To define a weak solution in the \(\varphi - \psi\) plane which is consistent with (2.6), one should express \(dx\), \(dy\) and \(dz\) in terms of \(d\varphi\), \(d\psi\) and \(d\varphi d\psi\). It follows from (2.2) and (2.3) that

\[
dy = uJ_1 d\psi + vJ_2 d\varphi, \quad (2.7)
\]
\[
dx = -vJ_1 d\psi + uJ_2 d\varphi, \quad (2.8)
\]
\[
dz = q^2 J_1 J_2 d\psi d\varphi, \quad (2.9)
\]

where

\[
J_1 = \frac{1}{B(\psi) y^* \rho q^2},
\]
\[
J_2 = \frac{1}{C q^2}.
\]
However, substitution of (2.7)--(2.9) into (2.6) shows that the left-hand sides of the relations (2.6.1) and (2.6.5) become identically zero and therefore the corresponding relations in the \( \varphi - \psi \) plane are both identically satisfied. The reason for this is that the transformation to the \( \varphi - \psi \) variables is only possible when the consistency relations (2.1.1) and (2.1.5) hold. In other words, the very existence of the variables \( \varphi - \psi \) as unique-valued functions of \( x \) and \( y \) (which is assumed before doing the change of variables in (2.6) ) implies that the equations (2.1.1) and (2.1.5) (together with the relations (2.6.1) and (2.6.5), corresponding to them) are valid. As a consequence, (2.6.1) and (2.6.5) are automatically satisfied in the \( \varphi - \psi \) plane.

At the same time, the right-hand sides of (2.7) and (2.8) must be full differentials, in order for these relations to be consistent. This gives the following conditions:

\[
\begin{align*}
\oint_{\partial \Omega(\varphi, \psi)} uJ_2 d\varphi + uJ_1 d\psi &= 0, \\
\oint_{\partial \Omega(\varphi, \psi)} uJ_2 d\varphi - vJ_1 d\psi &= 0,
\end{align*}
\]

for any domain \( \Omega(\varphi, \psi) \) from \( A'B'C'D' \).

Attaching (2.10) and (2.11) to (2.6.2)--(2.6.4), rewritten in the \( \varphi - \psi \) plane, one obtains the following system of integral conservation laws:
\[ \int_{\Omega(\phi, \psi)} W d\psi - F d\phi = \nu \int_{\Omega(\phi, \psi)} H d\phi d\psi, \]  

(2.12)

where

\[ W = \begin{bmatrix} uJ_1 \\ vJ_1 \\ u(p + \rho q^2)y\nu J_1 \\ v(p + \rho q^2)y\nu J_1 \\ D \end{bmatrix}, \quad F = \begin{bmatrix} -\nu J_2 \\ \nu J_2 \\ -\nu p y\nu J_2 \\ \nu p y\nu J_2 \\ 0 \end{bmatrix}, \quad H = \begin{bmatrix} 0 \\ 0 \\ 0 \\ p q^2 J_1 J_2 \end{bmatrix}. \]

For smooth flows (2.12) is equivalent to

\[ \frac{\partial W}{\partial \phi} + \frac{\partial F}{\partial \psi} = \nu H, \]  

(2.13)

which is the governing system of equations written in streamline-aligned orthogonal coordinates. In the following, the fluxes \( W \) and \( F \) in (2.13) are assumed to be functions of the vector of primitive variables \( U \), which in this work is chosen to be \( U = (z, \theta, C, S, D)^T \), where \( z = M^2 \) is the squared Mach number, \( \theta = \arctan(v/u) \) is the velocity vector argument, \( S = p/\rho^\kappa \) is the entropy function, and \( \kappa \) is the specific heat ratio.
Similar to the definition of the weak solution to (2.1), we say that (2.13) has a weak solution in $A'B'C'D'$ if (2.12) is valid for any subdomain $\Omega(\varphi, \psi)$.

2.2.3 Proof of the Governing System Formulation in Streamline Coordinates

In Section 2.2.2 the consistency conditions (2.10) and (2.11) were added to the system (2.13) as a substitute for the continuity equation (2.1.1) and the equation (2.1.5) which are automatically satisfied in the $\varphi - \psi$ plane. Unlike (2.1.1) and (2.1.5), equations (2.10) and (2.11) have no direct physical meaning and, in order to justify this substitution, a more formal proof is given in this Section.

Proposition 1 Assume the following is true:

1. $U = U(\varphi, \psi)$ is a weak solution of (2.13) in $A'B'C'D'$ ;

2. The functions $x = x(\varphi, \psi)$ and $y = y(\varphi, \psi)$ are defined on $A'B'C'D'$ as follows:

\[
x = x_0 + \int_{(\varphi_0, \psi_0)}^{(\varphi, \psi)} \psi J_1 d\psi + u J_2 d\varphi, \tag{2.14}
\]
\[
y = y_0 + \int_{(\varphi_0, \psi_0)}^{(\varphi, \psi)} \psi J_1 d\psi + v J_2 d\varphi, \tag{2.15}
\]

where $x_0$ and $y_0$ are the values of $x$ and $y$ at the point $(\varphi_0, \psi_0)$, which may be chosen arbitrarily, and
3. Equations (2.14) and (2.15) map $A'B'C'D'$ to a domain $ABCD$ in the $x$-$y$ plane and this mapping can be globally inverted, that is

$$
x = x(\varphi, \psi),
$$

$$
y = y(\varphi, \psi)
$$

$$
A'B'C'D' \quad \longleftrightarrow \quad ABCD \quad (2.16)
$$

$$
\varphi = \varphi(x, y)
$$

$$
\psi = \psi(x, y)
$$

Then, $\bar{U}(x, y) = U(\varphi(x, y), \psi(x, y))$ is a weak solution of (2.1).

**Remark.** Because of the relations (2.12.1) and (2.12.2) from the definition of the weak solution in the $\varphi$-$\psi$ plane, the values of the contour integrals (2.14) and (2.15) are path independent.

**Proof of Proposition 1.** To prove the proposition we need to prove the validity of (2.6) for all $\Omega(x, y) \subset ABCD$.

Let $\Omega(\varphi, \psi)$ be the image of $\Omega(x, y)$ under the mapping (2.16).

First notice that (2.12.3)—(2.12.5) from the definition of a weak solution in the $\varphi$ - $\psi$ plane have been derived in Section 2.2.2 from the relations (2.6.2)–(2.6.4) respectively, using the change of variables (2.7)–(2.9). Starting from (2.12.3)–(2.12.5)
(which are valid by the definition of a weak solution in the \( \varphi - \psi \) plane) and proceeding backwards, one can easily obtain (2.6.2)-(2.6.4).

To prove (2.6.1) we write

\[
\int_{\partial \Omega(x,y)} y^\nu \rho u dy - y^\nu \rho u dx = \\
\int_{\partial \Omega(\varphi,\psi)} y^\nu \rho u (u J_1 d\psi + v J_2 d\varphi) - y^\nu \rho u (-v J_1 d\psi + u J_2 d\varphi) = \\
\int_{\partial \Omega(\varphi,\psi)} 0 \, d\varphi + \frac{d\psi}{B(\psi)} \equiv 0.
\]

Similarly, for (2.6.5)

\[
\int_{\partial \Omega(x,y)} C u dy - C u dx = \\
\int_{\partial \Omega(\varphi,\psi)} C u (u J_1 d\psi + v J_2 d\varphi) - C u (-v J_1 d\psi + u J_2 d\varphi) = \\
\int_{\partial \Omega(\varphi,\psi)} d\varphi + 0 d\psi \equiv 0.
\]

This completes the proof of Proposition 1.

**Remark.** It is known [45] that, for discontinuous flows, (2.6) leads to a system of algebraic relations for values of the gas parameters along both sides of the
discontinuity, that is

\[ [w] \left( \frac{dy}{dx} \right)_\text{disc.} - [f] = 0, \]  \hspace{1cm} (2.17)

where \( \left( \frac{dy}{dx} \right)_\text{disc.} \) is the local slope of the discontinuity and \([a]\) denotes jump of the quantity \(a\).

It follows, from Proposition 1, that the weak solution of (2.12) mapped onto the \(x-y\) plane satisfies (2.17). Consequently, the conservative approximation of the equation (2.13) may be used for the calculation of flows with shocks.

We also mention that in addition to shock or contact discontinuity conditions for the gas parameters \([45]\), the unknown function \(C\) satisfies the jump relations

\[ [C] = 0 \quad \text{across a shock}, \]

\[ [Cq] = 0 \quad \text{across a contact discontinuity}. \]

### 2.2.4 The Governing System for Potential Flows

The assumptions that the velocity vector field is potential (irrotational) and that the flow is isentropic \((S \equiv \text{const})\) and isenthalpic \((D \equiv \text{const})\) are well established in modelling of compressible flows. Under these assumptions, the equations of mo-
mentum and energy follow from the continuity equation and the condition that the velocity vector has a potential. In this case, the governing system may be reduced to just one second-order equation for the velocity vector potential. However, these assumptions are not valid if the flow contains strong curved shocks.

Following [45], the governing system in Cartesian coordinates can be written as

\[ \frac{\partial \mathbf{w}^{(p)}}{\partial x} + \frac{\partial \mathbf{f}^{(p)}}{\partial y} = 0, \]  \hspace{1cm} (2.18)

where

\[ \mathbf{w}^{(p)} = \begin{bmatrix} y^\nu \rho u \\ -u \end{bmatrix}, \quad \mathbf{f}^{(p)} = \begin{bmatrix} y^\nu \rho u \\ -u \end{bmatrix}. \]

The streamline coordinates are again introduced through the relations (2.2) and (2.3). But, as it follows from (2.18.2) and these relations, there is no longer a need to resort to the unknown \( C \) (an integrating factor). \( C \) can be chosen equal to any positive function of \( \varphi \),

\[ C = C(\varphi). \]

Like the function \( B(\psi) \), \( C(\varphi) \) serves to control the refinement of the mesh in \( x-y \) coordinates.
The governing system in \( \varphi-\psi \) coordinates then reduces to

\[
\frac{\partial \mathbf{W}(p)}{\partial \varphi} + \frac{\partial \mathbf{F}(p)}{\partial \psi} = 0,
\]

(2.19)

where

\[
\mathbf{W}(p) = \begin{bmatrix} u J_1 \\ v J_1 \end{bmatrix}, \quad \mathbf{F}(p) = \begin{bmatrix} -v J_2 \\ u J_2 \end{bmatrix}.
\]

The vector of primitive variables in this case is chosen to be \( \mathbf{U}(p) = (z, \theta)^T \).

In the following, two full-potential-equivalent formulations of the governing system are considered. In each of these cases the result is a set of second order equations, each of which can be solved numerically using computational methods which have been developed for the full-potential equation.

**Reduction of the governing equations to a single second order equation**

It follows from (2.7) that

\[
\frac{u}{\rho q^2} = B(\psi) y' \frac{\partial y}{\partial \psi},
\]

(2.20)

\[
\frac{v}{q^2} = C(\varphi) \frac{\partial y}{\partial \varphi}.
\]

(2.21)

Substituting these relations into the condition for the right-hand side of (2.8) to
be a full differential yields

\[ \frac{\partial}{\partial \varphi} \left( f \frac{\partial y}{\partial \varphi} \right) + \frac{\partial}{\partial \psi} \left( \frac{1}{f} \frac{\partial y}{\partial \psi} \right) = 0, \quad (2.22) \]

where

\[ f = \frac{J_1}{J_2} = \frac{C(\varphi)}{B(\psi) y^\nu \rho}. \quad (2.23) \]

Note that equation (2.22) is (2.19.2), written in terms of \( y \) and its derivatives, and that (2.19.1) is also satisfied because of the relations (2.20)–(2.21). Therefore, (2.22) can be taken as a governing equation in place of the system (2.19).

If the function \( y = y(\varphi, \psi) \) is given, the primitive variables \( z \) and \( \theta \) can be calculated from (2.20)–(2.21), because in an isentropic and isenthalpic flow \( \rho \) and \( q \) are both known functions of \( z \), and \( u = q \cos \theta \) and \( v = q \sin \theta \).

However,

\[ \frac{\partial \left( \frac{u}{x^2}, \frac{v}{z^2} \right)}{\partial (z, \theta)} = \frac{1 - z \cos^2 \theta}{z(2 + (\kappa - 1)z)\rho q^2} \]

This means that for transonic flows containing so-called limiting lines on which
where $z \cos^2 \theta = 1$, the set of equations (2.20) and (2.21) cannot be uniquely resolved with respect to $z$ and $\theta$. To chose the correct root in (2.20) and (2.21), one more differential equation has to be applied in the transonic regions of flow [41].

System of two second order equations

In this section, an alternative formulation which makes it possible to avoid the problem of nonuniqueness of the solution to the system (2.20) and (2.21) is derived.

Using (2.7) and (2.8), the fluxes $W$ and $F$ can be written in terms of the partial derivatives of the functions $z = x(\varphi, \psi)$ and $y = y(\varphi, \psi)$:

$$ W(\rho) = \begin{bmatrix} f \frac{\partial x}{\partial \varphi} \\ f \frac{\partial y}{\partial \varphi} \end{bmatrix}, \quad F(\rho) = \begin{bmatrix} 1 \frac{\partial x}{\partial \psi} \\ 1 \frac{\partial y}{\partial \psi} \end{bmatrix}, $$

where $f$ is given by (2.23), but $z$ and $\theta$ are to be determined from

$$ \frac{u}{q^2} = C(\varphi) \frac{\partial x}{\partial \varphi}, \quad \frac{v}{q^2} = C(\varphi) \frac{\partial y}{\partial \varphi}. \quad (2.24) $$

Unlike the Jacobian of the system (2.20) and (2.21), the Jacobian of (2.24) and (2.25) does not become zero, and consequently the latter system can be resolved with respect to $z$ and $\theta$ in a unique way.

The set of equations (2.19) then becomes a system of two coupled second order
equations for $x$ and $y$.

### 2.2.5 Characteristics and Compatibility Conditions

In this section the characteristic directions and the compatibility conditions along these characteristics for the governing system (2.13) are derived.

In the case of plane geometry, equation (2.13) can be rewritten as

$$ A^{(\omega)} \frac{\partial U}{\partial \varphi} + A^{(\psi)} \frac{\partial U}{\partial \psi} = 0, \quad (2.26) $$

where $U$ is a vector of primitive variables, and

$$ A^{(\omega)} = \frac{\partial W}{\partial U} = \begin{pmatrix} (z - 1)g(z)W^{(1)} & -W^{(2)} & 0 & W^{(1)}/(\gamma S) & -(\kappa + 1)W^{(1)}/(2\gamma D) \\ (z - 1)g(z)W^{(2)} & W^{(1)} & 0 & W^{(2)}/(\gamma S) & -(\kappa + 1)W^{(2)}/(2\gamma D) \\ (z - 1)g(z)pW^{(1)} & -W^{(4)} & 0 & 0 & -W^{(3)}/(2D) \\ (z - 1)g(z)pW^{(2)} & W^{(3)} & 0 & 0 & -W^{(4)}/(2D) \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ A^{(\psi)} = \frac{\partial F}{\partial U} = \begin{pmatrix} \end{pmatrix} $$
\[
\begin{pmatrix}
-g(z)F^{(1)} & -F^{(2)} & -F^{(1)}/C & 0 & -F^{(1)}/(2D) \\
-g(z)F^{(3)} & F^{(1)} & -F^{(2)}/C & 0 & -F^{(2)}/(2D) \\
-(1 + \kappa z)g(z)F^{(3)} & -F^{(4)} & -F^{(3)}/C & -F^{(3)}/(\gamma S) & -(\kappa + 1)F^{(3)}/(2\gamma D) \\
-(1 + \kappa z)g(z)F^{(4)} & F^{(3)} & -F^{(4)}/C & -F^{(4)}/(\gamma S) & -(\kappa + 1)F^{(4)}/(2\gamma D) \\
0 & 0 & 0 & 0 & 0
\end{pmatrix}
\]

Here, \( W^{(i)} \) and \( F^{(i)} \) denote the \( i \)-th components of the vectors \( W \) and \( F \) respectively, and

\[
\gamma = \kappa - 1,
\]
\[
g(z) = \frac{1}{z(2 + \gamma z)}.
\]

Following [45], consider the eigenvector problem (the characteristic equation)

\[
\omega_i^T (\xi_i^{(\omega)} A^{(\omega)} + \xi_i^{(\psi)} A^{(\psi)}) = 0^T,
\]  
(2.27)

where \( \xi_i^{(\omega)} \) and \( \xi_i^{(\psi)} \) are the eigenvalues, and \( \omega_i \) are the corresponding eigenvectors.

The characteristics directions are then given by

\[
\xi_i^{(\omega)} d\varphi + \xi_i^{(\psi)} d\psi = 0.
\]  
(2.28)
To find the compatibility conditions along these characteristics we need to multiply (2.26) by rows $\omega_i^T$, while also taking into account (2.27).

This procedure yields the following:

1. **Contact characteristics** ($C_0$):

\[
\begin{align*}
\xi_1^{(\psi)} &= \xi_2^{(\psi)} = 0, \\
\xi_{1,2}^{(\psi)} &\text{ may be chosen arbitrarily,} \\
\omega_1^T &= (0, 0, 0, 0, 1), \\
\omega_2^T &= (-F^{(4)}, F^{(3)}, F^{(2)}, -F^{(1)}, 0).
\end{align*}
\]

The compatibility conditions are

\[
\begin{align*}
\frac{\partial D}{\partial \phi} &= 0, \\
\frac{\partial S}{\partial \phi} &= 0.
\end{align*}
\]

These characteristics are the streamlines, that is their direction is given by

\[
d\psi = 0.
\]
2. Normal characteristics \((C_n)\):

\[
\xi_3^{(\psi)} \quad \text{may be chosen arbitrarily},
\]

\[
\xi_3^{(\psi)} = 0,
\]

\[
\omega_3^T = (W^{(4)}, -W^{(3)}, -W^{(2)}, W^{(1)}, 0).
\]

The compatibility condition is

\[
\frac{1}{C} \frac{\partial C}{\partial \psi} - \frac{1}{\kappa \gamma z \bar{S}} \frac{\partial \bar{S}}{\partial \psi} + \frac{2 + \gamma z}{2 \gamma z D} \frac{\partial D}{\partial \psi} = 0. \tag{2.29}
\]

These characteristics are orthogonal to the streamlines.

3. Mach lines: \((C_{\pm})\):

\[
\xi_{4,5}^{(\psi)} = C \bar{s},
\]

\[
\xi_{4,5}^{(\psi)} = \pm B(\psi) \rho,
\]

\[
\omega_{4,5}^T = (s^2 F^{(4)} \mp s F^{(3)}, -s^2 F^{(3)} \mp s F^{(4)}, (1 + \gamma z) F^{(2)} \pm s F^{(1)},
\]

\[
-(1 + \gamma z) F^{(1)} \pm s F^{(2)}, \quad -p \kappa z (2 + \gamma z) J_1 J_2/(2D)
\),

where

\[
s = \sqrt{z - 1}.
\]
This family of characteristic lines only exists when \( z > 1 \).

Let

\[
D_\pm = sC \frac{\partial}{\partial \varphi} \pm B(\psi) \rho \frac{\partial}{\partial \psi},
\]

where \( D_\pm \) is the operator of differentiation along \( C_\pm \). The compatibility conditions can then be written as

\[
s g(z) D_\pm z \mp D_\pm \theta + \frac{s}{\kappa \gamma z S} D_\pm S - \frac{s}{\gamma z D} D_\pm D = 0.
\]

It follows from above that, in the case of a supersonic flow, the governing system (2.13) is hyperbolic. However, unlike the system of Euler equations in Cartesian coordinates, (2.13) has characteristics lines \( C_n \), being produced as a result of the introduction of the unknown \( C \).

In the case of axisymmetric flows, the fluxes in (2.13) depend not only on \( U \), but on \( y \) as well. This makes it necessary to add one more equation to (2.13),

\[
\frac{\partial y}{\partial \varphi} = v J_2,
\]

which follows from (2.7). The derivation of characteristics and compatibility condi-
tions for the resulting extended system may be done in the same manner as in the case of plane geometry. The equation (2.30) then becomes one more compatibility condition imposed along $C_0$.

### 2.2.6 Boundary Conditions

In this Section some problems associated with imposing boundary conditions in the $\varphi - \psi$ plane are addressed. In particular, the boundary conditions for the unknown $C$ are formulated and ways of applying the impermeability boundary conditions are discussed.

**Boundary conditions for the unknown $C$**

The compatibility condition (2.29), which is valid along the characteristics $C_n$ (see Fig. 2.2) shows that $C$ may be specified along $A'B'$. It also follows from (2.29) that,
if $C$ is prescribed strictly positive values along this part of the boundary, then it remains strictly positive in $A'B'C'D'$.

**Impermeability boundary conditions**

The impermeability boundary conditions must be satisfied on the rigid walls. Consider, as an example, the part of the boundary $A'B'$, which is an image of the wall $AB$ (see Fig. 2.1). According to impermeability conditions, in the $x$-$y$ plane the velocity vector must be tangential to the wall. This means that the velocity vector argument is specified as

$$\theta|_{AB} = \theta_0(x) \equiv \arctan \left( \frac{dy}{dz} |_{AB} \right), \quad (2.31)$$

where the derivative $\frac{dy}{dz}$ along $AB$ is known, because the shape of the wall is given.

In order to impose the same conditions along $A'B'$ in the $\phi - \psi$ plane, $\theta$ must be specified in terms of $\phi$. However, the dependence between $x$ and $\phi$ along the wall is not know beforehand. Several ways to overcome this difficulty are outlined below.

**Imposing the boundary conditions iteratively**

If the flow under consideration is transonic and some iterative approach is applied for its calculation, the values from the previous iteration can be used to resolve (2.8). This would give the “updated” boundary conditions to apply on the current
iteration. This procedure should be repeated on each of the following iterations until convergence of the iterative algorithm is achieved.

**Choice of proper boundary values of C**

In Section 2.2.6 it was established that the values of the unknown C can be specified along $A'B'$. If we choose

$$C |_{A'B'} = \frac{1}{q |_{A'B'}},$$

then, taking into account that $udy - vdx = 0$ along $A'B'$, we obtain

$$d\varphi |_{A'B'} = (1 + (\frac{dy}{dx} |_{A'B'})^2)^{\frac{1}{2}}dx \equiv dl,$$

where $l$ is the length of the arc measured along $AB$.

Consequently, we can choose $\varphi = l$ along $AB$, and the boundary conditions can be easily transferred from $AB$ to $A'B'$.

**Choice of $x$ and $y$ as independent variables**

In the case of potential flow, the governing equations reduce to the system of two second order equations with the unknowns $x$ and $y$ (see Section 2.2.4). For this
system, it is necessary to impose two boundary conditions along the rigid wall:

\[ y = y_0(x), \]

\[ \frac{\partial x}{\partial \varphi} \frac{\partial x}{\partial \psi} + \frac{\partial y}{\partial \varphi} \frac{\partial y}{\partial \psi} = 0, \]

where \( y_0 \) is a given wall shape function. The first condition states that the shape of the wall is specified, and the second ensures orthogonality of the lines \( \varphi = \text{const} \) to the wall. Note that, as this formulation uses both \( x \) and \( y \) as unknowns, it is possible to avoid difficulties associated with the unknown relation between \( \varphi \) and \( x \) along the boundary.

**Special types of boundary conditions**

There are certain classes of problems for which the boundary conditions may be applied without encountering difficulties associated with their mapping from \( x-y \) to \( \varphi-\psi \) plane. These are listed below:

1. **Purely supersonic flows**

   In this case the governing equations are hyperbolic, and the solution can be found using marching methods (see Section 3.2, for example). In any of these methods the solution is calculated in one sweep "from left to right" over the computational domain. In the course of this sweep the relation (2.8) can be solved to find \( x \) as a function of \( \varphi \) along the wall and then the boundary con-
ditions (2.31) can be imposed on $A'B'$.

2. *Flows with free surfaces*

Along the free surface, constant pressure or Mach number distribution must be imposed, while the shape of this part of the boundary is to be found from the solution. As this distribution is constant, it does not require a knowledge of the relation between $x$ and $\varphi$.

3. *Homogeneous boundary conditions*

If a part of the boundary is a straight line (the plane or axis of symmetry, for instance), the boundary conditions become $\theta = \text{const}$. As in the previous case, applying this condition does not require the knowledge of $x = x(\varphi)$ along the wall.

4. *Optimal design problems*

In this case, the shape of the boundary in $x-y$ plane is not specified, but it should be found from the solution procedure. This shape should give the optimal value for some functional (drag or lift, for instance), which is calculated using the contour integral along the wall. This integral can be rewritten in $\varphi-\psi$ plane, so that the optimal design problem can be reformulated in streamline coordinates.
Chapter 3

Numerical Solution of Steady Euler Equations in Streamline–Aligned Coordinates

3.1 Introduction

A well–known property of the system of steady Euler equations is that its type changes depending on the value of the local Mach number $M$ of the flow. Accordingly, different numerical methodologies need to be employed to discretize and to solve the governing equations depending on whether the flow in the domain under consideration is purely subsonic ($M < 1$ everywhere and the governing equations are of elliptic type), purely supersonic ($M > 1$ everywhere, hyperbolic governing equations) or mixed ($M$ assumes...
values greater and less than one, governing equations of mixed type).

In order to compute purely supersonic flows, a number of methods have been developed for the solution of the governing system of hyperbolic equations. Depending on the underlying methodology used to discretize the equations, these methods can typically be split into the following classes [59]:

- the Lax–Wendroff family of space–centered schemes;
- central schemes with independent time integration;
- upwind schemes, including second–order upwind and high–resolution schemes.

Most of the schemes currently utilized are based upon various refinements and extensions of the upwinding idea (e.g. multi–dimensional flux–vector splitting, or Godunov–type schemes).

The original idea of flux-splitting schemes dates back to 1981 [53], where the flux terms are split according to the sign of the eigenvalues of the flux Jacobian matrix. The disadvantage of this approach is that the resulting expressions for numerical fluxes are not smooth at the sonic point. This results in the possibility of occurrence of a non–smooth solution near the sonic point. A flux–splitting scheme with continuously differentiable expressions for numerical fluxes was developed in [54]. A survey of modern developments in flux–splitting methodology can be found in [61].

The original idea of Godunov-type schemes was suggested in [55] based on the idea of an exact solution of the Riemann problem in order to compute the numerical
fluxes. Because of the complexity of exact Riemann solvers, a simpler approach, which replaces the local Riemann problem with its linearized analogue, preserving the direction of the contact discontinuity, was developed [56]. Both exact and approximate Riemann solvers remain popular among the researchers and CFD code developers [21, 44].

One of the possibilities to adequately represent hyperbolicity of the problem in its discretization is based upon considering the characteristic relations of the governing equations and taking them into account when constructing a numerical approximation on the mesh consisting of characteristic lines (characteristic schemes) or, alternatively, on a regular pre-defined (in computational plane) mesh (grid-characteristic schemes). The latter approach is used in this chapter.

Because of the hyperbolic nature of the discretized problem, the numerical solution algorithms are typically marching (ie non-iterative) schemes. Solution algorithms for discretizations of elliptic or mixed boundary-value problems, corresponding to sub- or transonic steady flows, typically require iterations.

Also in this chapter, subsonic and transonic flows are treated under the assumptions of irrotationality of the velocity vector field and constant entropy, which are valid in the absence of strong shocks in the flowfield. Vortical and non-isentropic flows are treated in the next chapter.

A number of different approaches have been used in order to discretize the full-potential (or equivalent) equation and to solve it iteratively [50–52, 59]. The most
common approaches to handle the difficulties arising in numerical approximation of the full potential equation in the transonic case can be categorized as follows [59]:

- use of artificial viscosity in non-conservative or conservative form of full-potential equation;
- an artificial density approach (i.e. "density upwinding");
- an artificial flux or flux upwinding.

In the following sections a hybrid grid-characteristic scheme is constructed and used to compute supersonic flows, while the artificial density approach and approximate-factorization iterative technique are used to calculate the parameters of transonic and subsonic flows.

3.2 A Conservative Hybrid Grid–Characteristics Scheme

3.2.1 Conservative Approximation of the Governing System

Consider purely supersonic flow. In this case the governing system (2.13) becomes hyperbolic and it can be resolved using a marching scheme.

To construct the difference approximation, consider the finite volume $D_{n+1/2,m}$ shown on Figure 3.1 with a dashed line. Writing (2.12) for this volume gives
Approximation of the above relation on the stencil which consists of the nodes \((n, m - 1), (n, m), (n + 1, m)\) and \((n + 1, m + 1)\) (see Fig. 3.1) leads to

\[
\frac{W_{m}^{n+1} - W_{m}^{n}}{h_{\varphi}} + \frac{F_{m+1/2}^{n+1/2} - F_{m-1/2}^{n+1/2}}{h_{\psi}} = \nu H_{m}^{n+1/2}.
\] (3.1)

As the chosen stencil has two nodes on the upper layer, it is suitable for the approximation of the hyperbolic system (2.13), which has a family of characteristics \(\varphi = \text{const}\).

Since the vector of primitive variables \(U\) and the fluxes \(W\) and \(F\) are stored at
the nodes of the grid, it is necessary to interpolate these values to calculate $F_{m-1/2}^{n+1/2}$ and $F_{m+1/2}^{n+1/2}$. Consider the following interpolation procedure:

\begin{align*}
F_{m+1/2}^{n+1/2} &= R_{m+1/2} F_{m+1}^{n+1} + (I - R_{m+1/2}) F_m^n, \\
F_{m-1/2}^{n+1/2} &= R_{m-1/2} F_m^{n+1} + (I - R_{m-1/2}) F_{m-1}^n, \\
H_m^{n+1/2} &= (H_m^n + H_m^{n+1})/2,
\end{align*}

where $R_{m\pm 1/2}$ are interpolation matrices. To properly choose these matrices, Riemann invariants and characteristics of the linearized system must be examined.

### 3.2.2 Linearized System of Equations

For a finite volume $D_{n+1/2,m}$, consider the local linearization of the system (2.13):

\begin{equation}
A_0^{(\phi)} \frac{\partial U}{\partial \varphi} + A_0^{(\psi)} \frac{\partial U}{\partial \psi} = \nu H_0,
\end{equation}

where the Jacobian matrices $A_0^{(\phi)}$ and $A_0^{(\psi)}$ are calculated at the node $(n,m)$. As in Section 2.2.5, $\xi_i^{(\phi)}$, $\xi_i^{(\psi)}$ and $\omega_i$ denote the eigenvalues and left eigenvectors of the characteristic equation (2.27), respectively.

For the linear system (3.4), introduce the Riemann invariants $u_i$ given by

40
\[ u_i = \omega_i^T A_0^{(p)} U. \]

The compatibility condition along the respective characteristic line then becomes a linear advection equation:

\[ \xi_i^{(p)} \frac{\partial u_i}{\partial \varphi} - \xi_i^{(p)} \frac{\partial u_i}{\partial \psi} = r_i, \]

where

\[ r_i = \omega_i^T H_0. \]
For the four-node stencil being considered there exists a one-parameter family of finite difference equations, approximating the above advection equation [46],

\[
(2 - \sigma_i)(u_i)^{n+1}_m - \alpha^*(2 - \sigma_i)(u_i)^n_m - (1 - \alpha^*(1 - \sigma_i))(u_i)^n_{m-1} - (1 - \sigma_i - \alpha^*)(u_i)^{n+1}_{m+1} - (r_i)^n_m = 0,
\]

(3.5)

where

\[
\sigma_i = \frac{h_{\psi} \xi^{(\psi)}_i}{h_{\psi} \xi^{(\psi)}_i}
\]

is a CFL number.

Each value of the parameter \(\alpha^*\) gives a scheme which approximates the advection equation with at least first order accuracy. Consider Fig. 3.2. The following can be proved [46]:

1. The dashed line \(\alpha^* = 1\) corresponds to the schemes with at least second order accuracy;

2. Shaded area corresponds to the schemes with "positive approximation" [46], that is, the schemes with positive coefficients. This is sufficient for their monotonicity;
3. The coefficient of the effective (artificial) viscosity generated by the scheme, is proportional to the distance between the point corresponding to this scheme on the diagram and the dashed line $\alpha^* = 1$;

4. The stability condition for this family of approximations is $\sigma_i < 1$.

Therefore, for a given value of $\sigma_i = \sigma_{i_0} < 1$, we have the second order scheme $A_2$ (which is not monotone, unless $\sigma_{i_0} = 0$), as well as the first order monotone scheme $A_1$ located on the boundary of the shaded area (see Fig. 3.2). It follows that, for a given value of $\sigma_{i_0}$, the scheme $A_1$ has the minimal coefficient of the effective viscosity among all monotone schemes. Hence, the scheme $A_1$ possesses minimal smearing properties among those monotone schemes. The family of schemes which is of primary interest for us is then represented by the chord $A_1A_2$. One can choose a scheme from this family closer to $A_2$ to obtain better resolution at the price of possible non-monotonicity. In a similar way, a shift towards $A_1$ results in the damping of oscillations at the price of lesser local accuracy.

To apply this result to the linearized system (3.4), rewrite (3.1) as

$$\frac{(W_0)^{n+1}_m - (W_0)^n_m}{h_\psi} + \frac{(F_0)^{n+1/2}_m - (F_0)^{n+1/2}_{m-1/2}}{h_\psi} = \nu (H_0)^{n+1/2}_m$$ \hspace{2cm} (3.6)$$

where
\[
W_0 = A_0^{(\psi)}U,
\]
\[
F_0 = A_0^{(\psi)}U.
\]

To calculate the fluxes \((F_0)^{n+1/2}\), the interpolation procedure (3.2) and (3.3) is used. However, for the linear equation case, we can put \(R_{m-1/2} = R_{m+1/2} = R\), where \(R\) is a square matrix with constant entries.

Choose this matrix to be

\[
R = \Omega^{-1}\tilde{R}\Omega
\]

where \(\Omega\) is a matrix composed of the rows \(\omega_i^T\), and \(\tilde{R}\) is a diagonal matrix with diagonal entries \(\tilde{R}_{ii}\).

Multiplying (3.6) by \(\omega_i^T\) yields

\[
(1 - \sigma_i\tilde{R}_{ii})(u_i)^{n+1}_n - (1 - \sigma_i + \sigma_i\tilde{R}_{ii})(u_i)_m^n + (\sigma_i - \sigma_i\tilde{R}_{ii})(u_i)^{n-1}_m
+ \sigma_i\tilde{R}_{ii}(u_i)^{n+1}_{m+1} - \nu(1 - \sigma_i\tilde{R}_{ii})h_\psi h_\psi r^n_m = 0. \tag{3.8}
\]

Comparison between the expressions (3.5) and (3.8) gives the link between the
value of $\tilde{R}_{ii}$ and the parameter $\alpha^*$ from (3.5), i.e.,

$$
\tilde{R}_{ii} = \frac{\sigma_i + \alpha^* - 1}{(1 + \alpha^*)\sigma_i}.
$$

It follows from this relation and Fig. 3.2 that setting

$$
\tilde{R}_{ii} = \frac{1}{2}
$$

will result in the second order approximation, while

$$
\tilde{R}_{ii} = \frac{1}{2} - \frac{\text{Sgn}(\sigma_i)}{2}
$$

gives the first order monotone scheme with minimal effective viscosity coefficient.

To obtain the hybrid scheme [46] set

$$
\tilde{R}_{ii} = \frac{1}{2} - \beta \frac{\text{Sgn}(\sigma_i)}{2},
$$

(3.9)

where the parameter $\beta$ satisfies $0 \leq \beta \leq 1$. In accordance with the discussion above, the parameter $\beta$ should be chosen close to zero in the regions where the solution is
smooth enough and close to 1 in the regions where the solution has large gradients (eg., near shocks). The algorithm used in this work to choose the values of $\beta$ in accordance with the local behaviour of the solution is taken from [46, p.108].

3.2.3 Generalization to the Nonlinear Case

To generalise the idea of hybrid scheme to the nonlinear case, choose the matrices $R_{m\pm 1/2}$ in (3.2) and (3.3) in accordance with (3.7). Then (3.1), which gives the approximation for the finite volume $D_{m+1/2}^{n+1}$, can be rewritten as

$$
W_{m}^{n+1} - \frac{h_\psi}{h_\phi} \Omega_{m-1/2}^{-1} \tilde{R}_{m-1/2} \Omega_{m-1/2} F_{m}^{n+1} =
$$

$$\nu h_\psi H_m^{n+1/2} + W_m^n - \frac{h_\psi}{h_\phi} (F_m^n - F_{m-1}^n)$$

$$- \frac{h_\psi}{h_\phi} \Omega_{m+1/2}^{-1} \tilde{R}_{m+1/2} \Omega_{m+1/2} (F_{m+1}^{n+1} - F_m^n)$$

$$- \frac{h_\psi}{h_\phi} \Omega_{m-1/2}^{-1} \tilde{R}_{m-1/2} \Omega_{m-1/2} F_m^{n}. $$

At the current step of the computational algorithm, the right-hand side is known from the previous step. The left-hand side is then solved by Newton iterative technique with respect to $U_m^{n+1}$. At the next step, the same procedure is repeated for the equation corresponding to the finite volume $D_{m-1}^{n+1/2}$ to find $U_{m-1}^{n+1}$.

It should be noted that, although the governing system has a family of characteristic lines $\varphi = const$, normal to streamlines, the stable solution process can be
implemented by marching in the direction of streamlines. This is due to the fact that the computational stencil has two nodes on the upper layer, i.e. the scheme is implicit. At the same time, as there are only two nodes on the upper level, the resulting solution procedure does not require global iterations in order to find the values of $U^{n+1}$.

3.3 An Iterative Algorithm for the Full-Potential Equivalent Equation

In the case of transonic potential flow with constant entropy and full enthalpy the function $C$ from (2.13) can be set equal to any positive function of $\varphi$ and then the governing system can be reduced to just one second order partial differential equation of a mixed type (see Section 2.2.4 for details):

\[
\frac{\partial}{\partial \varphi} \left( f \frac{\partial y}{\partial \varphi} \right) + \frac{\partial}{\partial \psi} \left( \frac{1}{f} \frac{\partial y}{\partial \psi} \right) = 0,
\]

where

\[
f = \frac{J_1}{J_2} = \frac{C(\varphi)}{B(\psi) y'' \rho},
\]

\[
\rho = \rho(x).
\]
This second order equation for \( y \) may be considered as the equivalent to the full-potential equation. In this case, the difference approximations and the numerical algorithms developed for the full-potential equation may be applied.

The approximating system of finite difference equations is obtained using central difference approximation formulae and the concept of artificial density in supersonic regions:

\[
\Delta_\phi^+ \tilde{f} \Delta_\phi^- y + \Delta_\phi^+ \frac{1}{f} \Delta_\phi^- y = 0, \tag{3.10}
\]

where

\[
\tilde{f}_{ij} = \mu_{ij} f_{ij} + (1 - \mu_{ij}) f_{ij-1},
\]
\[
\mu_{ij} = \min(1, \max(0, \eta(M^2 - 1))),
\]

and \( \eta \) is a positive parameter.

To resolve the system of equations (3.10), the approximate factorization iterative technique has been applied to obtain

\[
(1 - \tau^* \Delta_\phi^+ \tilde{f}^{(n)}) (\Delta_\phi^- - \tau^* \Delta_\phi^+ \frac{1}{f^{(n)}} \Delta_\phi^-) \varepsilon^{(n)} = \tau^* \omega R(y^{(n)}), \tag{3.11}
\]

48
where $\varepsilon^{(n)} = y^{(n+1)} - y^{(n)}$ is the correction vector, $R(y^{(n)})$ is the residual vector of the equation (3.10) and $\tau^*$, $\omega$ are the iterative parameters.

The operator on the left-hand side of (3.11) can be inverted in two sweeps over the computational domain. This kind of approximate factorization has been successfully applied earlier to the solution of the velocity vector potential equation in [47].

It was established in Section 2.2.4, that the parameters of the flow $z$ and $\theta$ are given implicitly as functions of the unknown function $y$ and its derivatives by the relations

$$
\frac{u}{pq^2} = B(\psi) y' \frac{\partial y}{\partial \psi},
$$

$$
\frac{v}{q^2} = C(\varphi) \frac{\partial y}{\partial \varphi}.
$$

However, in the transonic range, the difficulty of non-uniqueness of the solution of the above system exists (see Section 2.2.4). In order to overcome this difficulty, instead of solving the system of algebraic relations (2.20) and (2.21), the following partial differential equation for the unknown $z$ is solved in transonic regions [41]

$$
\frac{\partial}{\partial \varphi} \left[ (1 - z) b \frac{\partial z}{\partial \varphi} \right] + \frac{\partial}{\partial \psi} \left[ a \frac{\partial z}{\partial \psi} \right] = - \frac{\partial c}{\partial \varphi}
$$

where
\[ a = \frac{g(z)}{f}, \quad b = fg(z), \]

\[ c = \nu \frac{v}{y^2 \rho q^2 B(\psi)}, \quad g(z) = \frac{d}{dz} \ln q. \]

This equation is solved only along those grid lines \( \varphi = \text{const} \) which have supersonic points. Following [42], the coefficients of the above equation are calculated from the previous iteration, and then the SOR iterative algorithm is applied to the resulting system of linear equations.

### 3.4 An Iterative Algorithm for the System of Two Equations

In this Section we consider an iterative algorithm, based upon the formulation of the governing system in the potential flow case which has been developed in Section 2.2.4. A disadvantage of the formulation proposed in the previous Section is that it is necessary to apply the additional equation for the unknown \( z \) in transonic regions. Unlike the second order equation for \( y \), the equation for the unknown \( z \) does not express conservation of any physical quantity (such as mass, momentum or energy). Therefore, this equation can not be applied in the vicinity of the discontinuities in the flow.

In Section 2.2.4 the governing system was reduced to the system of two equations
\[ \frac{\partial W^{(p)}}{\partial \varphi} + \frac{\partial F^{(p)}}{\partial \psi} = 0, \]

where fluxes \( W^{(p)} \) and \( F^{(p)} \) are expressed in terms of derivatives of \( x \) and \( y \) as

\[
W^{(p)} = \begin{bmatrix} f \frac{\partial x}{\partial \varphi} \\ f \frac{\partial y}{\partial \varphi} \end{bmatrix}, \quad F^{(p)} = \begin{bmatrix} \frac{1}{f} \frac{\partial x}{\partial \psi} \\ \frac{1}{f} \frac{\partial y}{\partial \psi} \end{bmatrix}.
\]

It has been previously established in Section 2.2.4, that if the unknowns \( x \) and \( y \) are known as functions of \( \varphi \) and \( \psi \), then the parameters of the flow \( z \) and \( \theta \) can be found in a unique way by solving the algebraic equations (2.24) and (2.25).

The approximation of (2.19) is performed using central differences and artificial compressibility:

\[
(W^{(p)}_{i,j+1/2} - W^{(p)}_{i,j-1/2})/h_\varphi + (F^{(p)}_{i+1/2,j} - F^{(p)}_{i-1/2,j})/h_\psi = 0, \tag{3.12}
\]

where

\[
W^{(p)}_{i,j+1/2} = \begin{bmatrix} \bar{f}_{i,j+1/2} \frac{x_{i+1} - x_{i,j}}{h_\varphi} \\ \bar{f}_{i,j+1/2} \frac{y_{i+1} - y_{i,j}}{h_\varphi} \end{bmatrix}, \quad F^{(p)}_{i+1/2,j} = \begin{bmatrix} \frac{1}{f_{i+1/2,j}} \frac{x_{i+1} - x_{i,j}}{h_\psi} \\ \frac{1}{f_{i+1/2,j}} \frac{y_{i+1} - y_{i,j}}{h_\psi} \end{bmatrix},
\]

and

51
\[
\begin{align*}
\tilde{f}_{ij+1/2} &= \frac{C(\psi)}{B(\psi)v_{ij+1/2}^\sigma \tilde{\sigma}_{ij+1/2}}, \\
\tilde{\rho}_{ij+1/2} &= \rho_{ij+1/2} + \mu_{ij+1/2}(\rho_{ij-1/2} - \rho_{ij+1/2}), \\
\mu_{ij+1/2} &= \max(0, 1 - \frac{1}{M_{ij+1/2}}).
\end{align*}
\]

To resolve the system \((3.12)\) the following iterative procedure is applied. First, using the values of \(x\) and \(y\) from previous iteration, the system \((2.24)\) and \((2.25)\) is solved by Newton iterations to find \(x\) and \(\theta\). The new values of \(x\) are used to update the values of the coefficient \(f\). Substituting these values into \((3.12)\) gives the linearized system for \(x\) and \(y\).

The resulting system of linear equations is solved by point Gauss–Seidel method based upon the natural ordering of the unknowns.

The convergence of iterations is accelerated by employing two-level multigrid cycles. The description of the multigrid procedure used in the calculations follows.

Consider a pair of rectangular uniform grids in the \(\varphi - \psi\) plane. The first grid is referred to as a fine grid. The step sizes in \(\varphi\) and \(\psi\) directions for this grid are \(h_\varphi\) and \(h_\psi\). The second grid, a coarse grid, is generated by skipping alternate nodes within the fine grid. Similar to the first grid, \(H_\varphi\), \(H_\psi\) denote step sizes in \(\varphi\) and \(\psi\) directions for the coarse grid.

Define operators of restriction \(R_h^H\) and prolongation \(P_h^H\) to be used to pass the information between the grids. Namely, the restriction operator \(R_h^H\) acts on a grid

\[52\]
function defined on a fine grid and produces a restriction of this grid function on a coarse grid. In this work, this restriction is accomplished by ignoring the values of the grid function at those nodes of the fine grid which do not correspond to nodes of the coarse grid. The prolongation operator $P^h_H$ interpolates the data from the coarse grid to the fine grid. In this work, linear interpolation is used for this purpose.

Denote the residual vector of the system of difference equations (3.12) computed on the fine grid by $N_h$. Also, denote the solution vector composed of the values of $x$ and $y$ at the grid nodes of the fine grid by $x_h$.

The multigrid algorithm used in this work can then be described as follows:

1. Using an initial guess on a fine grid, or data from a previous multigrid cycle, perform $k_h$ point Gauss–Seidel (PGS) iterations for the linearized system of difference equations (3.12):

$$G_h e_h^{(n)} = N_h^{(n)},$$

$$x_h^{(n+1)} = x_h^{(n)} + e_h^{(n)},$$

where $G_h$ is an iteration matrix corresponding to the PGS method, $e_h^{(n)}$ is a correction vector, superscripts serve to distinguish the quantities referring to the previous and the next iteration, while subscripts show that all the above quantities are calculated on the fine grid. After each iteration the coefficient $f$ needs to be evaluated to obtain an updated linearization.
2. Perform $k_H$ Gauss-Seidel iterations on the coarse grid as described below.

2.1 Using the residual on the fine grid, which is known either from step 1, or from the previous iteration on the coarse grid, compute the correction vector for the coarse grid $e_H^{(n)}$:

$$ G_H e_H^{(n)} = R_H^H N_h^{(n)}. $$

2.2 Update the solution on the fine grid, using $e_H^{(n)}$ and the prolongation operator $P_H^h$:

$$ x_h^{(n+1)} = x_h^{(n)} + P_H^h e_H^{(n)} $$

2.3 Using $x_h^{(n+1)}$, compute $R_H^H N_h^{(n+1)}$. Notice that, for the choice of restriction operator used in this work, this stage simply requires the calculation of the residual at alternate nodes of the fine grid.

2.4 Go to the step 2.1.

3. Go to the step 1.

Interpolating between the grids in a described way allows damping of both the high- and low-frequency components of the error in a more efficient way than on a single grid. Also, typically it is possible to carry the iterations mostly on the coarse grid, where the low-frequency components of the error can be removed using fewer
arithmetic operations. Switching to the fine grid could be done periodically when the high-frequency components become dominant.

3.5 Numerical Results

In this section results are presented for a series of supersonic and transonic flows calculated using methods which have been described above.

To demonstrate the properties of the hybrid scheme, the standard test problem of a supersonic flow \(M_\infty = 2.0\) over the wedge \(\delta = 15^\circ\) was solved using the scheme described in Section 3.2. Fig. 3.3 shows Mach number plotted against the coordinate \(\psi\) directed across the flow. These calculations were performed using first order monotone \((\beta = 1\) in (3.9)), second order \((\beta = 0)\) and hybrid schemes. In the case of the hybrid scheme the values of the parameter \(\beta\) at each node were chosen depending on the local behaviour of the solution as described in [46]. The comparison between these solutions shows that the first order monotone scheme smears the shock over 5 – 7 nodes; the second order scheme gives width of the shock equal to 3 – 4 nodes but also produces non-physical oscillations in the solution. The solution given by the hybrid scheme has the shock width of 4 – 5 nodes and it remains monotone.

Fig. 3.4 shows the pressure and entropy function contours as well as the streamline computational grid in the supersonic part of the nozzle and in the jet emerging from it. This flow was computed using the hybrid scheme given in Section 3.2. In this calculation, constant pressure was imposed along the free boundary which is a
streamline segment. This constant value of pressure was taken to be equal to its value at the end of the nozzle wall.

In Fig. 3.5 and Fig. 3.6, application of the iterative technique outlined in Section 3.3 is illustrated for the calculation of the transonic potential gas flow in a converging–diverging nozzle, which has been studied experimentally in [48]. The comparison of the computed pressure and Mach number distribution along the wall with the experimental data shows good agreement except in the region downstream of the nozzle throat. This loss of accuracy may be ascribed to the presence of the shock in this region resulting in non–zero vorticity in real flow and hence inappropriateness of the potential flow model.

Finally, Fig. 3.7 and Fig. 3.8 show the results obtained using the formulation and the iterative algorithm described in Section 3.4. The subsonic flow through a bumpy axisymmetric channel was calculated. Fig. 3.7 demonstrates the influence of the number of iterations at each grid level in the two–grid cycle on the convergence rate, and Fig. 3.8 shows the solution.
Figure 3.3: Mach number vs psi coordinate.
Figure 3.4: Jet emerging from supersonic nozzle.
Figure 3.5: A transonic flow in a nozzle.
Figure 3.6: Pressure and Mach number distributions along the wall.
Figure 3.7: Influence of the number of iterations on each grid level on the convergence rate.

Fine Grid (FG) - 31 x 61
Coarse Grid (CG) - 16 x 31

1 - iterations on a FG only,
2 - 1 iteration on a FG followed by 1 iteration on a CG,
3 - 1 iteration on a FG followed by 2 iterations on a CG,
4 - 1 iteration on a FG followed by 4 iterations on a CG.
Figure 3.8: Subsonic flowfield in the axisymmetric bumpy channel.
Chapter 4

A Streamline–Adaptive Formulation for Unsteady Euler Equations

4.1 Introduction

Formulating the governing equations of gas motion in a system of independent coordinates aligned with the streamlines provides a number of advantages in CFD problems.

In the case of steady 2D motion, the streamfunction \( \psi \) of the flow is defined by the relation

\[
\psi = \rho u dy - \rho v dx. \quad (4.1)
\]
This function can be chosen to be one of the new independent coordinates. With this choice, the boundaries of the domain on which the impermeability condition is imposed are automatically mapped into straight lines regardless of their actual geometry in the plane of Cartesian coordinates. This eliminates the necessity of the numerical grid generation procedure, thus resulting in reduction in both computation time and memory requirements.

In addition, the “streamfunction-as-a-coordinate” (SFC) approach has further advantages over the conventional techniques utilizing a priori grid generation in problems involving free or elastic boundaries and problems of inverse or optimal design of aerodynamical shapes. Since, in such problems, the shape of some boundary segment is not known in advance, the conventional approach would require generation of the grid at each step of an iterative procedure, followed by interpolation of the values of unknown functions from the old grid to the new one. In addition to being computationally expensive, there is a loss of accuracy due to the interpolations. These undesirable features of the conventional approach are not present in the SFC based techniques, because the unknown segment of the boundary is a streamline which is represented by a known straight line segment in the plane of the new coordinates. These advantages of the SFC methodology have been widely exploited in the numerical solution of steady CFD problems [6, 11, 16, 21, 23, 27, 28, 41, 49].

Thus far, research utilizing the SFC concept has been limited to steady state calculations. The primary reason for this limitation is that the relation (4.1) defines the
unique-valued function $\psi$ only when the steady form of continuity equation holds. In
the present work, the SFC concept is extended to the non-steady Euler equations.
This new extension is more general than previous formulations and it allows one to
control the grid in order to adapt it to the solution at the current time step. In par-
ticular, the 2D Euler equations are formulated in an arbitrary system of independent
coordinates described by its metric coefficients. Analysis shows that some of these
coefficients can be prescribed in advance. By appropriately choosing the values of
the coefficients to be prescribed, we can ensure that one family of coordinate lines is
aligned along the velocity vector field, and that the resulting grid can be clustered
in locations where large gradients are expected. The remaining metric coefficients,
the values of which are not prescribed, have to satisfy certain compatibility relations.
These relations, together with the Euler equations, form an extended system of partial
differential equations to be solved for certain physical and geometrical unknowns. In
order to obtain consistent boundary conditions for the unknown metric coefficients,
the characteristic properties of the extended system are analyzed.
4.2 Streamline–Aligned Coordinate System

Consider a nondegenerate transformation from the physical space of time $t$ and Cartesian coordinates $x$ and $y$ to the computational space of variables $\tau, \varphi, \psi$:

$$
\begin{align*}
\tau &= t, \\
\varphi &= \varphi(t, x, y), \\
\psi &= \psi(t, x, y),
\end{align*}
\iff
\begin{align*}
t &= \tau, \\
x &= x(\tau, \varphi, \psi), \\
y &= y(\tau, \varphi, \psi).
\end{align*}
\tag{4.2}
$$

Define the following quantities associated with this transformation:

$$
\begin{align*}
E &= x_\varphi^2 + y_\varphi^2, \\
J &= x_\varphi y_\psi - y_\varphi x_\psi, \\
\alpha &= \arctan\left(\frac{y_\varphi}{x_\varphi}\right), \\
G &= x_\varphi^2 + y_\varphi^2, \\
F &= x_\varphi x_\psi + y_\varphi y_\psi.
\end{align*}
\tag{4.3}
$$

These quantities characterize the local properties of the computational grid generated by the transformation (4.2). To be more precise, consider an infinitesimally small rectangle in the plane $\tau = \tau_0$ with sides of lengths $d\varphi$ and $d\psi$ which are parallel to $\varphi$ and $\psi$ axes respectively. Under the mapping (4.2) this rectangle transforms into an infinitesimal parallelogram in the plane $t = \tau_0$. The lengths of the sides of this parallelogram will be respectively $\sqrt{G}d\varphi$ and $\sqrt{E}d\psi$, its area will be equal to $Jd\varphi d\psi$, the angle between the side with length $\sqrt{G}d\varphi$ and $x$-axis will be equal to $\alpha$ and cosine of the angle between the sides of this parallelogram will be $F/\sqrt{EG}$.

It is convenient to group the first order partial derivatives of Cartesian coordinates into three columns of length two by introducing the notations: $t = [x_\tau, y_\tau]^T$, $w = \ldots$
Figure 4.1: Local geometry of the grid in terms of the metric coefficients.

$[x_\psi, y_\psi]^T$ and $f = [x_\phi, y_\phi]^T$. The relations (4.3) can be resolved with respect to the partial derivatives by $\phi$ and $\psi$ to give:

$$ w = \frac{F}{\sqrt{G}} \begin{bmatrix} \cos \alpha \\ \sin \alpha \end{bmatrix} - \frac{J}{\sqrt{G}} \begin{bmatrix} \sin \alpha \\ -\cos \alpha \end{bmatrix}, \quad (4.4) $$

$$ f = \sqrt{G} \begin{bmatrix} \cos \alpha \\ \sin \alpha \end{bmatrix}. \quad (4.5) $$

Requiring that cross derivatives of $t$, $x$ and $y$ with respect to $\tau$, $\phi$ and $\psi$ are equal, $t$, $w$ and $f$ must satisfy the relations

$$ w_\psi - f_\phi = 0, \quad (4.6) $$

$$ w_\tau - t_\psi = 0, \quad (4.7) $$

$$ f_\tau - t_\phi = 0. \quad (4.8) $$

It can be seen from (4.6)–(4.8) that only four of these six equations are independent.
Suppose, for example, (4.6) and (4.7) are satisfied everywhere. Differentiating these relations by $\tau$ and $\varphi$ respectively and subtracting the first of the resulting relations from the second, one obtains (4.8) differentiated by $\psi$. Therefore if, in addition to (4.6) and (4.7) being valid everywhere, (4.8) is satisfied on some plane $\psi = \text{const}$, then (4.6)–(4.8) will be satisfied everywhere.

In order for the grid to be aligned with the streamlines of the flow, $\alpha$ in (4.3) must be chosen equal to the angle between the velocity vector and $x$-axis. Then, one is left with two possible alternative ways to control the geometry of the grid. One option is to choose Jacobian $J$ in (4.3) to be some known function of $x$ and $y$ as in hyperbolic grid generation. By choosing this function properly, one can adjust the size of the cells of the grid to the expected behaviour of the solution. Another option is to enforce orthogonality of the grid in Cartesian coordinates by choosing $F = 0$ everywhere. In both of these cases the second quantity, either $F$ or $J$, can be found from the relation $F^2 + J^2 = EG$. The remaining four quantities in (4.3), namely $E$ and $G$ and two components of $t$ cannot be specified arbitrarily because of the restrictions imposed by equations (4.6)–(4.8). These restrictions consist of four partial differential equations (4.6) and (4.7) and the condition (4.8) which need to be satisfied on some plane $\psi = \text{const}$. After the solution to this system is found, the
coordinate transformation (4.2) can be calculated from:

\[
\mathbf{r} = \left[ \begin{array}{c} x(\tau, \varphi, \psi) \\ y(\tau, \varphi, \psi) \end{array} \right] = \int_{(\tau_0, \varphi_0, \psi_0)}^{(\tau, \varphi, \psi)} t \, d\tau' + f \, d\varphi' + w \, d\psi',
\]

(4.9)

This integral is path-independent because of (4.6)–(4.8).

4.3 Governing Equations in Streamline-Aligned Coordinate System

Consider the following system of \( n \) partial differential equations in conservative form:

\[
a_t + b_x + c_y = 0,
\]

(4.10)

where \( a = a(U), b = b(U), c = c(U) \) are known functions of the vector of unknown primitive variables \( U \) of length \( n \). The system of Euler equations can be written in the form (4.10). In the following discussion, the general system (4.10) will be treated until it becomes necessary to resort to the properties specific to the system of Euler equations.

Integrating (4.10) over an arbitrary domain \( \Omega(t, x, y) \) in the space of coordinates
\[ \int_{\partial \Omega(t,x,y)} a \, dxdy + b \, dydt + c \, dtdx = 0. \]  

(4.11)

To change independent variables in (4.11), apply the relationship between infinitesimal surface elements known from differential geometry, i.e.

\[
\begin{bmatrix}
  dxdy \\
  dydt \\
  dtdx
\end{bmatrix} =
\begin{bmatrix}
  \frac{\partial (x,y)}{\partial (\varphi,\psi)} & \frac{\partial (x,y)}{\partial (\psi,\tau)} & \frac{\partial (x,y)}{\partial (\tau,\varphi)} \\
  \frac{\partial (y,t)}{\partial (\varphi,\psi)} & \frac{\partial (y,t)}{\partial (\psi,\tau)} & \frac{\partial (y,t)}{\partial (\tau,\varphi)} \\
  \frac{\partial (t,x)}{\partial (\varphi,\psi)} & \frac{\partial (t,x)}{\partial (\psi,\tau)} & \frac{\partial (t,x)}{\partial (\tau,\varphi)}
\end{bmatrix}
\begin{bmatrix}
  d\varphi d\psi \\
  d\psi d\tau \\
  d\tau d\varphi
\end{bmatrix}.
\]

(4.12)

Using these relations in (4.11) and taking into account (4.4) and (4.5), the integral conservation formulation of (4.10) in \((\tau, \varphi, \psi)\) space is

\[
\int_{\partial \Omega(\tau,\varphi,\psi)} A \, d\varphi d\psi + B \, d\psi d\tau + C \, d\tau d\varphi = 0,
\]

(4.13)

where \(\Omega(\tau, \varphi, \psi)\) is the image of the domain \(\Omega(t,x,y)\) and

\[
A = J a, \quad (4.14)
\]

\[
B = S w, \quad (4.15)
\]

\[
C = -S f, \quad (4.16)
\]

\[
S = [-c + y_r a, b - x_r a] \quad \text{is an} \ n \times 2 \ \text{matrix.}
\]
Derivations of the relations (4.14)–(4.16) can be found in Section 4.5.1.

Since (4.13) is valid for any domain \( \Omega (\tau, \varphi, \psi) \), it is equivalent to the system of differential conservation laws

\[
A \tau + B \varphi + C \psi = 0, \quad (4.17)
\]

which is the governing system (4.10) written in the \((\tau, \varphi, \psi)\) coordinates. This system needs to be considered together with equations (4.6) and (4.7) which ensure that the coordinate transformation does not degenerate. These equations are combined together to obtain the extended governing system

\[
V \tau + W \varphi + F \psi = 0, \quad (4.18)
\]

where

\[
V(\varphi, \psi, Q) = \begin{bmatrix}
A \\
0 \\
w
\end{bmatrix}, \quad W(\varphi, \psi, Q) = \begin{bmatrix}
B \\
w \\
0
\end{bmatrix}, \quad F(\varphi, \psi, Q) = \begin{bmatrix}
c \\
-\tau
\end{bmatrix}, \quad Q = \begin{bmatrix}
U \\
u
\end{bmatrix},
\]

and \( u = [E, G]^T \).

It can be seen from (4.4), (4.5) and (4.14)–(4.16) that \( w, f, A, B \) and \( C \) are known functions of the vector of primitive variables \( U \), the "geometrical" unknown functions \( t \) and \( u \) and independent coordinates \( \varphi \) and \( \psi \). The explicit dependence
upon independent coordinates occurs only in the case when \( J \) is assumed to be a known function of independent variables \( \varphi \) and \( \psi \) in order to cluster or repel gridlines. The vectors \( \mathbf{w} \) and \( \mathbf{f} \) depend on the primitive variables \( \mathbf{U} \) because in order to align the lines \( \psi = \text{const} \) along the streamlines it is necessary to choose \( \alpha \) in (4.4) and (4.5) to be equal to the angle between the velocity vector and \( x \)-axis. System (4.18) must be solved subject to boundary conditions following from those for system (4.10) and boundary condition (4.8) imposed along one of the bounding surfaces \( \psi = \text{const} \). Also, additional boundary conditions must be imposed for the "geometrical" unknowns \( t \) and \( u \). These boundary conditions are discussed in the next subsection.

4.4 Characteristic Directions of the Extended Governing System

The governing system in Cartesian coordinates (4.10) can be considered as the system describing an evolution of some process in time. Such systems are often hyperbolic, as is the system of Euler equations. Consider the following eigenproblem associated with system (4.10):

\[
\omega_i^T (\xi_i^{(l)} A^{(l)} + \xi_i^{(s)} A^{(s)} + \xi_i^{(u)} A^{(u)} ) = 0^T, \quad (4.19)
\]

where \( \xi_i^{(l)} \), \( \xi_i^{(s)} \) and \( \xi_i^{(u)} \) is the set of eigenvalues corresponding to the eigenvector \( \omega_i \), and \( A^{(l)} = a_U \), \( A^{(s)} = b_U \) and \( A^{(u)} = c_U \) are \( n \times n \) matrices. The eigenvalues from
(4.19) can be found from the equation

\[ P_n(\zeta_t^{(t)}, \zeta_t^{(x)}, \zeta_t^{(y)}) \equiv \det(\overline{A}(\zeta_t^{(t)}, \zeta_t^{(x)}, \zeta_t^{(y)})) = 0, \]

where

\[ \overline{A}(\zeta_t^{(t)}, \zeta_t^{(x)}, \zeta_t^{(y)}) = (\zeta_t^{(t)}A^{(t)} + \zeta_t^{(x)}A^{(x)} + \zeta_t^{(y)}A^{(y)}) \]

and \( P_n \) is a polynomial function of degree \( n \) of the listed arguments.

For a hyperbolic system of partial differential equations, all eigenvalues of the problem (4.19) are real and the corresponding eigenvectors \( \omega \) are all linearly independent. Knowledge of the eigenvalues and eigenvectors of system (4.19) is important in practical computational applications because it allows one to determine the appropriate boundary conditions to be imposed on some particular segment of the boundary. Also, this information is used in numerical techniques in order to take into account the directions along which the disturbances in the flow propagate.

It is common knowledge that the change of independent variables (4.2), under which the new coordinates are specified as known functions of the old ones, maps characteristic lines or surfaces in the "old" coordinates to those in the "new" coordinates. However, in the present case the change of independent variables is defined by prescribing the dependence between the solution and some of the metric coefficients of the transformation. In addition, this change of coordinates requires adding four
new equations (4.6) and (4.7) to the governing system (4.17), which results in new families of characteristic directions. Because of this, we consider the characteristic equation for the extended system (4.18) below.

Using (4.14)–(4.16), one can write

\[
A(\xi^{(r)}, \xi^{(w)}, \xi^{(s)}) = (\xi^{(r)} A^{(r)} + \xi^{(w)} A^{(w)} + \xi^{(s)} A^{(s)}) =
\begin{bmatrix}
\bar{A} + S g_u & a J_u + S g_{u} & a [-\xi^{(2)}, -\xi^{(3)}] \\
g_u & g_u & 0 \\
\xi^{(r)} w_u & \xi^{(s)} w_u & -\xi^{(s)} I_2
\end{bmatrix}
\]  

(4.20)

where \( A^{(r)} = V Q, A^{(w)} = W Q, A^{(s)} = F Q, \) and \( \bar{A} = \bar{A}(\xi^{(1)}, \xi^{(2)}, \xi^{(3)}) \),

\[
\xi^{(1)} = J \xi^{(r)} - (t^T E w) \xi^{(w)} + (t^T E f) \xi^{(s)},
\]

(4.21)

\[
\xi^{(2)} = w^{(2)} \xi^{(w)} - f^{(2)} \xi^{(s)},
\]

(4.22)

\[
\xi^{(3)} = -w^{(1)} \xi^{(w)} + f^{(1)} \xi^{(s)},
\]

(4.23)

\[
g = \xi^{(w)} w - \xi^{(s)} f,
\]

\[
E = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}
\]

\( I_i \) is used to denote the \( i \times i \) identity matrix, and \( w^{(i)} \) and \( f^{(i)} \) are the respective components of the vectors \( w \) and \( f \). In order to find roots of the characteristic
polynomial \( \det A(\xi^{(r)}, \xi^{(\psi)}, \xi^{(\phi)}) \), observe that

\[
A(\xi^{(1)}, \xi^{(2)}, \xi^{(3)}) = \begin{bmatrix}
I_n & S & 0 \\
0 & I_2 & 0 \\
0 & 0 & I_2 \\
\end{bmatrix} \begin{bmatrix}
\overline{A} & aJ_u & ag^TE \\
gU & gu & 0 \\
\xi^{(r)}wU & \xi^{(r)}wu & -\xi^{(\psi)}I_2 \\
\end{bmatrix}, \tag{4.24}
\]

where zeros in the matrices denote zero blocks of the appropriate sizes. Since both matrices in (4.24) are square and the determinant of the first factor is 1, the characteristic polynomial is given by the determinant of the second factor. Premultiplying the second factor in (4.24) by the matrix (with the determinant equal to 1)

\[
\begin{bmatrix}
I_n & -\frac{\xi^{(r)}}{\xi^{(\psi)}}aw^TE & \frac{1}{\xi^{(\psi)}}ag^TE \\
0 & I_2 & 0 \\
0 & 0 & I_2 \\
\end{bmatrix} \tag{4.25}
\]

and assuming that \( \xi^{(\psi)} \neq 0 \), one obtains the matrix

\[
\begin{bmatrix}
\overline{A} + \xi^{(r)}aT_1 & \xi^{(r)}a(T_2 + J_u) & 0 \\
gU & gu & 0 \\
\xi^{(r)}wU & \xi^{(r)}wu & -\xi^{(\psi)}I_2 \\
\end{bmatrix}, \tag{4.26}
\]

where \( T_1 = f^TEwU - w^TEfU \) and \( T_2 = f^TEwU - w^TEfU \). Using (4.4) and (4.5), it
is easy to show (see Section 4.5.2) that

\[ T_1 = -J_U, \]  
\[ T_2 = -J_u. \]  

(4.27)  
(4.28)

Since it is assumed that \( J \) does not depend on \( U \) explicitly, the expression for the characteristic polynomial becomes

\[
\det(A(\xi^{(1)}, \xi^{(2)}, \xi^{(3)})) = \det(\overline{A}(\xi^{(1)}, \xi^{(2)}, \xi^{(3)})) \det(\overline{A}(\xi^{(\tau)}, \xi^{(\phi)}, \xi^{(\psi)})),
\]

(4.29)

where

\[
\det(\overline{A}(\xi^{(\tau)}, \xi^{(\phi)}, \xi^{(\psi)})) = \begin{bmatrix} g_u & 0 \\ \xi^{(\tau)}w_u & -\xi^{(\psi)}I_2 \end{bmatrix},
\]

(4.30)

Although it was assumed that \( \xi^{(\psi)} \neq 0 \), it is easy to verify that (4.29) remains true even if \( \xi^{(\psi)} = 0 \).

The factorization of characteristic polynomial (4.29) demonstrates that the set of characteristic values of the problem splits into two sets. The first one, obtained from \( \det(\overline{A}(\xi^{(1)}, \xi^{(2)}, \xi^{(3)})) = 0 \), can be derived from the characteristic eigenvalues of the system (4.10) using (4.21)–(4.23). This set of eigenvalues corresponds to the characteristic surfaces of the system (4.10) being mapped to the \((\tau, \varphi, \psi)\) space. The second set of eigenvalues results from \( \det(\overline{A}(\xi^{(\tau)}, \xi^{(\phi)}, \xi^{(\psi)})) = 0 \), and is due to equations
(4.6)-(4.7), which ensure required geometrical properties of the computational grid.

Consider the case when orthogonality of the grid is imposed by setting $F = 0$ and $J = \sqrt{EE'}$ in (4.4) and (4.5). Equation (4.30) then reduces to $\xi^q(\xi^q)^3 = 0$, and hence the family of characteristic lines consists of lines $\varphi = const$ and $\psi = const$. Calculating the eigenvectors $\omega_i$ and writing the corresponding characteristic relations along these families of characteristic lines shows that there is one characteristic relation valid along the line $\psi = const$ and involving the derivatives of $E$ by $\varphi$ only. Similarly, there are characteristic relations imposed along the lines $\varphi = const$ and each of these relations involves the derivatives of $G$ and the two components of $t$ only. These facts indicate that the boundary values for $E$ must be prescribed along the bounding segment $\varphi = const$, while the values of $G$ and $t$ need to be imposed along the boundary $\psi = const$.

4.5 Appendix: Some Derivations from Sections 4.3 and 4.4

4.5.1 Derivation of Eqns. (4.14) – (4.16)

First notice that, as follows from (4.2), $t$ depends on $\tau$ only, and therefore partial derivatives $t_\varphi$ and $t_\psi$ are both identically equal to zero. This fact allows to simplify
(4.12) to the following relation:

\[
\begin{bmatrix}
    dx \\
    dy \\
    dt \\
\end{bmatrix}
= \begin{bmatrix}
    \frac{\partial(x,y)}{\partial(\phi,\psi)} & \frac{\partial(x,y)}{\partial(\psi,\tau)} & \frac{\partial(x,y)}{\partial(\tau,\phi)} \\
    0 & y_\psi & -y_\phi \\
    0 & -x_\psi & x_\phi \\
\end{bmatrix}
\begin{bmatrix}
    d\phi \\
    d\psi \\
    d\tau \\
\end{bmatrix}.
\]

Using these relations in (4.11) results in (4.14)–(4.16). For instance, in notations of (4.14)–(4.16):

\[
B = a \frac{\partial(x,y)}{\partial(\psi,\tau)} + b \frac{\partial(y,t)}{\partial(\psi,\tau)} + c \frac{\partial(t,x)}{\partial(\psi,\tau)} = a(x_\psi y_\tau - y_\psi x_\tau) + b y_\psi - c x_\psi
\]

\[
= [-c + y_\tau a, b - x_\tau a][x_\psi, y_\psi]^T = Sw.
\]

\section*{4.5.2 Derivation of Eqns. (4.27)–(4.28)}

Using the definition of \(T_1\), write:

\[
T_1 = f^T E w U - w^T E f U
\]

\[
= [x_\psi y_\psi]
\begin{bmatrix}
    0 & 1 \\
    -1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
    x_\psi U \\
    y_\psi U \\
\end{bmatrix}
- [x_\psi y_\psi]
\begin{bmatrix}
    0 & 1 \\
    -1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
    x_\psi U \\
    y_\psi U \\
\end{bmatrix}
\]

\[
= (x_\psi y_\psi U - y_\psi x_\psi U) - (x_\psi y_\psi U - y_\psi x_\psi U)
\]

\[
= -(x_\psi y_\psi - y_\psi x_\psi) U = -J U.
\]

The expression for \(T_2\) is derived in the same way.
Chapter 5

Numerical Solution of Unsteady Euler Equations in Streamline–Aligned Coordinates

5.1 Introduction

A number of different approaches have been utilized in recent years to solve the system of unsteady Euler equations describing the motions of compressible inviscid gas. These approaches fall into the same classes as the methods to compute purely supersonic flows as described in Section 3.1, because in both cases the problems to be solved are hyperbolic.

As the analysis in Section 4.4 demonstrates, the extended system of Euler equa-
tions in streamline-aligned coordinates (4.18) is also hyperbolic. This extended system is comprised of transformed Euler equations (4.17) and coordinate transformation equations (4.6) and (4.7). In this work, in order to solve the extended system (4.18) numerically, the solution algorithm is set up in two intermittent steps:

1. using either initial data or the solution from the previous timestep the coordinate transformation equations (4.6) and (4.7) are solved, thus producing the mesh aligned along the velocity vector field on the previous timestep;

2. using the metric coefficients of the transformation found at the previous step of the algorithm, the system of transformed Euler equations (4.17) is solved and the flowfield is advanced to the next timestep.

The following sections describe implementation details of this algorithm.

5.2 Coordinate Transformation Equations

5.2.1 Boundary Conditions and Formulation of the Problem

In the system of equations (4.6) and (4.7), w and f are given by (4.4) and (4.5) and \( t = [x_r, y_r] \). Here and in the following, an orthogonal system of coordinates is chosen, so that \( F \) should be set to zero. Expressing \( \cos \alpha \) and \( \sin \alpha \) in terms of the
components of the velocity vector, (4.4) and (4.5) can be rewritten as

$$\mathbf{w} = -\frac{\sqrt{E}}{q} \begin{bmatrix} v \\ -u \end{bmatrix},$$

$$\mathbf{f} = \frac{\sqrt{G}}{q} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{5.2}$$

In the system of four equations (4.6) and (4.7) there are four unknown functions, namely $E$, $G$ and $t$, and appropriate boundary conditions need to be prescribed for them. It was established in Section 4.4 that the boundary values for $E$ can be prescribed along the bounding segment $\varphi = 0$, while the values of $G$ and $t$ can be imposed along the boundary $\psi = 0$. If $x$ and $y$ are known along these boundaries (i.e., the locations of grid nodes are specified), these values can be derived from (4.3).

### 5.2.2 Numerical Solution of Coordinate Transformation Equations

Equation (4.6) is approximated using one-sided differences to obtain

$$\frac{(w_{ij}^{(n)} - w_{ij}^{(n)}_{-1})}{h_\varphi} - \frac{(f_{ij}^{(n)} - f_{ij-1,j}^{(n)})}{h_\psi} = 0, \tag{5.3}$$

where the indices $i$ and $j$ increase in the directions of increasing values of $\varphi$ and $\psi$ respectively and the index $n$ increases with time. Provided that the values of $E$ and
$G$ at the nodes $(i,j - 1)$ and $(i - 1,j)$ are known, equation (5.3) can be resolved with respect to the values of $E$ and $G$ at the node $(i,j)$ and this process can then be repeated for the next node.

Similarly, equation (4.7) can be approximated as

$$\frac{t_{i,j}^{(n)} - t_{i-1,j}^{(n)}}{h_{\psi}} = \frac{(w_{i,j}^{(n)} + w_{i-1,j}^{(n)}) - (w_{i,j}^{(n-1)} + w_{i-1,j}^{(n-1)})}{2h_r} = 0. \tag{5.4}$$

The resulting system of finite-difference equations can easily be resolved with respect to the values of $t_{i,j}^{(n)}$ using known boundary conditions for $t$.

Although the approximation (5.4) uses central differences and is of second order of accuracy, the approximation (5.3) is of first order of accuracy. Use of one-sided differences in (5.3) leads to a system of difference equations which can easily be resolved. This reduction of the order of accuracy in the grid generation equations does not significantly affect accuracy of the solution.
5.3 Transformed Euler Equations

Before describing the numerical algorithm used to solve the transformed Euler equations, the expressions for fluxes $A, B$ and $C$ are rewritten, using (4.14)–(4.16), as

$$
A = \begin{bmatrix}
\rho J \\
\rho u J \\
\rho v J \\
(\frac{1}{2}\rho q + \frac{p}{\gamma-1})J
\end{bmatrix}, 
B = \begin{bmatrix}
A_0 \frac{\partial J}{\partial x} \\
A_1 \frac{\partial J}{\partial x} + w_2 p \\
A_2 \frac{\partial J}{\partial x} - w_1 p \\
A_3 \frac{\partial J}{\partial x} + U p
\end{bmatrix}, 
C = \begin{bmatrix}
-A_0 \frac{\partial J}{\partial y} \\
-A_1 \frac{\partial J}{\partial y} - f_2 p \\
-A_2 \frac{\partial J}{\partial y} + f_1 p \\
-A_3 \frac{\partial J}{\partial y} - V p
\end{bmatrix}
$$

(5.5)

where $A_i, i = 0, 1, 2, 3$ are the components of $A$ and

$$
J = f_1 w_2 - f_2 w_1,
J_1 = t_2 w_1 - t_1 w_2,
J_2 = t_2 f_1 - t_1 f_2,
U = u w_2 - v w_1,
V = u f_2 - v f_1.
$$

5.3.1 Internal Nodes

The algorithm used to compute the values of the flow parameters at the internal nodes is a classical two-step Lax–Wendroff algorithm [59] with the modifications described below.
With \( w^{(n)}, f^{(n)} \) and \( t^{(n)} \) known from the solution of the coordinate transformation equations, the first step of the Lax–Wendroff algorithm proceeds as

\[
A_{i,j}^{(n+1/2)} = \frac{1}{4}(A_{i+1,j}^{(n)} + A_{i-1,j}^{(n)} + A_{i,j+1}^{(n)} + A_{i,j-1}^{(n)}) - h_x \frac{B_{i,j+1}^{(n)} - B_{i,j-1}^{(n)}}{4h_x} - h_p \frac{C_{i+1,j}^{(n)} - C_{i-1,j}^{(n)}}{4h_p}.
\] (5.6)

After the values of \( A^{(n+1/2)} \) are computed, they need to be resolved with respect to the flow parameters \( u, v, p \) and \( \rho \) in order to find the values of \( B^{(n+1/2)} \) and \( C^{(n+1/2)} \). As can be seen from the expressions (5.5), this requires knowledge of the quantity \( J^{(n+1/2)} \), which is not known at this time.

Although the value of \( J^{(n+1/2)} \) could be taken to be equal to its value at \( \tau = \tau^{(n)} \), our experience shows that this results in significant instabilities of the solution process in the case when the grid is becoming strongly and rapidly distorted. These instabilities are due to strong variations of \( J \). As can be seen from the expression for the first component of \( A \) in (5.5), neglecting the difference between \( J^{(n+1/2)} \) and \( J^{(n)} \) is equivalent to perturbing the value of \( \rho^{(n+1/2)} \). It was observed that, if the value of \( J^{(n)} \) is used instead of \( J^{(n+1/2)} \) to resolve \( A^{(n+1/2)} \), then the calculations may lose stability even if the flow is locally close to being uniform and steady. Notice that even in this case the grid may be strongly distorted due to the influence of the flow upstream.

In order to avoid these instabilities, a formula to evaluate \( J^{(n+1/2)} \) was devised
and tested, and proved to be successful. Namely, $J^{(n+1/2)}$ is calculated in such a way that, for a uniform and steady flow, (5.6) is identically satisfied. This leads to

$$
J^{(n+1/2)}_{i,j} = \frac{1}{4} \left( J^{(n)}_{i+1,j} + J^{(n)}_{i-1,j} + J^{(n)}_{i,j+1} + J^{(n)}_{i,j-1} \right) - h_r \frac{(J_1)_{i,j+1}^{(n)} - (J_1)_{i,j-1}^{(n)}}{4h_\phi} - h_r \frac{(J_2)_{i+1,j}^{(n)} - (J_2)_{i-1,j}^{(n)}}{4h_\phi}.
$$

This choice ensures that uniform and steady solution is preserved by the discretization (5.6). Also, it can be easily demonstrated that this expression gives a first order extrapolation in time for the value of $J^{(n+1/2)}$.

Similarly, the second step of the Lax–Wendroff algorithm is

$$
A_{i,j}^{(n+1)} = A_{i,j}^{(n)} - h_r \frac{B_{i,j+1}^{(n+1/2)} - B_{i,j-1}^{(n+1/2)}}{2h_\phi} - h_r \frac{C_{i+1,j}^{(n+1/2)} - C_{i-1,j}^{(n+1/2)}}{2h_\phi}.
$$

Finally, the flow parameters $u, v, p$ and $\rho$ are calculated for $\tau = \tau^{(n+1)}$ using the components of $A^{(n+1)}$ and the value of $J^{(n+1)}$ which is evaluated from considerations similar to those used on the first step of the Lax–Wendroff algorithm, i.e.,

$$
J^{(n+1)}_{i,j} = J^{(n)}_{i,j} - h_r \frac{(J_1)_{i,j+1}^{(n+1/2)} - (J_1)_{i,j-1}^{(n+1/2)}}{2h_\phi} - h_r \frac{(J_2)_{i+1,j}^{(n+1/2)} - (J_2)_{i-1,j}^{(n+1/2)}}{2h_\phi}.
$$

85
5.3.2 Boundary Nodes

In order to compute the values of the flow parameters at the boundary nodes the same two-step Lax–Wendroff algorithm is applied at the boundary nodes. This procedure requires extrapolation of the flow and grid parameters to the fictitious cells and nodes located outside of the computational domain [59]. The extrapolation procedure depends on the type of boundary being considered as described below.

Solid Wall Boundary

Along a steady solid wall, an impermeability boundary condition must be imposed. This is equivalent to stating that the component of the velocity vector normal to the wall vanishes at the wall itself. If the solid wall is set in motion, an impermeability condition requires that the normal component of the flow velocity vector at each point of the wall is equal to the normal component of this wall point's speed of motion.

A simple and well-known way to impose this type of boundary conditions is to make use of the reflection technique [59], in which the values of the flow parameters in the fictitious (reflected) node are defined to ensure vanishing normal velocities (or relative normal velocities) at the wall. Denoting the values at the reflected node by subscript R and the values at near-wall node and wall node by subscripts P and W respectively, conditions at the reflected point for a stationary wall can be defined as
\[ \rho_R = \rho_P, \]
\[ v_R = f_w \frac{(v_P, f_w)}{(f_w, f_w)} - w_w \frac{(v_P, w_w)}{(w_w, w_w)}, \]
\[ p_R = 2pw - p_P. \]

Using the above values at the reflected nodes, steps of the Lax–Wendroff method are performed for the wall nodes resulting in the updated values of the unknown quantities at the wall.

In the case of a moving wall, the impermeability boundary conditions dictate that the normal (with respect to the wall) component of the velocity vector at the wall is equal to the normal component of the velocity of the wall motion at this location. In other words, this corresponds to imposing zero normal velocity in the frame of reference affixed to the wall. This leads to the following expression for the velocity vector at the reflected node:

\[ v_R = t_w + f_w \frac{((v_P - t_w), f_w)}{(f_w, f_w)} - w_w \frac{((v_P - t_w), w_w)}{(w_w, w_w)}. \]

Expressions for \( \rho_R \) and \( p_R \) remain as above.

The case of a moving wall boundary requires further special consideration because there is a conflicting set of requirements imposed on the streamline–aligned coordinate system near the moving wall boundary. On the one hand, all coordinate lines
\[ \psi = \text{const} \] are required to be aligned along the velocity vector field at any given time. On the other hand, this requirement cannot be enforced along the moving wall (assuming that the latter is a coordinate line \( \psi = \text{const} \)), because the velocity vector always has a non-zero component normal to the wall as a consequence of the wall motion. This contradiction is partially resolved during the numerical solution of the coordinate transformation equations. To be precise, the approximating relations (5.3) are written only for the internal nodes of the grid and therefore the grid is aligned along the velocity vector field only at the internal nodes. At the moving wall boundary the locations of the nodes are known as a function of time and only the boundary conditions for (5.3) (resulting from the known locations of the nodes as a function of time) are imposed. This treatment, however, has a major disadvantage in that the coordinate transformation can degenerate near the moving wall. As our experience shows, if the wall moves sufficiently fast, the near wall grid lines start to intersect with the wall after some time. This intersection of grid lines does not occur if the wall moves slow enough or the amplitude of its motion is less than the characteristic spacing between the near wall grid lines.

A more universal approach to the treatment of the moving wall is to align the coordinate lines \( \psi = \text{const} \) to the relative velocity field \( \mathbf{v} - \mathbf{t} \), instead of aligning it
to the velocity field \( \mathbf{v} \), i.e., the expressions (5.1) and (5.2) need to be modified to

\[
\mathbf{w} = -\frac{\sqrt{E}}{q} \begin{bmatrix} v - y_{\tau} \\ -u + x_{\tau} \end{bmatrix}, \tag{5.8}
\]

\[
\mathbf{f} = \frac{\sqrt{G}}{q} \begin{bmatrix} u - x_{\tau} \\ v - y_{\tau} \end{bmatrix}. \tag{5.9}
\]

In this case, the moving wall is tangential to the relative velocity field at all times and intersection of the grid lines with the moving wall does not occur. Also if the wall is not moving and the flow reaches the steady limit, the suggested treatment reduces to (5.1) and (5.2).

**Inlet and Outlet Boundaries**

At these boundaries, conditions are imposed using the following consideration based on the quasi-one-dimensional nature of the flow near these boundaries.

In the streamline-aligned coordinate system one of the coordinates is aligned along the local direction of the velocity vector. If the inlet and outlet are chosen at locations where the flow does not vary significantly, the flow can be assumed to be close to uniform in the vicinity of each node at these boundaries.

In all examples considered in the following, the boundary conditions imposed at the inlet and outlet boundaries are uniform and steady. In these circumstances (more precisely, if pressure and velocity vector are constant along the boundary segments),
the characteristic system of multi-dimensional Euler equations locally reduces to a form resembling the one-dimensional characteristic form [59, p.377–379]. Under these conditions, the characteristic conditions require that Riemann invariants of the system are constant along the characteristic lines. This fact can be utilized to devise a simple treatment of inlet and outlet boundary conditions based on the locally simplified form of the compatibility relations [59] which is similar to the treatment of far-field boundaries for steady-state flows [60].

The Riemann invariants $R^+_n$ and $R^-_n$ associated with the acoustic waves are introduced as [59]

$$
R^+_n = v_n + \frac{2c}{\kappa - 1} \\
R^-_n = v_n - \frac{2c}{\kappa - 1}
$$

where $v_n$ is the component of the velocity vector normal to the physical inlet/outlet boundary and $c$ is the local speed of sound.

Based on the above assumptions, the locally one-dimensional form of compatibility relations corresponding to acoustic waves at the inlet or outlet boundary becomes [59]

$$
R^\pm_n = \text{const} \text{ along the direction } C_\pm
$$

where $C_\pm$ are the characteristic directions with the components $(1, u \pm cn_x, v \pm cn_y)$ in the $(t, x, y)$ space and $n_x$ and $n_y$ are the components of the unit normal vector at
the boundary.

In addition to the above acoustic wave relations, entropy function and full enthalpy are constant along the characteristic direction \( C_0 \) with the components \( (1, u, v) \) in the \( (t, x, y) \) space.

In case of a subsonic inlet boundary, for each point on this boundary there are two positive incoming characteristic directions \( (C_0 \text{ and } C_+) \). Therefore, the values of the Riemann invariants corresponding to these directions \( (R^+, \text{ entropy and full enthalpy}) \) must be obtained from the imposed physical boundary conditions.

The value of the remaining invariant \( (R^-) \) must be estimated from inside the computational domain using an extrapolation procedure and the fact that this invariant remains constant along the direction \( C_- \).

After the values of all four invariants are known for the given boundary point, they are used to obtain the updated values of the primitive variables on the current timestep.

For a subsonic outlet boundary, there is only one characteristic direction \( (C_-) \) coming from outside of the domain and two characteristic directions \( (C_+ \text{ and } C_0) \) which come from inside the domain. Accordingly, only one of the invariants \( (R^-) \) is computed using the physical boundary conditions and the remaining ones are extrapolated from inside the domain.
5.3.3 Artificial Dissipation Terms

All second-order central-difference schemes of the Lax-Wendroff type generate oscillations around sharp discontinuities. It is typically possible to avoid these undesirable features by including additional dissipative terms in the approximation. Addition of these terms is equivalent to adding the fictitious viscosity which reduces the oscillations but, at the same time, reduces the local order of the second-order approximation to a first-order one.

The most elementary way to achieve this, while preserving the conservativeness of the approximation, is to modify the numerical fluxes participating in the second step of the Lax-Wendroff algorithm in such a way that additional dissipation is generated. Following the ideas from [59], the modified fluxes from (5.7) can be introduced as

\[
B_{i,j}^{(n+1/2)} = B_{i,j}^{(n+1/2)} + h_x d_{i,j} (A_{i,j+1}^{(n+1/2)} - A_{i,j-1}^{(n+1/2)}),
\]

\[
C_{i,j}^{(n+1/2)} = C_{i,j}^{(n+1/2)} + h_x d_{i,j} (A_{i+1,j}^{(n+1/2)} - A_{i-1,j}^{(n+1/2)}),
\]

where \(d_{i,j}\) is a coefficient of artificial viscosity.

5.4 Numerical Results

A number of test calculations have been performed using the approach described in the previous section. The results of these calculations are presented below.
Subsonic Channel with 20% Bump

A steady subsonic flow in a channel with 20% circular bump with the flow entering the channel at $M_\infty = 0.4$ has been computed and the results of these calculations are presented on Figs. 5.1–5.3. The figures represent the distributions of Mach number and pressure along the walls and pressure contours $((p/p_\infty)_{\text{max}} = 1.23, (p/p_\infty)_{\text{min}} = 0.57, \Delta(p/p_\infty) = 0.033)$ with solid lines representing the results obtained in this work, and the triangles representing the results from [44]. The grid used in the present calculation is shown in Fig. 5.4. This grid has 61 nodes in the direction transversal to the flow and 181 nodes in the streamwise direction. There is a good agreement with the results from [44] in pressure and Mach number plots, except for the regions near the corners of the bump. The loss in accuracy of the results in these regions can be attributed to the locally coarser grid used in our calculations and the smoothing generated by the Lax–Wendroff algorithm through numerical dissipation. The numerical treatment used in [44] is based on an advanced version of a flux-splitting algorithm [61, 62] which results in significantly less numerical dissipation.

Transonic Channel with 10% Bump

Another example is a test case of steady transonic flow in a channel with 10% circular bump and $M_\infty = 0.675$. Results are given in Figs. 5.5–5.7. Similar to the previous example, the distributions of Mach number and pressure along the walls and pressure contours $((p/p_\infty)_{\text{max}} = 1.23, (p/p_\infty)_{\text{min}} = 0.57, \Delta(p/p_\infty) = 0.033)$ are presented and
compared to the results of calculations from [44]. The grid used in this calculation is
the same as in the previous test case. Again, the present results demonstrate a good
overall agreement with the calculations performed in [44]. However, due to effects of
stronger numerical dissipation in the Lax–Wendroff algorithm, the shock is smeared
by approximately 3–5 nodes in the present work compared to just 1–2 nodes in [44].
One of the grids on which the calculations were also run is a 71 × 135 grid shown on
Fig. 5.8.

Bumpy Channel with Oscillating Membrane

In order to test the capabilities of the present approach in the unsteady case, we
consider the flow when one of the walls in the channel undergoes an oscillatory motion.

The shape of the lower oscillating wall of the channel varies according to

\[
\begin{align*}
  x(\varphi) &= 3\varphi, \\
y(\varphi) &= \begin{cases} 
  \frac{1}{20} \sin \left( \frac{2\pi t}{T} \right) \left( 1 - \cos \left( 6\pi (\varphi - \frac{1}{3}) \right) \right), & \text{if } \frac{1}{3} < \varphi < \frac{2}{3}; \\
  0, & \text{otherwise.}
\end{cases}
\end{align*}
\]

where 0 < \varphi < 1 and T is the period of oscillations, taken to be equal to 1 second in
this computed example. The flow parameters at the subsonic inlet were assumed to
be \( \rho_\infty = 1.3 \text{ kg/m}^3 \), \( p_\infty = 10^5 \text{ Pa} \) and \( q_\infty = 50 \text{ m/sec} \). At the outlet, constant value
of pressure \( p_{out} = 10^5 \text{ Pa} \) was imposed. Initial conditions (at \( t = 0 \text{ sec} \)) were taken
to be the uniform flow with the same parameters as the flow at the inlet.
Figures 5.9–5.14 present development of the flow pattern for the first 0.30 sec. The presented data correspond to times $t = 0.00 \text{ sec}$, $t = 0.05 \text{ sec}$, $t = 0.10 \text{ sec}$, $t = 0.15 \text{ sec}$, $t = 0.20 \text{ sec}$, and $t = 0.25 \text{ sec}$ and $t = 0.30 \text{ sec}$. For each of these instances, the velocity vector field and the contours of relative pressure $p_{rel} = p - p_\infty$ ($\left( p_{rel} \right)_{\text{max}} = 300 \text{ Pa}$, $\left( p_{rel} \right)_{\text{min}} = -200 \text{ Pa}$, $\Delta(p_{rel}) = 25 \text{ Pa}$) are presented.
Figure 5.1: Subsonic channel with 20% bump.

Mach number along the walls

96
Figure 5.2: Subsonic channel with 20% bump.

Pressure along the walls

\[ \frac{P}{P_0} \]

\[ x \]

97
Figure 5.3: Subsonic channel with 20% bump. Pressure contours.
Figure 5.4: Subsonic channel with 20% bump. Computational grid.
Figure 5.5: Transonic channel with 10% bump.
Figure 5.6: Transonic channel with 10% bump.

Pressure along the walls
Figure 5.7: Transonic channel with 10% bump. Pressure contours.
Figure 5.8: Transonic channel with 10% bump. Computational grid.
Figure 5.9: Flow about an oscillating membrane. $t = 0.05$ sec.
Figure 5.10: Flow about an oscillating membrane. $t = 0.10 \text{ sec.}$
Figure 5.11: Flow about an oscillating membrane. $t = 0.15$ sec.
Figure 5.12: Flow about an oscillating membrane. $t = 0.20$ sec.
Figure 5.13: Flow about an oscillating membrane. $t = 0.25 \text{ sec.}$
Figure 5.14: Flow about an oscillating membrane. $t = 0.30$ sec.
Chapter 6

Conclusions

Based on the research carried out in this dissertation, the following observations and conclusions can be made.

1. A new approach to the solution of steady and unsteady Euler and full-potential equations in 2D based on the original governing equations formulated in streamline-aligned coordinates has been developed. The approach can be used to calculate subsonic transonic and supersonic flows including flows with free surfaces (jets) and flows around moving walls. Both space-marching, time-marching and iterative numerical algorithms have been used to solve the governing equations. The calculated results compare well with the existing experimental data and the results of other calculations.

2. For potential flows, both plane 2D and axisymmetric, two new forms of governing equations in orthogonal streamline coordinates (streamfunction and velocity
vector potential) have been derived. Orthogonality of the coordinate transformation ensures that the coordinate lines do not intersect even in case of flows with varying direction. The first formulation is a single second-order full-potential equivalent equation cast in a conservative form. The second form is a set of two second-order coupled equations. To solve these equations numerically, an iterative algorithm based upon the approximate factorization technique and artificial density concept is used to solve the single second-order full-potential equation. The system of two second-order equations is solved using Gauss-Seidel iterative algorithm with multigrid acceleration. Both of the tested approaches have their own strengths and weaknesses. The approach based on a single second-order equation allows to utilize the solution methodologies developed for a full-potential equation. On the other hand, the coefficients of this single equation are not determined uniquely from the values of the unknown function and its derivatives and special treatment needs to be utilized to find the correct root in the implicit relations determining the values of the flow parameters. The alternative approach, based on a coupled system of two second order equations allows to avoid this non-uniqueness problem at the price of solving two equations instead of just one.

3. Steady supersonic Euler equations are considered in streamline-aligned orthogonal coordinates formed by a streamfunction of the flow and its orthogonal compliment. Because the velocity vector field is no longer irrotational in this case, velocity vector potential cannot be used as a second coordinate. A more general orthogonal
compliment function is introduced together with a new unknown function serving as an integrating factor, ensuring that the compatibility conditions for the first derivatives of the compliment function are satisfied. This leads to the necessity to consider an extended system of governing equations consisting of steady Euler equations cast into the new coordinates and the compatibility relation. The characteristic properties of this extended system are investigated and a grid-characteristic marching numerical algorithm taking into account the characteristic form of the equations is developed.

4. A novel formulation of unsteady Euler equations in streamline-aligned coordinates is derived. In case of unsteady compressible flow, the continuity equation contains the \( \rho_t \) term, which makes it impossible to introduce the streamfunction as in the case of steady flow. This difficulty is overcome by considering the metric coefficients of the coordinate transformation to ensure that one set of the coordinate lines is aligned along the velocity vector field. Similar to the case of steady Euler equations, this procedure leads to the appearance of "geometrical" unknowns (generalized integrating factors) and additional equations ensuring compatibility of the coordinate transformation. Characteristic properties of this extended system are investigated in order to determine the correct set of boundary conditions for the "geometrical" unknown functions. The resulting extended system of governing equations is solved numerically using a Lax–Wendroff time-stepping algorithm.

5. The approaches developed in this dissertation have certain limitations and can be extended and continued in a number of ways. Particularly, as experience
shows, in the case of transonic flows, the iterative algorithms used to solve potential flows require careful and time-consuming adjustment of the iterative parameters in order to achieve the converged solution. The grid-characteristic algorithm used to solve supersonic flows is based upon the simple linearization procedure. The Lax–Wendroff algorithm, used to solve the unsteady equations, is simple to implement and to use. However, it produces oscillations which require numerical dissipation to be added explicitly, which results in excessive (compared to other algorithms) smearing. More elaborate algorithms (particularly those using non-linear flux vector splitting or Godunov methodology) can produce reasonably monotone and sharp resolution of discontinuities in hyperbolic problems. Finally, use of streamline-adapting coordinates could lead to the development of a more efficient Riemann solver (or approximate Riemann solver) and possible extensions of this work should concentrate on utilizing these advantages.
References


116


[41] Latypov, A.M. and Shipilin, A.V. *On an iterational method for calculating sub- and transonic internal flows*. Computational Mathematics and Mathematical


1The original Russian version of this paper has appeared in Zh. Vychisl. Mat. i Mat. Fiz., 31, 5, 767-776, 1991


