UNSFLO: A Numerical Method For The Calculation Of Unsteady Flow In Turbomachinery

by

Michael Giles

GTL Report #205 May 1991

GAS TURBINE LABORATORY

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

CAMBRIDGE, MASSACHUSETTS

UNSFLO: A Numerical Method For The Calculation Of Unsteady Flow In Turbomachinery

by

Michael Giles

GTL Report #205 May 1991

Acknowledgements

This research has been supported by Rolls-Royce PLC and I am grateful to Dr. Peter Stow and Dr. Arj Suddhoo of Rolls-Royce for technical discussions and collaboration throughout this work. Bob Haimes contributed greatly to the implementation, debugging and validation of the UNSFLO suite of programs, and many students and other users have contributed with their suggestions, questions, and careful proof-reading of this report.

Contents

List of Figures

List of Symbols

Variables

A	area (or volume) of cell
c	speed of sound
$c_{1,2,3,4}$	characteristic variables
D	fractional wake velocity defect
E	total energy
F	axial flux
G	circumferential flux
h	streamtube thickness
H	total enthalpy
M	Mach number
M_x	axial Mach number
M_y	circumferential Mach number
N	sawtooth function
p	pressure
P	blade pitch
q	speed
Q	'tilted' conservation variables
s	entropy-related function
S	quasi-3D source term
T	blade-passing period
u	axial velocity
U	conservation variables
v	circumferential velocity
V	wheel speed
W	fractional wake width

x, y	coordinates
α	flow angle (from axial)
γ	ratio of specific heats
λ	'time-tilting' parameter
ϕ	potential function
ρ	density
σ	under-relaxation parameter
Δt	time-step
ΔT	time-lag in periodic b.c.
δU	change in variables at node
ΔU	change in variables in cell
$D^2 U$	second difference of U

Subscripts

F	flux-averaged
inl	prescribed inflow
out	prescribed outflow
r	rotor
s	stator

Chapter 1

Introduction

1.1 Unsteadiness in turbomachinery

There are four principal sources of unsteadiness in a single stage of a turbomachine in which there is one row of stationary blades (stators) and one row of moving blades (rotors). As shown in Fig. 1.1, wake/rotor interaction causes unsteadiness because the stator wakes, which one can consider to be approximately steady in the stator frame of reference, are unsteady in the rotor frame of reference since the rotor is moving through the wakes and chopping them into pieces. This causes unsteady forces on the rotor blades and generates unsteady pressure waves. Although the stator wakes are generated by viscosity, the subsequent interaction with the rotor blades is primarily an inviscid process and so can be modelled by the inviscid equations of motion. This allows two different approaches in numerical modelling. The first is to perform a full unsteady Navier-Stokes calculation of the stator and rotor blades. The second is to perform an unsteady inviscid calculation for just the rotor blade row, with the wakes being somehow specified as unsteady inflow boundary conditions. This latter approach is computationally much more efficient, but assumes that one is not concerned about the unsteady heat transfer and other viscous effects on the rotor blades.

Potential stator/rotor interaction causes unsteadiness due to the fact that the pressure in the region between the stator and rotor blade rows can be decomposed approximately into a part that is steady and uniform, a part that is non-uniform but steady in the rotor frame (due to the lift on the rotor blades) and a part that is non-uniform but steady in the stator frame (due to the lift on the stator blades). As the rotor blades move, the stator trailing edges experience an unsteady pressure due to the non-uniform

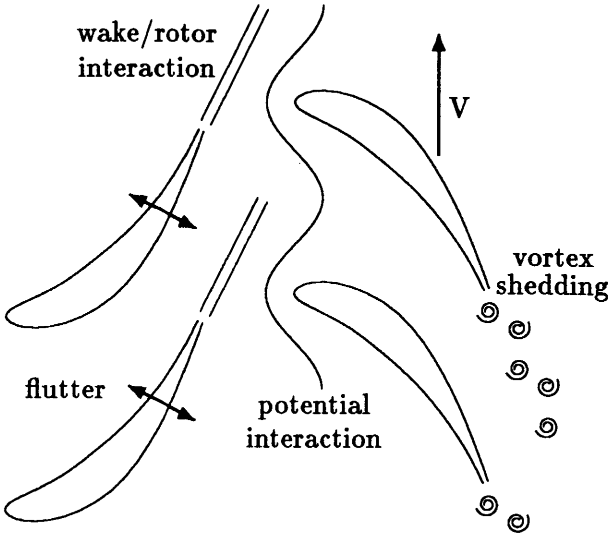

Figure 1.1: Sources of unsteadiness in turbomachinery flow

part that is locked to the rotors, and the rotor leading edges experience an unsteady pressure due to the non-uniform part that is locked to the stators. This is a purely inviscid interaction which is why it is labelled a "potential" interaction. There are again two approaches to modelling this interaction. The first is an unsteady, inviscid calculation of the stator and rotor blade rows. The second is an unsteady, inviscid calculation of just one of the blade rows, either the stator or the rotor, with the unsteady pressure being specified as a boundary condition. The latter approach is more efficient, but unfortunately the situation in which the potential stator/rotor interaction becomes important is when the spacing between the stator and rotor rows is extremely small, and/or there are shock waves moving in the region between them. Consequently, one does not usually know what values to specify as unsteady boundary conditions.

The first two sources of unsteadiness were both due to the relative motion of the stator and rotor rows. The remaining two sources are not. The viscous flow past a blunt turbine trailing edge results in vortex shedding, very similar to the Karman vortex street shed behind a cylinder. In fact real wakes lie somewhere between the two idealized limits of a Karman vortex street and a turbulent wake with steady mean velocity profile. It is believed that provided the integrated loss is identical the choice of model does not affect the subsequent interaction with the downstream rotor blade row. However, this is an assumption which needs to be investigated sometime in the future. The importance of vortex shedding lies in the calculation of the average pressure around the blunt trailing edge, which determines the base pressure loss, a significant component of the overall loss. There is also experimental evidence to suggest that the vortex shedding can be greatly amplified under some conditions by the potential stator/rotor interaction.

Finally, there can be unsteadiness due to the motion of the stator or rotor blades. The primary concern here is the avoidance of flutter. This is a condition in which a small oscillation of the blade produces an unsteady force and moment on the blade which due to its phase relationship to the motion does work on the blade and so increases the amplitude of the blade's unsteady motion. This can rapidly lead to very large amplitude blade vibrations, and ultimately blade failure.

1.2 A brief review

In the last eight years considerable effort has been devoted to the calculation of unsteady flow in turbomachinery. The first significant piece of work was by Erdos in 1977 [9]. In his paper he presented a calculation of unsteady flow in a fan stage, including the use of an algorithm to treat unequal pitches. Unfortunately, this method has some limitations which will be discussed later. In 1985, Koya extended Erdos' work to three dimensions [21].

In 1984, Hodson modified a program written by Denton, and used Erdos' technique, to calculate wake/rotor interactions in a low speed turbine [18]. The incoming wakes were specified as unsteady boundary conditions. The results show that the wake segments cut by the turbine rotors roll up into two counter-rotating passage vortices, and the wake fluid migrates to the suction surface.

In 1985, Rai presented a paper showing stator/rotor interaction calculated using a Navier-Stokes algorithm [29]. This paper generated considerable interest and sparked a lot of research activity. In 1988, Rai extended his techniques to three-dimensional, viscous calculations [30]. However, Rai, along with many other researchers since, assumed that the stator/rotor pitch ratio is 1:1 or some simple ratio such as 2:3 or 3:4. This assumption allows them to perform calculations with simple periodic boundary conditions, but requires modifications to the geometry when applied to real turbomachinery stages.

In the last few years there have been several papers: Fourmaux [10] and Lewis [22], inviscid, two-dimensional stator/rotor interaction; Jorgensen [20] viscous, quasi-three-dimensional stator/rotor interaction; Ni [27], inviscid three-dimensional stator/rotor interaction; Chen [4], three-dimensional, viscous stator/rotor interaction. In general, these papers have concentrated on numerical algorithm issues, and proof-of-concept demonstrations. Progressively now, the emphasis is turning to applications and mathematical modelling issues such as transition and turbulence modelling. Notable work in this latter category has been done by Sharma [34].

1.3 Overview of UNSFLO and report

The computer program UNSFLO which has been developed over the last five years has many capabilities. It can solve the steady or unsteady, inviscid or viscous equations of motion in two dimensions, with extensions to include quasi-three-dimensional effects. It can handle many different kinds of flow unsteadiness; wake/rotor and potential/rotor interactions in which the unsteadiness is generated by unsteady inflow or outflow boundary conditions; stator/rotor interactions in which a full stage is calculated and the unsteadiness is caused by the relative motion of the stators and rotors; flutter, in which the unsteadiness is due to blade vibration. One novel feature of UNSFLO is its ability to treat arbitrary wake/rotor and stator/rotor pitch ratios, which in extreme cases requires the computation to be performed on multiple rotor passages. Another is the incorporation of highly accurate non-reflecting boundary conditions which minimize non-physical reflections at inflow and outflow boundaries. A third feature is the use of unstructured grids which, in combination with an advancing front grid generator [24], makes it possible to perform calculations on complex geometries, such as pylon/strut/outlet-guide-vane combinations.

Several papers have been written about different algorithmic components of UNSFLO, as well as the use of UNSFLO to investigate various unsteady flows. On the algorithm side the papers present the 'time-inclined' computational planes used to handle arbitrary stator/rotor pitch ratios [12]; the use of "time-inclined" computational planes for convergence acceleration [11, 6]; the stator/rotor interface treatment for a transonic interaction analysis [15]; the mathematical theory behind non-reflecting boundary conditions [14]. On the application side, UNSFLO has been used to look at compressor interaction [8]; shock propagation in a shock-wave/rotor interaction [19]; complex steady and unsteady pylon/strut/outlet-guide-vane flows [24]; unsteady heat transfer in a transonic turbine stator/rotor interaction [1]. There is also a comprehensive validation paper with a number of unsteady test cases [16].

This report describes in detail the numerical method used in UNSFLO. Chapter 2 derives the explicit, Lax-Wendroff algorithm which is used to calculate the unsteady, inviscid flow. It also discusses the use of an unstructured, pointered grid system, and the formulation of the numerical smoothing which is critical to the accuracy of the method. Chapter 3 introduces the concept of "time-inclined" computational planes to handle unsteady calculations with arbitrary stator/rotor pitch ratios. Full details are presented on how this changes the basic Lax-Wendroff algorithm. Chapter 4 shows how stator/rotor calculations are performed by calculating on two separate stator and rotor grids using relative flow variables. The two are coupled together through an interface region with moving cells. Chapter 5 presents the steady inflow and outflow boundary conditions, using non-reflecting boundary condition theory to achieve accurate results on very small domains. Chapter 6 gives the unsteady boundary conditions, which allow for the specification of incoming wakes and potential disturbances, and again use the non-reflecting theory to prevent artificial reflections of outgoing waves. Finally, Chapter 7 presents the viscous flow algorithm and full details on how it is coupled to the external inviscid flow calculation.

Chapter 2

Lax-Wendroff Algorithm on Unstructured Meshes

2.1 Unsteady Euler equations

The unsteady Euler equations, describing the motion of an inviscid, compressible gas in two dimensions, are

$$\frac{\partial U}{\partial t} = -\left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y}\right), \tag{2.1}$$

where U, F and G are four component vectors given by,

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad F = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \end{pmatrix}, \quad G = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vH \end{pmatrix}. \tag{2.2}$$

The pressure p, and total enthalpy H, are related to the density ρ, velocity components u and v, and total energy per unit mass E by the following two equations which assume a perfect gas with a constant specific heat ratio γ.

$$p = (\gamma - 1)\,\rho\left(E - \frac{1}{2}(u^2 + v^2)\right) \tag{2.3}$$

$$H = E + \frac{p}{\rho}. \tag{2.4}$$

Additional equations which will be required are the definitions of the speed of sound, Mach number, stagnation pressure and stagnation density.

$$c = \sqrt{\frac{\gamma p}{\rho}} \tag{2.5}$$

$$M = \frac{\sqrt{u^2+v^2}}{c} \tag{2.6}$$

$$p_o = p\left(1+\frac{\gamma-1}{2}M^2\right)^{\gamma/(\gamma-1)} \tag{2.7}$$

$$\rho_o = \rho\left(1+\frac{\gamma-1}{2}M^2\right)^{1/(\gamma-1)} . \tag{2.8}$$

The flow variables are non-dimensionalized using the upstream stagnation density and stagnation speed of sound which leaves the equations unchanged and gives the following inlet stagnation quantities.

$$H = \frac{1}{\gamma-1}, \quad \rho_o = 1, \quad p_o = \frac{1}{\gamma}. \tag{2.9}$$

An extremely useful extension to the two-dimensional Euler equations, is the inclusion of a varying streamtube thickness in the third dimension. The resultant quasi-three-dimensional equations are

$$h\frac{\partial U}{\partial t} = -\left(\frac{\partial(hF)}{\partial x} + \frac{\partial(hG)}{\partial y}\right) + S, \tag{2.10}$$

where

$$S = \begin{pmatrix} 0 \\ p\frac{\partial h}{\partial x} \\ p\frac{\partial h}{\partial y} \\ 0 \end{pmatrix}, \tag{2.11}$$

and h is the streamtube thickness which in general varies only in the axial x-direction. This is the most important three-dimensional effect in axial turbomachinery, but in radial turbomachinery the radius change is also very important and one should include Coriolis and centrifugal body forces [5].

These equations also apply in a rotating frame of reference at constant radius if relative velocities, total energy and total enthalpy are used.

2.2 Unstructured meshes

Before beginning to present the numerical algorithm used to solve the unsteady Euler equations, it is necessary to first discuss the organization of the computational data. Historically, most algorithms and programs in computational fluid dynamics have been developed on structured meshes, which means that the computational grid is usually composed of quadrilateral cells which are arranged in a logically rectangular manner and so each grid coordinate has an (i, j) index. Each flow variable is then defined at a particular point in a two-dimensional array, and neighboring points in the array structure are also neighboring points in the physical computational domain.

The alternative approach of using unstructured meshes, is the one which has commonly been adopted in structural and thermal finite element analysis. Increasingly this approach is also being used in computational fluid dynamics [25], and it is the approach used here with UNSFLO. Each grid coordinate (and its associated flow variables) is associated with a particular index in a one-dimensional array. There are also one-dimensional arrays of cell-related variables, with one set of cell variables being pointers given the indices of the grid nodes which form the corners of the cell. As will be described in the next section, the flow algorithm is arranged to be implemented in a cell-by-cell manner, sweeping through the list of cells gathering the values from their corner nodes performing the necessary calculations and then distributing the appropriate changes in the flow variables back to the corner nodes.

There are several reasons for choosing to use unstructured meshes. They offer great flexibility in grid generation for complex geometries, and effectively separate the process of grid generation from the flow solver, since any structured mesh can always be turned into an unstructured mesh. For added flexibility, the mesh used in UNSFLO can be a mixture of quadrilateral and triangular cells. Another related advantage lies in the technique of adaptive meshes in which grids are locally refined through the addition of extra grid points to resolve high-gradient features such as shocks and slip surfaces. This is relatively easily done for unstructured meshes [25, 7], but can only be done in a very limited and inefficient way on structured meshes. Proponents and opponents of unstructured meshes disagree on both the relative ease of programming and the vector/parallel efficiency of the flow solvers. The first depends on the complexity of the geometry, since structured programs are simple for ducts, but get extremely complicated when dealing with entire aircraft, whereas the unstructured flow solvers do not change. The second point depends on the trade-off between the cost of gather/scatter operations required to address the flow variables at the corners of the computational cells, versus the increased efficiency of DO-loops which span the total number of cells rather than the number in any one particular direction. The only drawback of unstructured meshes is that they are generally unsuitable in applications where ADI algorithm are required, since those algorithms require connection lists of nodes along implicit inversion lines. Even in this case, however, it is possible to construct an appropriate partially structured mesh [28].

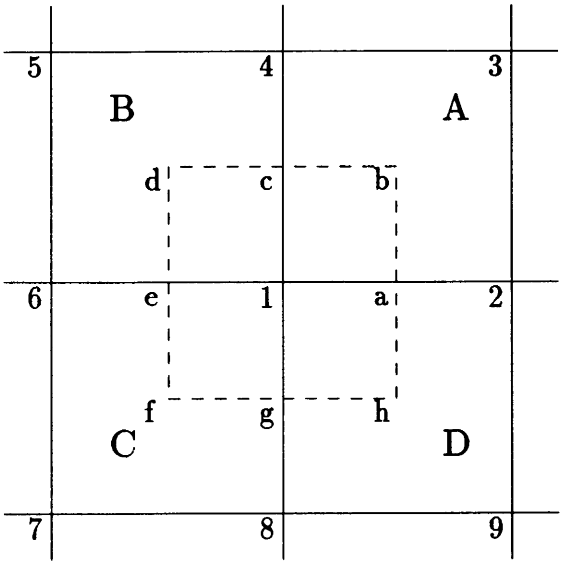

Figure 2.1: Control volume for quadrilateral Lax-Wendroff scheme

2.3 Quadrilateral Lax-Wendroff algorithm

The quadrilateral Lax-Wendroff scheme is very similar to that used by Ni [26] and Hall [17], but differs in precise detail for non-uniform grids. The algorithm will first be described for the two-dimensional Euler equations, and then the modified version for the quasi-three-dimensional equations will be given.

The second-order Taylor series expansion for $U^{n+1} = U((n+1)\Delta t)$ can be written as,

$$U^{n+1} = U^n + \Delta t \left(\frac{\partial U}{\partial t}\right)^n + \tfrac{1}{2}\Delta t^2 \left(\frac{\partial^2 U}{\partial t^2}\right)^n . \tag{2.12}$$

Substituting from Eq. (2.1) and changing the order of differentiation yields,

$$U^{n+1} = U^n - \Delta t \left(\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y}\right)^n - \frac{\Delta t}{2}\left(\frac{\partial}{\partial x}\Delta F + \frac{\partial}{\partial y}\Delta G\right)^n , \tag{2.13}$$

where

$$\Delta F^n = \Delta t \frac{\partial F}{\partial t}, \qquad \Delta G^n = \Delta t \frac{\partial G}{\partial t}. \tag{2.14}$$

Now consider the cells shown in Fig. 2.1. The grid nodes are numbered, and the letters correspond to other points which will be used in explaining the method. a,c,e,g are located at the center of their respective faces, and b,d,f,h are located at the center of their respective cells. Integrating Eq. (2.13) over the cell a-b-c-d-e-f-g-h-a, and applying Green's theorem, gives

$$\delta U_1 = -\frac{\Delta t}{A_1}\left(\oint (F\,dy\ -\ G\,dx) + \tfrac{1}{2}\oint (\Delta F\,dy\ -\ \Delta G\,dx)\right). \tag{2.15}$$

The first term can be split into four separate contour integrals around 1-a-b-c-1 etc., and each of these can be approximated as a quarter of the contour integral around the larger cells 1-2-3-4-1 etc., which are labelled A,B,C,D for convenience. In this manner δU_1 can be split into four parts,

$$\delta U_1 = \delta U_{1A} + \delta U_{1B} + \delta U_{1C} + \delta U_{1D} \tag{2.16}$$

where,

$$\begin{aligned} \delta U_{1A} &= \frac{1}{A_1}\left(-\frac{\Delta t}{4}\oint_{cellA}(F\,dy\ -\ G\,dx) - \frac{\Delta t}{2}\int_{a-b-c}(\Delta F\,dy\ -\ \Delta G\,dx)\right) \\ &= \frac{1}{4A_1}\left(A_A \Delta U_A - \Delta t\,(\Delta F_A\ (y_4 - y_2) - \Delta G_A\ (x_4 - x_2))\right) \end{aligned} \tag{2.17}$$

and the other terms are defined similarly.

ΔU_A is obtained by a simple trapezoidal integration around cell A. Defining the following face lengths in a counterclockwise direction,

$$\begin{aligned} \Delta x_{21} &= x_2 - x_1 \\ \Delta x_{32} &= x_3 - x_2 \\ \Delta x_{43} &= x_4 - x_3 \\ \Delta x_{14} &= x_1 - x_4 \end{aligned} \tag{2.18}$$

$$\begin{aligned} \Delta y_{21} &= y_2 - y_1 \\ \Delta y_{32} &= y_3 - y_2 \\ \Delta y_{43} &= y_4 - y_3 \\ \Delta y_{14} &= y_1 - y_4 \end{aligned} \tag{2.19}$$

the equation for ΔU_A is

$$\begin{aligned} \Delta U_A = \frac{\Delta t}{A_A}\Big(&-\tfrac{1}{2}(F_1 + F_2)\Delta y_{21} + \tfrac{1}{2}(G_1 + G_2)\Delta x_{21} \\ &-\tfrac{1}{2}(F_2 + F_3)\Delta y_{32} + \tfrac{1}{2}(G_2 + G_3)\Delta x_{32} \\ &-\tfrac{1}{2}(F_3 + F_4)\Delta y_{43} + \tfrac{1}{2}(G_3 + G_4)\Delta x_{43} \\ &-\tfrac{1}{2}(F_4 + F_1)\Delta y_{14} + \tfrac{1}{2}(G_4 + G_1)\Delta x_{14}\Big). \end{aligned} \tag{2.20}$$

The cell area A_A is obtained from

$$A_A = \frac{1}{2}\left(\,(x_3 - x_1)(y_4 - y_2) - (x_4 - x_2)(y_3 - y_1)\,\right), \tag{2.21}$$

and the area A_1 associated with node 1 is simply an average of the four cells A_A,A_B,A_C and A_D.

ΔF_A and ΔG_A are obtained from

$$\Delta F_A = \left(\frac{\partial F}{\partial U}\right)_A \Delta U_A, \quad \Delta G_A = \left(\frac{\partial G}{\partial U}\right)_A \Delta U_A, \tag{2.22}$$

with the Jacobians being evaluated using U_A, the cell average of the four nodes. For computational efficiency it is best not to actually form the Jacobian matrix and perform the matrix-vector multiplication. Instead, the following equations are used.

$$\begin{aligned}
\Delta u &= (\Delta(\rho u) - u\Delta\rho)/\rho \\
\Delta v &= (\Delta(\rho v) - v\Delta\rho)/\rho \\
\Delta p &= (\gamma-1)\left(\Delta(\rho E) - u\Delta(\rho u) - v\Delta(\rho v) + \tfrac{1}{2}(u^2+v^2)\Delta\rho\right)
\end{aligned} \tag{2.23}$$

$$\begin{aligned}
\Delta F_1 &= \Delta(\rho u) \\
\Delta F_2 &= u\Delta(\rho u) + \rho u\Delta u + \Delta p \\
\Delta F_3 &= u\Delta(\rho v) + \rho v\Delta u \\
\Delta F_4 &= u(\Delta(\rho E) + \Delta p) + \rho H\Delta u
\end{aligned} \tag{2.24}$$

$$\begin{aligned}
\Delta G_1 &= \Delta(\rho v) \\
\Delta G_2 &= v\Delta(\rho u) + \rho u\Delta v \\
\Delta G_3 &= v\Delta(\rho v) + \rho v\Delta v + \Delta p \\
\Delta G_4 &= v(\Delta(\rho E) + \Delta p) + \rho H\Delta v.
\end{aligned} \tag{2.25}$$

By construction, the Lax-Wendroff scheme as formulated here is ideally suited for calculations on an unstructured grid. In the program the algorithm is accomplished in three passes. The first pass calculates F and G at all nodes. The second pass calculates for each cell the $\Delta U, \Delta F, \Delta G$ and then the contributions to the changes at each of its nodes. The third pass adds the changes onto the flow variables at each node and evaluates the convergence checks.

The algorithm is also very suitable for calculations on a computer with either vector pipelines or multi-processors. In this case the middle pass is split into several passes.

The cells are "colored" such that there are no two cells of the same color touching. Then each pass calculates the update contributions from one color of cell. In this manner there are no conflicts from two cells sending their contributions to the same node at the same time. The coloring algorithm need not be too sophisticated. If there are 10,000 cells then using ten colors is not much less efficient than using four.

Several modifications are needed to convert the basic two-dimensional algorithm into the quasi-three-dimensional form. The first is that all cell areas become volumes.

$$A'_A = \tfrac{1}{8}(h_1+h_2+h_3+h_4)\left(\,(x_3-x_1)(y_4-y_2) - (x_4-x_2)(y_3-y_1)\,\right). \tag{2.26}$$

It is also helpful to define the following face area terms.

$$\begin{aligned}
\Delta x'_{21} &= \tfrac{1}{2}(h_1+h_2)\Delta x_{21}\\
\Delta x'_{32} &= \tfrac{1}{2}(h_2+h_3)\Delta x_{32}\\
\Delta x'_{43} &= \tfrac{1}{2}(h_3+h_4)\Delta x_{43}\\
\Delta x'_{14} &= \tfrac{1}{2}(h_4+h_1)\Delta x_{14}
\end{aligned} \tag{2.27}$$

$$\begin{aligned}
\Delta y'_{21} &= \tfrac{1}{2}(h_1+h_2)\Delta y_{21}\\
\Delta y'_{32} &= \tfrac{1}{2}(h_2+h_3)\Delta y_{32}\\
\Delta y'_{43} &= \tfrac{1}{2}(h_3+h_4)\Delta y_{43}\\
\Delta y'_{14} &= \tfrac{1}{2}(h_4+h_1)\Delta y_{14}
\end{aligned} \tag{2.28}$$

$$\begin{aligned}
\Delta x''_{24} &= \tfrac{1}{2}(\Delta x'_{21} + \Delta x'_{14} - \Delta x'_{43} - \Delta x'_{32})\\
\Delta x''_{31} &= \tfrac{1}{2}(\Delta x'_{32} + \Delta x'_{21} - \Delta x'_{14} - \Delta x'_{43})
\end{aligned} \tag{2.29}$$

$$\begin{aligned}
\Delta y''_{24} &= \tfrac{1}{2}(\Delta y'_{21} + \Delta y'_{14} - \Delta y'_{43} - \Delta y'_{32})\\
\Delta y''_{31} &= \tfrac{1}{2}(\Delta y'_{32} + \Delta y'_{21} - \Delta y'_{14} - \Delta y'_{43}).
\end{aligned} \tag{2.30}$$

Next, in the definition of ΔU_A, there are two changes, one due to the multiplication of the fluxes F and G by the average streamtube thickness at the centers of the faces, and the other due to the inclusion of the source term S. This latter term can be approximated as

$$\int\!\!\int S dx\, dy \;\approx\; p_A \int\!\!\int \begin{pmatrix} 0 \\ \frac{\partial h}{\partial x} \\ \frac{\partial h}{\partial y} \\ 0 \end{pmatrix} dx\, dy$$

$$
\begin{aligned}
&= p_A \begin{pmatrix} 0 \\ \oint h\,dy \\ -\oint h\,dx \\ 0 \end{pmatrix} \\
&\approx \tfrac{1}{4}(p_1+p_2+p_3+p_4) \begin{pmatrix} 0 \\ \Delta y'_{21}+\Delta y'_{32}+\Delta y'_{43}+\Delta y'_{14} \\ -(\Delta x'_{21}+\Delta x'_{32}+\Delta x'_{43}+\Delta x'_{14}) \\ 0 \end{pmatrix}. \qquad (2.31)
\end{aligned}
$$

Inserting these two changes into the flux residual equation (2.20) gives, after some tedious algebra,

$$
\Delta U_A = -\frac{\Delta t}{2A'_A}\begin{pmatrix} \hat{V}_1\rho_1 + \hat{V}_2\rho_2 + \hat{V}_3\rho_3 + \hat{V}_4\rho_4 \\ \hat{V}_1(\rho u)_1 + \hat{V}_2(\rho u)_2 + \hat{V}_3(\rho u)_3 + \hat{V}_4(\rho u)_4 - (p_3-p_1)\Delta y''_{24} - (p_4-p_2)\Delta y''_{31} \\ \hat{V}_1(\rho v)_1 + \hat{V}_2(\rho v)_2 + \hat{V}_3(\rho v)_3 + \hat{V}_4(\rho v)_4 + (p_3-p_1)\Delta x''_{24} + (p_4-p_2)\Delta x''_{31} \\ \hat{V}_1(\rho H)_1 + \hat{V}_2(\rho H)_2 + \hat{V}_3(\rho H)_3 + \hat{V}_4(\rho H)_4 \end{pmatrix} \qquad (2.32)
$$

where the $\hat{V}$ terms are volume fluxes defined by

$$
\begin{aligned}
\hat{V}_1 &= u_1(\Delta y'_{21}+\Delta y'_{14}) - v_1(\Delta x'_{21}+\Delta x'_{14}) \\
\hat{V}_2 &= u_2(\Delta y'_{32}+\Delta y'_{21}) - v_2(\Delta x'_{32}+\Delta x'_{21}) \\
\hat{V}_3 &= u_3(\Delta y'_{43}+\Delta y'_{32}) - v_3(\Delta x'_{43}+\Delta x'_{32}) \qquad (2.33) \\
\hat{V}_4 &= u_4(\Delta y'_{14}+\Delta y'_{43}) - v_4(\Delta x'_{14}+\Delta x'_{43}).
\end{aligned}
$$

ΔF and ΔG are calculated in exactly the same manner as before, but the second order flux terms are slightly modified, so that the distributed changes to the nodes are

$$
\begin{aligned}
\delta U_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\ \Delta y''_{24} - \tfrac{1}{4}\Delta G_A\ \Delta x''_{24}\right) \\
\delta U_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\ \Delta y''_{31} - \tfrac{1}{4}\Delta G_A\ \Delta x''_{31}\right) \\
\delta U_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A - \tfrac{1}{4}\Delta F_A\ \Delta y''_{24} + \tfrac{1}{4}\Delta G_A\ \Delta x''_{24}\right) \qquad (2.34) \\
\delta U_{4A} &= \left(\frac{\Delta t}{A'}\right)_4 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A - \tfrac{1}{4}\Delta F_A\ \Delta y''_{31} + \tfrac{1}{4}\Delta G_A\ \Delta x''_{31}\right).
\end{aligned}
$$

Note the rearrangement of the Δt terms. Expressed in this way the numerical scheme remains conservative for steady-state calculations which use spatially varying timesteps to achieve faster convergence.

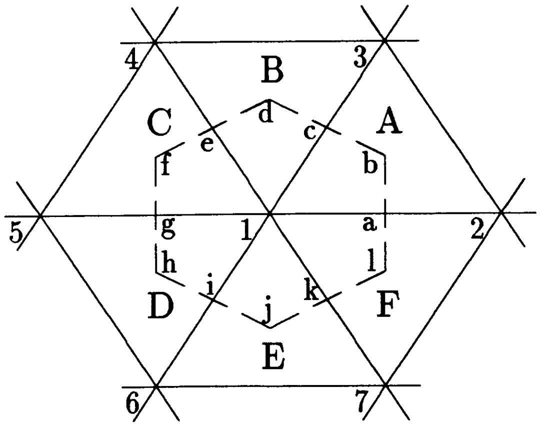

Figure 2.2: Control volume for triangular Lax-Wendroff scheme

2.4 Triangular Lax-Wendroff algorithm

The quadrilateral Lax-Wendroff algorithm has been extended by Lindquist [23] for triangular cells. The algorithm is very similar to the quadrilateral method. Fig. 2.2 shows the triangular control volume for a situation in which six triangles meet at node 1. a,c,e,g,i,k are located at the center of their respective faces and b,d,f,h, j,l are at the centers of their respective triangular cells. The counter-clockwise lengths of the faces of cell A are defined by

$$\begin{aligned} \Delta x_{21} &= x_2 - x_1 \\ \Delta x_{32} &= x_3 - x_2 \\ \Delta x_{13} &= x_1 - x_3 \end{aligned} \tag{2.35}$$

$$\begin{aligned} \Delta y_{21} &= y_2 - y_1 \\ \Delta y_{32} &= y_3 - y_? \\ \Delta y_{13} &= y_1 - y_3. \end{aligned} \tag{2.36}$$

The volume and face areas of cell A are defined by

$$A'_A = \tfrac{1}{6}(h_1+h_2+h_3)(\Delta x_{21}\Delta y_{32} - \Delta x_{32}\Delta y_{21}), \tag{2.37}$$

and

$$\begin{aligned} \Delta x'_{21} &= \tfrac{1}{2}(h_1+h_2)\Delta x_{21} \\ \Delta x'_{32} &= \tfrac{1}{2}(h_2+h_3)\Delta x_{32} \\ \Delta x'_{13} &= \tfrac{1}{2}(h_3+h_1)\Delta x_{13} \end{aligned} \tag{2.38}$$

$$\begin{aligned}\Delta y'_{21} &= \tfrac{1}{2}(h_1+h_2)\Delta y_{21}\\ \Delta y'_{32} &= \tfrac{1}{2}(h_2+h_3)\Delta y_{32}\\ \Delta y'_{13} &= \tfrac{1}{2}(h_3+h_1)\Delta y_{13}\end{aligned} \tag{2.39}$$

$$\begin{aligned}\Delta x''_{12} &= -\Delta x'_{21} + \tfrac{1}{3}(\Delta x'_{21}+\Delta x'_{32}+\Delta x'_{13})\\ \Delta x''_{23} &= -\Delta x'_{32} + \tfrac{1}{3}(\Delta x'_{21}+\Delta x'_{32}+\Delta x'_{13})\\ \Delta x''_{31} &= -\Delta x'_{13} + \tfrac{1}{3}(\Delta x'_{21}+\Delta x'_{32}+\Delta x'_{13})\end{aligned} \tag{2.40}$$

$$\begin{aligned}\Delta y''_{12} &= -\Delta y'_{21} + \tfrac{1}{3}(\Delta y'_{21}+\Delta y'_{32}+\Delta y'_{13})\\ \Delta y''_{23} &= -\Delta y'_{32} + \tfrac{1}{3}(\Delta y'_{21}+\Delta y'_{32}+\Delta y'_{13})\\ \Delta y''_{31} &= -\Delta y'_{13} + \tfrac{1}{3}(\Delta y'_{21}+\Delta y'_{32}+\Delta y'_{13}).\end{aligned} \tag{2.41}$$

The change ΔU_A in cell A is given by

$$\Delta U_A = -\frac{\Delta t}{2A_A}\begin{pmatrix} \hat{V}_1\rho_1 + \hat{V}_2\rho_2 + \hat{V}_3\rho_3 \\ \hat{V}_1(\rho u)_1 + \hat{V}_2(\rho u)_2 + \hat{V}_3(\rho u)_3 + p_1\Delta y''_{23} + p_2\Delta y''_{31} + p_3\Delta y''_{12} \\ \hat{V}_1(\rho v)_1 + \hat{V}_2(\rho v)_2 + \hat{V}_3(\rho v)_3 - p_1\Delta x''_{23} - p_2\Delta x''_{31} - p_3\Delta x''_{12} \\ \hat{V}_1(\rho H)_1+\hat{V}_2(\rho H)_2+\hat{V}_3(\rho H)_3 \end{pmatrix} \tag{2.42}$$

where the $\hat{V}$ terms are volume fluxes defined by

$$\begin{aligned}\hat{V}_1 &= u_1(\Delta y'_{21}+\Delta y'_{13}) - v_1(\Delta x'_{21}+\Delta x'_{13})\\ \hat{V}_2 &= u_2(\Delta y'_{32}+\Delta y'_{21}) - v_2(\Delta x'_{32}+\Delta x'_{21})\\ \hat{V}_3 &= u_3(\Delta y'_{13}+\Delta y'_{32}) - v_3(\Delta x'_{13}+\Delta x'_{32}).\end{aligned} \tag{2.43}$$

ΔF and ΔG are calculated as before, but the first order term in the distributed changes is slightly different since the flux residual has to be distributed in equal thirds to the three corner nodes, and the second order term is different because of the geometric differences between quadrilaterals and triangles.

$$\begin{aligned}\delta U_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{3}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\ \Delta y''_{23} - \tfrac{1}{4}\Delta G_A\ \Delta x''_{23}\right)\\ \delta U_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{3}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\ \Delta y''_{31} - \tfrac{1}{4}\Delta G_A\ \Delta x''_{31}\right)\\ \delta U_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{3}\left(\frac{A'}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\ \Delta y''_{12} - \tfrac{1}{4}\Delta G_A\ \Delta x''_{12}\right).\end{aligned} \tag{2.44}$$

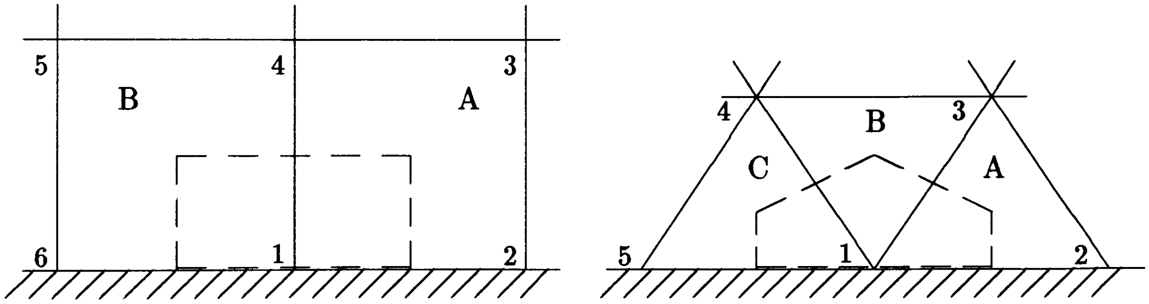

Figure 2.3: Cells at a wall

2.5 Wall boundary conditions

At solid walls the analytic boundary condition is that there is no flow normal to the wall. Computationally this is implemented easily by setting to zero the mass flux through wall faces when calculating the change ΔU in any cell which has a solid wall face. To maintain vector efficiency, the node-numbering of cells with wall faces is altered if necessary to ensure that the wall face is the face between nodes 1 and 2. Also the cell-coloring algorithm discussed earlier is modified to ensure that all cells of a particular color either *do* have wall faces, or *do not* have wall faces. Then, when looping over cells of a color with wall faces, the definitions of the volume fluxes $\hat{V}_1$ and $\hat{V}_2$ are changed to

$$\begin{aligned} \hat{V}_1 &= u_1 \Delta y'_{14} - v_1 \Delta x'_{14} \\ \hat{V}_2 &= u_2 \Delta y'_{32} - v_2 \Delta x'_{32} \end{aligned} \tag{2.45}$$

for quadrilateral cells, and

$$\begin{aligned} \hat{V}_1 &= u_1 \Delta y'_{13} - v_1 \Delta x'_{13} \\ \hat{V}_2 &= u_2 \Delta y'_{32} - v_2 \Delta x'_{32} \end{aligned} \tag{2.46}$$

for triangular cells.

In addition to setting the normal mass flux to zero in the residual evaluation, at the end of each timestep the velocity is also made tangent to the wall at each surface grid node by eliminating the component of the momentum normal to the surface.

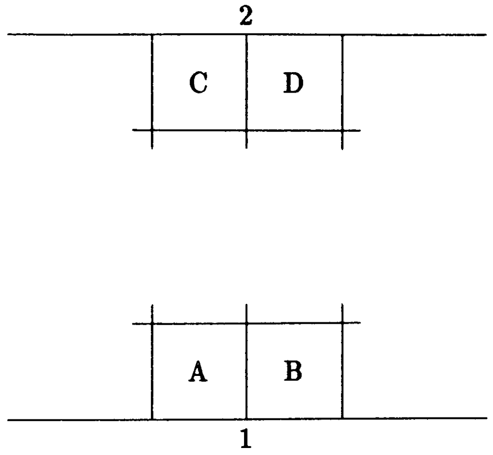

Figure 2.4: Grid nodes in periodic boundary condition

2.6 Periodic boundary condition

The periodic condition for steady flows, and unsteady flows with equal stator and rotor pitches, is implemented by adding the update contributions that one periodic node 1 obtains from its contributing cells A and B, see Figure 2.4, to the contributions that the corresponding upper periodic node 2 obtains from its cells C and D, and using the sum to update the flow variables at 1 and 2.

2.7 Numerical smoothing

Two types of numerical smoothing are added to the basic Lax-Wendroff algorithm. To stabilize shock calculations and prevent large overshoots a carefully tailored second-difference shock smoothing is used. Also, unwanted high-frequency waves in smooth flow regions are suppressed by adding a form of fourth-difference damping.

2.7.1 Shock smoothing

The shock smoothing is similar to the second difference smoothing used by Ni [26], but with an adaptive smoothing coefficent based upon an idea of von Neumann and Richtmeyer [31].

The internal structure of a physical shock is determined by the balance of the inviscid flux and the flux due to the bulk viscosity of the fluid. Thus von Neumann and

Richtmeyer suggested modifying the Euler equations, Eq. (2.1), into the following form,

$$\frac{\partial U}{\partial t} = -\left(\frac{\partial(F-F^v)}{\partial x} + \frac{\partial(G-G^v)}{\partial y}\right), \tag{2.47}$$

where F^v and G^v are

$$F^v = \begin{pmatrix} 0 \\ \kappa\nabla.\vec{u} \\ 0 \\ 0 \end{pmatrix}, \quad G^v = \begin{pmatrix} 0 \\ 0 \\ \kappa\nabla.\vec{u} \\ 0 \end{pmatrix}. \tag{2.48}$$

The shock width is proportional to the bulk viscosity κ divided by the magnitude of the velocity jump across the shock, and so they proposed the following formula for κ.

$$\kappa = \begin{cases} \rho l^2|\nabla.\vec{u}| & ,\nabla.\vec{u} < 0 \\ 0 & ,\nabla.\vec{u} > 0 \end{cases} \tag{2.49}$$

Making the viscosity zero when the flow divergence is positive prevents smoothing of expansion regions. The variable l is the desired shock width which is chosen to be proportional to the local mesh spacing.

The shock smoothing in UNSFLO starts with Ni's second difference smoothing, which can be written as an additional distribution from each cell to its corner nodes. Using the same labelling system as the description of the basic algorithm, the additional smoothing distribution to node 1 due to cell A is

$$[(\delta U)_{1A}]_{smoothing} = -\kappa\left(\frac{\Delta t}{A}\right)_1\left(\frac{A}{\Delta t}\right)_A(U_1 - U_A). \tag{2.50}$$

In this equation U_A is the average value of U in cell A, defined by a simple arithmetic average of the nodal values. If κ was taken to have a small, positive, uniform value, then this smoothing would be very similar to Ni's smoothing as described in his original paper [26]. However, in UNSFLO it is defined to depend upon the flow divergence in a manner based on the idea of von Neumann and Richtmeyer.

Firstly, a scaled flow divergence in cell A is defined by

$$\mathrm{div}(\vec{u}) = \frac{(u_1-u_3)\Delta y_{24} - (v_1-v_3)\Delta x_{24} + (u_2-u_4)\Delta y_{31} - (v_2-v_4)\Delta x_{31}}{c\ \sqrt{\Delta x_{24}\Delta y_{31} - \Delta x_{31}\Delta y_{24}}} \tag{2.51}$$

for quadrilateral cells, and

$$\mathrm{div}(\vec{u}) = \frac{u_1\Delta y_{23} - v_1\Delta x_{23} + u_2\Delta y_{31} - v_2\Delta x_{31} + u_3\Delta y_{12} - v_3\Delta x_{12}}{c\ \sqrt{\Delta x_{12}\Delta y_{23} - \Delta x_{23}\Delta y_{12}}} \tag{2.52}$$

for triangular cells. These definitions mean that in smooth regions $\mathrm{div}(\vec{u})$ is approximately the flow divergence multiplied by a cell length, and in regions with a discontinuity due to a shock it is approximately the velocity jump across the shock.

Next, κ is defined by

$$\kappa = \min\Big(0.02,\ \ \max\Big(\nu^{(2)},\ \ -0.1M^2\mathrm{div}(\vec{u}),\ \ 0.1M^2(\mathrm{div}(\vec{u})-0.2),\ \ 0.02(M^2-2)\Big)\Big) \tag{2.53}$$

The different terms in the above definition require explanantion. The first term sets an upper limit on the magnitude of κ; this is needed to prevent a numerical parabolic instability. The second term is a constant which is usually zero, but can be set by the user to be a small positive constant, in which case it acts like Ni's fixed-coefficient smoothing. This is usually done only when there is some difficulty in performing the computation without this smoothing, which might happen if there is some excessively strong flow transient. The third term is the regular bulk viscosity term which is positive only when the flow is decelerating and the divergence is negative. The Mach number is used to prevent excessive smoothing at stagnation points. The fourth term is designed to prevent the possibility of expansion shocks. It is positive only when the flow is accelerating strongly, and in almost all computations this term will be zero throughout the flow field. The final term introduces smoothing when the local Mach number exceeds $\sqrt{2}$, which is above the values to be expected in most turbomachinery calculations. This term is included to cope with particular nasty transients in steady-state calculations without having to resort to a non-zero value for $\nu^{(2)}$.

It should be clear to the reader that there is a great deal of empiricism and practical experience built into the above shock smoothing formulation. All of the constants have evolved over four years of calculations, and the values above work for a wide variety of steady and unsteady turbomachinery flows. One final important observation is that in smooth flow regions $\mathrm{div}(\vec{u})$ is proportional to the local cell dimension and so if $\nu^{(2)}$ is zero and the Mach number is below $\sqrt{2}$, then the shock smoothing is second order in magnitude and does not alter the global order of accuracy.

2.7.2 Fourth difference smoothing

Conceptually, the fourth difference smoothing corresponds to adding a term of the form

$$-\nabla.\left(l\nabla(l^2\nabla^2 U)\right)$$

to the right-hand-side of Eq. (2.1) or Eq. (2.10). The variable l is a length which is comparable to the local cell length, and so the error produced by this smoothing will be second-order at worst. As stated earlier, the function of this smoothing is to suppress certain highly oscillatory steady-state modes which are otherwise allowed by the basic Lax-Wendroff scheme.

The first step in formulating the fourth difference smoothing is to calculate a discrete approximation to a Laplacian of the state vector U at each node. The average gradients of U in a triangular cell can be found by an application of Green's theorem.

$$\begin{aligned}\left(\frac{\partial U}{\partial x}\right)_A &= \frac{1}{A_A}\oint_{cell A} U\,dy \\ &= -\frac{1}{2A_A}(U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21}) \qquad (2.54)\end{aligned}$$

$$\begin{aligned}\left(\frac{\partial U}{\partial y}\right)_A &= -\frac{1}{A_A}\oint_{cell A} U\,dx \\ &= \frac{1}{2A_A}(U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21}). \qquad (2.55)\end{aligned}$$

Having obtained the cell gradients, a second difference of U at node 1 can be defined by

$$(D^2U)_1 \equiv 2A_1\nabla^2 U_1 \approx 2\oint\left(\frac{\partial U}{\partial x}\,dy - \frac{\partial U}{\partial y}\,dx\right). \qquad (2.56)$$

The line integral is around the same control volume used to assemble the second order flux terms in the Lax-Wendroff algorithm. In fully discrete form $(D^2U)_1$ is composed of contributions from all of the cells bordering node 1, and the contribution from cell A (as defined in Fig. 2.2) is

$$\begin{aligned}(D^2U)_{1A} &= -\frac{1}{2A_A}(\,(U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21})\Delta y_{32} + \\ &\qquad (U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21})\Delta x_{32}\,) \qquad (2.57)\end{aligned}$$

Similarly the contributions from cell A to D^2U at nodes 2 and 3 are

$$\begin{aligned}(D^2U)_{2A} &= -\frac{1}{2A_A}(\,(U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21})\Delta y_{13} + \\ &\qquad (U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21})\Delta x_{13}\,). \qquad (2.58)\end{aligned}$$

$$\begin{aligned}(D^2U)_{3A} &= -\frac{1}{2A_A}(\,(U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21})\Delta y_{21} + \\ &\qquad (U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21})\Delta x_{21}\,). \qquad (2.59)\end{aligned}$$

A noteworthy feature of this second difference operator D^2 is that when applied to a linear function U on an irregular grid, it returns a value of zero. The proof is

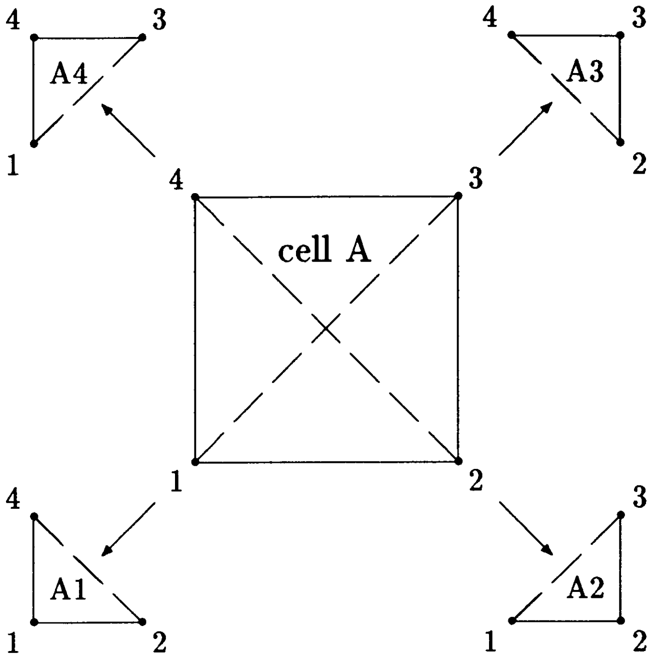

Figure 2.5: Division of quadrilateral cell into triangles

simple: if U is linear then ∇U must be uniform and so the line integral of the gradients around the node's control volume must give zero. For this to remain true at solid wall boundaries, the distribution formulae must be modified to include the contribution due to the control volume face lying on the wall surface. For example, in the case of cell A in Fig. 2.3, the modified distributions to nodes 1 and 2 are

$$\begin{aligned}(D^2U)_{1A} &= \frac{1}{2A_A}(\ (U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21})\Delta y_{13} + \\ &\qquad (U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21})\Delta x_{13}\ ). \end{aligned} \tag{2.60}$$

$$\begin{aligned}(D^2U)_{2A} &= \frac{1}{2A_A}(\ (U_1\Delta y_{32} + U_2\Delta y_{13} + U_3\Delta y_{21})\Delta y_{32} + \\ &\qquad (U_1\Delta x_{32} + U_2\Delta x_{13} + U_3\Delta x_{21})\Delta x_{32}\ ). \end{aligned} \tag{2.61}$$

The discussion so far has been for triangular cells. The natural extension to quadrilateral cells would involve computing the flux of ∇U in each cell through the usual control volume. However, this leads to a very poor smoothing operator because an odd-even sawtooth error mode (positive at nodes 1 and 3, and negative at nodes 2 and 4) would give a ∇U at the cell center which is zero. Thus this error mode would not be suppressed by the smoothing.

Instead, the approach for quadrilateral cells is to use the triangular algorithm by dividing each quadrilateral cell into four different triangles (as shown in Fig. 2.5) when calculating the distributions to each of the nodes, i.e $(D^2U)_{1A}$ is based upon triangle A1, $(D^2U)_{2A}$ is based upon triangle A2, $(D^2U)_{3A}$ is based upon triangle A3 and $(D^2U)_{4A}$ is based upon triangle A4.

In UNSFLO, the second difference function is evaluated by a preliminary sweep over all of the cells before beginning the Lax-Wendroff algorithm. The fourth difference smoothing is then built in as part of the Lax-Wendroff sweep. This part of the smoothing is very similar to the shock smoothing, except that we smooth D^2U instead of U itself. In each cell the average value of D^2U is calculated and then an extra distribution is sent to each node based upon the difference from the average value. For node 1 in either a quadrilateral or a triangular cell this addition is

$$[(\delta U)_{1A}]_{smoothing} = \nu^{(4)} \left(\frac{\Delta t}{A}\right)_1 \left(\frac{A}{\Delta t}\right)_A \left((D^2U)_1 - (D^2U)_A\right), \qquad (2.62)$$

with $\nu^{(4)}$ being a smoothing coefficient whose value is typically taken to be 0.001.

2.8 Timestep

A conservative estimate for the maximum stable timestep in each cell is given by

$$\begin{aligned} \frac{2A}{\Delta t_{max}} = & \; |u\Delta y_{21} - v\Delta x_{21}| + c\sqrt{\Delta y_{21}^2 + \Delta x_{21}^2} + \\ & |u\Delta y_{32} - v\Delta x_{32}| + c\sqrt{\Delta y_{32}^2 + \Delta x_{32}^2} + \\ & |u\Delta y_{43} - v\Delta x_{43}| + c\sqrt{\Delta y_{43}^2 + \Delta x_{43}^2} + \\ & |u\Delta y_{14} - v\Delta x_{14}| + c\sqrt{\Delta y_{14}^2 + \Delta x_{14}^2} \end{aligned} \qquad (2.63)$$

for quadrilateral cells, and

$$\begin{aligned} \frac{2A}{\Delta t_{max}} = & \; |u\Delta y_{21} - v\Delta x_{21}| + c\sqrt{\Delta y_{21}^2 + \Delta x_{21}^2} + \\ & |u\Delta y_{32} - v\Delta x_{32}| + c\sqrt{\Delta y_{32}^2 + \Delta x_{32}^2} + \\ & |u\Delta y_{13} - v\Delta x_{13}| + c\sqrt{\Delta y_{13}^2 + \Delta x_{13}^2} \end{aligned} \qquad (2.64)$$

for triangular cells. All terms are as defined earlier in this chapter, with u, v, c based upon the cell-averaged flow quantities.

For unsteady calculations the uniform global timestep is taken to be the minimum over all of the cells of the local maximum timestep, multiplied by a CFL number which is typically taken to be 0.9.

For steady calculations, local time steps are used to march to steady-state convergence as quickly as possible, so one used the local maximum timestep multiplied again by a CFL number which is typically 0.9. The area/timestep ratio associated with a grid node is then defined by

$$\left(\frac{A}{\Delta t}\right)_{node} = \sum_{cells} f_{cell} \left(\frac{A}{\Delta t}\right)_{cell}, \tag{2.65}$$

where the sum is over all of the neighboring cells and f_{cell} is $\frac{1}{4}$ for quadrilateral cells and $\frac{1}{3}$ for triangular cells.

2.9 Conservation

In earlier sections it has been stated that the Lax-Wendroff algorithm, as implemented here, is conservative in the solution of the nonlinear Euler equations. It is appropriate now to discuss what this statement means for both steady and unsteady flows, and to outline the proof of conservation for the given algorithm.

Steady-state solutions of the two-dimensional Euler equations satisfy the following integral equation, evaluated by a counter-clockwise integration around the domain.

$$\oint (F\,dy - G\,dx) = 0 \tag{2.66}$$

A steady, discrete solution is said to be conservative if, for any domain composed of a group of cells, there is a corresponding discrete equation which approximates this integral equation, and becomes equal to it in the limit of infinite grid resolution. The importance of conservation is due to the fact that this property guarantees the correct Rankine-Hugoniot jump relations across a shock and the correct treatment of other discontinuities such as slip lines (assuming the solution is sufficiently smooth away from the discontinuity). Thus conservation for nonlinear discontinuous solutions is similar to consistency for nonlinear smooth solutions as a requirement in order to obtain a discrete solution which will approach the analytic solution as the mesh is refined.

Similarly, unsteady analytic solutions satisfy the following equation.

$$\frac{d}{dt}\iint U\,dx\,dy + \oint (F\,dy - G\,dx) = 0 \tag{2.67}$$

Discrete solutions are conservative if they satisfy an equivalent discrete solution.

To prove that the Lax-Wendroff scheme is conservative we must show that

$$\sum_i \left(\frac{A}{\Delta t}\delta U\right)_i = \sum(\text{boundary fluxes}). \tag{2.68}$$

The change δU_i is equal to a sum of the contributions from all of the cells of which node i is a corner. The order of summation can then be interchanged to obtain

$$\sum_i \left(\frac{A}{\Delta t}\delta U\right)_i = \sum_{cells} (\text{sum of contributions to corner nodes}) \qquad (2.69)$$

The second order inviscid flux terms and both the shock and fourth-difference smoothing terms were written in such a way that the sum of their contributions to the corner nodes of a cell is zero. This leaves only the first order inviscid flux terms, and hence

$$\begin{aligned} \sum_i \left(\frac{A}{\Delta t}\delta U\right)_i &= \sum_{cells} \left(\frac{A}{\Delta t}\Delta U\right) \\ &= \sum_{cells} (\text{inviscid fluxes out of cell}) \qquad (2.70) \end{aligned}$$

The final step is the observation that the flux out of a particular cell across a particular face is equal and opposite to the flux out of the neighboring cell across the same face. Thus all interior fluxes cancel leaving the desired result, Eq. (2.68).

For quasi-three-dimensional flows the theory is modified slightly by the presence of the pressure source term in the momentum equations, and so the analytic and discrete conservation relations have an additional area integral/summation of the source term. The basic concept remains the same, however, and so does the proof.

Chapter 3

Time-Inclined Computational Planes

3.1 Lagged periodic condition

When the stator/rotor pitch ratio is unity the periodic boundary condition is simply,

$$U(x,y,t) = U(x,y+P,t), \tag{3.1}$$

meaning that what is happening on the lower periodic line is exactly the same as is happening on the upper periodic line at exactly the same time. When the stator pitch is different from the rotor pitch this has to be changed. Considering the case of wake/rotor interaction, in which the stator pitch is larger than the rotor pitch, then an incoming wake (moving downwards in the rotor frame) crosses the inlet boundary/upper periodic boundary junction a small time ΔT after the neighboring wake crosses the inlet/lower periodic junction.

Thus the inlet boundary conditions satisfy the lagged periodic condition,

$$U(x,y,t) = U(x,y+P_r,t+\Delta T), \tag{3.2}$$

where the time lag, ΔT, is equal to the difference in pitches divided by the rotor wheel speed.

$$\Delta T = (P_s - P_r)/V \tag{3.3}$$

The next step is to apply this lagged periodic condition to the upper and lower periodic lines. Strictly speaking this is an assumption about the nature of the flow

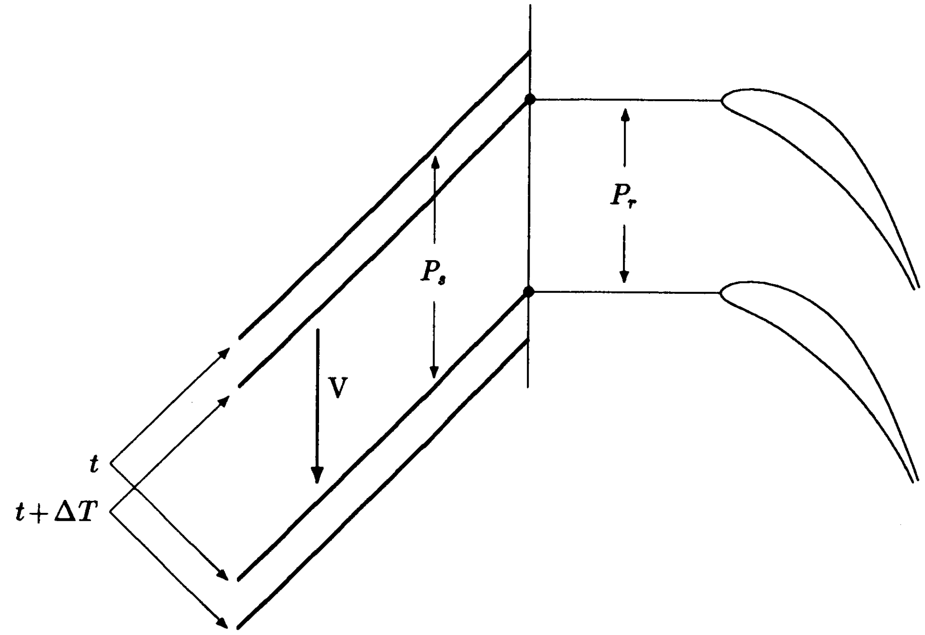

Figure 3.1: Origin of lagged periodic boundary condition

produced by the wake rotor interaction. There are many examples in mathematics (including some fairly simple examples in dynamics) in which periodic terms (either as forcing terms or time-varying coefficients) produce solutions with a subharmonic component, a component whose period is a multiple of the original period.

As an example, consider vortex shedding from a turbine row. Imposition of spatially periodic boundary conditions forces the solution to exhibit synchronous shedding, in which each blade sheds vortices of the same sign at the same time. However it may be true that in actuality the blades shed at the same time, but shed vortices of alternating sign, with one blade shedding a vortex of positive sign at the same time that its two neighbors shed vortices of negative sign. This would be an example of a spatial subharmonic whose period is $2P_r$. Mathematically, the spatially periodic solution produced by the program would be a valid solution to the unsteady Euler equations, but it would have a linear, subharmonic instability which would grow into the fully nonlinear, subharmonic shedding. In this case to compute the true solution would require a computational domain spanning two blade passages.

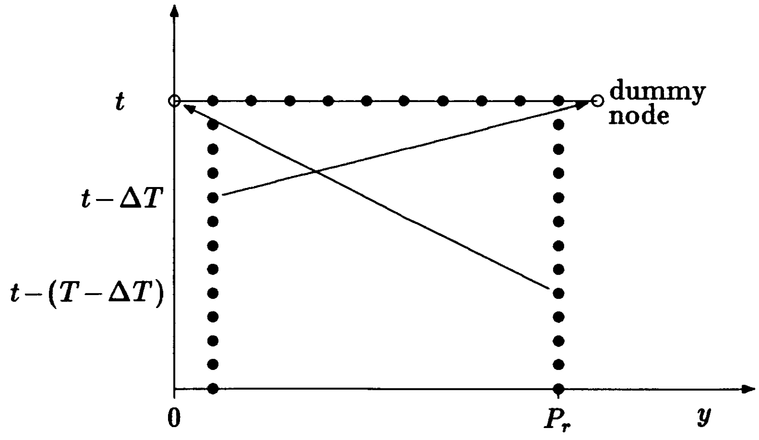

Figure 3.2: Erdos' periodic boundary treatment

3.2 Erdos method

Erdos [9] was the first researcher to develop a solution to the problem of the lagged periodic boundary condition. As illustrated in Figure 3.2, his procedure involves setting values at dummy points along each periodic line from stored values at points along the other periodic line at earlier times. The value at the dummy point on the upper periodic line is obtained from the equation

$$U(x,y,t) = U(x,y-P_r,t-\Delta T). \tag{3.4}$$

To obtain the value on the lower periodic line, it must be assumed that the flow is periodic in time, with period equal to the blade passing period $T = P_s/V$. With this assumption, it follows that

$$U(x,y,t) = U(x,y+P_r,t+\Delta T) = U(x,y+P_r,t-(T-\Delta T)), \tag{3.5}$$

The implementation of this requires storing the full solution along the periodic lines for a whole period. This can involve a considerable amount of storage. However, the primary drawback of this method is the assumption of periodicity in time. This is probably valid only when calculating inviscid flows. In viscous flows there are physical instabilities and oscillations, such as vortex shedding at the trailing edge, in which the frequency is not a multiple of the blade-passing frequency. In these situations Erdos' method would fail to converge to a consistent periodic solution. The new computational method using inclined computational planes avoids this assumption.

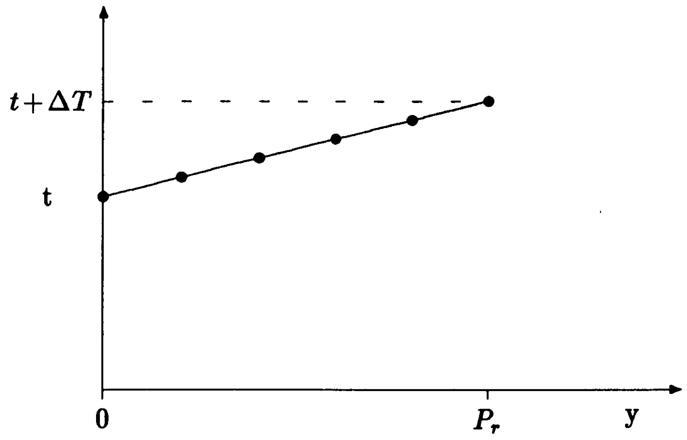

Figure 3.3: Concept of inclined computational plane

3.3 New computational method

Computationally it is very easy to enforce the spatial periodicity for steady flows, as described in an earlier section. For unsteady flows with the lagged periodicity condition it was desired to have as simple an implementation. This led to the following idea: suppose that instead of a computational "time level" being at a fixed time, it is sloped in time such that if a node at $y=0$ is at time t, then the corresponding periodic node at $y=P_r$ is at time $t+\Delta T$, and so once again one has simple spatial periodicity in this inclined computational plane. Fig. 3.3 illustrates this concept.

Mathematically this corresponds to the following coordinate transformation.

$$\begin{aligned} x' &= x \\ y' &= y \\ t' &= t - \left(\frac{\Delta T}{P_r}\right) y \end{aligned} \tag{3.6}$$

In this new coordinate system each computational plane corresponds to t' =constant. When one transforms the unsteady Euler equations the resultant equations are,

$$\frac{\partial}{\partial t'}(U - \lambda G) + \frac{\partial F}{\partial x'} + \frac{\partial G}{\partial y'} = 0 \tag{3.7}$$

with $\lambda = \Delta T/P_r$. Thus, the conservation state variables have changed from U to $U{-}\lambda G$. An alternative way of arriving at the same conclusion is to consider the conservation cell shown in Fig. 3.4 in the original (y,t) plane. The flux through the "time-like" face is $U\Delta y - G\Delta t = (U-\lambda G)\Delta y$.

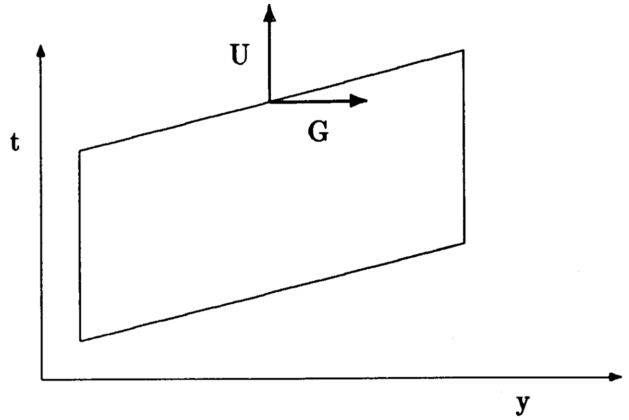

Figure 3.4: Inclined conservation cell

The change in the conservation variables requires just minor changes to the Lax-Wendroff algorithm, because fortunately one can calculate U from $Q = U - \lambda G$ in closed form for a perfect gas.

$$q_1 = \rho - \lambda \rho v \tag{3.8}$$

$$\begin{aligned} q_2 &= \rho u - \lambda \rho u v \\ &= q_1 u \end{aligned} \tag{3.9}$$

$$\begin{aligned} q_3 &= \rho v - \lambda(\rho v^2 + p) \\ &= q_1 v - \lambda p \end{aligned} \tag{3.10}$$

$$\begin{aligned} q_4 &= \frac{1}{\gamma-1} p + \frac{1}{2}\rho(u^2+v^2) - \lambda v \left(\frac{\gamma}{\gamma-1} p + \frac{1}{2}\rho(u^2+v^2)\right) \\ &= q_1 \left(\frac{1}{2}(u^2+v^2)\right) + \frac{1}{\gamma-1} p - \lambda \frac{\gamma}{\gamma-1} p v \end{aligned} \tag{3.11}$$

Eliminating u and v using the last three equations gives a quadratic equation for p.

$$Ap^2 - 2Bp + C = 0 \tag{3.12}$$

where,

$$\begin{aligned} A &= (\gamma+1)\lambda^2 \\ B &= q_1 - \lambda q_3 \\ C &= (\gamma-1)(2q_1 q_4 - q_2^2 - q_3^2) \end{aligned} \tag{3.13}$$

This has solutions,

$$p = \frac{C}{B \pm \sqrt{B^2 - AC}}. \tag{3.14}$$

The positive root is chosen because this gives the correct value in the limit $\lambda = 0$. u, v and ρ are then obtained from

$$\begin{aligned} u &= \frac{q_2}{q_1} && (3.15) \\ v &= \frac{q_3 + \lambda p}{q_1} && (3.16) \\ \rho &= \frac{q_1}{1 - \lambda v}. && (3.17) \end{aligned}$$

Eqs. (3.14)-(3.17) can also be linearized to obtain the following equations.

$$\begin{aligned} \Delta p &= \frac{(\gamma-1)\left(\frac{1}{2}(u^2+v^2)q_1\,\Delta q_1 + q_1\,\Delta q_4 - q_2\Delta q_2 - q_3\Delta q_3\right) + \lambda p(\Delta q_3 - \gamma v\Delta q_1)}{\rho(1-\lambda v)^2 - \lambda^2\gamma p} \\ \Delta u &= \frac{\Delta q_2 - u\Delta q_1}{q_1} \\ \Delta v &= \frac{\Delta q_3 - v\Delta q_1 + \lambda\Delta p}{q_1} \qquad (3.18) \\ \Delta \rho &= \frac{\Delta q_1 + \lambda\rho\,\Delta v}{1 - \lambda v} \end{aligned}$$

The fact that the independent variable is now Q instead of U requires two changes to the basic Lax-Wendroff algorithm. The flow variables that are stored are still the standard conservation variables U. These are used as before to calculate the fluxes F and G, and the cell residual on both quadrilateral and triangular cells. However, these cell residuals, which before defined the change ΔU, now give the change ΔQ. The linearized equations Eqs. (3.18)-(3.19) are then used to evaluate ΔF and ΔG in the cell. The distribution equations now give changes in Q at the nodes. For example, the equations for quadrilateral cells are

$$\begin{aligned} \delta Q_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{24} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{24}\right) \\ \delta Q_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{31} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{31}\right) \\ \delta Q_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A - \tfrac{1}{4}\Delta F_A\,\Delta y''_{24} + \tfrac{1}{4}\Delta G_A\,\Delta x''_{24}\right) \qquad (3.19) \\ \delta Q_{4A} &= \left(\frac{\Delta t}{A'}\right)_4 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A - \tfrac{1}{4}\Delta F_A\,\Delta y''_{31} + \tfrac{1}{4}\Delta G_A\,\Delta x''_{31}\right) \end{aligned}$$

The smoothing terms are handled exactly as before. The final step is to take the old variables U^n, calculate Q^n, add the change δQ^n to obtain Q^{n+1} and then use Eqs. (3.14)-(3.17) to convert back to U^{n+1}. The additional work involved in these steps is approximately 15% of the cost of the basic algorithm.

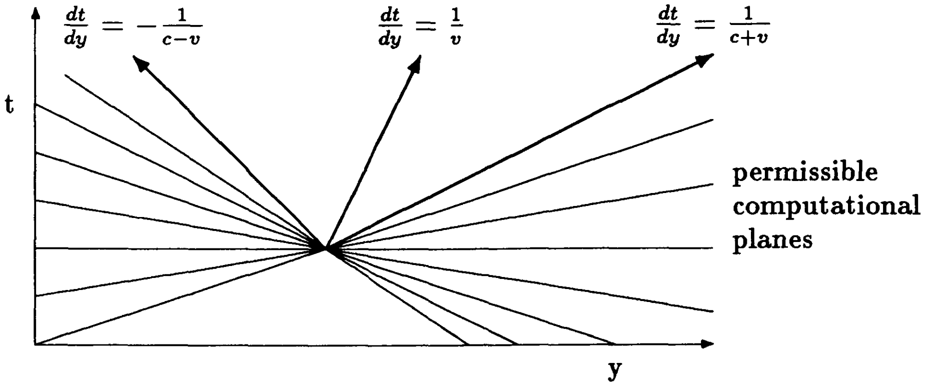
$\frac{dt}{dy} = -\frac{1}{c-v}$
$\frac{dt}{dy} = \frac{1}{v}$
$\frac{dt}{dy} = \frac{1}{c+v}$
t
permissible
computational
planes
y

Figure 3.5: Physical characteristics and permissible values of λ

3.4 Multiple blade passages

The need for multiple blade passages in some calculations arises from a fundamental limit on the magnitude of λ. Re-examining Eqs. (3.8)-(3.14), and defining $r_v = \lambda v$ and $r_c = \lambda c$, it can be shown that,

$$\begin{aligned} B &= \rho\left((1-r_v)^2 + \frac{1}{\gamma}r_c^2\right) && (3.20) \\ C &= \rho p\left(2(1-r_v)^2 - \frac{\gamma-1}{\gamma}r_c^2\right), && (3.21) \end{aligned}$$

and hence that Eq. (3.14) reduces to

$$p = \frac{\left(2(1-r_v)^2 - \frac{\gamma-1}{\gamma}r_c^2\right)\, p}{(1-r_v)^2 + \frac{1}{\gamma}r_c^2 \pm \sqrt{((1-r_v)^2 - r_c^2)^2}} \qquad (3.22)$$

When $\lambda = 0$ the positive root reduces to p, while the negative root is infinite, and so, as stated earlier, the positive root is chosen for all values of λ. This remains correct at non-zero values of λ provided $(1-r_v)^2 > r_c^2$. Assuming that the flow is subsonic in the y-direction this condition can be re-expressed as,

$$-\frac{1}{c-v} < \lambda < \frac{1}{c+v} \qquad (3.23)$$

As shown in Fig. 3.5 this condition means that the slope of the computational plane may be increased or decreased up to the point at which it is coincident with one of the three physical characteristics of the Euler equations. This is clearly a fundamental physical limitation because beyond this point a signal which propagates forward in time in the physical coordinates would be propagating backward in time in the computational coordinates, which is clearly inconsistent with the numerical procedure which marches forward in time.

Substituting the definition of λ gives the corresponding limits on the stator/rotor pitch ratio.

$$1-\frac{M_r}{1-M_y}<\frac{P_s}{P_r}<1+\frac{M_r}{1+M_y} \tag{3.24}$$

$M_y=v/c$ is the Mach number in the y-direction, and $M_r=V/c$ is the Mach number associated with the rotor speed V. The range of possible pitch ratios clearly depends most strongly on M_r. In most practical turbomachinery applications M_r lies in the range 0.3-0.6, allowing pitch ratios in the range 0.6-1.5. Unfortunately many detailed experiments are performed for good experimental reasons on large scale, low speed rigs for which M_r is substantially lower (0.05-0.2) producing a much smaller range of possible pitch ratios. In either case there are plenty of examples of situations in which the geometry to be analyzed lies outside the range of pitch ratios which can be analyzed by the current method as described so far.

The solution to this problem is to perform calculations on multiple blade passages. If, for example, the stator/rotor pitch ratio is exactly 2.0, then this case could be calculated on a grid covering two rotor passages, without requiring any time inclination of the computational plane, i.e. with $\lambda=0$. At the other extreme, if the ratio is exactly 0.5 then this case could be calculated on a single rotor passage, but with two wakes specified at the inlet plane.

In the most general case the calculation is performed on m rotor passages, with n wakes (or potential disturbances) specified at the inlet (or outlet) plane. The ratio m/n is chosen to be approximately equal to the pitch ratio. If it is exactly equal then no time inclination is required. If it is not exactly equal then λ "makes up the difference" in the same way as before.

$$\begin{aligned}\lambda=\frac{\Delta T}{m\,P_r} &= \frac{1}{m\,P_r}\,\frac{n\,P_s-m\,P_r}{V}\\ &= \frac{1}{V}\left(\frac{P_s}{P_r}/\frac{m}{n}-1\right)\end{aligned} \tag{3.25}$$

In operation, the user of UNSFLO specifies m, which controls the size of the computational grid and the corresponding computational cost, and the program calcuates the value of n which minimizes the magnitude of λ. Clearly the larger the value of m, the closer the fraction m/n will be to the pitch ratio P_s/P_r, and so the smaller λ will be. Thus for any pitch ratio and any values of M_r and M_y it is possible to find a value for m such that λ will not violate the domain of dependence restrictions discussed earlier.

Chapter 4

Stator/Rotor Interface Region

This section describes the computational algorithm used for calculations in which there are two blade rows moving relative to each other. The algorithm for the case in which the blade rows have equal pitches is presented first, because it is relatively easy to visualize and it contains all of the essential new algorithm components. Then, the algorithm for the general case of unequal pitches is presented. This uses the time-inclined computational planes described in the last section, and viewed from a purely mathematical viewpoint it is a straightforward extension of the equal pitch method. However, it becomes extremely difficult to visualize the shearing, inclined computational cells which are involved. [1]

4.1 Algorithm for equal pitches

The basic geometric approach is shown in Fig. 4.1. The computational grid is composed of two parts, one part fixed to the stator blade row (which in this discussion will be assumed to the upstream blade row) and the other part fixed to, and moving with, the rotor blade row. The two parts are separated by a cell width at the interface, with equal grid node spacing along the interface on either side. In this section we will assume that there are the same number of grid nodes on both sides of the interface, so that the gap between the two halves can be spanned by a set of quadrilateral cells defined by connecting each stator grid node to the nearest rotor grid node. In a later section we will present an alternative treatment with triangular cells which allows unequal number of nodes on either side of the interface.

[1] In fact, in my experience trying to visualize and understand it can quickly cause a severe headache which can only be relieved by taking a long walk!

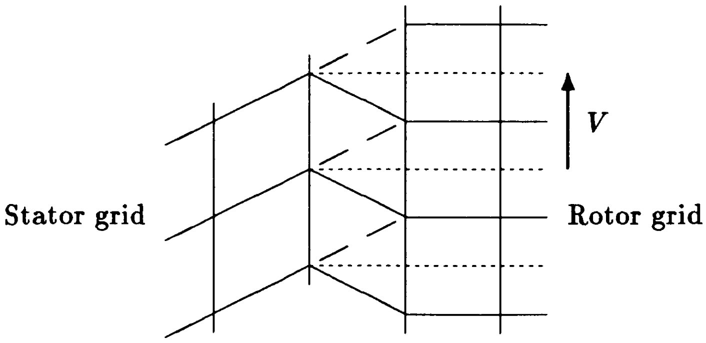

Figure 4.1: Shearing cells at unsteady stator/rotor interface

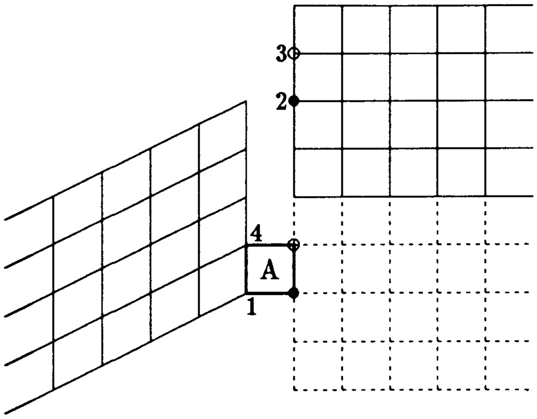

Figure 4.2: Periodic extension of rotor grid

As time progresses, the rotor moves and the cells change from State 1 (with solid lines) to State 2 (with dotted lines) to State 3 (with dashed lines). At that time the connecting lines are redefined to maintain nearest neighbor connections, and the cells revert to State 1.

As shown in Fig. 4.2, spatial periodicity is used to extend the rotor grid as needed as the rotor grid moves. The solid lines denote the actual position of the rotor grid and the dotted lines show the position of the rotor grid shifted by one pitch. The open and closed circles denote matching pairs of rotor nodes, so that when the computation is performed on cell A, the distributions really go to nodes 1, 2, 3 and 4.

On each half of the grid the flow solution is calculated using local grid-relative flow variables. This allows one to use the Lax-Wendroff algorithm described in Chapter 2 without modification. At the interface cells the basic algorithm has to be modified for two reasons.

Firstly, all flow variables have to be converted into some chosen frame of reference and when the flow change are calculated they must be converted back into the local frame of reference. In the analysis presented, and in the implementation in UNSFLO, the chosen frame of reference is the absolute stator frame. It can be verified (and has been both on paper and by programming) that using another frame of reference will produce the same final results.

The rotor-relative and stator-relative flow variables are related by

$$\begin{aligned} \rho_s &= \rho_r \\ u_s &= u_r \\ v_s &= v_r + V \\ p_s &= p_r, \end{aligned} \tag{4.1}$$

where V is the rotor wheel speed, and the subscript s denotes stator-relative values and the subscript r denotes rotor-relative values. Hence

$$U_s = \begin{pmatrix} U_{r1} \\ U_{r2} \\ U_{r3} + VU_{r1} \\ U_{r4} + VU_{r3} + \frac{1}{2}V^2U_{r1} \end{pmatrix} \tag{4.2}$$

$$U_r = \begin{pmatrix} U_{s1} \\ U_{s2} \\ U_{s3} - VU_{s1} \\ U_{s4} - VU_{s3} + \frac{1}{2}V^2U_{s1} \end{pmatrix}, \tag{4.3}$$

and

$$\delta U_r = \begin{pmatrix} \delta U_{s1} \\ \delta U_{s2} \\ \delta U_{s3} - V\delta U_{s1} \\ \delta U_{s4} - V\delta U_{s3} + \frac{1}{2}V^2\delta U_{s1} \end{pmatrix}. \tag{4.4}$$

Eq. (4.2) is needed at the beginning of the cell calculation to convert the rotor-relative values on the rotor side of the interface into stator-relative values. Eq. (4.4) is needed at the end of the cell calculation because the flow variable changes distributed to the nodes on the rotor side of the interface are changes in stator-relative quantities that have to be converted into changes in rotor-relative quantities.

The second modification to the basic Lax-Wendroff algorithm is due to the movement of the computational cell. The change is best understood by considering the

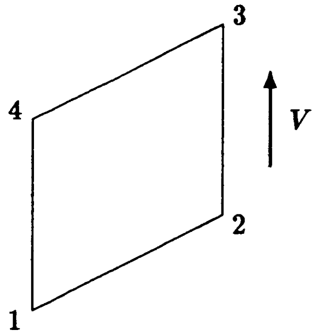

Figure 4.3: Shearing interface cell

following integral form of the two-dimensional Euler equations on a control volume whose boundary has a unit normal vector $\vec{n}$ and is moving with velocity $\vec{V} = (V_x, V_y)^T$.

$$\begin{aligned} \frac{d}{dt}\iint U\,dx\,dy &= \iint \frac{\partial U}{\partial t}\,dx\,dy + \oint U(\vec{V}.\vec{n})ds \\ &= -\oint (F\,dy - G\,dx) + \oint (UV_x\,dy - UV_y\,dx) \end{aligned} \tag{4.5}$$

Considering the computational cell shown in Fig. 4.3, the extra flux term across face 1-2 is approximated by treating U as being linear.

$$\begin{aligned} \int_{x_1}^{x_2} UV_y\,dx &= \Delta x_{21}\int_0^1 (U_1 + \xi(U_2 - U_1))\,\xi V\,d\xi \\ &= \Delta x_{21}\left(\tfrac{1}{6}U_1 + \tfrac{1}{3}U_2\right)V \end{aligned} \tag{4.6}$$

Including the corresponding term on face 3-4, the modified equation for the cell change ΔU_A is

$$\begin{aligned} \Delta U_A = -\frac{\Delta t}{2A_A}(\ & (F_1+F_2)\Delta y_{21} - (G_1+G_2)\Delta x_{21} + (\tfrac{1}{3}U_1 + \tfrac{2}{3}U_2)V\Delta x_{21} \\ & + (F_2+F_3)\Delta y_{32} - (G_2+G_3)\Delta x_{32} \\ & + (F_3+F_4)\Delta y_{43} - (G_3+G_4)\Delta x_{43} + (\tfrac{2}{3}U_3 + \tfrac{1}{3}U_4)V\Delta x_{43} \\ & + (F_4+F_1)\Delta y_{14} - (G_4+G_1)\Delta x_{14}\ ). \end{aligned} \tag{4.7}$$

The second order terms in the distribution formulae also change because of the motion of the control volume.

$$\begin{aligned} \delta U_{1A} &= \left(\frac{\Delta t}{A}\right)_1 \left(\tfrac{1}{4}\left(\frac{A}{\Delta t}\right)_A \Delta U_A - \tfrac{1}{4}\Delta F_A(y_4 - y_2) + \tfrac{1}{4}\Delta G_A(x_4 - x_2) - \tfrac{1}{4}\Delta U_A \tfrac{V}{4}(x_4 - x_2)\right) \\ \delta U_{2A} &= \left(\frac{\Delta t}{A}\right)_2 \left(\tfrac{1}{4}\left(\frac{A}{\Delta t}\right)_A \Delta U_A - \tfrac{1}{4}\Delta F_A(y_1 - y_3) + \tfrac{1}{4}\Delta G_A(x_1 - x_3) - \tfrac{1}{4}\Delta U_A \tfrac{3V}{4}(x_1 - x_3)\right) \end{aligned}$$

$$\begin{aligned}
\delta U_{3A} &= \left(\frac{\Delta t}{A}\right)_3 \left(\tfrac{1}{4}\left(\frac{A}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\,(y_4-y_2) - \tfrac{1}{4}\Delta G_A\,(x_4-x_2) + \tfrac{1}{4}\Delta U_A \tfrac{3V}{4}(x_4-x_2)\right) \\
\delta U_{4A} &= \left(\frac{\Delta t}{A}\right)_4 \left(\tfrac{1}{4}\left(\frac{A}{\Delta t}\right)_A \Delta U_A + \tfrac{1}{4}\Delta F_A\,(y_1-y_3) - \tfrac{1}{4}\Delta G_A\,(x_1-x_3) + \tfrac{1}{4}\Delta U_A \tfrac{V}{4}(x_1-x_3)\right).
\end{aligned} \tag{4.8}$$

As explained earlier, the distributions to nodes 2 and 3 must be converted into rotor-relative changes using Eq. (4.4).

4.2 Algorithm for unequal pitches

When the pitches of the stator and rotor are unequal, the conceptual approach remains the same, but the details become much more complicated. Time-inclined computational planes are used to calculate the flow in both the stator and rotor halves of the computational grid, but the time-step and inclination parameter λ are different in the two halves. In the stator frame of reference the blade-passing period is

$$T_s = P_r/V, \tag{4.9}$$

whereas in the rotor frame it is

$$T_r = P_s/V, \tag{4.10}$$

The calculation has the same number of time-steps per period on each half, so the time-steps on the two halves are related by

$$\frac{\Delta t_s}{\Delta t_r} = \frac{P_r}{P_s}. \tag{4.11}$$

Similarly, the lagged period boundary condition in the stator frame is

$$U(x,y,t) = U(x, y + P_s, t + \Delta T), \tag{4.12}$$

and in the rotor frame it is

$$U(x,y,t) = U(x, y + P_r, t + \Delta T), \tag{4.13}$$

with the time lag ΔT given by

$$\Delta T = \frac{P_s - P_r}{V} = T_r - T_s. \tag{4.14}$$

Consequently, the time-inclination parameters in the two frames of reference are

$$\lambda_s = \frac{\Delta T}{P_s} = \frac{P_s - P_r}{V P_s}, \tag{4.15}$$

and

$$\lambda_r = \frac{\Delta T}{P_r} = \frac{P_s - P_r}{VP_r}. \tag{4.16}$$

The differing values of Δt and λ in the two frames of reference are extremely confusing; it is hard to understand how this can be consistent at the stator/rotor interface. Fig. 4.4 attempts to explain this by showing both the stator and rotor inclined computational grids in the stator frame of reference.

The figure shows a case in which the the stator pitch P_s is greater than the rotor pitch P_r, and, for simplicity, there are only six timesteps per period, and only five cells spanning one pitch. The stator and rotor nodes are aligned so that they coincide at the beginning of a computational period. There are several important things to note in the diagram. At each time-level, the stator and rotor grids lie on the same inclined computational plane, but the rotor grid is displaced relative to the stator grid. The circles denote three points which are defined to be equal through the lagged periodic boundary condition, and so computationally correspond to the same two points (one on the stator side of the interface, the other on the rotor side). The diagram shows that the stator node spacing Δy_s is greater than the rotor node spacing Δy_r at fixed time t, but that on the inclined computational plane the rotor node spacing becomes equal to Δy_s. Thus, viewed on the inclined stator computational plane, the shearing cell in the interface region is a parallelogram, exactly the same as in the case of equal stator and rotor pitches. The diagram also shows that the stator timestep Δt_s is not equal to the rotor timestep Δt_r, even though both grids are consistently at the same computational time level. This is because the spatial shift of the moving rotor grid produces a temporal shift on the inclined computational grid. One final observation is that the velocity of the rotor changes when viewed in the stator inclined computational plane. It travels one pitch P_s in an apparent time of P_r/V, the time between the first and last time-level in a stator period, and so its apparent speed is

$$V_s = V\frac{P_s}{P_r}. \tag{4.17}$$

Similarly, in the inclined rotor frame of reference the apparent speed is

$$V_r = V\frac{P_r}{P_s}. \tag{4.18}$$

The algorithm for the interface region follows the same approach as for equal pitches. The first step is to form a set of shearing parallelograms in the stator inclined computational plane, by connecting stator nodes to rotor nodes, using spatial periodicity as

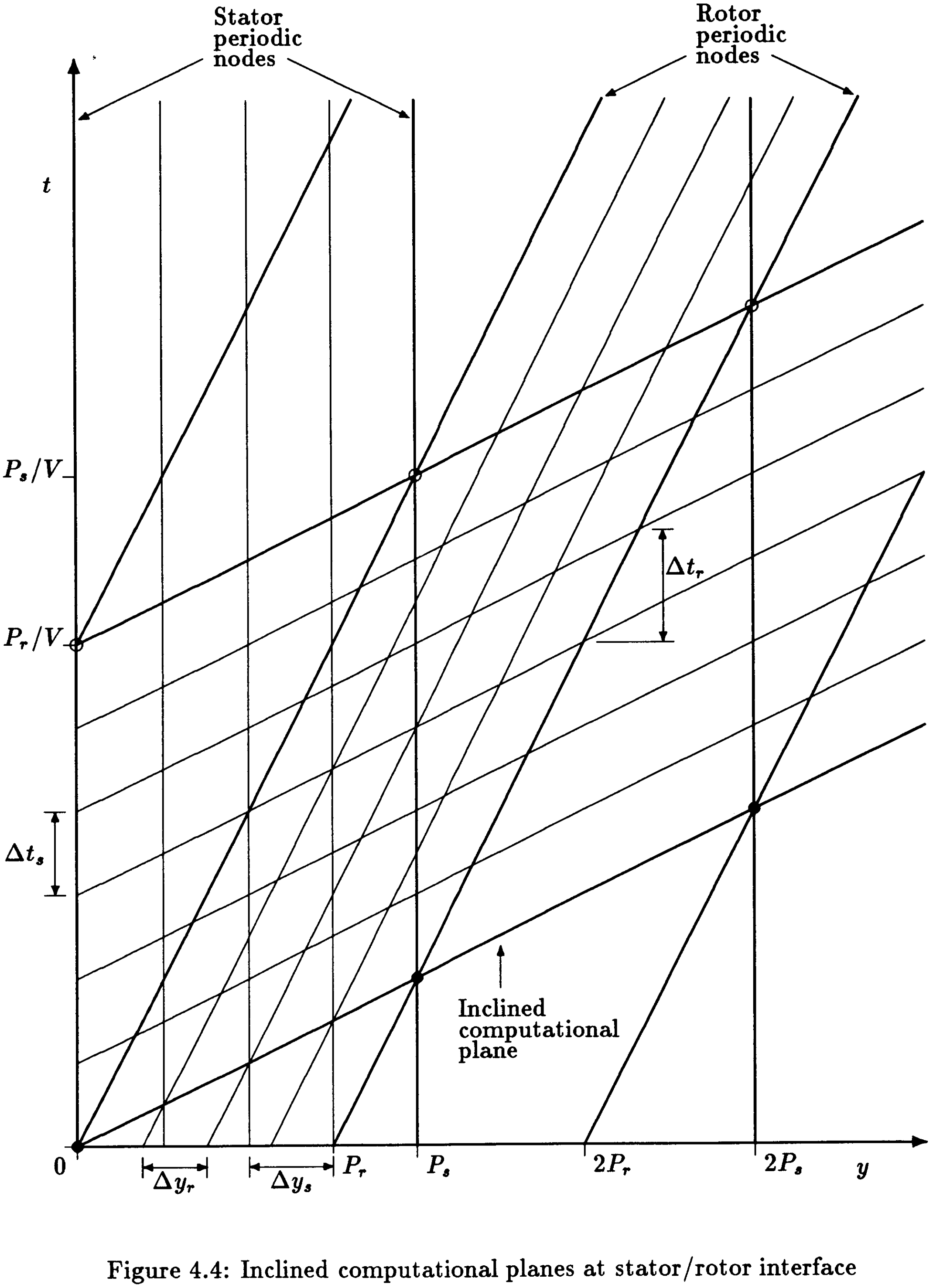

Figure 4.4: Inclined computational planes at stator/rotor interface

needed. The second step is to convert the rotor-relative flow variables at the rotor nodes to stator-relative flow variables using Eq. (4.2).

The modified equation for ΔQ_A at the center of the shearing cell is similar to Eq. (4.7). However the apparent rotor speed V_s and the modified conservation variable Q must be used instead of V and U respectively. Also introducing the quasi-three-dimensional terms gives the following equation.

$$\Delta Q_A = -\frac{\Delta t}{2A'_A}\begin{pmatrix} \hat{V}_1\rho_1 \ + \ \hat{V}_2\rho_2 \ + \ \hat{V}_3\rho_3 \ + \ \hat{V}_4\rho_4 \\ \hat{V}_1(\rho u)_1 + \hat{V}_2(\rho u)_2 + \hat{V}_3(\rho u)_3 + \hat{V}_4(\rho u)_4 - (p_3 - p_1)\Delta y''_{24} - (p_4 - p_2)\Delta y''_{31} \\ \hat{V}_1(\rho v)_1 + \hat{V}_2(\rho v)_2 + \hat{V}_3(\rho v)_3 + \hat{V}_4(\rho v)_4 + (p_3 - p_1)\Delta x''_{24} + (p_4 - p_2)\Delta x''_{31} \\ \hat{V}_1(\rho H)_1 + \hat{V}_2(\rho H)_2 + \hat{V}_3(\rho H)_3 + \hat{V}_4(\rho H)_4 \end{pmatrix}$$

$$-\frac{\Delta t}{2A'_A}\left((\tfrac{1}{3}Q_1 + \tfrac{2}{3}Q_2)V_s\Delta x'_{21} + (\tfrac{2}{3}Q_3 + \tfrac{1}{3}Q_4)V_s\Delta x'_{43}\right) \tag{4.19}$$

The geometric and $\hat{V}$ variables are as defined in Chapter 2.

ΔU,ΔF and ΔG are calculated from ΔQ in the usual manner, and then the distribution formulae are

$$\begin{aligned} \delta Q_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{24} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{24} + \tfrac{1}{4}\Delta Q_A \tfrac{V_s}{4}\Delta x''_{24}\right) \\ \delta Q_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{31} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{31} + \tfrac{1}{4}\Delta Q_A \tfrac{3V_s}{4}\Delta x''_{31}\right) \\ \delta Q_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A - \tfrac{1}{4}\Delta F_A\,\Delta y''_{24} + \tfrac{1}{4}\Delta G_A\,\Delta x''_{24} - \tfrac{1}{4}\Delta Q_A \tfrac{3V_s}{4}\Delta x''_{24}\right) \\ \delta Q_{4A} &= \left(\frac{\Delta t}{A'}\right)_4 \left(\tfrac{1}{4}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A - \tfrac{1}{4}\Delta F_A\,\Delta y''_{31} + \tfrac{1}{4}\Delta G_A\,\Delta x''_{31} - \tfrac{1}{4}\Delta Q_A \tfrac{V_s}{4}\Delta x''_{31}\right). \end{aligned} \tag{4.20}$$

The smoothing terms are calculated and distributed as normal. The final step is the conversion of the distributed changes to nodes 2 and 3 from stator-relative changes to rotor-relative changes. There are two components to this. One is due to the different Δt and Δy in the two frames of reference (as discussed earlier).

$$\left(\frac{A}{\Delta t}\right)_r = \left(\frac{P_r}{P_s}\right)^2 \left(\frac{A}{\Delta t}\right)_s. \tag{4.21}$$

The other is similar to the conversion from δU_s to δU_r, except that the conversion is now from δQ_s to δQ_r, making the algebra considerably more complicated although the final result is almost identical.

$$
\begin{aligned}
Q_r &= \begin{pmatrix} \rho - \lambda\rho v \\ \rho u - \lambda\rho u v \\ \rho v - \lambda(\rho v^2 + p) \\ \rho E - \lambda(\rho E + p)v \end{pmatrix}_{rotor} \\
&= \begin{pmatrix} \rho - \dfrac{P_s - P_r}{VP_r}\rho v \\ \rho u - \dfrac{P_s - P_r}{VP_r}\rho u v \\ \rho v - \dfrac{P_s - P_r}{VP_r}(\rho v^2 + p) \\ \rho E - \dfrac{P_s - P_r}{VP_r}(\rho E + p)v \end{pmatrix}_{rotor} \\
&= \frac{P_s}{P_r}\begin{pmatrix} \dfrac{P_r}{P_s}\rho - \dfrac{P_s - P_r}{VP_s}\rho(v-V) \\ \dfrac{P_r}{P_s}\rho u - \dfrac{P_s - P_r}{VP_s}\rho u(v-V) \\ \dfrac{P_r}{P_s}\rho(v-V) - \dfrac{P_s - P_r}{VP_s}(\rho(v-V)^2 + p) \\ \dfrac{P_r}{P_s}\rho(E - vV + \frac{1}{2}V^2) - \dfrac{P_s - P_r}{VP_s}(\rho(E - vV + \frac{1}{2}V^2) + p)(v-V)) \end{pmatrix}_{stator} \\
&= \frac{P_s}{P_r}\begin{pmatrix} \rho - \dfrac{P_s - P_r}{VP_s}\rho v \\ \rho u - \dfrac{P_s - P_r}{VP_s}\rho u v \\ \rho(v-V) - \dfrac{P_s - P_r}{VP_s}(\rho v(v-V) + p) \\ \rho(E - vV + \frac{1}{2}V^2) - \dfrac{P_s - P_r}{VP_s}(\rho(E - vV + \frac{1}{2}V^2) + p)v - pV) \end{pmatrix}_{stator} \\
&= \frac{P_s}{P_r}\begin{pmatrix} \rho - \lambda\rho v \\ \rho u - \lambda\rho u v \\ \rho(v-V) - \lambda(\rho v(v-V) + p) \\ \rho(E - vV + \frac{1}{2}V^2) - \lambda(\rho(E - vV + \frac{1}{2}V^2) + p)v - pV) \end{pmatrix}_{stator} \\
&= \frac{P_s}{P_r}\begin{pmatrix} Q_1 \\ Q_2 \\ Q_3 - VQ_1 \\ Q_4 - VQ_3 + \frac{1}{2}V^2 Q_1 \end{pmatrix}_{stator} \qquad (4.22)
\end{aligned}
$$

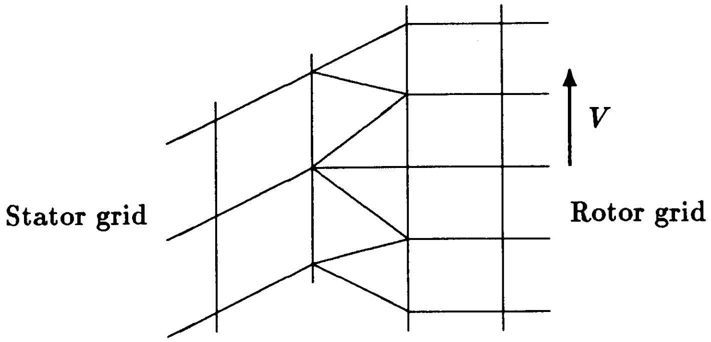

Figure 4.5: Triangular cells at unsteady stator/rotor interface

Thus, the equation to convert the distributed changes into rotor-relative changes is

$$\left(\frac{A}{\Delta t}\right)_r \delta Q_r = \frac{P_r}{P_s}\left(\frac{A}{\Delta t}\right)_s \begin{pmatrix} \delta Q_1 \\ \delta Q_2 \\ \delta Q_3 - V\delta Q_1 \\ \delta Q_4 - V\delta Q_3 + \frac{1}{2}V^2\delta Q_1 \end{pmatrix}_s . \qquad (4.23)$$

4.3 Unequal numbers of interface nodes

In the first section in this chapter, an assumption was made that the number of grid nodes on either side of the stator/rotor interface is equal. This is generally desirable, but sometimes it is useful to be able to perform calculations with differing numbers of nodes on the two sides. In this case, it is no longer possible to span the interface gap with quadrilaterals. Instead, triangular cells are created in the time-inclined stator frame by joining each cell face on either side to the node on the opposing side which is closest to the face's midpoint. Fig. 4.5 shows a typical interface region created by this technique.

The discrete equations for the shearing triangular cells in the interface region take differing forms depending on whether the majority of the nodes are on the stator or rotor side. If nodes 1 and 3 are on the stator side of the interface, then the cell change ΔQ_A is given by

$$\Delta Q_A = -\frac{\Delta t}{2A'_A} \begin{pmatrix} \hat{V}_1\rho_1 + \hat{V}_2\rho_2 + \hat{V}_3\rho_3 \\ \hat{V}_1(\rho u)_1 + \hat{V}_2(\rho u)_2 + \hat{V}_3(\rho u)_3 + p_1\Delta y''_{23} + p_2\Delta y''_{31} + p_3\Delta y''_{12} \\ \hat{V}_1(\rho v)_1 + \hat{V}_2(\rho v)_2 + \hat{V}_3(\rho v)_3 - p_1\Delta x''_{23} - p_2\Delta x''_{31} - p_3\Delta x''_{12} \\ \hat{V}_1(\rho H)_1 + \hat{V}_2(\rho H)_2 + \hat{V}_3(\rho H)_3 \end{pmatrix}$$

$$-\frac{\Delta t}{2A'_A}\left((\tfrac{1}{3}Q_1 - \tfrac{1}{3}Q_3)V_s\Delta x'_{21}\right) \tag{4.24}$$

The geometric and $\hat{V}$ variables are again as defined in Chapter 2. ΔU,ΔF and ΔG are calculated from ΔQ in the usual manner, and then the distribution formulae are

$$\begin{aligned}
\delta Q_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{2}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{23} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{23} + \tfrac{1}{4}\Delta Q_A \frac{V_s}{4}\Delta x''_{23}\right)\\
\delta Q_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{4}\Delta F_A\,\Delta y''_{31} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{31} \right)\\
\delta Q_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{2}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{12} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{12} + \tfrac{1}{4}\Delta Q_A \frac{V_s}{4}\Delta x''_{12}\right)
\end{aligned} \tag{4.25}$$

Note the fact that the first order distribution term going to node 2 is zero, and so nodes 1 and 3 equally share the first order changes. This was done because the only way for the scheme to remain conservative is to consider half of the triangle's area to 'belong' to node 1 and the other half to node 3. This way the nodal areas of 1 and 3 remain constant because each 'owns' half of the two triangles on either side. Given this apportioning of the cell area, the first order change has to be distributed consistently.

For a triangular cell with nodes 2 and 3 on the rotor side, the corresponding equations are,

$$\begin{aligned}
\Delta Q_A = -\frac{\Delta t}{2A'_A} &\begin{pmatrix} \hat{V}_1\rho_1 \ + \ \hat{V}_2\rho_2 \ + \ \hat{V}_3\rho_3 \\ \hat{V}_1(\rho u)_1 + \hat{V}_2(\rho u)_2 + \hat{V}_3(\rho u)_3 + p_1\Delta y''_{23} + p_2\Delta y''_{31} + p_3\Delta y''_{12} \\ \hat{V}_1(\rho v)_1 + \hat{V}_2(\rho v)_2 + \hat{V}_3(\rho v)_3 - p_1\Delta x''_{23} - p_2\Delta x''_{31} - p_3\Delta x''_{12} \\ \hat{V}_1(\rho H)_1 + \hat{V}_2(\rho H)_2 + \hat{V}_3(\rho H)_3 \end{pmatrix}\\
&-\frac{\Delta t}{2A'_A}\left((\tfrac{2}{3}Q_2 - \tfrac{2}{3}Q_3)V_s\Delta x'_{21}\right),
\end{aligned} \tag{4.26}$$

and

$$\begin{aligned}
\delta Q_{1A} &= \left(\frac{\Delta t}{A'}\right)_1 \left(\tfrac{1}{4}\Delta F_A\,\Delta y''_{23} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{23} \right)\\
\delta Q_{2A} &= \left(\frac{\Delta t}{A'}\right)_2 \left(\tfrac{1}{2}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{31} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{31} + \tfrac{1}{4}\Delta Q_A \tfrac{3V_s}{4}\Delta x''_{31}\right)\\
\delta Q_{3A} &= \left(\frac{\Delta t}{A'}\right)_3 \left(\tfrac{1}{2}\left(\frac{A'}{\Delta t}\right)_A \Delta Q_A + \tfrac{1}{4}\Delta F_A\,\Delta y''_{12} - \tfrac{1}{4}\Delta G_A\,\Delta x''_{12} + \tfrac{1}{4}\Delta Q_A \tfrac{3V_s}{4}\Delta x''_{12}\right)
\end{aligned} \tag{4.27}$$

The smoothing terms are calculated and distributed as usual. Finally, the conversion of the distributions from stator-relative changes to rotor-relative changes is exactly the same as for the quadrilateral algorithm.

4.4 Multiple blades

As with wake/rotor calculations, there are situations in which stator/rotor calculations with just one stator and one rotor leads to time-inclination parameters which are too large. In this case it may be necessary to perform the calculation with m rotors and n stators. All of the preceding discussion in this chapter remains valid by considering an equivalent stator pitch which is equal to nP_s and an equivalent rotor pitch which is equal to mP_r. Thus, the time-tilting parameters in the stator and rotor frames are

$$\begin{aligned}\lambda_s &= \frac{1}{V}\left(\frac{nP_s}{mP_r}-1\right)\left(\frac{nP_s}{mP_r}\right)^{-1} \\ \lambda_r &= \frac{1}{V}\left(\frac{nP_s}{mP_r}-1\right),\end{aligned} \tag{4.28}$$

and the periods in the two frames of reference are

$$\begin{aligned}T_s &= \frac{mP_r}{|V|} \\ T_r &= \frac{nP_s}{|V|}.\end{aligned} \tag{4.29}$$

Chapter 5

Steady Boundary Conditions

5.1 Overall approach

This section describes the numerical inflow and outflow boundary conditions which are used for calculations of steady flow. They are an implementation of a theoretical development in non-reflecting boundary conditions which is presented in Refs. [13, 14]. The objective of non-reflecting boundary conditions for steady flows is to be able to bring the far-field boundary location relatively close to the blades without affecting the flow field in the neighborhood of the blades. The smaller computational domain leads to much more efficient calculations. It is also very important when doing steady stator/rotor calculations since the spacing between the blade rows can be quite small.

The approach is based upon a characteristic analysis of the linearized Euler equations. At each inflow or outflow boundary there is a certain number of incoming modes and a certain number of outgoing ones. The changes in the outgoing characteristic values are taken from the changes distributed by the Lax-Wendroff algorithm. The average changes in the incoming characteristics are determined to satisfy a number of user-specified average quantities. At the inflow these are flow angle, stagnation denity and stagnation enthalpy. At the outflow it is the average static pressure. The remaining changes, the spatial harmonics of the incoming characteristics are specified by the non-reflecting boundary condition theory based upon the the amplitudes of the corresponding spatial harmonics of the outgoing characteristics.

5.2 Average flow definitions

The computational grid is constructed so that the grid nodes are equally spaced along the inflow and outflow boundaries. The average values of any flow quantity ϕ can be defined by

$$\bar{\phi} = \frac{1}{N}\sum_j \phi_j \tag{5.1}$$

where the sum is over the N nodes at the boundary, including the periodic grid node only once.

If the Euler equations were linear then this definition would be all that was required. However, because of non-linearities, the average value of a variable will not in general be equal to its value based upon averages of other variables. For example,

$$\bar{p} \neq (\gamma-1)\bar{\rho}\left(\bar{E} - \tfrac{1}{2}(\bar{u}^2+\bar{v}^2)\right) \tag{5.2}$$

Hence, this raises the question of what is the correct way in which to perform the averaging procedure. The only rigorous definition is based upon the 'mixed-out' flow-field. This approach starts from the two-dimensional Euler equations, integrated in the y-direction over one pitch.

$$\begin{aligned} \frac{d}{dx}\int_0^P F\,dy &= \int_0^P \frac{\partial F}{\partial x}\,dy \\ &= \int_0^P -\frac{\partial G}{\partial y}\,dy \\ &= G(0) - G(P) \\ &= 0 \end{aligned} \tag{5.3}$$

Thus, if one assumes that sufficiently far upstream or downstream the flow is uniform, then the flux F based upon this uniform value U_F must be equal to the average flux $\bar{F}$ at the boundary under consideration. This gives the following set of equations for U_F.

$$\begin{aligned} \rho_F u_F &= \bar{F}_1 \\ \rho_F u_F^2 + p_F &= \bar{F}_2 \\ \rho_F u_F v_F &= \bar{F}_3 \\ \rho_F u_F H_F &= \bar{F}_4 \end{aligned} \tag{5.4}$$

Together with the equation

$$H_F = \frac{\gamma}{\gamma-1}\frac{p_F}{\rho_F} + \tfrac{1}{2}\left(u_F^2 + v_F^2\right) \tag{5.5}$$

these can be solved to obtain

$$\begin{aligned}
p_F &= \frac{1}{\gamma+1}\left(\bar{F}_2 + \sqrt{\bar{F}_2^2 + (\gamma^2-1)(\bar{F}_2^2 + \bar{F}_3^2 - 2\bar{F}_1\bar{F}_4)}\right) \\
u_F &= \frac{\bar{F}_2 - p_F}{\bar{F}_1} \\
v_F &= \frac{\bar{F}_3}{\bar{F}_1} \\
\rho_F &= \frac{\bar{F}_1}{u_F}
\end{aligned} \tag{5.6}$$

Based on these values, 'mixed-out' values of all other flow variables can be defined, and will be denoted by the subscript $_F$. An important point to note is that the physical mixing process implied in this procedure will generate viscous losses, and will result in a flow with a higher entropy level. Hence, when applied to flow at the outflow boundary this averaging procedure will tend to produce higher 'measured' losses than other averaging methods, such as averaging the outgoing entropy.

The same flux-averaging method can also be used to determine losses in an unsteady flow. In this case the flux components must be averaged in time as well as in space, over a time interval which for periodic flows is the period, and for non-periodic flows is large compared to any other time-scales in the flow.

5.3 Characteristic variables

When calculating the change in the boundary values from time level n to time level n+1, the characteristic variables are defined in terms of perturbations to the average inflow or outflow flow field at time level n. As shown in Reference [13], the one-dimensional characteristic variables are related to the perturbations in the primitive variables by the following two equations.

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 \\ 0 & \rho c & 0 & 1 \\ 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} \rho - \bar{\rho} \\ u - \bar{u} \\ v - \bar{v} \\ p - \bar{p} \end{pmatrix} \tag{5.7}$$

$$\begin{pmatrix} \rho - \bar{\rho} \\ u - \bar{u} \\ v - \bar{v} \\ p - \bar{p} \end{pmatrix} = \begin{pmatrix} -\frac{1}{c^2} & 0 & \frac{1}{2c^2} & \frac{1}{2c^2} \\ 0 & 0 & \frac{1}{2\rho c} & -\frac{1}{2\rho c} \\ 0 & \frac{1}{\rho c} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} \tag{5.8}$$

The first characteristic variable is the linearized perturbation in entropy, the second variable is the tangential velocity at the boundary and is associated with the vorticity, and the remaining two variables are downstream and upstream running pressure waves, assuming that the axial Mach number is subsonic. At the inflow boundary the first three characteristics are incoming and so must be specified. The fourth is outgoing and so must be extrapolated or obtained in some other manner from the interior flow field. At the outflow boundary the roles are reversed and it is the fourth characteristic variable which must be set.

The boundary conditions are implemented at the point in the overall algorithm at which the Lax-Wendroff algorithm has distributed changes δU to all of the nodes, including nodes on the boundaries, but the nodal values have not yet been updated. The Lax-Wendroff changes at the boundary nodes can be used to define changes in the characteristic variables at each boundary node.

$$\begin{pmatrix} \delta c_1 \\ \delta c_2 \\ \delta c_3 \\ \delta c_4 \end{pmatrix}_{LW} = \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 \\ 0 & \rho c & 0 & 1 \\ 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} \delta \rho \\ \delta u \\ \delta v \\ \delta p \end{pmatrix}_{LW} \tag{5.9}$$

These Lax-Wendroff changes in the characteristic variables are used for the outgoing characteristic variables, since the Law-Wendroff algorithm should correctly calculate and distribute the changes due to the outgoing characteristic waves. The Lax-Wendroff changes in the incoming characteristic variables are discarded, since these are the changes which are to be specified by the non-reflecting theory and the user-specified average flow quantities. Once these have all been specified in the manner shown in the next sections, the changes in the characteristic variables at the boundary nodes can be converted back into changes in the primitive variables.

$$\begin{pmatrix} \delta \rho \\ \delta u \\ \delta v \\ \delta p \end{pmatrix} = \begin{pmatrix} -\frac{1}{c^2} & 0 & \frac{1}{2c^2} & \frac{1}{2c^2} \\ 0 & 0 & \frac{1}{2\rho c} & -\frac{1}{2\rho c} \\ 0 & \frac{1}{\rho c} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} \delta c_1 \\ \delta c_2 \\ \delta c_3 \\ \delta c_4 \end{pmatrix} \tag{5.10}$$

These can then be converted into changes in the conservation variables and the entire flow field can be updated.

The remaining sections in this chapter describe how the changes in the incoming characteristic variables are determined in the various different inflow and outflow cases.

5.4 Subsonic inflow

The subsonic inflow boundary conditions are the most complicated. The reason for this is that a straightforward implementation of the non-reflecting boundary condition theory would produce a flow field which to first order would have uniform entropy and stagnation enthalpy. However, due to second order effects neglected in the linear theory, there would be small variations in the entropy and stagnation enthalpy, which might be comparable in magnitude to the small losses produced in a viscous calculation. To avoid this problem, the boundary conditions which are used are a combination of the non-reflecting theory together with the requirement that the entropy and stagnation enthalpy are perfectly uniform across the inflow.

The changes in the incoming characteristic variables at each point on the inflow boundary can be split into two components, one part which is an average change along the boundary, and a second which is due to the harmonic variations in the characteristic variables along the boundary.

The average characteristic changes are calculated from the requirement that the average entropy, flow angle and stagnation enthalpy have certain values. This is achieved by driving to zero the following three residuals.

$$\begin{aligned} R_1 &= p\bar{S} \\ R_2 &= \rho c\left(v_F - \tan(\alpha_{inl})u_F\right) \\ R_3 &= \rho\left(\bar{H} - \tfrac{1}{\gamma-1}\right) \end{aligned} \tag{5.11}$$

S is an entropy-related function defined by

$$S = \log(\gamma p) - \gamma \log \rho, \tag{5.12}$$

and because of the non-dimensionalization chosen earlier, the correct inflow values for S and the stagnation enthalpy H are 0 and $\frac{1}{\gamma-1}$ respectively. α_{inl} is the user-specified average inflow angle, and note that in defining R_2 the flux-averaged values of the velocity have been used.

The average changes in the incoming characteristic variables, which are required to drive the residuals to zero, are obtained by one step of a Newton-Raphson procedure.

$$\begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix}^n + \frac{\partial(R_1, R_2, R_3)}{\partial(c_1, c_2, c_3)} \begin{pmatrix} \delta\bar{c}_1 \\ \delta\bar{c}_2 \\ \delta\bar{c}_3 \end{pmatrix} = 0 \tag{5.13}$$

The Jacobian matrix is obtained as the product of two other matrices.

$$
\begin{aligned}
\frac{\partial(R_1,R_2,R_3)}{\partial(c_1,c_2,c_3)} &= \frac{\partial(R_1,R_2,R_3)}{\partial(\rho,u,v,p)}\;\frac{\partial(\rho,u,v,p)}{\partial(c_1,c_2,c_3)} \\
&= \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & -\rho c\tan(\alpha_{inl}) & \rho c & 0 \\ -\frac{1}{\gamma-1}c^2 & \rho u & \rho v & \frac{\gamma}{\gamma-1} \end{pmatrix} \begin{pmatrix} -\frac{1}{c^2} & 0 & \frac{1}{2c^2} \\ 0 & 0 & \frac{1}{2\rho c} \\ 0 & \frac{1}{\rho c} & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -\frac{1}{2}\tan(\alpha_{inl}) \\ \frac{1}{\gamma-1} & M_y & \frac{1}{2}(1+M_x) \end{pmatrix} \qquad (5.14)
\end{aligned}
$$

In forming the matrix $\frac{\partial(R_1,R_2,R_3)}{\partial(\rho,u,v,p)}$ several terms which are proportional to the residuals were neglected since these are zero in the converged limit. Inverting the Jacobian matrix gives the following equation for the average changes.

$$
\begin{pmatrix} \delta\bar{c}_1 \\ \delta\bar{c}_2 \\ \delta\bar{c}_3 \end{pmatrix} = \frac{-1}{1+M_x+M_y\tan(\alpha_{inl})} \begin{pmatrix} 1+M_x+M_y\tan(\alpha_{inl}) & 0 & 0 \\ -\frac{1}{\gamma-1}\tan(\alpha_{inl}) & 1+M_x & \tan(\alpha_{inl}) \\ -\frac{2}{\gamma-1} & -2M_y & 2 \end{pmatrix} \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix} \qquad (5.15)
$$

The next step is to calculate the local changes in the characteristic variables at each point on the inflow boundary due to the variation in the characteristic variables along the boundary. Firstly, the outgoing fourth characteristic variable is evaluated at each point, and its discrete Fourier transform is calculated for a range of values of k from $-N/2+1$ to $+N/2-1$.

$$
\hat{c}_{4k} = \frac{1}{N}\sum_{j=1}^{N} c_{4j}\exp\left(\tfrac{-i2\pi jk}{N}\right) \qquad (5.16)
$$

Because of the definition of the characteristic variables as perturbations from the current uniform state, the Fourier component corresponding to $k=0$ is zero.

According to the steady-state non-reflecting theory presented in Ref. [13], the correct steady-state amplitude of the Fourier transform of the second characteristic is

$$
\hat{c}_{2k\,s} = -\frac{\beta+M_y}{1+M_x}\,\hat{c}_{4k}, \qquad (5.17)
$$

where

$$
\beta = i\,\mathrm{sign}(k)\,\sqrt{1-M^2}. \qquad (5.18)
$$

Transforming back into the physical domain gives

$$
c_{2j\,s} = \sum_{k=-N/2+1}^{N/2-1} \hat{c}_{2k\,s}\exp\left(\tfrac{i2\pi jk}{N}\right) \qquad (5.19)
$$

Because terms corresponding to ±k form complex conjugate pairs, this expression can be rewritten as

$$c_{2j\,s} = 2\,\mathrm{Re}\left\{ \sum_{k=1}^{N/2-1} \hat{c}_{2k\,s} \exp\left(\frac{i2\pi jk}{N}\right) \right\}, \tag{5.20}$$

reducing the amount of computation required.

The ideal steady-state correction to the local second characteristic variable is the difference between the correct steady-state value and the current value.

$$\delta c_{2j\,s} = c_{2j\,s} - c_{2j} \tag{5.21}$$

The ideal steady-state corrections to the local first and third characteristic variables are obtained from the condition that the local entropy and stagnation enthalpy should match the average values. This is achieved by the same Newton-Raphson procedure used earlier to obtain the average changes. This time, the residuals are given by perturbations from the average entropy and stagnation enthalpy values.

$$\begin{aligned} R_{1j} &= p\,(S_j - \bar{S}) \\ R_{3j} &= \rho\,(H_j - \bar{H}) \end{aligned} \tag{5.22}$$

The Newton-Raphson equation is

$$\begin{pmatrix} R_{1j} \\ R_{3j} \end{pmatrix}^n + \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{\gamma-1} & M_y & \frac{1}{2}(1+M_x) \end{pmatrix} \begin{pmatrix} \delta c_{1j} \\ \delta c_{2j} \\ \delta c_{3j} \end{pmatrix} = 0, \tag{5.23}$$

and the solution is

$$\begin{aligned} \delta c_{1j\,s} &= -R_{1j} \\ \delta c_{3j\,s} &= -\frac{2}{1+M_x}\left(\tfrac{1}{\gamma-1}\,\delta c_{1j\,s} + M_y\,\delta c_{2j\,s} + R_{3j}\right). \end{aligned} \tag{5.24}$$

Now that the ideal local non-reflecting corrections have been calculated, these are added to the average global changes, and multiplied by an under-relaxation factor, σ.

$$\begin{aligned} \delta c_{1j} &= \sigma(\delta\bar{c}_1 + \delta c_{1j\,s}) \\ \delta c_{2j} &= \sigma(\delta\bar{c}_2 + \delta c_{2j\,s}) \\ \delta c_{3j} &= \sigma(\delta\bar{c}_3 + \delta c_{3j\,s}) \end{aligned} \tag{5.25}$$

Ref. [13] discusses the need for this under-relaxation to guarantee the wellposedness of the mathematical formulation. The value of σ which has been found to work well is

$1/N$. This corresponds to a relaxation time which is similar in magnitude to the blade pitch divided by the speed of sound.

Together with the change in the outgoing fourth characteristic given by the Lax-Wendroff algorithm,

$$\delta c_{4j} = \delta c_{4jLW}, \tag{5.26}$$

this completes the calculation of the characteristic changes, and the final step is to convert the changes into the conservation variables before updating the flow field.

5.5 Supersonic inflow

The treatment of the inflow boundary conditions when the flow is supersonic, but still axially subsonic, is almost the same as for subsonic flow. The only difference is in the definition of β.

$$\beta = -\text{sign}(v)\sqrt{M^2-1} \tag{5.27}$$

Because β is now independent of the Fourier mode k, it is no longer necessary to perform the discrete Fourier transforms. Instead, the ideal steady-state values for the incoming second characteristic variables are given by

$$c_{2j_s} = -\frac{\beta+M_y}{1+M_x}\, c_{4j}. \tag{5.28}$$

The remainder of the boundary condition implementation is exactly the same as for subsonic flow. There is a physical significance in the fact that the discrete Fourier transforms are not needed for supersonic flow. The linear steady-state non-reflecting boundary conditions for supersonic flow specify that the incoming linearized supersonic Reimann invariant is uniform along the inflow boundary. The Riemann variables are locally defined quantities and so it is natural that this leads to a local boundary condition.

An additional option for supersonic inflows, is to specify the value of the incoming nonlinear supersonic Reimann invariant as an alternative to the specification of the mean flow angle. In this case the second mean flow residual becomes

$$R_2 = \rho c^2(\alpha + sgn(\alpha)\,\nu(M) - r_{inl}), \tag{5.29}$$

where α is the current flow angle, r_{inl} is the specified value of the Riemann invariant, and $\nu(M)$ is the Prandtl-Meyer function,

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}}\ \tan^{-1}\left(\sqrt{\frac{\gamma-1}{\gamma+1}(M^2-1)}\right)\ -\ \tan^{-1}\left(\sqrt{M^2-1}\right). \tag{5.30}$$

5.6 Subsonic outflow

The subsonic outflow boundary conditions are a straightforward implementation of the non-reflecting boundary condition theory. The first three characteristics are outgoing, so only the fourth characteristic variable needs to be set.

The average change in the characteristic is determined to achieve a user-specified average exit pressure. The derivative of pressure with respect to variations in the fourth characteristic is

$$\frac{\partial p}{\partial c_4} = \tfrac{1}{2}, \tag{5.31}$$

and so the equation for the average change in the fourth characteristic variables is

$$\delta \bar{c}_4 = -2(p_{_F} - p_{exit}). \tag{5.32}$$

p_{exit} is the user-specified exit pressure, and $p_{_F}$ is the flux-averaged pressure of the current flow field.

The next step is to calculate the local changes. Firstly, the outgoing second and third characteristic variables are evaluated, and the discrete Fourier transforms are calculated.

$$\begin{aligned} \hat{c}_{2k} &= \frac{1}{N}\sum_{j=1}^{N} c_{2j} \exp\left(\tfrac{-i2\pi jk}{N}\right) \\ \hat{c}_{3k} &= \frac{1}{N}\sum_{j=1}^{N} c_{3j} \exp\left(\tfrac{-i2\pi jk}{N}\right) \end{aligned} \tag{5.33}$$

From Ref. [13], the correct steady-state amplitude of the Fourier transform of the incoming fourth characteristic variable is

$$\hat{c}_{4k\,s} = \frac{2M_x}{\beta - M_y}\hat{c}_{2k} - \frac{\beta + M_y}{\beta - M_y}\hat{c}_{3k}. \tag{5.34}$$

Again using the simplification due to the complex conjugate pairs, the ideal non-reflecting steady-state values for the incoming fourth characteristic variables are

$$c_{4j\,s} = 2\,\mathrm{Re}\left\{ \sum_{k=1}^{N/2-1} \hat{c}_{4k\,s} \exp\left(\frac{i2\pi jk}{N}\right) \right\}. \tag{5.35}$$

The ideal local change is then

$$\delta c_{4j\,s} = c_{4j\,s} - c_{4j}, \tag{5.36}$$

and when added to the average change, and under-relaxed as before to ensure well-posedness, this gives the final change in the incoming characteristic.

$$\delta c_{4j} = \sigma(\delta \bar{c}_4 + \delta c_{4j_s}) \tag{5.37}$$

The changes in the outgoing characteristics are again taken from the Lax-Wendroff algorithm.

$$\begin{aligned} \delta c_{1j} &= \delta c_{1jLW} \\ \delta c_{2j} &= \delta c_{2jLW} \\ \delta c_{3j} &= \delta c_{3jLW} \end{aligned} \tag{5.38}$$

5.7 Supersonic outflow

As with the inflow boundary conditions, the supersonic outflow boundary condition is identical to the subsonic outflow boundary condition, except in the definition of β which is

$$\beta = -\mathrm{sign}(v)\sqrt{M^2-1} \tag{5.39}$$

Because β is again independent of the Fourier mode k, it is again possible to simplify the computation by not performing the discrete Fourier transforms. Instead, the ideal steady-state values for the incoming fourth characteristic variables are given by

$$c_{4j_s} = \frac{2M_x}{\beta - M_y} c_{2j} - \frac{\beta + M_y}{\beta - M_y} c_{3j}. \tag{5.40}$$

The remainder of the boundary condition implementation is exactly the same as for subsonic flow.

5.8 Steady stator/rotor interaction

One interesting application for a steady-state flow analysis is a coupled calculation of a stator/rotor configuration. In this problem one simultaneously calculates the steady flow in both a stator and a rotor stage, with boundary conditions implemented to couple the two calculations together. Assuming that the stator is upstream of the rotor, then the stator inflow and rotor outflow boundary conditions are treated as usual. Each blade row is calculated using local relative flow variables, and the user can specify the stator inflow angle and the rotor exit pressure.

The interesting boundaries are the stator outflow and the rotor inflow, since these are the two that must be matched together. To a large extent these are treated by the same methods presented above. The only difference is in the calculation of the average changes of the three incoming characteristics at the rotor inflow and the one incoming characteristic at the stator outflow. To conserve mass, momentum and energy, the objective is to make the flux of these out of the stator equal to the flux into the rotor. If flux-averaging is used then an equivalent objective is to match the average flow quantities.

$$\begin{aligned} \rho_{F\,stator} &= \rho_{F\,rotor} \\ u_{F\,stator} &= u_{F\,rotor} \\ v_{F\,stator} &= v_{F\,rotor} + V \\ p_{F\,stator} &= p_{F\,rotor} \end{aligned} \tag{5.41}$$

Note that because of the use of relative flow variables, the rotor wheel speed V has to be introduced into the condition of matching circumferential velocities.

If the current computed solution does not satisfy these matching conditions then it can be interpreted as a jump in characteristic values.

$$\begin{pmatrix} \Delta c_1 \\ \Delta c_2 \\ \Delta c_3 \\ \Delta c_4 \end{pmatrix} = \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 \\ 0 & \rho c & 0 & 1 \\ 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} \rho_{F\,stator} - \rho_{F\,rotor} \\ u_{F\,stator} - u_{F\,rotor} \\ v_{F\,stator} - v_{F\,rotor} - V \\ p_{F\,stator} - p_{F\,rotor} \end{pmatrix} \tag{5.42}$$

The average characteristic changes at the stator outflow and rotor inflow are now set to eliminate each of these characteristic jumps, taking note of the direction of propagation of each characteristic. At the stator outflow the characteristic change is

$$\delta \bar{c}_4 = -\sigma \Delta c_4 \tag{5.43}$$

and at the rotor inflow the changes are

$$\begin{aligned} \delta \bar{c}_1 &= \sigma \Delta c_1 \\ \delta \bar{c}_2 &= \sigma \Delta c_2 \\ \delta \bar{c}_3 &= \sigma \Delta c_3 \end{aligned} \tag{5.44}$$

Again the under-relaxation is used to ensure wellposedness and convergence. Now that the average characteristic changes have been calculated for both sides of the interface, the remainder of the boundary condition treatment is exactly the same as for a standard inflow and outflow boundary.

Chapter 6

Unsteady Boundary Conditions

6.1 Overall approach

The unsteady inflow and outflow boundary conditions have to fulfill two distinct functions. The first is the ability to specify unsteady, incoming disturbances. These fall into two categories, wakes and potential disturbances. At the inflow boundary probably the most important unsteady effect is due to wakes shed by the upstream blade row. These can produce a significant unsteady lift on the subsequent blade row which can eventually lead to fatigue and blade failure. Thus it is important to be able to prescribe incoming wakes to calculate their effect. The other type of disturbance is due to the potential pressure field associated with the blade row upstream or downstream of the blade row being calculated. Because of the relative motion of the blade rows this causes an unsteady, isentropic pressure disturbance, which can also produce significant unsteady forces if the spacing between the blade rows is small.

The first three sections in this chapter present the mathematical models of the wakes and potential disturbances which can be specified in UNSFLO, and the combined specified flow field at the inflow and outflow boundaries. These incoming disturbances are only used in UNSFLO when it is being used to calculate a single blade row. In the stator/rotor interaction mode it is assumed that there are no unsteady incoming waves, and so the prescribed inflow and outflow are steady, and uniform except for small non-uniformities taken from a previous steady calculation. Also, in presenting the details of the wake and potential disturbances, it is written as if the blade row being calculated is the rotor, and so the wake has been shed by an upstream stator row. Thus the wake pitch is equal to the stator pitch P_s, and the wake frame of reference is moving with velocity $(0, -V)^T$ relative to the rotor blade row. Of course, UNSFLO also handles the case in which there is a rotor wake being swept into a stator blade row, by changing the sign of V and interchanging the roles of the stator and rotor pitches.

The second function of the boundary conditions is to be transparent to outgoing waves, particularly pressure waves, so that they are not artificially reflected when they reach the boundary. This is achieved by implementing the non-reflecting boundary condition theory presented in Ref. [13]. The last two sections of this chapter show how this is done for both the inflow and the outflow boundaries.

6.2 Prescribed wake models

The wake models, describing the form of incoming wakes at the inflow boundary, assume that in the wake's frame of reference the flow is parallel, with uniform static pressure, uniform total enthalpy, and a prescribed velocity defect.

$$\begin{aligned}
p_w &= p_{F_w} \\
u_w &= (1 - D\,d(\eta))\,u_{F_w} \\
v_w &= (1 - D\,d(\eta))\,v_{F_w} \\
\rho_w &= \frac{\gamma}{\gamma-1}\frac{p_{F_w}}{\left(H_{F_w} - \frac{1}{2}(u_w^2 + v_w^2)\right)}
\end{aligned} \tag{6.1}$$

The subscript $_w$ denotes the wake flow values and the flow variables with the subscript $_{Fw}$ are flux-averaged values. These are obtained from a prior calculation of a steady flow, but have to be modified since the wake frame is assumed to be moving with velocity $(0, -V)^T$ relative to the blade row being calculated.

$$\begin{aligned}
\rho_{F_w} &= \rho_F \\
u_{F_w} &= u_F \\
v_{F_w} &= v_F + V \\
p_{F_w} &= p_F \\
\tan(\alpha_w) &= \frac{v_{F_w}}{u_{F_w}}
\end{aligned} \tag{6.2}$$

D is the fractional velocity defect, $d(\eta)$ is a shape function describing the form of the velocity defect, and η is defined as

$$\eta = \frac{y - \tan(\alpha_w)\,x}{P_s}. \tag{6.3}$$

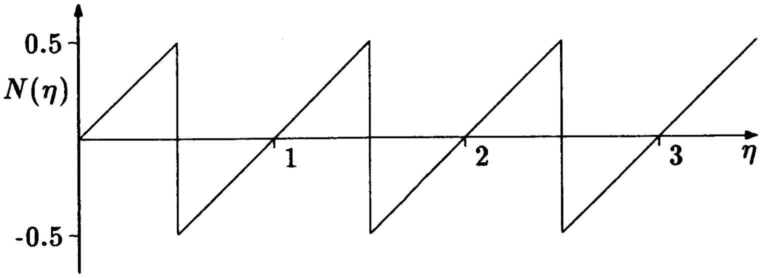

Figure 6.1: Definition of sawtooth function $N(\eta)$

Three different shape functions have been implemented in UNSFLO. The first is a simple sinusoidal function, which has been used to validate the program and the concept of time-inclined planes [12]. The second is a Gaussian velocity defect, and the third is a very similar shape function used by Hodson [18].

$$\begin{aligned} d_1(\eta) &= \cos(2\pi\eta) \\ d_2(\eta) &= \exp\left(-\frac{N(\eta)^2}{2W^2}\right) \\ d_3(\eta) &= \left(\max\left\{0, 1 - \left(\frac{N(\eta)}{W}\right)^{\frac{3}{2}}\right\}\right)^2 \end{aligned} \tag{6.4}$$

W is the wake width (expressed as a fraction of the wake pitch) and $N(\eta)$ is a periodic sawtooth function, shown in Fig. 6.1, which can be expressed as,

$$N(\eta) = \eta - n\ , \qquad n - \frac{1}{2} < \eta < n + \frac{1}{2} \tag{6.5}$$

The assumption that the total enthalpy is uniform is a good approximation for wakes shed from adiabatic blades. One could also specify a periodic variation in the total enthalpy to model the effects of hot streaks, injected film cooling or cooled blades.

6.3 Prescribed potential disturbances

The potential disturbances are derived as linear, isentropic, irrotational perturbations to a uniform flow. The steady, linear potential equation is

$$(1 - M_x^2)\frac{\partial^2\phi}{\partial x^2} - 2M_xM_y\frac{\partial^2\phi}{\partial x\partial y} + (1 - M_y^2)\frac{\partial^2\phi}{\partial y^2} = 0. \tag{6.6}$$

M_x and M_y are the axial and circumferential Mach numbers in the stator frame of reference, in which the flow is steady. Consider a steady disturbance which is periodic in the circumferential direction and has some form of exponential behavior in the axial direction.

$$\phi(x,y) = A\ \exp(iky + \lambda x) \tag{6.7}$$

This is a solution of Eq. (6.6) provided that

$$(1-M_x^2)\lambda^2 - 2iM_xM_yk\lambda - (1-M_y^2)k^2 = 0. \tag{6.8}$$

Solution of this quadratic equation for λ yields

$$\lambda = \frac{ikM_xM_y \pm k\sqrt{1-M^2}}{1-M_x^2}. \tag{6.9}$$

Clearly the qualitative behavior of this perturbation depends on whether the flow is subsonic or supersonic, and so these two possibilities will now be considered separately.

6.3.1 Subsonic case

If $M < 1$ then λ has both a real and an imaginary component. If one is interested in perturbations at the inlet boundary then it is not physical to have a perturbation growing exponentially downstream, and so we choose the negative root in Eq. (6.9). The rate of decay depends linearly on k the wavenumber of the Fourier mode in the y-direction, which must be some multiple of $2\pi/P_s$ to satisfy the periodicity condition. As a consequence the most important mode is the fundamental mode $k=2\pi/P_s$ and so the model which is used assumes that only this mode is present.

$$\phi(x,y) = A\ \exp\left(-\frac{2\pi}{P_s}\frac{\sqrt{1-M^2}}{1-M_x^2}x\right)\exp\left(i\frac{2\pi}{P_s}(y-\tan(\alpha_p)\,x)\right) \tag{6.10}$$

where,

$$\tan(\alpha_p) = -\frac{M_xM_y}{1-M_x^2} \tag{6.11}$$

Differentiating ϕ gives the perturbation velocities with the understanding that the physical velocity is the real part of the following expressions.

$$\delta v = i\frac{2\pi}{P_s}\phi \tag{6.12}$$

$$\begin{aligned}\delta u &= -i\frac{2\pi}{P_s}\tan(\alpha_p)\,\phi - \frac{2\pi}{P_s}\frac{\sqrt{1-M^2}}{1-M_x^2}\phi \\ &= \left(-\tan(\alpha_p) + i\frac{\sqrt{1-M^2}}{1-M_x^2}\right)\delta v\end{aligned} \tag{6.13}$$

Since the complex potential amplitude A has no simple physical meaning it is more convenient to define Δv_p to be the maximum v perturbation at x_{te}, the nominal location of the trailing edge of the upstream stator blade row. With this definition the final forms for δu and δv are,

$$\begin{aligned} \delta v &= -\Delta v_p \exp\left(-\frac{2\pi}{P_s}\frac{\sqrt{1-M^2}}{1-M_x^2}(x-x_{te})\right)\sin(2\pi(\epsilon+\Delta\epsilon)) && (6.14) \\ \delta u &= -\frac{\sqrt{1-M^2}}{1-M_x^2}\Delta v_{ip}\exp\left(-\frac{2\pi}{P_s}\frac{\sqrt{1-M^2}}{1-M_x^2}(x-x_{te})\right)\cos(2\pi(\epsilon+\Delta\epsilon)) \\ &\quad - \tan(\alpha_p)\,\delta v && (6.15) \end{aligned}$$

where,

$$\epsilon = \frac{y-\tan(\alpha_p)\,x}{P_s} \qquad (6.16)$$

and $\Delta\epsilon$ is a phase constant.

The corresponding density and pressure perturbations are obtained from the conditions that there are no variations in either the entropy or the total enthalpy.

$$\frac{\delta p}{p} - \gamma\frac{\delta\rho}{\rho} = 0 \qquad (6.17)$$

$$\frac{\gamma}{\gamma-1}\left(\frac{\delta p}{\rho} - \frac{p\,\delta\rho}{\rho^2}\right) + u\,\delta u + v\,\delta v = 0 \qquad (6.18)$$

Combining these two equations gives

$$\begin{aligned} \delta p &= -\rho(u\,\delta u + v\,\delta v) \\ \delta\rho &= \delta p/c^2, && (6.19) \end{aligned}$$

and substituting for δu and δv gives the following expression for the pressure variation at $x=x_{te}$.

$$\begin{aligned} \delta p &= \rho u\,\Delta v_p\left(\frac{\sqrt{1-M^2}}{1-M_x^2}\cos(2\pi(\epsilon+\Delta\epsilon)) + \left(\frac{M_y}{M_x}+\frac{M_xM_y}{1-M_x^2}\right)\sin(2\pi(\epsilon+\Delta\epsilon))\right) \\ &= \frac{\rho u\,\Delta v_p}{1-M_x^2}\left(\sqrt{1-M^2}\cos(2\pi(\epsilon+\Delta\epsilon)) + \frac{M_y}{M_x}\sin(2\pi(\epsilon+\Delta\epsilon))\right) \\ &= \frac{\rho c M\,\Delta v_p}{\sqrt{1-M_x^2}}\cos\left(2\pi(\epsilon+\Delta\epsilon) - \tan^{-1}\left(\frac{\tan(\alpha_w)}{\sqrt{1-M^2}}\right)\right) && (6.20) \end{aligned}$$

Thus the maximum pressure disturbance, Δp_{ip} is related to the maximum velocity disturbance by

$$\Delta p_p = \frac{\rho c M\,\Delta v_{ip}}{\sqrt{1-M_x^2}}, \qquad (6.21)$$

and the phase constant $\Delta\epsilon$ is chosen to be

$$\Delta\epsilon = \frac{1}{2\pi}\tan^{-1}\left(\frac{\tan(\alpha_w)}{\sqrt{1-M^2}}\right) + \frac{(\tan(\alpha_p) - \tan(\alpha_w))\, x_{te}}{P_s} \tag{6.22}$$

so that the pressure maximum from the potential disturbance is located on the wake centerline at the position corresponding to the stator trailing edge.

So far the discussion has dealt solely with the incoming potential disturbance at the inflow boundary. The corresponding formulae for the outflow boundary are only slightly different due to the different choice of root discussed earlier.

6.3.2 Supersonic case

If $M > 1$, but $M_x < 1$ so that the flow is still axially subsonic, then λ has only an imaginary component and ϕ can be written as,

$$\phi(x,y) = A\,\exp\left(i\frac{2\pi}{P_s}(y - \tan(\alpha_p)x)\right) \tag{6.23}$$

where,

$$\tan(\alpha_p) = \frac{-M_xM_y \pm \sqrt{M^2-1}}{1 - M_x^2} \tag{6.24}$$

Combining all the Fourier modes, since now there is no spatial decay of any mode, the most general form of ϕ is,

$$\phi(x,y) = A\, f(y - \tan(\alpha_p)x) \tag{6.25}$$

where f is some periodic function.

To determine which root should be chosen in Eq. (6.24), note that using the usual supersonic characteristic theory the angle of the incoming supersonic characteristic should be equal to the flow angle, α_w, *minus* $\beta = \sqrt{M^2-1}$ if $\alpha_w > 0$, and *plus* β if $\alpha_w < 0$. Using standard trigonometric results the following identity is derived.

$$\begin{aligned}
\tan(\alpha_w \mp \beta) &= \frac{\tan(\alpha_w) \mp \tan(\beta)}{1 \pm \tan(\alpha_w)\tan(\beta)} \\
&= \frac{\frac{M_y}{M_x} \mp \frac{1}{\sqrt{M^2-1}}}{1 \pm \frac{M_y}{M_x}\frac{1}{\sqrt{M^2-1}}} \\
&= \frac{M_y\sqrt{M^2-1} \mp M_x}{M_x\sqrt{M^2-1} \mp M_y}\,\frac{M_x\sqrt{M^2-1} \pm M_y}{M_x\sqrt{M^2-1} \mp M_y} \\
&= \frac{M_xM_yM^2 \mp M^2\sqrt{M^2-1}}{M_x^2(M^2-1) - M_y^2}
\end{aligned}$$

$$= \frac{-M_x M_y \pm \sqrt{M^2 - 1}}{1 - M_x^2} \tag{6.26}$$

Thus the positive root in Eq. (6.24) should be chosen if $\alpha_w > 0$, and the negative root if $\alpha_w < 0$.

The model used for the incoming perturbation represents a weak oblique shock generated at the trailing edge of the stator. As in the subsonic case it is easier to deal directly with the velocities rather than the potential and so the chosen form is

$$\begin{aligned} \delta v &= 2\Delta v_p N(\epsilon + \Delta\epsilon) && (6.27) \\ \delta u &= -\tan(\alpha_p)\,\delta v, && (6.28) \end{aligned}$$

where

$$\epsilon = \frac{y - \tan(\alpha_p)\,x}{P_s}. \tag{6.29}$$

The function $N(\epsilon)$ is the same sawtooth function used in the wake definition. $\delta\rho$ and δp are obtained again from Eq. (6.19). The maximum pressure disturbance Δp_p is

$$\Delta p_p = \rho u \Delta v_p \left(\tan(\alpha_w) - \tan(\alpha_p)\right) \tag{6.30}$$

and the phase constant $\Delta\epsilon$ is chosen to be

$$\Delta\epsilon = \tfrac{1}{2} + \frac{(\tan(\alpha_p) - \tan(\alpha_w))\,x_{te}}{P_s} \tag{6.31}$$

so that the shock crosses the centerline of the wake at the stator trailing edge.

6.4 Combined flow field specification

The prescribed inlet flow in the rotor frame is a combination of the nonlinear wake model, plus the linear potential disturbance, plus the steady nonuniformity across the rotor inflow boundary which comes from a previous steady calculation. In addition the rotor wheel speed must be subtracted from the circumferential velocity because of the shift from the stator frame to the rotor frame.

$$\begin{aligned} \rho_{inl}(x,y,t) &= \rho_w(\eta) + \delta\rho_p(x,\epsilon) + \delta\rho_j \\ u_{inl}(x,y,t) &= u_w(\eta) + \delta u_p(x,\epsilon) + \delta u_j \\ v_{inl}(x,y,t) &= v_w(\eta) + \delta v_p(x,\epsilon) + \delta v_j - V && (6.32) \\ p_{inl}(x,y,t) &= p_w(\eta) + \delta p_p(x,\epsilon) + \delta p_j \end{aligned}$$

where,

$$\begin{aligned}\eta &= \frac{y+Vt-\tan(\alpha_w)\,x}{P_s}\\ \epsilon &= \frac{y+Vt-\tan(\alpha_p)\,x}{P_s},\end{aligned} \tag{6.33}$$

and the flow variables with subscript $_j$ are the difference between local values and average values coming from a steady rotor blade row calculation.

$$\begin{aligned}\delta\rho_j &= \rho_j-\rho_F\\ \delta u_j &= u_j-u_F\\ \delta v_j &= v_j-v_F\\ \delta p_j &= p_j-p_F\end{aligned} \tag{6.34}$$

The point of including the steady state nonuniformity in the prescribed flow definition is that when no incoming wake or potential disturbance is specified the steady state flow solution should be the correct solution to the unsteady flow problem.

With the coordinate transformation from the physical coordinates to the inclined computational coordinates, the equations above remain the same except for the definitions of η and ϵ which become

$$\begin{aligned}\eta &= \frac{y'}{P_r}+\frac{Vt'-\tan(\alpha_w)\,x'}{P_s}\\ \epsilon &= \frac{y'}{P_r}+\frac{Vt'-\tan(\alpha_p)\,x'}{P_s},\end{aligned} \tag{6.35}$$

The effect of the time-tilting in solving the problem of unequal pitches is clear in that both η and ϵ are increased by 1 when y' is increased by one blade pitch P_r, and so the inclined inflow specification is spatially periodic, with period P_r.

The outflow specification is simpler because there is no wake flow, and so it is simply the flux-averaged flow field plus the potential disturbance and the steady state nonuniformity.

$$\begin{aligned}\rho_{out}(x,y,t) &= \rho_F+\delta\rho_p(x,\epsilon)+\delta\rho_j\\ u_{out}(x,y,t) &= u_F+\delta u_p(x,\epsilon)+\delta u_j\\ v_{out}(x,y,t) &= v_F+\delta v_p(x,\epsilon)+\delta v_j\\ p_{out}(x,y,t) &= p_F+\delta p_p(x,\epsilon)+\delta p_j\end{aligned} \tag{6.36}$$

where,

$$\epsilon = \frac{y+Vt-\tan(\alpha_p)\,x}{P_s}. \tag{6.37}$$

In the inclined computational plane ϵ becomes

$$\epsilon = \frac{y'}{P_r} + \frac{Vt' - \tan(\alpha_p)\, x'}{P_s}. \qquad (6.38)$$

6.5 Inflow boundary

At the inflow boundary the characteristic variables are defined in terms of perturbations of the flow from the prescribed inlet flow.

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 \\ 0 & \rho c & 0 & 1 \\ 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} \rho - \rho_{inl} \\ u - u_{inl} \\ v - v_{inl} \\ p - p_{inl} \end{pmatrix} \qquad (6.39)$$

The objective now is to construct boundary conditions for these perturbations so that the boundary is transparent to outgoing waves and does not produce significant spurious reflections. Ref. [13] shows that this is achieved analytically by setting the first characteristic variable to zero,

$$c_1 = 0, \qquad (6.40)$$

and letting the second and third characteristics satisfy a first order partial differential equation along the inflow boundary.

$$\frac{\partial}{\partial t}\begin{pmatrix} c_2 \\ c_3 \end{pmatrix} + \begin{pmatrix} v & \frac{1}{2}(c+u) & \frac{1}{2}(c-u) \\ \frac{1}{2}(c-u) & v & 0 \end{pmatrix} \frac{\partial}{\partial y} \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix} = 0 \qquad (6.41)$$

When transformed into the inclined computational coordinates, the last equation becomes

$$\begin{pmatrix} 1-\lambda v & -\frac{1}{2}\lambda(c+u) & -\frac{1}{2}\lambda(c-u) \\ -\frac{1}{2}\lambda(c-u) & 1-\lambda v & 0 \end{pmatrix} \frac{\partial}{\partial t'} \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix} +$$

$$\begin{pmatrix} v & \frac{1}{2}(c+u) & \frac{1}{2}(c-u) \\ \frac{1}{2}(c-u) & v & 0 \end{pmatrix} \frac{\partial}{\partial y'} \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix} = 0 \qquad (6.42)$$

Premultiplying by the correct matrix inverse converts this into the following form.

$$\frac{\partial}{\partial t'}\begin{pmatrix} c_2 \\ c_3 \end{pmatrix} + B_1 \frac{\partial c_4}{\partial t'} + B_2 \frac{\partial}{\partial y'} \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix} = 0 \qquad (6.43)$$

B_1 is a 2×1 vector and B_2 is a 2×3 matrix.

The numerical implementation of these boundary conditions begins by calculating U^n_{inl} and U^{n+1}_{inl}, the prescribed inlet flow at the beginning and end of the time step. The next step is to evaluate the four characteristic variables at the beginning of the time-step, using Eq. (6.39). At the new time level $n{+}1$ the first characteristic variable is zero, but the other three are evaluated by calculating their changes over the timestep.

The change in the outgoing fourth characteristic is obtained from the distributed Lax-Wendroff changes at the boundary node *minus* the change due to the unsteady prescribed flow. Thus,

$$\delta c_4 = \delta p_{LW-inl} - \rho c\, \delta u_{LW-inl}, \tag{6.44}$$

where

$$\begin{aligned}
\delta \rho_{LW-inl} &= \delta U_{1LW} - \delta U_{1\,inl} \\
\delta u_{LW-inl} &= (\delta U_{2LW} - \delta U_{2\,inl} - u\,\delta \rho_{LW-inl})/\rho \\
\delta v_{LW-inl} &= (\delta U_{3LW} - \delta U_{3\,inl} - v\,\delta \rho_{LW-inl})/\rho \\
\delta p_{LW-inl} &= (\gamma-1)(\delta U_{4LW} - \delta U_{4\,inl} - \rho u\,\delta u_{LW-inl} - \rho v\,\delta v_{LW-inl} - \tfrac{1}{2}(u^2+v^2)\delta \rho_{LW-inl})
\end{aligned} \tag{6.45}$$

The changes in the second and third characteristics are obtained by integrating in time Eq. (6.43) using a one-dimensional Lax-Wendroff algorithm. The changes at the center of the j^{th} face are given by

$$\Delta c_{4\,j+\frac{1}{2}} = \tfrac{1}{2}(\delta c_{4\,j} + \delta c_{4\,j+1}), \tag{6.46}$$

and

$$\begin{pmatrix} \Delta c_2 \\ \Delta c_3 \end{pmatrix}_{j+\frac{1}{2}} = -\,B_1 \Delta c_{4\,j+\frac{1}{2}} - \frac{\Delta t}{\Delta y} B_2 \left\{ \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix}_{j+1} - \begin{pmatrix} c_2 \\ c_3 \\ c_4 \end{pmatrix}_j \right\} \tag{6.47}$$

The changes distributed to the two boundary nodes from the face are

$$\begin{aligned}
\delta c_{2\,j} &= \tfrac{1}{2}(\Delta c_{2\,j+\frac{1}{2}} - \Delta g_{2\,j+\frac{1}{2}}) \\
\delta c_{3\,j} &= \tfrac{1}{2}(\Delta c_{3\,j+\frac{1}{2}} - \Delta g_{3\,j+\frac{1}{2}}) \\
\delta c_{2\,j+1} &= \tfrac{1}{2}(\Delta c_{2\,j+\frac{1}{2}} + \Delta g_{2\,j+\frac{1}{2}}) \\
\delta c_{3\,j+1} &= \tfrac{1}{2}(\Delta c_{3\,j+\frac{1}{2}} + \Delta g_{3\,j+\frac{1}{2}}),
\end{aligned} \tag{6.48}$$

where the second order fluxes, Δg, are defined by

$$\begin{pmatrix} \Delta g_2 \\ \Delta g_3 \end{pmatrix}_{j+\frac{1}{2}} = \frac{\Delta t}{\Delta y} B_2 \begin{pmatrix} \Delta c_2 \\ \Delta c_3 \\ \Delta c_4 \end{pmatrix}_{j+\frac{1}{2}} - \sigma \left\{ \begin{pmatrix} c_2 \\ c_3 \end{pmatrix}_{j+1} - \begin{pmatrix} c_2 \\ c_3 \end{pmatrix}_j \right\} \tag{6.49}$$

The second term in the Δg definition is a numerical smoothing term. A typical value for the coefficient σ is 0.05. Distributing the changes from all of the boundary faces gives the characteristic changes at the nodes.

One final modification must be made. Eq. (6.43) can be integrated over one pitch to obtain the following ordinary differential equation for the pitch-averaged characteristic variables.

$$\frac{d}{dt'}\left\{\begin{pmatrix} \bar{c}_2 \\ \bar{c}_3 \end{pmatrix} + B_1\bar{c}_4\right\} = 0 \tag{6.50}$$

When integrated in time, there is an arbitrary constant of integration. To ensure that the final solution does not depend on the initial conditions, this constant is set equal to zero by subtracting the average values from the local values.

$$\begin{pmatrix} \delta c_2 \\ \delta c_3 \end{pmatrix}^{new} = \begin{pmatrix} \delta c_2 \\ \delta c_3 \end{pmatrix}^{old} - \begin{pmatrix} \bar{c}_2 \\ \bar{c}_3 \end{pmatrix} - B_1\bar{c}_4 \tag{6.51}$$

Using these corrected changes, the new characteristic variables are obtained. The new perturbations to the prescribed inlet flow are then calculated and added to get the new flow solution on the inflow boundary.

6.6 Outflow boundary

The unsteady outflow boundary conditions are similar to the inflow conditions, but are slightly simpler since there is only one incoming characteristic. The characteristic variables are defined by

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} -c^2 & 0 & 0 & 1 \\ 0 & 0 & \rho c & 0 \\ 0 & \rho c & 0 & 1 \\ 0 & -\rho c & 0 & 1 \end{pmatrix} \begin{pmatrix} \rho - \rho_{out} \\ u - u_{out} \\ v - v_{out} \\ p - p_{out} \end{pmatrix} \tag{6.52}$$

and Ref. [13] shows that the best non-reflecting outflow boundary condition is

$$\frac{\partial c_4}{\partial t} + \begin{pmatrix} u & v \end{pmatrix} \frac{\partial}{\partial y} \begin{pmatrix} c_2 \\ c_4 \end{pmatrix} = 0 \tag{6.53}$$

When transformed into the inclined computational coordinates, this equation becomes

$$\begin{pmatrix} -\lambda u & 1-\lambda v \end{pmatrix} \frac{\partial}{\partial t'} \begin{pmatrix} c_2 \\ c_4 \end{pmatrix} + \begin{pmatrix} u & v \end{pmatrix} \frac{\partial}{\partial y'} \begin{pmatrix} c_2 \\ c_4 \end{pmatrix} = 0, \tag{6.54}$$

which can be rearranged into

$$\frac{\partial c_4}{\partial t'} - \frac{\lambda u}{1-\lambda v}\frac{\partial c_2}{\partial t'} + \frac{v}{1-\lambda v}\frac{\partial c_4}{\partial y'} + \frac{u}{1-\lambda v}\frac{\partial c_2}{\partial y'} = 0 \tag{6.55}$$

The numerical implementation begins again by calculating the prescribed outlet flow U^n_{out} and U^{n+1}_{out}, and the characteristic variables at the beginning of the time-step.

The changes in the three outgoing characteristics are obtained from the distributed Lax-Wendroff changes at the boundary node *minus* the changes due to the unsteady prescribed flow.

$$\begin{aligned} \delta c_1 &= \delta p_{LW-out} - c^2 \delta \rho_{LW-out} \\ \delta c_2 &= \rho c \delta v_{LW-out} \\ \delta c_3 &= \delta p_{LW-out} + \rho c \delta u_{LW-out} \end{aligned} \tag{6.56}$$

where

$$\begin{aligned} \delta \rho_{LW-out} &= \delta U_{1LW} - \delta U_{1\,out} \\ \delta u_{LW-out} &= (\delta U_{2LW} - \delta U_{2\,out} - u\,\delta \rho_{LW-out})/\rho \\ \delta v_{LW-out} &= (\delta U_{3LW} - \delta U_{3\,out} - v\,\delta \rho_{LW-out})/\rho \\ \delta p_{LW-out} &= (\gamma-1)(\delta U_{4LW} - \delta U_{4\,out} - \rho u\,\delta u_{LW-out} - \rho v\,\delta v_{LW-out} - \tfrac{1}{2}(u^2+v^2)\delta \rho_{LW-out}) \end{aligned} \tag{6.57}$$

The change in the fourth characteristic is obtained by integrating Eq. (6.55), again using a one-dimensional Lax-Wendroff algorithm. The face center changes are

$$\Delta c_{2\,j+\frac{1}{2}} = \tfrac{1}{2}(\delta c_{2\,j} + \delta c_{2\,j+1}), \tag{6.58}$$

and

$$\Delta c_{4\,j+\frac{1}{2}} = -\frac{\lambda u}{1-\lambda v}\Delta c_{2\,j+\frac{1}{2}} - \frac{\Delta t}{\Delta y}\left(\frac{v}{1-\lambda v}(c_{4\,j+1} - c_{4\,j}) + \frac{u}{1-\lambda v}(c_{2\,j+1} - c_{2\,j})\right) \tag{6.59}$$

The distributed changes to the nodes are

$$\begin{aligned} \delta c_{4\,j} &= \tfrac{1}{2}(\Delta c_{4\,j+\frac{1}{2}} - \Delta g_{4\,j+\frac{1}{2}}) \\ \delta c_{4\,j+1} &= \tfrac{1}{2}(\Delta c_{4\,j+\frac{1}{2}} + \Delta g_{4\,j+\frac{1}{2}}), \end{aligned} \tag{6.60}$$

where the second order flux, Δg_4, is defined by

$$\Delta g_{4\,j+\frac{1}{2}} = \frac{\Delta t}{\Delta y}\left(\frac{v}{1-\lambda v}\Delta c_{4\,j+\frac{1}{2}} + \frac{u}{1-\lambda v}\Delta c_{2\,j+\frac{1}{2}}\right) - \sigma(c_{4\,j+1} - c_{4\,j}). \tag{6.61}$$

The numerical smoothing coefficient, σ, again has a typical value of 0.05. As with the inflow boundary conditions, Eq. (6.55) can be integrated to obtain

$$\frac{d}{dt'}\left(\bar{c}_4 - \frac{\lambda u}{1-\lambda v}\bar{c}_2\right) = 0. \tag{6.62}$$

To set the constant of integration to zero, the characteristic changes are modified by subtracting the corresponding pitch-averaged values.

$$\delta c_{4j}^{new} = \delta c_{4j}^{old} - \bar{c}_4 + \frac{\lambda u}{1-\lambda v}\bar{c}_2 = 0. \tag{6.63}$$

Using these characteristic changes, the new characteristic variables are obtained, and hence the new outflow variables.

Chapter 7

Viscous Algorithm

7.1 Overview

The standard approach to viscous calculations is to use everywhere throughout the computational domain one numerical algorithm, which is suitable for viscous calculations but may not evaluate the viscous stresses in regions where they are small. In UNSFLO an alternative approach was chosen, in which there are two numerical algorithms. A viscous algorithm is used in a relatively thin viscous grid around each blade, and the inviscid Lax-Wendroff algorithm is used in the rest of the domain as described in earlier chapters.

There were several reasons for this choice of approach. The explicit Lax-Wendroff algorithm could have been modified to peform viscous calculations, but in this case the maximum stable timestep is extremely small in the boundary layer, and so the computational cost would have been excessive. However, the Lax-Wendroff algorithm is very good for the inviscid region because of its efficieny, its second order accuracy on irregular meshes, and the ability to add grid adaptation at a later time. To avoid the excessive timestep restriction in the boundary layer requires an implicit algorithm. All of the established implicit codes use structured grids and are cell-based, meaning that the flow variables are stored at the centers of the computational cells. Such methods are more significantly more expensive than explicit methods, and lack all of the other features of the Lax-Wendroff algorithm listed above. Therefore, it was concluded that the best approach was the hybrid one, using two separate algorithms on two separate grids, suitably connected at the interface.

7.2 Basic algorithm

In the viscous boundary layer region there are two grids. The first grid is created by the grid generator, and is structured, with each grid node having a (j,k) index. $k=1$ corresponds to the line of nodes on the blade surface. $k=K$ corresponds to the line of nodes at the interface with the inviscid grid; each of these nodes is at the same position as an inviscid grid node. The viscous flow variables are stored at the grid nodes of this first grid, and the grid is used in plotting contours in the viscous region.

The second grid is the one which is used by the viscous algorithm. It is created by joining the centers of the cells of the first grid, to form an overlapping grid with the flow variables being stored approximately at the center of each new cell. Thus,

$$\begin{aligned} x_{j+\frac{1}{2},k+\frac{1}{2}} &= \frac{1}{4}\left(x_{j,k}+x_{j,k+1}+x_{j+1,k}+x_{j+1,k+1}\right) \\ y_{j+\frac{1}{2},k+\frac{1}{2}} &= \frac{1}{4}\left(y_{j,k}+y_{j,k+1}+y_{j+1,k}+y_{j+1,k+1}\right) \end{aligned} \tag{7.1}$$

The basic algorithm is based upon the ADI scheme of Beam and Warming [2], but uses the upwind flux-difference splitting developed by Roe [32]. It is similar to the scheme used by Rai [30]. The two-dimensional unsteady Navier-Stokes equations, with variable streamtube thickness, are

$$h\frac{\partial U}{\partial t}+\frac{\partial(hF)}{\partial x}+\frac{\partial(hG)}{\partial y}-\frac{\partial(hV_x)}{\partial x}-\frac{\partial(hV_y)}{\partial y}=S, \tag{7.2}$$

where U,F,G and S are the same as in the Euler equations, and V_x and V_y are viscous fluxes defined by

$$V_x=\begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx}+v\tau_{xy}-q_x \end{pmatrix} \tag{7.3}$$

$$V_y=\begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy}+v\tau_{yy}-q_y \end{pmatrix}. \tag{7.4}$$

The heat flux terms q_x,q_y are given by

$$\begin{aligned} q_x &= -k\frac{\partial T}{\partial x} = -\frac{\mu}{(\gamma-1)Pr}\frac{\partial(c^2)}{\partial x} \\ q_y &= -k\frac{\partial T}{\partial y} = -\frac{\mu}{(\gamma-1)Pr}\frac{\partial(c^2)}{\partial y}, \end{aligned} \tag{7.5}$$

and the stress terms are given by

$$\begin{aligned} \tau_{xx} &= \mu(\frac{4}{3}\frac{\partial u}{\partial x} - \frac{2}{3}\frac{\partial v}{\partial y}) \\ \tau_{xy} &= \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) \\ \tau_{yy} &= \mu(\frac{4}{3}\frac{\partial v}{\partial y} - \frac{2}{3}\frac{\partial u}{\partial x}) \end{aligned} \tag{7.6}$$

Under a high Reynolds number approximation in which streamwise derivatives are neglected relative to normal derivatives across the boundary layer, the viscous contributions simplify considerably.

$$\frac{\partial(hV_x)}{\partial x} + \frac{\partial(hV_y)}{\partial y} \approx \frac{\partial(hV_n)}{\partial n}, \tag{7.7}$$

where

$$V_n = \begin{pmatrix} 0 \\ \mu\frac{\partial u}{\partial n} \\ \mu\frac{\partial v}{\partial n} \\ \mu\frac{\partial}{\partial n}(\frac{1}{2}u^2 + \frac{1}{2}v^2 + \frac{1}{(\gamma-1)Pr}c^2) \end{pmatrix} \tag{7.8}$$

To complete the specification of these thin-layer Navier-Stokes equations, the Prandtl number Pr is a constant (0.72 for air), and the viscosity μ is given by Sutherland's law.

$$\begin{aligned} \mu &= \mu_{ref}\left(\frac{T}{T_{ref}}\right)^{3/2}\frac{T_{ref}+T_{con}}{T+T_{con}} \\ &= \mu_{ref}\left(\frac{c^2}{c^2_{ref}}\right)^{3/2}\frac{c^2_{ref}+c^2_{con}}{c^2+c^2_{con}} \end{aligned} \tag{7.9}$$

T_{con} is a temperature constant in the Sutherland model, T_{ref} is the reference temperature at which $\mu = \mu_{ref}$, and c_{con} and c_{ref} are the corresponding speeds of sound.

Approximating the spatial derivatives on the computational cell shown in Figure 7.1 produces the following semi-discrete equation.

$$A'_{j,k}\frac{dU_{j,k}}{dt} + (F^*_{j+\frac{1}{2},k} - F^*_{j-\frac{1}{2},k}) + (G^*_{j,k+\frac{1}{2}} - G^*_{j,k-\frac{1}{2}}) - (V^*_{j,k+\frac{1}{2}} - V^*_{j,k-\frac{1}{2}}) = S^*_{j,k} \tag{7.10}$$

$A'_{j,k}$ is the volume of the computational cell, which is the product of the area and the streamtube thickness. F^* is the inviscid flux through one of the cell faces lying approximately in the normal direction, and G^* and V^* are the inviscid and viscous fluxes through one of the faces lying approximately in the streamwise direction. Omitting

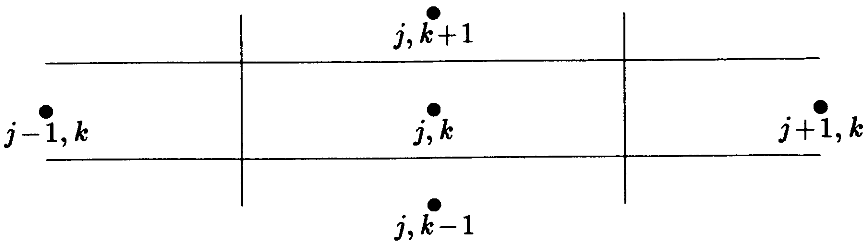

Figure 7.1: Grid geometry for viscous algorithm

certain upwinding terms which will be discussed later, these discrete fluxes are defined by the following equations.

$$F^*_{j+\frac{1}{2},k} = \frac{1}{2}\left((F_{j,k} + F_{j+1,k})\Delta y'_{j+\frac{1}{2},k} - (G_{j,k} + G_{j+1,k})\Delta x'_{j+\frac{1}{2},k}\right) \tag{7.11}$$

$$G^*_{j,k+\frac{1}{2}} = \frac{1}{2}\left((F_{j,k} + F_{j,k+1})\Delta y'_{j,k+\frac{1}{2}} - (G_{j,k} + G_{j,k+1})\Delta x'_{j,k+\frac{1}{2}}\right) \tag{7.12}$$

$$\begin{aligned} V^*_{j,k+\frac{1}{2}} &= \frac{\mu_{j,k} + \mu_{j,k+1}}{2}\,\frac{\Delta s'}{\Delta n} \times \\ &\begin{pmatrix} 0 \\ u_{j,k+1} - u_{j,k} \\ v_{j,k+1} - v_{j,k} \\ (\frac{1}{2}u^2_{j,k+1} + \frac{1}{2}v^2_{j,k+1} + \frac{1}{(\gamma-1)Pr}c^2_{j,k+1}) - (\frac{1}{2}u^2_{j,k} + \frac{1}{2}v^2_{j,k} + \frac{1}{(\gamma-1)Pr}c^2_{j,k}) \end{pmatrix} \end{aligned} \tag{7.13}$$

where

$$\begin{aligned} \Delta x'_{j+\frac{1}{2},k} &= \frac{h_{j,k} + h_{j+1,k}}{2}\,(x_{j+\frac{1}{2},k+\frac{1}{2}} - x_{j+\frac{1}{2},k-\frac{1}{2}}) \\ \Delta y'_{j+\frac{1}{2},k} &= \frac{h_{j,k} + h_{j+1,k}}{2}\,(y_{j+\frac{1}{2},k+\frac{1}{2}} - y_{j+\frac{1}{2},k-\frac{1}{2}}) \\ \Delta x'_{j,k+\frac{1}{2}} &= \frac{h_{j,k} + h_{j,k+1}}{2}\,(x_{j-\frac{1}{2},k+\frac{1}{2}} - x_{j+\frac{1}{2},k+\frac{1}{2}}) \\ \Delta y'_{j,k+\frac{1}{2}} &= \frac{h_{j,k} + h_{j,k+1}}{2}\,(y_{j-\frac{1}{2},k+\frac{1}{2}} - y_{j+\frac{1}{2},k+\frac{1}{2}}) \end{aligned} \tag{7.14}$$

and

$$\frac{\Delta s'}{\Delta n} = \frac{(\Delta x'_{j,k+\frac{1}{2}})^2 + (\Delta y'_{j,k+\frac{1}{2}})^2}{\Delta x'_{j,k+\frac{1}{2}}(y_{j,k} - y_{j,k+1}) - \Delta y'_{j,k+\frac{1}{2}}(x_{j,k} - x_{j,k+1})} \tag{7.15}$$

The source term is approximated in a manner which ensures that a uniform pressure field produces no net force on the control volume.

$$S^*_{j,k} = p_{j,k} \left(\begin{array}{c} 0 \\ +\Delta y'_{j+\frac{1}{2},k} - \Delta y'_{j-\frac{1}{2},k} + \Delta y'_{j,k+\frac{1}{2}} - \Delta y'_{j,k-\frac{1}{2}} \\ -\Delta x'_{j+\frac{1}{2},k} + \Delta x'_{j-\frac{1}{2},k} - \Delta x'_{j,k+\frac{1}{2}} + \Delta x'_{j,k-\frac{1}{2}} \\ 0 \end{array} \right) \tag{7.16}$$

The fully discrete equations are obtained by approximating the time derivative using backward differencing.

$$\frac{A'_{j,k}}{\Delta t}\left(U^{n+1}_{j,k} - U^{n}_{j,k}\right) + (F^{*^{n+1}}_{j+\frac{1}{2},k} - F^{*^{n+1}}_{j-\frac{1}{2},k}) + (G^{*^{n+1}}_{j,k+\frac{1}{2}} - G^{*^{n+1}}_{j,k-\frac{1}{2}}) - (V^{*^{n+1}}_{j,k+\frac{1}{2}} - V^{*^{n+1}}_{j,k-\frac{1}{2}}) = S^{*^{n+1}}_{j,k} \tag{7.17}$$

This is a nonlinear system of equations which cannot in general be solved to obtain U^{n+1}. Instead, each of the flux terms is linearized about U^n to obtain the following 'unfactored linearized-delta' equation.

$$\begin{aligned} &\left(\frac{A'}{\Delta t} + \frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j,k}} - \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j,k}} + \frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} + \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}}\right) \Delta U_{j,k} \\ &\qquad + \frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j+1,k}} \Delta U_{j+1,k} - \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j-1,k}} \Delta U_{j-1,k} \\ &+ \left(\frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}}\right) \Delta U_{j,k+1} - \left(\frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}} - \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}}\right) \Delta U_{j,k-1} \quad = \\ &- (F^{*^n}_{j+\frac{1}{2},k} - F^{*^n}_{j-\frac{1}{2},k}) - (G^{*^n}_{j,k+\frac{1}{2}} - G^{*^n}_{j,k-\frac{1}{2}}) + (V^{*^n}_{j,k+\frac{1}{2}} - V^{*^n}_{j,k-\frac{1}{2}}) + S^{*^n}_{j,k} \end{aligned} \tag{7.18}$$

For simplicity, the linearized change in the source term has been ignored.

This is now a linear system of equations, but its solution is computationally time-consuming since the work for a viscous grid of size $J \times K$ is proportional to the smaller of J^3K and K^3J. This direct solution cost is avoided by an iterative line relaxation approach. At the beginning of each time step $\Delta U^{(0)}$ is initialized to zero. An approximate solution $\Delta U^{(1)}$ is obtained by solving

$$\begin{aligned} &\left(\frac{A'}{\Delta t} + \frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j,k}} - \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j,k}}\right) \Delta U^{(1)}_{j,k} + \frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j+1,k}} \Delta U^{(1)}_{j+1,k} - \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j-1,k}} \Delta U^{(1)}_{j-1,k} \quad = \\ &\qquad - (F^{*^n}_{j+\frac{1}{2},k} - F^{*^n}_{j-\frac{1}{2},k}) - (G^{*^n}_{j,k+\frac{1}{2}} - G^{*^n}_{j,k-\frac{1}{2}}) + (V^{*^n}_{j,k+\frac{1}{2}} - V^{*^n}_{j,k-\frac{1}{2}}) + S^{*^n}_{j,k} \\ &\qquad - \left(\frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} + \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}}\right) \Delta U^{(0)}_{j,k} \end{aligned}$$

$$-\left(\frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}}\right)\Delta U^{(0)}_{j,k+1} + \left(\frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}} - \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}}\right)\Delta U^{(0)}_{j,k-1} \qquad (7.19)$$

and an better approximation $\Delta U^{(2)}$ is obtained by solving

$$\left(\frac{A'}{\Delta t} + \frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} + \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}}\right)\Delta U^{(2)}_{j,k}$$

$$+\left(\frac{\partial G^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}}\right)\Delta U^{(2)}_{j,k+1} - \left(\frac{\partial G^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}} - \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}}\right)\Delta U^{(2)}_{j,k-1} \quad =$$

$$-(F^{*^n}_{j+\frac{1}{2},k} - F^{*^n}_{j-\frac{1}{2},k}) - (G^{*^n}_{j,k+\frac{1}{2}} - G^{*^n}_{j,k-\frac{1}{2}}) + (V^{*^n}_{j,k+\frac{1}{2}} - V^{*^n}_{j,k-\frac{1}{2}}) + S^{*^n}_{j,k}$$

$$-\left(\frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j,k}} - \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j,k}}\right)\Delta U^{(1)}_{j,k} - \frac{\partial F^*_{j+\frac{1}{2},k}}{\partial U_{j+1,k}}\Delta U^{(1)}_{j+1,k} + \frac{\partial F^*_{j-\frac{1}{2},k}}{\partial U_{j-1,k}}\Delta U^{(1)}_{j-1,k} \qquad (7.20)$$

The first of these steps requires the solution of a 4x4 block-tridiagonal system of equations for each k, and the second step requires a block-tridiagonal solution for each j. The combined work is proportional to JK.

Although it is not immediately obvious, this procedure is exactly equivalent to Beam and Warming's factored ADI procedure. Also, if both steps are repeated, always using the latest known ΔU on the right-hand-side, the procedure will converge to the solution of the 'unfactored linearized-delta' equations, and is exactly equivalent to Rai's ADI method with sub-iteration.

7.3 Flux difference upwinding

A Fourier stability analysis of the above algorithm would conclude that it is unconditionally stable, but in fact it has two major problems. The first is that both the unfactored and the block-triagonal systems become extremely ill-conditioned when the Reynolds number is large and the timestep is large. This is because the off-diagonal matrices are independent of Δt, whereas the inviscid flux terms in the diagonal matrix almost perfectly cancel, leaving only terms proportional to $1/Re$ or $1/\Delta t$. The second problem is that there is no numerical smoothing to suppress the emergence of the 'sawtooth' error mode which is the standard problem of central differencing algorithms.

Both of these problems become clear when one considers the scalar one-dimensional convection problem.

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \qquad (7.21)$$

Using central spatial differencing and backward time differencing gives

$$u_j^{n+1} - u_j^n + \frac{c\Delta t}{2\Delta x}(u_{j+1}^{n+1} - u_{j-1}^{n+1}) = 0 \tag{7.22}$$

which can be rearranged as

$$\frac{r}{2}\Delta u_{j+1} + \Delta u_j - \frac{r}{2}\Delta u_{j-1} = -\frac{r}{2}(u_{j+1}^n - u_{j-1}^n) \tag{7.23}$$

The right-hand-side shows that $u_j = (-1)^j$ is a valid steady-state solution. The left-hand-side shows that when $r = c\Delta t/\Delta x$ becomes large the diagonal matrix term is smaller than the off-diagonal terms, leading to an ill-conditioned matrix.

The simplest solution to these problems is to use upwinding, and for $c > 0$ approximate the spatial derivative by a discrete finite difference using nodes j and $j-1$. This gives

$$u_j^{n+1} - u_j^n + \frac{c\Delta t}{\Delta x}(u_j^{n+1} - u_{j-1}^{n+1}) = 0 \tag{7.24}$$

which can be rearranged as

$$(1+r)\Delta u_j - r\Delta u_{j-1} = -r(u_j^n - u_{j-1}^n). \tag{7.25}$$

This is well-conditioned and does not allow a sawtooth error mode.

Next consider a linear hyperbolic system,

$$\frac{\partial U}{\partial t} + A\frac{\partial U}{\partial x} = 0 \tag{7.26}$$

This equation can be diagonalized by use of a matrix T whose columns are eigenvectors of A. If the eigenvalues of A are λ_i, ordered consistently with the vectors in T, then

$$T^{-1}AT = \Lambda \equiv \mathrm{diag}(\lambda_1, \lambda_2, ...) \tag{7.27}$$

and so if one defines $U = TV$ then

$$\frac{\partial V}{\partial t} + \Lambda\frac{\partial V}{\partial x} = 0. \tag{7.28}$$

Since the characteristic equation are uncoupled, each can be upwinded separately. For the i^{th} characteristic, the upwinded equation can be written as

$$v_{ij}^{n+1} - v_{ij}^n + \frac{\lambda_i^+\Delta t}{\Delta x}(v_{ij}^{n+1} - v_{ij-1}^{n+1}) + \frac{\lambda_i^-\Delta t}{\Delta x}(v_{ij+1}^{n+1} - v_{ij}^{n+1}) = 0, \tag{7.29}$$

where

$$\lambda_i^+ = \max(0,\, \lambda_i), \quad \lambda_i^- = \min(0,\, \lambda_i) \tag{7.30}$$

Combining all of the characteristics into one equation gives

$$V_j^{n+1} - V_j^n + \frac{\Delta t}{\Delta x}\Lambda^+(V_j^{n+1} - V_{j-1}^{n+1}) + \frac{\Delta t}{\Delta x}\Lambda^-(V_{j+1}^{n+1} - V_j^{n+1}) = 0, \tag{7.31}$$

where

$$\Lambda_i^+ = \mathrm{diag}(\lambda_i^+), \quad \Lambda_i^- = \mathrm{diag}(\lambda_i^-) \tag{7.32}$$

Converting back to original variables gives

$$U_j^{n+1} - U_j^n + \frac{\Delta t}{\Delta x}A^+(U_j^{n+1} - U_{j-1}^{n+1}) + \frac{\Delta t}{\Delta x}A^-(U_{j+1}^{n+1} - U_j^{n+1}) = 0, \tag{7.33}$$

where

$$A^\pm = T\Lambda^\pm T^{-1} \tag{7.34}$$

In 'delta-form' this can also be written as

$$\begin{aligned}
&\frac{\Delta t}{\Delta x}A^-\Delta U_{j+1} + \left(I + \frac{\Delta t}{\Delta x}(A^+ - A^-)\right)\Delta U_j - \frac{\Delta t}{\Delta x}A^+\Delta U_{j-1} \\
&= -\frac{\Delta t}{\Delta x}A^+(U_j^n - U_{j-1}^n) - \frac{\Delta t}{\Delta x}A^-(U_{j+1}^n - U_j^n) \qquad (7.35) \\
&= -\frac{\Delta t}{\Delta x}\left(\tfrac{1}{2}A(U_{j+1}^n + U_j^n) - \tfrac{1}{2}A(U_j^n + U_{j-1}^n)\right) + \frac{\Delta t}{2\Delta x}(A^+ - A^-)(U_{j+1}^n - 2U_j^n + U_{j-1}^n))
\end{aligned}$$

This last form has a right-hand-side which is expressed as a central difference term plus a smoothing term. This is important for the next complication, moving to a nonlinear system of equations,

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{7.36}$$

for which the upwinded implicit scheme is

$$\begin{aligned}
&\frac{\Delta t}{\Delta x}A_{j+\frac{1}{2}}^-\Delta U_{j+1} + \left(I + \frac{\Delta t}{\Delta x}(A_{j+\frac{1}{2}}^+ - A_{j-\frac{1}{2}}^-)\right)\Delta U_j - \frac{\Delta t}{\Delta x}A_{j-\frac{1}{2}}^+\Delta U_{j-1} \\
&= -\frac{\Delta t}{\Delta x}\left(\tfrac{1}{2}(F_{j+1}^n + F_j^n) - \tfrac{1}{2}(F_j^n + F_{j-1}^n)\right) \\
&\quad + \frac{\Delta t}{\Delta x}\left(\tfrac{1}{2}(A^+ - A^-)_{j+\frac{1}{2}}(U_{j+1}^n - U_j^n) - \tfrac{1}{2}(A^+ - A^-)_{j-\frac{1}{2}}(U_j^n - U_{j-1}^n)\right) \\
&= -\frac{\Delta t}{\Delta x}\Big\{\left(\tfrac{1}{2}(F_{j+1}^n + F_j^n) - \tfrac{1}{2}(A^+ - A^-)_{j+\frac{1}{2}}(U_{j+1}^n - U_j^n)\right) \qquad (7.37) \\
&\qquad - \left(\tfrac{1}{2}(F_j^n + F_{j-1}^n) - \tfrac{1}{2}(A^+ - A^-)_{j-\frac{1}{2}}(U_j^n - U_{j-1}^n)\right)\Big\}
\end{aligned}$$

with

$$A_{j+\frac{1}{2}} = \left(\frac{\partial F}{\partial U}\right)_{j+\frac{1}{2}} \tag{7.38}$$

being evaluated using a value of U which is some average of U_j and U_{j+1}. Note that the nonlinear upwinded equation is written in a form which is clearly conservative, since the right-hand-side is equal to the difference of two fluxes through faces $j+\frac{1}{2}$ and $j-\frac{1}{2}$. Before finishing with these model problems, a final note is that the upwinded schemes presented so far have only first order spatial accuracy. Third order spatial accuracy is achieved with the following scheme.

$$\frac{\Delta t}{\Delta x} A^-_{j+\frac{1}{2}} \Delta U_{j+1} + \left(I + \frac{\Delta t}{\Delta x} (A^+_{j+\frac{1}{2}} - A^-_{j-\frac{1}{2}}) \right) \Delta U_j - \frac{\Delta t}{\Delta x} A^+_{j-\frac{1}{2}} \Delta U_{j-1} = -\frac{\Delta t}{\Delta x} (\overline{F}^n_{j+\frac{1}{2}} - \overline{F}^n_{j-\frac{1}{2}}) \tag{7.39}$$

where

$$\overline{F}^n_{j+\frac{1}{2}} = \tfrac{1}{2}(F^n_{j+1} + F^n_j) - \tfrac{1}{6} A^+_{j+\frac{1}{2}} (U^n_{j+1} - 2U^n_j + U^n_{j-1}) - \tfrac{1}{6} A^-_{j+\frac{1}{2}} (U^n_{j+2} - 2U^n_{j+1} + U^n_j) \tag{7.40}$$

Returning now to the Navier-Stokes equations, the final form of the upwinded, unfactored, discrete Navier-Stokes equations is as follows.

$$\begin{aligned}
&\left(\frac{A'}{\Delta t} + \left(\frac{\partial F^*}{\partial U}\right)^+_{j+\frac{1}{2},k} - \left(\frac{\partial F^*}{\partial U}\right)^-_{j-\frac{1}{2},k} + \left(\frac{\partial G^*}{\partial U}\right)^+_{j,k+\frac{1}{2}} - \left(\frac{\partial G^*}{\partial U}\right)^-_{j,k-\frac{1}{2}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k}} + \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k}} \right) \Delta U_{j,k} \\
&+ \left(\frac{\partial F^*}{\partial U}\right)^-_{j+\frac{1}{2},k} \Delta U_{j+1,k} - \left(\frac{\partial F^*}{\partial U}\right)^+_{j-\frac{1}{2},k} \Delta U_{j-1,k} \qquad\qquad (7.41) \\
&+ \left(\left(\frac{\partial G^*}{\partial U}\right)^-_{j,k+\frac{1}{2}} - \frac{\partial V^*_{j,k+\frac{1}{2}}}{\partial U_{j,k+1}} \right) \Delta U_{j,k+1} - \left(\left(\frac{\partial G^*}{\partial U}\right)^+_{j,k-\frac{1}{2}} - \frac{\partial V^*_{j,k-\frac{1}{2}}}{\partial U_{j,k-1}} \right) \Delta U_{j,k-1} \\
&= \; -(\overline{F}^{*^n}_{j+\frac{1}{2},k} - \overline{F}^{*^n}_{j-\frac{1}{2},k}) - (\overline{G}^{*^n}_{j,k+\frac{1}{2}} - \overline{G}^{*^n}_{j,k-\frac{1}{2}}) + (V^{*^n}_{j,k+\frac{1}{2}} - V^{*^n}_{j,k-\frac{1}{2}}) + S^{*^n}_{j,k}
\end{aligned}$$

The upwinding matrices $\left(\frac{\partial F^*}{\partial U}\right)^{\pm}$ and $\left(\frac{\partial G^*}{\partial U}\right)^{\pm}$ are both constructed in the same way. Both F^* and G^* are inviscid fluxes of the form $F\Delta y' - G\Delta x'$. The corresponding linearization matrix is

$$\begin{aligned}
A &= \frac{\partial(F\Delta y' - G\Delta x')}{\partial U} \\
&= \Delta s' \begin{pmatrix} 0 & n_x & n_y & 0 \\ -u_n u + \frac{\gamma-1}{2}(u^2+v^2)n_x & u_n - (\gamma-2)un_x & un_y - (\gamma-1)vn_x & (\gamma-1)n_x \\ -u_n v + \frac{\gamma-1}{2}(u^2+v^2)n_y & vn_x - (\gamma-1)un_y & u_n - (\gamma-2)vn_y & (\gamma-1)n_y \\ -u_n H + \frac{\gamma-1}{2}(u^2+v^2)u_n & Hn_x - (\gamma-1)uu_n & Hn_y - (\gamma-1)vu_n & \gamma u_n \end{pmatrix},
\end{aligned} \tag{7.42}$$

where the unit normal vector is

$$\begin{pmatrix} n_x \\ n_y \end{pmatrix} = \frac{1}{\Delta s'} \begin{pmatrix} \Delta y' \\ -\Delta x' \end{pmatrix} \tag{7.43}$$

and the normal and tangential velocities are

$$\begin{aligned} u_n &= un_x + vn_y \\ u_t &= -un_y + vn_x \end{aligned} \tag{7.44}$$

The transformation matrix T which is used to diagonalize A is

$$T = \begin{pmatrix} 1 & 1 & 0 & 1 \\ u - cn_x & u & -cn_y & u + cn_x \\ v - cn_y & v & cn_x & v + cn_y \\ H - u_n c & \frac{1}{2}(u^2 + v^2) & u_t c & H + u_n c \end{pmatrix}, \tag{7.45}$$

and its inverse is

$$T^{-1} = \frac{1}{c^2} \begin{pmatrix} \frac{1}{2}u_n c + \frac{\gamma-1}{4}(u^2+v^2) & -\frac{1}{2}cn_x - \frac{\gamma-1}{2}u & -\frac{1}{2}cn_y - \frac{\gamma-1}{2}v & \frac{\gamma-1}{2} \\ c^2 - \frac{\gamma-1}{2}(u^2+v^2) & (\gamma-1)u & (\gamma-1)v & -(\gamma-1) \\ -u_t c & -cn_y & cn_x & 0 \\ -\frac{1}{2}u_n c + \frac{\gamma-1}{4}(u^2+v^2) & \frac{1}{2}cn_x - \frac{\gamma-1}{2}u & \frac{1}{2}cn_y - \frac{\gamma-1}{2}v & \frac{\gamma-1}{2} \end{pmatrix}. \tag{7.46}$$

The corresponding eigenvalues are

$$\begin{aligned} \lambda_1 &= \Delta s'(u_n - c) \\ \lambda_2 &= \Delta s' u_n \\ \lambda_3 &= \Delta s' u_n \\ \lambda_4 &= \Delta s'(u_n + c) \end{aligned} \tag{7.47}$$

As discussed earlier, A^+ and A^- are then defined by

$$A^{\pm} = T\Lambda^{\pm}T^{-1} \tag{7.48}$$

The upwinded matrices are evaluated based on a special average of U^- and U^+, the state vectors on either side of the face. The average is defined as

$$\begin{aligned} u &= \frac{\sqrt{\rho^+}u^+ + \sqrt{\rho^-}u^-}{\sqrt{\rho^+} + \sqrt{\rho^-}} \\ v &= \frac{\sqrt{\rho^+}v^+ + \sqrt{\rho^-}v^-}{\sqrt{\rho^+} + \sqrt{\rho^-}} \\ H &= \frac{\sqrt{\rho^+}H^+ + \sqrt{\rho^-}H^-}{\sqrt{\rho^+} + \sqrt{\rho^-}}. \end{aligned} \tag{7.49}$$

All other needed quantities can be derived from these three. This form of average was developed by Roe, and reasons for its use are presented in Ref. [32].

Another small but important addition to the basic upwinding ideas is the upwinding treatment in areas in which one of the characteristic speeds passes through zero. An improper treatment can lead to the formation of non-physical expansion shocks and problems with stability. The treatment that is used in UNSFLO is based upon the ideas of van Leer [35], and involves modifying the definition of $\lambda^{\pm}$.

$$\begin{aligned}\lambda^{+} &= \max(0,\lambda) + 0.5\max(0,\ 2\Delta\lambda - |\lambda|)\\ \lambda^{-} &= \min(0,\lambda) - 0.5\max(0,\ 2\Delta\lambda - |\lambda|)\end{aligned} \tag{7.50}$$

In these definitions, λ is the eigenvalue, evaluated using the Roe-averaged state, and $\Delta\lambda$ is the difference between the corresponding eigenvalues evaluated at states U^{+} and U^{-}. Note four things about this formulation. First, in most of the flow field, away from points where λ goes through zero, this returns to the standard definition. Second, λ^{+} is always positive (or zero), and λ^{-} is always negative (or zero); this maintains the upwind/dissipative character. Third, $\lambda^{+}+\lambda^{-} = \lambda$; this maintains the consistency of the discretization. Fourth, when $\lambda = 0$, $\lambda^{\pm} = \pm\Delta\lambda$; the effect of this is to maintain a certain minimum level of numerical smoothing which prevents non-physical behavior but without unnecessary corruption of the physical solution.

The fluxes on the right-hand-side of Eq. (7.41) are defined as

$$\begin{aligned}\overline{F}^{*}_{j+\frac{1}{2},k} &= \tfrac{1}{2}(F^{n}_{j,k}+F^{n}_{j+1,k})\Delta y'_{j+\frac{1}{2},k} - \tfrac{1}{2}(G^{n}_{j,k}+G^{n}_{j+1,k})\Delta x'_{j+\frac{1}{2},k}\\ &\quad -\frac{\sigma_{2v}}{2}\left(\left(\frac{\partial F^{*}}{\partial U}\right)^{+}_{j+\frac{1}{2},k} - \left(\frac{\partial F^{*}}{\partial U}\right)^{-}_{j+\frac{1}{2},k}\right)(U^{n}_{j+1,k}-U^{n}_{j,k})\\ &\quad -\frac{1-\sigma_{2v}}{6}\left(\frac{\partial F^{*}}{\partial U}\right)^{+}_{j+\frac{1}{2},k}(U^{n}_{j+1,k} - 2U^{n}_{j,k} + U^{n}_{j-1,k})\\ &\quad -\frac{1-\sigma_{2v}}{6}\left(\frac{\partial F^{*}}{\partial U}\right)^{-}_{j+\frac{1}{2},k}(U^{n}_{j+2,k} - 2U^{n}_{j+1,k} + U^{n}_{j,k})\end{aligned} \tag{7.51}$$

and

$$\begin{aligned}\overline{G}^{*}_{j,k+\frac{1}{2}} &= \tfrac{1}{2}(F^{n}_{j,k}+F^{n}_{j,k+1})\Delta y'_{j,k+\frac{1}{2}} - \tfrac{1}{2}(G^{n}_{j,k}+G^{n}_{j,k+1})\Delta x'_{j,k+\frac{1}{2}}\\ &\quad -\frac{\sigma_{2v}}{2}\left(\left(\frac{\partial G^{*}}{\partial U}\right)^{+}_{j,k+\frac{1}{2}} - \left(\frac{\partial G^{*}}{\partial U}\right)^{-}_{j,k+\frac{1}{2}}\right)(U^{n}_{j,k+1}-U^{n}_{j,k})\\ &\quad -\frac{1-\sigma_{2v}}{6}\left(\frac{\partial G^{*}}{\partial U}\right)^{+}_{j,k+\frac{1}{2}}(U^{n}_{j,k+1} - 2U^{n}_{j,k} + U^{n}_{j,k-1})\\ &\quad -\frac{1-\sigma_{2v}}{6}\left(\frac{\partial G^{*}}{\partial U}\right)^{-}_{j,k+\frac{1}{2}}(U^{n}_{j,k+2} - 2U^{n}_{j,k+1} + U^{n}_{j,k})\end{aligned} \tag{7.52}$$

When $\sigma_{2v} = 1$ these give first order upwinded fluxes; when $\sigma_{2v} = 0$ they give third order fluxes.

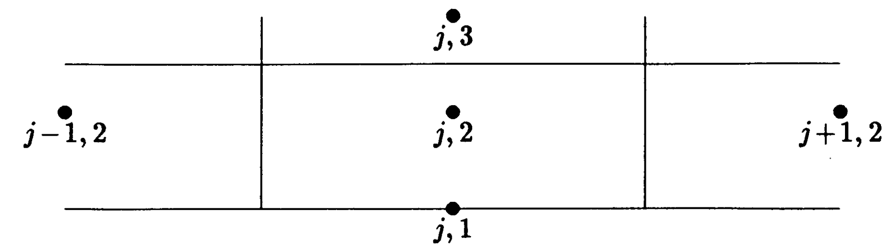

Figure 7.2: Wall boundary cell

7.4 Wall boundary conditions

The computational cell next to a wall boundary is different from the standard viscous computational cells, in that it extends from $k=5/2$, to $k=1$, with $U_{j,2}$ being the average value of U in the cell. $U_{j,1}$ is used for only two purposes, computing the third order upwinding term for the face at $k=5/2$, and post-processing. $U_{j,2}$ is the first point that is actually calculated as part of the implicit viscous scheme, using the wall boundary cell.

Because the wall boundary cell is different, the first change to the basic viscous algorithm is in the definitions of $\Delta x'_{j+\frac{1}{2},2}$ and $\Delta y'_{j+\frac{1}{2},2}$.

$$\begin{aligned}\Delta x'_{j+\frac{1}{2},2} &= \frac{h_{j,2}+h_{j+1,2}}{2}(x_{j+\frac{1}{2},2\frac{1}{2}} - x_{j+\frac{1}{2},1\frac{1}{2}}) + \frac{h_{j,1}+h_{j+1,1}}{2}(x_{j+\frac{1}{2},1\frac{1}{2}} - x_{j+\frac{1}{2},1}) \\ \Delta y'_{j+\frac{1}{2},2} &= \frac{h_{j,2}+h_{j+1,2}}{2}(y_{j+\frac{1}{2},2\frac{1}{2}} - y_{j+\frac{1}{2},1\frac{1}{2}}) + \frac{h_{j,1}+h_{j+1,1}}{2}(y_{j+\frac{1}{2},1\frac{1}{2}} - y_{j+\frac{1}{2},1})\end{aligned} \tag{7.53}$$

Apart from these changes, $\overline{F}^*_{j+\frac{1}{2},2}$ and $\left(\frac{\partial F^*}{\partial U}\right)^{\pm}_{j+\frac{1}{2},2}$ are evaluated as normal.

The other changes for the wall boundary cell are all related to the wall boundary face. The inviscid flux $\overline{G}^*_{j,1}$ through the wall face has only a pressure force term.

$$\overline{G}^*_{j,1} = \begin{pmatrix} 0 \\ p_{j,2}\Delta y'_{j,1} \\ -p_{j,2}\Delta x'_{j,1} \\ 0 \end{pmatrix}, \tag{7.54}$$

where

$$\begin{aligned}\Delta x'_{j,1} &= h_{j,1}(x_{j-\frac{1}{2},1} - x_{j+\frac{1}{2},1}) \\ \Delta y'_{j,1} &= h_{j,1}(y_{j-\frac{1}{2},1} - y_{j+\frac{1}{2},1}).\end{aligned} \tag{7.55}$$

This flux is linearized directly to obtain the matrix to be used in the implicit operator; no flux-difference upwinding is performed on the wall face.

For adiabatic walls, the viscous flux term $V^*_{j,1}$ is

$$V^*_{j,1} \;=\; \mu_{j,1}\frac{\Delta s'}{\Delta n}\begin{pmatrix} 0 \\ u_{j,2} \\ v_{j,2} \\ \frac{1}{2}u^2_{j,2}+\frac{1}{2}v^2_{j,2} \end{pmatrix}. \tag{7.56}$$

For walls with a specified temperature (and a corresponding speed of sound c_{wall}), the viscous flux term is

$$V^*_{j,1} \;=\; \mu_{j,1}\frac{\Delta s'}{\Delta n}\begin{pmatrix} 0 \\ u_{j,2} \\ v_{j,2} \\ (\frac{1}{2}u^2_{j,2}+\frac{1}{2}v^2_{j,2}+\frac{1}{(\gamma-1)Pr}c^2_{j,2}) - \frac{1}{(\gamma-1)Pr}c^2_{wall} \end{pmatrix}. \tag{7.57}$$

In both cases,

$$\frac{\Delta s'}{\Delta n} = \frac{(\Delta x'_{j,1})^2 + (\Delta y'_{j,1})^2}{\Delta x'_{j,1}(y_{j,1}-y_{j,2}) - \Delta y'_{j,1}(x_{j,1}-x_{j,2})} \tag{7.58}$$

$V^*_{j,1}$ is linearized as usual to obtain the viscous implicit matrix term.

The final form of the implicit unfactored equation for the wall boundary cell is

$$\begin{aligned} &\left(\frac{A'}{\Delta t}+\left(\frac{\partial F^*}{\partial U}\right)^+_{j+\frac{1}{2},2}-\left(\frac{\partial F^*}{\partial U}\right)^-_{j-\frac{1}{2},2}+\left(\frac{\partial G^*}{\partial U}\right)^+_{j,2\frac{1}{2}}-\left(\frac{\partial \overline{G}^*}{\partial U}\right)_{j,1}-\frac{\partial V^*_{j,2\frac{1}{2}}}{\partial U_{j,2}}+\frac{\partial V^*_{j,1}}{\partial U_{j,2}}\right)\Delta U_{j,2} \\ &+\left(\frac{\partial F^*}{\partial U}\right)^-_{j+\frac{1}{2},2}\Delta U_{j+1,2}-\left(\frac{\partial F^*}{\partial U}\right)^+_{j-\frac{1}{2},2}\Delta U_{j-1,2} \\ &+\left(\left(\frac{\partial G^*}{\partial U}\right)^-_{j,2\frac{1}{2}}-\frac{\partial V^*_{j,2\frac{1}{2}}}{\partial U_{j,3}}\right)\Delta U_{j,3} \\ &= \; -(\overline{F}^{*^n}_{j+\frac{1}{2},2}-\overline{F}^{*^n}_{j-\frac{1}{2},2})-(\overline{G}^{*^n}_{j,2\frac{1}{2}}-\overline{G}^{*^n}_{j,1})+(V^{*^n}_{j,2\frac{1}{2}}-V^{*^n}_{j,1})+S^{*^n}_{j,2} \end{aligned} \tag{7.59}$$

7.5 Inviscid interface treatment

The computational cell next to the viscous/inviscid grid interface is also different from the standard viscous computational cells, in that it extends from $k = K - \frac{1}{2}$ to $k = K$. $U_{j,K}$ is the average value of U in the cell, and is also (by definition) equal to the value of U at the inviscid node on the interface.

The first change to the basic viscous algorithm is in the definitions of $\Delta x'_{j+\frac{1}{2},K}$ and $\Delta y'_{j+\frac{1}{2},K}$.

$$\Delta x'_{j+\frac{1}{2},K} \;=\; \frac{h_{j,K}+h_{j+1,K}}{2}(x_{j+\frac{1}{2},K} - x_{j+\frac{1}{2},K-\frac{1}{2}})$$

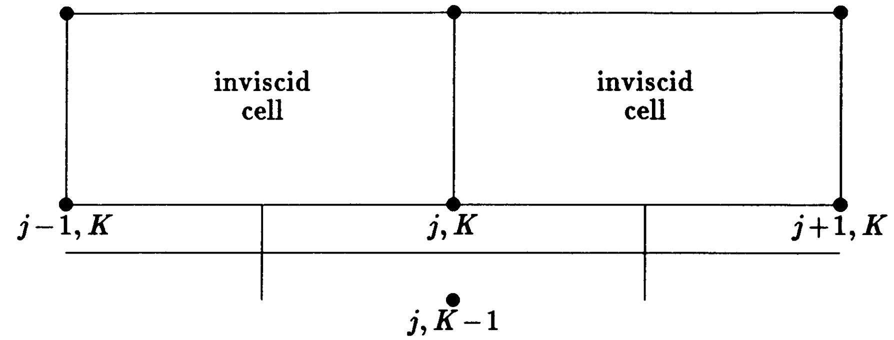

Figure 7.3: Interface boundary viscous cell plus inviscid cells

$$\Delta y'_{j+\frac{1}{2},K} \;=\; \frac{h_{j,K}+h_{j+1,K}}{2}(y_{j+\frac{1}{2},K} - y_{j+\frac{1}{2},K-\frac{1}{2}}) \tag{7.60}$$

Apart from these changes, $\overline{F}^{*}_{j+\frac{1}{2},K}$ and $\left(\frac{\partial F^*}{\partial U}\right)^{\pm}_{j+\frac{1}{2},K}$ are evaluated as normal.

The second change is in the flux through the interface. To make the coupling conservative, the viscous flux is set to zero, and the inviscid flux is treated explicitly, in a manner that is consistent with the flux evaluation used in the inviscid Lax-Wendroff calculation on the outer inviscid cells.

$$\overline{G}^{*}_{j,K} = F_{j,K}\Delta y'_{j,K} - G_{j,K}\Delta x'_{j,K} \tag{7.61}$$

where

$$\begin{aligned}\Delta x'_{j,K} &= h_{j,K}(x_{j-\frac{1}{2},K} - x_{j+\frac{1}{2},K})\\ \Delta y'_{j,K} &= h_{j,K}(y_{j-\frac{1}{2},K} - y_{j+\frac{1}{2},K}).\end{aligned} \tag{7.62}$$

The third change is that since $U_{j,K}$ is associated with both the viscous node/cell, and the inviscid node at the same location, the cell volume $A'_{j,K}$ must be equal to the sum of the viscous cell volume and the inviscid node's share of the volume of the two neighboring inviscid cells, and the inviscid changes which have previously been calculated and distributed to the inviscid node must be added in to the viscous calculation as an additional explicit contribution. Note that the combined volume A' must also be the volume used for the inviscid calculation.

The final form of the implicit unfactored equation for the interface cell is

$$\left(\frac{A'}{\Delta t} + \left(\frac{\partial F^*}{\partial U}\right)^{+}_{j+\frac{1}{2},K} - \left(\frac{\partial F^*}{\partial U}\right)^{-}_{j-\frac{1}{2},K} - \left(\frac{\partial G^*}{\partial U}\right)^{-}_{j,K-\frac{1}{2}} + \frac{\partial V^*_{j,K-\frac{1}{2}}}{\partial U_{j,K}}\right)\Delta U_{j,K}$$
$$+ \left(\frac{\partial F^*}{\partial U}\right)^{-}_{j+\frac{1}{2},K}\Delta U_{j+1,K} - \left(\frac{\partial F^*}{\partial U}\right)^{+}_{j-\frac{1}{2},K}\Delta U_{j-1,K}$$

$$
\begin{aligned}
&- \left(\left(\frac{\partial G^*}{\partial U} \right)^+_{j,K-\frac{1}{2}} - \frac{\partial V^*_{j,K-\frac{1}{2}}}{\partial U_{j,K-1}} \right) \Delta U_{j,K-1} \qquad (7.63) \\
&= -(\overline{F}^{*^n}_{j+\frac{1}{2},K} - \overline{F}^{*^n}_{j-\frac{1}{2},K}) - (\overline{G}^{*^n}_{j,K} - \overline{G}^{*^n}_{j,K-\frac{1}{2}}) - V^{*^n}_{j,K-\frac{1}{2}} + S^{*^n}_{j,K} \\
&\quad + \sum_{inviscid} \frac{A'\,\delta U}{\Delta t}
\end{aligned}
$$

7.6 Algebraic turbulence model

The algebraic turbulence model which is used is due to Cebeci and Smith [3]. The formulation splits the boundary layer into inner and outer regions. The inner region is a combination of the log-law layer and the laminar sublayer; the Prandtl mixing length is taken to the normal distance to the wall surface multiplied by the von Karman constant, with the van Driest exponential damping term to give the correct behavior in the sub-layer.

$$(\mu_t)_{inner} = \rho l^2 \left| \frac{\partial u}{\partial y} \right| \qquad (7.64)$$

$$l = 0.4\, y\, (1 - e^{-y/A}) \qquad (7.65)$$

$$A = \frac{26\mu}{\rho u_\tau} \qquad (7.66)$$

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \qquad (7.67)$$

The outer region formulation is based on the velocity defect.

$$(\mu_t)_{outer} = 0.0168\, \rho \int_o^\delta |u_e - u| dy \qquad (7.68)$$

The transition between the inner and outer formulations occurs at the value of y (the coordinate normal to the wall) for which the two are equal.

A tricky numerical aspect is the definition of δ, used for the velocity defect integration and to define the edge velocity value u_e. After some experimentation the definition which is used is

$$\delta = \frac{5}{2} \frac{\int_0^\infty y(|\frac{\partial u}{\partial y}| + |\frac{\partial T}{\partial y}|)\, dy}{\int_0^\infty (|\frac{\partial u}{\partial y}| + |\frac{\partial T}{\partial y}|)\, dy}, \qquad (7.69)$$

where the integrals are performed across the entire boundary layer grid. The effect of this definition is to get a value for δ which is at the edge of the physical boundary layer, but is still typically much less than the thickness of the viscous grid. Using the edge of the viscous grid instead for δ gives much poorer values for u_e because of variations in u outside the boundary layer due to flow curvature.

To avoid the need for very small grid spacing at the wall, a law-of-the-wall formulation is used to calculate the wall shear stress based on the velocity at the first grid point off the wall. Spaulding's law-of-the-wall formula [33] is

$$y^+ = u^+ + e^{-\kappa B}\left(e^{\kappa u^+} - 1 - \kappa u^+ - \tfrac{1}{2}(\kappa u^+)^2 - \tfrac{1}{6}(\kappa u^+)^3\right) \tag{7.70}$$

where

$$y^+ = \frac{y u_\tau \rho_w}{\mu_w} \tag{7.71}$$

$$u^+ = \frac{u}{u_\tau} \tag{7.72}$$

and the constants κ and B have values 0.4 and 5.5 respectively.

Two limits are worth noting. If τ_w is very small, then y^+ and u^+ are small, and so

$$u^+ \approx y^+ \quad\Longrightarrow\quad \tau_w \approx \frac{\mu_w u}{y}, \tag{7.73}$$

which is the laminar limit. If τ_w is very large, then y^+ and u^+ are large, and so

$$y^+ \approx e^{-\kappa B} e^{\kappa u^+} \quad\Longrightarrow\quad u^+ \approx B + \tfrac{1}{\kappa}\log y^+, \tag{7.74}$$

which is the log-law limit. Thus the full Spaulding formula combines both behaviors and will remain valid even through separation.

If y_2 and u_2 are the normal coordinate and tangential velocity at the first grid point off the surface, and Re_2 is defined as

$$Re_2 = \frac{\rho_w u_2 y_2}{\mu_w}, \tag{7.75}$$

then the definition for y^+ can be re-expressed as

$$y^+ = \frac{Re_2}{u^+}. \tag{7.76}$$

Substituting this into Spaulding's formula gives

$$u^+ + e^{-\kappa B}\left(e^{\kappa u^+} - 1 - \kappa u^+ - \tfrac{1}{2}(\kappa u^+)^2 - \tfrac{1}{6}(\kappa u^+)^3\right) - \frac{Re_2}{u^+} = 0. \tag{7.77}$$

For a given value of Re_2, this equation is solved using a Newton-Raphson procedure to obtain u^+ and hence τ_w.

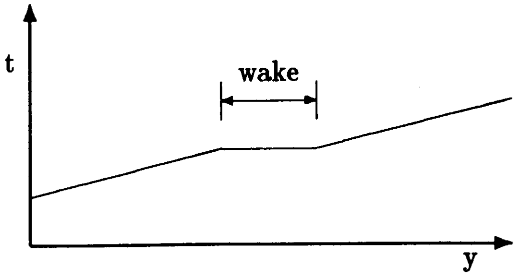

Figure 7.4: Alternative inclined computational plane for viscous calculations

7.7 Time tilting

The technique of time-tilting can also be applied to the Navier-Stokes equations. The transformed equations are

$$\frac{\partial Q}{\partial t'} + \frac{\partial F}{\partial x'} + \frac{\partial G}{\partial y'} - \frac{\partial V_x}{\partial x'} - \frac{\partial V_y}{\partial y'} = 0 \tag{7.78}$$

where

$$Q = U - \lambda G + \lambda V_y \tag{7.79}$$

The difference from the inviscid time-inclined equations is that now Q contains derivatives of U and there is no simple algebraic transformation from Q to U.

The solution to this problem is to simply ignore the derivative terms in Q, or equivalently to replace Q by $U - \lambda G$. This procedure can be justified for large Reynolds numbers through the following argument. Time derivatives are comparable in magnitude to streamwise spatial derivatives, and so

$$\frac{\partial^2 u}{\partial t' \partial y'} = O\left(\frac{\partial^2 u}{\partial x' \partial y'}\right) = O\left(\mathrm{Re}^{-\frac{1}{2}}\right) \times \frac{\partial^2 u}{\partial y'^2} \tag{7.80}$$

Therefore, the neglected terms are comparable in magnitude to the terms which are dropped in the usual thin-shear-layer N-S equations, due to their being much smaller than the dominant diffusive terms in the boundary layer.

An alternative approach which would also work for high Reynolds numbers is to use "time-tilting" only outside the narrow viscous regions around each blade, and in each wake. This idea is illustrated in Fig. 7.4. The high-Reynolds number assumption is again required to ensure that the wakes and boundary layers form a small fraction of the total domain.

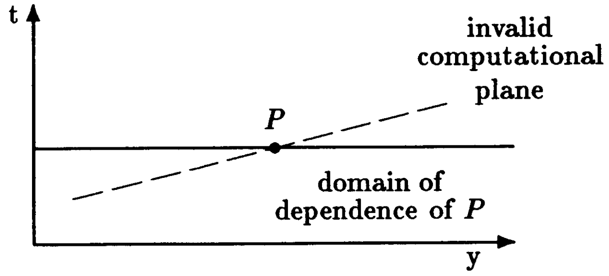

Figure 7.5: Low Reynolds number domain of dependence

Both of the above approaches fail at low Reynolds numbers. This is correct and proper since, as shown in Fig. 7.5, parabolic equations have infinite speed of propagation of information and any inclined computational plane will not fully include this domain of dependency, and so cannot produce the correct solution.

In the discrete Navier-Stokes equations used in UNSFLO, the only change due to time-tilting is to replace the term

$$\frac{A'}{\Delta t}$$

in the implicit operator, by

$$\frac{A'}{\Delta t}(I - \lambda\frac{\partial G}{\partial U})$$

thereby approximating $\frac{\partial Q}{\partial t}$ instead of $\frac{\partial U}{\partial t}$.

7.8 Moving blades

One capability of UNSFLO is to perform viscous calculations for moving blades, in which the blade motion is prescribed usually as a combination of bending and torsion. The viscous computational grid for this problem moves accordingly, with the wall nodes moving with the blade, and the interface nodes remaining stationary. This approach allows the inviscid solver to remain unchanged, with moving grid modifications being necessary only for the viscous solver.

Consider the two-dimensional Navier-Stokes equations being solved in a control volume Ω whose boundary is moving at velocity $\vec{u}_b$ and has an outward pointing unit normal vector $\vec{n}$. Because of the motion of the boundary $\partial\Omega$,

$$\frac{d}{dt}\iint_{\Omega} U\,dx\,dy = \iint_{\Omega}\frac{\partial U}{\partial t}\,dx\,dy + \int_{\partial\Omega} U\,\vec{u}_b\cdot\vec{n}\,ds \qquad (7.81)$$

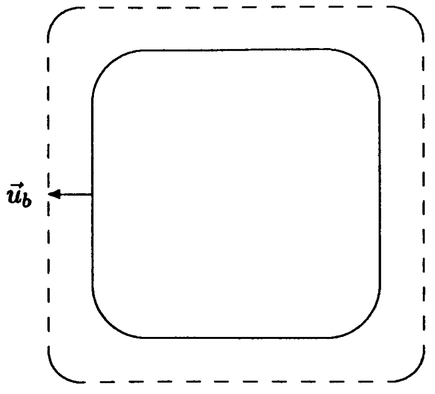

Figure 7.6: Moving control volume

As shown in Fig. 7.6, the second term corresponds to the volume being swept out by the moving boundary. Hence,

$$\begin{aligned}\frac{d}{dt}\iint_{\Omega} U\,dx\,dy &= -\iint_{\Omega} \frac{\partial(F-V_x)}{\partial x}+\frac{\partial(G-V_y)}{\partial y}\,dx\,dy+\int_{\partial\Omega} U\,\vec{u}_b\cdot\vec{n}\,ds \\ &= -\int_{\partial\Omega}(F-V_x)\,n_x+(G-V_y)\,n_y-U\,\vec{u}_b\cdot\vec{n}\,ds. \end{aligned}\tag{7.82}$$

Next, define $\overline{U}$ to be the average value of U in the control volume. Therefore,

$$\frac{d}{dt}\iint_{\Omega} U\,dx\,dy \equiv \frac{d}{dt}(A\overline{U}) = A\frac{d\overline{U}}{dt}+\overline{U}\frac{dA}{dt} = A\frac{d\overline{U}}{dt}+\overline{U}\int_{\partial\Omega}\vec{u}_b\cdot\vec{n}\,ds \tag{7.83}$$

Combining with Eq. (7.82), gives

$$A\frac{d\overline{U}}{dt} = -\int_{\partial\Omega}(F-V_x)\,n_x+(G-V_y)\,n_y-(U-\overline{U})\,\vec{u}_b\cdot\vec{n}\,ds. \tag{7.84}$$

The corresponding equation for the quasi-3D Navier-Stokes equations on a moving control volume is very similar. In discretizing this, the viscous algorithm on standard viscous cells has two changes. On the explicit right-hand-side the following terms are added.

$$\begin{aligned} &+\ \left(u_{g_{j+\frac{1}{2},k}}\Delta y'_{j+\frac{1}{2},k}-v_{g_{j+\frac{1}{2},k}}\Delta x'_{j+\frac{1}{2},k}\right)\tfrac{1}{2}(U_{j+1,k}-U_{j,k}) \\ &+\ \left(u_{g_{j-\frac{1}{2},k}}\Delta y'_{j-\frac{1}{2},k}-v_{g_{j-\frac{1}{2},k}}\Delta x'_{j-\frac{1}{2},k}\right)\tfrac{1}{2}(U_{j,k}-U_{j-1,k}) \\ &+\ \left(u_{g_{j,k+\frac{1}{2}}}\Delta y'_{j,k+\frac{1}{2}}-v_{g_{j,k+\frac{1}{2}}}\Delta x'_{j,k+\frac{1}{2}}\right)\tfrac{1}{2}(U_{j,k+1}-U_{j,k}) \\ &+\ \left(u_{g_{j,k-\frac{1}{2}}}\Delta y'_{j,k-\frac{1}{2}}-v_{g_{j,k-\frac{1}{2}}}\Delta x'_{j,k-\frac{1}{2}}\right)\tfrac{1}{2}(U_{j,k}-U_{j,k-1}) \end{aligned}$$

The grid velocity $(u_g, v_g)^T$ is evaluated at face centers by averaging the appropriate nodal velocities.

The second change is in the implicit left-hand-side, which is constructed in exactly the same way as usual apart from the subtraction of $\vec{u}_g{\cdot}\vec{n}$ from all four eigenvalues. This is because

$$\frac{\partial(F\Delta y' - G\Delta x' - \vec{u}_g{\cdot}\vec{n}U)}{\partial U} \;=\; \frac{\partial(F\Delta y' - G\Delta x')}{\partial U} \;-\; \vec{u}_g{\cdot}\vec{n}\, I \qquad (7.85)$$

and so the eigenvectors are the same, and the eigenvalues are reduced by amount $\vec{u}_g{\cdot}\vec{n}$.

There are also changes to the wall boundary conditions. The combined inviscid flux term is

$$\begin{aligned}
&\overline{G}^{*}_{j,1} - (u_{wall}\Delta y'_{j,1} - v_{wall}\Delta x'_{j,1})(U_{j,2} - U_{j,1}) \\
&= \begin{pmatrix} \rho_{j,2}(u_{wall}\Delta y'_{j,1} - v_{wall}\Delta x'_{j,1}) \\ \rho_{j,2}u_{j,2}(u_{wall}\Delta y'_{j,1} - v_{wall}\Delta x'_{j,1}) + p_{j,2}\Delta y'_{j,1} \\ \rho_{j,2}v_{j,2}(u_{wall}\Delta y'_{j,1} - v_{wall}\Delta x'_{j,1}) - p_{j,2}\Delta x'_{j,1} \\ \rho_{j,2}H_{j,2}(u_{wall}\Delta y'_{j,1} - v_{wall}\Delta x'_{j,1}) \end{pmatrix}, \qquad (7.86)
\end{aligned}$$

where $\vec{u}_{wall}$ is the wall velocity. The viscous flux term is also modified due to the motion of the wall. For adiabatic walls, $V^{*}_{j,1}$ becomes

$$V^{*}_{j,1} \;=\; \mu_{j,1}\frac{\Delta s'}{\Delta n}\begin{pmatrix} 0 \\ u_{j,2} - u_{wall} \\ v_{j,2} - v_{wall} \\ (\frac{1}{2}u^2_{j,2} + \frac{1}{2}v^2_{j,2}) - (\frac{1}{2}u^2_{wall} + \frac{1}{2}v^2_{wall}) \end{pmatrix}, \qquad (7.87)$$

while for walls with a specified temperature, it becomes

$$V^{*}_{j,1} \;=\; \mu_{j,1}\frac{\Delta s'}{\Delta n}\begin{pmatrix} 0 \\ u_{j,2} - u_{wall} \\ v_{j,2} - v_{wall} \\ (\frac{1}{2}u^2_{j,2} + \frac{1}{2}v^2_{j,2} + \frac{1}{(\gamma-1)Pr}c^2_{j,2}) - (\frac{1}{2}u^2_{wall} + \frac{1}{2}v^2_{wall}) + \frac{1}{(\gamma-1)Pr}c^2_{wall}) \end{pmatrix}. \qquad (7.88)$$

Bibliography

[1] R. S. Abhari, G. R. Guenette, A. H. Epstein, and M. B. Giles. Comparison of time-resolved trubine rotor blade heat transfer measurements and numerical calculations. submitted to the 36th IGTI Gas Turbine Congress, June 1991.

[2] R. Beam and R. Warming. An Implicit Factored Scheme for the Compressible Navier-Stokes equations. *AIAA Journal*, 16:393–403, 1978.

[3] T. Cebeci and A. M. O. Smith. *Analysis of Turbulent Boundary Layers.* Academic Press, 1974.

[4] Y. S. Chen. 3-D stator-rotor interaction of the SSME. AIAA Paper 88-3095, 1988.

[5] R. V. Chima. Explicit Multigrid Algorithm for Quasi-Three-Dimensional Viscous Flows in Turbomachinery. *AIAA Journal of Propulsion and Power*, 3(5):397–405, 1987.

[6] J. F. Dannenhoffer and M. B. Giles. Convergence acceleration through the use of time inclining. *AIAA Journal*, 28(8):1457–1463, 1990.

[7] J. F. Dannenhoffer III and J. R. Baron. Robust grid adaptation for complex transonic flows. AIAA Paper 86-0495, 1986.

[8] A. H. Epstein, M. B. Giles, T. Shang, and A. K. Sehra. Blade row interaction effects on compressor measurements. AGARD 74th Specialists Meeting on Unsteady Aerodynamic Phenomena in Turbomachines, Aug 1989.

[9] J. I. Erdos, E. Alzner, and W. McNally. Numerical Solution of Periodic Transonic Flow Through a Fan Stage. *AIAA Journal*, 15(11):1559–1568, Nov 1977.

[10] A. Fourmaux. Unsteady flow calculation in cascades. ASME Paper 86-GT-178, 1986.

[11] M. B. Giles. Generalized conservation cells for finite volume calculations. AIAA Paper 87-1118-CP, June 1987.

[12] M. B. Giles. Calculation of Unsteady Wake Rotor Interaction. *AIAA Journal of Propulsion and Power*, 4(4):356–362, July/August 1988.

[13] M. B. Giles. Non-reflecting boundary conditions for the Euler equations. Technical Report TR-88-1, MIT Computational Fluid Dynamics Laboratory, 1988.

[14] M. B. Giles. Non-Reflecting Boundary Conditions For Euler Equation Calculations. *AIAA Journal*, 28(12), Dec. 1990.

[15] M. B. Giles. Stator/Rotor Interaction in a Transonic Turbine. *AIAA Journal of Propulsion and Power*, 6(5), September-October 1990.

[16] M. B. Giles and R. Haimes. Validation of a numerical method for unsteady flow calculations. submitted to the 36th IGTI Gas Turbine Congress, June 1991.

[17] M. G. Hall. Cell-vertex multigrid schemes for solution of the Euler equations. Technical Report 2029, Royal Aircraft Establishment, Mar 1985.

[18] H. P. Hodson. An inviscid blade-to-blade prediction of a wake-generated unsteady flow. ASME Paper 84-GT-43, June 1984.

[19] A. B. Johnson, M. J. Rigby, M. L. G. Oldfield, and M. B. Giles. Nozzle guide vane shock wave propagation and bifurcation in a transonic turbine rotor. ASME Paper 90-GT-310, June 1990.

[20] P. C. E. Jorgenson and R. V. Chima. An explicit runge-kutta method for unsteady rotor/stator interaction. AIAA Paper 88-0049, 1988.

[21] M. Koya and S. Kotake. Numerical Analysis of Fully Three-Dimensional Periodic Flows Through a Turbine Stage. *Journal of Engineering for Gas Turbines and Power*, 107:945–952, October 1985.

[22] J. P. Lewis, R. A. Delaney, and E. J. Hall. Numerical prediction of turbine vane-blade interaction. AIAA Paper 87-2149, June 1987.

[23] D. R. Lindquist. A comparison of numerical schemes on triangular and quadrilateral meshes. Master's thesis, M.I.T., May 1988.

[24] D. R. Lindquist and M. B. Giles. Generation and use of unstructured grids for turbomachinery. Proceedings of Computational Fluid Dynamics Symposium on Aeropropulsion, NASA CP-10045, April 1990.

[25] R. Löhner, K. Morgan, J. Peraire, and O. C. Zienkiewicz. Finite element methods for high speed flows. AIAA Paper 85-1531, 1985.

[26] R.-H. Ni. A Multiple Grid Scheme for Solving the Euler Equations. *AIAA Journal*, 20(11):1565–1571, Nov 1981.

[27] R.-H. Ni and O. SHarma. Using 3d euler flow simulations to assess effects of periodic unsteady flow through turbines. AIAA Paper 90-2357, 1990.

[28] R. J. G. Norton, W. T. Thompkins, and R. Haimes. Implicit finite difference schemes with non-simply connected grids - a novel approach. AIAA Paper 84-0003, 1984.

[29] M. M. Rai. Navier-Stokes simulations of rotor-stator interaction using patched and overlaid grids. *AIAA Journal of Propulsion and Power*, 3(5):387–396, 1987.

[30] M. M. Rai. Unsteady three-dimensional Navier-Stokes simulations of turbine rotor-stator interaction. AIAA Paper 87-2058, June 1987.

[31] R. D. Richtmeyer and K. W. Morton. *Difference Methods for Initial-Value Problems.* John Wiley & Sons, second edition, 1967.

[32] P. Roe. Characteristic-Based Schemes for the Euler Equations. *Ann. Rev. Fluid Mech.*, 18:337–65, 1986.

[33] H. Schlichting. *Boundary Layer Theory.* McGraw-Hill, 1979.

[34] O. P. Sharma, E. Renaud, T. L. Butler, K. Millsaps, R. P. Dring, and H. D. Joslyn. Rotor-stator interaction in multi-stage axial-flow turbines. AIAA Paper 88-3013, 1988.

[35] B. van Leer, W. Lee, and K. Powell. Sonic-point capturing. AIAA Paper 89-1945, 1989.