

by Zenotech

Solver Reference Guide

October 14, 2026

Solver Reference Guide

Zenotech

October 14, 2026

©Zenotech Ltd

Preface

This document contains algorithmic details of the Computational Fluid Dynamics solver zCFD. This is primarily a reference for users and developers of the software. It does not detail the software implementation of the equations although the most of the equations map directly on the code.

Contents

Mathematical Notation 7

1 zCFD 11
1.1 Introduction . . . 11

2 Navier Stokes Equation 13
2.1 Compressible Form . . . 13
2.2 Incompressible Form . . . 15
2.3 Integral Form . . . 16
2.4 Semi Discrete Form . . . 17
2.5 Dual Time Step . . . 17
2.6 Precondition . . . 17
2.7 Geometric Conservation Law . . . 18
2.8 Arbitrary Lagrangian-Eulerian . . . 18

3 Thermodynamics 19
3.1 Basic . . . 19
3.2 Equation Of State - EOS . . . 20
3.3 Material Properties . . . 22

4 Non Dimensionalisation 23
4.1 Basic . . . 23

5 Solver 25
5.1 Explicit . . . 25
5.2 Implicit . . . 25
5.3 Point Implicit . . . 27
5.4 Preconditioner . . . 27
5.5 Geometric Multigrid . . . 29
5.6 Prolongate Operator . . . 30
5.7 ALE Preconditioned HLLC Flux . . . 30
5.8 Gradients . . . 36
5.9 MUSCL Reconstruction . . . 37
5.10 Time Step . . . 40
5.11 Boundary Conditions . . . 41

6 PISO 55
6.1 Predictor Step . . . 55
6.2 Corrector Step . . . 57

 ©Zenotech Ltd

7 SIMPLE **59**
7.1 Predictor Step . . . 59
7.2 Corrector Step . . . 61

8 Rhie-Chow Interpolation **63**

9 Coupled Incompressible **65**

10 Turbulence Modelling **69**
10.1 Menter SST-2003Vsust Two Equation Model . . . 69
10.2 Hybrid RANS LES . . . 73
10.3 Diffusive Jacobian . . . 74
10.4 Source Term Jacobian . . . 80
10.5 Wall Function . . . 81
10.6 Menter SST SAS . . . 87

11 Gravity **89**
11.1 Force Potential . . . 89
11.2 Pressure Gradient . . . 89
11.3 Lapse Rates . . . 89

12 Stratified Atmospheric Flow **93**
12.1 Hydrostatic Base State . . . 93
12.2 Anelastic Continuity . . . 95
12.3 Buoyancy . . . 95
12.4 Perturbation Form of the Energy Equation . . . 95
12.5 Pressure Correction . . . 96
12.6 Boundary Conditions . . . 97
12.7 Sponge Layer . . . 97
12.8 Resolution Requirement . . . 98

13 Atmospheric Physics **99**
13.1 Richardson Number . . . 99
13.2 Potential temperature . . . 99
13.3 Obukhov Length . . . 99
13.4 Coriolis Force . . . 99
13.5 Turbulence Profile . . . 101

14 Radial Basis Functions (RBF) **103**
14.1 RBF interpolation method . . . 103

15 Wall Distance **105**

16 Overset Meshes **107**
16.1 Hole Cutting . . . 107
16.2 Fringe Cells . . . 108
16.3 Donor Search . . . 108
16.4 Interpolation . . . 108
16.5 Transformation of the Solution Between Frames . . . 109
16.6 Coupling Within the Dual Time Step . . . 109

16.7 Treatment of Blanked Cells 109

Bibliography **113**

 ©Zenotech Ltd

Mathematical Notation

α	Radial Basis Function scaling parameter
α_m	Runge Kutta stage coefficient
$\bar{\rho}$	Base-state (hydrostatic) density
$\bar{\theta}$	Base-state potential temperature
$\bar{p}$	Base-state pressure
$\bar{T}$	Base-state temperature
δ_{ij}	Defined as $\begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$
$\dot{x}$	Control volume interface velocity vector
Γ	Lapse rate $-\partial T/\partial z$
γ	Ratio of specific heats
λ_z	Vertical wavelength $2\pi U/N$
μ	Dynamic viscosity
μ_{ref}	Experimentally determined dynamic viscosity
∇	Defined as $\left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right]^T$
ν	Viscosity
Ω	Volume
Ω_E	Planetary rotation rate
ϕ	Transported quantity
π	Exner function $(\bar{p}/p_0)^{R/C_p}$
ψ	Angular velocity $[\psi_x, \psi_y, \psi_z]^T$
ρ	Density
ρ_0	Total density
ρ_p	Partial derivative $\frac{\partial \rho}{\partial p}$ at constant T

ρ_T	Partial derivative $\frac{\partial \rho}{\partial T}$ at constant p
σ	Sponge layer damping rate
σ_{ij}	Viscous stress tensor
τ	Pseudo time step
τ_{ij}	Shear stress
θ	Potential temperature
θ'	Potential temperature perturbation $\theta - \bar{\theta}$
^{k}	Pseudo time step
^{m}	Runge Kutta stage
^{n}	Time step
$_i$	Cell index
A	Face area
b	Buoyancy acceleration
c	speed of sound
C_p	Specific heat constant pressure
C_v	Specific heat constant volume
d	wall distance
E	Specific total energy
e	Internal energy
F	Advected flux tensor
f	Coriolis parameter $2\Omega_E \sin\phi$
f	Inviscid wall flux
F_f	Volume flux through face f
G	Diffusive flux tensor
g	Gravity
H	Total Enthalpy
h	Enthalpy
h_p	Partial derivative $\frac{\partial h}{\partial p}$ at constant T
h_T	Partial derivative $\frac{\partial h}{\partial T}$ at constant p
I	Identity Matrix

©Zenotech Ltd

k	Turbulent kinetic energy
k_h	Heat transfer coefficient
Log	Is to base e
N	Brunt-Väisälä (buoyancy) frequency
n	Vector normal surface
P	Preconditioning matrix
p	Pressure
p'	Pressure correction
p_0	Total pressure
P_c	Molecular attraction between water molecules
p_s	Static pressure
p_{s_∞}	Reference static pressure at infinity
Pr	Prandtl number
Q	Primitive variables $[p, u, v, w, T, \phi]^T$
q_i	Heat flux component i
R	Gas constant
r	Coordinate Vector $[x, y, z]^T$
$R()$	Residual function
$R^c()$	Coarse mesh residual function
S	Source terms
s	Entropy
S_f	Face area vector
S_{ij}	Viscous strain rate tensor
T	Temperature
t	Time
T_0	Total temperature
T_s	Static temperature
T_{ref}	Experimentally determined temperature
T_{s_∞}	Reference static temperature at infinity
U_r	Reference velocity for preconditioner

V	Velocity vector $[u, v, w]^T$
V_g	Geostrophic wind
W	Conserved variables $[\rho, \rho u, \rho v, \rho w, \rho E, \rho \phi]^T$
x	x ordinate
y	y ordinate
z	z ordinate

©Zenotech Ltd

Chapter 1

zCFD

1.1 Introduction

zCFD is a many-core accelerated Computational Fluid Dynamics solver that solves the Steady or Unsteady Reynolds Averaged Navier Stokes (RANS) equations using the finite volume technique. This is based on a cell centred approach on arbitrary polyhedral meshes with the boundary conditions applied in a weak formulation. The scheme implemented is formally 2^{nd} order accurate in time and 3^{rd} order accurate in space.

The spacial accuracy is achieved by using a slope limited MUSCL reconstruction at cell interfaces. The HLLC scheme is then used with the reconstructed values to calculate the interface flux.

The primary solver is based on a multi-stage Explicit solver using local time stepping and geometric multigrid to accelerate convergence.

Turbulence modelling is provided by an implementation of Menter SST two equation model which is can be used in low Reynolds number mode (i.e. integration to the wall) or high Reynolds number mode (i.e. wall functions)

Chapter 2

Navier Stokes Equation

2.1 Compressible Form

Continuity

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \tag{2.1}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho V = 0 \tag{2.2}$$

Momentum

$$\begin{aligned}\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u\, u)}{\partial x} + \frac{\partial(\rho u\, v)}{\partial y} + \frac{\partial(\rho u\, w)}{\partial z} =& \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} \\ &+ \frac{\partial \varphi}{\partial x} + \rho f_{c_x} + \rho F_{ext_x}\end{aligned} \tag{2.3}$$

$$\begin{aligned}\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v\, u)}{\partial x} + \frac{\partial(\rho v\, v)}{\partial y} + \frac{\partial(\rho v\, w)}{\partial z} =& \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} \\ &+ \frac{\partial \varphi}{\partial y} + \rho f_{c_y} + \rho F_{ext_y}\end{aligned} \tag{2.4}$$

$$\begin{aligned}\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho w\, u)}{\partial x} + \frac{\partial(\rho w\, v)}{\partial y} + \frac{\partial(\rho w\, w)}{\partial z} =& \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} \\ &+ \frac{\partial \varphi}{\partial z} + \rho f_{c_z} + \rho F_{ext_z}\end{aligned} \tag{2.5}$$

In vector notation

$$\frac{\partial}{\partial t}(\rho V) + \nabla \cdot (\rho V \cdot V) = \nabla \cdot \sigma + \nabla \varphi + \rho f_c + \rho F_{ext} \tag{2.6}$$

For air

$$\frac{\partial}{\partial t}(\rho V) + \nabla \cdot (\rho V \cdot V) = -\nabla p + \nabla \cdot \tau + \nabla \varphi + \rho f_c + \rho F_{ext} \tag{2.7}$$

Note that gravity is treated as a force potential

$$F = -\nabla \varphi = -\rho g \tag{2.8}$$

Energy

$$\begin{aligned}\frac{\partial}{\partial t}(\rho E)+\frac{\partial}{\partial x}(\rho E\,u)+\frac{\partial}{\partial y}(\rho E\,v)+\frac{\partial}{\partial z}(\rho E\,w) = &-\frac{\partial q}{\partial x}-\frac{\partial q}{\partial y}-\frac{\partial q}{\partial z}\\ &+\frac{\partial}{\partial x}(u\sigma_{xx}+v\sigma_{xy}+w\sigma_{xz})\\ &+\frac{\partial}{\partial y}(u\sigma_{xy}+v\sigma_{yy}+w\sigma_{yz})\\ &+\frac{\partial}{\partial z}(u\sigma_{xz}+v\sigma_{yz}+w\sigma_{zz})\\ &+\frac{\partial}{\partial x}(u\,\varphi)+\frac{\partial}{\partial y}(v\,\varphi)+\frac{\partial}{\partial z}(w\,\varphi)\\ &+u\left(\rho f_{c_x}+\rho F_{ext_x}\right)\\ &+v\left(\rho f_{c_y}+\rho F_{ext_y}\right)\\ &+w\left(\rho f_{c_z}+\rho F_{ext_z}\right)\end{aligned} \tag{2.9}$$

In vector notation

$$\frac{\partial}{\partial t}(\rho E)+\nabla\cdot(\rho E\cdot V)=\nabla\cdot(V\cdot(\varphi))-\nabla\cdot q+\nabla\cdot(V\cdot\sigma)+V\cdot(\rho f_c+\rho F_{ext}) \tag{2.10}$$

For air

$$\frac{\partial}{\partial t}(\rho E)+\nabla\cdot(\rho E\cdot V)=-\nabla\cdot(V\cdot(p-\varphi))-\nabla\cdot q+\nabla\cdot(V\cdot\tau)+V\cdot(\rho f_c+\rho F_{ext}) \tag{2.11}$$

Newtonian Stress tensor σ_{ij} is

$$\sigma_{ij}=-\left[p+\left(\frac{2}{3}\mu-\lambda\right)\nabla\cdot V\right]\delta_{ij}+\mu 2S_{ij} \tag{2.12}$$

For dilute monatomic gases the expansion viscosity $\lambda=0$, for water $\lambda=3\mu$ and for complex liquids like benzene $\lambda=100\mu$

For air assume

$$\sigma_{ij}=-p\delta_{ij}+\tau_{ij} \tag{2.13}$$

Viscous stress

$$\tau_{ij}=2\mu\left(S_{ij}-\frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) \tag{2.14}$$

Viscous strain rate tensor

$$S_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) \tag{2.15}$$

Heat flux from Fourier's Law

$$q_i=k_h\frac{\partial T}{\partial x_j}=-C_p\frac{\mu}{Pr}\frac{\partial T}{\partial x_j} \tag{2.16}$$

Note when gravity is active the temperature gradient is replaced by the potential temperature gradient so as to recover the correct temperature lapse rate.

$$q_i=k_h\frac{\partial T}{\partial x_j}=-C_p\frac{\mu}{Pr}\left[\frac{\partial T}{\partial x_j}-\frac{g_j}{C_p}\right] \tag{2.17}$$

©Zenotech Ltd

2.2 Incompressible Form

Continuity for incompressible flows

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho V = 0 \tag{2.18}$$

$$\nabla \cdot \rho V = 0 \tag{2.19}$$

Momentum equation

$$\frac{\partial}{\partial t}(\rho V) + \nabla \cdot (\rho V \cdot V) = \nabla \cdot \sigma_{ij} + \nabla \cdot \rho F_{ext} \tag{2.20}$$

Expanding the Newtonian stress tensor assuming $\nabla \cdot V = 0$

$$\nabla \cdot \sigma_{ij} = -\nabla \cdot p + \nabla \cdot 2\mu S_{ij} = -\nabla \cdot p + \nabla \cdot \mu \nabla V = -\nabla \cdot p + \mu \nabla^2 V \tag{2.21}$$

Therefore

$$\frac{\partial}{\partial t}(\rho V) + \nabla \cdot (\rho V \cdot V) = -\nabla \cdot p + \nabla \cdot \mu \nabla V + \nabla \cdot \rho F_{ext} \tag{2.22}$$

Derive incompressible internal energy equation

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\rho E \cdot V) = -\nabla \cdot q + \nabla \cdot (\sigma_{ij} V) + \rho V F_{ext} \tag{2.23}$$

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\rho (e + \frac{V^2}{2}) \cdot V) = -\nabla \cdot q + \nabla \cdot (\sigma_{ij} V) + \rho V F_{ext} \tag{2.24}$$

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\rho e \cdot V) + \nabla (\rho V \cdot V) = -\nabla \cdot q + V \nabla \cdot \sigma_{ij} - p \nabla \cdot V + V \nabla . \tau_{ij} + \rho V F_{ext} \tag{2.25}$$

Cancelling terms that relate to momentum conservation and expanding viscous stress τ_{ij}

$$\frac{\partial}{\partial t}(\rho e) + \nabla \cdot (\rho e \cdot V) = -\nabla \cdot k \nabla T - p \nabla \cdot V + \nabla \cdot (V \cdot \mu \nabla V) \tag{2.26}$$

For incompressible flow $\nabla \cdot V = 0$

$$\frac{\partial}{\partial t}(\rho e) + \nabla \cdot (\rho e \cdot V) = -\nabla \cdot k_h \nabla T + \nabla \cdot (V \cdot \mu \nabla V) \tag{2.27}$$

2.3 Integral Form

Combining the vector form of each individual equation into tensor form and grouping the derivatives in advective F, diffusive G and source terms S.

$$\frac{\partial W}{\partial t} + \nabla \cdot F - \nabla \cdot G = S \tag{2.28}$$

This equation is integrated over each control volume Ω

$$\frac{\partial}{\partial t}\int_\Omega W dV + \int_\Omega \nabla \cdot F dV - \int_\Omega \nabla \cdot G dV = \int_\Omega S dV \tag{2.29}$$

The next step relies on application of the Divergence theorem which states that the surface integral of a scalar function is equal to the volume integral of the gradient of the scalar function

$$\int_\Omega \nabla \cdot F dV = \int F \cdot n dA \tag{2.30}$$

Hence

$$\frac{\partial}{\partial t}\int_\Omega W dV + \int F \cdot n dA - \int G \cdot n dA = \int_\Omega S dV \tag{2.31}$$

where

$$W = [\rho, \rho u, \rho v, \rho w, \rho E, \rho \phi]^T$$

$$Q = [p, u, v, w, T, \phi]^T$$

$$F = \begin{pmatrix} \rho u & \rho v & \rho w \\ \rho u^2 + p - \varphi & \rho uv & \rho uw \\ \rho uv & \rho v^2 + p - \varphi & \rho vw \\ \rho uw & \rho vw & \rho w^2 + p - \varphi \\ \rho uE + up - u\varphi & \rho vE + vp - v\varphi & \rho wE + wp - w\varphi \\ \rho u\phi & \rho v\phi & \rho w\phi \end{pmatrix}$$

$$G = \begin{pmatrix} 0 & 0 & 0 \\ \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{xy} & \tau_{yy} & \tau_{yz} \\ \tau_{xz} & \tau_{yz} & \tau_{zz} \\ \tau_{xx}u + \tau_{xy}v + \tau_{xz}w + k_h \frac{\partial T}{\partial x} & \tau_{xy}u + \tau_{yy}v + \tau_{yz}w + k_h \frac{\partial T}{\partial y} & \tau_{xz}u + \tau_{yz}v + \tau_{zz}w + k_h \frac{\partial T}{\partial z} \\ 0 & 0 & 0 \end{pmatrix}$$

$$S = \left[0, \rho f_{c_x} + \rho F_{ext_x}, \rho f_{c_y} + \rho F_{ext_y}, \rho f_{c_z} + \rho F_{ext_z}, V \cdot (\rho f_c + \rho F_{ext}), 0\right]^T$$

©Zenotech Ltd

2.4 Semi Discrete Form

Assuming piecewise constant solution over the volume i

$$\frac{\partial\left(\Omega_i W_i\right)}{\partial t}+\sum\left[F(W)-G(W)\right]\cdot A-\Omega_i S_i=0 \tag{2.32}$$

Re-arrange and define Residual function $R(W)=\sum\left[F(W)-G(W)\right]\cdot A-\Omega_i S_i$

$$\frac{\partial\left(\Omega_i W_i\right)}{\partial t}+R(W)=0 \tag{2.33}$$

2.5 Dual Time Step

Expanding the time derivative terms with 2^{nd} order accurate Backward Difference Formula with a view to solving implicitly.

$$\frac{\left(3\Omega_i^{n+1}W_i^{n+1}-4\Omega_i^{n}W_i^{n}+\Omega_i^{n-1}W_i^{n-1}\right)}{2\Delta t}+R(W^{n+1})=0 \tag{2.34}$$

This equation can be treated as a modified steady state problem solved using local time steps and multigrid in a fictious/pseudo time step τ.

$$\Omega_i^{n+1}\frac{\partial W_i}{\partial\tau}+R^*(W)=0 \tag{2.35}$$

$$R^*(W)=R(W)+\frac{\left(3\Omega_i^{n+1}W_i-4\Omega_i^{n}W_i^{n}+\Omega_i^{n-1}W_i^{n-1}\right)}{2\Delta t} \tag{2.36}$$

See [7] for more details.

Expand the pseudo time derivative using a forward difference approximation .

$$\Omega_i^{n+1}\frac{\left(W_i^{k+1}-W_i^{k}\right)}{\Delta\tau}=-R^*(W) \tag{2.37}$$

2.6 Precondition

We introduce a preconditioner to improve the condition number of the system at low speeds. This is achieved through the use of a modified Jacobian that effectively changes the wave speeds in the preconditioned system. This can be intuitively specified in primitive variables Q but the solver operates on conserved variables. Therefore the preconditioning matrix consist of transforming the variables into primitive form before converting them back using the modified/preconditioned transformation which is described in more detail on page 17.

$$P=\frac{\partial W}{\partial Q}\left(\frac{\partial W}{\partial Q}\right)^{-1}\Bigg|_m \tag{2.38}$$

$$\frac{\partial W}{\partial Q}=\begin{pmatrix}\rho_p & 0 & 0 & 0 & \rho_T & 0\\ u\rho_p & \rho & 0 & 0 & u\rho_T & 0\\ v\rho_p & 0 & \rho & 0 & v\rho_T & 0\\ w\rho_p & 0 & 0 & \rho & w\rho_T & 0\\ \rho_p H-1+\rho h_p & \rho u & \rho v & \rho w & H\rho_T+\rho h_T & 0\\ \phi\rho_p & 0 & 0 & 0 & \phi\rho_T & \rho\end{pmatrix} \tag{2.39}$$

$$\left(\frac{\partial W}{\partial Q}\right)^{-1} = \begin{pmatrix} \frac{\rho h_T + \rho_T\left(H-V^2\right)}{d} & \frac{\rho_T}{d}u & \frac{\rho_T}{d}v & \frac{\rho_T}{d}w & -\frac{\rho_T}{d} & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 & 0 \\ \frac{-\rho_p\left(H-V^2\right)+1-\rho h_p}{d} & -\frac{\rho_p}{d}u & -\frac{\rho_p}{d}v & -\frac{\rho_p}{d}w & \frac{\rho_p}{d} & 0 \\ -\frac{\phi}{\rho} & 0 & 0 & 0 & 0 & \frac{1}{\rho} \end{pmatrix} \tag{2.40}$$

where $d = \rho\rho_p h_T + \rho_T\left(1 - \rho h_p\right)$

Introduce the transformation matrix $P^{-1} = \left(\frac{\partial W}{\partial Q}\right)\Big|_m \frac{\partial Q}{\partial W}$

$$\Omega_i^{n+1} P^{-1} \frac{\partial W_i}{\partial \tau} + R^*(W) = 0 \tag{2.41}$$

Rearrange and expand the pseudo time derivative using a forward difference approximation

$$\Omega_i^{n+1} P^{-1} \frac{\left(W_i^{k+1} - W_i^k\right)}{\Delta\tau} = -R^*(W) \tag{2.42}$$

2.7 Geometric Conservation Law

When the control volume is moving and/or deforming, the conservation property may be violated unless the following geometric conservation law (GCL) is satisfied

$$\frac{\partial \Omega}{\partial t} - \int \dot{x} dA = 0 \tag{2.43}$$

Expanding the time derivative using a 2^{nd} order accurate backward difference

$$\frac{3\Omega_i^{n+1} - 4\Omega_i^n + \Omega_i^{n-1}}{2\Delta t} = \sum \dot{x} \cdot A_f \tag{2.44}$$

Rearranging

$$\Omega_i^{n+1} = \frac{2\Delta t}{3}\left(\sum \dot{x} \cdot A_f + 4\Omega_i^n - \Omega_i^{n-1}\right) \tag{2.45}$$

2.8 Arbitrary Lagrangian-Eulerian

The advective flux tensor F for a moving control volume (i.e when Ω is a function of time) is

$$F = \begin{pmatrix} \rho(u - \dot{x}_u) & \rho(v - \dot{x}_v) & \rho(w - \dot{x}_w) \\ \rho u(u - \dot{x}_u) + p & \rho u(v - \dot{x}_v) & \rho u(w - \dot{x}_w) \\ \rho v(u - \dot{x}_u) & \rho v(v - \dot{x}_v) + p & \rho v(w - \dot{x}_w) \\ \rho w(u - \dot{x}_u) & \rho w(v - \dot{x}_v) & \rho w(w - \dot{x}_w) + p \\ \rho E(u - \dot{x}_u) + pu & \rho E(v - \dot{x}_v) + pv & \rho E(w - \dot{x}_w) + pw \\ \rho\phi(u - \dot{x}_u) & \rho\phi(v - \dot{x}_v) & \rho\phi(w - \dot{x}_w) \end{pmatrix}$$

when $\dot{x} = 0$ this system corresponds to the Eulerian description whereas $\dot{x} = V$ results in the Lagrangian description where the control volume moves with the fluid velocity.

©Zenotech Ltd

Chapter 3

Thermodynamics

3.1 Basic

Total Energy

$$E = e + \frac{V^2}{2} + k \tag{3.1}$$

Enthalpy

$$h = e + \frac{p}{\rho} \tag{3.2}$$

Total Enthalpy

$$H = h + \frac{V^2}{2} + k \tag{3.3}$$

$$H = E + \frac{p}{\rho} \tag{3.4}$$

Entropy

$$ds = \frac{dh}{T} - \frac{1}{\rho T} dp \tag{3.5}$$

$$\left.\frac{\partial h}{\partial T}\right|_p = \left.\frac{\partial e}{\partial T}\right|_p + P \left.\frac{\partial v}{\partial T}\right|_p = C_p \tag{3.6}$$

$$\left.\frac{\partial e}{\partial T}\right|_v = C_v \tag{3.7}$$

$$\frac{\partial e}{\partial s} = T \tag{3.8}$$

$$\frac{\partial e}{\partial V} = P \tag{3.9}$$

Specific Heats

$$\gamma = \frac{C_p}{C_v} \tag{3.10}$$

$$R = C_p - C_v \tag{3.11}$$

$$C_p = \frac{R}{\gamma - 1} + R = \frac{\gamma R}{\gamma - 1} \tag{3.12}$$

$$C_v = \frac{R}{\gamma - 1} \tag{3.13}$$

3.2 Equation Of State - EOS

Definition

The equation of state defines the relationship between pressure, density and energy

$$p = p\left(\rho, e\right) \tag{3.14}$$

$$c^2 \equiv \left.\frac{dp(\rho, s)}{d\rho}\right|_s = \frac{\partial p}{\partial e}\frac{p}{\rho^2} + \frac{\partial p}{\partial \rho} = \frac{\rho\frac{\partial h}{\partial T}}{\rho\frac{\partial \rho}{\partial p}\frac{\partial h}{\partial T} + \frac{\partial \rho}{\partial T}\left(1 - \rho\frac{\partial h}{\partial p}\right)} \tag{3.15}$$

The following partial derivatives have been calculated in their most general forms with the arbitrary constants ξ and β varying for different EOS.

$$\rho_p = \left.\frac{\partial \rho}{\partial p}\right|_T = \rho\xi \tag{3.16}$$

$$\rho_T = \left.\frac{\partial \rho}{\partial T}\right|_p = -\rho\beta \tag{3.17}$$

$$h_T = \left.\frac{\partial h}{\partial T}\right|_p = C_p \tag{3.18}$$

$$h_p = \left.\frac{\partial h}{\partial p}\right|_T = \frac{1 - \beta T}{\rho} \tag{3.19}$$

Ideal Gas EOS

$$p = (\gamma - 1)\rho e = (\gamma - 1)\left(\rho E - \frac{\rho V^2}{2} - \rho k\right) \tag{3.20}$$

$$T = \frac{(\gamma - 1)}{R}e \tag{3.21}$$

Assuming a thermally perfect fluid

$$h = C_p T \tag{3.22}$$

$$e = C_v T \tag{3.23}$$

$$p = \rho(C_p - C_v)T = \rho RT \tag{3.24}$$

©Zenotech Ltd

$$c^2 = \gamma \frac{p}{\rho} \tag{3.25}$$

$$\xi = \frac{1}{p} \tag{3.26}$$

$$\beta = \frac{1}{T} \tag{3.27}$$

Tammann EOS

$$p = (\gamma - 1)\left(\rho e - \frac{\rho V^2}{2} - P_c\right) = (\gamma - 1)\,\rho e - \gamma P_c \tag{3.28}$$

Assuming a thermally perfect fluid

$$h = C_p T \tag{3.29}$$

$$e = C_v T \tag{3.30}$$

$$p = \rho(C_p - C_v)T - \gamma P_c = \rho R T - \gamma P_c \tag{3.31}$$

$$c^2 = \gamma \frac{p + P_c}{\rho} \tag{3.32}$$

$$\xi = \frac{1}{p - \gamma p_c} \tag{3.33}$$

$$\beta = \frac{1}{T} \tag{3.34}$$

Tait EOS

$$p = B\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right] + A \tag{3.35}$$

$$A = 1.0 \times 10^6,\ B = 3.31 \times 10^9,\ \rho_0 = 1.0,\ \gamma = 7.15 \tag{3.36}$$

$$c^2 = \gamma \frac{B\left(\frac{\rho}{\rho_0}\right)^{\gamma}}{\rho} \tag{3.37}$$

$$c^2 = \gamma \frac{p + B - A}{\rho} \tag{3.38}$$

$$\xi = \gamma \rho_0 \left[\frac{p - A}{B} + 1\right]^{\frac{1-\gamma}{\gamma}} \frac{\rho_0}{\rho B} \tag{3.39}$$

$$\beta = 0 \tag{3.40}$$

$$h = C_p T \tag{3.41}$$

$$e = C_v T \tag{3.42}$$

When C_p not a function of temperature

$$T = \frac{E - \frac{V^2}{2} - gz + \frac{p}{\rho}}{C_p} \tag{3.43}$$

3.3 Material Properties

Air

Dynamic viscosity is calculated using Sutherland's Law

$$\mu = \mu_{ref} \left(\frac{T}{T_{ref}} \right)^{\frac{3}{2}} \frac{T_{ref} + S}{T + S} \tag{3.44}$$

where for air the Sutherland temperature is $S = 110.4K$

Water

Dynamic viscosity is calculated using

$$\mu = A\,(10)^{\frac{B}{T-C}} \tag{3.45}$$

where for water $A = 2.414 \times 10^{-5} Pas$; $B = 247.8K$ and $C = 140K$

©Zenotech Ltd

Chapter 4

Non Dimensionalisation

4.1 Basic

$$\frac{\rho}{\rho_\infty} = \frac{\rho^*}{\rho^*_\infty} \quad \text{with} \quad \rho_\infty = 1 \tag{4.1}$$

$$\frac{p}{p_\infty} = \frac{p^*}{p^*_\infty} \quad \text{with} \quad p_\infty = 1 \tag{4.2}$$

$$\frac{T}{T_\infty} = \frac{T^*}{T^*_\infty} \quad \text{with} \quad T_\infty = 1 \tag{4.3}$$

$$\frac{U}{U_\infty} = \frac{U^*}{U^*_\infty} \quad \text{with} \quad U_\infty = M_\infty c_\infty = M_\infty \sqrt{\frac{\gamma p_\infty}{\rho_\infty}} = \sqrt{\gamma} M_\infty \tag{4.4}$$

$$\mu_\infty = \frac{\rho_\infty U_\infty L_{ref}}{Re} \tag{4.5}$$

$$\frac{t}{t^*} = \frac{c^*_\infty}{c_\infty} = \frac{c^*_\infty}{\sqrt{\gamma}} = \sqrt{R^* T^*_\infty} \tag{4.6}$$

$$\frac{g}{g^*} = \left(\frac{t^*}{t}\right)^2 = \left(\frac{c_\infty}{c^*_\infty}\right)^2 = \frac{1}{R^* T^*_\infty} \tag{4.7}$$

Ideal Gas EOS

$$p_\infty = \rho_\infty R T_\infty \quad \text{with} \quad R = 1 \tag{4.8}$$

Chapter 5

Solver

5.1 Explicit

An explicit scheme by definition uses the current solution to calculate the derivative terms which are then used to march the solution forward in time.

$$\Omega_i^{n+1} P^{-1} \frac{\partial W_i}{\partial \tau} + R^*(W^k) = 0 \tag{5.1}$$

Using (2.42)

$$\Omega_i^{n+1} P^{-1} \frac{\left(W_i^{k+1} - W_i^k\right)}{\Delta \tau} = -R^*(W^k) \tag{5.2}$$

Runge Kutta

A multi stage time marcher enables larger time steps to be taken compared to purely Euler marching through increased stability. A typical m stage scheme with α_m coefficients:

$$W_i^{k,m=1} = W_i^k \tag{5.3}$$

$$W_i^{k,m+1} = W_i^{k,m} - \alpha_m P \frac{\Delta \tau_i}{\Omega_i^{n+1}} R^*(W^{k,m}) \tag{5.4}$$

$$\vdots$$

$$W_i^{k+1} = W_i^{k,m=5} \tag{5.5}$$

5.2 Implicit

An implicit scheme calculates the derivative terms using the solution at the next time step. This results in a linear system of equations that can be solved using a range of linear solvers and benefits from increased stability enabling very large time steps to be taken.

$$\Omega_i^{n+1} P^{-1} \frac{\partial W_i}{\partial \tau} + R^*(W^{k+1}) = 0 \tag{5.6}$$

Using (2.42)

$$\Omega_i^{n+1} P^{-1} \frac{\left(W_i^{k+1} - W_i^k\right)}{\Delta\tau} = -R^*\left(W^{k+1}\right) \tag{5.7}$$

Estimate residual at next time level through a truncated Taylor series expansion

$$\Omega_i^{n+1} P^{-1} \frac{\left(W_i^{k+1} - W_i^k\right)}{\Delta\tau} = -\left[R^*(W^k) + \frac{\partial R^*}{\partial W}\left(W_i^{k+1} - W_i^k\right)\right] \tag{5.8}$$

Rearranging

$$\left[\frac{\Omega_i^{n+1}}{\Delta\tau} P^{-1} + \frac{\partial R^*}{\partial W}\right]\left(W_i^{k+1} - W_i^k\right) = -R^*(W^k) \tag{5.9}$$

Solve system of linear equations for δW_i by rewriting (4.9) in the form

$$A\delta W_i = B \tag{5.10}$$

With $A = \left[\frac{\Omega_i^{n+1}}{\Delta\tau} P^{-1} + \frac{\partial R^*}{\partial W}\right]$, $\delta W_i = \left(W_i^{k+1} - W_i^k\right)$ and $B = -R^*(W^k)$. Then, the solution at the next pseudo time step is given by

$$W_i^{k+1} = \delta W_i + W_i^k \tag{5.11}$$

Note that by using (2.15)

$$\frac{\partial R^*}{\partial W} = \sum_{faces} \frac{\partial R}{\partial W} + \Omega \frac{3}{2\Delta t} I \tag{5.12}$$

$$\frac{\partial R^*}{\partial W} = \sum_{faces} \left[\left(\frac{\partial F}{\partial W} - \frac{\partial G}{\partial W}\right) \cdot A_f\right] - \Omega \frac{\partial S}{\partial W} + \Omega \frac{3}{2\Delta t} I \tag{5.13}$$

Finite Difference Jacobian

This approximates the jacobian using the Finite Difference approximation to perturb the solution vector one variable at a time by the value of ϵ, enabling numerical evaluation of the matrix entries.

$$\frac{\partial R^*}{\partial W} = \frac{R^*\left(W^k + \epsilon\right) - R^*\left(W^k\right)}{\epsilon} \tag{5.14}$$

Analytic Jacobian

See [1]

The total size of the system can be reduced by not storing the solution for the halo cells but accounting for their effect in the adjacent interior cell. Spliting contributions from interior cells k and halo cell l, the equation for cell i in the linear system becomes

$$\begin{aligned}\left[\frac{\Omega_i^{n+1}}{\Delta\tau} P^{-1} + \left(\sum_n \left[\left(\frac{\partial F}{\partial W_i} - \frac{\partial G}{\partial W_i}\right) \cdot A_f\right] - \Omega_i^{n+1} \frac{\partial S}{\partial W_i} + \Omega_i^{n+1} \frac{3}{2\Delta t} I\right)\right] \delta W_i \\ + \sum_{n, n \neq l} \left[\left(\frac{\partial F}{\partial W_n} - \frac{\partial G}{\partial W_n}\right) \cdot A_f\right] \delta W_n \\ + \left[\left(\frac{\partial F}{\partial W_l} - \frac{\partial G}{\partial W_l}\right) \cdot A_f\right] \delta W_l = -R^*(W^k)\end{aligned} \tag{5.15}$$

©Zenotech Ltd

To remove the dependency on cell l we can establish an implicit relation between the interior cell and the halo.

$$\delta W_l = C\delta W_i \tag{5.16}$$

$$\left[\frac{\Omega_i^{n+1}}{\Delta\tau}P^{-1} + \left(\sum_n \left(\frac{\partial F}{\partial W_i} - \frac{\partial G}{\partial W_i}\right) - \Omega_i^{n+1}\frac{\partial S}{\partial W_i} + \Omega_i^{n+1}\frac{3}{2\Delta t}I - \left(\frac{\partial F}{\partial W_l} - \frac{\partial G}{\partial W_l}\right)C\right)\right]\delta W_i + \sum_{n,n\neq l}\left(\frac{\partial F}{\partial W_n} - \frac{\partial G}{\partial W_n}\right)\delta W_n = -R^*(W^k) \tag{5.17}$$

5.3 Point Implicit

$$\Omega_i^{n+1}P^{-1}\frac{\left(W_i^{k+1} - W_i^k\right)}{\Delta\tau} = -R^*\left(W^{k+1}\right) \tag{5.18}$$

$$\frac{\partial R^*}{\partial W} = \frac{R^*\left(W^k + \epsilon\right) - R^*\left(W^k - \epsilon\right)}{2\epsilon} = A \tag{5.19}$$

A can then be split into 3 components, the viscous component A_v, positive and negative components A^+ and A^- with $T\lambda T^{-1}$ the standard eigenvalue decomposition of matrix A.

$$A = A^+ + A^- + A_v = \left(T\Lambda^+T^{-1} + \lambda_m I\right) + \left(T\Lambda^-T^{-1} - \lambda_m I\right) + A_v \tag{5.20}$$

with m eigenvalues given by

$$\lambda_m = \beta\left(|V_n| + c\right) \tag{5.21}$$

For point implicit consider only positive wave speeds

$$A = A^+ + \frac{A_v}{2} \tag{5.22}$$

5.4 Preconditioner

The behaviour of the preconditioner is to de-couple pressure and density at low speeds, therefore we are looking to provide an alternative formulation for ρ_p. See [19] for more details.

Define

$$\Theta = \left(\frac{1}{U_r^2} - \frac{\rho_T}{\rho h_T}\right) \tag{5.23}$$

$$U_r = \min\left[\max\left(V, \frac{\nu}{\Delta x}, \frac{\alpha}{\Delta x}, \epsilon\sqrt{\frac{\Delta p}{\rho}}\right), c\right] \; where\, \epsilon \approx 10^{-3}$$

Define

$$\left(\frac{\partial W}{\partial Q}\right)^{-1}\Bigg|_m = \left(\frac{\partial W}{\partial Q}\right)^{-1} \; with\, \rho_p = \Theta \tag{5.24}$$

Wave Speeds

Define the face normal velocity as

$$q = V \cdot n \tag{5.25}$$

The wave speeds before and after preconditioning are

$$\lambda = q \pm c \tag{5.26}$$

$$\lambda^{'} = q^{'} \pm c^{'} \tag{5.27}$$

where

$$q^{'} = q\left(1 - \frac{1}{2}\left(1 - \frac{U_r^2}{c^2}\right)\right) = \frac{q}{2}\left(1 + \frac{U_r^2}{c^2}\right)$$

$$c^{'} = \sqrt{\left(\frac{1}{2}\left(1 - \frac{U_r^2}{c^2}\right)\right)^2 q^2 + U_r^2}$$

Matrix P

Substitute ρ_p with Θ in (2.40) and multiply by (2.39)

$$P = \frac{\partial W}{\partial Q}\left(\frac{\partial W}{\partial Q}\right)^{-1}\Big|_m = \begin{pmatrix} A & Bu & Bv & Bw & -B & 0 \\ (A-1)\,u & Bu^2+1 & Buv & Buw & -Bu & 0 \\ (A-1)\,v & Buv & Bv^2+1 & Bvw & -Bv & 0 \\ (A-1)\,w & Buw & Bvw & Bw^2+1 & -Bw & 0 \\ D - V^2 & (C+1)\,u & (C+1)\,v & (C+1)\,w & -C & 0 \\ (A-1)\,\phi & Bu\phi & Bv\phi & Bw\phi & -B\phi & 1 \end{pmatrix} \tag{5.28}$$

Where

$$A = \frac{\rho\rho_p h_T - \rho_T\rho_p\left(H - V^2\right) - \rho\rho_T h_p - \rho_T\Theta\left(H - V^2\right) + \rho_T}{d_m}$$
$$= \frac{\rho_p\left(\rho h_T + \rho_T\left(H - V^2\right)\right) - \rho_T\left(\rho h_p + \Theta\left(H - V^2\right) - 1\right)}{d_m}$$

$$B = \frac{\left(\rho_p\rho_T - \rho_T\Theta\right)}{d_m} = \frac{\rho_T\left(\rho_p - \Theta\right)}{d_m}$$

$$C = \frac{\rho\rho_T h_p + \rho_T\rho_p H - \rho_T - \rho\Theta h_T - \rho_T\Theta H}{d_m}$$
$$= \frac{\rho_T\left(\rho h_p + \rho_p H - 1\right) - \Theta\left(\rho h_T + \rho_T H\right)}{d_m}$$

$$D = \frac{\left(\rho h_p + \rho_p H - 1\right)\left(\rho h_T + \rho_T\left(H - V^2\right)\right) - \left(\rho h_T + \rho_T H\right)\left(\rho h_p + \Theta\left(H - V^2\right) - 1\right)}{d_m}$$

$$d_m = \rho\Theta h_T + \rho_T\left(1 - \rho h_p\right)$$

©Zenotech Ltd

5.5 Geometric Multigrid

Geometric multigrid is used as a low frequency smoother for the explicit solver. This operates on a hierarchy of pre generated coarse meshes that have a fixed child to parent cell relationship. The algorithm is split into three stages:

1. Restrict from fine to coarse (subscript r)
2. Solve on coarse mesh (coarse cell subscript i_c)
3. Prolongate from coarse to fine (subscript p)

This is applied in a V cycle i.e. Stages 1 and 2 are applied recursively from fine to coarsest mesh before stage 3 is applied recursively from coarse to fine.

Residual Restrict Operator

$$R_{r,i_c} = \sum_{1}^{children} PR^* \left(W_i^k\right) \tag{5.29}$$

Solution Restrict Operator

$$W_{i_c}^k = \frac{1}{\Omega_{i_c}^{n+1}} \sum_{1}^{children} \Omega_i^{n+1} W_i^k \tag{5.30}$$

Coarse Mesh Explicit Solve

Define

$$R^c\left(W_{i_c}^{k,m}\right) = \left[R(W_{i_c}^{k,m}) + \frac{\left(3\Omega_{i_c}^{k+1} W_{i_c}^{k,m}\right)}{2\Delta t}\right] + \underbrace{\left[R_{r,i_c} - \left(R(W_{i_c}^k) + \frac{\left(3\Omega_{i_c}^{k+1} W_{i_c}^k\right)}{2\Delta t}\right)\right]}_{\text{Constant part of residual}} \tag{5.31}$$

The second term represents the error associated with solving on a coarser representation of the volume and is constant for all Runge Kutta stages.

$$W_{i_c}^{k,m=1} = W_{i_c}^k \tag{5.32}$$

$$W_{i_c}^{k,m=1} = W_{i_c}^{k,m=1} - \alpha_m P \frac{\Delta\tau_i}{\Omega_{i_c}^{k+1}} \left(R^c\left(W_{i_c}^{k,m}\right)\right) \tag{5.33}$$

$$\vdots$$

$$W_{i_c}^{k+1} = W_{i_c}^{k,m=5} \tag{5.34}$$

Coarse Residual Restrict Operator

$$R_{r,i_{c+1}} = \sum_{1}^{children} R^c\left(W_{i_c}^k\right) \tag{5.35}$$

5.6 Prolongate Operator

The prolongation operator applies the correction from coarse to fine mesh. The relaxed correction is directly applied to the fine mesh before the application of a smoother to remove any high frequency noise.

Correction

The correction is the difference between the currently computed solution and the restricted solution from the finer mesh

$$W_{p,i_{c-1}} = 0.75\left(W^k_{i_c} - W_{r,i_c}\right) \tag{5.36}$$

Laplace Smooth

The correction is then smoothed on the finer mesh using a Laplace smoother with $f = 0.5$

$$W_{p,i_{c-1}} = \frac{1}{area_{total}} \sum_{n=1}^{neighbours} area\left(fW_{p,i_{c-1}} + (1-f)\,W_{p,n_{c-1}}\right) \tag{5.37}$$

Update solution

The smoothed correction is then applied to the finer mesh solution recursively for all meshes from coarse to fine.

$$W^k_{i_{c-1}} = W^k_{i_{c-1}} + W_{p,i_{c-1}} \tag{5.38}$$

$$\vdots$$

$$W^k_i = W^k_i + W_{p,i} \tag{5.39}$$

5.7 ALE Preconditioned HLLC Flux

See [14]

The HLLC flux can be defined as

$$F_{HLLC} = \begin{cases} F_l, & \text{if } S_L > 0, \\ F(W^*_l), & \text{if } S_L \leq 0 < S_M, \\ F(W^*_r), & \text{if } S_M \leq 0 \leq S_R, \\ F_r, & \text{if } S_R < 0, \end{cases} \tag{5.40}$$

where

$$W^*_l = \begin{pmatrix} \rho^*_l \\ (\rho u)^*_l \\ (\rho v)^*_l \\ (\rho w)^*_l \\ (\rho E)^*_l \\ (\rho\phi)^*_l \end{pmatrix} = \frac{1}{S_L - S_M} \begin{pmatrix} \rho_l\,(S_L - q_l) \\ (S_L - q_l)\,(\rho u)_l + (p^* - p_l)\,n_x \\ (S_L - q_l)\,(\rho v)_l + (p^* - p_l)\,n_y \\ (S_L - q_l)\,(\rho w)_l + (p^* - p_l)\,n_z \\ (S_L - q_l)\,(\rho E)_l - p_l(V_l \cdot n) + p^* S_M \\ (\rho\phi)_l\,(S_L - q_l) \end{pmatrix} \tag{5.41}$$

©Zenotech Ltd

$$W_r^* = \begin{pmatrix} \rho_r^* \\ (\rho u)_r^* \\ (\rho v)_r^* \\ (\rho w)_r^* \\ (\rho E)_r^* \\ (\rho \phi)_r^* \end{pmatrix} = \frac{1}{S_R - S_M} \begin{pmatrix} \rho_r \left(S_R - q_r\right) \\ \left(S_R - q_r\right)(\rho u)_r + \left(p^* - p_r\right) n_x \\ \left(S_R - q_r\right)(\rho v)_r + \left(p^* - p_r\right) n_y \\ \left(S_R - q_r\right)(\rho w)_r + \left(p^* - p_r\right) n_z \\ \left(S_R - q_r\right)(\rho E)_r - p_r(V_r \cdot n) + p^* S_M \\ (\rho \phi)_r \left(S_R - q_r\right) \end{pmatrix} \tag{5.42}$$

$$F_l = \begin{pmatrix} \rho_l q_l \\ (\rho u)_l q_l + p_l n_x \\ (\rho v)_l q_l + p_l n_y \\ (\rho w)_l q_l + p_l n_z \\ (\rho E)_l q_l + p_l (V_l \cdot n) \\ (\rho \phi)_l q_l \end{pmatrix} \quad F_r = \begin{pmatrix} \rho_r q_r \\ (\rho u)_r q_r + p_r n_x \\ (\rho v)_r q_r + p_r n_y \\ (\rho w)_r q_r + p_r n_z \\ (\rho E)_r q_r + p_r (V_r \cdot n) \\ (\rho \phi)_l q_r \end{pmatrix} \tag{5.43}$$

$$F_l^* \equiv F(W_l^*) = F_l + S_L \left(W_l^* - W_l\right) - (\dot{x} \cdot n) W_l^* = \begin{pmatrix} \rho_l^* \left(S_M - \dot{x} \cdot n\right) \\ (\rho u)_l^* \left(S_M - \dot{x} \cdot n\right) + p^* n_x \\ (\rho v)_l^* \left(S_M - \dot{x} \cdot n\right) + p^* n_y \\ (\rho w)_l^* \left(S_M - \dot{x} \cdot n\right) + p^* n_z \\ (\rho E)_l^* \left(S_M - \dot{x} \cdot n\right) + p^* S_M \\ (\rho \phi)_l^* \left(S_M - \dot{x} \cdot n\right) \end{pmatrix} \tag{5.44}$$

$$F_r^* \equiv F(W_r^*) = F_r + S_R \left(W_r^* - W_r\right) - (\dot{x} \cdot n) W_r^* = \begin{pmatrix} \rho_r^* \left(S_M - \dot{x} \cdot n\right) \\ (\rho u)_r^* \left(S_M - \dot{x} \cdot n\right) + p^* n_x \\ (\rho v)_r^* \left(S_M - \dot{x} \cdot n\right) + p^* n_y \\ (\rho w)_r^* \left(S_M - \dot{x} \cdot n\right) + p^* n_z \\ (\rho E)_r^* \left(S_M - \dot{x} \cdot n\right) + p^* S_M \\ (\rho \phi)_r^* \left(S_M - \dot{x} \cdot n\right) \end{pmatrix} \tag{5.45}$$

$$p^* = \rho_l \left(q_l - S_L\right)\left(q_l - S_M\right) + p_l = \rho_r \left(q_r - S_R\right)\left(q_r - S_M\right) + p_r \tag{5.46}$$

$$q = (V - \dot{x}) \cdot n \text{ and } n = [n_x, n_y, n_z]^T \text{ is the unit normal to the interface} \tag{5.47}$$

with $\hat{V} = [\hat{u}, \hat{v}, \hat{w}]^T$ and $\hat{c}$ are Roe's average variables for velocity and speed of sound.

$$\hat{u} = \frac{(u_l + u_r R_p)}{(1 + R_p)} \tag{5.48}$$

$$\hat{v} = \frac{(v_l + v_r R_p)}{(1 + R_p)} \tag{5.49}$$

$$\hat{w} = \frac{(w_l + w_r R_p)}{(1 + R_p)} \tag{5.50}$$

$$\hat{c}^2 = (\gamma - 1)\left[\hat{H} - \frac{1}{2}\left(\hat{u}^2 + \hat{v}^2 + \hat{w}^2\right)\right] \tag{5.51}$$

$$\hat{H} = \frac{(H_l + H_r R_P)}{(1 + R_p)} \tag{5.52}$$

$$R_p = \sqrt{\frac{\rho_r}{\rho_l}} \tag{5.53}$$

$$S_M = \frac{\rho_r q_r \left(S_R - q_r\right) - \rho_l q_l \left(S_L - q_l\right) + p_l - p_r}{\rho_r \left(S_R - q_r\right) - \rho_l \left(S_L - q_l\right)} \tag{5.54}$$

The signal velocities for the un-preconditioned system are defined as

$$S_L = \min\left[q_l - c_l, \left(\hat{V}_l - \dot{x}\right)\cdot n - \hat{c}_l\right] \tag{5.55}$$

$$S_R = \max\left[q_r + c_r, \left(\hat{V}_r - \dot{x}\right)\cdot n + \hat{c}_r\right] \tag{5.56}$$

The signal velocities for the preconditioned system are defined as

$$S_L = \left[q_l^{'} - c_l^{'}\right] \tag{5.57}$$

$$S_R = \left[q_r^{'} + c_r^{'}\right] \tag{5.58}$$

Implicit HLLC Flux

$$F_{HLLC}^{n+1} = \begin{cases} F_l^n + \frac{\partial F_l}{\partial W_l}\delta W_l, & \text{if } S_L^n > 0, \\ (F_l^*)^n + \frac{\partial F_l^*}{\partial W_l}\delta W_l + \frac{\partial F_l^*}{\partial W_r}\delta W_r, & \text{if } S_L^n \le 0 < S_M^n, \\ (F_r^*)^n + \frac{\partial F_r^*}{\partial W_l}\delta W_l + \frac{\partial F_r^*}{\partial W_r}\delta W_r, & \text{if } S_M^n \le 0 \le S_R^n, \\ F_r^n + \frac{\partial F_r}{\partial W_r}\delta W_r, & \text{if } S_R^n < 0, \end{cases} \tag{5.59}$$

Assuming the conserved variables are independent define a new variable set.

$$W = [\rho, \rho u, \rho v, \rho w, \rho E, \rho\phi]^T = [x_1, x_2, x_3, x_4, x_5, x_6]^T \tag{5.60}$$

Substituting new variable set into equation for F

$$q_i = \left[\begin{pmatrix} \frac{x_2}{x_1} \\ \frac{x_3}{x_1} \\ \frac{x_4}{x_1} \end{pmatrix} - \dot{x}\right]\cdot n = \frac{1}{x_1}\left[\begin{pmatrix} x_2 \\ x_3 \\ x_4 \end{pmatrix} - x_1\dot{x}\right]\cdot n \tag{5.61}$$

$$\frac{\partial q_i}{\partial x_1} = \frac{1}{x_1}\left[-\dot{x}\cdot n - q_i\right] = -\frac{V\cdot n}{x_1} \tag{5.62}$$

$$\frac{\partial q_i}{\partial x_2} = \frac{n_x}{x_1} \tag{5.63}$$

$$\frac{\partial q_i}{\partial x_3} = \frac{n_y}{x_1} \tag{5.64}$$

$$\frac{\partial q_i}{\partial x_4} = \frac{n_z}{x_1} \tag{5.65}$$

$$F_i = \begin{pmatrix} x_1 q_i \\ x_2 q_i + (\gamma-1)\left(x_5 - \frac{1}{2}\frac{1}{x_1}\left(x_2^2 + x_3^2 + x_4^2\right)\right) n_x \\ x_3 q_i + (\gamma-1)\left(x_5 - \frac{1}{2}\frac{1}{x_1}\left(x_2^2 + x_3^2 + x_4^2\right)\right) n_y \\ x_4 q_i + (\gamma-1)\left(x_5 - \frac{1}{2}\frac{1}{x_1}\left(x_2^2 + x_3^2 + x_4^2\right)\right) n_z \\ x_5 q_i + q_i(\gamma-1)\left(x_5 - \frac{1}{2}\frac{1}{x_1}\left(x_2^2 + x_3^2 + x_4^2\right)\right) \\ x_6 q_i \end{pmatrix} \tag{5.66}$$

©Zenotech Ltd

$$\frac{\partial F}{\partial x_1} = \begin{pmatrix} -\dot{x}\cdot n \\ x_2\frac{-V\cdot n}{x_1} + \theta n_x \\ x_3\frac{-V\cdot n}{x_1} + \theta n_y \\ x_4\frac{-V\cdot n}{x_1} + \theta n_z \\ \frac{-V\cdot n}{1}\left(\gamma\frac{x_5}{x_1} - \theta\right) + q_i\theta \\ x_6\frac{-V\cdot n}{x_1} \end{pmatrix} \tag{5.67}$$

$$\frac{\partial F}{\partial x_2} = \begin{pmatrix} n_x \\ q_i + \frac{x_2}{x_1}(2-\gamma)n_x \\ x_3\frac{n_x}{x_1} - (\gamma-1)\frac{x_2}{x_1}n_y \\ x_4\frac{n_x}{x_1} - (\gamma-1)\frac{x_2}{x_1}n_z \\ \gamma\frac{x_5}{x_1}n_x - \theta n_x - q_i(\gamma-1)\frac{x_2}{x_1} \\ x_6\frac{n_x}{x_1} \end{pmatrix} \tag{5.68}$$

$$\frac{\partial F}{\partial x_3} = \begin{pmatrix} n_y \\ x_2\frac{n_y}{x_1} - (\gamma-1)\frac{x_3}{x_1}n_x \\ q_i + \frac{x_3}{x_1}(2-\gamma)n_y \\ x_4\frac{n_y}{x_1} - (\gamma-1)\frac{x_3}{x_1}n_z \\ \gamma\frac{x_5}{x_1}n_y - \theta n_y - q_i(\gamma-1)\frac{x_3}{x_1} \\ x_6\frac{n_y}{x_1} \end{pmatrix} \tag{5.69}$$

$$\frac{\partial F}{\partial x_4} = \begin{pmatrix} n_z \\ x_2\frac{n_z}{x_1} - (\gamma-1)\frac{x_4}{x_1}n_x \\ x_3\frac{n_z}{x_1} - (\gamma-1)\frac{x_4}{x_1}n_y \\ q_i + \frac{x_4}{x_1}(2-\gamma)n_z \\ \gamma\frac{x_5}{x_1}n_z - \theta n_z - q_i(\gamma-1)\frac{x_4}{x_1} \\ x_6\frac{n_z}{x_1} \end{pmatrix} \tag{5.70}$$

$$\frac{\partial F}{\partial x_5} = \begin{pmatrix} 0 \\ (\gamma-1)n_x \\ (\gamma-1)n_y \\ (\gamma-1)n_z \\ \gamma q_i \\ 0 \end{pmatrix} \tag{5.71}$$

$$\frac{\partial F}{\partial x_6} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ q_i \end{pmatrix} \tag{5.72}$$

where $\theta = (\gamma-1)\left(\frac{1}{2}\frac{1}{x_1^2}\left(x_2^2+x_3^2+x_4^2\right)\right) = \frac{(\gamma-1)}{2}V^2$

Transforming back and building jacobian matrix

$$\frac{\partial F}{\partial W} = \begin{pmatrix} -\dot{x}\cdot n & n_x & n_y & n_z & 0 & 0 \\ \frac{(\gamma-1)}{2}V^2 n_x - u(V\cdot n) & q - a_3 u n_x & u n_y - a_2 v n_x & u n_z - a_2 w n_x & a_2 n_x & 0 \\ \frac{(\gamma-1)}{2}V^2 n_y - v(V\cdot n) & v n_x - a_2 u n_y & q - a_3 v n_y & v n_z - a_2 w n_y & a_2 n_y & 0 \\ \frac{(\gamma-1)}{2}V^2 n_z - w(V\cdot n) & w n_x - a_2 u n_z & w n_y - a_2 v n_z & q - a_3 w n_z & a_2 n_z & 0 \\ \frac{(\gamma-1)}{2}V^2 q - a_1(V\cdot n) & a_1 n_x - a_2 u q & a_1 n_y - a_2 v q & a_1 n_z - a_2 w q & \gamma q & 0 \\ -\phi(V\cdot n) & \phi n_x & \phi n_y & \phi n_z & 0 & q \end{pmatrix} \tag{5.73}$$

where $a_1 = \gamma E - \frac{(\gamma-1)}{2}V^2$, $a_2 = \gamma - 1$, $a_3 = \gamma - 2$

$$\frac{\partial F_l^*}{\partial W_l} = \begin{bmatrix} \left(\frac{\partial \rho_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T \rho_l^* \\ \left(\frac{\partial (\rho u)_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T (\rho u)_l^* + \left(\frac{\partial p^*}{\partial W_l}\right)^T n_x \\ \left(\frac{\partial (\rho v)_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T (\rho v)_l^* + \left(\frac{\partial p^*}{\partial W_l}\right)^T n_y \\ \left(\frac{\partial (\rho w)_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T (\rho w)_l^* + \left(\frac{\partial p^*}{\partial W_l}\right)^T n_z \\ \left(\frac{\partial (\rho E)_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T (\rho E)_l^* + \left(\frac{\partial p^*}{\partial W_l}\right)^T S_M + \left(\frac{\partial S_M}{\partial W_l}\right)^T p^* \\ \left(\frac{\partial (\rho \phi)_l^*}{\partial W_l}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_l}\right)^T (\rho \phi)_l^* \end{bmatrix} \tag{5.74}$$

$$\frac{\partial F_l^*}{\partial W_r} = \begin{bmatrix} \left(\frac{\partial \rho_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T \rho_l^* \\ \left(\frac{\partial (\rho u)_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T (\rho u)_l^* + \left(\frac{\partial p^*}{\partial W_r}\right)^T n_x \\ \left(\frac{\partial (\rho v)_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T (\rho v)_l^* + \left(\frac{\partial p^*}{\partial W_r}\right)^T n_y \\ \left(\frac{\partial (\rho w)_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T (\rho w)_l^* + \left(\frac{\partial p^*}{\partial W_r}\right)^T n_z \\ \left(\frac{\partial (\rho E)_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T (\rho E)_l^* + \left(\frac{\partial p^*}{\partial W_r}\right)^T S_M + \left(\frac{\partial S_M}{\partial W_r}\right)^T p^* \\ \left(\frac{\partial (\rho \phi)_l^*}{\partial W_r}\right)^T (S_M - \dot{x}\cdot n) + \left(\frac{\partial S_M}{\partial W_r}\right)^T (\rho \phi)_l^* \end{bmatrix} \tag{5.75}$$

Using the quotient rule and defining $S_M = \frac{B}{A}$ therefore $\frac{\partial S_M}{\partial x} = \frac{A\frac{\partial B}{\partial x} - B\frac{\partial A}{\partial x}}{A^2}$

where $A = \rho_r (S_R - q_r) - \rho_l (S_L - q_l)$ and $B = \rho_r q_r (S_R - q_r) - \rho_l q_l (S_L - q_l) + p_l - p_r$

Expressing S_M in terms of independent variables

$$\frac{\partial S_M}{\partial \rho_l} = \frac{1}{A}\left[-q_l^2 + \frac{\partial p_l}{\partial \rho_l} - \frac{B}{A}(-q_l - (S_L - q_l))\right] \tag{5.76}$$

$$= \frac{1}{A}\left[-q_l^2 + \frac{(\gamma-1)}{2}V^2 + S_M S_L\right] \tag{5.77}$$

$$\frac{\partial S_M}{\partial (\rho u)} = \frac{1}{A}\left[n_x(2q_l - S_L - S_M) + (1-\gamma)u_l\right] \tag{5.78}$$

$$\frac{\partial S_M}{\partial (\rho v)} = \frac{1}{A}\left[n_y(2q_l - S_L - S_M) + (1-\gamma)v_l\right] \tag{5.79}$$

©Zenotech Ltd

$$\frac{\partial S_M}{\partial(\rho w)} = \frac{1}{A}\left[n_z(2q_l - S_L - S_M) + (1-\gamma)w_l\right] \tag{5.80}$$

$$\frac{\partial S_M}{\partial(\rho E)} = \frac{1}{A}\left[\gamma - 1\right] \tag{5.81}$$

$$\frac{\partial S_M}{\partial(\rho \phi)} = \frac{1}{A}\left[0\right] \tag{5.82}$$

$$\frac{\partial S_M}{\partial W_l} = \frac{1}{A}\begin{bmatrix} -q_l^2 + \frac{(\gamma-1)}{2}V^2 + S_M S_L \\ n_x(2q_l - S_L - S_M) + (1-\gamma)u_l \\ n_y(2q_l - S_L - S_M) + (1-\gamma)v_l \\ n_z(2q_l - S_L - S_M) + (1-\gamma)w_l \\ \gamma - 1 \\ 0 \end{bmatrix} \tag{5.83}$$

$$\frac{\partial S_M}{\partial W_r} = \frac{1}{A}\begin{bmatrix} q_r^2 - \frac{(\gamma-1)}{2}V^2 - S_M S_R \\ n_x(-2q_r + S_R + S_M) - (1-\gamma)u_r \\ n_y(-2q_r + S_R + S_M) - (1-\gamma)v_r \\ n_z(-2q_r + S_R + S_M) - (1-\gamma)w_r \\ 1 - \gamma \\ 0 \end{bmatrix} \tag{5.84}$$

$$\frac{\partial p^*}{\partial W_l} = \rho_r\left(S_R - q_r\right)\frac{\partial S_M}{\partial W_l} \tag{5.85}$$

$$\frac{\partial p^*}{\partial W_r} = \rho_l\left(S_l - q_l\right)\frac{\partial S_M}{\partial W_r} \tag{5.86}$$

$$\frac{\partial \rho_l^*}{\partial W_l} = \frac{1}{S_L - S_M}\begin{bmatrix} S_L \\ -n_x \\ -n_y \\ -n_z \\ 0 \\ 0 \end{bmatrix} + \frac{1}{S_L - S_M}\rho_l^*\frac{\partial S_M}{\partial W_l} \tag{5.87}$$

$$\frac{\partial \rho_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\rho_l^*\frac{\partial S_M}{\partial W_r} \tag{5.88}$$

$$\frac{\partial(\rho u)_l^*}{\partial W_l} = \frac{1}{S_L - S_M}\begin{bmatrix} q_l u_l - n_x\frac{(\gamma-1}{2}V^2 \\ S_L - q_l + n_x(\gamma-2)u_l \\ -u_l n_y + n_x(\gamma-1)v_l \\ -u_l n_z + n_x(\gamma-1)w_l \\ (1-\gamma)n_x \\ 0 \end{bmatrix} + \frac{1}{S_L - S_M}\left(n_x\frac{\partial p^*}{\partial W_l} + (\rho u)_l^*\frac{\partial S_M}{\partial W_l}\right) \tag{5.89}$$

$$\frac{\partial(\rho u)_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\left(n_x\frac{\partial p^*}{\partial W_r} + (\rho u)_l^*\frac{\partial S_M}{\partial W_r}\right) \tag{5.90}$$

$$\frac{\partial(\rho v)_l^*}{\partial W_l} = \frac{1}{S_L - S_M}\begin{bmatrix} q_l v_l - n_y\frac{(\gamma-1}{2}V^2 \\ -v_l n_x + n_y(\gamma-1)u_l \\ S_L - q_l + n_y(\gamma-2)v_l \\ -v_l n_z + n_y(\gamma-1)w_l \\ (1-\gamma)n_y \\ 0 \end{bmatrix} + \frac{1}{S_L - S_M}\left(n_y\frac{\partial p^*}{\partial W_l} + (\rho v)_l^*\frac{\partial S_M}{\partial W_l}\right) \tag{5.91}$$

$$\frac{\partial(\rho v)_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\left(n_y \frac{\partial p^*}{\partial W_r} + (\rho v)_l^* \frac{\partial S_M}{\partial W_r}\right) \tag{5.92}$$

$$\frac{\partial(\rho w)_l^*}{\partial W_l} = \frac{1}{S_L - S_M}\begin{bmatrix} q_l w_l - n_z \frac{(\gamma-1}{2} V^2 \\ -w_l n_x + n_z(\gamma-1)u_l \\ -w_l n_y + n_z(\gamma-1)v_l \\ S_L - q_l + n_z(\gamma-2)w_l \\ (1-\gamma)n_z \\ 0 \end{bmatrix} + \frac{1}{S_L - S_M}\left(n_z \frac{\partial p^*}{\partial W_l} + (\rho w)_l^* \frac{\partial S_M}{\partial W_l}\right) \tag{5.93}$$

$$\frac{\partial(\rho w)_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\left(n_z \frac{\partial p^*}{\partial W_r} + (\rho w)_l^* \frac{\partial S_M}{\partial W_r}\right) \tag{5.94}$$

$$\begin{aligned}\frac{\partial(\rho E)_l^*}{\partial W_l} =& \frac{1}{S_L - S_M}\begin{bmatrix} ((\rho E)_l + p_l)\frac{q_l}{\rho_l} - q_l \frac{(\gamma-1)}{2} V^2 \\ -n_x((\rho E)_l + p_l)/\rho_l + (\gamma-1)u_l q_l \\ -n_y((\rho E)_l + p_l)/\rho_l + (\gamma-1)v_l q_l \\ -n_z((\rho E)_l + p_l)/\rho_l + (\gamma-1)w_l q_l \\ S_L - q_l\gamma \\ 0 \end{bmatrix} \\ &+ \frac{1}{S_L - S_M}\left(S_M \frac{\partial p^*}{\partial W_l} + (p^* + (\rho E)_l^*)\frac{\partial S_M}{\partial W_l}\right)\end{aligned} \tag{5.95}$$

$$\frac{\partial(\rho E)_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\left(S_M \frac{\partial p^*}{\partial W_r} + (p^* + (\rho E)_l^*)\frac{\partial S_M}{\partial W_r}\right) \tag{5.96}$$

$$\frac{\partial(\rho\phi)_l^*}{\partial W_l} = \frac{1}{S_L - S_M}\begin{bmatrix} \phi q_l \\ -\phi n_x \\ -\phi n_y \\ -\phi n_z \\ 0 \\ S_L - q_l \end{bmatrix} + \frac{1}{S_L - S_M}\left((\rho\phi)_l^* \frac{\partial S_M}{\partial W_l}\right) \tag{5.97}$$

$$\frac{\partial(\rho\phi)_l^*}{\partial W_r} = \frac{1}{S_L - S_M}\left((\rho\phi)_l^* \frac{\partial S_M}{\partial W_r}\right) \tag{5.98}$$

5.8 Gradients

This is based on the Green Gauss theorem and, hence by assuming ϕ is piecewise constant over the volume

$$\Omega_i \nabla\phi_i = \sum_{faces} \phi_f A_f \tag{5.99}$$

$$\nabla\phi_i = \frac{1}{\Omega_i}\sum_{faces} \phi_f A_f \tag{5.100}$$

Face Gradient

Simplest gradient estimate at the face is an average

$$\overline{\nabla W_{ij}} = \frac{1}{2}\left(\nabla W_i + \nabla W_j\right) \tag{5.101}$$

©Zenotech Ltd

The average gradient results in odd even de-coupling of the solution, therefore we need a more accurate estimate. We can replace the normal gradient at the face calculated by the average with the accurate face normal gradient. Therefore the most accurate approximation of the gradient normal to a face is

$$\nabla W_{ij} \cdot n \approx \frac{W_j - W_i}{\|x_j - x_i\|} \tag{5.102}$$

where x_i is the cell centre of cell i

An improved estimate of the gradient at the face is

$$\nabla W_{ij} = \frac{W_j - W_i}{\|x_j - x_i\|}\Delta x + \left[\overline{\nabla W_{ij}} - \left(\overline{\nabla W_{ij}} \cdot \Delta x\right)\Delta x\right] \tag{5.103}$$

where $\Delta x = (x_j - x_i)/\|x_j - x_i\|$

To reduce the stencil size for the viscous flux jacobians used in the implicit scheme we neglect the face tangential gradient component defining

$$\nabla W_{ij}^{TSL} \cdot n \approx \frac{W_j - W_i}{\|x_j - x_i\|} \tag{5.104}$$

$$\nabla W_{ij}^{TSL} \approx \frac{W_j - W_i}{\|x_j - x_i\|} \cdot \Delta x \tag{5.105}$$

Where TSL stands for Thin Shear Layer.

5.9 MUSCL Reconstruction

Monotone Upstream-centered Schemes for Conservation Laws (MUSCL) provides Total Variation Diminishing (TVD) behaviour when reconstructing the states at cell boundaries from piecewise constant values at the cell enabling the higher spatial accuracy to be achieved. See [10]

Consider the scalar quantity ϕ either side of an interface between cells i and j respectively.

First Order

First order spatial accuracy is achieved by setting the reconstructed states to the cell values.

$$\phi_i^+ = \phi_i \tag{5.106}$$

$$\phi_j^- = \phi_j \tag{5.107}$$

Higher Order

The upwind biased interpolations for ϕ_i^+ and ϕ_j^- are defined by:

$$\phi_i^+ = \phi_i + \frac{s_i}{4}\left[(1 - k_c s_i)\,\Delta_i^- + (1 + k_c s_i)\,(\phi_j - \phi_i)\right] \tag{5.108}$$

$$\phi_j^- = \phi_j + \frac{s_j}{4}\left[(1 - k_c s_j)\,\Delta_j^+ + (1 + k_c s_j)\,(\phi_j - \phi_i)\right] \tag{5.109}$$

where the forward and backward difference operators are given by:

$$\Delta_i^- = \phi_i - \phi_{i-1} = 2\nabla\phi_i \cdot l_{ij} - (\phi_j - \phi_i) = 2\nabla\phi_i \cdot 2\,(x_f - x_i) - (\phi_j - \phi_i) \tag{5.110}$$

$$\Delta_j^+ = \phi_{j+1} - \phi_j = 2\nabla\phi_j \cdot l_{ij} - (\phi_j - \phi_i) = 2\nabla\phi_j \cdot 2\,(x_j - x_f) - (\phi_j - \phi_i) \tag{5.111}$$

where $l_{ij} = x_j - x_i \approx 2\,(x_f - x_i) \approx 2\,(x_j - x_f)$ and s_i is a gradient or slope limiter

Parameter k_c provides control over the family of schemes

Scheme	k_e
Upwind	-1
Semi upwind (Fromm's scheme)	0
Third order upwind	1/3
Central difference	1

Van Albada Limiter

This gives improved estimates for s_i and s_j.

$$s_i = \max\left[0, \frac{2\Delta_i^-\,(\phi_j - \phi_i) + \epsilon^2}{\left(\Delta_i^-\right)^2 + (\phi_j - \phi_i)^2 + \epsilon^2}\right] \tag{5.112}$$

$$s_j = \max\left[0, \frac{2\Delta_j^+\,(\phi_j - \phi_i) + \epsilon^2}{\left(\Delta_j^+\right)^2 + (\phi_j - \phi_i)^2 + \epsilon^2}\right] \tag{5.113}$$

$$\epsilon = k_1 \frac{1}{2}\,(\phi_j + \phi_i) + k_2 \tag{5.114}$$

$$k_1 = 0.001\; k_2 = 1.0e^{-8} \tag{5.115}$$

Writing $a = \phi_j - \phi_i$ for the face jump and b for the reconstructed backward difference (Δ_i^- above), the Van Albada limiter is the symmetric function

$$\psi_{\mathrm{VA}}\,(a, b) = \max\left[0, \frac{2ab + \epsilon^2}{a^2 + b^2 + \epsilon^2}\right] \tag{5.116}$$

and the remaining selectable limiters share the same arguments and the same ϵ.

Van Leer Limiter

$$\psi_{\mathrm{VL}}\,(a, b) = \max\left[0, \frac{4ab + \epsilon^2}{(a + b)^2 + \epsilon^2}\right] \tag{5.117}$$

Slightly more compressive than Van Albada. Like Van Albada its denominator is a sum of squares and is therefore strictly positive, and its ϵ^2 cancels on a linear field so that $\psi = 1$ exactly when $a = b$.

©Zenotech Ltd

Symmetric Ratio Limiter

Selected by the configuration value `symmetric ratio`. This was previously called `barth jespersen`, which was a misnomer: the function takes no minimum or maximum over the neighbours of a cell and is therefore not the Barth–Jespersen limiter. That name now selects the real limiter, described in the following section.

$$\sigma = \operatorname{sign}(a+b) \tag{5.118}$$

$$\psi_{\mathrm{SR}}(a,b) = \max\left[0, \min\left(1, \frac{4\min(\sigma a, \sigma b) + \epsilon}{|a+b| + \epsilon}\right)\right] \tag{5.119}$$

On a linear field $a = b = \delta$ this reduces to $(4|\delta| + \epsilon)/(2|\delta| + \epsilon)$, which exceeds unity at every magnitude and is clipped to exactly 1, so a linear field is reconstructed unlimited and the limiter is invariant under $\phi \to -\phi$. The denominator satisfies $|a+b| + \epsilon \geq k_2 > 0$ and so cannot vanish.

Earlier releases regularised this limiter with ϵ^2 added to $4b$, $4a$ and $a+b$. Since ϵ^2 carries units of the variable squared while those terms are linear in the variable, and since the numerator carries 4δ against the denominator's 2δ, the regularisation could not cancel on a linear field. The limiter consequently returned $\psi < 1$ for a smooth decreasing gradient – reaching $\psi = 0$ at $\delta = -\epsilon^2/4$ – was not invariant under $\phi \to -\phi$, and where ϵ^2 became comparable to the local slopes it inverted the sign of the denominator and collapsed to first order in smooth flow.

No Limiter

Setting the limiter to `none` gives $s_i = s_j = 1$, i.e. an unlimited reconstruction. This retains formal second order everywhere but is not monotonicity preserving.

Cell Based Multidimensional Limiters

The limiters above are one-dimensional functions of two directional differences and do not bound the reconstructed face value by the range of values over the cell's neighbours. The limiters in this section do. For cell i, with $u_{\max}$ and $u_{\min}$ taken over the face neighbours and $\Delta_f = \nabla u_i \cdot (x_f - x_i)$ the unlimited extrapolation to face f,

$$y_f = \begin{cases} (u_{\max} - u_i)/\Delta_f & \Delta_f > 0 \\ (u_{\min} - u_i)/\Delta_f & \Delta_f < 0 \\ \infty & \Delta_f = 0 \end{cases} \qquad \phi_i = \min_f P(y_f) \tag{5.120}$$

so that $y_f \geq 0$ always. The limiter scales the gradient term of the reconstruction only; because the κ blend is a convex combination of $u_i + \phi_i \Delta_f$ and $\frac{1}{2}(u_i + u_j)$, both of which lie inside the neighbourhood bound, the bound survives the blend and the face difference term needs no limiting.

Barth–Jespersen

Selected by `barth jespersen`.

$$P(y) = \min(1, y) \tag{5.121}$$

Bound preserving by inspection, but only C^0: the kink at $y = 1$ makes the limiter value flicker between iterations and can stall the residual. As published it carries no smoothness deactivation, so it limits wherever the bound binds, including in smooth flow.

Michalak–Ollivier-Gooch

Selected by `michalak gooch`. The unique cubic with $P(0) = 0$, $P'(0) = 1$, $P(y_t) = 1$ and $P'(y_t) = 0$ is $P = y + by^2 + ay^3$ with $a = (y_t - 2)/y_t^3$ and $b = (3 - 2y_t)/y_t^2$. At $y_t = 3/2$ the quadratic coefficient vanishes and

$$P(y) = \begin{cases} y - \frac{4}{27}y^3 & y < 3/2 \\ 1 & y \geq 3/2 \end{cases} \tag{5.122}$$

which is C^1 and satisfies $P(y) \leq \min(1, y)$, so it is never less monotonicity preserving than Barth–Jespersen. The price of differentiability is $P(1) = 1 - 4/27 = 0.852$ rather than 1.

Limiting is deactivated in smooth flow by a grid scaled threshold $\epsilon^2 = (Kh/L_{ref})^3$, blended with a C^1 smoothstep, where h is the cell length scale and K is the configuration value `limiter coefficient`. The cube falls faster than the $O(h^2)$ reconstruction error, so under refinement the limiter switches itself off where the flow is smooth and the formal order of the scheme survives. The reference length is required because the solution is non-dimensional while the mesh is not.

Venkatakrishnan

Selected by `venkatakrishnan`.

$$P(y) = \min\left(1, \frac{y^2 + 2y}{y^2 + y + 2}\right) \tag{5.123}$$

Smooth, but the unclipped function exceeds unity for $y > 2$, so the clip is required and reintroduces a C^0 kink at $y = 2$.

5.10 Time Step

Local time step

The local time step is computed for each cell in accordance with

$$\Delta t = \min\left[CFL\frac{\Omega}{\lambda}, CFLVIS\frac{\Omega}{\lambda_v}\right] \tag{5.124}$$

where λ is the integrated convective spectral radius, λ_v is the corresponding viscous spectral radius and the parameters CFL and CFLVIS are user input and chosen according to the stability region of the Runge Kutta scheme used.

The time step for the turbulence equations is determined as follows

$$\Delta t_t = \Delta t - \min(0, \min(k_{source}, \omega_{source})) \tag{5.125}$$

Convective spectral radius

$$\lambda = |V \cdot n| \pm c \tag{5.126}$$

 ©Zenotech Ltd

Viscous spectral radius

$$\lambda_v = \pm \frac{|S|}{r} \left[\max\left(\frac{4}{3\rho}, \frac{\gamma}{\rho}\right) \left(\frac{\mu}{\mathrm{Pr}} + \frac{\mu_t}{Pr_t}\right)\right] \tag{5.127}$$

5.11 Boundary Conditions

Open boundary conditions

The Riemann invariants for the Euler equations are:

$$W = \begin{pmatrix} \rho - \frac{p}{\bar{c}^2} \\ u_t \\ u_n + \frac{p}{\overline{\rho c}} \\ -u_n + \frac{p}{\overline{\rho c}} \end{pmatrix} \tag{5.128}$$

Where u_n is the normal velocity and u_t is the tangential velocity given an inwards normal at the boundary points. The Riemann invariants are defined only locally in time for systems in general. The idea relies on restricting the variation of the Riemann invariant to zero on a subsonic out flow for the invariants associated with an ingoing waves. In this case, only the third Riemann invariant needs to be imposed. All other values are taken as predicted states computed by the scheme inside the volume.Predicted states are noted with a star and bar states are averages between predicted states and last time step values. This may be written:

$$\begin{pmatrix} \rho - \frac{p^*}{\bar{c}^2} \\ u_t \\ u_n + \frac{p}{\overline{\rho c}} \\ -u_n + \frac{p}{\overline{\rho c}} \end{pmatrix} = \begin{pmatrix} \rho^* - \frac{p^*}{\bar{c}^2} \\ u_t^* \\ u_n^n + \frac{p^n}{\overline{\rho c}} \\ -u_n^* + \frac{p^n}{\overline{\rho c}} \end{pmatrix} \tag{5.129}$$

After some algebraic manipulations the method consists in setting:

$$u_t = u_t^* \tag{5.130}$$

$$p = \frac{1}{2}(p^n + p^* + \overline{\rho c}(u_n^n - u_n^*) \tag{5.131}$$

$$\rho = \rho^* + \frac{(p - p^*)}{\bar{c}^2} \tag{5.132}$$

$$u_n = u_n^n + \frac{p^n - p}{\overline{\rho c}} \tag{5.133}$$

As these conditions are rather crude, the order of the scheme is locally lowered to simulate a sponge layernear the open boundary by adding more diffusion. In an unstructured context, structured layers may besimulated by marching from the boundary and marking the edges that touch the boundary points, andmarking the points neighbors of boundary points. At each pass, a new layer of edges is added to the markededges. For more information see [16].

Euler Wall and Symmetry Plane

At an Euler wall the normal component of the velocity is zero

$$V \cdot n = 0 \tag{5.134}$$

and hence the inviscid wall flux is strongly enforced as

$$f = \begin{pmatrix} 0 \\ pn_x A \\ pn_y A \\ pn_z A \\ 0 \end{pmatrix} \tag{5.135}$$

Viscous Wall (Adiabatic and Isothermal)

On a viscous wall the velocity is imposed strongly through a no-slip boundary condition

$$V = 0 \tag{5.136}$$

see section 5.3 for more details of gradient boundary conditions at the wall.

Mass flow outlet

This boundary condition can be prescribed at a outflow boundary which requires a given value of mass flow. This boundary condition is in principal identical to the static pressure inflow boundary condition where the value of static pressure is a function of mass. The most useful description is the specification of a non-dimensional mass flow function which is defined as

$$W^* = \frac{\dot{m}\sqrt{C_p T_0}}{p_0} \tag{5.137}$$

then rearrange to find $\dot{m}$, the mass flow rate. Then solve the following implicit equation for velocity u

$$\dot{m} = \rho_0 u \left(1 - \frac{\gamma - 1}{2c^2} u^2\right)^{\frac{1}{\gamma - 1}} A \tag{5.138}$$

then use this to calculate the value of the Laval number

$$\lambda_{lav} = \frac{u}{c\sqrt{\frac{2}{\gamma+1}}} \tag{5.139}$$

the value of λ_{lav} is then used to determine the required value of static pressure on the outflow boundary

$$p_{static} = p_0 \left(1 - \frac{\gamma - 1}{\gamma + 1} \lambda_{lav}^2\right)^{\frac{\gamma}{\gamma - 1}} \tag{5.140}$$

Farfield

Characteristic boundary conditions are used at the farfield. These boundary conditions can be used for both subsonic and supersonic in and outflow where the characteristics are either set from free stream quantities for ingoing characteristics or extrapolated from the interior for outgoing characteristics. Conserved variables are used and stored in the program since they lead to sparse and computationally less expensive expressions, using a similar procedure to the open boundary conditions described at the start of this section.

The condition is applied by constructing a boundary state and storing it in the halo cell adjacent to the boundary face; the boundary flux is then formed from the average of the

 ©Zenotech Ltd

interior and halo states. Throughout this section n is the *outward* face normal, subscript i denotes the interior state, subscript e the prescribed exterior (free stream) state and subscript b the boundary state being constructed. The face normal velocities are

$$q_i = V_i \cdot n - V_{face} \cdot n \qquad q_e = V_e \cdot n - V_{face} \cdot n \tag{5.141}$$

where V_{face} is the mesh velocity, so that q is measured relative to a moving face.

If the flow is supersonic ($|q_i| \geq c_i$) every characteristic travels the same way and the boundary state is simply the free stream for inflow or the interior state for outflow. The subsonic case is described below.

Subsonic farfield

Without preconditioning the Riemann invariants of the one dimensional Euler equations along n are

$$R^{\pm} = q \pm \frac{2c}{\gamma - 1} \tag{5.142}$$

For subsonic flow R^+ leaves the domain and is taken from the interior while R^- enters and is taken from the exterior, giving

$$q_b = \frac{1}{2}(q_i + q_e) + \frac{c_i - c_e}{\gamma - 1} \tag{5.143}$$

$$c_b = \frac{1}{4}(\gamma - 1)(q_i - q_e) + \frac{1}{2}(c_i + c_e) \tag{5.144}$$

The remaining variables are carried on the entropy and shear waves, which travel at q, and are therefore taken from whichever side is upstream of the boundary. Writing u for that upwind state ($u = e$ when $q_b < 0$, $u = i$ otherwise)

$$V_b = V_u + (q_b - q_u)\, n \tag{5.145}$$

$$\left(\frac{p}{\rho}\right)_b = \frac{c_b^2}{\gamma} \qquad \rho_b = \left(\left(\frac{p}{\rho}\right)_b \frac{\rho_u^{\gamma}}{p_u}\right)^{\frac{1}{\gamma-1}} \tag{5.146}$$

$$E_b = \frac{1}{\gamma - 1}\left(\frac{p}{\rho}\right)_b + \frac{1}{2} V_b \cdot V_b + k \tag{5.147}$$

For viscous flow a subsonic outflow is instead treated as a pressure outflow, with the pressure taken from the reference condition and the density and velocity extrapolated from the interior.

Preconditioned subsonic farfield

When the low Mach number preconditioner of section 5.4 is active the invariants above are no longer the characteristic variables of the system being solved, and using them leaves the farfield responding at the acoustic speed that the preconditioner has deliberately removed from the interior scheme. The preconditioned characteristics are used instead [19].

The preconditioned normal flux Jacobian has eigenvalues q, carrying the entropy and shear waves as before, and the acoustic pair $q' \pm c'$ defined there. Writing

$$\alpha = \frac{1}{2}\left(1 - \frac{U_r^2}{c_0^2}\right) \qquad c' = \sqrt{\alpha^2 q_0^2 + U_r^2} \tag{5.148}$$

and linearising about the face averaged state $\rho_0 = \frac{1}{2}(\rho_i + \rho_e)$, $q_0 = \frac{1}{2}(q_i + q_e)$, $c_0 = \frac{1}{2}(c_i + c_e)$ with U_r averaged over the two sides, the acoustic characteristic variables are

$$W^{\pm} = p + \rho_0 \left(\alpha q_0 \pm c^{'} \right) q \tag{5.149}$$

carried at $q^{'} \pm c^{'}$. Taking W^{+} from the interior and W^{-} from the exterior and solving the resulting 2×2 system gives the boundary pressure and normal velocity directly

$$q_b = \frac{1}{2}(q_i + q_e) + \frac{p_i - p_e}{2\rho_0 c^{'}} + \frac{\alpha q_0}{2c^{'}}(q_i - q_e) \tag{5.150}$$

$$p_b = \frac{1}{2}(p_i + p_e) + \frac{\alpha q_0}{2c^{'}}(p_e - p_i) + \frac{\rho_0 U_r^2}{2c^{'}}(q_i - q_e) \tag{5.151}$$

The velocity and energy then follow as above, with the density obtained from the upwind entropy using the boundary pressure

$$\rho_b = \left(p_b \frac{\rho_u^{\gamma}}{p_u} \right)^{\frac{1}{\gamma}} \tag{5.152}$$

Because the relations are linearised, the boundary pressure is limited below by $0.1\, p_e$ so that a large transient cannot drive it to zero or below.

Two properties of these relations are worth noting. Setting $U_r = c_0$ switches the preconditioner off, giving $\alpha = 0$ and $c^{'} = c_0$, and the pair reduces to

$$q_b = \frac{1}{2}(q_i + q_e) + \frac{p_i - p_e}{2\rho_0 c_0} \qquad p_b = \frac{1}{2}(p_i + p_e) + \frac{1}{2}\rho_0 c_0 (q_i - q_e)$$

which is the linearised form of the unpreconditioned invariants and has the same structure as the open boundary condition given at the start of this section, which is written with an inward normal and so carries the opposite signs. Secondly

$$q^{'} - c^{'} > 0 \iff q_0 > 0 \text{ and } q_0^2 (1 - 2\alpha) > U_r^2 \iff q_0^2 \frac{U_r^2}{c_0^2} > U_r^2 \text{ with } q_0 > 0 \iff q_0 > c_0$$

since $1 - \alpha > 0$, and in the same way $q^{'} + c^{'} < 0 \iff q_0 < -c_0$. The acoustic eigenvalues therefore change sign at exactly the same places whatever the value of U_r, and preconditioning does not move the subsonic to supersonic switch.

The practical effect is carried by the acoustic impedance $\rho_0 c^{'}$. At low Mach number $U_r \to |V|$ and so $c^{'} \to U_r$, making the boundary pressure response to a normal velocity mismatch smaller by a factor of order M than the unpreconditioned relations would give, which is what the preconditioned interior scheme expects.

Total States Inlet

This condition concerns the inflow into the domain of interest. Assuming that at the boundary

$$p_s = p_f \tag{5.153}$$

©Zenotech Ltd

where p_f is the static pressure at the boundary. Then given the reference static temperature T_{s_∞} and the total to static ratios $\frac{T_o}{T_{s_\infty}}$, $\frac{p_o}{p_{s_\infty}}$, the total temperature T_o can then be calculated using

$$\frac{T_o}{T_s} = \frac{T_o}{T_{s_\infty}} \frac{T_{s_\infty}}{T_s} \tag{5.154}$$

then local boundary face static temperature T_s is calculated using the isentropic relationship

$$\frac{T_0}{T_s} = 1 + \frac{\gamma - 1}{2} M^2 \tag{5.155}$$

Outflow pressure

This boundary condition concerns the outflow from a arbitrary cell, with the ratio p_s/p_{s_∞} known. Extrapolation of the conserved variables in vector W is then possible, under the assumption that for all conserved quantities with the exception of ρE are the same at the boundary as in the cell. Then using these ρE can be recalculated using material from chapter 3.

Actuator Disc

Actuator discs are commonly used in aerospace as a pragmatic surrogate for a fully rotating propellor. The gross effect of the propellor is communicated to the solver via a prescribed discontinuity in the conserved variables - usually the momentum terms - across the disc. This method is also becoming popular in wind turbine modelling, especially where large arrays are to be simulated. zCFD allows a high level of customisation of this functionality, and users can define axial and radial variation of the forcing function - which can itself be a function of the flow variables.

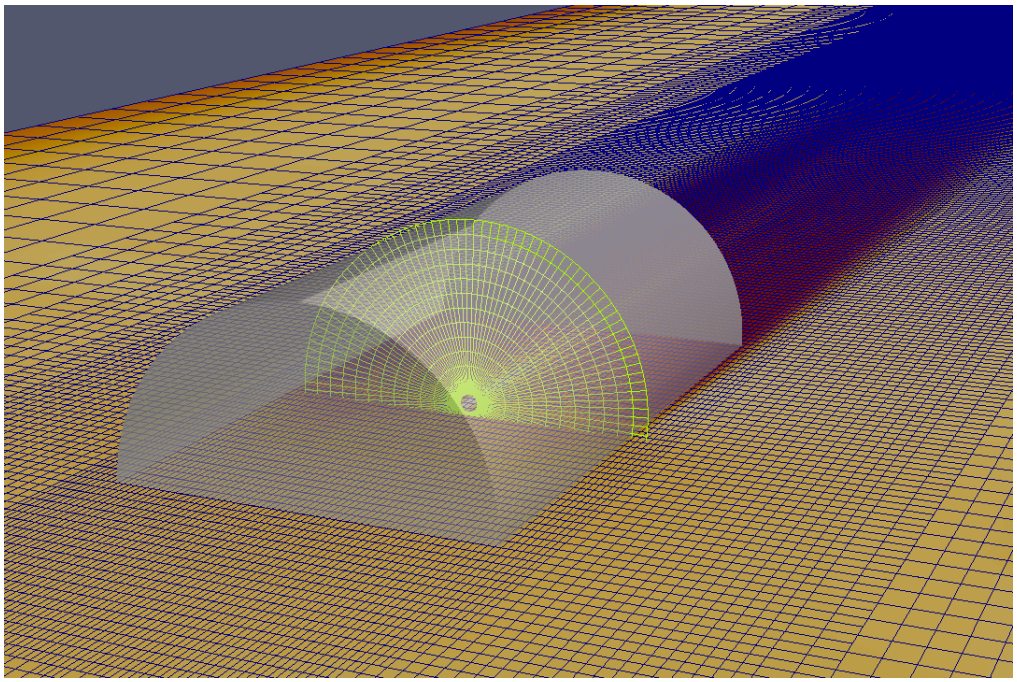


Figure 5.1: *Actuator disc (light green) and turbine fluid zone (grey shaded volume) within an unstructured CFD mesh (2D slice through a 3D mesh shown). The effect of the discontinuity in flow variables across the disc is smoothed (convolved) in 3D using a Gaussian distribution. The disc is split up into N_θ^D angular sectors and N_R^D radial steps. Each region on the disc is associated to volume mesh elements upstream and downstream of the disc within the fluid zone. The forcing function at the disc is distributed so that the mesh cells nearest the disc have the greatest effect, and that the volume integral of the source terms applied is equivalent to the step change prescribed across the disc.*

Note that the disc elements are not present in the original mesh. These are created automatically by the zCFD solver using parameters that can be set by the user in the input file. It should be noted that the mesh spacing used in the original mesh must be sufficient to resolve the actuator. Parameters governing the actuator disc model are: the disc centre, the disc radius, the disc normal vector n^D , the number of angular sectors N_θ^D and the number of radial segments N_R^D. Assuming that the disc model is supplied with thrust $T(r, \theta)$ and torque $Q(r, \theta)$ functions, the convolution function applies a Gaussian distribution in 3D so that the value of thrust per unit volume at a cell centre x_i is given as follows:

 ©Zenotech Ltd

$$T(x_i) = \sum_{j=0}^{N_R^D} \sum_{k=0}^{N_\theta^D} dT_{jk}\, dA_{jk}^D\, n^D\, \eta(\|x_i - x_{jk}\|)/W_{jk} \tag{5.156}$$

where dT_{jk} is the thrust per unit area and dA_{jk}^D is the area of disc element (j,k) respectively. The 3D convolution kernel η is defined by

$$\eta(d) = 1/(\pi^{3/2}\,\epsilon^2\,\exp(-(d/\epsilon)^2)) \tag{5.157}$$

The scaling parameter ϵ governs the degree of spreading of the convolution, and can be easily customised. It does not have to have a constant value and can have a different value for each segment on the disc. The default value is $\epsilon = \epsilon_{jk} = 2\,dr_j$, where dr_j is the change in radius over radial segment j. The weighted sum W_{jk} is determined as a discrete summation, to ensure exact correspondence between the actuator disc forces and the convolved forces:

$$W_{jk} = \sum_{i=0}^{N_{CELLS}} \max(\delta, \eta(\|x_i - x_{jk}\|))V_i \tag{5.158}$$

where δ is a small number (10^{-16}) to prevent numerical underflow and V_i is the volume of cell i. This summation includes all cells in the fluid zone corresponding to the actuator disk.

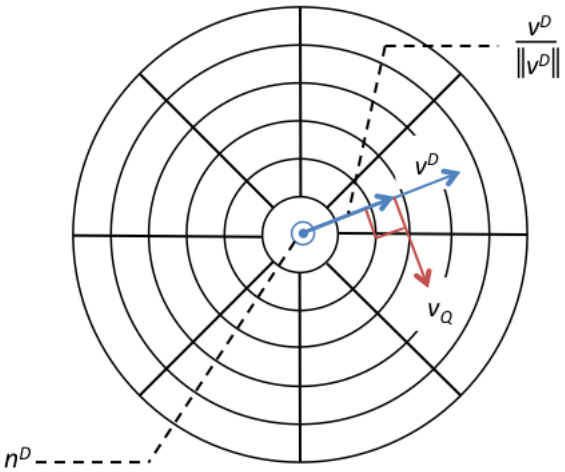


Figure 5.2: *The torque distribution on the actuator disc is converted to a force via the torque vector v_Q. Note that the disc normal n^D, and the normalised disc vector $v^D/\|v^D\|$ have unit length so the torque vector v_Q also has unit length.*

The same procedure is used to distribute the torque $Q(x_i)$ but in this case the per unit area torque is converted to to a per unit volume force in order to modify the momentum terms:

$$Q(x_i) = \sum_{j=0}^{N_R^D} \sum_{k=0}^{N_\theta^D} \underbrace{\frac{dQ_{jk}^D\, dA_{jk}^D}{\|v^D\|}\, v_Q}_{\text{Equivalent force}}\, \eta(\|x_i - x_{jk}^D\|)/W_{jk} \tag{5.159}$$

where the disc vector v^D is given by

$$v^D = x^D_{jk} - x^D_C \tag{5.160}$$

where x^D_{jk} is the centre point of the disc segment (j, k) and x^D_C is the disc centre. The torque vector v_Q is given by

$$v_Q = \frac{v^D}{\|v^D\|} \times n^D \tag{5.161}$$

Turbine Blade Modelling

When a wind turbine is modelled by an actuator disk, certain assumptions are made about the distribution of elements on the disc. The user can specify the inner radius r^D_I, the outer radius r^D_O and the number of angular segments N^D_θ into which to divide the disc. For convenience, and to maintain a uniform aspect ratio for elements on the disc, zCFD determines the radial extent dr for each element on the disc to match the (approximate) length scale determined by the angle $d\theta$ at the midpoint of the element $r + dr$. Ideally we want $dr \simeq (r + dr)\, d\theta$, so rearranging gives

$$dr = \frac{r\, d\theta}{1 - d\theta/2} \tag{5.162}$$

The axial induction factor $a = (u_\infty - u^D)/u^D$ from 1D momentum theory can be related [6] to the thrust coefficient C_T on any element on the disk by the relation

$$C_T = 4a(1 - a) \tag{5.163}$$

Note that for more complex models (multiple turbines, wake interactions or complex terrain), the free-stream velocity u_∞ for any given turbine may not be straightforward to define, and is an input to the model for each turbine. The velocity u^D at the disc does not need to be defined at this stage, because it is the thrust coefficient C_T that is usually supplied (as a function of windspeed) for turbines. Thus we solve for the (mininum) root of the quadratic (Equation 5.163) to obtain

$$a = \frac{1 - \sqrt{1 - C_T}}{2} \tag{5.164}$$

The thrust per unit area on the disc dT is given as

$$dT = 2\, \rho\, u^2_\infty\, a\, (1 - a) \tag{5.165}$$

The angular induction factor a_θ may be determined [6] to be

©Zenotech Ltd

$$a_\theta = -\frac{1}{2}\sqrt{\frac{1}{4} + (a(1-a))/\lambda_r^2} \tag{5.166}$$

where the local speed ratio $\lambda_r = (r + dr/2)\,\omega/u_\infty$ and ω is the angular velocity of the turbine.

The torque per unit area on the disc dQ is then

$$dQ = 2\,a_\theta\,(1-a)\,\rho\,u_\infty\,\omega\,(r + dr/2)^2 \tag{5.167}$$

zCFD can apply user-defined radial and azimuthal distribution functions of thrust and torque on the disc. By default the distributions are uniform, i.e. $dT^D_{ij} = dT^D$ and $dQ^D_{ij} = dQ^D$.

Blade Element Theory

Following [5] a blade element model can be used to determine the thrust and torque loading for each segment on the actuator disc, according to the local flow field providing that the turbine blade characteristics are known and that the tip speed ratio is provided. zCFD will automatically determine the optimal blade pitch to maximise torque. The values of dT and dQ are instead calculated for each segment as:

$$\begin{aligned} dT &= F_L \cos\theta_{rel} + F_D \sin\theta_{rel} \\ dQ &= F_L \sin\theta_{rel} - F_D \cos\theta_{rel} \end{aligned} \tag{5.168}$$

where

$$F_L = \frac{N_{blades}}{2\pi r} f_L \quad , \quad F_D = \frac{N_{blades}}{2\pi r} f_D \tag{5.169}$$

and

$$f_L = \frac{1}{2}\rho\,U^2_{rel}\,c\,C_l \quad , \quad f_D = \frac{1}{2}\rho\,U^2_{rel}\,c\,C_d \tag{5.170}$$

where c is the local chord length and the aerofoil section lift and drag coefficients C_l and C_d are provided as functions of the local angle of attack α. zCFD will interpolate values from lookup tables provided in the fluid zone definition.

The angle of attack α is given by

$$\alpha = \theta_{rel} - \beta \tag{5.171}$$

where θ_{rel} is the angle between the local flow and the actuator disc plane:

$$\theta_{rel} = \arctan\left(\frac{U}{r\omega_{rel}}\right) \tag{5.172}$$

where U is the local flow component normal to the disc and the relative anglular velocity ω_{rel} is given by

$$\omega_{rel} = \omega - \omega_{air} \tag{5.173}$$

where ω is the rotation rate of the rotor (as above) and ω_{air} is the (counter-rotating) local air, determined from the local flow velocity. Typically ω_{air} is less than 5% of ω and is often ignored.

The local blade angle β is a combination of the blade pitch and the local blade twist:

$$\beta = \beta_{pitch} + \beta_{twist} \tag{5.174}$$

The local twist angle β_{twist} must be provided in the fluid zone as a lookup table, or else is presumed to be zero. We presume that the turbine will automatically optimise the power output by adjusting the blade pitch angle β_{pitch} and that this angle is the same for all turbine blades. Currently we presume that the adjustment to optimal power power production is instantaneous and as the power production tends to be a monomodal function of β_{pitch}, zCFD uses a golden section search to find the local optimum over a user-specified range.

The relative velocity U_{rel} of the local aerofoil section accounts for the rate of rotation of the rotor, and is given by

$$U_{rel} = \sqrt{(r\omega_{rel})^2 + U^2} \tag{5.175}$$

Betz Limit

It is useful to compare the power production for a given windspeed against the theoretical optimum for an ideal horizontal axis turbine - known as the Betz Limit:

$$P_{BETZ} = \frac{16}{27} \times \frac{1}{2}\rho U_{ref}^3 A \tag{5.176}$$

Where U_{ref} is the reference freestream velocity at hub height. In practise, most turbines operate signifcantly below the Betz Limit, so a ratio greater than 80% should be regarded with caution.

Glauert Optimum

A more accurate ideal turbine performance estimate accounts for the wake rotation at a given tip speed ratio. zCFD will calculate the Glauert Optimum power production as a reference based on the closed-form solution of [15]:

$$P_{GLAUERT} = G(y) \times \frac{1}{2}\rho U_{ref}^3 A \tag{5.177}$$

where $G(y)$ is given by

$$G(y) = \frac{16(1-2y)}{27(1+y/4)}\left[1 + \frac{457}{1280}y + \frac{51}{640}y^2 + \frac{1}{160}y^3 + \frac{3}{2}y\frac{\log(2y) + (1-2y) + \frac{1}{2}(1-2y)^2}{(1-2y)^3}\right] \tag{5.178}$$

©Zenotech Ltd

where $y = (3b_0 - 1)$ and b_0 is given as an inverse function of the local inflow ratio λ_r where $r = R$ at the tip:

$$\lambda_r = \frac{\sqrt{1+b}(1-2b)}{\sqrt{3b-1}} \tag{5.179}$$

Since λ_r at the tip is equal to the tip speed ratio, we pre-compute a lookup table for $\lambda_r(b)$ over the domain $1/3 < b < 1/2$ and determine the value of $b = b_0$ for any given value of the tip speed ratio.

The resulting function of $P_{GLAUERT}$ for a given value of tip speed ratio is a smooth curve that aproaches the Betz Limit as tip speed ratio increases. This closed form solution is preferable to a number of integrable solutions provided in the literature, as these are often oscillatory integrands with poor convergence characteristics requiring significant numerical effort to evaluate.

Rotor Inertia

Acceleration of the rotor is presumed to be instantaneous if there is no value for the moment of inertia for the rotor. However if a value for the moment of inertia I_{ROTOR} is available, or can be estimated then we can provide a more realistic model for the rate of rotation. Provided with the aerofoil section upper and lower surface definitions, zCFD will calculate the rotor inertia as:

$$I_{ROTOR} = N_{blades}\, \rho_{material} \int_{r=0}^{r=R} r^2 A(r) dr \tag{5.180}$$

where the cross-sectional area of the blade is given by the section area $A(r)$. In practise this is computed in zCFD by numerically integrating the normalised aerofoil section area and scaling by the local chord. The blade material density $\rho_{material}$ has a default value of 200kg/m^3 and can be specified in the fluid zone definition.

Tip Loss Correction

There are a number of options for tip loss correction for wind turbines, and zCFD allows for use of the most common ones. Following the literature, we apply a tip loss correction to the lift coefficient only, according to the normalised distance to the blade tip:

$$TLC_{elliptic}(r/R) = \sqrt{1-(r/R)^2} \tag{5.181}$$

$$TLC_{acos-fit}(r/R) = \frac{2}{\pi}\arccos(\exp(-63(1-(r/R)^2))) \tag{5.182}$$

$$TLC_{acosshift-fit}(r/R) = \frac{2}{\pi}\arccos(\exp(-48(1-(r/R)^2)-0.5)) \tag{5.183}$$

$$TLC_{f-fit}(r/R) = 1 - 2.5\frac{(1-(r/R)^2)^{0.39})}{(2-(r/R)^2)^{64}} \tag{5.184}$$

Turbine Controller

A controller function is provided in zCFD to apply simple rules to govern the allowable state of the turbine. This function is modifiable by the user in *zutil* and will allow for external data sources to be integrated - for example to control the aggregate behaviour of many turbines in a farm.

The controller will automatically stop the rotor from turning if the rated power is exceeded or if the windspeed is not in the range between the cut-in and cut-out values. If the rated tip-speed is not yet met, the rotor will be accelerated by setting the blade pitch to produce optimal instanous power (using a 50% fraction of the available torque) and the remaining torque will be supplied to the turbine generator. If the rated tip speed is met, then the rotor blades will set to recover optimal power until the rated power is met. When the rotor is at maximum power and maximum tip speed, the blades are feathered to maintain the rated power. A user-specified frictional loss is applied.

The controller will identify whether a critical damage turbulence intensity (default 15%) and wind speed (default 10m/s) are exceeded at any point on the rotor disc and if so the rotor is automatically slowed down and stowed.

The turbine may be set to rotate clockwise (as most modern turbines do) or anti-clockwise when viewed from the front, and in each case the rotational force applied to the air is adjusted accordingly. zCFD also calculates the average onset air angular induction as a fraction of the rate of rotor rotation, and this is reported in the result file. Typical values lie in the range (-2%, -5%).

©Zenotech Ltd

Aircraft Propellors

The mechanics of a propellor when used to provide thrust to an aircraft are similar to those of a wind turbine, but differ in some conventions. As aircraft propellors are driven by motors, we neglect inertial effects from the mass of the propellor itself. zCFD provides a blade element model specifically for use with propellors for aerodynamic thrust. Usage is described in the zCFD User Guide.

An aircraft propellor zone is fully specified in zCFD using the *number of blades*, *number of segments*, *status* (on or off), *omega* (which is assumed to be constant), *blade chord*, *blade twist*, *rotation direction*, *update frequency*, *aerofoil cl*, *aerofoil cd*, *outer radius*, *inner radius*, *tip loss correction* (True or False), *def* and *model* parameters as above.

Note that the propellor blades are oriented so that lift primarily produces thrust, rather than torque (as for a wind turbine) so the convention for the relative orientation of the blade chord to the local wind velocity θ_{rel} is rotated by 90 degrees. zCFD automtically applied this rotation for propellors.

Tip Loss Correction

The correction for tip losses for aircraft propellors are typically elliptic and depend only on the distance (as a proportion of the rotor outer radius) from the hub at which to apply the loss. If tip loss corrections are used (by specifying a value for *tip loss correction*, for example *elliptic*), a single number between 0 and 1 defines the *tip loss correction radius*. By default this is 0.8.

Chapter 6

PISO

PISO (Pressure-Implicit with Splitting Operators) algorithm.

6.1 Predictor Step

Continuity for incompressible flows

$$\nabla \cdot \rho V = 0 \tag{6.1}$$

Momentum equation

$$\frac{\partial}{\partial t}(\rho V) + \nabla \cdot (\rho V \cdot V) = -\nabla \cdot p + \nabla \cdot \mu \nabla V \tag{6.2}$$

Discretise and approximate implicit non linear flux term

$$\frac{\Omega}{\Delta t}\rho V_i^n - \frac{\Omega}{\Delta t}\rho V_i^m + \sum_f^{nb} \rho V_f^n . V_f^m . S_f - \sum_f^{nb} \mu \nabla V_f^n \cdot S_f = -\sum_f^{nb} p_f^m S_f + \Omega S_i^m \tag{6.3}$$

Let $\rho V_f = f_i \rho V_i + f_j \rho V_j$

Define

$$\nabla V_f^n \cdot S_f = \Delta . (\nabla V)_f + k . (\nabla V)_f \tag{6.4}$$

Where Δ is the orthogonal contribution and k is the non-orthogonal correction

$$\nabla V_f^n \cdot S_f = |\Delta| \frac{V_j^n - V_i^n}{|d_{ij}|} + k(\nabla V^m)_f \tag{6.5}$$

With the two vectors Δ and k satisfying

$$S = \Delta + k \tag{6.6}$$

Choose $\Delta = \frac{d_{ij}}{|d_{ij}|}|S|$ known as the Orthogonal correction approach (See [1]) for further information

$$\nabla V_f^n \cdot S_f = |\Delta| \frac{V_j^n - V_i^n}{|d_{ij}|} + k(g_f \nabla V_i^m + (1 - g_f)\nabla V_j^m) \tag{6.7}$$

Expanding

$$\begin{aligned} &\frac{\Omega}{\Delta t}\rho V_i^n - \frac{\Omega}{\Delta t}\rho V_i^m + \sum_f^{nb}(f_i \rho V_i^n + f_j \rho V_j^n) V_f^m . S_f \\ &- \sum_f^{nb} \mu(|\Delta| \frac{V_j^n - V_i^n}{|d_{ij}|} + k(g_f \nabla V_i^m + (1 - g_f)\nabla V_j^m)) = \\ &- \sum_f^{nb} p_f^m S_f + \Omega S_i^m \end{aligned} \tag{6.8}$$

Re-arranging

$$\begin{aligned} &\left[\frac{\Omega}{\Delta t} + \sum_f^{nb} f_i V_f^m . S_f + \sum_f^{nb} \frac{\mu}{\rho}\frac{|\Delta|}{|d_{ij}|}\right] \rho V_i^n + \\ &\left[\sum_f^{nb} f_j V_f^m . S_f - \sum_f^{nb} \frac{\mu}{\rho}\frac{|\Delta|}{|d_{ij}|}\right] \rho V_j^n = \\ &\frac{\Omega}{\Delta t}\rho V_i^m - \sum_f^{nb} p_f^m S_f + \Omega S_i^m + \sum_f^{nb} \frac{\mu}{\rho} k(g_f \nabla V_i^m + (1 - g_f)\nabla V_j^m) \end{aligned} \tag{6.9}$$

$$A\rho V_i^n + H^{'} \rho V_j^n = R - \sum_f^{nb} p_f^m S_f \tag{6.10}$$

$$A\rho V^* + H^{'} \rho V^* = R - \sum_f^{nb} p_f^m S_f \tag{6.11}$$

Add implicit under-relaxation term.

[1] http://powerlab.fsb.hr/ped/kturbo/OpenFOAM/docs/HrvojeJasakPhD.pdf

©Zenotech Ltd

$$\frac{A}{\alpha_u}\rho V^* + H^{'}\rho V^* = R - \sum_f^{nb} p_f^m S_f + \frac{1-\alpha_u}{\alpha_u} A\rho V^m \tag{6.12}$$

Solve for ρV^* using a linear solver. The final predicted velocity V^* is a relaxed combination of V and computed V^*

6.2 Corrector Step

Define explicit equation for corrected velocity V^{**} using predicted velocity V^*

$$A\rho V^{**} + H^{'}\rho V^* = R - \nabla \cdot p^* \tag{6.13}$$

With vector $H = R - H^{'}\rho V^*$

$$\rho V^{**} = A^{-1}H - A^{-1}\nabla \cdot p^* \tag{6.14}$$

Taking the divergence and recalling that the continuity equation defines $\nabla \rho V = 0$

$$\nabla^2(A^{-1}p^*) = \nabla.(A^{-1}H) \tag{6.15}$$

Integrate over a cell applying divergence theorem

$$\sum_f^{nb}(A^{-1}\nabla p^*)|_f.S_f = \sum_f^{nb}(A^{-1}H)|_f.S_f \tag{6.16}$$

Define $\nabla p^*|_f \cdot S_f = \frac{p_j - p_i}{|d_{ij}|}|S_f|$

$$\sum_f^{nb} A^{-1}|_f.\frac{S_f}{|S_f|}\frac{p_j^* - p_i^*}{|d_{ij}|}|S_f| = \sum_f^{nb}(A^{-1}H)|_f.S_f \tag{6.17}$$

$$\left[\sum_f^{nb}\frac{A^{-1}}{|d_{ij}|}.S_f\right]p_j^* - \left[\sum_f^{nb}\frac{A^{-1}}{|d_{ij}|}.S_f\right]p_i^* = \sum_f^{nb}(A^{-1}H)|_f.S_f + \phi_c \tag{6.18}$$

Note the RHS needs and additional correction term to account for the mass flux through a face. (See [2])

$$\phi^d = (\rho V_f.S_f - (\rho V^{old}.S_f)) \tag{6.19}$$

[2] https://github.com/OpenFOAM/OpenFOAM-2.2.x/blob/master/applications/solvers/incompressible/pisoFoam/ line 90

Note this is set to zero at the boundary. (See [3] [4] for Kc

$$K_c = 1 - min\left[\frac{|\phi^d|}{|\rho V_f.S_f| + \epsilon}, 1\right] \tag{6.20}$$

$$\phi^c = \phi^d K_c \frac{A^{-1}}{\Delta t} \tag{6.21}$$

The boundary value of the RHS term also need to be scaled to so that the total mass flow if zero. (See https://github.com/OpenFOAM/OpenFOAM-2.2.x/blob/master/src/finiteVolume/cfdTools/gene

Solve for corrected pressure p^* then compute corrected velocity u^{**}. Note the corrected pressure is not relaxed.

$$\rho V^{**} = A^{-1}H - A^{-1}\sum_f^{nb} p_f^*.S_f \tag{6.22}$$

Update mass flow through face to be used in the correction of the RHS of the pressure equation.

Compute the face mass flux using Rhie-Chow interpolation

$$\rho V_f = \overline{\rho V_f} + (A^{-1})|_f(\overline{\nabla p_f} - \nabla p_f) \tag{6.23}$$

Further correction steps can be applied using same A and H hence the name Split Operators.

[3] https://github.com/OpenFOAM/OpenFOAM-2.2.x/blob/master/src/finiteVolume/finiteVolume/ddtSchemes/EulerDdtSc
line 392

[4] https://github.com/OpenFOAM/OpenFOAM-2.2.x/blob/master/src/finiteVolume/finiteVolume/ddtSchemes/ddtScheme/

©Zenotech Ltd

Chapter 7

SIMPLE

SIMPLE algorithm for steady state incompressible flows.

7.1 Predictor Step

Continuity for incompressible flows

$$\nabla.\rho V = 0 \tag{7.1}$$

Momentum equation with implicit flux terms

$$\nabla \cdot (\rho V \cdot V) = -\nabla \cdot p + \nabla \cdot \mu \nabla V \tag{7.2}$$

Discretise and approximate non linear flux term assuming interpolation to the face

$$\sum_{f}^{nb} \rho V_f^n . V_f^m . S_f - \sum_{f}^{nb} \mu \nabla V_f^n \cdot S_f = -\sum_{f}^{nb} p_f^m S_f + \Omega S_i^m \tag{7.3}$$

Let $\rho V_f = f_i \rho V_i + f_j \rho V_j$

Define

$$\nabla V_f^n \cdot S_f = \Delta.(\nabla V)_f + k.(\nabla V)_f \tag{7.4}$$

Where Δ is the orthogonal contribution and k is the non-orthogonal correction

$$\nabla V_f^n \cdot S_f = |\Delta| \frac{V_j^n - V_i^n}{|d_{ij}|} + k(\nabla V^m)_f \tag{7.5}$$

With the two vectors Δ and k satisfying

$$S = \Delta + k \tag{7.6}$$

Choose $\Delta = \frac{d_{ij}}{|d_{ij}|}|S|$ known as the Orthogonal correction approach (See [1]) for further information

$$\nabla V_f^n \cdot S_f = |\Delta| \frac{V_j^n - V_i^n}{|d_{ij}|} + k(g_f \nabla V_i^m + (1 - g_F)\nabla V_j^m) \tag{7.7}$$

Inserting

$$\sum_f^{nb}(f_i \rho V_i^n + f_j \rho V_j^n) V_f^m . S_f - \sum_f^{nb} \mu(|\Delta| \frac{V_i^n - V_j^n}{|d_{ij}|} + k(g_f \nabla V_i^m + (1-g_f)\nabla V_j^m)) = -\frac{1}{\Omega}\sum_f^{nb} p_f^m S_f + \Omega S_i^m \tag{7.8}$$

Re-arranging

$$\begin{aligned} &\left[\sum_f^{nb} f_i V_f^m . S_f - \sum_f^{nb} \frac{\mu}{\rho}\frac{|\Delta|}{|d_{ij}|}\right] \rho V_i^n + \left[\sum_f^{nb} f_j V_f^m . S_f + \sum_f^{nb} \frac{\mu}{\rho}\frac{|\Delta|}{|d_{ij}|}\right] \rho V_j^n = \\ &- \sum_f^{nb} p_f^m S_f + \Omega S_i^m + \sum_f^{nb} \frac{\mu}{\rho} k(g_f \nabla V_i^m + (1 - g_f)\nabla V_j^m) \end{aligned} \tag{7.9}$$

$$A \rho V_i^n + H^{'} \rho V_j^n = R - \sum_f^{nb} p_f^m S_f \tag{7.10}$$

$$A \rho V^* + H^{'} \rho V^* = R - \sum_f^{nb} p_f^m S_f \tag{7.11}$$

Add implicit relaxation term

$$\frac{A}{\alpha_u} \rho V^* + H^{'} \rho V^* = R - \sum_f^{nb} p_f^m S_f + \frac{1 - \alpha_u}{\alpha_u} A \rho V^m \tag{7.12}$$

Solve for V^* using a linear solver

[1] http://powerlab.fsb.hr/ped/kturbo/OpenFOAM/docs/HrvojeJasakPhD.pdf

©Zenotech Ltd

7.2 Corrector Step

Define explicit equation for corrected velocity V^{**} using predicted velocity V^{*}

$$A\rho V^{**} + H^{'}\rho V^{*} = R - \nabla p^{*} \tag{7.13}$$

With vector $H = R - H^{'}\rho V^{*}$

$$\rho V^{**} = A^{-1}H - A^{-1}\nabla p^{*} \tag{7.14}$$

Taking the divergence and recalling that the continuity equation defines $\nabla \cdot \rho V = 0$

$$\nabla^{2}(A^{-1}p^{*}) = \nabla.(A^{-1}H) \tag{7.15}$$

Integrate over a cell applying divergence theorem

$$\sum_{f}^{nb}(A^{-1}\nabla p^{*})|_{f}.S_{f} = \sum_{f}^{nb}(A^{-1}H)|_{f}.S_{f} \tag{7.16}$$

Define $\nabla p^{*}|_{f} = \frac{p_{j}-p_{i}}{|d_{ij}|}$

$$\sum_{f}^{nb}A^{-1}.n_{f}\frac{p_{j}^{*}-p_{i}^{*}}{|d_{ij}|}.S_{f} = \sum_{f}^{nb}(A^{-1}H)|_{f}.S_{f} \tag{7.17}$$

$$\left[\sum_{f}^{nb}\frac{A^{-1}}{|d_{ij}|}.S_{f}\right]p_{j}^{*} - \left[\sum_{f}^{nb}\frac{A^{-1}}{|d_{ij}|}.S_{f}\right]p_{i}^{*} = \sum_{f}^{nb}(A^{-1}H)|_{f}.S_{f} \tag{7.18}$$

The $A^{-1}H$ term on the RHS needs to be interpolated to faces

Note boundary values of RHS need to be scaled so that total mass flow is zero

Solve for corrected pressure p^{*} then compute corrected velocity V^{**}

$$\rho V^{**} = A^{-1}H - A^{-1}\sum_{f}^{nb}p_{f}^{*}.S_{f} \tag{7.19}$$

Compute the face mass flux using Rhie-Chow interpolation

$$\rho V_{f} = \overline{\rho V_{f}} + (A^{-1})|_{f}(\overline{\nabla p_{f}} - \nabla p_{f}) \tag{7.20}$$

Repeat starting from predictor step using ρV^{**}, p^* and $\rho V_f^{**}.S_f$

©Zenotech Ltd

Chapter 8

Rhie-Chow Interpolation

The development of the Rhie-Chow interpolation procedure for the calculation of the velocity components at a control-volume face enabled the formulation of the SIMPLE algorithm on collocated grids. In this procedure the interface velocity vector is obtained by constructing a pseudo-momentum equation at the control-volume face from the momentum equations of cells i and j straddling the interface. These equations are given by

$$\rho V_i = A^{-1}H - A^{-1}\nabla \cdot p_i \tag{8.1}$$

with $\nabla \cdot p_i = \Omega \nabla p_i$

$$\rho V_i = A^{-1}H - A^{-1}\Omega \nabla p_i \tag{8.2}$$

The pseudo-momentum equation for V_f should be similar and can be written as

$$\rho V_f = (A^{-1}H)|_f - (A^{-1}\Omega)|_f \nabla p_f \tag{8.3}$$

Assuming linear interpolation to the face let

$$(A^{-1}\Omega)|_f = g_f(A^{-1})_i\Omega_i + (1-g_f)(A^{-1})_j\Omega_j \tag{8.4}$$

and

$$(A^{-1}H)|_f = g_f(A^{-1}H)_i + (1-g_f)(A^{-1}H)_j \tag{8.5}$$

Expanding the $A^{-1}H$ term

$$(A^{-1}H)|_f = g_f(\rho V_i + A^{-1}\Omega_i \nabla p_i) + (1-g_f)(\rho V_j + A^{-1}\Omega_j \nabla p_j) \tag{8.6}$$

$$(A^{-1}H)|_f = \overline{\rho V_f} + (A^{-1}\Omega)|_f \overline{\nabla p_f} \tag{8.7}$$

where $\overline{\rho V_f}$ is the face interpolated velocity and $\overline{\nabla p_f}$ is the face interpolated cell pressure gradient

Substituting $(A^{-1}H)|_f$ in the equation for ρV_f

$$\rho V_f = \overline{\rho V_f} + (A^{-1}\Omega)|_f \overline{\nabla p_f} - (A^{-1}\Omega)|_f \nabla p_f \tag{8.8}$$

$$\rho V_f = \overline{\rho V_f} + (A^{-1}\Omega)|_f (\overline{\nabla p_f} - \nabla p_f) \tag{8.9}$$

©Zenotech Ltd

Chapter 9

Coupled Incompressible

Coupled algorithm for steady state incompressible flows.

Continuity for incompressible flows

$$\nabla.\rho V = 0 \tag{9.1}$$

Momemtum equation with implicit flux terms

$$\nabla \cdot (\rho V \cdot V) = -\nabla \cdot p + \nabla \cdot \mu \nabla V + \rho F_{ext} \tag{9.2}$$

Discretise and approximate non linear flux term assuming interpolation to the face

$$\sum_{f}^{nb} \rho V_f^n . V_f^m . S_f - \sum_{f}^{nb} \mu \nabla V_f^n \cdot S_f + \sum_{f}^{nb} p_f^n S_f = \Omega S_i^m \tag{9.3}$$

Let $\rho V_f = f_i \rho V_i + f_j \rho V_j$

Let $p_f = g_f p_i + (1 - g_f) p_j$

The discretised continuity equation at a face is

$$\sum \rho V_f . S_f = 0 \tag{9.4}$$

Using Rhie-Chow interpolation

$$\sum (\overline{\rho V_f} + (A^{-1})|_f (\overline{\nabla p_f} - \nabla p_f)) . S_f = 0 \tag{9.5}$$

Expanding

$$\sum \overline{\rho V_f}.S_f - \sum (A^{-1})|_f \nabla p_f.S_f = -\sum (A^{-1})|_f \overline{\nabla p_f}.S_f \tag{9.6}$$

Recall that

$$\overline{\rho V_f} = g_f \rho V_i + (1 - g_f)\rho V_j \tag{9.7}$$

$$\overline{\nabla p_f} = g_f \nabla p_i + (1 - g_f)\nabla p_j \tag{9.8}$$

Incompressible internal energy equation

$$\nabla \cdot (\rho e \cdot V) = -\nabla \cdot k \nabla T + \nabla \cdot (V \cdot \mu \nabla V) \tag{9.9}$$

Discretise

$$\sum_f^{nb} \rho e_f^n.V_f^m.S_f - \sum_f^{nb} \mu V_f^m \nabla V_f^n \cdot S_f + \sum_f^{nb} k \nabla T^n.S_f = 0 \tag{9.10}$$

$$\sum_f^{nb} \rho e_f^n.V_f^m.S_f - \sum_f^{nb} \mu V_f^m \nabla V_f^n \cdot S_f + \sum_f^{nb} \frac{k}{\rho C_p} k \nabla \rho e^n.S_f = 0 \tag{9.11}$$

©Zenotech Ltd

Diagonal terms

$$\begin{bmatrix} \sum f_i V_f^m . S_f + \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & 0 & 0 & \sum g_f S_f^x & 0 \\ 0 & \sum f_i V_f^m . S_f + \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & 0 & \sum g_f S_f^y & 0 \\ 0 & 0 & \sum f_i V_f^m . S_f + \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & \sum g_f S_f^z & 0 \\ \sum g_f . S_f^x & \sum g_f . S_f^y & \sum g_f . S_f^z & + \sum \frac{|(A^{-1})|_f S_f|}{|d_{ij}|} & 0 \\ -\sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^x \frac{|\Delta_x|}{|d_{ij}|} & -\sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^x \frac{|\Delta_y|}{|d_{ij}|} & -\sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^x \frac{|\Delta_z|}{|d_{ij}|} & 0 & \sum f_i V_f^m . S_f + \sum_f^{nb} \frac{k}{\rho C_p} \frac{|\Delta|}{|d_{ij}|} \end{bmatrix} \begin{bmatrix} \rho u_i \\ \rho v_i \\ \rho w_i \\ p_i \\ \rho e_i \end{bmatrix} \quad (9.12)$$

Off diagonal terms

$$\begin{bmatrix} \sum f_j V_f^m . S_f - \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & 0 & 0 & \sum (1 - g_f) S_f^x & 0 \\ 0 & \sum f_j V_f^m . S_f - \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & 0 & \sum (1 - g_f) S_f^y & 0 \\ 0 & 0 & \sum f_j V_f^m . S_f - \sum_f^{nb} \frac{\mu}{\rho} \frac{|\Delta|}{|d_{ij}|} & \sum (1 - g_f) S_f^z & 0 \\ \sum (1 - g_f) . S_f^x & \sum (1 - g_f) . S_f^y & \sum (1 - g_f) . S_f^z & - \sum \frac{|(A^{-1})|_f S_f|}{|d_{ij}|} & 0 \\ \sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^x \frac{|\Delta_x|}{|d_{ij}|} & \sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^x \frac{|\Delta_y|}{|d_{ij}|} & \sum_f^{nb} \frac{\mu}{\rho} V_f^m . n_f^s \frac{|\Delta_z|}{|d_{ij}|} & 0 & \sum f_j V_f^m . S_f - + \sum_f^{nb} \frac{k}{\rho C_p} \frac{|\Delta|}{|d_{ij}|} \end{bmatrix} \begin{bmatrix} \rho u_j \\ \rho v_j \\ \rho w_j \\ p_j \\ \rho e_j \end{bmatrix} \quad (9.13)$$

©Zenotech Ltd

Right hand side

$$\begin{bmatrix} \Omega S_i^m + \sum_f^{nb} \frac{\mu}{\rho} k_x (g_f \nabla V_i^m + (1-g_f)\nabla V_j^m) \\ \Omega S_i^m + \sum_f^{nb} \frac{\mu}{\rho} k_y (g_f \nabla V_i^m + (1-g_f)\nabla V_j^m) \\ \Omega S_i^m + \sum_f^{nb} \frac{\mu}{\rho} k_y (g_f \nabla V_i^m + (1-g_f d)\nabla V_j^m) \\ -\sum (A^{-1})|_f (g_f \nabla p_i^m + (1-g_f)\nabla p_j^m).S_f \\ \sum_f^{nb} \frac{\mu}{\rho} V_f^m.S_f k(g_f \nabla V_i^m + (1-g_f)\nabla V_j^m) - \sum_f^{nb} \frac{k}{\rho C_p} k(g_f \nabla V_i^m + (1-g_f)\nabla V_j^m) \end{bmatrix} \quad (9.14)$$

Solve using a linear solver.

Update ρV_f using Rhie-Chow interpolation.

Solve other scalar equations then repeat until convergence

©Zenotech Ltd

Chapter 10

Turbulence Modelling

The Favre-averaged or Reynolds Averaged Navier Stokes (RANS) equations result in four terms that need to be modelled. These are a Reynolds stress term in the momentum equation and turbulent heat flux, molecular diffusion and turbulent transport in the energy equation.

The Reynolds Stress term is usually represented by the Boussinesq approximation

$$\tau_{ij}^t = 2\mu_t \left(S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij} \right) - \frac{2}{3}\rho k \delta_{ij} \tag{10.1}$$

with τ_{ij}^t representing Turbulent stress and the strain rate tensor S_{ij} defined as before. The turbulent heat flux is modelled using the Reynolds analogy and Fourier's law

$$C_p \overline{\rho u_j'' T''} \approx -\frac{C_p \mu_t}{Pr_t}\frac{\partial T}{\partial x_j} \tag{10.2}$$

where Pr_t is a turbulent Prandtl number (e.g. 0.9 for air).

The terms associated with molecular diffusion and turbulent transport in the energy equation are lumped together.

$$\overline{\sigma_{ij} u_i''} - \frac{1}{2}\overline{\rho u_i'' u_j'' u_k''} \approx (\mu + \sigma_k \mu_t)\frac{\partial k}{\partial x_j} \tag{10.3}$$

where σ_k is coefficient associated with the modelling equation for k.

These terms are added to the equations previously described and the turbulence model is then used to calculate the eddy viscosity μ_t and the turbulent kinetic energy k.

10.1 Menter SST-2003Vsust Two Equation Model

A detailed description of the Standard Menter Shear Stress Transport (SST) two equation turbulence model can be found in [12]. A variation of this model that was published in 2003 (SST-2003) has several minor variations of the original SST version. See [11]. The Menter SST Model with Vorticity Source Term (SST-V) makes use of the vorticity magnitude in the production term (see [13]) and the SST Model with Controlled Decay (SST-sust) eliminates the non-physical decay of turbulence in the freestream for external aerodynamic problems through addition of sustaining terms to the equations. See [17]. The current implementation combines all three updates to the Standard model.

Turbulent kinetic energy equation

$$\begin{aligned}\frac{\partial(\rho k)}{\partial \tau}+\frac{\partial(u\,\rho k)}{\partial x}+\frac{\partial(v\,\rho k)}{\partial y}+\frac{\partial(w\,\rho k)}{\partial z} =& P_m-\beta^*\rho\omega k \\ &+\frac{\partial}{\partial x}\left[(\mu+\sigma_k\mu_t)\frac{\partial k}{\partial x}\right] \\ &+\frac{\partial}{\partial y}\left[(\mu+\sigma_k\mu_t)\frac{\partial k}{\partial y}\right] \\ &+\frac{\partial}{\partial z}\left[(\mu+\sigma_k\mu_t)\frac{\partial k}{\partial z}\right] \\ &+\beta^*\rho\omega_{amb}k_{amb} \\ &+P_{kb}\end{aligned} \tag{10.4}$$

$$\frac{\partial(\rho k)}{\partial \tau}+\nabla\cdot(V\cdot\rho k)=P_m-\beta^*\rho\omega k+\nabla\cdot\left[(\mu+\sigma_k\mu_t)\nabla k\right]+\beta^*\rho\omega_{amb}k_{amb}+P_{kb} \tag{10.5}$$

Equation that defines the turn over time of turbulent eddies ω

$$\begin{aligned}\frac{\partial(\rho\omega)}{\partial \tau}+\frac{\partial(u\,\rho\omega)}{\partial x}+\frac{\partial(v\,\rho\omega)}{\partial y}+\frac{\partial(w\,\rho\omega)}{\partial z} =& \frac{\gamma_m}{\nu_t}P_m-\beta\rho\omega^2 \\ &+\frac{\partial}{\partial x}\left[(\mu+\sigma_\omega\mu_t)\frac{\partial \omega}{\partial x}\right] \\ &+\frac{\partial}{\partial y}\left[(\mu+\sigma_\omega\mu_t)\frac{\partial \omega}{\partial y}\right] \\ &+\frac{\partial}{\partial z}\left[(\mu+\sigma_\omega\mu_t)\frac{\partial \omega}{\partial z}\right] \\ &+2\,(1-F_1)\,\frac{\rho\sigma_{\omega 2}}{\omega}\left(\frac{\partial k}{\partial x}\frac{\partial \omega}{\partial x}+\frac{\partial k}{\partial y}\frac{\partial \omega}{\partial y}+\frac{\partial k}{\partial z}\frac{\partial \omega}{\partial z}\right) \\ &+\beta\rho\omega_{amb}^2 \\ &+\frac{\gamma_m}{\nu_t}P_{\omega b}\end{aligned} \tag{10.6}$$

$$\begin{aligned}\frac{\partial(\rho\omega)}{\partial \tau}+\nabla\cdot(V\cdot\rho k) =& \frac{\gamma_m}{\nu_t}P_m-\beta\rho\omega^2+\nabla\cdot\left[(\mu+\sigma_\omega\mu_t)\nabla\omega\right] \\ &+2\,(1-F_1)\,\frac{\rho\sigma_{\omega 2}}{\omega}\left(\nabla k\cdot\nabla\omega\right)+\beta\rho\omega_{amb}^2+\frac{\gamma_m}{\nu_t}P_{\omega b}\end{aligned} \tag{10.7}$$

Turbulent production term

$$P_m=\mu_t\Omega_v^2-\frac{2}{3}\rho k\delta_{ij}\frac{\partial u_i}{\partial x_j}$$

Buoyancy production terms

$$P_{kb}=-\frac{\mu_t}{\rho\sigma_\rho}g_i\left[\frac{\partial \rho}{\partial x_i}-\frac{\rho\Gamma}{(\gamma-1)T}\right] \tag{10.8}$$

Turbulence Schmidt Number $\sigma_\rho=1.0$

©Zenotech Ltd

$$P_{\omega b} = C_3 \max\left(0, P_{kp}\right) \tag{10.9}$$

Dissipation coefficient $C_3 = 1$

Turbulent viscosity

$$\mu_t = \rho k T \tag{10.10}$$

$$T = \min\left[\frac{a_1}{\max\left(a_1\omega, \Omega_v F_2\right)}, \frac{0.6}{\sqrt{3}\Omega_v}\right] \tag{10.11}$$

Vorticity magnitude

$$\Omega_v = \sqrt{2W_{ij}W_{ij}} \tag{10.12}$$

$$W_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right) \tag{10.13}$$

RANS length scale

$$L_{RANS} = \frac{\sqrt{k}}{\beta^{*\frac{1}{4}}\omega} \tag{10.14}$$

The turbulent kinetic energy dissipation term can be expressed as a function of the length scale

$$\beta^*\omega\rho k = \rho k \frac{\sqrt{k}\beta^{*\frac{3}{4}}}{L_{RANS}} \tag{10.15}$$

For eddy resolving techniques like DES, DDES and LES the lengthscale can be set according to the appropriate filter function Blending function

$$\phi = F_1\phi_1 + (1 - F_1)\,\phi_2 \tag{10.16}$$

$$\gamma_m = F_1\gamma_1 + (1 - F_1)\,\gamma_2 \tag{10.17}$$

$$\beta_m = F_1\beta_1 + (1 - F_1)\,\beta_2 \tag{10.18}$$

$$F_1 = \max\left(F_3, \tanh\left(arg_1^4\right)\right) \tag{10.19}$$

$$arg_1 = \min\left[\max\left(\frac{\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right), \frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}d^2}\right] \tag{10.20}$$

$$CD_{k\omega} = \max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right) \tag{10.21}$$

$$F_2 = \tanh\left(arg_2^2\right) \tag{10.22}$$

$$arg_2 = \max\left(2\frac{\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right) \tag{10.23}$$

$$F_3 = \exp\left(-\left(\frac{R_y}{120}\right)^8\right) \tag{10.24}$$

$$R_y = \frac{\rho d k^{1/2}}{\mu} \tag{10.25}$$

Production limiter

$$P_{m_{lim}} = \min\left(P_m, 10\beta^*\rho\omega k\right) \tag{10.26}$$

Boundary conditions

$$\omega_{wall} = 10\frac{6\nu}{\beta_1 \Delta d^2} \tag{10.27}$$

$$k_{wall} = 0 \tag{10.28}$$

$$\omega_{farfield} = \omega_{amb} = \frac{5U_\infty}{L} \tag{10.29}$$

$$\omega_{farfield} = \omega_{amb} = \frac{k}{\nu}\frac{\mu}{\mu_t} \tag{10.30}$$

Note

$$\frac{\mu_t}{\mu} = \frac{\sqrt{k}}{\nu}\beta^{*\frac{1}{4}} L_{RANS} \tag{10.31}$$

With U_∞ the speed at infinity.

$$k_{farfield} = k_{amb} = 10^{-6}U_\infty^2 \tag{10.32}$$

$$k_{farfield} = k_{amb} = \frac{3}{2}\left(U_\infty I\right)^2 \tag{10.33}$$

Where I is the turbulence intensity

Constants

$$\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}} \tag{10.34}$$

$$\gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^*}} \tag{10.35}$$

$$\sigma_{k1} = 0.85\ \sigma_{k2} = 1.0 \tag{10.36}$$

$$\sigma_{\omega 1} = 0.5\ \sigma_{\omega 2} = 0.856 \tag{10.37}$$

$$\beta_1 = 0.075\ \beta_2 = 0.0828 \tag{10.38}$$

$$\beta^* = 0.09 \tag{10.39}$$

$$\kappa = 0.41 \tag{10.40}$$

$$a_1 = 0.31 \tag{10.41}$$

The RANS conserved variables are

$$W = \left[\rho, \rho u, \rho v, \rho w, \rho E, \rho k, \rho\omega, \rho\phi\right]^T$$

corresponding to primitive variables

$$Q = \left[p, u, v, w, T, k, \omega, \phi\right]^T.$$

©Zenotech Ltd

The advective flux terms are given by

$$F^{RANS} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \rho uk & \rho vk & \rho wk \\ \rho u\omega & \rho v\omega & \rho w\omega \\ 0 & 0 & 0 \end{pmatrix}$$

The diffusive flux terms are given by

$$G^{RANS} = \begin{pmatrix} 0 & 0 & 0 \\ \tau^t_{xx} & \tau^t_{xy} & \tau^t_{xz} \\ \tau^t_{xy} & \tau^t_{yy} & \tau^t_{yz} \\ \tau^t_{xz} & \tau^t_{yz} & \tau^t_{zz} \\ \tau^t_{xx}u + \tau^t_{xy}v + \tau^t_{xz}w + k^t\frac{\partial T}{\partial x} & \tau^t_{xy}u + \tau^t_{yy}v + \tau^t_{yz}w + k^t\frac{\partial T}{\partial y} & \tau^t_{xz}u + \tau^t_{yz}v + \tau^t_{zz}w + k^t\frac{\partial T}{\partial z} \\ (\mu + \sigma_k\mu_t)\frac{\partial k}{\partial x} & (\mu + \sigma_k\mu_t)\frac{\partial k}{\partial y} & (\mu + \sigma_k\mu_t)\frac{\partial k}{\partial z} \\ (\mu + \sigma_\omega\mu_t)\frac{\partial \omega}{\partial x} & (\mu + \sigma_\omega\mu_t)\frac{\partial \omega}{\partial y} & (\mu + \sigma_\omega\mu_t)\frac{\partial \omega}{\partial z} \\ 0 & 0 & 0 \end{pmatrix}$$

The source flux terms are given by

$$S^{RANS} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ P_m - \beta^*\rho\omega k + \beta^*\rho\omega_{amb}k_{amb} \\ \frac{\gamma_m}{\nu_t}P_m - \beta\rho\omega^2 + 2\left(1 - F_1\right)\frac{\rho\sigma_{\omega 2}}{\omega}\left(\nabla k \cdot \nabla\omega\right) + \beta\rho\omega^2_{amb} \\ 0 \end{pmatrix}$$

10.2 Hybrid RANS LES

Hybrid RANS LES combines elements of both Reynolds-Averaged Navier-Stokes (RANS) and Large Eddy Simulation (LES). The key idea is to use RANS in regions where the turbulent length scales are smaller than a specified cutoff length, and LES in regions where the turbulent length scales are larger.

Detached Eddy Simulation DES

In DES, the determination of the cutoff lengthscale is critical. One common approach is to use a blending function to smoothly transition between RANS and LES modes based on the grid resolution. The lengthscale can be defined as:

$$L_{\text{DES}} = \min\left(L_{LES}, L_{\text{RANS}}\right) \tag{10.42}$$

$$L_{\text{LES}} = C_{DES}\Delta$$

Delayed Detached Eddy Simulation DDES

Delayed Detached Eddy Simulation (DDES) is an extension of Detached Eddy Simulation (DES) that introduces a delay in the transition from RANS to LES modes. The lengthscale equation for DDES is typically expressed as follows:

$$L_{\text{DDES}} = L_{RANS} - f_d \max\left(0, L_{RANS} - L_{LES}\right)$$

$$L_{\text{LES}} = C_{DES}\Delta$$

$$f_d = 1 - \tanh\left[\left(C_{d1} r_d\right)^{C_{d2}}\right]$$

$$r_d = \frac{\nu_t + \nu}{\kappa^2 d_w^2 \sqrt{0.5\left(S^2 + \Omega^2\right)}}$$

$$C_{d1} = 20$$

$$C_{d2} = 3$$

The DDES approach aims to delay the transition from RANS to LES in regions where the grid is still too coarse to resolve the smallest turbulent structures. It provides a more gradual transition compared to standard DES, making it potentially more suitable for practical engineering simulations.

Improved Delayed Detached Eddy Simulation IDDES

$$L_{\text{IDDES}} = \tilde{f}_d L_{RANS} + \left(1 - \tilde{f}_d\right) L_{LES}$$

$$\tilde{f}_d = \max\{\left(1 - f_{dt}\right), f_b\}$$

$$f_{dt} = 1 - \tanh\left[\left(C_{dt1} r_{dt}\right)^{C_{dt2}}\right]$$

$$r_{dt} = \frac{\nu_t}{\kappa^2 d_w^2 \sqrt{0.5\left(S^2 + \Omega^2\right)}}$$

$$f_b = \min\{2\exp\left(-9\alpha^2\right), 1.0\}$$

$$\alpha = 0.25 - d_w/\Delta$$

$$C_{dt1} = 20$$

$$C_{dt2} = 3$$

10.3 Diffusive Jacobian

To reduce the stencil size we make use of a thin shear layer TSL gradient approximation at the face, therefore the combined viscous and reynolds stress terms can be represented as

$$\tau_{xx} = 2(\mu + \mu_t)\left(\frac{\partial u}{\partial x} - \frac{1}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right)\right) - \frac{2}{3}\rho k \tag{10.43}$$

$$= (\mu + \mu_t)\left(\frac{4}{3}\frac{u_j - u_i}{d_{ij}}\Delta x_x - \frac{2}{3}\frac{v_j - v_i}{d_{ij}}\Delta x_y - \frac{2}{3}\frac{w_j - w_i}{d_{ij}}\Delta x_z\right) - \frac{2}{3}\rho k \tag{10.44}$$

©Zenotech Ltd

$$\tau_{yy} = 2(\mu+\mu_t)\left(\frac{\partial v}{\partial y} - \frac{1}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right)\right) - \frac{2}{3}\rho k \tag{10.45}$$

$$= (\mu+\mu_t)\left(-\frac{2}{3}\frac{u_j - u_i}{d_{ij}}\Delta x_x + \frac{4}{3}\frac{v_j - v_i}{d_{ij}}\Delta x_y - \frac{2}{3}\frac{w_j - w_i}{d_{ij}}\Delta x_z\right) - \frac{2}{3}\rho k \tag{10.46}$$

$$\tau_{zz} = 2(\mu+\mu_t)\left(\frac{\partial w}{\partial z} - \frac{1}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right)\right) - \frac{2}{3}\rho k \tag{10.47}$$

$$= (\mu+\mu_t)\left(-\frac{2}{3}\frac{u_j - u_i}{d_{ij}}\Delta x_x - \frac{2}{3}\frac{v_j - v_i}{d_{ij}}\Delta x_y + \frac{4}{3}\frac{w_j - w_i}{d_{ij}}\Delta x_z\right) - \frac{2}{3}\rho k \tag{10.48}$$

$$\tau_{xy} = 2(\mu+\mu_t)\left(\frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right) = (\mu+\mu_t)\left(\frac{u_j - u_i}{d_{ij}}\Delta x_y + \frac{v_j - v_i}{d_{ij}}\Delta x_x\right) \tag{10.49}$$

$$\tau_{xz} = 2(\mu+\mu_t)\left(\frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)\right) = (\mu+\mu_t)\left(\frac{u_j - u_i}{d_{ij}}\Delta x_z + \frac{w_j - w_i}{d_{ij}}\Delta x_x\right) \tag{10.50}$$

$$\tau_{yz} = 2(\mu+\mu_t)\left(\frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right)\right) = (\mu+\mu_t)\left(\frac{v_j - v_i}{d_{ij}}\Delta x_z + \frac{w_j - w_i}{d_{ij}}\Delta x_y\right) \tag{10.51}$$

Assuming

$$T = \frac{e}{C_v} \tag{10.52}$$

The total heat flux can be expressed by using equations from Chapter 3 in a form more suitable for subsequent manipulation

$$\begin{aligned}(k+k^t)\frac{\partial T}{\partial x} &= C_p\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{e}{C_v}\right) \\ &= \frac{C_p}{C_v}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{\rho E}{\rho} - \frac{V^2}{2}\right) \\ &= \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{\rho E}{\rho} - \frac{\rho^2 V^2}{\rho^2 2}\right)\end{aligned} \tag{10.53}$$

The jacobian of the combined diffusive flux matrix is

$$\frac{\partial G \cdot n}{\partial(\rho)} = \begin{pmatrix} 0 \\ -(\mu+\mu_t)\left[\left(\frac{4}{3}\frac{\partial u/\rho}{\partial x} - \frac{2}{3}\frac{\partial v/\rho}{\partial y} - \frac{2}{3}\frac{\partial w/\rho}{\partial z}\right)n_x + \left(\frac{\partial u/\rho}{\partial y} + \frac{\partial v/\rho}{\partial x}\right)n_y + \left(\frac{\partial u/\rho}{\partial z} + \frac{\partial w/\rho}{\partial x}\right)n_z\right] \\ -(\mu+\mu_t)\left[\left(\frac{\partial u/\rho}{\partial y} + \frac{\partial v/\rho}{\partial x}\right)n_x + \left(\frac{4}{3}\frac{\partial v/\rho}{\partial y} - \frac{2}{3}\frac{\partial u/\rho}{\partial x} - \frac{2}{3}\frac{\partial w/\rho}{\partial z}\right)n_y + \left(\frac{\partial v/\rho}{\partial z} + \frac{\partial w/\rho}{\partial y}\right)n_z\right] \\ -(\mu+\mu_t)\left[\left(\frac{\partial u/\rho}{\partial z} + \frac{\partial w/\rho}{\partial x}\right)n_x + \left(\frac{\partial v/\rho}{\partial z} + \frac{\partial w/\rho}{\partial y}\right)n_y + \left(\frac{4}{3}\frac{\partial w/\rho}{\partial z} - \frac{2}{3}\frac{\partial u/\rho}{\partial x} - \frac{2}{3}\frac{\partial v/\rho}{\partial y}\right)n_z\right] \\ m_{15} \\ -(\mu+\sigma_k\mu_t)\left[\frac{\partial k/\rho}{\partial x}n_x + \frac{\partial k/\rho}{\partial y}n_y + \frac{\partial k/\rho}{\partial z}n_z\right] \\ -(\mu+\sigma_\omega\mu_t)\left[\frac{\partial \omega/\rho}{\partial x}n_x + \frac{\partial \omega/\rho}{\partial y}n_y + \frac{\partial \omega/\rho}{\partial z}n_z\right] \\ 0 \end{pmatrix} \tag{10.54}$$

©Zenotech Ltd

$$
\begin{aligned}
m_{15} = -(\mu+\mu_t)\Bigg[&u\left(\frac{4}{3}\frac{\partial u/\rho}{\partial x}-\frac{2}{3}\frac{\partial v/\rho}{\partial y}-\frac{2}{3}\frac{\partial w/\rho}{\partial z}\right)\\
&+v\left(\frac{\partial u/\rho}{\partial y}+\frac{\partial v/\rho}{\partial x}\right)\\
&+w\left(\frac{\partial u/\rho}{\partial z}+\frac{\partial w/\rho}{\partial x}\right)\Bigg]n_x\\
&-\frac{1}{\rho}(u\tau_{xx}+v\tau_{xy}+w\tau_{xz})n_x-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{E}{\rho}-\frac{V^2}{\rho}\right)n_x\\
-(\mu+\mu_t)\Bigg[&u\left(\frac{\partial u/\rho}{\partial y}+\frac{\partial v/\rho}{\partial x}\right)\\
&+v\left(\frac{4}{3}\frac{\partial v/\rho}{\partial y}-\frac{2}{3}\frac{\partial u/\rho}{\partial x}-\frac{2}{3}\frac{\partial w/\rho}{\partial z}\right)\\
&+w\left(\frac{\partial v/\rho}{\partial z}+\frac{\partial w/\rho}{\partial y}\right)\Bigg]n_y\\
&-\frac{1}{\rho}(u\tau_{xy}+v\tau_{yy}+w\tau_{yz})n_y-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial y}\left(\frac{E}{\rho}-\frac{V^2}{\rho}\right)n_y\\
-(\mu+\mu_t)\Bigg[&u\left(\frac{\partial u/\rho}{\partial z}+\frac{\partial w/\rho}{\partial x}\right)\\
&+v\left(\frac{\partial v/\rho}{\partial z}+\frac{\partial w/\rho}{\partial y}\right)\\
&+w\left(\frac{4}{3}\frac{\partial w/\rho}{\partial z}-\frac{2}{3}\frac{\partial u/\rho}{\partial x}-\frac{2}{3}\frac{\partial v/\rho}{\partial y}\right)\Bigg]n_z\\
&-\frac{1}{\rho}(u\tau_{xz}+v\tau_{yz}+w\tau_{zz})n_z-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial z}\left(\frac{E}{\rho}-\frac{V^2}{\rho}\right)n_z
\end{aligned}
$$

$$
\frac{\partial G\cdot n}{\partial(\rho u)}=\begin{pmatrix}
0\\
(\mu+\mu_t)\left[\frac{4}{3}\frac{\partial 1/\rho}{\partial x}n_x+\frac{\partial 1/\rho}{\partial y}n_y+\frac{\partial 1/\rho}{\partial z}n_z\right]\\
(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial y}n_x-\frac{2}{3}\frac{\partial 1/\rho}{\partial x}n_y\right]\\
(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial z}n_x-\frac{2}{3}\frac{\partial 1/\rho}{\partial x}n_z\right]\\
m_{25}\\
0\\
0\\
0
\end{pmatrix}
\tag{10.55}
$$

$$
\begin{aligned}
m_{25} = (\mu+\mu_t)\Bigg[& u\left(\frac{4}{3}\frac{\partial 1/\rho}{\partial x}n_x + \frac{\partial 1/\rho}{\partial y}n_y + \frac{\partial 1/\rho}{\partial z}n_z\right) \\
& + v\left(\frac{\partial 1/\rho}{\partial y}n_x - \frac{2}{3}\frac{\partial 1/\rho}{\partial x}n_y\right) \\
& + w\left(\frac{\partial 1/\rho}{\partial z}n_x - \frac{2}{3}\frac{\partial 1/\rho}{\partial x}n_z\right)\Bigg] \\
& + \frac{1}{\rho}(\tau_{xx}n_x + \tau_{xy}n_y + \tau_{xz}n_z) \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{u}{\rho}\right)n_x \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial y}\left(\frac{u}{\rho}\right)n_y \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial z}\left(\frac{u}{\rho}\right)n_z
\end{aligned}
$$

$$
\frac{\partial G\cdot n}{\partial(\rho v)} = \begin{pmatrix}
0 \\
(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial x}n_y - \frac{2}{3}\frac{\partial 1/\rho}{\partial y}n_x\right] \\
(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial x}n_x + \frac{4}{3}\frac{\partial 1/\rho}{\partial y}n_y + \frac{\partial 1/\rho}{\partial z}n_z\right] \\
(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial z}n_y - \frac{2}{3}\frac{\partial 1/\rho}{\partial y}n_z\right] \\
m_{35} \\
0 \\
0 \\
0
\end{pmatrix} \tag{10.56}
$$

$$
\begin{aligned}
m_{35} = (\mu+\mu_t)\Bigg[& u\left(\frac{\partial 1/\rho}{\partial x}n_y - \frac{2}{3}\frac{\partial 1/\rho}{\partial y}n_x\right) \\
& + v\left(\frac{\partial 1/\rho}{\partial x}n_x + \frac{4}{3}\frac{\partial 1/\rho}{\partial y}n_y + \frac{\partial 1/\rho}{\partial z}n_z\right) \\
& + w\left(\frac{\partial 1/\rho}{\partial z}n_y - \frac{2}{3}\frac{\partial 1/\rho}{\partial y}n_z\right)\Bigg] \\
& + \frac{1}{\rho}(\tau_{xy}n_x + \tau_{yy}n_y + \tau_{yz}n_z) \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{v}{\rho}\right)n_x \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial y}\left(\frac{v}{\rho}\right)n_y \\
& - \gamma\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial z}\left(\frac{v}{\rho}\right)n_z
\end{aligned}
$$

©Zenotech Ltd

$$\frac{\partial G\cdot n}{\partial(\rho w)}=\begin{pmatrix}0\\(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial x}n_z-\frac{2}{3}\frac{\partial 1/\rho}{\partial z}n_x\right]\\(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial y}n_z-\frac{2}{3}\frac{\partial 1/\rho}{\partial z}n_y\right]\\(\mu+\mu_t)\left[\frac{\partial 1/\rho}{\partial x}n_x+\frac{\partial 1/\rho}{\partial y}n_y+\frac{4}{3}\frac{\partial 1/\rho}{\partial z}n_z\right]\\m_{45}\\0\\0\\0\end{pmatrix} \tag{10.57}$$

$$\begin{aligned}m_{45}=(\mu+\mu_t)\Bigg[&u\left(\frac{\partial 1/\rho}{\partial x}n_z-\frac{2}{3}\frac{\partial 1/\rho}{\partial z}n_x\right)\\&+v\left(\frac{\partial 1/\rho}{\partial y}n_z-\frac{2}{3}\frac{\partial 1/\rho}{\partial z}n_y\right)\\&+w\left(\frac{\partial 1/\rho}{\partial x}n_x+\frac{\partial 1/\rho}{\partial y}n_y+\frac{4}{3}\frac{\partial 1/\rho}{\partial z}n_z\right)\Bigg]\\&+\frac{1}{\rho}\left(\tau_{xz}n_x+\tau_{yz}n_y+\tau_{zz}n_z\right)\\&-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial x}\left(\frac{w}{\rho}\right)n_x\\&-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial y}\left(\frac{w}{\rho}\right)n_y\\&-\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\frac{\partial}{\partial z}\left(\frac{w}{\rho}\right)n_z\end{aligned}$$

$$\frac{\partial G\cdot n}{\partial(\rho E)}=\begin{pmatrix}0\\0\\0\\0\\\gamma\left(\frac{\mu}{Pr}+\frac{\mu_t}{Pr_t}\right)\left(\frac{\partial 1/\rho}{\partial x}n_x+\frac{\partial 1/\rho}{\partial y}n_y+\frac{\partial 1/\rho}{\partial z}n_z\right)\\0\\0\\0\end{pmatrix} \tag{10.58}$$

$$\frac{\partial G\cdot n}{\partial(\rho k)}=\begin{pmatrix}0\\-\frac{2}{3}n_x\\-\frac{2}{3}n_y\\-\frac{2}{3}n_z\\0\\(\mu+\sigma_k\mu_t)\left[\frac{\partial 1/\rho}{\partial x}n_x+\frac{\partial 1/\rho}{\partial y}n_y+\frac{\partial 1/\rho}{\partial z}n_z\right]\\0\\0\end{pmatrix} \tag{10.59}$$

$$\frac{\partial G \cdot n}{\partial(\rho\omega)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ (\mu + \sigma_\omega \mu_t)\left[\frac{\partial 1/\rho}{\partial x} n_x + \frac{\partial 1/\rho}{\partial y} n_y + \frac{\partial 1/\rho}{\partial z} n_z\right] \\ 0 \end{pmatrix} \tag{10.60}$$

$$\frac{\partial G \cdot n}{\partial(\rho\phi)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{10.61}$$

10.4 Source Term Jacobian

It is common practice in turbulence modelling to ensure that the source term Jacobian is positive thereby avoiding the use of a limiter (Spalart Allmaras, 1992). Since the source term is subtracted from the residual, only negative terms are kept.

$$\frac{\partial S^{RANS}}{\partial\left(\rho k\right)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\beta^* \omega \\ 0 \\ 0 \end{pmatrix} \tag{10.62}$$

$$\frac{\partial S^{RANS}}{\partial\left(\rho\omega\right)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\beta^* k \\ -2\beta\omega + \min\left(-2\left(1 - F_1\right)\frac{\sigma_{\omega 2}}{\omega^2}\left(\nabla k \cdot \nabla\omega\right), 0\right) \\ 0 \end{pmatrix} \tag{10.63}$$

This treatment has a serious deficiency, it may be the case that the production and destruction Jacobians are both very large, but almost cancel each other out. But from (5.52) a large contribution will be made to the Jacobian, which may be interpreted as a reduction of the time step. In such a situation very poor convergence will occur. However in this case stability considerations outweigh efficiency considerations.

©Zenotech Ltd

10.5 Wall Function

For a quasi-equilibrium boundary layer, the region between the wall and the outer edge of the logarithmic layer is universal. Near the wall derivatives in the streamwise direction can be neglected and the flow and turbulence variables depend only on the y direction, normal to the wall.

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}},\ u^+ = \frac{U}{u_\tau},\ y^+ = \frac{y u_\tau}{\nu},\ \nu_\tau^+ = \frac{\nu_\tau}{\nu},\ p^+ = \frac{\nu}{\rho u_\tau^3}\frac{dp}{dx},\ k^+ = \frac{k}{u_\tau^2},\ \omega^+ = \frac{\omega\nu}{u_\tau^2} \tag{10.64}$$

For reference

$$u^* = u^+ C_\mu^{1/4}\sqrt{k}/u_\tau \tag{10.65}$$

$$y^* = y^+ C_\mu^{1/4}\sqrt{k}/u_\tau \tag{10.66}$$

Over a flat plate at zero pressure gradient the RANS equations simplify in the region near the wall to the equilibrium stress balance model leading to universal near-wall solutions

$$\frac{d}{dy}\left((\mu+\mu_t)\frac{dU}{dy}\right) = f \tag{10.67}$$

where $f = 0$ for equilibrium stress balance model and $f = 1/\rho\, dp/dx$ otherwise. Integration along the wall normal coordinate for $f = 0$.

$$(\mu+\mu_t)\frac{dU}{dy} = \rho u_\tau^2 = \tau_w \tag{10.68}$$

Non-dimensionalise

$$\left(1+\nu_t^+\right)\frac{du^+}{dy^+} = 1 \tag{10.69}$$

$$\nu_t^+ = \frac{dy^+}{du^+} - 1 \tag{10.70}$$

Reynolds number obtained from the first cell

$$Re_y = \frac{y_1 u_1}{\nu} = y_1^+ u_1^+ \tag{10.71}$$

Incompressible adiabatic law of the wall valid for logarithmic region valid for $0 < y^+ < 1000$

$$U = u_\tau\left[\frac{1}{\kappa}\log\left(\frac{u_\tau y}{\nu}\right) + B\right] \tag{10.72}$$

Non-dimensionalise

$$u^+ = \frac{1}{\kappa}\log\left(y^+\right) + B \tag{10.73}$$

where $B = 5.3271$ for hydraulically smooth walls
The velocity in the sublayer $0 < y^+ < 10$ is given by

$$u^+ = y^+ \tag{10.74}$$

Boundary Layer Profile

The boundary layer profile equation proposed by Spalding

$$y^+ = u^+ + e^{-\kappa B}\left(e^{\kappa u^+} - 1 - \kappa u^+ - \frac{(\kappa u^+)^2}{2} - \frac{(\kappa u^+)^3}{6}\right) \tag{10.75}$$

$$\frac{dy^+}{du^+} = 1 - \kappa e^{-\kappa B}\left(e^{\kappa u^+} - 1 - \kappa u^+ - \frac{(\kappa u^+)^2}{2}\right) \tag{10.76}$$

Compressibility and heat transfer effects can be included by replacing the incompressible adiabatic law of the wall term $e^{-\kappa B}e^{\kappa u^+}$ in Spalding's equations with the outer velocity form the White and Christoph.

$$u^+ = \frac{1}{2\Gamma}\left[\beta + Q\sin\left(\phi + \frac{\sqrt{\Gamma}}{\kappa} ln\left(\frac{y^+}{y_o^+}\right)\right)\right] \tag{10.77}$$

where

$$\Gamma = \frac{ru_\tau^2}{2c_pT_w} \tag{10.78}$$

$$\beta = \frac{q_w\mu_w}{\rho_w T_w k_w u_\tau} \tag{10.79}$$

$$\varphi = \sin^{-1}\left(\frac{-\beta}{Q}\right) \tag{10.80}$$

$$Q = \left(\beta^2 + 4\Gamma\right)^{1/2} \tag{10.81}$$

$$y_o^+ = e^{=\kappa B} \tag{10.82}$$

$$r = Pr^{1/3} \tag{10.83}$$

The nondimensional parameter Γ models the compressibility effects and parameter β models the heat transfer effects. The outer velocity form of the White and Christoph reduces to the standard law of the wall for incompressible adiabatic flow can be then be written as

$$y^+ = u^+ + y_{white}^+ - e^{-\kappa B}\left(1 + \kappa u^+ + \frac{(\kappa u^+)^2}{2} + \frac{(\kappa u^+)^3}{6}\right) \tag{10.84}$$

where

$$y_{white}^+ = e^{\frac{\kappa}{\sqrt{\Gamma}}\left[\sin^{-1}\left(\frac{2\Gamma u^+ - \beta}{Q}\right) - \varphi\right]}e^{-\kappa B} \tag{10.85}$$

The temperature distribution within the boundary layer is given by the Crocco-Busemann equation

$$T = T_w\left(1 + \beta u^+ - \Gamma\left(u^+\right)^2\right) \tag{10.86}$$

For adiabatic wall cases $\beta = 0$ and the Crocco-Busemann equation reduces to

$$T = T_w\left(1 + \frac{(\gamma - 1)\, r}{2}u^2\right) \tag{10.87}$$

©Zenotech Ltd

Boundary Conditions

Omega at the wall

$$\omega_{vis}^{+} = \frac{6}{\beta_1 \left(y^{+}\right)^2} \tag{10.88}$$

$$\omega_{log}^{+} = \frac{1}{\kappa\sqrt{\beta^*}y^{+}} \tag{10.89}$$

$$\omega^{+} = \sqrt{\left(\omega_{vis}^{+}\right)^2 + \left(\omega_{log}^{+}\right)^2} \tag{10.90}$$

Turbulent kinetic energy k at the wall

$$k^{+} = \nu_t^{+}\omega^{+} = \left(\frac{dy^{+}}{du^{+}} - 1\right)\omega^{+} \tag{10.91}$$

See [9] for more details

Gradient Boundary Conditions

The gradient at the wall calculated using the material in section 4.7 will return the incorrect shear stress for wall functions due to $V = 0$ at the surface. Therefore the gradient in the interior and halo cell adjacent to the wall are modified to return the correct shear stress. Construct $\frac{dU}{dy}$ in the halo cell to return the correct wall stress assuming constant shear

$$\frac{dU}{dy} = \frac{\rho u_\tau^2}{(\mu + \mu_t)} = \frac{\tau_w}{(\mu + \mu_t)} \tag{10.92}$$

$$\left(\frac{dU}{dy}\right)_{halo} = 2\left(\frac{dU}{dy}\right)_{wall} - \left(\frac{dU}{dy}\right)_{interior} \tag{10.93}$$

Assuming $V = 0$ at the surface

$$\left(\frac{dU}{dy}\right)_{local} = (\nabla V \cdot n) \cdot V_{tangent} \tag{10.94}$$

The gradients are reflected therefore

$$\left(\frac{dU}{dy}\right)_{halolocal} = -\left(\frac{dU}{dy}\right)_{local} \tag{10.95}$$

$$\nabla V_{halo} = -\nabla V_{interior} + \left(n\left(\left(\frac{dU}{dy}\right)_{halo} - \left(\frac{dU}{dy}\right)_{halolocal}\right)\right)V_{tangent} \tag{10.96}$$

$$\nabla V_{interior} = \nabla V_{interior} + \left(n\left(\left(\frac{dU}{dy}\right)_{interior} - \left(\frac{dU}{dy}\right)_{local}\right)\right)V_{tangent} \tag{10.97}$$

Roughness

Roughness k_r is the equivalent sand grain roughness height and μ_r is the associated dynamic viscosity.

$$k_r^{+} = \frac{k_r u_\tau}{\nu} \tag{10.98}$$

Law of the wall recast in roughness variables

$$u^+ = \left[\frac{1}{\kappa}\log\left(y^+\right) + B_r\left(k_r^+\right)\right] \tag{10.99}$$

$$u^+ = \left[\frac{1}{\kappa}\log\left(\frac{y}{k_r}\frac{k_r u_\tau}{\nu}\right) + B_r\left(k_r^+\right)\right] \tag{10.100}$$

$$u^+ = \left[\frac{1}{\kappa}\log\left(\frac{y}{k_r}\right) + \frac{1}{\kappa}\log k_r^+ + B_r\left(k_r^+\right)\right] \tag{10.101}$$

See [4] for details

The Richard and Hoxey Law assumes that the $y = 0$ is at the top of the roughness objects at roughness length d_0. This enables the specification of a roughness value that is greater than the first cell height with the assumption of fully rough conditions $k_r^+ > 90.0$

$$u^+ = \frac{1}{\kappa}\log\left(\frac{y + d_0}{d_0}\right) \tag{10.102}$$

Recast into the same form as the equation above

$$u^+ = \frac{1}{\kappa}\log\left(y + d_0\right) - \frac{1}{\kappa}\log\left(d_0\right) \tag{10.103}$$

Roughness length $d_0 = k_r e^{-B\kappa}$ with $B = 8.5$ for fully rough conditions.

$$u^+ = \frac{1}{\kappa}\log\left(y + d_0\right) - \frac{1}{\kappa}\log\left(k_r e^{-B\kappa}\right) \tag{10.104}$$

$$u^+ = \frac{1}{\kappa}\log\left(y + k_r e^{-B\kappa}\right) - \frac{1}{\kappa}\log\left(k_r\right) + B \tag{10.105}$$

$$u^+ = \frac{1}{\kappa}\log\left(y/k_r + e^{-B\kappa}\right) + B \tag{10.106}$$

Comparing to the law of the wall recast in roughness variables above

$$B = B_r + \frac{1}{\kappa}\log\left(k_r^+\right) \tag{10.107}$$

This is the height where the velocity according to the log law falls to zero. This is approximately equal to 0.1 times the height of the roughness elements.

If the height of the first cell is much greater than the roughness value then d_0 can be replaced with the displacement height d_1 that represents the offset to allow for the upward

©Zenotech Ltd

displacement of the flow by the surface objects. For atmospheric flows the displacement height is around two-thirds of the canopy height.

$$u^+ = \frac{1}{\kappa}\log\left(y - d_1\right) - \frac{1}{\kappa}\log\left(k_r\right) + B \tag{10.108}$$

Boundary conditions from [4]

$$k_w = \frac{u_\tau^2}{\sqrt{C_\mu}}\min\left(1, \frac{k_r^+}{90}\right) \tag{10.109}$$

$$\omega_w = \frac{u_\tau}{\sqrt{C_\mu}\kappa d_0 \phi_{r2}} \tag{10.110}$$

$$\phi_{r2} = \min\left[\min\left[1, \left(\frac{k_r^+}{20}\right)^{\frac{2}{3}}\right]\min\left[1, \left(\frac{k_r^+}{45}\right)^{\frac{1}{4}}\right]\min\left[1, \left(\frac{k_r^+}{60}\right)^{\frac{1}{4}}\right]\right] \tag{10.111}$$

From [3]

$$B_r = B \qquad k_r^+ < 2.25 \tag{10.112}$$

$$B_r = \xi\left(8.5 - \frac{1}{\kappa}\log k_r^+ - B\right) + B \qquad 2.25 \leq k_r^+ \leq 90 \tag{10.113}$$

$$B_r = 8.5 - \frac{1}{\kappa}\log k_r^+ \qquad k_r^+ > 90.0 \tag{10.114}$$

where $\xi = \sin\left(\frac{\frac{\pi}{2}\log\left(\frac{k_r^+}{2.225}\right)}{\log\left(\frac{90}{2.25}\right)}\right)$

Temperature Wall Function

$$T^+ = \frac{\left(T_w - T\right)\rho C_p u_\tau}{\dot{q}} \tag{10.115}$$

$$T^* = \frac{\left(T_w - T\right)\rho C_p C_\mu^{1/4}\sqrt{k}}{\dot{q}} \tag{10.116}$$

Assuming Fourier's Law

$$\dot{q} = -\frac{C_p \mu}{Pr}\frac{\partial T}{\partial x} \tag{10.117}$$

For $y^+ < 11.639$

$$T^+ = Pr\, y^+ + \frac{1}{2}\rho\frac{u_\tau}{\dot{q}} Pr\, U^2 \tag{10.118}$$

For $y^+ > 11.639$

$$T^+ = Pr_t \left[\frac{1}{\kappa} \log\left(y^+\right) + B + P \right] + \frac{1}{2} \rho \frac{u_\tau}{\dot{q}} \left[Pr\, U^2 + \left(Pr - Pr_t\right) U_c^2 \right] \tag{10.119}$$

where U_c is the velocity at $y^+ = 11.639$

The additional terms on the right are due to heating by viscous dissipation for compressible flows. See [18]

$P \approx -1.12$ is function of the ratio of the molecular to turbulent Prandtl number and for smooth wall it is as follows. See [8].

$$P = 9.24 \left[\left(\frac{Pr}{Pr_t} \right)^{3/4} - 1 \right] \left[1 + 0.28 e^{-0.007 Pr/Pr_t} \right] \tag{10.120}$$

For rough walls P_{rough} is defined as follows

$$P_{rough} = 3.15 Pr^{0.695} \left(\frac{1}{B_r} - \frac{1}{B} \right)^{0.359} + \left(\frac{B_r}{B} \right)^{0.6} P \tag{10.121}$$

ABL Stability

For Atmospheric Boundary Layer wall functions we include the ability to model boundary layer stability. This assumes that the boundary layer is fully turbulent $y^+ > 30$ in the first cell adjacent to the wall. L is the Obukhov Length

$$u^+ = \left[\frac{1}{\kappa} \log\left(y^+\right) + B_r\left(k_r^+\right) - \psi_m\left(\frac{y}{L}\right) \right] \tag{10.122}$$

For stable conditions $y/L > 0\,; 0 < R_i < 0.2$

$$\frac{y^+}{L} = \frac{u_\tau}{\nu} \frac{R_i}{1 - 5R_i} \tag{10.123}$$

$$\psi_m\left(\frac{y}{L}\right) = -5\frac{y}{L} \tag{10.124}$$

$$\frac{du^+}{dy^+} = \frac{1}{\kappa y^+} + \frac{5}{L}\frac{\nu}{u_\tau} \tag{10.125}$$

For unstable conditions $y/L < 0\,; R_i < 0$

$$\frac{y^+}{L} = \frac{u_\tau}{\nu} R_i \tag{10.126}$$

$$\psi_m\left(\frac{y}{L}\right) = 2\log\left(\frac{1+X}{2}\right) + \log\left(\frac{1+X^2}{2}\right) - 2\arctan\left(X\right) + \frac{\pi}{2} \tag{10.127}$$

©Zenotech Ltd

where

$$X = \left(1 - 16\frac{y}{L}\right)^{1/4} \tag{10.128}$$

$$\frac{du^+}{dy^+} = \frac{1}{\kappa y^+} - \left[-\frac{8}{L\left(\sqrt{A}+1\right)\sqrt{A}} + \frac{8}{L\left(\sqrt{A}+1\right)A^{3/4}} - \frac{8}{L\left(A^{1/4}+1\right)A^{3/4}} \right] \tag{10.129}$$

where

$$A = 1 - 16\frac{y^+}{L}\frac{\nu}{u_\tau} \tag{10.130}$$

Temperature

$$T^+ = Pr_t \left[\frac{1}{\kappa}\log\left(y^+\right) + B + P_r - \psi_h\left(\frac{y}{L}\right) \right] \tag{10.131}$$

10.6 Menter SST SAS

The SAS approach developed by Menter and co-workers resulted from revisiting the K-KL model of Rotta where K is the turbulent kinetic energy and L the traditional notation for the macro-length of turbulence.

The model can be expressed as a modification of the SST model by the inclusion of an additional source term to the ω equation.

$$Q_{SAS} = \max\left\{ \rho\zeta_2\kappa S^2 \left(\frac{L_{RANS}}{L_{vk}}\right)^2 - C\frac{2\rho k}{\sigma_\phi}\max\left[\frac{1}{\omega^2}\frac{\partial\omega}{\partial x_j}\frac{\partial\omega}{\partial x_j}, \frac{1}{k^2}\frac{\partial k}{\partial x_j}\frac{\partial k}{\partial x_j}\right], 0 \right\} \tag{10.132}$$

$$L_{vk} = \kappa\frac{|U'|}{|U''|} \qquad |U'| = \sqrt{\frac{\partial U_i}{\partial x_j}\frac{\partial U_i}{\partial x_j}} \qquad |U''| = \sqrt{\frac{\partial^2 U_i}{\partial x_j \partial x_j}\frac{\partial^2 U_i}{\partial x_j \partial x_j}} \tag{10.133}$$

$$S = \sqrt{2S_{ij}S_{ij}} \qquad S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \tag{10.134}$$

where $\zeta_2 = 3.52$, $C = 2$ and $\sigma_\phi = 2/3$

In certain situations L_{RANS} can become very large if ω is small and therefore Q_{SAS} will be too large. Physically $L_{RANS} \approx L_{vk}$.

$$L_{RANS} = \begin{cases} L_{RANS} & L_{RANS} \leq 10^5 L_{vk} \\ L_{vk} & L_{RANS} > 10^5 L_{vk} \end{cases} \tag{10.135}$$

High frequency limiting is handled via the WALE model.

1. The model contains two length scales, the classical one related to the first derivative of the resolved velocity, and a second one related to higher derivatives of the resolved velocity.

2. a) The model provides RANS performance in stable flow regions (without explicit grid or time-step dependency).
   b) The model allows the break-up of large unsteady structures into a turbulent spectrum (without explicit grid or time-step dependency).
   c) The model provides proper damping of resolved turbulence at the resolution limit of the grid. This requires some information about the step size of the grid, either explicitly or implicitly.

WALE Model

For high frequency analysis, the SAS model can suffer from giving too little damping without proper high wavenumber treatment. To eliminate any possible criticism of the SAS behavior in the infinite grid limit, Egorov and Menter prefer to limit the predicted eddy viscosity by the eddy viscosity of an LES model. The LES model suggested by Menter and Egorov, is the Wall-Adapting Local Eddy-viscosity (WALE) model of Nicoud and Ducros. The WALE eddy viscosity is computed in a similar fashion as the Smagorinsky method, yet the WALE model is able to respond to near-wall effects and correctly lower the eddy viscosity in laminar regions. The WALE eddy viscosity can be expressed as the following

$$\mu_t = \rho C_w^2 \Delta^2 \frac{\left(G_{ij}^d G_{ij}^d\right)^{3/2}}{\left(S_{ij}S_{ij}\right)^{5/2} + \left(G_{ij}^d G_{ij}^d\right)^{5/4}} \tag{10.136}$$

G_{ij}^d is the traceless symmetric part of the square of the velocity gradient tensor

$$G_{ij}^d = \frac{1}{2}\left(g_{ij}^2 + g_{ji}^2\right) - \frac{1}{3}\delta_{ij} g_{kk}^2 \tag{10.137}$$

where $g_{ij}^2 = \partial u_i/\partial x_k \partial u_k/\partial x_j$ and $C_w = 0.5$ and $\Delta = \sqrt[3]{V}$

$$G_{ij}^d = \sum_k \frac{1}{2}\left(\frac{\partial u_i}{\partial x_k}\frac{\partial u_k}{\partial x_j} + \frac{\partial u_j}{\partial x_k}\frac{\partial u_k}{\partial x_i}\right) - \delta_{ij}\frac{1}{3}\sum_i\sum_j \frac{\partial u_i}{\partial x_j}\frac{\partial u_j}{\partial x_i} \tag{10.138}$$

©Zenotech Ltd

Chapter 11

Gravity

11.1 Force Potential

Gravity is treated as a force potential

$$F = -\nabla\varphi = -\rho g \tag{11.1}$$

By integration and introducing z the height in the direction of the gravity vector

$$\varphi = g \int \rho \, dz \tag{11.2}$$

11.2 Pressure Gradient

For a fluid in hydrostatic equilibrium

$$\frac{\partial p}{\partial z} = -\rho g \tag{11.3}$$

For an ideal gas

$$\frac{\partial p}{\partial z} = -\frac{p}{RT} g \tag{11.4}$$

11.3 Lapse Rates

For an isentropic process

$$\frac{p}{\rho^\gamma} = constant \tag{11.5}$$

Temperature

Substituting the ideal gas equation of state

$$p^{1-\gamma}T^{\gamma} = constant \tag{11.6}$$

Differentiating by parts with respect to altitude

$$T^{\gamma}\left(1-\gamma\right)p^{-\gamma}\frac{\partial p}{\partial z} + p^{1-\gamma}\gamma T^{\gamma-1}\frac{\partial T}{\partial z} = 0 \tag{11.7}$$

$$\frac{\left(1-\gamma\right)}{\gamma p}\frac{\partial p}{\partial z} + \frac{1}{T}\frac{\partial T}{\partial z} = 0 \tag{11.8}$$

Substitute the pressure gradient definition (11.4)

$$\frac{\partial T}{\partial z} = -\frac{g\left(\gamma-1\right)}{\gamma R} = -\frac{g}{C_p} = -\Gamma \approx 9.8^{\circ}K/km \tag{11.9}$$

This temperature gradient is known as the dry adiabatic lapse rate.

Integrate

$$T = T_0 - \frac{g\left(\gamma-1\right)}{\gamma R}z = T_0 - \Gamma z \tag{11.10}$$

Also,

$$\frac{\partial T}{T} = \frac{\left(\gamma-1\right)}{\gamma}\frac{\partial p}{p} \tag{11.11}$$

Integrate

$$\ln\left(\frac{T}{T_0}\right) = \frac{\left(\gamma-1\right)}{\gamma}\ln\left(\frac{p}{p_0}\right) \tag{11.12}$$

Take exponent

$$\frac{T}{T_0} = \left(\frac{p}{p_0}\right)^{\frac{(\gamma-1)}{\gamma}} \tag{11.13}$$

Density

Recall entropy equation

$$\frac{p}{\rho^{\gamma}} = constant \tag{11.14}$$

©Zenotech Ltd

or

$$p\rho^{-\gamma} = constant \tag{11.15}$$

Differentiating by parts with respect to altitude

$$\rho^{-\gamma}\frac{\partial p}{\partial z} - \gamma\rho^{-\gamma-1}p\frac{\partial \rho}{\partial z} = 0 \tag{11.16}$$

$$\frac{1}{p}\frac{\partial p}{\partial z} - \frac{\gamma}{\rho}\frac{\partial \rho}{\partial z} = 0 \tag{11.17}$$

Substitute the pressure gradient definition (11.4)

$$\frac{\partial \rho}{\partial z} = -\frac{\rho g}{\gamma RT} = -\frac{\rho g}{(\gamma-1)C_pT} = \frac{\rho\Gamma}{(\gamma-1)T} \tag{11.18}$$

Also,

$$\frac{\partial p}{p} = \gamma\frac{\partial \rho}{\rho} \tag{11.19}$$

Integrate

$$\ln\left(\frac{p}{p_0}\right) = \gamma\ln\left(\frac{\rho}{\rho_0}\right) \tag{11.20}$$

Take exponent

$$\frac{p}{p_0} = \left(\frac{\rho}{\rho_0}\right)^{\gamma} \tag{11.21}$$

Pressure

Substitute the definition of T (11.10) in (11.13)

$$p = p_0\left[1 - \frac{g(\gamma-1)}{\gamma RT_0}z\right]^{\gamma/(\gamma-1)} = p_0\left(1 - \frac{\Gamma}{T_0}z\right)^{\gamma/(\gamma-1)} \tag{11.22}$$

Summary

$$T = T_0 - \Gamma z \tag{11.23}$$

$$p = p_0 \left[1 - \frac{\Gamma}{T_0} z\right]^{\gamma/(\gamma-1)} \tag{11.24}$$

$$\rho = \rho_0 \left[1 - \frac{\Gamma}{T_0} z\right]^{1/(\gamma-1)} \tag{11.25}$$

Force Potential

Recall

$$\varphi = g \int \rho \, dz = g \int \rho_0 \left[1 - \frac{\Gamma}{T_0} z\right]^{1/(\gamma-1)} dz \tag{11.26}$$

Integrating by substitution

$$\varphi = g\rho_0 \int U^{1/(\gamma-1)} \frac{-T_0}{\Gamma} dU = -g\rho_0 \frac{T_0}{\Gamma} \frac{\gamma - 1}{\gamma} \left[1 - \frac{\Gamma}{T_0} z\right]^{\gamma/(\gamma-1)} \tag{11.27}$$

$$\varphi = p_0 \left[1 - \frac{\Gamma}{T_0} z\right]^{\gamma/(\gamma-1)} \tag{11.28}$$

©Zenotech Ltd

Chapter 12

Stratified Atmospheric Flow

The incompressible solver can be run with a deep, hydrostatically stratified atmosphere using the anelastic approximation. It is selected by setting `stratified` in the `incomp viscous` (or `incomp rans`/`incomp les`) dictionary, and requires the energy equation with potential temperature as the energy variable.

The Boussinesq branch described elsewhere holds density constant except in the buoyancy term and refers buoyancy to a single global temperature. That is valid only when the domain depth is small compared with the density scale height $H = RT/g \approx 8.4$ km. The anelastic system instead carries density as a prescribed function of height, so it remains in equilibrium through a deep column:

$$\nabla \cdot (\bar{\rho}(z) V) = 0 \tag{12.1}$$

$$b = -g \frac{\big(\theta - \bar{\theta}(z)\big)}{\bar{\theta}(z)} \tag{12.2}$$

Throughout this chapter an overbar denotes a base-state quantity, a prime denotes a perturbation about it, z is height above the reference location measured along $-g$ (so z increases upwards), and g is the gravity vector with magnitude $|g|$.

12.1 Hydrostatic Base State

The base state is held in Exner form, which is what keeps the constant-N profile a closed expression. The Exner function is

$$\pi = \left(\frac{\bar{p}}{p_0} \right)^{R/C_p} \tag{12.3}$$

and hydrostatic balance written in terms of π and the potential temperature is

$$\frac{\partial \pi}{\partial z} = -\frac{|g|}{C_p \bar{\theta}} \tag{12.4}$$

The remaining base-state quantities follow from π and $\bar{\theta}$:

$$\bar{p} = p_0 \pi^{\gamma/(\gamma-1)}, \qquad \bar{T} = \bar{\theta}\,\pi, \qquad \bar{\rho} = \frac{\bar{p}}{R\bar{T}} \tag{12.5}$$

Three base-state profiles are implemented, each specified by a surface temperature T_0 and a surface pressure p_0 at $z = 0$.

Isothermal

$\bar{T} = T_0$ everywhere, so integrating (12.4) gives

$$\pi = \exp\left(-\frac{|g|z}{C_p T_0}\right), \qquad \bar{\theta} = \frac{T_0}{\pi}, \qquad \frac{\partial\bar{\theta}}{\partial z} = \frac{|g|}{C_p \pi} \tag{12.6}$$

Constant Lapse Rate

With a prescribed environmental lapse rate $\Gamma = -\partial\bar{T}/\partial z$,

$$\bar{T} = T_0 - \Gamma z, \qquad \pi = \left(1 - \frac{\Gamma z}{T_0}\right)^{|g|/(C_p\Gamma)}, \qquad \frac{\partial\bar{\theta}}{\partial z} = \frac{1}{\pi}\left(\frac{|g|}{C_p} - \Gamma\right) \tag{12.7}$$

Note that $\partial\bar{\theta}/\partial z$ is identically zero when $\Gamma = |g|/C_p$, the dry adiabatic lapse rate written as Γ in the Gravity chapter. That is the neutral atmosphere, and it is the case for which $\bar{\theta}$ is uniform. The profile is only valid while $\bar{T} > 0$; a constant lapse rate taken high enough drives the temperature through zero.

Constant Brunt-Väisälä Frequency

This is the stratification the gravity-wave benchmarks specify. With constant N,

$$\bar{\theta} = \theta_0 \exp\left(\frac{N^2 z}{|g|}\right), \qquad \pi = 1 + \frac{|g|^2}{C_p \theta_0 N^2}\left(\exp\left(-\frac{N^2 z}{|g|}\right) - 1\right) \tag{12.8}$$

Recovering N from a General Profile

In general the buoyancy frequency is recovered from the base state as

$$N^2 = \frac{|g|}{\bar{\theta}}\frac{\partial\bar{\theta}}{\partial z} \tag{12.9}$$

It is used to size the sponge layer and by the buoyancy-aware sub-grid-scale models.

©Zenotech Ltd

12.2 Anelastic Continuity

Because $\bar{\rho} > 0$, the constraint $\nabla \cdot (\bar{\rho} V) = 0$ is equivalent to the divided form

$$\nabla \cdot V + V \cdot \nabla (\ln \bar{\rho}) = 0 \tag{12.10}$$

This form is used because the existing continuity row already assembles the discrete $\nabla \cdot V$ as a face sum. The second term is then a cell-based contribution on the velocity columns, which leaves the face fluxes – and with them the Rhie-Chow pressure-velocity coupling – untouched. Rescaling every face flux instead would disturb both.

Differentiating $\ln \bar{\rho} = \text{const} - \ln \bar{\theta} + (C_v/R) \ln \pi$ and substituting (12.4) gives the coefficient in (12.10) as

$$\frac{\partial \ln \bar{\rho}}{\partial z} = -\frac{1}{\bar{\theta}} \frac{\partial \bar{\theta}}{\partial z} - \frac{|g|}{\gamma R \bar{T}} = -\frac{N^2}{|g|} - \frac{|g|}{c^2} \tag{12.11}$$

the standard anelastic result: a stratification part and an acoustic part. Only the second survives in a neutral atmosphere, which is why $\bar{\rho}$ still varies with height even when $\bar{\theta}$ does not.

Discretely, for cell i the term contributes to the continuity row

$$\Omega_i \left. \frac{\partial \ln \bar{\rho}}{\partial z} \right|_i \hat{k} \cdot V_i, \qquad \hat{k} = -\frac{g}{|g|} \tag{12.12}$$

12.3 Buoyancy

The buoyancy source added to the momentum equation is

$$b = -g \frac{\big(\theta - \bar{\theta}(z)\big)}{\bar{\theta}(z)} \tag{12.13}$$

The sign places warm air ($\theta > \bar{\theta}$) in the $-g$ direction. This reduces exactly to the Boussinesq form $-g\beta(T - T_{ref})$ when $\bar{\theta}$ is uniform and $\beta = 1/T_{ref}$. What differs is that the reference is a function of height, so it remains an equilibrium throughout a deep column rather than balancing at one level only: a neutrally stratified atmosphere ($\theta \equiv \bar{\theta}$) gives identically zero buoyancy at every height.

12.4 Perturbation Form of the Energy Equation

The base state must be an exact solution of the *discrete* equations, not merely of the continuous ones, or it generates spurious motion. The energy diffusion operator is linear, so rather than rebuilding the matrix to act on $\theta' = \theta - \bar{\theta}$ the matrix is left acting on θ and the

diffusion of the base state is added to the right-hand side. For a face between cells i and j with discrete diffusion coefficient D_f,

$$S_i \mathrel{+}= D_f \left(\bar{\theta}_i - \bar{\theta}_j\right) \tag{12.14}$$

D_f must be the same discrete coefficient the matrix used. That is what makes the two cancel exactly rather than approximately when $\theta \equiv \bar{\theta}$. This applies to the non-orthogonal correction as well as the orthogonal part; leaving the non-orthogonal term in full-field form while its sibling is in perturbation form leaves a residual forcing on any terrain-following mesh.

12.5 Pressure Correction

With the anelastic constraint the pressure-correction equation is weighted by the base-state density at the face. Both the operator and the right-hand side carry $\bar{\rho}_f$:

$$\sum_f^{nb} \bar{\rho}_f D_f \left(p'_i - p'_j\right) = -\sum_f^{nb} \bar{\rho}_f F_f \tag{12.15}$$

where F_f is the predicted volume flux through face f and D_f the usual $(A^{-1}\Omega)\,|_f/|d_{ij}|$ Laplacian coefficient. The weighting must be applied over *all* faces of a cell, interior and boundary alike; weighting the interior faces while leaving a boundary face unweighted mixes two different constraints in the same row, and the size of the resulting error grows with the departure of $\bar{\rho}_f$ from unity, i.e. with domain depth.

The flux correction itself keeps the unweighted D_f, which is what makes the constraint close:

$$\sum_f^{nb} \bar{\rho}_f \left(F_f + F'_f\right) = \sum_f^{nb} \bar{\rho}_f F_f + \sum_f^{nb} \bar{\rho}_f D_f \left(p'_i - p'_j\right) = 0 \tag{12.16}$$

A consistency source is added at each face so that the constraint the cell enforces is the weighted one referred to the cell's own base density:

$$S_i \mathrel{+}= -\sum_f^{nb} \left(\frac{\bar{\rho}_f}{\bar{\rho}_i} - 1\right) F_f \tag{12.17}$$

This uses the *actual* face flux. Substituting the analytic equivalent $-\int_\Omega V \cdot \nabla \ln \bar{\rho}\, d\Omega$ is valid only at convergence, and using it to impose the very constraint that would make it valid does not converge.

©Zenotech Ltd

12.6 Boundary Conditions

Slip Wall and Symmetry

The velocity is mirrored about the true face normal, $V_{halo} = V_i - 2\,(V_i \cdot n)\,n$, which makes the wall flux identically zero.

For pressure the correct condition in a buoyant flow is the normal momentum balance rather than plain zero gradient,

$$\frac{\partial p'}{\partial n} = b \cdot n \tag{12.18}$$

so the halo value carries the buoyancy work over the cell-to-halo offset:

$$p_{halo} = p_i + (b \cdot n)\,\delta n, \qquad \delta n = 2\,(x_f - x_i) \cdot n \tag{12.19}$$

At rest $b \equiv 0$ and this collapses to $p_{halo} = p_i$.

For the energy equation it is the *perturbation* that is mirrored, not the full potential temperature:

$$\theta_{halo} = \theta_i + \left(\bar{\theta}_{halo} - \bar{\theta}_i\right) \tag{12.20}$$

A vertical symmetry plane is unaffected by this because both sides sit at the same height, but a horizontal one – the rigid free-slip lid every atmospheric benchmark uses – would otherwise flatten the background stratification.

12.7 Sponge Layer

To absorb upward-propagating gravity waves at the model top a height-based Rayleigh damping layer is applied above a start height z_s, with D the depth from z_s to the domain top:

$$\sigma(z) = \sigma_{max} \sin^2\left(\frac{\pi}{2} \min\left(\frac{z - z_s}{D}, 1\right)\right), \qquad z > z_s \tag{12.21}$$

and zero below. The damping is applied to momentum and to the energy equation as

$$\frac{\partial Q}{\partial t} \mathrel{+}= -\sigma(z)\,(Q - Q_{target}) \tag{12.22}$$

It is added *implicitly*, with $\sigma\Omega$ on the diagonal and $\sigma\Omega Q_{target}$ on the right-hand side, so the fixed point is $(A + \sigma\Omega)\,V = b + \sigma\Omega V_{target}$. An explicit sponge carries a $\sigma\Delta t < 1$ stability limit, and the published range of σ_{max} against a typical atmospheric timestep puts cases

directly on it. Adding to the diagonal also strengthens diagonal dominance, so it cannot destabilise the linear solve.

The sponge is deliberately *not* applied to continuity: damping mass would fight the pressure solve.

Two sizing criteria should be satisfied, and both are reported at start-up. The layer should span at least one vertical wavelength,

$$D \geq \lambda_z = \frac{2\pi U}{N} \tag{12.23}$$

and the damping rate should satisfy $1 \leq \sigma_{max}/N \leq 10$.

12.8 Resolution Requirement

A stratified simulation carrying gravity waves must resolve the vertical wavelength $\lambda_z = 2\pi U/N$, not merely the boundary layer. The vertical spacing Δz aloft is usually set by a stretching cap rather than by the wave, and it is easy to end up with a mesh that is well resolved near the ground and badly under-resolved at the height the wave occupies.

The error is second order in $\Delta z/\lambda_z$, and in a mountain-wave calculation it shows up as a vertical momentum flux that grows with height instead of being constant – an apparent violation of the Eliassen-Palm theorem rather than an obvious amplitude error, which makes it easy to misread as a spurious momentum source in the interior.

Measured on the linear hydrostatic mountain-wave case ($U = 20$ m/s, $N = 0.0196$ 1/s, so $\lambda_z = 6423$ m), varying only the vertical spacing cap and holding the near-wall spacing fixed at 20 m:

$\lambda_z/\Delta z$	M/M_H (2–12 km)	std	$d(M/M_H)/dz$ per km
8.0	1.470	0.271	+0.079
16.2	1.144	0.037	−0.005
32.2	1.023	0.030	−0.009

where $M(z) = \int \bar{\rho} u' w' \, dx$ is the Eliassen-Palm momentum flux and $M_H = -(\pi/4)\rho_0 U N h_m^2$ its analytic hydrostatic value. At 8 points per vertical wavelength the flux is 47 per cent high and grows with height; by 16 points the growth with height has gone and the amplitude error is 14 per cent; by 32 points the result is 1.023 and constant. Eight points per vertical wavelength is not enough, even though the same mesh resolves the surface layer to 20 m.

©Zenotech Ltd

Chapter 13

Atmospheric Physics

13.1 Richardson Number

The Richardson Number defines the ratio of buoyancy to shear within a boundary layer. Non turbulent flow becomes turbulent when $R_i < 0.25$. Turbulent flow becomes non turbulent when $R_i > 1$. Note there is hysteresis in the transition to and from a turbulent state.

$$R_i = \frac{1}{\rho} \frac{g_i \frac{\partial \rho}{\partial x_i}}{\left(\frac{g_i}{|g|} \frac{\partial u_i}{\partial x_i} \right)^2} \tag{13.1}$$

13.2 Potential temperature

The potential temperature θ is defined as the temperature that an air parcel at pressure p and temperature T would acquire if adiabatically brought to the reference pressure p_0

$$\theta = T \left(\frac{p_0}{p} \right)^{\frac{R}{C_p}} \tag{13.2}$$

Equivalently $\theta = T/\pi$ with π the Exner function defined in the Stratified Atmospheric Flow chapter. Potential temperature is the energy variable used by the stratified solver.

13.3 Obukhov Length

$$L = \frac{u_\tau^2 T_0}{\kappa g \theta_\tau} \tag{13.3}$$

where the scaling temperature θ_τ is defined as

$$\theta_\tau = \frac{-\dot{q}_0}{\rho C_p u_\tau} \tag{13.4}$$

13.4 Coriolis Force

The Coriolis acceleration vector can de defined as follows

In the Northern hemisphere - acceleration to the right of the velocity vector

$$f_c = -\left(V \times \frac{g}{|g|}\right) 2\Omega_E \sin\phi \tag{13.5}$$

In the Southern hemisphere - acceleration to the left of the velocity vector

$$f_c = \left(V \times \frac{g}{|g|}\right) 2\Omega_E \sin\phi \tag{13.6}$$

where $\Omega_E = 7.292115 \times 10^{-5}\, rad/s$ is the earth's rotation rate and ϕ is the latitude (0° at equator and 90° at the poles) and g is the gravity vector.

Note that this is the *sidereal* rate, which is the one that belongs in the Coriolis parameter: the planet turns once per sidereal day (86164 s), not once per solar day. The value $2\pi/86400 = 7.2722 \times 10^{-5}\, rad/s$ quoted in earlier revisions of this document is the solar-day rate and is not what the solver uses. Both the rotation rate and the latitude may be overridden per material with `planet rotation` and `latitude`.

Only the component of the Coriolis force that acts perpendicular to the direction of the wind is considered, when viewed from a rotating reference frame. The Coriolis force in the vertical direction is neglected in atmospheric flows as this component is small compared to the gravitational acceleration.

f-plane and Geostrophic Forcing

For atmospheric boundary layer calculations the Coriolis term is applied on an f-plane, with a single Coriolis parameter evaluated at the case latitude,

$$f = 2\Omega_E \sin\phi \tag{13.7}$$

and combined with a geostrophic wind V_g in a single expression,

$$a = -f\,\hat{k} \times (V - V_g)\,, \qquad \hat{k} = -\frac{g}{|g|} \tag{13.8}$$

With $V_g = 0$ this is plain Coriolis; with $V = V_g$ it is identically zero, which is the geostrophic-balance property and the reason the two terms are written together. Applying the large-scale forcing this way rather than as a fixed body force means the balanced state is a solution rather than something the solver has to be driven towards.

The term is evaluated on the right-hand side, lagged on the pseudo-iteration. Because the scheme is dual-time, the fixed point of the pseudo-iteration is $V_{lagged} = V$, so this is equivalent to the implicit skew-symmetric operator; the two forms have been measured to agree to six decimal places on the inertial-oscillation case.

Note that a horizontal geostrophic forcing cannot be balanced in a domain with symmetry planes on the lateral boundaries, since those pin the horizontal velocity; a doubly periodic box is required.

©Zenotech Ltd

13.5 Turbulence Profile

$$k = \frac{u_\tau^2}{\sqrt{\beta^*}} \left[1 - \frac{z}{z_h}\right]^2 \tag{13.9}$$

$$\epsilon = \frac{1}{\kappa} \frac{u_\tau^3}{z} \tag{13.10}$$

Chapter 14

Radial Basis Functions (RBF)

Radial Basis Functions (RBF) are useful in rapidly and consistently deforming a mesh according to a prescribed motion of some or all of the boundary nodes. ZCFD allows users to specify a deformation function, which may be selectively applied to different zones on the boundary. The motion of these nodes is then automatically blended throughout the volume mesh in order to maintain mesh quality. The motion is formally based on a subset of the (potentially very large number) of boundary nodes specified, for numerical (conditioning) reasons. ZCFD will automatically select the subset according to user-specified tolerances and the number of nodes initially selected.

14.1 RBF interpolation method

The interpolation space consists of all functions $s(x)$ of the form

$$s(x) = \sum_{i \in \gamma_b} (\gamma_i) \Phi(\|x - x_i\|) + h(x) \tag{14.1}$$

Where Φ is a radial basis function, with compact support over the domain $[-\alpha, \alpha]$, where α is a user-specified parameter. $h(x)$ is a first degree polynomial and γ_b is a subset of the N vertices of the CFD grid covering the deformable domain. Given the known discrete displacement of boundary vertices $V_b = \{x_i\}_{i \in \gamma_b}$

$$u_i = g_i \qquad i \in \gamma_b \tag{14.2}$$

Then, the respective coefficients γ_i and the polynomial $h(x)$ are determined by imposing the interpolation conditions

$$s(x_i) = g_i \qquad i \in \gamma_b \tag{14.3}$$

$$\sum_{i \in \gamma_b} (\gamma_i) q(x_i) = 0 \tag{14.4}$$

For all polynomials q with degree less or equal than that of polynomial h (i.e. first degree). Hence leading to the following linear system of equations

$$\begin{pmatrix} M & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \gamma \\ \beta \end{pmatrix} = \begin{pmatrix} g \\ 0 \end{pmatrix} \tag{14.5}$$

With the vector $\gamma = (\gamma_1 \dots \gamma_N)$, M is the interpolation matrix with elements $M_{ij} = \Phi(\|x_i - x_j\|)$, $i,j \in \gamma_b$, matrix P has elements P_{i4} with row i given by $(1\ x_i\ y_i\ z_1)$ $i \in \gamma_b$. Vector

$g = \{g_i\}_{i \in \gamma_b}$ contains the known boundary displacements and β, the vector of coefficients for the polynomial h. Subsequently defined as

$$h(x) = \beta_1 + \beta_2 x + \beta_3 y + \beta_4 z \tag{14.6}$$

For more information see [2]

©Zenotech Ltd

Chapter 15

Wall Distance

The discrete wall distance d_i is calculated automatically at the start of each simulation for every cell centre (x_i, y_i, z_i) in the mesh. The distance function d is mesh-based and discrete, meaning that

$$d_i = \min_{j=0..(N_s-1)} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{15.1}$$

where $(x_j, y_j, z_j), j = 0, .., (N_s - 1)$ are the face centres for the wall surface elements bounding the domain.

Chapter 16

Overset Meshes

An overset case is a set of meshes $\mathcal{M}_0, \ldots, \mathcal{M}_{M-1}$ that overlap in space. Each mesh keeps its own discretisation, its own partitioning and its own boundary conditions; the meshes are never merged. The coupling is imposed weakly, by interpolating the solution of one mesh into a layer of cells of another and treating those values as a boundary condition.

Each mesh declares the mesh or meshes that supply it with donor data. This defines a directed acyclic graph over the meshes, which is topologically sorted so that a mesh always follows those it depends on. Every edge of the graph becomes an ordered pair $(\mathcal{M}_b, \mathcal{M}_o)$, written below as background and overset, carrying two mappings: one from $\mathcal{M}_o$ into $\mathcal{M}_b$ and one from $\mathcal{M}_b$ into $\mathcal{M}_o$.

16.1 Hole Cutting

Let $\chi_i \in \{0, 1\}$ be the active flag of background cell i, and let $\mathcal{N}(i)$ be its set of vertices. A cell is blanked when every one of its vertices lies inside the overset mesh domain Ω_o,

$$\chi_i = \begin{cases} 0 & \text{if } x \in \Omega_o \quad \forall x \in \mathcal{N}(i) \\ 1 & \text{otherwise} \end{cases} \tag{16.1}$$

so a cell straddling the overset mesh boundary, with at least one vertex outside it, remains active. Blanked cells take no part in the solution.

This minimal blanking is recorded before it is modified further, and is what the overset output variable reports. An overlap band, in which both meshes carry an independently computed solution, is then opened by N promotion sweeps over the interior faces. Writing $\mathcal{A}(i)$ for the set of cells sharing a face with i, each sweep applies

$$\chi_i^{(l+1)} = \begin{cases} 1 & \text{if } \chi_i^{(l)} = 0 \text{ and } \exists\, j \in \mathcal{A}(i) \text{ with } \chi_j^{(l)} = 1 \\ \chi_i^{(l)} & \text{otherwise} \end{cases} \tag{16.2}$$

for $l = 0, \ldots, N-1$, so the hole retreats by N cell layers. N is set by the `additional active layers` option on the overset boundary condition.

When a mesh is the background of several overset meshes its flags are reset once, every hole is then cut, and only then is any mapping computed. Cutting and mapping pair by pair would allow a later pair's reset to erase an earlier pair's hole.

16.2 Fringe Cells

The cells that receive interpolated data, the fringe set $\mathcal{F}$, are drawn from the blanked cells adjacent to the surviving active region,

$$\mathcal{F} = \{\, i : \chi_i = 0 \text{ and } \operatorname{dist}(i, \{j : \chi_j = 1\}) \le L \,\} \tag{16.3}$$

where distance is counted in cell layers across faces and $L = 2$ by default.

When the overset mesh moves, a cell blanked at this time step may be uncovered at the next, so $\mathcal{F}$ is taken to be every blanked cell. That is safe but requires interpolating the entire hole volume; setting `moving mesh halo layers` to $L > 0$ restores the layered definition, which is sufficient when the hole is re-cut every real time step.

16.3 Donor Search

Rigid mesh motion is described by the transformation

$$T(x) = \mathcal{R}(\theta)\,(x - x_0) + x_0 + d \tag{16.4}$$

with $\mathcal{R}$ the rotation about an axis through x_0 through angle θ, evaluated by the Rodrigues formula, and d a translation.

The mesh coordinates and the search structures are held fixed in each mesh's own reference frame and are never moved. Instead, a query point x belonging to the receiving mesh is mapped into the donor's frame before the search,

$$\tilde{x} = T^{-1}(x) \tag{16.5}$$

and the donor cell is the cell of the stationary donor mesh containing $\tilde{x}$. The bounding volume hierarchy used for that containment query is therefore built once per mesh and reused until the relative position of the meshes changes, rather than being rebuilt every time step.

A donor cell is only accepted if it is itself active. A fringe point falling in a blanked region of the donor mesh has no valid donor, and the corresponding face falls back to the unmapped face treatment instead of interpolated data. This is the usual reason to increase N.

16.4 Interpolation

Given a donor cell c containing $\tilde{x}$, the interpolated value is

$$W(\tilde{x}) = \frac{\sum_{j \in \mathcal{D}} w_j W_j}{\sum_{j \in \mathcal{D}} w_j} \tag{16.6}$$

The donor set $\mathcal{D}$ and the weights depend on the requested order. At zeroth order $\mathcal{D} = \{c\}$ with $w_c = 1$, giving a piecewise constant transfer. At first order $\mathcal{D}$ is c together with its face neighbours, and with $d_j = \|x_j - \tilde{x}\|$ the distance from the query point to each donor cell centre,

$$w_j = \frac{R_m - d_j}{R_m + 4 d_j}, \qquad R_m = 1.2 \max_{j \in \mathcal{D}} d_j \tag{16.7}$$

so that the weight falls to zero at the edge of the donor stencil.

©Zenotech Ltd

16.5 Transformation of the Solution Between Frames

An overset mesh attached to a moving fluid zone is solved in its own mesh-attached frame, and only the mapped data crosses between frames. Scalar components of W, the density and the total energy, are frame invariant and are interpolated directly. The momentum components are vectors and must be rotated,

$$(\rho\mathbf{u})^{\text{inertial}} = \mathcal{R}(\theta)\,(\rho\mathbf{u})^{\text{mesh}} \tag{16.8}$$

with the inverse rotation applied in the opposite direction.

The dual time step differences the solution against the two previous time levels, so those levels must be expressed in the same frame as the current one. At the end of each real time step the accumulated angle is advanced by $\theta \leftarrow \theta + \omega\Delta t$ and the momentum of W^{n+1}, W^{n} and W^{n-1} is counter-rotated through $-\omega\Delta t$, which returns all three to the new frame before the next step differences them.

16.6 Coupling Within the Dual Time Step

The interpolated values are exchanged once per pseudo time cycle, before any mesh is advanced. Writing k for the pseudo time index of the modified steady problem of section *Dual Time Step*, the fringe cells of each mesh are assigned

$$W_i^{k} = W(\tilde{x}_i)\Big|_{\text{donor at } k}, \qquad i \in \mathcal{F} \tag{16.9}$$

and those values are then held fixed while the interior is advanced from k to $k+1$.

This has a consequence for the choice of pseudo time scheme. A multi-stage explicit update advances the interior several times within one cycle, and every stage sees the same frozen interface data, so the interface lags the interior by an amount that depends on the stage count. The implicit schemes take a single update per cycle, which is the rate at which the exchange itself runs. An overset case therefore requires `implicit euler` or `lu-sgs`, and an explicit pseudo time scheme is rejected during input validation.

16.7 Treatment of Blanked Cells

Blanking is carried through the rest of the solver by the same flag. The residual is masked,

$$R_i \leftarrow \chi_i R_i \tag{16.10}$$

so blanked cells neither drive the solution nor contribute to the reported residual norms, and the implicit update is masked identically so that a blanked row is left untouched.

On the coarse levels of the geometric multigrid the flag is inherited from the fine mesh. Writing $\mathcal{C}(I)$ for the fine cells agglomerated into coarse cell I,

$$\chi_I = \min_{i \in \mathcal{C}(I)} \chi_i \tag{16.11}$$

so a coarse cell is blanked if any of its children is blanked. A coarse cell whose children are a mixture of active and blanked is reported in the log; a large count indicates that the agglomeration and the hole boundary are poorly aligned. Multigrid is in any case unavailable with the implicit pseudo time schemes an overset case requires.

Appendix A

$$\frac{\partial W}{\partial Q'} = \begin{pmatrix} 1 & 0 & 0 & 0 & \rho_p & 0 \\ u & \rho & 0 & 0 & u\rho_p & 0 \\ v & 0 & \rho & 0 & v\rho_p & 0 \\ w & 0 & 0 & \rho & w\rho_p & 0 \\ H + h_\rho & \rho u & \rho v & \rho w & \rho_p H - 1 + \rho h_p & 0 \\ \phi & 0 & 0 & 0 & \phi\rho_p & 1 \end{pmatrix} \tag{16.12}$$

where $Q^{'} = [\rho, u, v, w, p, \phi]^T$

$$\frac{\partial Q'}{\partial W} = \begin{pmatrix} \frac{(V^2 - H - h_r)\rho_p}{d} + 1 & -\frac{\rho_p u}{d} & -\frac{\rho_p v}{d} & -\frac{\rho_p w}{d} & \frac{\rho_p}{d} & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 & 0 \\ -\frac{V^2 - H - h_r}{d} & \frac{u}{d} & \frac{v}{d} & \frac{w}{d} & -\frac{1}{d} & 0 \\ -\phi & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{16.13}$$

where $d = (H + h_r)\rho_p - H\rho_p - h_p\rho + 1$

$$\frac{\partial F}{\partial Q'} = \begin{pmatrix} q & \rho n_x & \rho n_y & \rho n_z & \rho_p q & 0 \\ uq + p_\rho n_x & \rho(q + un_x) & \rho u n_y & \rho u n_z & \rho_p uq + n_x & 0 \\ vq + p_\rho n_y & \rho v n_x & \rho(q + vn_y) & \rho v n_z & \rho_p vq + n_y & 0 \\ wq + p_\rho n_z & \rho w n_x & \rho w n_z & \rho(q + wn_z) & \rho_p wq + n_z & 0 \\ (H + \rho h_p)q & \rho H n_x + q\rho u & \rho H n_y + q\rho v & \rho H n_z + q\rho w & q(\rho_p H + \rho h_p) & 0 \\ \phi q & 0 & 0 & 0 & 0 & \rho q \end{pmatrix} \tag{16.14}$$

Bibliography

[1] P. Batten, M.A. Leschziner, and U.C. Goldberg. Average-state jacobians and implicit methods for compressible viscous and turbulent flows. *Journal of Computational Physics*, 137(1):38 – 78, 1997.

[2] O. Estruch O. Lehmkuhl R. Borrell and C.D. Pérez Segarra. A parallel radial basis function interpolation method for unstructured dynamic meshes. *Computers and Fluids*, 2013.

[3] P A Durbin. Limiters and wall treatments in applied turbulence modeling. *Fluid Dynamics Research*, 41(1):012203, 2009.

[4] Bernhard Eisfeld, Tobias Knopp, and JavierBartolome Calvo. Application of a new roughness extension for k - ω turbulence models. In Andreas Dillmann, Gerd Heller, Michael Klaas, Hans-Peter Kreplin, Wolfgang Nitsche, and Wolfgang Schröder, editors, *New Results in Numerical and Experimental Fluid Mechanics VII*, volume 112 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, pages 43–50. Springer Berlin Heidelberg, 2010.

[5] S. Creech et. al. *Simulations of an Offshore Wind Farm using Large Eddy Simulation and a Torque-Controlled Actuator Disc Model*. University of Edinburgh, 2014.

[6] S. S. A. Ivanell. *Numerical Computations of Wind Turbine Wakes*. Gotland University, 2009.

[7] Antony Jameson. Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings. In *Proc. 10th Comp. Fluid Dyn. Conf., Honolulu, HI, USA, June 24-26, 1991, AIAA-Paper 91-1596*, 1991.

[8] Chandra Laksham Vaidyaratna. Jayatilleke. The influence of Prandtl number and surface roughness on the resistance of the laminar sub-layer to momentum and heat transfer. *Prog. Heat Mass Transfer*, 1:193-321, Jan 1969.

[9] Georgi Kalitzin, Gorazd Medic, Gianluca Iaccarino, and Paul Durbin. Near-wall behavior of rans turbulence models and implications for wall functions. *J. Comput. Phys.*, 204(1):265–291, mar 2005.

[10] Hong Luo, Joseph D. Baum, and Rainald Löhner. A hybrid cartesian grid and gridless method for compressible flows. *J. Comput. Phys.*, 214(2):618–632, may 2006.

[11] F. Menter, M. Kuntz, and R. Langtry. Ten years of industrial experience with the SST model. In K. Hanjalić, Y. Nagano, and M. Tummers, editors, *Turbulence, Heat and Mass Transfer 4*, 2003.

[12] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, 32:1598–1605, aug 1994.

[13] Florian R. Menter and Ames Research Center. *Improved two-equation k - [omega] turbulence models for aerodynamic flows [microform] / Florian R. Menter*. National Aeronautics and Space Administration, Ames Research Center ; National Technical Information Service, distributor Moffett Field, Calif. : Springfield, Va, 1992.

[14] Geraldine Olivier and Frederic Alauzet. *A New Changing-Topology ALE Scheme for Moving Mesh Unsteady Simulations.*

[15] D. A. Peters and R. Modarres. *A Compact, Closed-form Solution for the Optimum, Ideal Wind Turbine*. Washington University in St. Louis, 2012.

[16] J.M. Cela R. Aubry M. Vazquez G. Houzeaux and S. Marras. An unstructured cfd approach to numerical weather prediction. In *48th AIAA Aerospace Sciences Meeting 4-7 Jan 2010 Orlando World Center Marriott Orlando Florida*, 2010.

[17] Philippe R. Spalart and Christopher L. Rumsey. Effective Inflow Conditions for Turbulence Models in Aerodynamic Calculations. *AIAA Journal*, 45(10):2544–2553, oct 2007.

[18] J. R. Viegas and M. W. Rubesin. On the use of wall functions as boundary conditions for two-dimensional separated compressible flows. *AIAA Paper*, Jan 1985.

[19] J. M. Weiss and W. A. Smith. Preconditioning applied to variable and constant density flows. *AIAA Journal*, 33:2050–2057, nov 1995.

 ©Zenotech Ltd