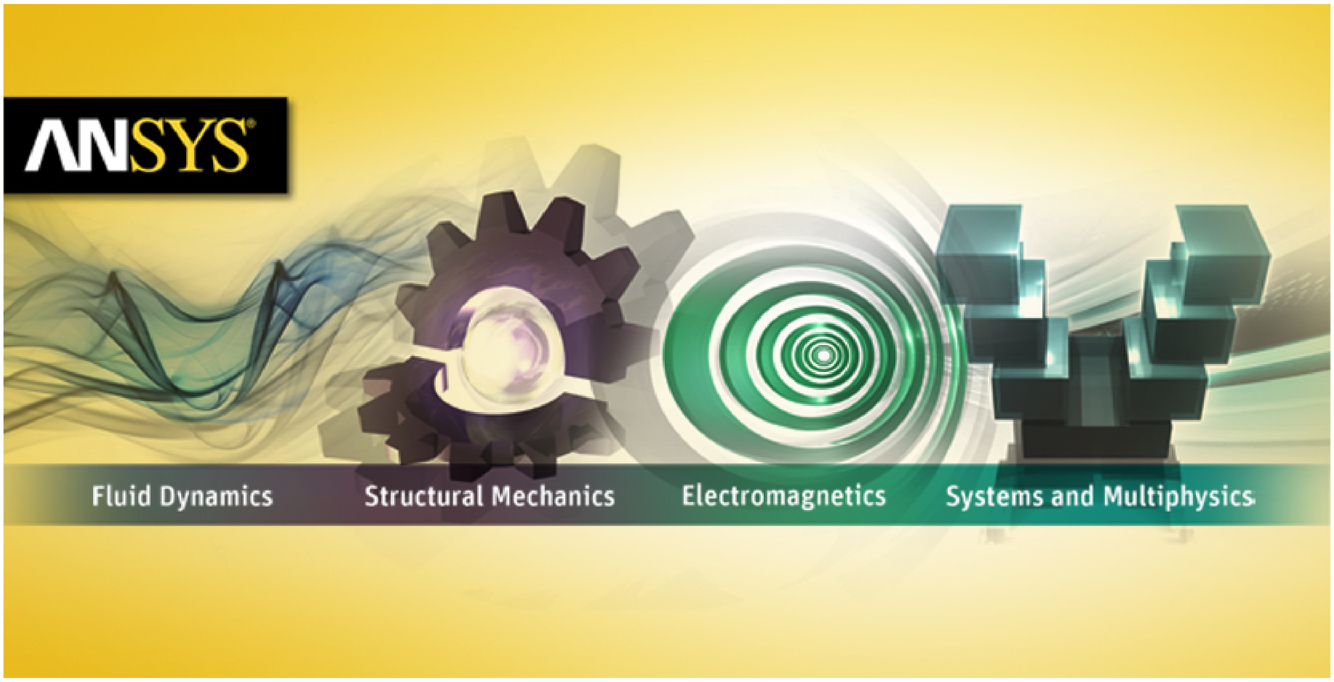


ANSYS CFX-Solver Theory Guide

ANSYS, Inc.
Southpointe
275 Technology Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
http://www.ansys.com
(T) 724-746-3304
(F) 724-514-9494

Release 14.0
November 2011

Published in the U.S.A.

Table of Contents

Chapter 1: Basic Solver Capability Theory

This chapter describes the mathematical equations used to model fluid flow, heat, and mass transfer in ANSYS CFX for single-phase, single and multi-component flow without combustion or radiation. It is designed to be a reference for those users who desire a more detailed understanding of the mathematics underpinning the CFX-Solver, and is therefore not essential reading. It is not an exhaustive text on CFD mathematics; a reference section is provided should you want to follow up this chapter in more detail.

This chapter describes:

For information on dealing with multiphase flow:

- *Multiphase Flow Theory* (p. 165)
- *Particle Transport Theory* (p. 239)

For information on combustion and radiation theory:

- *Combustion Theory* (p. 305)
- *Radiation Theory* (p. 351)

Descriptions of the theory for other physical models are provided in:

- *Turbulence and Wall Function Theory* (p. 89)
- *GGI and MFR Theory* (p. 155)
- *Transient Blade Row Modeling Theory* (p. 159)
- *Electromagnetic Hydrodynamic Theory* (p. 363)
- *Rigid Body Theory* (p. 371)
- *Discretization and Solution Theory* (p. 375)
- *Wall Condensation Theory* (p. 83)

Recommended books for further reading on CFD and related subjects:

- Further Background Reading in the *CFX Introduction*

1.1. Documentation Conventions

The topics in this section are:

1.1.1. Dimensions

Throughout this manual, dimensions are given in terms of the fundamental magnitudes of length (L), mass (M), time (T), temperature (Θ) and chemical amount (A).

1.1.2. List of Symbols

This section lists symbols used in this chapter, together with their meanings, dimensions and where applicable, values. Dimensionless quantities are denoted by 1. The values of physical constants (or their default values) are also given.

More information on the notation used in the multiphase and multicomponent chapters is available:

- *Multiphase Notation* (p. 165)
- *Multicomponent Notation* (p. 44).

Symbol	Description	Dimensions	Value
C_E	linear energy source coefficient	$\mathrm{M\,L^{-1}\,T^{-3}}\,\Theta^{-1}$	
C_{R1}	linear resistance coefficient	$\mathrm{M\,L^{-3}\,T^{-1}}$	
C_{R2}	quadratic resistance coefficient	$\mathrm{M\,L^{-4}}$	
$C_{\varepsilon 1}$	k-ε turbulence model constant		1.44
$C_{\varepsilon 1 \mathrm{RNG}}$	RNG k-ε turbulence model coefficient		$1.42 - f_h$
$C_{\varepsilon 2}$	k-ε turbulence model constant		1.92
$C_{\varepsilon 2 \mathrm{RNG}}$	RNG k-ε turbulence model constant		1.68
C_μ	k-ε turbulence model constant		0.09

Symbol	Description	Dimensions	Value
$C_{\mu\,\mathrm{RS}}$	Reynolds Stress model constant		
$C_{\mu\,\mathrm{RNG}}$	RNG k-ε turbulence model constant		0.085
c	fluid speed of sound	$\mathrm{L\,T^{-1}}$	
$c_{\mathrm{A}}, c_{\mathrm{B}}$	concentration of components A and B (that is, mass per unit volume of components A and B (single-phase flow))	$\mathrm{M\,L^{-3}}$	
c_{S}	Reynolds Stress model constant		0.22
c_p	specific heat capacity at constant pressure	$\mathrm{L^2\,T^{-2}}\,\Theta^{-1}$	
c_v	specific heat capacity at constant volume	$\mathrm{L^2\,T^{-2}}\,\Theta^{-1}$	
$c_{\varepsilon\,1}$	Reynolds Stress model constant		1.45
$c_{\varepsilon\,2}$	Reynolds Stress model constant		1.9
$D_{A\,B}$	binary diffusivity of component A in component B	$\mathrm{L^2\,T^{-1}}$	
D_{Φ}	kinematic diffusivity of an Additional Variable, Γ_{Φ}/ρ	$\mathrm{L^2\,T^{-1}}$	
d	distance or length	L	
E	constant used for near-wall modeling		9.793
f_{μ}	Zero Equation turbulence model constant		0.01
f_h	RNG-k-ε turbulence model coefficient		
$\boldsymbol{g}$	gravity vector	$\mathrm{L\,T^{-2}}$	
h, h_{stat}	specific static (thermodynamic) enthalpy	$\mathrm{L^2\,T^{-2}}$	For details, see *Static Enthalpy* (p. 8).
Ha	Hartmann Number: the ratio of the Lorentz forces to the viscous forces. $Ha = BL\left(\frac{\sigma}{\mu}\right)^{\frac{1}{2}}$		
h_{c}	heat transfer coefficient	$\mathrm{M\,T^{-3}}\,\Theta^{-1}$	

Symbol	Description	Dimensions	Value
h_{tot}	specific total enthalpy	$\mathrm{L}^2\ \mathrm{T}^{-2}$	For details, see *Total Enthalpy* (p. 10).
k	turbulence kinetic energy per unit mass	$\mathrm{L}^2\ \mathrm{T}^{-2}$	
M	local Mach number, U/c		
$\dot{m}$	mass flow rate	$\mathrm{M}\ \mathrm{T}^{-1}$	
P_k	shear production of turbulence	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-3}$	
p, p_{stat}	static (thermodynamic) pressure	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-2}$	For details, see *Static Pressure* (p. 8).
p_{ref}	reference pressure	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-2}$	For details, see *Reference Pressure* (p. 7).
p_{tot}	total pressure	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-2}$	For details, see *Total Pressure* (p. 16).
p'	modified pressure	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-2}$	For details, see *Modified Pressure* (p. 8).
R_0	universal gas constant	$\mathrm{L}^2\ \mathrm{T}^{-2}\ \Theta^{-1}$	8314.5
Re	Reynolds number, $r\ U\ d/m$		
Re_m	Magnetic Reynolds Number: the ratio between the magnetic advection and magnetic diffusion. $Re_m = \frac{UL}{\mu}$		
$\boldsymbol{r}$	location vector	L	
r_α	volume fraction of phase α		
S_E	energy source	$\mathrm{M}\ \mathrm{L}^{-1}\ \mathrm{T}^{-3}$	
$\boldsymbol{S}_M$	momentum source	$\mathrm{M}\ \mathrm{L}^{-2}\ \mathrm{T}^{-2}$	
S_{MS}	mass source	$\mathrm{M}\ \mathrm{L}^{-3}\ \mathrm{T}^{-1}$	
Sc_{t}	turbulent Schmidt number, μ_t / Γ_t		

Symbol	Description	Dimensions	Value
St	Stanton number		
$s_{\alpha\beta}$	mass flow rate from phase α to phase β.	$\mathrm{M\ T^{-1}}$	
T, T_{stat}	static (thermodynamic) temperature	Θ	For details, see Static Temperature (p. 10).
T_{dom}	domain temperature	Θ	For details, see Domain Temperature (p. 10).
T_{ref}	buoyancy reference temperature used in the Boussinesq approximation	Θ	
T_{sat}	saturation temperature	Θ	
T_{tot}	total temperature	Θ	For details, see Total Temperature (p. 11).
$\boldsymbol{U}$	vector of velocity $\boldsymbol{U}_{x,y,z}$	$\mathrm{L\ T^{-1}}$	
U	velocity magnitude	$\mathrm{L\ T^{-1}}$	
$\boldsymbol{u}$	fluctuating velocity component in turbulent flow	$\mathrm{L\ T^{-1}}$	
W_f	fluid viscous and body force work term	$\mathrm{M\ L^{-1}\ T^{-3}}$	
w	molecular weight (Ideal Gas fluid model)		
Y_{A}	mass fraction of component A in the fluid		
α	used as a subscript to indicate that the quantity applies to phase α		
β	used as a subscript to indicate that the quantity applies to phase β		
β	coefficient of thermal expansion (for the Boussinesq approximation)	Θ^{-1}	
β_{RNG}	RNG k-ε turbulence model constant		0.012
Γ	diffusivity	$\mathrm{M\ L^{-1}\ T^{-1}}$	

Symbol	Description	Dimensions	Value
Γ_{A}	molecular diffusion coefficient of component A	$\mathrm{M\,L^{-1}\,T^{-1}}$	
Γ_{Φ}	dynamic diffusivity of an Additional Variable	$\mathrm{M\,L^{-1}\,T^{-1}}$	
Γ_{t}	turbulent diffusivity	$\mathrm{M\,L^{-1}\,T^{-1}}$	
δ	identity matrix or Kronecker Delta function		
ε	turbulence dissipation rate	$\mathrm{L^{2}\,T^{-3}}$	
ζ	bulk viscosity	$\mathrm{M\,L^{-1}\,T^{-1}}$	
κ	Von Karman constant		0.41
λ	thermal conductivity	$\mathrm{M\,L\,T^{-3}}\,\Theta^{-1}$	
μ	molecular (dynamic) viscosity	$\mathrm{M\,L^{-1}\,T^{-1}}$	
μ_{t}	turbulent viscosity	$\mathrm{M\,L^{-1}\,T^{-1}}$	
μ_{eff}	effective viscosity, $\mu + \mu_t$	$\mathrm{M\,L^{-1}\,T^{-1}}$	
ρ	density	$\mathrm{M\,L^{-3}}$	
Pr	laminar Prandtl number, $c_{\mathrm{p}}\,\mu/\lambda$		
Pr_{t}	turbulent Prandtl number, $c_{\mathrm{p}}\,\mu_t/\lambda_t$		
σ_k	turbulence model constant for the k equation		1.0
σ_{ε}	k-ε turbulence model constant		1.3
σ_{ω}	k-ω turbulence model constant		2
$\sigma_{\varepsilon\,\mathrm{RS}}$	Reynolds Stress model constant		
$\sigma_{k\,\mathrm{RNG}}$	RNG k-ε turbulence model constant		0.7179
$\sigma_{\varepsilon\,\mathrm{RNG}}$	RNG k-ε turbulence model constant		0.7179
τ	shear stress or sub-grid scale stress molecular stress tensor	$\mathrm{M\,L^{-1}\,T^{-2}}$	
υ	specific volume	$\mathrm{M^{-1}\,L^{3}}$	

Symbol	Description	Dimensions	Value
Φ	Additional Variable (non-reacting scalar)	$\mathrm{M\,L^{-3}}$	
φ	general scalar variable		
ω	angular velocity	$\mathrm{T^{-1}}$	

1.1.2.1. Subscripts

Quantities that appear with subscripts A, B, C refer to that quantity for component A, B, C in a multicomponent fluid.

Quantities that appear with subscripts α, β, γ refer to that quantity for phase α, β, γ in a multiphase flow.

Such quantities are used only in the chapters describing multicomponent and multiphase flows.

- *Multicomponent Flow* (p. 44)
- *Multiphase Flow Theory* (p. 165)

1.1.3. Variable Definitions

1.1.3.1. Isothermal Compressibility

The isothermal compressibility defines the rate of change of the system volume with pressure. For details, see Variables Relevant for Compressible Flow in the *CFX Reference Guide*.

$$\left(\frac{1}{\rho}\right)\left(\frac{d\rho}{dp}\right)_T \tag{1–1}$$

1.1.3.2. Isentropic Compressibility

Isentropic compressibility is the extent to which a material reduces its volume when it is subjected to compressive stresses at a constant value of entropy. For details, see Variables Relevant for Compressible Flow in the *CFX Reference Guide*.

$$\left(\frac{1}{\rho}\right)\left(\frac{d\rho}{dp}\right)_S \tag{1–2}$$

1.1.3.3. Reference Pressure

The `Reference Pressure` *Equation 1–3* (p. 8) is the absolute pressure datum from which all other pressure values are taken. All relative pressure specifications in ANSYS CFX are relative to the `Reference Pressure`. For details, see Setting a Reference Pressure in the *CFX-Solver Modeling Guide*.

$$P_{\text{ref}} \tag{1–3}$$

1.1.3.4. Static Pressure

CFX solves for the relative `Static Pressure` (thermodynamic pressure) *Equation 1–4* (p. 8) in the flow field, and is related to `Absolute Pressure` *Equation 1–5* (p. 8).

$$p_{\text{stat}} \tag{1–4}$$

$$p_{\text{abs}} = p_{\text{stat}} + p_{\text{ref}} \tag{1–5}$$

1.1.3.5. Modified Pressure

A modified pressure is used in the following circumstances:

- When certain turbulence models are used, (for example k-ε, k-ω, and Reynolds Stress), the modified pressure includes an additional term due to the turbulent normal stress. For details, see *Equation 2–14* (p. 93).
- When buoyancy is activated, the modified pressure excludes the hydrostatic pressure field. For details, see Buoyancy and Pressure.

1.1.3.6. Static Enthalpy

Specific static enthalpy *Equation 1–6* (p. 8) is a measure of the energy contained in a fluid per unit mass. Static enthalpy is defined in terms of the internal energy of a fluid and the fluid state:

$$h_{\text{stat}} = u_{\text{stat}} + \frac{p_{\text{stat}}}{\rho_{\text{stat}}} \tag{1–6}$$

When you use the thermal energy model, the CFX-Solver directly computes the static enthalpy. General changes in enthalpy are also used by the solver to calculate thermodynamic properties such as temperature. To compute these quantities, you need to know how enthalpy varies with changes in both temperature and pressure. These changes are given by the general differential relationship *Equation 1–7* (p. 8):

$$dh = \left(\frac{\partial h}{\partial T}\right)_p dT + \left(\frac{\partial h}{\partial p}\right)_T dp \tag{1–7}$$

which can be rewritten as *Equation 1–8* (p. 9)

$$dh = c_p\, dT + \frac{1}{\rho}\left[1 + \frac{T}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p\right] dp \tag{1–8}$$

where c_p is specific heat at constant pressure and ρ is density. For most materials the first term always has an effect on enthalpy, and, in some cases, the second term drops out or is not included. For example, the second term is zero for materials that use the Ideal Gas equation of state or materials in a solid thermodynamic state. In addition, the second term is also dropped for liquids or gases with constant specific heat when you run the thermal energy equation model.

1.1.3.6.1. Material with Variable Density and Specific Heat

In order to support general properties, which are a function of both temperature and pressure, a table for $h\,(T, p)$ is generated by integrating *Equation 1–8* (p. 9) using the functions supplied for ρ and c_p. The enthalpy table is constructed between the upper and lower bounds of temperature and pressure (using flow solver internal defaults or those supplied by the user). For any general change in conditions from $\left(p_1, T_1\right)$ to $\left(p_2, T_2\right)$, the change in enthalpy, dh, is calculated in two steps: first at constant pressure, and then at constant temperature using *Equation 1–9* (p. 9).

$$h_2 - h_1 = \int_{T_1}^{T_2} c_p\, dT + \int_{p_1}^{p_2} \frac{1}{\rho}\left[1 + \frac{T}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p\right] dp \tag{1–9}$$

To successfully integrate *Equation 1–9* (p. 9), the CFX-Solver must be provided thermodynamically consistent values of the equation of state, ρ, and specific heat capacity, c_p. "Thermodynamically consistent" means that the coefficients of the differential terms of *Equation 1–8* (p. 9) must satisfy the exact differential property that:

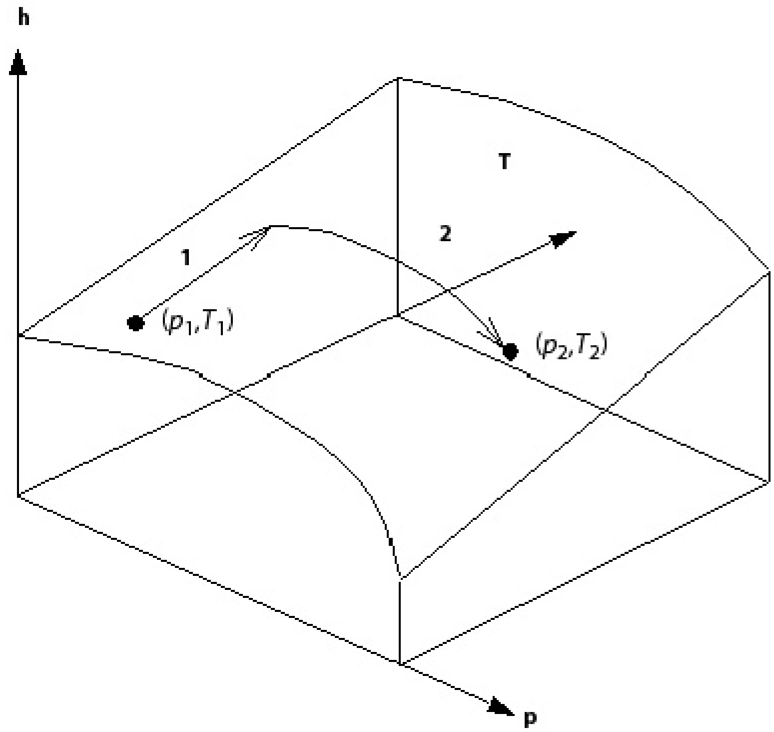
$$\frac{\partial}{\partial p}\Big|_T \left(\frac{\partial h}{\partial T}\right)_p = \frac{\partial}{\partial T}\Big|_p \left(\frac{\partial h}{\partial p}\right)_T \tag{1–10}$$

or, in terms of the expressions given in *Equation 1–9* (p. 9):

$$\left(\frac{\partial c_p}{\partial p}\right)_T = \frac{1}{\rho^2}\left[T\left(\frac{\partial^2 \rho}{\partial T^2}\right)_p - \left(1 + \frac{T}{\rho}\right)\left(\frac{\partial \rho}{\partial T}\right)_p + 1\right] \tag{1–11}$$

To satisfy c_p in *Equation 1–11* (p. 9), variables of the form '<Material Name>.Property Residual' are computed and should resolve to zero.

The equation of state derivative within the second integral of *Equation 1–9* (p. 9) is numerically evaluated from the using a two point central difference formula. In addition, the CFX-Solver uses an adaptive number of interpolation points to construct the property table, and bases the number of points on an absolute error tolerance estimated using the enthalpy and entropy derivatives.



1.1.3.7. Total Enthalpy

Total enthalpy is expressed in terms of a static enthalpy and the flow kinetic energy:

$$h_{\text{tot}} = h_{\text{stat}} + \frac{1}{2}(\boldsymbol{U} \cdot \boldsymbol{U}) \tag{1–12}$$

where $\boldsymbol{U}$ is the flow velocity. When you use the total energy model the CFX-Solver directly computes total enthalpy, and static enthalpy is derived from this expression. In case of turbulence models for which the kinetic turbulent energy k is available (such as k-epsilon, k-omega, SST, and so on), k is included in the total enthalpy. In rotating frames of reference, the total enthalpy includes the relative frame kinetic energy. For details, see *Rotating Frame Quantities* (p. 18).

1.1.3.8. Domain Temperature

The domain temperature, T_{dom}, is the absolute temperature at which an isothermal simulation is performed. For details, see Isothermal in the *CFX-Solver Modeling Guide*.

1.1.3.9. Static Temperature

The static temperature, T_{stat}, is the thermodynamic temperature, and depends on the internal energy of the fluid. In ANSYS CFX, depending on the heat transfer model you select, the flow solver calculates either total or static enthalpy (corresponding to the total or thermal energy equations).

The static temperature is calculated using static enthalpy and the constitutive relationship for the material under consideration. The constitutive relation simply tells us how enthalpy varies with changes in both temperature and pressure.

1.1.3.9.1. Material with Constant Density and Specific Heat

In the simplified case where a material has constant ρ and c_p, temperatures can be calculated by integrating a simplified form of the general differential relationship for enthalpy:

$$h_{\text{stat}} - h_{\text{ref}} = c_p \left(T_{\text{stat}} - T_{\text{ref}} \right) \tag{1–13}$$

which is derived from the full differential form for changes in static enthalpy. The default reference state in the CFX-Solver is $T_{\text{ref}} = 0 \ [K]$ and $h_{\text{ref}} = 0 \ [J/(k\,g)]$.

1.1.3.9.2. Ideal Gas or Solid with c_p=f(T)

The enthalpy change for an ideal gas or CHT solid with specific heat as a function of temperature is defined by:

$$h_{\text{stat}} - h_{\text{ref}} = \int_{T_{\text{ref}}}^{T_{\text{stat}}} c_p(T)\ dT \tag{1–14}$$

When the solver calculates static enthalpy, either directly or from total enthalpy, you can back static temperature out of this relationship. When c_p varies with temperature, the CFX-Solver builds an enthalpy table and static temperature is backed out by inverting the table.

1.1.3.9.3. Material with Variable Density and Specific Heat

To properly handle materials with an equation of state and specific heat that vary as functions of temperature and pressure, the CFX-Solver needs to know enthalpy as a function of temperature and pressure, $h(T, p)$.

$h(T, p)$ can be provided as a table using, for example, an `.rgp` file. If a table is not pre-supplied, and the equation of state and specific heat are given by CEL expressions or CEL user functions, the CFX-Solver will calculate $h(T, p)$ by integrating the full differential definition of enthalpy change.

Given the knowledge of $h(T, p)$ and that the CFX-Solver calculates both static enthalpy and static pressure from the flow solution, you can calculate static temperature by inverting the enthalpy table:

$$h_{\text{stat}} - h_{\text{ref}} = h\left(T_{\text{stat}}, p_{\text{stat}}\right) - h\left(T_{\text{ref}}, p_{\text{ref}}\right) \tag{1–15}$$

In this case, you know h_{stat}, p_{stat} from solving the flow and you calculate T_{stat} by table inversion.

1.1.3.10. Total Temperature

The total temperature is derived from the concept of total enthalpy and is computed exactly the same way as static temperature, except that total enthalpy is used in the property relationships.

1.1.3.10.1. Material with Constant Density and Specific Heat

If ρ and c_p are constant, then the total temperature and static temperature are equal because incompressible fluids undergo no temperature change due to addition of kinetic energy. This can be illustrated by starting with the total enthalpy form of the constitutive relation:

$$h_{\mathrm{tot}} - h_{\mathrm{ref}} = c_p \left(T_{\mathrm{tot}} - T_{\mathrm{ref}} \right) \tag{1–16}$$

and substituting expressions for *Static Enthalpy* (p. 8) and *Total Pressure* (p. 16) for an incompressible fluid:

$$h_{\mathrm{tot}} = h_{\mathrm{stat}} + \frac{1}{2} \left(\boldsymbol{U} \cdot \boldsymbol{U} \right) \tag{1–17}$$

$$p_{\mathrm{tot}} = p_{\mathrm{stat}} + \frac{1}{2} \rho \left(\boldsymbol{U} \cdot \boldsymbol{U} \right) \tag{1–18}$$

some rearrangement gives the result that:

$$T_{\mathrm{tot}} = T_{\mathrm{stat}} \tag{1–19}$$

for this case.

1.1.3.10.2. Ideal Gas with constant c_p

For this case, enthalpy is only a function of temperature and the constitutive relation is:

$$h - h_{\mathrm{ref}} = c_p \left(T - T_{\mathrm{ref}} \right) \tag{1–20}$$

which, if one substitutes the relation between static and total enthalpy, yields:

$$T_{\mathrm{tot}} = T_{\mathrm{stat}} + \frac{\boldsymbol{U} \cdot \boldsymbol{U}}{2\, c_p} \tag{1–21}$$

If the $k - \varepsilon$ turbulence model is employed, turbulence kinetic energy, k, is added to the total enthalpy, giving the modified total temperature for constant c_p:

$$T_{\mathrm{tot}} = T_{\mathrm{stat}} + \frac{\boldsymbol{U} \cdot \boldsymbol{U}}{2\, c_p} + \frac{k}{c_p} \tag{1–22}$$

1.1.3.10.3. Ideal Gas with $c_p = f(T)$

The total temperature is evaluated with:

$$h_{tot} - h_{ref} = \int_{T_{ref}}^{T_{tot}} c_p(T)\ dT \qquad \textbf{(1–23)}$$

using table inversion to back out the total temperature.

1.1.3.10.4. Material with Variable Density and Specific Heat

In this case, total temperature is calculated in the exact same way as static temperature except that total enthalpy and total pressure are used as inputs into the enthalpy table:

$$h_{tot} - h_{ref} = h\left(T_{tot}, p_{tot}\right) - h\left(T_{ref}, p_{ref}\right) \qquad \textbf{(1–24)}$$

In this case you know h_{tot}, and you want to calculate T_{tot}, but you do not know p_{tot}. So, before calculating the total temperature, you need to compute total pressure. For details, see *Total Pressure* (p. 16).

For details, see *Rotating Frame Quantities* (p. 18).

1.1.3.11. Entropy

The concept of entropy arises from the second law of thermodynamics:

$$T\ ds = dh - \frac{dp}{\rho} \qquad \textbf{(1–25)}$$

which can be rearranged to give:

$$ds = \frac{dh}{T} - \frac{dp}{\rho\,T} \qquad \textbf{(1–26)}$$

Depending on the equation of state and the constitutive relationship for the material, you can arrive at various forms for calculating the changes in entropy as a function of temperature and pressure.

1.1.3.11.1. Material with Constant Density and Specific Heat

In this case, changes in enthalpy as a function of temperature and pressure are given by:

$$dh = c_p\,dT + \frac{dp}{\rho} \qquad \textbf{(1–27)}$$

and when this is substituted into the second law gives the following expression for changes in entropy as a function of temperature only:

$$ds = c_p \frac{dT}{T} \tag{1–28}$$

which, when integrated, gives an analytic formula for changes in entropy:

$$s - s_{\text{ref}} = c_p \log\left(\frac{T}{T_{\text{ref}}}\right) \tag{1–29}$$

1.1.3.11.2. Ideal Gas with constant c_p or $c_p = f(T)$

For ideal gases changes in entropy are given by the following equation:

$$s - s_{\text{ref}} = \int_{T_{\text{ref}}}^{T} \frac{c_p(T)}{T}\, dT - R \log\left(\frac{p}{p_{\text{ref}}}\right) \tag{1–30}$$

which for general functions for c_p the solver computes an entropy table as a function of both temperature and pressure. In the simplified case when c_p is a constant, then an analytic formula is used:

$$s - s_{\text{ref}} = c_p \log\left(\frac{T}{T_{\text{ref}}}\right) - R \log\left(\frac{p}{p_{\text{ref}}}\right) \tag{1–31}$$

1.1.3.11.3. Material with Variable Density and Specific Heat

This is the most general case handled by the CFX-Solver. The entropy function, $s(T,p)$, is calculated by integrating the full differential form for entropy as a function of temperature and pressure. Instead of repetitively performing this integration the CFX-Solver computes a table of values at a number of temperature and pressure points. The points are chosen such that the approximation error for the entropy function is minimized.

The error is estimated using derivatives of entropy with respect to temperature and pressure. Expressions for the derivatives are found by substituting the formula for general enthalpy changes into the second law to get the following expression for changes in entropy with temperature and pressure:

$$ds = \frac{c_p}{T}\, dT + \frac{1}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_p dp \tag{1–32}$$

which when compared with the following differential form for changes in entropy:

$$ds = \left(\frac{\partial s}{\partial T}\right)_p dT + \left(\frac{\partial s}{\partial p}\right)_T dp \tag{1–33}$$

gives that:

$$\left(\frac{\partial s}{\partial T}\right)_p = \frac{c_p}{T} \tag{1–34}$$

$$\left(\frac{\partial s}{\partial p}\right)_T = \frac{1}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{1–35}$$

The derivative of entropy with respect to temperature is exactly evaluated, while the derivative with respect to pressure must be computed by numerically differentiating the equation of state. Note that when properties are specified as a function of temperature and pressure using CEL expressions the differential terms in *Equation 1–32* (p. 14) must also satisfy the exact differential relationship:

$$\frac{\partial}{\partial p}\Big|_T \left(\frac{\partial s}{\partial T}\right)_p = \frac{\partial}{\partial T}\Big|_p \left(\frac{\partial s}{\partial p}\right)_T \tag{1–36}$$

or,

$$\frac{\partial}{\partial p}\Big|_T \left(\frac{c_p}{T}\right) = \frac{\partial}{\partial T}\Big|_p \left[\frac{1}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_p\right] \tag{1–37}$$

Also, see the previous section on *Static Enthalpy* (p. 8) for more details.

Unless an externally provided table is supplied, an entropy table is built by using the ρ and c_p functions supplied by the user and integrating the general differential form:

$$s_2 - s_1 = \int_{T_1}^{T_2} \frac{c_p}{T}\, dT + \int_{p_1}^{p_2} \frac{1}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_p dp \tag{1–38}$$

To calculate total pressure, you also need to evaluate entropy as a function of enthalpy and pressure, rather than temperature and pressure. For details, see *Total Pressure* (p. 16).

The recipe to do this is essentially the same as for the temperature and pressure recipe. First, you start with differential form for $s\,(h,p)$:

$$ds = \left(\frac{\partial s}{\partial h}\right)_p dh + \left(\frac{\partial s}{\partial p}\right)_h dp \tag{1–39}$$

and comparing this with a slightly rearranged form of the second law:

$$ds = \frac{dh}{T} - \frac{dp}{\rho\,T} \tag{1–40}$$

you get that:

$$\left(\frac{\partial s}{\partial h}\right)_p = \frac{1}{T} \tag{1–41}$$

$$\left(\frac{\partial s}{\partial p}\right)_h = -\frac{1}{\rho\,T} \tag{1–42}$$

In this case, the entropy derivatives with respect to h and p can be evaluated from the constitutive relationship $h\,(T,p)$ (getting T from h and p by table inversion) and the equation of state $r\,(T,p)$. Points in the $s\,(h,p)$ table are evaluated by performing the following integration:

$$s - s_{\mathrm{ref}} = \int_{h_{\mathrm{ref}}}^{h} \frac{1}{T}\,dh - \int_{p_{\mathrm{ref}}}^{p} \frac{1}{\rho\,T}\,dp \tag{1–43}$$

over a range of h and p, which are determined to minimize the integration error.

1.1.3.12. Total Pressure

The total pressure, p_{tot}, is defined as the pressure that would exist at a point if the fluid was brought instantaneously to rest such that the dynamic energy of the flow converted to pressure without losses. The following three sections describe how total pressure is computed for a pure component material with constant density, ideal gas equation of state and a general equation of state (CEL expression or RGP table). For details, see *Multiphase Total Pressure* (p. 166).

The terms Pressure and Total Pressure are *absolute* quantities and must be used to derive the Total Pressure when using an equation of state (compressible) formulation (particularly for *Equation 1–48* (p. 17)).

The setting of the **Total Pressure Option**, described in Compressibility Control in the *CFX-Pre User's Guide*, determines which equation is used to derive the total pressure.

1.1.3.12.1. Incompressible Fluids

For incompressible flows, such as those of liquids and low speed gas flows, the total pressure is given by Bernoulli's equation:

$$p_{\mathrm{tot}} = p_{\mathrm{stat}} + \frac{1}{2}\rho\,(\boldsymbol{U}\cdot\boldsymbol{U}) \tag{1–44}$$

which is the sum of the static and dynamic pressures.

1.1.3.12.2. Ideal Gases

When the flow is compressible, total pressure is computed by starting with the second law of thermodynamics assuming that fluid state variations are locally isentropic so that you get:

$$dh = \frac{dp}{\rho} \tag{1–45}$$

The left hand side of this expression is determined by the constitutive relation and the right hand side by the equation of state. For an ideal gas, the constitutive relation and equation of state are:

$$dh = c_p\,(T)\ dT \tag{1–46}$$

$$\rho = \frac{p}{R\ T} \tag{1–47}$$

which, when substituted into the second law and assuming no entropy variations, gives:

$$p_{\text{tot}} = p_{\text{stat}} \exp\left(\frac{1}{R} \int_{T_{\text{stat}}}^{T_{\text{tot}}} \frac{c_p\,(T)}{T}\ dT \right) \tag{1–48}$$

where T_{stat} and T_{tot} are the static and total temperatures respectively (calculation of these quantities was described in two previous sections, *Static Temperature* (p. 10) and *Total Temperature* (p. 11)). If c_p is a constant, then the integral can be exactly evaluated, but if c_p varies with temperature, then the integral is numerically evaluated using quadrature. For details, see:

- *Entropy* (p. 13)
- *Ideal Gas Equation of State* (p. 28).

1.1.3.12.3. Material with Variable Density and Specific Heat

Total pressure calculations in this case are somewhat more involved. You need to calculate total pressure given the static enthalpy and static pressure, which the CFX-Solver solves for. You also want to assume that the local state variations, within a control volume or on a boundary condition, are isentropic.

Given that you know $h\,(T,p)$ (from integrating the differential form for enthalpy) and $r\,(T,p)$, you can compute two entropy functions $s\,(T,p)$ and $s\,(h,p)$. There are two options for generating these functions:

- If $s\,(T,p)$ and $h\,(T,p)$ are provided by a `.rgp` file then $s\,(h,p)$ is evaluated by interpolation from $h\,(T,p)$ and $s\,(T,p)$ tables.
- If CEL expressions have been given for ρ and c_p only, then $h\,(T,p)$, $s\,(T,p)$ and $s\,(h,p)$ are all evaluated by integrating their differential forms.

Once you have the $s\,(h,p)$ table, calculated as described in *Entropy* (p. 13), computing total pressure for a single pure component is a relatively trivial procedure:

1. The CFX-Solver solves for h_{tot} and p_{stat}
2. Calculate h_{stat} from h_{tot}
3. Calculate entropy $s_{\text{stat}} = s\left(h_{\text{stat}}, p_{\text{stat}}\right)$
4. Using the isentropic assumption set $s_{\text{tot}} = s_{\text{stat}}$
5. Calculate total pressure by inverting $s_{\text{tot}} = s\left(h_{\text{tot}}, p_{\text{tot}}\right)$

For details, see *Rotating Frame Quantities* (p. 18).

1.1.3.13. Shear Strain Rate

The strain rate tensor is defined by:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \tag{1–49}$$

This tensor has three scalar invariants, one of which is often simply called the shear strain rate:

$$\text{sstrnr} = \left[2\,\frac{\partial U_i}{\partial x_j} S_{ij}\right]^{\frac{1}{2}} \tag{1–50}$$

with velocity components U_x, U_y, U_z, this expands to:

$$\begin{aligned}\text{sstrnr} = \Bigg[& 2\left\{\left(\frac{\partial U_x}{\partial x}\right)^2 + \left(\frac{\partial U_y}{\partial y}\right)^2 + \left(\frac{\partial U_z}{\partial z}\right)^2\right\} \\ & + \left(\frac{\partial U_x}{\partial y} + \frac{\partial U_y}{\partial x}\right)^2 + \left(\frac{\partial U_x}{\partial z} + \frac{\partial U_z}{\partial x}\right)^2 + \left(\frac{\partial U_y}{\partial z} + \frac{\partial U_z}{\partial y}\right)^2\Bigg]^{\frac{1}{2}}\end{aligned} \tag{1–51}$$

The viscosity of non-Newtonian fluids is often expressed as a function of this scalar shear strain rate.

1.1.3.14. Rotating Frame Quantities

The velocity in the rotating frame of reference is defined as:

$$\boldsymbol{U}_{\text{rel}} = \boldsymbol{U}_{\text{stn}} - \boldsymbol{\omega} \times \boldsymbol{R} \tag{1–52}$$

where ω is the angular velocity, $\boldsymbol{R}$ is the local radius vector, and $\boldsymbol{U}_{\text{stn}}$ is velocity in the stationary frame of reference.

1.1.3.14.1. Incompressible Fluids

For incompressible flows, the total pressure is defined as:

$$p_{\text{tot}} = p_{\text{stat}} + \frac{1}{2}\rho\left(\boldsymbol{U}_{\text{rel}} \cdot \boldsymbol{U}_{\text{rel}} - \left(\boldsymbol{\omega} \times \boldsymbol{R} \cdot \boldsymbol{\omega} \times \boldsymbol{R}\right)\right) \tag{1–53}$$

where p_{stat} is static pressure. The stationary frame total pressure is defined as:

$$p_{\text{tot,stn}} = p_{\text{stat}} + \frac{1}{2}\rho\left(\boldsymbol{U}_{\text{stn}} \cdot \boldsymbol{U}_{\text{stn}}\right) \tag{1–54}$$

1.1.3.14.2. Ideal Gases

For compressible flows relative total pressure, rotating frame total pressure and stationary frame total pressure are computed in the same manner as in *Total Pressure* (p. 16). First you start with the relative total enthalpy, rothalpy and stationary frame total enthalpy:

$$h_{\text{tot}} = h_{\text{stat}} + \frac{1}{2}\left(\boldsymbol{U}_{\text{rel}} \cdot \boldsymbol{U}_{\text{rel}}\right) \tag{1–55}$$

$$I = h_{\text{stat}} + \frac{1}{2}\left(\boldsymbol{U}_{\text{rel}} \cdot \boldsymbol{U}_{\text{rel}} - \left(\boldsymbol{\omega} \times \boldsymbol{R} \cdot \boldsymbol{\omega} \times \boldsymbol{R}\right)\right) \tag{1–56}$$

$$h_{\text{tot,stn}} = h_{\text{stat}} + \frac{1}{2}\left(\boldsymbol{U}_{\text{stn}} \cdot \boldsymbol{U}_{\text{stn}}\right) \tag{1–57}$$

where h_{stat} is the static enthalpy. In a rotating frame of reference, the CFX-Solver solves for the total enthalpy, h_{tot}, which includes the relative kinetic energy.

Important

Rothalpy (I) is not a positive definite quantity. If the rotation velocity is large, then the last term can be significantly larger than the static enthalpy plus the rotating frame kinetic energy. In this case, it is possible that total temperature and rotating total pressure are undefined and will be clipped at internal table limits or built in lower bounds. However, this is only a problem for high angular velocity rotating systems.

If you again assume an ideal gas equation of state with variable specific heat capacity you can compute relative total temperature, total temperature and stationary frame total temperature using:

$$h_{\text{tot}} - h_{\text{ref}} = \int_{T_{\text{ref}}}^{T_{\text{tot,rel}}} c_p(T)\, dT \tag{1–58}$$

and:

$$I - h_{\text{ref}} = \int_{T_{\text{ref}}}^{T_{\text{tot}}} c_p(T)\, dT \tag{1–59}$$

and:

$$h_{\text{tot,stn}} - h_{\text{ref}} = \int_{T_{\text{ref}}}^{T_{\text{tot,stn}}} c_p(T)\, dT \tag{1–60}$$

where all the total temperature quantities are obtained by inverting the enthalpy table. If c_p is constant, then these values are obtained directly from these definitions:

$$T_{\text{tot,rel}} = T_{\text{stat}} + \frac{\boldsymbol{U}_{\text{rel}} \cdot \boldsymbol{U}_{\text{rel}}}{2\, c_p} \tag{1–61}$$

$$T_{\text{tot}} = T_{\text{stat}} + \frac{(\boldsymbol{U}_{\text{rel}} \cdot \boldsymbol{U}_{\text{rel}} - (\boldsymbol{\omega} \times \boldsymbol{R} \cdot \boldsymbol{\omega} \times \boldsymbol{R}))}{2\, c_p} \tag{1–62}$$

$$T_{\text{tot,stn}} = T_{\text{stat}} + \frac{\boldsymbol{U}_{\text{stn}} \cdot \boldsymbol{U}_{\text{stn}}}{2\, c_p} \tag{1–63}$$

At this point, given $T_{\text{tot,rel}}$, T_{tot}, $T_{\text{tot,stn}}$, p_{stat} and T_{stat} you can compute relative total pressure, total pressure or stationary frame total pressure using the relationship given in the section describing total pressure. For details, see *Total Pressure* (p. 16).

The names of the various total enthalpies, temperatures, and pressures when visualizing results in CFD-Post or for use in CEL expressions is as follows.

Table 1.1 Variable naming: Total Enthalpies, Temperatures, and Pressures

Variable	Long Variable Name	Short Variable Name
h_{tot}	Total Enthalpy	htot
I	Rothalpy	rothalpy

Variable	Long Variable Name	Short Variable Name
$h_{\text{tot,stn}}$	Total Enthalpy in Stn Frame	htotstn
$T_{\text{tot,rel}}$	Total Temperature in Rel Frame	Ttotrel
T_{tot}	Total Temperature	Ttot
$T_{\text{tot,stn}}$	Total Temperature in Stn Frame	Ttotstn
$P_{\text{tot,rel}}$	Total Pressure in Rel Frame	ptotrel
P_{tot}	Total Pressure	ptot
$P_{\text{tot,stn}}$	Total Pressure in Stn Frame	ptotstn

The Mach Number and stationary frame Mach numbers are defined as:

$$M = \frac{|\boldsymbol{U}_{\text{rel}}|}{c} \tag{1–64}$$

$$M_{\text{stn}} = \frac{|\boldsymbol{U}_{\text{stn}}|}{c} \tag{1–65}$$

where c is the local speed of sound.

1.1.3.14.3. Material with Variable Density and Specific Heat

Rotating and stationary frame total temperature and pressure are calculated the same way as described in *Total Temperature* (p. 11) and *Total Pressure* (p. 16). The only changes in the recipes are that rotating frame total pressure and temperature require rothalpy, I, as the starting point and stationary frame total pressure and temperature require stationary frame total enthalpy, $h_{\text{tot,stn}}$.

1.1.3.15. Courant Number

The Courant number is of fundamental importance for transient flows. For a one-dimensional grid, it is defined by:

$$\text{Courant} = \frac{u\ \Delta t}{\Delta x} \tag{1–66}$$

where u is the fluid speed, Δt is the timestep and Δx is the mesh size. The Courant number calculated in ANSYS CFX is a multidimensional generalization of this expression where the velocity and length scale are based on the mass flow into the control volume and the dimension of the control volume.

For explicit CFD methods, the timestep must be chosen such that the Courant number is sufficiently small. The details depend on the particular scheme, but it is usually of order unity. As an implicit code, ANSYS CFX does not require the Courant number to be small for stability. However, for some transient calculations (for example, LES), one may need the Courant number to be small in order to accurately resolve transient details.

For certain compressible flow calculations, the acoustic Courant number is also calculated. For a one-dimensional grid, the acoustic Courant number is defined as:

$$\mathrm{Co}^{+} = \frac{(u+c)\cdot \Delta \mathrm{t}}{\Delta \mathrm{x}} \quad \textbf{(1–67)}$$

$$\mathrm{Co}^{-} = \frac{(\mathrm{u}-c)\cdot \Delta \mathrm{t}}{\Delta \mathrm{x}} \quad \textbf{(1–68)}$$

$$\mathrm{Co}_{\mathrm{acoustic}} = \max\left(\mathrm{Co}^{+}, \mathrm{Co}^{-}\right) \quad \textbf{(1–69)}$$

where c is the local speed of sound.

CFX uses the Courant number in a number of ways:

1. The timestep may be chosen adaptively based on a Courant number condition (for example, to reach RMS or Courant number of 5). The acoustic Courant number is used for compressible flow calculations having the expert parameter setting 'compressible timestepping = t'.
2. For transient runs using the **Automatic** timestep initialization option, the Courant number is used to calculate the blend between the previous timestep and extrapolation options. The acoustic Courant number is used for compressible flow calculations.
3. For transient runs, the maximum and RMS Courant numbers are written to the output file every timestep.
4. The Courant number field is written to the results file.

1.1.4. Mathematical Notation

This section describes the basic notation that is used throughout the CFX-Solver documentation.

1.1.4.1. The Vector Operators

Assume a Cartesian coordinate system in which $\boldsymbol{i}$, $\boldsymbol{j}$ and $\boldsymbol{k}$ are unit vectors in the three coordinate directions. ∇ is defined such that:

$$\nabla = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right] \quad \textbf{(1–70)}$$

1.1.4.1.1. Gradient Operator

For a general scalar function $\varphi(x, y, z)$, the gradient of φ is defined by:

$$\nabla \varphi = \frac{\partial \varphi}{\partial x} \boldsymbol{i} + \frac{\partial \varphi}{\partial y} \boldsymbol{j} + \frac{\partial \varphi}{\partial z} \boldsymbol{k} \tag{1–71}$$

1.1.4.1.2. Divergence Operator

For a vector function $\boldsymbol{U}\ (x, y, z)$ where:

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{U}_x \\ \boldsymbol{U}_y \\ \boldsymbol{U}_z \end{bmatrix} \tag{1–72}$$

the divergence of $\boldsymbol{U}$ is defined by:

$$\nabla \cdot \boldsymbol{U} = \frac{\partial U_x}{\partial x} + \frac{\partial U_y}{\partial y} + \frac{\partial U_z}{\partial z} \tag{1–73}$$

1.1.4.1.3. Dyadic Operator

The dyadic operator (or tensor product) of two vectors, $\boldsymbol{U}$ and $\boldsymbol{V}$, is defined as:

$$\boldsymbol{U} \otimes \boldsymbol{V} = \begin{bmatrix} U_x V_x & U_x V_y & U_x V_z \\ U_y V_x & U_y V_y & U_y V_z \\ U_z V_x & U_z V_y & U_z V_z \end{bmatrix} \tag{1–74}$$

By using specific tensor notation, the equations relating to each dimension can be combined into a single equation. Thus, in the specific tensor notation:

$$\nabla \cdot \left(\rho\, \boldsymbol{U} \otimes \boldsymbol{U} \right) = \begin{bmatrix} \frac{\partial}{\partial x} (\rho\, \boldsymbol{U}_x \boldsymbol{U}_x) + \frac{\partial}{\partial y} \left(\rho\, \boldsymbol{U}_y \boldsymbol{U}_x\right) + \frac{\partial}{\partial z} (\rho\, \boldsymbol{U}_z \boldsymbol{U}_x) \\ \frac{\partial}{\partial x} \left(\rho\, \boldsymbol{U}_x \boldsymbol{U}_y\right) + \frac{\partial}{\partial y} \left(\rho\, \boldsymbol{U}_y \boldsymbol{U}_y\right) + \frac{\partial}{\partial z} \left(\rho\, \boldsymbol{U}_z \boldsymbol{U}_y\right) \\ \frac{\partial}{\partial x} (\rho\, \boldsymbol{U}_x \boldsymbol{U}_z) + \frac{\partial}{\partial y} \left(\rho\, \boldsymbol{U}_y \boldsymbol{U}_z\right) + \frac{\partial}{\partial z} (\rho\, \boldsymbol{U}_z \boldsymbol{U}_z) \end{bmatrix} \tag{1–75}$$

1.1.4.2. Matrix Transposition

The transpose of a matrix is defined by the operator T. For example, if the matrix is defined by:

$$\nabla \varphi = \left[\begin{array}{c} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \\ \frac{\partial \varphi}{\partial z} \end{array} \right] \tag{1–76}$$

then:

$$\left[\nabla \varphi \right]^{\mathrm{T}} = \left[\begin{array}{ccc} \frac{\partial \varphi}{\partial x} & \frac{\partial \varphi}{\partial y} & \frac{\partial \varphi}{\partial z} \end{array} \right] \tag{1–77}$$

1.1.4.3. The Identity Matrix (Kronecker Delta function)

The Identity matrix is defined by:

$$\delta = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array} \right] \tag{1–78}$$

1.1.4.4. Index Notation

Although index notation is not generally used in this documentation, the following may help you if you are used to index notation.

In index notation, the divergence operator can be written:

$$\nabla \cdot \boldsymbol{U} = \frac{\partial U_i}{\partial x_i} \tag{1–79}$$

where the summation convention is followed; that is, the index i is summed over the three components.

The quantity $\boldsymbol{U} \otimes \boldsymbol{V}$ can be represented by $U_i \, V_j$ (when $\boldsymbol{U}$ and $\boldsymbol{V}$ are vectors), or by $U_i \, V_{jk}$ (when $\boldsymbol{U}$ is a vector and $\boldsymbol{V}$ is a matrix), and so on.

Hence, the quantity $\nabla \cdot \left(\rho \, \boldsymbol{U} \otimes \boldsymbol{U} \right)$ can be represented by:

$$\frac{\partial}{\partial x_i} \left(\rho \, U_i \, U_j \right) \tag{1–80}$$

Note the convention that the derivatives arising from the divergence operator are derivatives with respect to the same coordinate as the first listed vector. That is, the quantity $\nabla \cdot \left(\rho \, \boldsymbol{U} \otimes \boldsymbol{U} \right)$ is represented by:

$$\frac{\partial}{\partial x_i}\left(\rho\, U_i\, U_j\right) \tag{1–81}$$

and not:

$$\frac{\partial}{\partial x_j}\left(\rho\, U_i\, U_j\right) \tag{1–82}$$

The quantity $a \cdot b$ (when a and b are matrices) can be written by $a_{ij}\, b_{ij}$.

1.2. Governing Equations

The set of equations solved by ANSYS CFX are the unsteady Navier-Stokes equations in their conservation form.

If you are new to CFD, review Computational Fluid Dynamics in the *CFX Introduction*.

A list of recommended books on CFD and related subjects is available in Further Background Reading in the *CFX Introduction*.

For all the following equations, static (thermodynamic) quantities are given unless otherwise stated.

1.2.1. Transport Equations

In this section, the instantaneous equation of mass, momentum, and energy conservation are presented. For turbulent flows, the instantaneous equations are averaged leading to additional terms. These terms, together with models for them, are discussed in *Turbulence and Wall Function Theory* (p. 89).

The instantaneous equations of mass, momentum and energy conservation can be written as follows in a stationary frame:

1.2.1.1. The Continuity Equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho\, \boldsymbol{U}) = 0 \tag{1–83}$$

1.2.1.2. The Momentum Equations

$$\frac{\partial\, (\rho\, \boldsymbol{U})}{\partial t} + \nabla \cdot \left(\rho\, \boldsymbol{U} \otimes \boldsymbol{U}\right) = -\nabla p + \nabla \cdot \tau + \boldsymbol{S}_M \tag{1–84}$$

Where the stress tensor, τ, is related to the strain rate by

$$\tau = \mu \left(\nabla \boldsymbol{U} + (\nabla \boldsymbol{U})^T - \frac{2}{3} \delta \nabla \cdot \boldsymbol{U} \right) \tag{1–85}$$

1.2.1.3. The Total Energy Equation

$$\frac{\partial (\rho h_{\text{tot}})}{\partial t} - \frac{\partial p}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} h_{\text{tot}}) = \nabla \cdot (\lambda \nabla T) + \nabla \cdot (\boldsymbol{U} \cdot \tau) + \boldsymbol{U} \cdot \boldsymbol{S}_M + \boldsymbol{S}_E \tag{1–86}$$

Where h_{tot} is the total enthalpy, related to the static enthalpy $h(T, p)$ by:

$$h_{\text{tot}} = h + \frac{1}{2} \boldsymbol{U}^2 \tag{1–87}$$

The term $\nabla \cdot (\boldsymbol{U} \cdot \tau)$ represents the work due to viscous stresses and is called the viscous work term. This models the internal heating by viscosity in the fluid, and is negligible in most flows.

The term $\boldsymbol{U} \cdot \boldsymbol{S}_M$ represents the work due to external momentum sources and is currently neglected.

1.2.1.4. The Thermal Energy Equation

An alternative form of the energy equation, which is suitable for low-speed flows, is also available. To derive it, an equation is required for the mechanical energy K.

$$K = \frac{1}{2} \boldsymbol{U}^2 \tag{1–88}$$

The mechanical energy equation is derived by taking the dot product of $\boldsymbol{U}$ with the momentum equation (*Equation 1–84* (p. 25)):

$$\frac{\partial (\rho K)}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} K) = -\boldsymbol{U} \cdot \nabla p + \boldsymbol{U} \cdot (\nabla \cdot \tau) + \boldsymbol{U} \cdot \boldsymbol{S}_M \tag{1–89}$$

Subtracting this equation from the total energy equation (*Equation 1–86* (p. 26)) yields the thermal energy equation:

$$\frac{\partial (\rho h)}{\partial t} - \frac{\partial p}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} h) = \nabla \cdot (\lambda \nabla T) + \boldsymbol{U} \cdot \nabla p + \tau : \nabla \boldsymbol{U} + \boldsymbol{S}_E \tag{1–90}$$

The term $\tau : \nabla \boldsymbol{U}$ is always positive and is called the viscous dissipation. This models the internal heating by viscosity in the fluid, and is negligible in most flows.

With further assumptions discussed in a moment, we obtain the thermal energy equation:

$$\frac{\partial (\rho h)}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} h) = \nabla \cdot (\lambda \nabla T) + \tau : \nabla \boldsymbol{U} + \boldsymbol{S}_E \quad \textbf{(1–91)}$$

This equation can be derived from *Equation 1–90* (p. 26) with two different sets of assumptions:

- If h is actually interpreted as internal energy,

  $$e = h - \frac{P}{\rho} \quad \textbf{(1–92)}$$

  then *Equation 1–90* (p. 26)can be written as

  $$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} e) = \nabla \cdot (\lambda \nabla T) - p \nabla \cdot \boldsymbol{U} + \tau : \nabla \boldsymbol{U} + \boldsymbol{S}_E \quad \textbf{(1–93)}$$

  which is equivalent to *Equation 1–91* (p. 27) if we neglect $-p \nabla \cdot \boldsymbol{U}$ and interpret h as e. This interpretation is appropriate for liquids, where variable-density effects are negligible. Note that the principal variable is still called 'Static Enthalpy' in CFD-Post, although it actually represents internal energy. Note also that, for liquids that have variable specific heats (for example, set as a CEL expression or using an RGP table or Redlich Kwong equation of state) the solver includes the P/ρ contribution in the enthalpy tables. This is inconsistent, because the variable is actually internal energy. For this reason, the thermal energy equation should not be used in this situation, particularly for subcooled liquids.

- On the other hand if $\frac{\partial p}{\partial t}$ and $\boldsymbol{U} \cdot \nabla p$ are neglected in *Equation 1–90* (p. 26) then *Equation 1–91* (p. 27) follows directly. This interpretation is appropriate for low Mach number flows of compressible gases.

The thermal energy equation, despite being a simplification, can be useful for both liquids and gases in avoiding potential stability issues with the total energy formulation. For example, the thermal energy equation is often preferred for transient liquid simulations. On the other hand, if proper acoustic behavior is required (for example, predicting sound speed), or for high speed flow, then the total energy equation is required.

1.2.2. Equations of State

The transport equations described above must be augmented with constitutive equations of state for density and for enthalpy in order to form a closed system. In the most general case, these state equations have the form:

$$\rho = \rho(p, T)$$

$$dh = \frac{\partial h}{\partial T}\Big|_p dT + \frac{\partial h}{\partial p}\Big|_T dp$$

$$= c_p \, dT + \frac{\partial h}{\partial p}\Big|_T dp$$

$$c_p = c_p(p, T)$$

Various special cases for particular material types are described below.

1.2.2.1. Incompressible Equation of State

This is the simplest case: density is constant and c_p can be (at most) a function of temperature.

$$\rho = \rho_{spec}$$

$$d h = c_p \, d T + \frac{d p}{\rho}$$

$$c_p = c_p (T)$$

1.2.2.2. Ideal Gas Equation of State

For an ideal gas, density is calculated from the Ideal Gas law and c_p can be (at most) a function of temperature:

$$\rho = \frac{w p_{abs}}{R_0 T}$$

$$d h = c_p \, d T$$

$$c_p = c_p (T)$$

where w is the molecular weight, p_{abs} is the absolute pressure, and R_0 is the universal gas constant.

1.2.2.3. Real Gas and Liquid Equations of State

In the current version of ANSYS CFX, the Redlich Kwong equation of state is available as a built-in option for simulating real gases. It is also available through several pre-supplied CFX-TASCflow RGP files. The Vukalovich Virial equation of state is also available but currently only by using CFX-TASCflow RGP tables.

Note

It is not possible to couple particles with gases that use the 'real gas' equations of state.

This limitation is caused by the way the particle tracker stores and accesses material properties for coupled gas components and particle components. For performance reasons, the particle tracker uses its own material database--which is limited to either constant properties or temperature-dependent properties (specific heats, enthalpy). The property database of the particle tracker does not support pressure-dependent properties.

1.2.2.3.1. Real Gas Properties

Cubic equations of state are a convenient means for predicting real fluid behavior. They are highly useful from an engineering standpoint because they generally only require that the user know the fluid critical point properties, and for some versions, the acentric factor. These properties are well known for many pure substances or can be estimated if not available. They are called cubic equations of state because, when rearranged as a function volume they are cubic in volume. This means that cubic state equations can be used to predict both liquid and vapor volumes at a given pressure and temperature. Generally the lowest root is the liquid volume and the higher root is the vapor volume.

Four versions of cubic state equations are available: Standard Redlich Kwong, Aungier Redlich Kwong, Soave Redlich Kwong, and Peng Robinson. The Redlich-Kwong equation of state was first published in 1949 [85] and is considered one of the most accurate two-parameter corresponding states equations of state. More recently, Aungier (1995) [96] has modified the Redlich-Kwong equation of state so that it provides much better accuracy near the critical point. The Aungier form of this equation of state is the default cubic equation used by ANSYS CFX. The Peng Robinson [157]] and Soave Redlich Kwong equation of state were developed to overcome the shortcomings of the Redlich Kwong equations to accurately predict liquid properties and vapor-liquid equilibrium.

1.2.2.3.1.1. Redlich Kwong Models

The Redlich Kwong variants of the cubic equations of state are written as:

$$p = \frac{R\,T}{v - b + c} - \frac{a\,(T)}{v\,(v + b)} \tag{1–94}$$

where v is the specific volume $(v = 1/\rho)$.

1.2.2.3.1.1.1. The Standard Redlich Kwong Model

The Standard Redlich Kwong model sets the parameter c to zero, and the function a to:

$$a = a_0 \left(\frac{T}{T_c} \right)^{-n} \tag{1–95}$$

where n is 0.5 and

$$a_0 = \frac{0.42747\,R^2\,T_c^2}{p_c} \tag{1–96}$$

$$b = \frac{0.08664\,R\,T_c}{p_c} \tag{1–97}$$

1.2.2.3.1.1.2. The Aungier Redlich Kwong Model

The Aungier form differs from the original by a non-zero parameter c which is added to improve the behavior of isotherms near the critical point, as well as setting the exponent n differently. The parameter c in Equation 1–94 (p. 29) is given by:

$$c = \frac{R\,T_c}{p_c + \frac{a_0}{v_c\left(v_c + b\right)}} + b - v_c \tag{1–98}$$

and the standard Redlich Kwong exponent of $n = 0.5$ is replaced by a general exponent n. Optimum values of n depend on the pure substance. Aungier (1995) [96] presented values for twelve experimental data sets to which he provided a best fit polynomial for the temperature exponent n in terms of the acentric factor, ω:

$$n = 0.4986 + 1.1735\,\omega + 0.4754\,\omega^2 \tag{1–99}$$

1.2.2.3.1.1.3. The Soave Redlich Kwong Model

The Soave Redlich Kwong real gas model was originally published by Soave (1972). The model applies to non-polar compounds and improves on the original Redlich Kwong model by generalizing the attraction term to depend on the acentric factor, which accounts for molecules being non-spherical, and accounting for a range of vapor pressure data in the development of the temperature dependency of this parameter.

The Soave Redlich Kwong equation is explicit in pressure, and given by:

$$p = \frac{R\,T}{v - b + c} - \frac{a\,(T\,)}{v\,(\,v + b\,)} \tag{1–100}$$

where:

$$b = 0.08664\frac{RT_c}{p_c} \tag{1–101}$$

as in the original Redlich Kwong equation, and

$$a\,(T\,) = a_0\left(1 + n\left(1 - \sqrt{\frac{T}{T_c}}\right)\right)^2 \tag{1–102}$$

where:

$$a_0 = 0.42747\frac{R^2 T_c^{\,2}}{P_c} \tag{1–103}$$

and the parameter *n* is computed as a function of the acentric factor, ω:

$$n = 0.480 + 1.574\omega - 0.176\omega^2 \quad \textbf{(1–104)}$$

1.2.2.3.1.2. Peng Robinson Model

The Peng Robinson model also gives pressure as a function of temperature and volume:

$$p = \frac{RT}{v-b} - \frac{a(T)}{v^2 - 2bv + b^2} \quad \textbf{(1–105)}$$

where:

$$b = 0.0778\frac{RT_c}{p_c} \quad \textbf{(1–106)}$$

as in the original Redlich Kwong equation, and

$$a(T) = a_0\left(1 + n\left(1 - \sqrt{\frac{T}{T_c}}\right)\right)^2 \quad \textbf{(1–107)}$$

where:

$$a_0 = 0.45724\frac{R^2 T_c^{\,2}}{P_c} \quad \textbf{(1–108)}$$

and the parameter *n* is computed as a function of the acentric factor, ω:

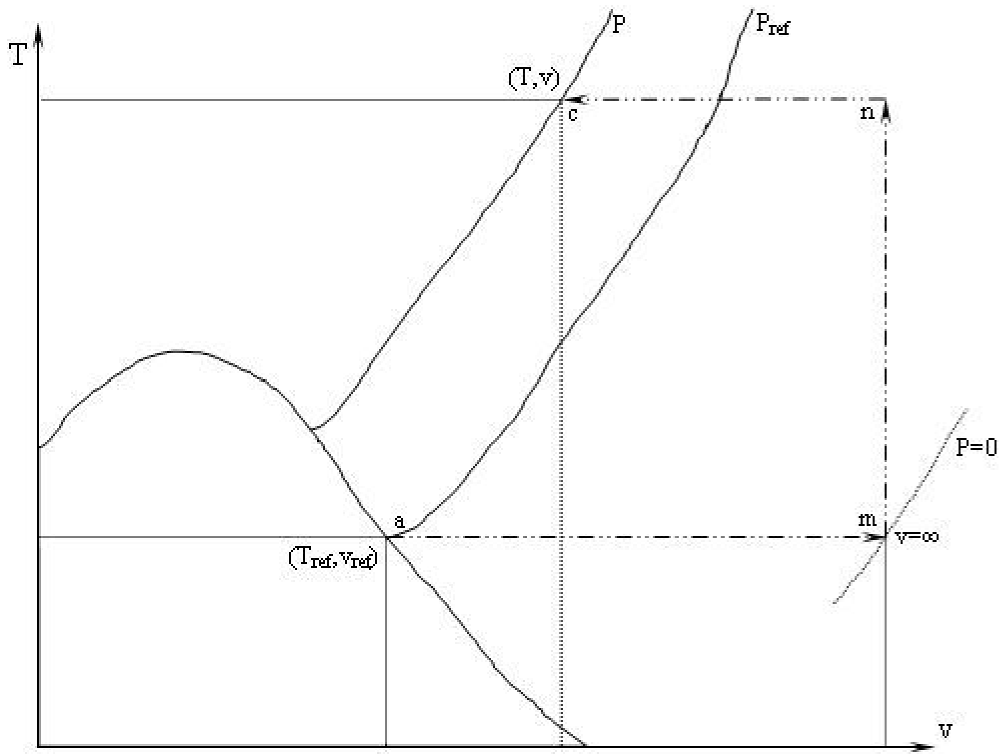
$$n = 0.37464 + 1.54226\omega - 0.26993\omega^2 \quad \textbf{(1–109)}$$

1.2.2.3.1.3. Real Gas Constitutive Relations

In order to provide a full description of the gas properties, the flow solver must also calculate enthalpy and entropy. These are evaluated using slight variations on the general relationships for enthalpy and entropy that were presented in the previous section on variable definitions. The variations depend on the zero pressure, ideal gas, specific heat capacity and derivatives of the equation of state. The zero pressure specific heat capacity must be supplied to ANSYS CFX while the derivatives are analytically evaluated from *Equation 1–94* (p. 29)and *Equation 1–105* (p. 31).

Internal energy is calculated as a function of temperature and volume (T,v) by integrating from the reference state (T_{ref},v_{ref}) along path 'amnc' (see diagram below) to the required state (T,v) using the following differential relationship:

$$du = c_{\mathrm{v}}\, dT + \left(T \left(\frac{dp}{dT} \right)_{\mathrm{v}} - p \right) dv \tag{1–110}$$



First the energy change is calculated at constant temperature from the reference volume to infinite volume (ideal gas state), then the energy change is evaluated at constant volume using the ideal gas c_{v}. The final integration, also at constant temperature, subtracts the energy change from infinite volume to the required volume. In integral form, the energy change along this path is:

$$u\,(T, v) - u\,(T_{\mathrm{ref}}, v_{\mathrm{ref}}) = \int_{v_{\mathrm{ref}}}^{\infty} \left(T \left(\frac{dp}{dT} \right)_{\mathrm{v}} - p \right) dv_{T_{\mathrm{ref}}} + \int_{T_{\mathrm{ref}}}^{T} c_{\mathrm{v}0}\, dT - \int_{v}^{\infty} \left(T \left(\frac{dp}{dT} \right)_{\mathrm{v}} - p \right) dv_{T} \tag{1–111}$$

Once the internal energy is known, then enthalpy is evaluated from internal energy:

$$h = u + p\, v$$

The entropy change is similarly evaluated:

$$s(T, v) - s(T_{\text{ref}}, v_{\text{ref}}) = \int_{v_{\text{ref}}}^{\infty} \left(\frac{dp}{dT} \right)_{\text{v}} dv_{T_{\text{ref}}} + \int_{T_{\text{ref}}}^{T} \frac{c_{p0}}{T} dT - R \ln \left(\frac{p}{p_{\text{ref}}} \right) - \int_{v}^{\infty} \left(\frac{dp}{dT} \right)_{\text{v}} dv_T \tag{1–112}$$

where $c_{p\,o}$ is the zero pressure ideal gas specific heat capacity. By default, ANSYS CFX uses a 4th order polynomial for this and requires that coefficients of that polynomial are available. These coefficients are tabulated in various references including Poling et al. [84].

In addition, a suitable reference state must be selected to carry out the integrations. The selection of this state is arbitrary, and can be set by the user, but by default, ANSYS CFX uses the normal boiling temperature (which is provided) as the reference temperature and the reference pressure is set to the value of the vapor pressure evaluated using *Equation 1–118* (p. 34) at the normal boiling point. The reference enthalpy and entropy are set to zero at this point by default, but can also be overridden if desired.

Other properties, such as the specific heat capacity at constant volume, can be evaluated from the internal energy. For example, the Redlich Kwong model uses:

$$c_v = \left(\frac{\partial u}{\partial T} \right)_v = \frac{\partial u_0}{\partial T} - \frac{n(n+1)\ a}{b\ T} \log \left(1 + \frac{b}{v} \right) \tag{1–113}$$

where u_0 is the ideal gas portion of the internal energy:

$$u_0 = u - u_{\text{ref}} = \int_{T_{\text{ref}}}^{T} \left(c_{po}(T) - R \right) dT \tag{1–114}$$

specific heat capacity at constant pressure, c_p, is calculated from c_v using:

$$c_p = c_v + v\ T\ \frac{\beta^2}{\kappa} \tag{1–115}$$

where β and κ are the volume expansivity and isothermal compressibility, respectively. These two values are functions of derivatives of the equation of state and are given by:

$$\beta = -\frac{\left(\frac{\partial p}{\partial T}\right)_v}{v\left(\frac{\partial p}{\partial v}\right)_T} \tag{1–116}$$

$$\kappa = -\frac{1}{v\left(\frac{\partial p}{\partial v}\right)_T} \tag{1–117}$$

1.2.2.3.2. Real Gas Saturated Vapor Properties

When running calculations with liquid condensing out of the vapor (equilibrium phase change or Eulerian thermal phase change model) or dry calculations, the flow solver needs to know the form of the vapor pressure curve. Pressure and temperature are not independent in the saturation dome, so vapor saturation properties are evaluated by first assuming an equation that gives the dependence of vapor pressure on temperature, and then substituting that into the equation of state.

For materials that use the cubic equations of state ANSYS CFX approximates the vapor pressure curve using a form given by Poling et al. [84]:

$$\log_{10}\left(\frac{p_{\mathrm{v}}}{p_{\mathrm{c}}}\right) = \frac{7}{3}\,(1+\omega)\left(1-\frac{T_{\mathrm{c}}}{T}\right) \tag{1–118}$$

Vapor saturation properties are calculated by evaluating the equation of state and constitutive relations along the saturation curve.

1.2.2.3.3. Real Gas Liquid Properties

As previously mentioned, it is possible to derive liquid densities directly from the cubic state equations, however, this is not always desirable. For example, the Redlich Kwong models are very inaccurate, and sometimes completely wrong, in the compressed liquid regime.

Instead, when you select to use one of the Redlich Kwong variants for a liquid, the properties are assumed to vary along the vapor pressure curve as a function of saturation temperature. These properties are approximate and should only be used when the amount of liquid in your calculation will be small. For example, they work well with the equilibrium condensation model or non-equilibrium small droplet phase change model.

To derive the liquid enthalpy and entropy, such that they are completely consistent with the gas phase, requires all the same data as is provided for the gas phase: the critical point data, the acentric factor and the zero pressure specific heat coefficients.

To calculate saturated liquid densities, an alternative equation of state originally published by Yamada and Gunn (1973), is used by CFX that gives liquid specific volume as a function of temperature:

$$v = v_c\,(0.29056 - 0.08775\,\omega)^{(1 - T/T_c)^{2/7}} \tag{1–119}$$

This equation is convenient because it only requires knowledge of the critical volume and temperature as well as the acentric factor. The valid temperature range for the liquid equation of state is $0.4T_c < T < 0.99T_c$. The solver will clip the temperature used in this equation to that range.

Saturated liquid enthalpy is calculated using knowledge of the gas saturation enthalpy and the following equation:

$$h_{f,s} = h_{g,s} - h_{fg} \tag{1–120}$$

where the enthalpy of vaporization, h_{fg}, is given by the following expression taken from Poling et al. [84]:

$$\frac{h_{fg}}{R\,T_c} = 7.08\left(1 - \frac{T}{T_c}\right)^{0.354} + 10.95\,\omega\left(1 - \frac{T}{T_c}\right)^{0.456} \tag{1–121}$$

Saturated liquid entropy can easily be derived using the second law and the gas saturation entropy:

$$s_f = s_g - \frac{h_{fg}}{T} \tag{1–122}$$

Prediction of liquid specific heat capacity with the Redlich Kwong equation has a similar problem to the liquid density, so ANSYS CFX uses an alternative form presented by Aungier (2000):

$$c_p = c_{p\,0} + R\left(3.67 + 11.64\left(1 - \frac{T}{T_c}\right)^{4} + 0.634\left(1 - \frac{T}{T_c}\right)^{-1}\right) \tag{1–123}$$

which requires knowledge of the zero pressure heat capacity coefficients, as well as the acentric factor. For the saturated liquid it is assumed that $c_p = c_v$.

Note that the use either of the Redlich Kwong models for flows of almost entirely pure liquid is highly discouraged. If you want to use one of the cubic equations of state for this type of problem, then use the Peng Robinson equation of state which, by default, will force the liquid phase properties to be dependent on temperature and pressure fully consistent with that equation of state.

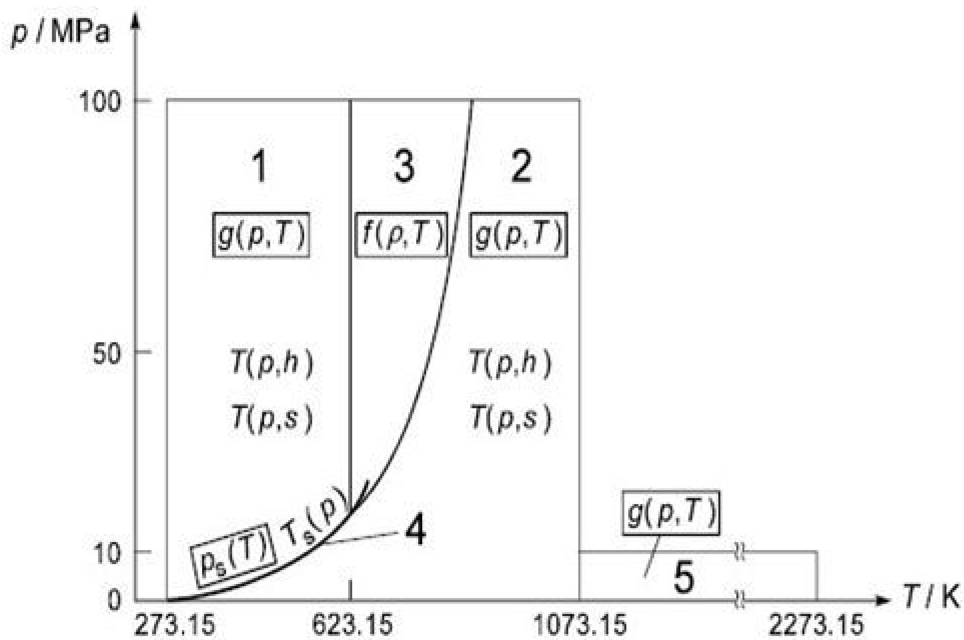
1.2.2.3.4. IAPWS Equation of State

The IAPWS-IF97 database represents an accurate equation of state for water and steam properties. The database is fully described elsewhere [125], but a summary will be provided in this section. The IAPWS database uses formulations for five distinct thermodynamic regions for water and steam, namely:

- subcooled water (1)
- supercritical water/steam (2)

- superheated steam (3)
- saturation data (4)
- high temperature steam (5)

Figure 1.1 Regions and Equations of IAPWS-IF97



Region 5 has not been implemented in ANSYS CFX because it represents a thermodynamic space at very high temperatures (1073.15 - 2273.15 K) and reasonably low pressures (0-10 MPa) that can be adequately described using other property databases already in ANSYS CFX (that is, Ideal Gas EOS with NASA specific heat and enthalpy). Furthermore, because this region is not defined for pressures up to 100 MPa, as is the case for regions 1, 2 and 3, problems arise in filling out the pressure-temperature space in the tables when temperatures exceed 1073.15 K and pressures exceed 10 MPa. The database implemented in CFX therefore covers temperatures ranging from 273.15 to 1073.15 K and pressures ranging from 611 Pa to 100 MPa.

The reference state for the IAPWS library is the triple point of water. Internal energy, entropy and enthalpy are all set to zero at this point.

T_{ref} = 273.16 K, P_{ref} = 611.657 Pa, u_{liquid} = 0 J/kg, s_{liquid} = 0 J/kg/K, h_{liquid} = 0 J/kg

In ANSYS CFX, the analytical equation of state is used to transfer properties into tabular form, which can be evaluated efficiently in a CFD calculation. These IAPWS tables are defined in terms of pressure and temperature, which are then inverted to evaluate states in terms of other property combinations (such as pressure/enthalpy or entropy/enthalpy). When developing the IAPWS database for ANSYS CFX, therefore, properties must be evaluated as functions of pressure and temperature. For the most part, this involves a straightforward implementation of the equations described in the IAPWS theory [125]. Region 4 involves saturation data that uses only pressure or temperature information.

However, some difficulties are encountered when evaluating the properties around Region 3 (near the critical point), where the EOS is defined explicitly in terms of density and temperature. In this region, the density must be evaluated using Newton-Raphson iteration. This algorithm is further complicated in that the EOS is applicable on both the subcooled liquid and superheated vapor side leading up to critical conditions. Therefore, depending on the pressure-temperature state, one may be evaluating a subcooled liquid or a superheated vapor with the same EOS. To apply the Newton-Raphson scheme in a reliable way, one must detect on which side of the saturation dome the pressure-temperature state applies, and apply an appropriate initial guess. Such an iteration scheme, including logic for an initial guess, has been implemented in ANSYS CFX so that table generation around the critical region is possible.

1.2.2.3.5. Metastable Superheated Liquid/Supercooled Vapor States

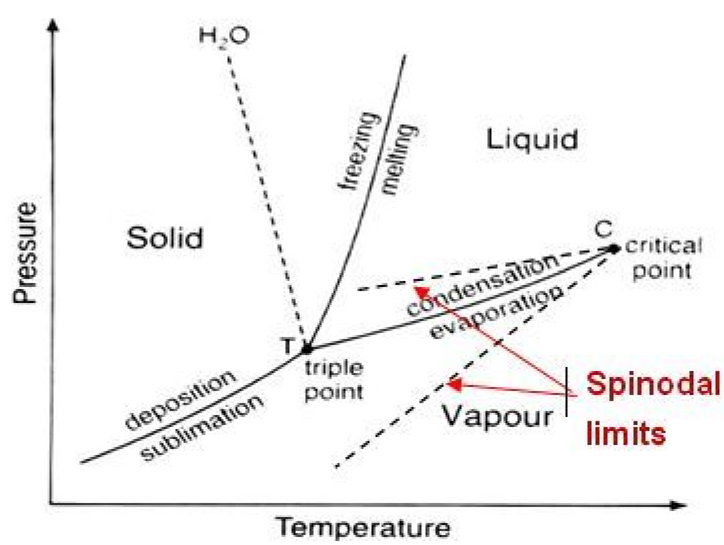
The IAPWS library also extends to metastable states, so that this equation of state is available for non-equilibrium phase change models such as the Droplet Condensation Model. The EOS for regions 1 and 3 in *Figure 1.1* (p. 36) are stated to have reasonable accuracy for metastable states close to the saturation line d liquid states [125]. However, the term "reasonable" is not quantified, and therefore the degree to which the extrapolation of the EOS can be applied is unknown.

In region 2, an additional set of equations have been developed for supercooled vapor conditions under 10 MPa. These equations have been tuned to match the saturation data. Above 10 MPa, the EOS for the superheated region can safely be extrapolated into supercooled conditions, but it does not match smoothly with the specialized supercooled equations below 10 MPa.

1.2.2.3.6. Numerical Testing to Delineate Metastable Regions

In order to make the IAPWS database as robust as possible, numerical testing has been done to determine approximate metastable vapor/liquid spinodal lines. *Figure 1.2* (p. 37) is given to demonstrate these spinodal lines that are essentially boundaries up to which metastable conditions can exist. These would be defined similar to saturation curves as functions of either temperature or pressure. The IAPWS database tables are always generated up to these limits regardless of what flow models are specified (equilibrium or non-equilibrium) and thus allow non-equilibrium phase change models to be applied.

Figure 1.2 Spinodal Limits Built into Tables



1.2.2.3.7. The Acentric Factor

The acentric factor must be supplied when running the real gas models and is tabulated for many common fluids in Poling et al. [84]. If you do not know the acentric factor, or it is not printed in a common reference, it can be estimated using knowledge of the critical point and the vapor pressure curve with this formula:

$$\omega = -\log_{10}\left(\frac{p_{\mathrm{v}}}{p_{\mathrm{c}}}\right) - 1 \quad \textbf{(1–124)}$$

where the vapor pressure, p_{v}, is calculated at $T = 0.7\ T_{\mathrm{c}}$. In addition to the critical point pressure, this formula requires knowledge of the vapor pressure as a function of temperature.

1.2.2.4. General Equation of State

User-defined equations of state are also supported:

$$\rho = \rho(p, T)$$

$$c_p = c_p(p, T)$$

Noting that:

$$\frac{\partial h}{\partial p}\Big|_T = v - T\frac{\partial v}{\partial T}\Big|_p$$

where $v = 1/\rho$,

the equation of state for enthalpy therefore follows as:

$$dh = c_p\, dT + \left[v - T\frac{\partial v}{\partial T}\Big|_p\right]_T dp \quad \textbf{(1–125)}$$

Note, however, that user-defined expressions for ρ and c_p must be thermodynamically consistent. Consistency requires that mathematical properties for exact differentials be satisfied. For example, suppose $dz(x, y)$ is an exact differential defined as:

$$dz\ (x, y) = M\,dx + N\,dy$$

$$M = \frac{\partial z}{\partial x}\Big|_y$$

$$N = \frac{\partial z}{\partial y}\Big|_x$$

Consistency then requires that:

$$\frac{\partial M}{\partial y}\Big|_x = \frac{\partial N}{\partial x}\Big|_y$$

Applying this concept to *Equation 1–125* (p. 38), it therefore follows that general equations of state must obey:

$$\frac{\partial c_p}{\partial p} = \frac{\partial}{\partial T}\left(v - T \frac{\partial v}{\partial T} \Big|_p \right)$$

1.2.3. Conjugate Heat Transfer

CFX enables you to create solid regions in which the equations for heat transfer are solved, but with no flow. This is known as *conjugate heat transfer*, and the solid regions are known as *solid domains*.

Within solid domains, the conservation of energy equation can account for heat transport due to solid motion, conduction, and volumetric heat sources:

$$\frac{\partial (\rho h)}{\partial t} + \nabla \cdot (\rho \boldsymbol{U}_s h) = \nabla \cdot (\boldsymbol{\lambda} \nabla T) + S_E \tag{1–126}$$

where h, ρ, and λ are the enthalpy, density, and thermal conductivity of the solid, respectively. $\boldsymbol{U}_s$ is the solid velocity, if specified, and S_E is an optional volumetric heat source. The solid motion advection term (the term including $\boldsymbol{U}_s$) is optional and is added only when a solid motion velocity is set.

Additional information on plotting variables at a solid-fluid interface is available in Solid-Fluid Interface Variable Values in the *CFX Reference Guide*.

1.3. Buoyancy

For buoyancy calculations, a source term is added to the momentum equations as follows:

$$S_{\mathrm{M,buoy}} = \left(\rho - \rho_{\mathrm{ref}}\right) g \tag{1–127}$$

The density difference $\rho - \rho_{\mathrm{ref}}$ is evaluated using either the Full Buoyancy model or the Boussinesq model, depending on the physics.

When buoyancy is activated, the pressure in the momentum equation excludes the hydrostatic gradient due to ρ_{ref}. This pressure is related to the absolute pressure as follows:

$$p_{\mathrm{abs}} = p + p_{\mathrm{ref}} + \rho_{\mathrm{ref}} \vec{g} \left(\vec{r} - \vec{r}_{\mathrm{ref}} \right) \tag{1–128}$$

where $\vec{r}_{\mathrm{ref}}$ is a reference location. The buoyancy reference location option is set under the `Ref. Location` option in CFX-Pre and can be set to a particular location. For details, see Buoyancy and Pressure.

Absolute pressure is used to evaluate fluid properties that are functions of pressure; it can be visualized in CFD-Post.

1.3.1. Full Buoyancy Model

For buoyancy calculations involving variable density, $\rho - \rho_{\mathrm{ref}}$ is evaluated directly. This option is set automatically when the simulation involves multicomponent flow, multiphase flow, or a fluid having density set as a function of pressure, temperature, or other field variables.

1.3.2. Boussinesq Model

For buoyant flows where the density variation is driven only by small temperature variations, the Boussinesq model is used. In this model, a constant reference density ρ_{ref} is used for all terms other than the buoyancy source term. The buoyancy source term is approximated as:

$$\rho - \rho_{\mathrm{ref}} = -\rho_{\mathrm{ref}}\beta\left(T - T_{\mathrm{ref}}\right) \tag{1–129}$$

where β is the thermal expansivity:

$$\beta = -\frac{1}{\rho}\frac{\partial\rho}{\partial T}\Big|_{p} \tag{1–130}$$

and T_{ref} is the buoyancy reference temperature.

1.4. Immersed Solids

The following topics are discussed:

1.4.1. Immersed Solid Boundary Details

This section describes notation used in the following sections pertaining to immersed solids.

Figure 1.3 Notation for Nodes Near the Immersed Solid Boundary

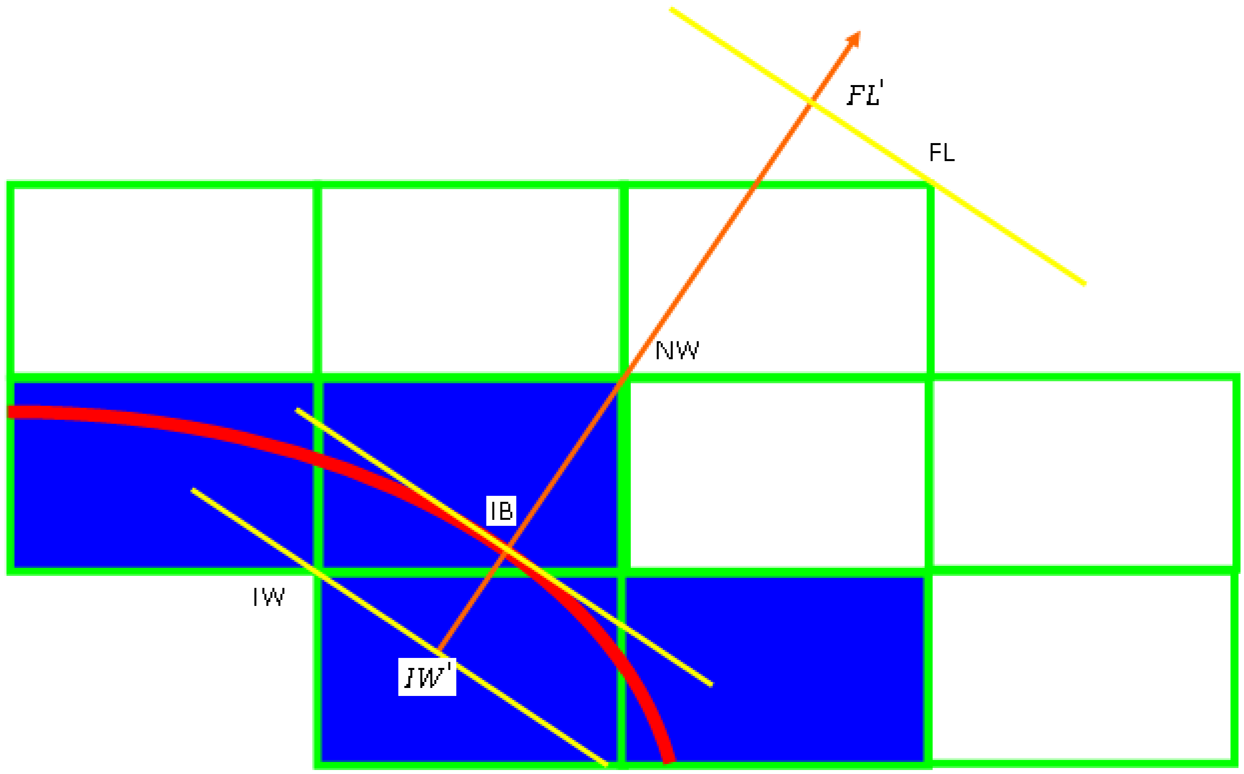


Figure 1.3 (p. 41) shows a schematic diagram of an immersed solid boundary (shown as a red line). The nodes and points near the boundary are labeled for reference in the following sections. The meaning of each label is tabulated below.

Label	Meaning
IW	in-wall node
IW′	in-wall point
NW	near-wall node
IB	point on the immersed solid boundary
FL	fluid node
FL′	fluid point

In the figure, the orange arrow represents the wall normal direction corresponding to the given NW node.

Other notes about notation:

- "NIBG" elements are elements between corresponding NW and IW nodes. These are elements which the immersed solid surface cuts through, and are colored blue in the figure above.
- "NIBF" elements are elements between corresponding FL and NW nodes. These are elements which are immediately adjacent to the "NIBG" elements and are colored white in the figure above.
- Δy is the wall distance between the NW node and the IB point along the wall normal direction.
- $(\Delta y)_{\mathrm{NIBG}}$ is the wall distance between the NW node and the IW′ point.

- $(\Delta y)_{\mathrm{NIBF}}$ is the wall distance between the FL$'$ point and the NW node.

1.4.2. Laminar Flow Treatment

For laminar flows, no wall scale equation is solved. Instead, the velocity near the immersed boundary is computed based on the wall distance, assuming constant wall shear in the boundary profile region.

The tangential fluid velocity at the near-wall fluid nodes is governed by:

$$(U_{\mathrm{NW}}{}^{\mathrm{F}})_{\mathrm{Tang}} = (U_{\mathrm{IB}}{}^{\mathrm{IMS}})_{\mathrm{Tang}} + \frac{\mu_{\mathrm{FL}}}{\mu_{\mathrm{NW}}} \frac{\Delta y}{(\Delta y)_{\mathrm{NIBF}} + \Delta y} [(U_{\mathrm{FL}})_{\mathrm{Tang}} - (U_{\mathrm{IB}}{}^{\mathrm{IMS}})_{\mathrm{Tang}}] \quad \textbf{(1–131)}$$

where $(U_{\mathrm{NW}}{}^{\mathrm{F}})_{\mathrm{Tang}}$ is the tangential component of the forcing velocity at the NW node, μ_{NW} is the dynamic viscosity at the NW node, U_{FL} and μ_{FL} are, respectively, the fluid velocity and dynamic viscosity at the FL node, and $(\Delta y)_{\mathrm{NIBF}}$ is the distance between the corresponding FL$'$ point and NW node. Note that $[(\Delta y)_{\mathrm{NIBF}} + \Delta y]$ is essentially the normal distance between this node and the immersed boundary.

The normal component of the forcing velocity at the NW node is computed as the normal component of the velocity at the IB point on the immersed boundary:

$$\left(U_{\mathrm{IB}}{}^{\mathrm{IMS}}\right)_{\mathrm{Norm}}$$

The fluid velocity at the near-wall nodes is forced to be:

$$U_{\mathrm{NW}}{}^{\mathrm{F}} = (U_{\mathrm{IB}}{}^{\mathrm{IMS}})_{\mathrm{Norm}} + (U_{\mathrm{NW}}{}^{\mathrm{F}})_{\mathrm{Tang}} \quad \textbf{(1–132)}$$

In this way, the velocity field is forced to flow around the immersed boundary to avoid (or alleviate) the problem of streamlines penetrating the immersed solid.

1.4.3. Turbulent Flow Treatment

For moving immersed solids or high Reynolds number turbulence cases, it is usually not possible to generate a fine mesh that resolves the laminar sub-layer. Hence, wall functions are used for fluid elements near the immersed boundary. The implementation of wall functions for immersed solids is more complicated than for wall functions near regular walls because of the non-uniform wall distance for elements next to the immersed boundary.

1.4.3.1. Wall Distance

For fluid nodes that are inside the immersed solid, the wall distance is set to zero. For fluid nodes near the immersed solid, the wall distance is computed as a function of the wall scales. The wall scales are set algebraically to zero for fluid nodes inside the immersed solid, but the shape of the immersed boundary is not properly taken into account for the wall scale equation.

For the near-immersed-boundary nodes, the physical distance from the immersed boundary tracking method is used to implement the turbulent wall functions.

1.4.3.2. SST Model

The wall distance from the modified wall scale equation is used to compute the two blending factors. The production and dissipation terms inside the immersed solid are reset to zero for both the turbulent kinetic energy equation and turbulent frequency equation. This way, the turbulent kinetic energy and turbulent frequency will be solved without any source terms and solutions will serve as a natural condition for the fluid turbulence equations.

In the SST turbulence model, the turbulent eddy viscosity is computed from the turbulent kinetic energy and turbulent frequency:

$$\mu_{\mathrm{T}} = \rho k / \omega \tag{1–133}$$

1.4.3.3. Scalable Wall Function

In general, it is impractical (if not impossible) to have a mesh which resolves the sublayer, and therefore, it is necessary to specify ω_{wall} taking account of the logarithmic region.

$$\omega_{\log} = \frac{u^{*2}}{C_{\mu}^{1/2} \kappa v y^{*}} \tag{1–134}$$

$$u^{*} = C_{\mu}^{1/4} k^{1/2} \tag{1–135}$$

$$y^{*} = u^{*} \Delta y / v \tag{1–136}$$

$$u_{\tau}^{\log} = \frac{\Delta U}{1/\kappa \ln(y^{*}) + C} = \frac{\Delta U}{u^{+}} \tag{1–137}$$

$$\tau_{\mathrm{wall}} = \rho u^{*} u_{\tau} \tag{1–138}$$

Here, ΔU is the fluid tangential velocity relative to the immersed boundary, and Δy is computed as the distance from the near-immersed-solid nodes to the immersed boundary. In the scalable wall function, the lower value of y^{*} is limited to 11.06 at the intersection between the logarithmic and the linear near wall profile.

Further, the velocity at the NW nodes is forced according to the wall function:

$$\left(U_{\mathrm{NW}}{}^{\mathrm{F}}\right)_{\mathrm{Tang}} = \left(U_{\mathrm{IB}}{}^{\mathrm{IMS}}\right)_{\mathrm{Tang}} + \frac{\left(u^{*}/u^{+}\right)_{\mathrm{FL}}}{\left(u^{*}/u^{+}\right)_{\mathrm{NW}}}\left[\left(U_{\mathrm{FL}}\right)_{\mathrm{Tang}} - \left(U_{\mathrm{IB}}{}^{\mathrm{IMS}}\right)_{\mathrm{Tang}}\right] \quad \textbf{(1–139)}$$

Similar to the case of laminar flow, the tangential velocity on the NW nodes is enforced according to equation *Equation 1–139* (p. 44) and the momentum source term is computed taking into account both the wall normal velocity and tangential forcing velocity:

$$U_{\mathrm{NW}}{}^{\mathrm{F}} = \left(U_{\mathrm{IB}}{}^{\mathrm{IMS}}\right)_{\mathrm{Norm}} + \left(U_{\mathrm{NW}}{}^{\mathrm{F}}\right)_{\mathrm{Tang}} \quad \textbf{(1–140)}$$

In this way, the velocity field will be forced to flow around the immersed boundary to avoid (or alleviate) the problem of streamlines penetrating the immersed solid.

1.5. Multicomponent Flow

The following topics will be discussed:

1.5.1. Multicomponent Notation

Components are denoted using capital letters A, B, C, etc. In general, a quantity subscribed with A, B, C, and so on, refers to the value of the quantity corresponding to A, B, C, etc. For example, the density (ρ) of component B would be written ρ_{B}.

1.5.2. Scalar Transport Equation

For a multicomponent fluid, scalar transport equations are solved for velocity, pressure, temperature and other quantities of the fluid. For details, see *Governing Equations* (p. 25). However, additional equations must be solved to determine how the components of the fluid are transported within the fluid.

The bulk motion of the fluid is modeled using single velocity, pressure, temperature and turbulence fields. The influence of the multiple components is felt only through property variation by virtue of differing properties for the various components. Each component has its' own equation for conservation of mass. After Reynolds-averaging (see *Turbulence Models* (p. 89)) this equation can be expressed in tensor notation as:

$$\frac{\partial\ \tilde{\rho}_i}{\partial\ t} + \frac{\partial\left(\tilde{\rho}_i \tilde{\boldsymbol{U}}_j\right)}{\partial\ x_j} = -\frac{\partial}{\partial\ x_j}\left(\rho_i\left(\tilde{\boldsymbol{U}}_{ij} - \tilde{\boldsymbol{U}}_j\right) - \overline{\rho_i'' \boldsymbol{U}_j''}\right) + S_i \tag{1–141}$$

where:

$\tilde{\rho}_i$ is the mass-average density of fluid component i in the mixture, that is, the mass of the component per unit volume,

$\tilde{\boldsymbol{U}}_j = \sum\left(\tilde{\rho}_i \tilde{\boldsymbol{U}}_{ij}\right)/\overline{\rho}$ is the mass-average velocity field,

$\tilde{\boldsymbol{U}}_{ij}$ is the mass-average velocity of fluid component i,

$\rho_i\left(\tilde{\boldsymbol{U}}_{ij} - \tilde{\boldsymbol{U}}_j\right)$ is the relative mass flux,

S_i is the source term for component i which includes the effects of chemical reactions.

Note that if all the terms in the expanded form of *Equation 1–141* (p. 45) are summed over all components, the result is the standard continuity equation,

$$\frac{\partial\ \overline{\rho}}{\partial\ t} + \frac{\partial\left(\overline{\rho}\ \tilde{\boldsymbol{U}}_j\right)}{\partial\ x_j} = 0 \tag{1–142}$$

because the reaction rates S_i must sum to zero.

The relative mass flux term accounts for differential motion of the individual components. This term may be modeled in a number of ways to include effects of concentration gradients, a pressure gradient, external forces or a temperature gradient. Of these possible sources of relative motion among the mixture components, the primary effect is that of concentration gradient. The model for this effect gives rise to a diffusion-like term in *Equation 1–141* (p. 45).

$$\rho_i\left(\tilde{\boldsymbol{U}}_{ij} - \tilde{\boldsymbol{U}}_j\right) = -\frac{\Gamma_i}{\overline{\rho}}\frac{\partial\ \tilde{\rho}_i}{\partial\ x_j} \tag{1–143}$$

The molecular diffusion coefficient, Γ_i, is assumed to be equal to $\rho\, D_i$, where D_i is the **Kinematic Diffusivity** set on the **Fluid Models** tab for a domain in CFX-Pre. For details, see Fluid Models Tab in the *CFX-Pre User's Guide*. A detailed description of the effects of the relative mass flux term and various models for it may be found in reference [29].

Now, define the mass fraction of component i to be:

$$\tilde{Y}_i = \frac{\tilde{\rho}_i}{\bar{\rho}} \tag{1–144}$$

Note that, by definition, the sum of component mass fractions over all components is 1. Substituting *Equation 1–144* (p. 46) and *Equation 1–143* (p. 45) into *Equation 1–141* (p. 45), you have:

$$\frac{\partial\,(\bar{\rho}\,\tilde{Y}_i)}{\partial\,t} + \frac{\partial\left(\bar{\rho}\,\tilde{\boldsymbol{U}}_j\,\tilde{Y}_i\right)}{\partial\,x_j} = \frac{\partial}{\partial\,x_j}\left(\Gamma_i\,\frac{\partial\,\tilde{Y}_i}{\partial\,x_j}\right) - \frac{\partial}{\partial\,x_j}\left(\overline{\rho\,Y_i''\,\boldsymbol{U}_j''}\right) + S_i \tag{1–145}$$

The turbulent scalar fluxes are modeled using the eddy dissipation assumption as:

$$-\overline{\rho\,Y_i''\,\boldsymbol{U}_j''} = \frac{\mu_{\mathrm{t}}}{Sc_{\mathrm{t}}}\,\frac{\partial\,\tilde{Y}_i}{\partial\,x_j} \tag{1–146}$$

where Sc_{t} is the turbulent Schmidt number. Substituting *Equation 1–146* (p. 46) into *Equation 1–145* (p. 46) and assuming now that you have mass weighted averages of Y_i:

$$\frac{\partial\,(\bar{\rho}\,\tilde{Y}_i)}{\partial\,t} + \frac{\partial\left(\bar{\rho}\,\tilde{\boldsymbol{U}}_j\,\tilde{Y}_i\right)}{\partial\,x_j} = \frac{\partial}{\partial\,x_j}\left(\Gamma_{i_{\mathrm{eff}}}\,\frac{\partial\,Y_i}{\partial\,x_j}\right) + S_i \tag{1–147}$$

where:

$$\Gamma_{i_{\mathrm{eff}}} = \Gamma_i + \frac{\mu_{\mathrm{t}}}{Sc_{\mathrm{t}}} \tag{1–148}$$

Equation 1–147 (p. 46) is simply a general advection-diffusion equation of the form common to the equations solved for each of the other dependent variables in the fluid flow calculation. Thus, it is convenient to solve for the Y_i in order to establish the composition of the fluid mixture.

1.5.3. Algebraic Equation for Components

The specified equation is used to calculate the component mass fraction throughout the solution domain.

1.5.4. Constraint Equation for Components

The CFX-Solver solves $N_{\mathrm{C}} - 1$ mass fraction equations (either transport equations or algebraic equations) for all but one of the components. The remaining component is known as the constraint component because its mass fraction is determined by the constraint equation:

$$\sum_{i=A,B,C,\ldots}^{N_C} Y_i = 1 \tag{1–149}$$

The performance of the CFX-Solver will not be affected by your choice of constraint component.

1.5.5. Multicomponent Fluid Properties

The physical properties of general multicomponent mixtures are difficult to specify. The default treatment in ANSYS CFX 14.0 makes the assumption that the components form an ideal mixture. For details, see Ideal Mixture in the *CFX-Solver Modeling Guide*.

Now consider a given volume V of the fluid mixture. Let M_i be the mass of component i present in this volume, then $\rho_i = M_i/V$. The *partial volume* of component i is defined to be the volume, V_i, that would be occupied by the given mass of the component at the same (local) temperature and pressure as the mixture. The "thermodynamic density" of the component, which results from evaluating its equation of state at the mixture temperature and pressure, may be expressed as $\langle \rho_i \rangle = M_i/V_i$. Because the partial volumes of all components must sum to the total volume, V, you have:

$$\begin{aligned} 1 &= \sum_{i=A,B\ldots}^{N_C} \frac{V_i}{V} = \sum_{i=A,B\ldots}^{N_C} \frac{M_i/\langle \rho_i \rangle}{M_i/\rho_i} \\ &= \sum_{i=A,B\ldots}^{N_C} \frac{\rho_i}{\langle \rho_i \rangle} = \sum_{i=A,B\ldots}^{N_C} \frac{\rho\, Y_i}{\langle \rho_i \rangle} \end{aligned} \tag{1–150}$$

or:

$$\frac{1}{\rho} = \sum_{i=A,B\ldots}^{N_C} \frac{Y_i}{\langle \rho_i \rangle} \tag{1–151}$$

Thus, the mixture density may be calculated from the mass fractions Y_i and the thermodynamic density of each component, which may require knowledge of the mixture temperature and pressure, as well as an appropriate equation of state for each component.

Note carefully the distinction between ρ_i and $\langle \rho_i \rangle$. The component mass density, ρ_i, is a quantity relating to the composition of the mixture, while the thermodynamic density, $\langle \rho_i \rangle$, is a material property of the component.

An arbitrary constitutive fluid property may be calculated from:

$$\alpha = \sum_{i = A, B \ \ldots}^{N_C} Y_i \, \alpha_i \tag{1–152}$$

where α_i is the property value for fluid component i. While it may appear anomalous at first sight that density does not conform to this expression, the *specific volume* (volume per unit mass, that is, $1/\left\langle \rho_i \right\rangle$) does indeed conform, as can be seen by considering *Equation 1–152* (p. 48). Properties that may be evaluated for a multicomponent mixture using *Equation 1–152* (p. 48) include the laminar viscosity μ, the specific heat at constant volume c_v, the specific heat at constant pressure c_p, and the laminar thermal conductivity λ.

1.5.6. Energy Equation

Recall that *Equation 2–5* (p. 91) is the Reynolds-averaged conservation equation for energy of a single component fluid. Extending this equation for multicomponent fluids involves adding an additional diffusion term to the energy equation:

$$\frac{\partial}{\partial x_j} \left[\sum_{i}^{N_C} \Gamma_i \, h_i \frac{\partial \ Y_i}{\partial \ x_j} \right] \tag{1–153}$$

For turbulent flow, this term is Reynolds-averaged (see *Turbulence Models* (p. 89)) giving:

$$\frac{\partial}{\partial x_j} \left[\overline{\sum_{i}^{N_C} \left(\tilde{\Gamma}_i + \Gamma_i'' \right) \left(\tilde{h}_i + h_i'' \right) \frac{\partial \left(\tilde{Y}_i + Y_i'' \right)}{\partial x_j}} \right] \tag{1–154}$$

This expression introduces several terms involving the fluctuations of diffusion coefficient, component enthalpy and species concentration. Under certain circumstances, the fluctuating components could be an important component of the diffusion process. However, adequate models are not available within the existing turbulence model to account for these effects. Thus, only the mean component is retained in the current version of ANSYS CFX.

The implemented conservation of energy equation for multicomponent fluids involves only mean scalar components and is expressed as:

$$\frac{\partial}{\partial t}(\rho H) - \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho \boldsymbol{U}_j H\right) = \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j} + \sum_{i}^{N_C} \Gamma_i h_i \frac{\partial Y_i}{\partial x_j} + \frac{\mu_t}{Pr_t}\frac{\partial h}{\partial x_j}\right) + S_E \tag{1–155}$$

1.5.7. Multicomponent Energy Diffusion

The energy equation can be simplified in the special case that all species diffusivities are the same and equal to thermal conductivity divided by specific heat capacity,

$$\Gamma_i = \Gamma = \frac{\lambda}{c_p} \tag{1–156}$$

Equation 1–156 (p. 49) holds when the Lewis number is unity for all components: $Le_i = \lambda / \left(c_p \Gamma_i\right) = 1$. For turbulent flow, assuming $Le_i = 1$ for all components is usually just as good as the common practice of using the fluid viscosity for the default component diffusivity (unity Schmidt number, $Sc_i = \mu / \Gamma_i = 1$). For $Le_i = 1$, the energy equation (*Equation 1–155* (p. 49)) simplifies exactly to the following:

$$\frac{\partial}{\partial t}(\rho H) - \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho \boldsymbol{U}_j H\right) = \frac{\partial}{\partial x_j}\left[\left(\frac{\lambda}{c_p} + \frac{\mu_t}{\mathrm{Pr}_t}\right)\frac{\partial h}{\partial x_j}\right] + S_E \tag{1–157}$$

Equation 1–157 (p. 49) has the advantage that only a single diffusion term needs to be assembled, rather than one for each component plus one for heat conduction. This can significantly reduce numerical cost, in particular when the fluid consists of a large number of components.

When component-dependent turbulent Schmidt numbers are specified, the turbulent energy flux needs to be generalized. This is achieved by splitting the turbulent fluctuation of enthalpy into the contributions from temperature fluctuation, pressure fluctuation and fluctuations of component mass fractions:

$$h'' \approx C_p T'' + \frac{\partial h}{\partial P} P'' + \sum_{i=1}^{N_C} h_i Y''_i \tag{1–158}$$

Using this transformation, the turbulent energy flux can be modeled by applying the eddy diffusivity model with turbulent Prandtl number to the temperature fluctuations plus the sum of the secondary enthalpy transport terms derived from the component mass fluxes:

$$-\overline{\rho u''_j h''} \approx -\overline{\rho u''_j \left(C_p T'' + \frac{\partial h}{\partial p} P'' + \sum_{i=1}^{N_C} h_i Y''_i \right)} \tag{1–159}$$

$$-\overline{\rho u''_j h''} \approx -\overline{C_p \rho u''_j T''} - \overline{\frac{\partial h}{\partial p} u''_j P''} - \sum_{i=1}^{N_C} \overline{h_i \rho u''_j Y''_i} \tag{1–160}$$

$$-\overline{\rho u''_j h''} \approx C_p \frac{\mu_t}{Pr_t} \cdot \frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P} \cdot \frac{\mu_t}{Pr_t} \right) \frac{\partial P}{\partial x_j} - \sum_{i=1}^{N_C} \left(h_i \overline{\rho u''_j Y''_i} \right) \tag{1–161}$$

Inserting the eddy diffusivity model for the turbulent component mass fluxes yields the following model for the turbulent enthalpy flux:

$$-\overline{\rho u''_j h''} \approx C_p \frac{\mu_t}{Pr_t} \cdot \frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P} \cdot \frac{\mu_t}{Pr_t} \right) \frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} h_i \left(\frac{\mu_t}{Sc_{t,i}} \cdot \frac{\partial Y_i}{\partial x_j} \right) \tag{1–162}$$

The previous relation is equivalent to the eddy diffusivity model applied to enthalpy and generalized for different Pr_t and $Sc_{t,i}$, as can be derived by expanding the enthalpy gradient term according to the chain rule $h = h(T, P, Y_i)$:

$$-\overline{\rho u''_j h''} \approx \frac{\mu_t}{Pr_t} \cdot \frac{\partial h}{\partial x_j} = \frac{\mu_t}{Pr_t} \left(\frac{\partial h}{\partial T} \cdot \frac{\partial T}{\partial x_j} + \frac{\partial h}{\partial P} \cdot \frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} \frac{\partial h}{\partial Y_i} \cdot \frac{\partial Y_i}{\partial x_j} \right) \tag{1–163}$$

$$-\overline{\rho u''_j h''} \approx \frac{\mu_t}{Pr_t} \left(C_p \frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P} \right) \frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} h_i \frac{\partial Y_i}{\partial x_j} \right) \tag{1–164}$$

$$-\overline{\rho u''_j h''} \approx C_p \frac{\mu_t}{Pr_t} \cdot \frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P} \cdot \frac{\mu_t}{Pr_t} \right) \frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} h_i \left(\frac{\mu_t}{Pr_t} \cdot \frac{\partial Y_i}{\partial x_j} \right) \tag{1–165}$$

Using the above relations, the energy equation for generalized turbulent component transport becomes:

$$\frac{\partial}{\partial t}(\rho H) - \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_J}\left(\rho U_j H\right) = \frac{\partial}{\partial x_j}\left[\left(\lambda + C_p\frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P}\cdot\frac{\mu_t}{Pr_t}\right)\frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} h_i\left(\Gamma_i\frac{\partial Y_i}{\partial x_j} - \overline{\rho u''_j Y''_i}\right)\right] + S_E \tag{1–166}$$

And for the special case that the eddy diffusivity assumption is used for modelling the turbulent mass flux for each component:

$$\frac{\partial}{\partial t}(\rho H) - \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_J}\left(\rho U_j H\right) = \frac{\partial}{\partial x_j}\left[\left(\lambda + C_p\frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j} + \left(\frac{\partial h}{\partial P}\cdot\frac{\mu_t}{Pr_t}\right)\frac{\partial P}{\partial x_j} + \sum_{i=1}^{N_C} h_i\left(\Gamma_i + \frac{\mu_t}{Sc_{t,i}}\right)\frac{\partial Y_i}{\partial x_j}\right] + S_E \tag{1–167}$$

For the **Thermal Energy** heat transfer model the pressure gradients are assumed to be small compared to the other terms, and the corresponding terms are neglected from the equation.

1.6. Additional Variables

There are several types of Additional Variable equations supported by CFX-Solver:

1.6.1. Transport Equations for Additional Variables

The general form of the transport equation for an Additional Variable is:

$$\frac{\partial\,(\rho\,\varphi)}{\partial t} + \nabla\cdot(\rho\,\boldsymbol{U}\,\varphi) = \nabla\cdot(\rho\,D_{\Phi}\,\nabla\varphi) + S_{\varphi} \tag{1–168}$$

where:

- $\boldsymbol{U}$ is the fluid velocity in the case of a fluid or porous domain, or $\boldsymbol{U} \equiv \boldsymbol{U}_s$ in the case of a solid domain with a specified solid motion velocity, where $\boldsymbol{U}_s$ is the solid velocity.
- ρ is the mixture density, mass per unit volume
- Φ is the conserved quantity per unit volume, or *concentration*
- $\varphi = \Phi/\rho$ is the conserved quantity per unit mass

- S_φ is a volumetric source term, with units of conserved quantity per unit volume per unit time
- D_Φ is the kinematic diffusivity for the scalar

For turbulent flow, this equation is Reynolds-averaged (see *Turbulence Models* (p. 89)) and becomes:

$$\frac{\partial\ (\rho\,\varphi)}{\partial\,t} + \nabla\cdot(\rho\,\boldsymbol{U}\,\varphi) = \nabla\cdot\left(\left(\rho\,D_\Phi + \frac{\mu_{\mathrm{t}}}{Sc_{\mathrm{t}}}\right)\nabla\varphi\right) + S_\varphi \tag{1–169}$$

where:

- Sc_{t} is the turbulence Schmidt number
- μ_{t} is the turbulence viscosity

1.6.2. Diffusive Transport Equations for Additional Variables

The general form of the diffusive transport equation for an Additional Variable (non-reacting scalar) is:

$$\frac{\partial\ (\rho\,\varphi)}{\partial\,t} = \nabla\cdot(\rho\,D_\Phi\,\nabla\varphi) + S_\varphi \tag{1–170}$$

where:

- ρ is the mixture density, mass per unit volume.
- Φ is the conserved quantity per unit volume, or *concentration*.
- $\varphi = \Phi/\rho$ is the conserved quantity per unit mass.
- S_φ is a volumetric source term, with units of conserved quantity per unit volume per unit time
- D_Φ is the kinematic diffusivity for the scalar.

1.6.3. Poisson Equations for Additional Variables

The general form of the Poisson equation for an Additional Variable (non-reacting scalar) is:

$$\nabla\cdot(\rho\,D_\Phi\,\nabla\varphi) + S_\varphi = 0 \tag{1–171}$$

where:

- ρ is the mixture density, mass per unit volume.
- Φ is the conserved quantity per unit volume, or *concentration*.
- $\varphi = \Phi/\rho$ is the conserved quantity per unit mass.
- S_φ is a volumetric source term, with units of conserved quantity per unit volume per unit time
- D_Φ is the kinematic diffusivity for the scalar.

1.6.4. Algebraic Equations for Additional Variables

Additional Variables may be set up as either algebraic equations or transport equations. For algebraic Additional Variables, you must provide an expression for its value throughout the domain. An algebraic Additional Variable definition can be as simple as a copy of an existing variable.

Important

Results obtained using an Additional Variable that references an existing variable may differ from results obtained using the referenced variable directly. This is because the calculation recipes used for such Additional Variables may differ from those used for the variable referenced. An example of this occurs when the referenced variable is a wall or boundary only variable. In particular, a variable-specific recipe may exist to gather integration point values to nodes and this recipe may differ from the generic recipe applied for Additional Variables.

1.7. Rotational Forces

For flows in a rotating frame of reference, rotating at a constant angular velocity ω, additional sources of momentum are required to account for the effects of the Coriolis force and the centrifugal force:

$$\boldsymbol{S}_{M,\mathrm{rot}} = \boldsymbol{S}_{\mathrm{Cor}} + \boldsymbol{S}_{\mathrm{cfg}} \tag{1–172}$$

where:

$$\boldsymbol{S}_{\mathrm{Cor}} = -2\,\rho\,\omega \times \boldsymbol{U} \tag{1–173}$$

$$\boldsymbol{S}_{\mathrm{cfg}} = -\rho\,\omega \times (\omega \times \boldsymbol{r}) \tag{1–174}$$

and where $\boldsymbol{r}$ is the location vector and $\boldsymbol{U}$ is the relative frame velocity (that is, the rotating frame velocity for a rotating frame of reference).

In the energy equation, the advection and transient terms use the rothalpy, I, defined by *Equation 1–175* (p. 53), in place of the total enthalpy.

$$I = h_{\mathrm{stat}} + \frac{1}{2} U^2 - \frac{1}{2} \omega^2 R^2 \tag{1–175}$$

1.7.1. Alternate Rotation Model

By default, the advection term in the momentum equation models the relative frame velocity:

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \left(\rho \, \boldsymbol{U} \otimes \boldsymbol{U} \right) = \nabla \cdot \left(-p \, \delta + \mu \left(\nabla \boldsymbol{U} + \left(\nabla \boldsymbol{U} \right)^{\mathrm{T}} \right) \right) - 2 \, \rho \, \omega \times \boldsymbol{U} - \rho \, \omega \times \left(\omega \times \boldsymbol{r} \right) \tag{1–176}$$

The alternate rotation model modifies the advection and transient terms so they involve the absolute frame velocity, $\boldsymbol{U}_{\mathrm{abs}}$, rather than the relative frame velocity, $\boldsymbol{U}$.

The change of variable in the advection term requires a modification of the original Coriolis source term. The final form of the momentum equation becomes:

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \left(\rho \, \boldsymbol{U} \otimes \boldsymbol{U}_{\mathrm{abs}} \right) = \nabla \cdot \left(-p \, \delta + \mu \left(\nabla \boldsymbol{U} + \left(\nabla \boldsymbol{U} \right)^{\mathrm{T}} \right) \right) - \rho \, \omega \times \boldsymbol{U} - \rho \, \omega \times \left(\omega \times \boldsymbol{r} \right) \tag{1–177}$$

Modeling details and recommendations on when this should be used are available in Alternate Rotation Model in the *CFX-Solver Modeling Guide*.

1.8. Sources

Additional source terms can be applied either to a volume defined by a subdomain, or to a point within a domain. A point source is actually implemented as a volumetric source within a single domain element whose center is nearest to the specified point.

This section describes:

For additional details, see Sources in the *CFX-Solver Modeling Guide*.

1.8.1. Momentum Sources

A momentum source is implemented in ANSYS CFX as a force per unit volume acting on the fluid(s). Momentum sources can be used to model isotropic losses in porous regions, directional losses in porous regions, or other processes. These situations are described further in the following sections. More information on how to use the momentum source models is available in Momentum Sources in the *CFX-Solver Modeling Guide*.

1.8.1.1. Isotropic Loss Model

The momentum loss through an isotropic porous region can be formulated using permeability and loss coefficients as follows:

$$S_{M,x} = -\frac{\mu}{K_{\text{perm}}} U_x - K_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| U_x$$
$$S_{M,y} = -\frac{\mu}{K_{\text{perm}}} U_y - K_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| U_y \quad \textbf{(1–178)}$$
$$S_{M,z} = -\frac{\mu}{K_{\text{perm}}} U_z - K_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| U_z$$

where K_{perm} is the permeability and K_{loss} is the quadratic loss coefficient. The linear component of this source represents viscous losses and the quadratic term represents inertial losses.

Note

K_{loss} may be thought of as a 'pressure gradient per dynamic head'. An engineering handbook may provide data in the form of a 'pressure drop per unit head'. To calculate K_{loss}, divide by the distance over which the pressure drop occurs (which could be the thickness of a perforated plate, for example).

The source may alternatively be formulated using linear and quadratic resistance coefficients, substituting two coefficients $C_{R\,1}$ and $C_{R\,2}$ as follows:

$$C_{R\,1} = \frac{\mu}{K_{\text{perm}}} \quad \textbf{(1–179)}$$

$$C_{R\,2} = K_{\text{loss}} \frac{\rho}{2} \quad \textbf{(1–180)}$$

1.8.1.2. Directional Loss Model

The momentum source through an anisotropic porous region (such as a honeycomb or perforated plate) may be modeled using the directional loss model. With this model, the streamwise direction (which is permitted to vary in space), must be specified. Consider a streamwise-oriented coordinate system (x', y', z') such that the x'-axis is aligned with the streamwise direction and the y', z' axes lie on the transverse plane. The momentum losses in these directions are:

$$\begin{aligned} \boldsymbol{S}_{M,x'} &= -\frac{\mu}{K^{S}_{\text{perm}}} U_{x'} - K^{S}_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| \, U_{x'} \\ \boldsymbol{S}_{M,y'} &= -\frac{\mu}{K^{T}_{\text{perm}}} U_{y'} - K^{T}_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| \, U_{y'} \\ \boldsymbol{S}_{M,z'} &= -\frac{\mu}{K^{T}_{\text{perm}}} U_{z'} - K^{T}_{\text{loss}} \frac{\rho}{2} |\boldsymbol{U}| \, U_{z'} \end{aligned} \quad \textbf{(1–181)}$$

where $K^{\mathrm{S}}_{\text{perm}}$ and $K^{\mathrm{T}}_{\text{perm}}$ are the streamwise and transverse permeabilities, and $K^{\mathrm{S}}_{\text{loss}}$ and $K^{\mathrm{T}}_{\text{loss}}$ are the streamwise and transverse quadratic loss coefficients. These quantities may also be expressed in terms of linear and quadratic resistance coefficients.

In addition, the transverse loss may be modeled by multiplying the streamwise coefficients by some factor. If this option is used together with a streamwise permeability, the implied transverse permeability is equal to the streamwise permeability *divided* by this factor.

Note

Each of $K^{\mathrm{S}}_{\text{loss}}$ and $K^{\mathrm{T}}_{\text{loss}}$ may be thought of as a 'pressure gradient per dynamic head'. An engineering handbook may provide data in the form of a 'pressure drop per unit head'. To calculate $K^{\mathrm{S}}_{\text{loss}}$ or $K^{\mathrm{T}}_{\text{loss}}$, divide by the distance over which the pressure drop occurs (which could be the thickness of a perforated plate, for example).

1.8.1.3. General Momentum Sources

The general momentum source is available for specifying momentum sources that are not covered by the isotropic or directional loss models. A different source can be specified for each direction as follows:

$$\boldsymbol{S}_{M,x} = S_{\text{spec},x} \, \boldsymbol{i} \quad \textbf{(1–182)}$$

$$\boldsymbol{S}_{M,y} = S_{\text{spec},y} \, \boldsymbol{j} \quad \textbf{(1–183)}$$

$$\boldsymbol{S}_{M,z} = S_{\text{spec},z} \, \boldsymbol{k} \quad \textbf{(1–184)}$$

where the $S_{\text{spec},x}$ quantities are the specified momentum components. In ANSYS CFX 14.0, a representative scalar linearization coefficient based on the derivative:

$$\frac{\partial \ \overrightarrow{S_m}}{\partial \ \overrightarrow{U}} \quad \textbf{(1–185)}$$

may be specified to provide robust convergence when a general momentum source is present.

1.8.1.4. Immersed Solid Sources

The presence of an immersed solid in the flow field is modeled through a body force similar to the general momentum source. The solver automatically updates the mesh positions of the immersed solid at the beginning of each time step, and then sets up a list of fluid nodes that lie inside the given immersed solid. The solver applies the immersed solid sources to the fluid nodes which lie inside the immersed solid in order to drive the fluid velocity towards the immersed solid velocity. You can apply a scaling factor to the solver-calculated source coefficient to control how strongly the fluid velocity is forced to match the immersed solid velocity; this affects accuracy and robustness; for details, see Immersed Solid Control in the *CFX-Pre User's Guide*.

1.8.2. General Sources

Any scalar equation may have a source term S_φ. This covers the energy, Additional Variable, turbulence, and mass fraction equations.

A linearization coefficient

$$\frac{\partial S_\varphi}{\partial \varphi} \tag{1–186}$$

may also be specified to provide robust convergence when sources are present. For the energy equation, the derivative is actually taken with respect to temperature.

For details, see General Sources in the *CFX-Solver Modeling Guide*.

1.8.3. Mass (Continuity) Sources

The mass source contribution, S_{MS}, to the conservation equation for the fluid mass is specified exactly as for a general source.

The mass source contributes an additional term to all other transported variables φ:

$$S_{MS\varphi} = \max(S_{MS}, 0)\ \varphi_{MS} + \min(S_{MS}, 0)\ \varphi \tag{1–187}$$

The exterior φ_{MS} must be specified in the case that S_{MS} is positive, in which case, the source behaves very much like an inlet boundary condition.

For details, see Mass (Continuity) Sources in the *CFX-Solver Modeling Guide*.

1.8.4. Bulk Sources

For details, see *Sources in Multiphase Flow* (p. 216).

1.8.5. Radiation Sources

For details, see *Radiation Theory* (p. 351).

1.8.6. Boundary Sources

For details, see Boundary Sources in the *CFX-Solver Modeling Guide*.

1.9. Boundary Conditions

The following topics will be discussed:

1.9.1. Inlet (Subsonic)

1.9.1.1. Mass and Momentum

1.9.1.1.1. Normal Speed in

The magnitude of the inlet velocity is specified and the direction is taken to be normal to the boundary. The direction constraint requires that the flow direction, D_i, is parallel to the boundary surface normal, which is calculated at each element face on the inlet boundary.

1.9.1.1.2. Cartesian Velocity Components

The boundary velocity components are specified, with a non-zero resultant into the domain.

$$\boldsymbol{U}_{\text{Inlet}} = U_{\text{spec}}\,\boldsymbol{i} + V_{\text{spec}}\,\boldsymbol{j} + W_{\text{spec}}\,\boldsymbol{k} \quad \textbf{(1–188)}$$

1.9.1.1.3. Cylindrical Velocity Components

In this case the velocity boundary condition is specified in a local cylindrical coordinate system. Only the axial direction of the local coordinate system needs to be given and the components of velocity in the r, theta and z directions are automatically transformed by the CFX-Solver into Cartesian velocity components. So, in this case you would specify:

$$\boldsymbol{U}_{\text{Inlet}} = U_{r,\text{spec}}\,r + U_{\theta,\text{spec}}\,\theta + U_{z,\text{spec}}\,z \quad \textbf{(1–189)}$$

and the solver will compute the rotation matrix that transforms these components from the cylindrical components to the Cartesian components such that the boundary condition is the same as if Cartesian components were specified:

$$U_{\text{Inlet}} = U_{\text{spec}}\, \boldsymbol{i} + V_{\text{spec}}\, \boldsymbol{j} + W_{\text{spec}}\, \boldsymbol{k} \tag{1–190}$$

For details, see Cylindrical Velocity Components in the *CFX-Solver Modeling Guide*.

1.9.1.1.4. Total Pressure

The *Total Pressure* (p. 16), p_{tot}, is specified at an inlet boundary condition and the CFX-Solver computes the static pressure needed to properly close the boundary condition. For rotating frames of reference one usually specifies the stationary frame total pressure instead.

The direction constraint for the **Normal To Boundary** option is the same as that for the **Normal Speed In** option. Alternatively, the direction vector can be specified explicitly in terms of its three components. In both cases, the boundary mass flow is an implicit result of the flow simulation.

1.9.1.1.5. Mass Flow Rate

The boundary mass flow rate is specified along with a direction component. If the flow direction is specified as normal to the boundary, a uniform mass influx is assumed to exist over the entire boundary. Also, if the flow direction is set using Cartesian or cylindrical components, the component normal to the boundary condition is ignored and, again, a uniform mass influx is assumed. The mass influx is calculated using:

$$\rho\, U = \frac{\dot{m}}{\int_S dA} \tag{1–191}$$

where

$$\int_S dA \tag{1–192}$$

is the integrated boundary surface area at a given mesh resolution. The area varies with mesh resolution because the resolution determines how well resolved the boundary surfaces are. The value of $\rho\, U$ is held constant over the entire boundary surface.

1.9.1.2. Turbulence

For the k-ε turbulence model and Reynolds stress models, the inlet turbulence quantities, k and ε, are either specified directly or calculated using expressions that scale the distribution at the inlet according to the turbulence intensity, I, where:

$$I = \frac{u}{U} \tag{1–193}$$

The inlet flows of k and ε involve advection and diffusion.

$$Q^{k}_{\text{Inlet}} = Q^{k}_{\text{advect}} + Q^{k}_{\text{diffus}} \tag{1–194}$$

$$Q^{\varepsilon}_{\text{Inlet}} = Q^{\varepsilon}_{\text{advect}} + Q^{\varepsilon}_{\text{diffus}} \tag{1–195}$$

The advection flows are evaluated using the computed inlet values of k and ε:

$$Q^{k}_{\text{advect}} = \dot{m}\, k_{\text{spec}} \tag{1–196}$$

$$Q^{\varepsilon}_{\text{advect}} = \dot{m}\, \varepsilon_{\text{spec}} \tag{1–197}$$

The diffusion flows are assumed to be negligible compared to advection, and are equated to zero.

1.9.1.2.1. Default Intensity and Autocompute Length Scale

When default inlet turbulence intensity is selected, the value is set to:

$$I = \frac{u}{U} = 0.037 \tag{1–198}$$

which is an approximate value for internal pipe flow. The inlet turbulence energy is calculated using:

$$k_{\text{Inlet}} = \frac{3}{2} I^2\, U^2 \tag{1–199}$$

and the turbulence dissipation calculated using:

$$\varepsilon_{\text{Inlet}} = \rho\, C_\mu \frac{k^2}{\mu_{\text{t}}} \tag{1–200}$$

where:

$$\mu_{\text{t}} = 1000\, I\, \mu \tag{1–201}$$

1.9.1.2.2. Intensity and Autocompute Length Scale

The turbulence intensity is specified directly and the distributions of k and ε at the inlet calculated using the same relationships as the `Default Intensity and Autocompute Length Scale` option.

1.9.1.2.3. Intensity and Length Scale

The turbulence intensity and length scale are both specified. The turbulence kinetic energy and dissipation are calculated using:

$$k_{\mathrm{Inlet}} = \frac{3}{2} I^2 U^2 \qquad (1\text{–}202)$$

and

$$\varepsilon_{\mathrm{Inlet}} = \frac{k^{\frac{3}{2}}}{l_{\mathrm{t}}} \qquad (1\text{–}203)$$

1.9.1.2.4. k and Epsilon

Both k and ε are specified directly:

$$k_{\mathrm{Inlet}} = k_{\mathrm{spec}} \qquad (1\text{–}204)$$

and

$$\varepsilon_{\mathrm{Inlet}} = \varepsilon_{\mathrm{spec}} \qquad (1\text{–}205)$$

When the Reynolds stress model is employed, the Inlet boundary conditions are specified with the same turbulence options as those for the k-ε model. Additionally, the stress tensors are extracted using the computed value of k. This is done by assuming the Inlet boundary to be isotropic with respect to the Reynolds stresses, such that the normal stress components are:

$$\boldsymbol{u}_x \boldsymbol{u}_x = \boldsymbol{u}_y \boldsymbol{u}_y = \boldsymbol{u}_z \boldsymbol{u}_z = \frac{2}{3} k \qquad (1\text{–}206)$$

and the shear stress components are equal to zero:

$$\boldsymbol{u}_x \boldsymbol{u}_y = \boldsymbol{u}_x \boldsymbol{u}_z = \boldsymbol{u}_y \boldsymbol{u}_z = 0 \qquad (1\text{–}207)$$

1.9.1.3. Heat Transfer

1.9.1.3.1. Static Temperature

The inlet static temperature is specified:

$$T_{\mathrm{stat,Inlet}} = T_{\mathrm{spec}} \qquad (1\text{–}208)$$

The inlet energy flow involves advection and diffusion.

$$Q_{\mathrm{Inlet}} = Q_{\mathrm{advect}} + Q_{\mathrm{diffus}} \qquad (1\text{–}209)$$

The energy flow by advection is a function of the specific total enthalpy, h_{tot}:

$$Q_{\text{advect}} = \dot{m}\, h_{\text{tot}} \tag{1–210}$$

where h_{tot} is computed from the specific static enthalpy, h_{stat}, and the inlet boundary velocity:

$$h_{\text{tot}} = h_{\text{stat}} + \frac{1}{2} U^2 \tag{1–211}$$

The static enthalpy is computed using the specified value of T_{spec}, the boundary values of U and p, and the thermodynamic relationship for $h = h\,(p, T\,)$ for the given fluid. The evaluation of U depends upon the nature of the mass and momentum specification for the boundary condition.

The Inlet energy flow by diffusion is assumed to be negligible compared to advection, and equated to zero.

1.9.1.3.2. Total Temperature

The boundary advection and diffusion terms for specified total temperature are evaluated in exactly the same way as specified static temperature, except that the static temperature is dynamically computed from the definition of total temperature:

$$T_{\text{tot,Inlet}} = T_{\text{spec}} \tag{1–212}$$

which for a fluid with constant heat capacity is:

$$T_{\text{stat,Inlet}} = T_{\text{tot,Inlet}} - \frac{U^2}{2\, c_p} \tag{1–213}$$

Additional information on the treatment of variable specific heat is available in *Ideal Gas Equation of State* (p. 28).

1.9.1.4. Additional Variables

The value of the Additional Variable is specified explicitly at an inlet:

$$\Phi_{\text{Inlet}} = \Phi_{\text{spec}} \tag{1–214}$$

The inlet flow of Φ involves advection and diffusion:

$$Q^{\Phi}_{\text{Inlet}} = Q^{\Phi}_{\text{advect}} + Q^{\Phi}_{\text{diffus}} \tag{1–215}$$

and the advection quantity is evaluated using the specified value of Φ:

$$Q^{\Phi}_{\mathrm{advect}} = \dot{m}\, \Phi_{\mathrm{spec}} \tag{1–216}$$

The inlet flow by diffusion is assumed to be negligible compared to advection, and set to zero.

1.9.2. Inlet (Supersonic)

1.9.2.1. Heat Transfer

1.9.2.1.1. Static Temperature

The static temperature is specified at the supersonic inlet boundary:

$$T_{\mathrm{stat,Inlet}} = T_{\mathrm{spec}} \tag{1–217}$$

1.9.2.1.2. Total Temperature

Total temperature is specified at the supersonic inlet boundary:

$$T_{\mathrm{tot,Inlet}} = T_{\mathrm{spec}} \tag{1–218}$$

Static temperature is dynamically computed from the definition of total temperature.

1.9.3. Outlet (Subsonic)

1.9.3.1. Mass and Momentum

1.9.3.1.1. Static Pressure (Uniform)

Relative Static Pressure is specified over the outlet boundary:

$$p_{\mathrm{stat,Outlet}} = p_{\mathrm{spec}} \tag{1–219}$$

1.9.3.1.2. Normal Speed

The magnitude of the outlet velocity is specified and the direction is taken to be normal to the boundary at mesh resolution.

1.9.3.1.3. Cartesian Velocity Components

The boundary velocity components are specified, with a non-zero resultant out of the domain.

$$\boldsymbol{U}_{\mathrm{Outlet}} = U_{\mathrm{spec}}\, \boldsymbol{i} + V_{\mathrm{spec}}\, \boldsymbol{j} + W_{\mathrm{spec}}\, \boldsymbol{k} \tag{1–220}$$

1.9.3.1.4. Cylindrical Velocity Components

These are handled the same way as for an *Inlet (Subsonic)* (p. 58) boundary condition.

1.9.3.1.5. Average Static Pressure: Over Whole Outlet

The Outlet Relative Static Pressure allows the pressure profile at the outlet to vary based on upstream influences while constraining the average pressure to a user-specified value $\overline{p}_{spec}$, where:

$$\overline{p}_{\mathrm{spec}} = \frac{1}{A} \int_S p_{ip} \, dA \tag{1–221}$$

where p_{ip} is the imposed pressure at each integration point and the integral is evaluated over the entire outlet boundary surface. To enforce this condition, the pressure at each boundary integration point is set to:

$$p_{ip} = \overline{p}_{\mathrm{spec}} + \left(p_{\mathrm{n}} - \overline{p}_{\mathrm{n}}\right) \tag{1–222}$$

So, the integration point pressure in this case is set to the specified value plus the difference between the local nodal value and the average outlet boundary pressure. In this way the outlet pressure profile can vary, but the average value is constrained to the specified value.

1.9.3.1.6. Average Static Pressure: Above or Below Specified Radius

In this case, the average pressure is only constrained in the region above or below the specified radius by shifting the calculated pressure profile by the difference between the specified average and the nodal average above or below the specified radius.

1.9.3.1.7. Average Static Pressure: Circumferential

The circumferential averaging option divides the exit boundary condition into circumferential bands (oriented radially or axially depending on the geometry). The pressure within each band is constrained to the specified average pressure value the same way as is done for the overall averaging:

$$p_{ip} = \overline{p}_{\mathrm{spec}}\left(\eta_k\right) + \left(p_n - \overline{p}_n\left(\eta_k\right)\right) \tag{1–223}$$

where the specified value is applied within a band and the nodal average pressure is also calculated within a band.

1.9.3.1.8. Mass Flow Rate: Scale Mass Flows

The mass flux distribution across the outlet is determined by starting with the local mass flow rate distribution calculated by the flow solver at each integration point:

$$\dot{m}_{ip} = \rho_{ip} A_{ip} \boldsymbol{U}_{ip} \tag{1–224}$$

From that distribution, you calculate the estimated total mass flow rate through the outlet boundary condition:

$$\dot{m}_{\mathrm{tot}}^{\mathrm{est}} = \sum_{\mathrm{all}} \dot{m}_{ip} \tag{1–225}$$

where the summation is over all boundary integration points. A scaling factor is computed at the end of each coefficient loop that is used to scale the local integration point mass flows such that they add up to the specified mass flow rate:

$$F = \frac{\dot{m}_{\mathrm{spec}}}{\dot{m}_{\mathrm{tot}}^{\mathrm{est}}} \tag{1–226}$$

Iteratively, during the computation, F can be greater than or less than unity. The final integration point mass flows are reset by multiplying the integration point mass flows by the scaling factor:

$$\dot{m}_{ip} = F\, \rho_{ip}\, A_{ip}\, \boldsymbol{U}_{ip} \tag{1–227}$$

In this way, the mass flux profile is an implicit result of the solution and at the same time gives exactly the specified mass flow rate.

1.9.3.1.9. Mass Flow Rate: Shift Pressure with or without Pressure Profile

This condition differs from the last one in that pressure is shifted in the continuity equation to get the specified mass flow rate. Generally speaking, the mass flow rate at each boundary integration point is dependent upon both velocity and pressure:

$$\dot{m}_{ip} = \rho_{ip}\, A_{ip}\, \boldsymbol{U}_{ip} \left(p_{ip}, U_{\mathrm{node}} \right) \tag{1–228}$$

where the integration point velocity depends upon nodal velocity and integration point pressures through the Rhie-Chow coupling. For this boundary condition, the integration point pressures are given by an expression of the form:

$$p_{ip} = F\, p_{\mathrm{prof}} + (1 - F) \left(p_{\mathrm{node}} - \overline{p}_{\mathrm{node}} \right) + p_{\mathrm{shift}} \tag{1–229}$$

where p_{prof} is an optional specified pressure profile, p_{node} is the boundary node pressure, $\overline{p}_{\mathrm{node}}$ is the outlet boundary nodal average pressure, F is the **Pressure Profile Blend** factor that sets how much the specified profile influences the boundary condition, and p_{shift} is the level shift factor automatically computed by the CFX-Solver each coefficient loop to enforce the specified mass flow rate, such that:

$$\dot{m}_{\mathrm{spec}} = \sum_{\substack{ip \\ ip}} \rho\, A_{ip}\, \boldsymbol{U}_{ip} \left(p_{ip}, U_{\mathrm{node}} \right) \tag{1–230}$$

where the sum, in this case, is over all the outlet boundary condition integration points.

1.9.3.1.10. Mass Flow Rate: Shift Pressure with Circumferential Pressure Averaging

A further extension of the shift pressure feature for outlet mass flow rate conditions (or outlet boundaries using an Average Static Pressure specification) enforces the specified profile as an average pressure profile (or average pressure) in circumferential bands (radial or axial), held at a particular value.

Starting with the original formula for the integration point pressure in *Equation 1–229* (p. 65): instead of imposing a particular profile distribution, an average pressure profile within bands is introduced:

$$p_{ip} = F\ \overline{p}_{\text{prof}}\left(\eta_k\right) + \left(p_{\text{node}} - \overline{p}_{\text{node}}\left(\eta_k\right)\right) + p_{\text{shift}} \qquad \textbf{(1–231)}$$

where: $\overline{p}_{\text{prof}}\left(\eta_k\right)$ is the average pressure desired in band η_k, and $\overline{p_{\text{node}}}\left(\eta_k\right)$ is the current average nodal value in band η_k. F corresponds to the **Pressure Profile Blend** factor. When the specified profile spatially varies, the flow solver will compute the average of that profile within each band and then use those values for $\overline{p}_{\text{prof}}\left(\eta_k\right)$.

1.9.3.1.11. Radial Equilibrium

The radial equilibrium option divides the boundary condition into radially oriented circumferential bands. The pressure within each band is constrained by the radial equilibrium condition:

$$\frac{d p_k}{dr} = \overline{\rho}\frac{\overline{u}_\theta^2}{r_k}$$

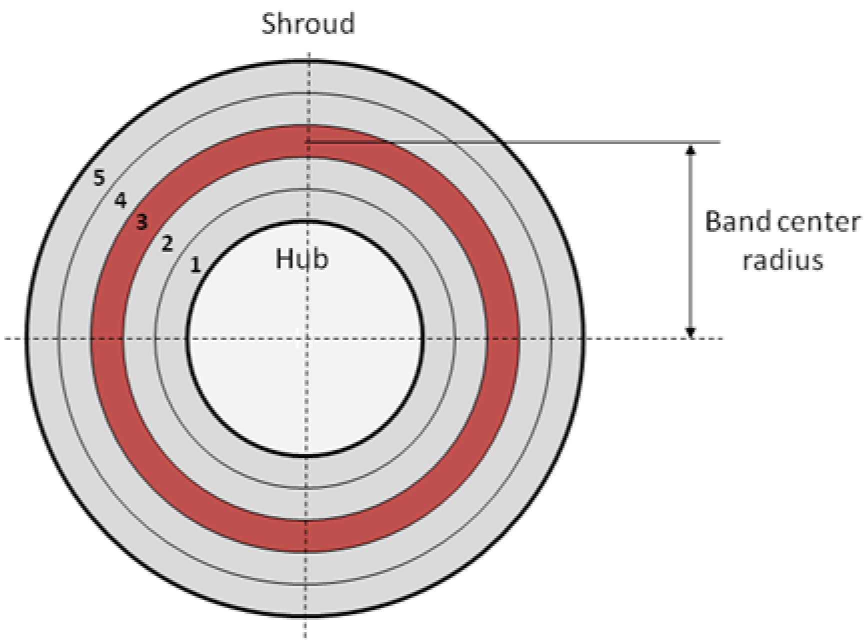
This equation is used to calculate the pressure gradient $d p_k / dr$ in each band, η_k, based on the band averaged tangential velocity, $\overline{u}_\theta$, and density, $\overline{\rho}$, as well as the band centre radius, r_k. The band edge pressures, p_k, are obtained by integrating this equation starting from the specified pressure at the radial reference position. This pressure is used to constrain the solution pressure with the following equation:

$$p_{ip} = F p_k\left(r, \eta_k\right) + (1 - F)\left(p_n + p_k\left(\eta_k\right) - \overline{p}_n\left(\eta_k\right)\right)$$

where:

- $\overline{p}_n(\eta_k)$ is the area average nodal pressure, within a band η_k
- $p_k(r, \eta_k)$ is linearly interpolated between the band edge values to the local face radius, *r*
- and $p_k(\eta_k)$ is the evaluated at the band mid-radius.

As for all average static-pressure options, the pressure profile blend factor, *F*, can be applied to this boundary condition. If the factor *F*=,1 then the pressure at each face is fully constrained to the pressure obtained from the radial equilibrium condition, $p_k(r, \eta_k)$, so that a circumferential profile is not allowed within the band. If the factor *F*=0, then a circumferential profile is allowed to develop with the average of that profile equal to the radial equilibrium pressure.



1.9.3.2. Turbulence, Heat Transfer, Convected Additional Variables, and Other Scalars

For scalar quantities, the CFX-Solver imposes a constant gradient constraint (generally non-zero) at the outlet boundary.

1.9.4. Outlet (Supersonic)

The specification of a supersonic outlet boundary condition requires no accompanying values, except for the radiation intensity if radiation is modeled.

1.9.5. Opening

An opening boundary condition allows the fluid to cross the boundary surface in either direction. For example, all of the fluid might flow into the domain at the opening, or all of the fluid might flow out of the domain, or a mixture of the two might occur. An opening boundary condition might be used where it is known that the fluid flows in both directions across the boundary.

1.9.5.1. Mass and Momentum

1.9.5.1.1. Cartesian Velocity Components

The Cartesian components of the flow velocity are specified at the opening boundary:

$$U_{\text{Opening}} = U_{\text{spec}}\, \boldsymbol{i} + V_{\text{spec}}\, \boldsymbol{j} + W_{\text{spec}}\, \boldsymbol{k} \qquad (1–232)$$

1.9.5.1.2. Cylindrical Velocity Components

These are handled the same way as for an *Inlet (Subsonic)* (p. 58) boundary condition.

1.9.5.1.3. Pressure and Direction

An opening boundary condition can also be specified with a Relative Pressure value:

$$p_{\text{Opening}} = p_{\text{spec}} \tag{1–233}$$

The value is interpreted as relative total pressure for inflow and relative static pressure for outflow.

The direction component, normal to boundary condition or direction components, must also be specified. The magnitude of the velocity at the opening boundary is then part of the solution.

1.9.5.1.3.1. Loss Coefficient

You can optionally specify a loss coefficient, f:

$$f = f_{\text{spec}} \tag{1–234}$$

with a velocity component specification at an opening boundary. The pressure drop is calculated using:

$$\Delta p_{\text{loss}} = \frac{1}{2} f \rho U_n^2 \tag{1–235}$$

where U_n is the magnitude of the velocity component normal to the opening boundary. For inflows, the constraint on pressure and velocity becomes:

$$p_{\text{spec}} - \frac{1}{2} f \rho U_n^2 = p_{\text{stat}} \tag{1–236}$$

and for outflows, the constraint becomes:

$$p_{\text{spec}} + \frac{1}{2} f \rho U_n^2 = p_{\text{stat}} \tag{1–237}$$

The coefficient can be used to model the pressure drop across a screen or other planar resistance.

1.9.5.2. Heat Transfer

1.9.5.2.1. Static Temperature

Opening temperature or Static temperature can be specified at the opening boundary:

$$T_{\text{stat,Opening}} = T_{\text{spec}} \tag{1–238}$$

1.9.5.3. Additional Variables

For flow into the domain, the value of the Additional Variable at an opening boundary is that specified on the **Additional Variable Details** tab:

$$\Phi_{\text{Opening}} = \Phi_{\text{spec}} \tag{1–239}$$

For flow out of the domain the specified value is not used. The value for flow out of the domain is calculated by the CFX-Solver and is the Additional Variable value taken from the solution field.

1.9.6. Wall

1.9.6.1. Mass and Momentum

1.9.6.1.1. No Slip (Not Moving, No Wall Velocity)

The velocity of the fluid at the wall boundary is set to zero, so the boundary condition for the velocity becomes:

$$\boldsymbol{U}_{\text{Wall}} = 0 \tag{1–240}$$

1.9.6.1.2. Free Slip

In this case, the velocity component parallel to the wall has a finite value (which is computed), but the velocity normal to the wall, and the wall shear stress, are both set to zero:

$$\boldsymbol{U}_{\text{n,Wall}} = 0 \tag{1–241}$$

$$\tau_{\text{w}} = 0 \tag{1–242}$$

1.9.6.1.3. Finite Slip

For details on finite slip wall, see Finite Slip Wall in the *CFX-Solver Modeling Guide*

1.9.6.1.4. Specified Shear

For details on specified shear wall, see Specified Shear in the *CFX-Solver Modeling Guide*

1.9.6.1.5. No Slip (Moving, with Wall Velocity)

In this case, the fluid at the wall boundary moves at the same velocity as the wall. There are three different options for the wall velocity:

- **Cartesian Components**: You can directly specify Cartesian components in a local coordinate frame or the global coordinate frame:

$$\boldsymbol{U}_{\text{Wall}} = U_{\text{spec}}\, \boldsymbol{i} + V_{\text{spec}}\, \boldsymbol{j} + W_{\text{spec}}\, \boldsymbol{k} \tag{1–243}$$

- **Cylindrical Components**: You can directly specify cylindrical components in a local cylindrical coordinate system:

$$\boldsymbol{U}_{\mathrm{Wall}} = U_{\mathrm{r,spec}}\, r + U_{\theta,\mathrm{spec}}\, \theta + U_{\mathrm{z,spec}}\, z \tag{1–244}$$

and the solver automatically transforms the cylindrical velocity components into the global Cartesian coordinate system.

- **Counter-rotating Wall** and **Rotating Wall**: A counter-rotating wall can be specified for walls in rotating frames. In this case, the wall is stationary in the absolute frame; in the relative frame, it moves with a velocity:

$$\boldsymbol{U}_{\mathrm{Wall}} = -\omega\, \boldsymbol{R} \tag{1–245}$$

where $\boldsymbol{R}$ is the radial vector from the domain axis of rotation to the wall and ω is the domain angular velocity.

A rotating wall can be specified in both stationary and rotating frames. This option is useful to use in stationary domains when you would like to create a spinning wall. In this case, you enter a local rotation axis for the wall boundary and the wall velocity:

$$\boldsymbol{U}_{\mathrm{Wall}} = \omega\, \boldsymbol{R} \tag{1–246}$$

The solver automatically transforms the specified wall velocity into Cartesian components. This option could be used to duplicate the counter rotating wall option in rotating frames by explicitly setting the angular velocity equal to minus one times the domain angular velocity.

1.9.6.2. Turbulence

The treatment of wall boundary conditions for turbulent flow is the same as for laminar flow, except for No Slip. For details, see *Modeling Flow Near the Wall* (p. 139).

1.9.6.3. Heat Transfer

1.9.6.3.1. Adiabatic

The Adiabatic Wall boundary condition allows no heat transfer across the Wall boundary:

$$q_{\mathrm{w}} = 0 \tag{1–247}$$

1.9.6.3.2. Fixed Temperature

Static Temperature is specified at the wall boundary:

$$T_{\mathrm{stat,w}} = T_{\mathrm{spec}} \tag{1–248}$$

1.9.6.3.3. Heat Flux

Heat flux at the wall boundary is specified:

$$q_w = q_{spec} \quad \textbf{(1–249)}$$

1.9.6.3.4. Heat Transfer Coefficient

Heat flux at the wall boundary is calculated using:

$$q_w = h_c \, (T_b - T_{nw}) \quad \textbf{(1–250)}$$

where h_c is a specified heat transfer coefficient, T_b is the specified boundary temperature, (that is, outside the fluid domain) and T_{nw} is the temperature at the internal near-wall boundary element center node. For details, see Heat Transfer in the *CFX-Solver Modeling Guide*.

1.9.6.4. Additional Variables

The options for specifying Additional Variable quantities at wall boundaries are analogous to those for heat transfer.

1.9.7. Symmetry Plane

The symmetry plane boundary condition imposes constraints that 'mirror' the flow on either side of it.

For example, the normal velocity component at the symmetry plane boundary is set to zero:

$$\boldsymbol{U}_n = 0 \quad \textbf{(1–251)}$$

and the scalar variable gradients normal to the boundary are also set to zero:

$$\frac{\partial \varphi}{\partial \boldsymbol{n}} = 0 \quad \textbf{(1–252)}$$

1.10. Automatic Time Scale Calculation

This section describes the way in which a timestep is calculated when using the `Auto Timescale` or `Auto Timescale` with a `Maximum Timescale` option for setting the timestep used during the calculation of a solution. For details, see Timestep Selection in the *CFX-Solver Modeling Guide*.

1.10.1. Fluid Time Scale Estimate

The following length scales are calculated by ANSYS CFX:

$$L_{\text{vol}} = \sqrt[3]{V}$$
$$L_{\text{ext}} = \max\left(L_x, L_y, L_z\right) \tag{1–253}$$
$$L_{\text{bc}} = \min\sqrt{A_{\text{bc}}}$$

where V is the domain volume (over all domains), L_x, L_y and L_z are the x, y and z extents of the domain (over all domains) and A_{bc} is the area of an 'open' boundary (that is, inlets, outlets or openings).

Further length scales are then calculated depending on the **Length Scale Option** parameter, which can be `Conservative`, `Aggressive` or `Specified Length Scale`:

$$L_{\text{scale}} = \begin{cases} \min\left(L_{\text{vol}}, L_{\text{ext}}\right) & \text{for Conservative} \\ \max\left(L_{\text{vol}}, L_{\text{ext}}\right) & \text{for Aggressive} \\ L_{\text{user}} & \text{for Specified} \end{cases} \tag{1–254}$$

The velocity scales used to calculate a timestep are:

$$U_{\text{bc}} = \max\left|\overline{\boldsymbol{U}}_{\text{bc}}\right|$$
$$U_{\text{node}} = \left|\overline{\boldsymbol{U}}_{\text{node}}\right| \tag{1–255}$$
$$U_{\Delta P} = \sqrt{\frac{p_{\text{bc,max}} - p_{\text{bc,min}}}{\overline{\rho}_{\text{node}}}}$$

where U_{bc} is the arithmetic average of the velocity on a boundary, U_{node} is the arithmetic average of the nodal velocities, $p_{\text{bc,max}}$ and $p_{\text{bc,min}}$ are the maximum and minimum pressure values on an 'open' boundary and $\overline{\rho}_{\text{node}}$ is the arithmetic average nodal density.

For compressible flows, a Mach number for the simulation is calculated as:

$$M = \frac{max\left(U_{\text{bc}}, U_{\text{node}}, U_{\Delta P}\right)}{\overline{c}}$$
$$\overline{c} = \overline{\left(\frac{\partial \rho}{\partial p}\right)}^{-1} \tag{1–256}$$

where $\overline{c}$ is the arithmetic averaged speed of sound over all nodes.

For buoyant flows using the full buoyancy model:

$$g = |\boldsymbol{g}| \tag{1–257}$$

and for the Boussinesq model:

$$g = |\mathbf{g}| \; \overline{\beta} \; (T_{\max} - T_{\min}) \tag{1–258}$$

where β is the thermal expansivity and $T_{\max}$ and $T_{\min}$ are the maximum and minimum domain temperatures (over all domains).

The final fluid time scale used is calculated as:

$$\Delta t = \min\left(\Delta t_U, \Delta t_{\Delta P}, \Delta t_g, \Delta t_{\mathrm{rot}}, \Delta t_{\mathrm{c}} \right) \tag{1–259}$$

where:

$$\Delta t_U = 0.3 \left[\frac{L_{scale}}{max\,(U_{bc}, U_{node}, U_{\Delta P})}, \frac{\rho L^2{}_{scale}}{\mu}, \frac{m}{\dot{m}_{in}} \right]$$

with the dynamic viscosity μ, the total mass m and the total mass flow into the system $\dot{m}_{in}$.

For very small geometries, where the Reynolds number may be very small, including the diffusion scale in the automatic timescale calculation can lead to very small timesteps, which can unnecessarily slow down convergence. For such cases it may be better to use a physical timescale based on advection scales instead.

$$\Delta t_g = \sqrt{\frac{L_{\mathrm{scale}}}{g}} \qquad \Delta t_{\mathrm{rot}} = \frac{0.1}{\omega} \tag{1–260}$$

If $M > 0.3$, then

$$\Delta t_c = \frac{L_{\mathrm{bc}}}{\max\,(U_{\mathrm{bc}}, U_{\mathrm{node}}, U_{\Delta P}, \overline{c}\,)} \tag{1–261}$$

1.10.2. Solid Time Scale Estimate

The solid time scale estimate will, in general, be much larger than the fluid time scale.

The length scale used to calculate a solid time scale, L_{scale}, is calculated in the same manner as for fluid time scales. The default value is taken as the cube root of the volume of the solid domain.

A volume averaged diffusion is calculated as:

$$\overline{\alpha} = \frac{\overline{\lambda}}{\overline{\rho}\;\overline{c}_p} \tag{1–262}$$

where each quantity is volume averaged over the solid. The solid time scale is finally calculated as:

$$\Delta t = f \frac{L_{\text{scale}}^2}{\bar{\alpha}} \tag{1–263}$$

where f is the specified solid timescale factor, which has a default value of 1.

When a rotating motion is enabled in a solid domain, the advective time scale is then computed as $1/\omega$ where ω is the angular velocity. The solid timestep is taken as the minimum between the diffusive and this new advective $(1/\omega)$ timescale.

The value of the computed timestep size is recorded in the `.out` file.

1.11. Mesh Adaption

Mesh adaption in ANSYS CFX is the process in which, once or more during a run, the mesh is selectively refined in areas that depend on the adaption criteria specified. This means that as the solution is calculated, the mesh can automatically be refined in locations where solution variables are changing most rapidly, in order to resolve the features of the flow in these regions.

More information on the mesh adaption process and controls is available in Mesh Adaption in the *CFX-Pre User's Guide*.

Within the mesh adaption step itself, three processes take place:

1. Adaption criteria are calculated for each mesh element.
2. The appropriate number of nodes are added to the existing mesh according to the adaption criteria calculated.
3. The solution already calculated on the older mesh is linearly interpolated onto the new mesh.

This section describes the details of exactly which elements are refined and how the refinement takes place.

1.11.1. Adaption Criteria

This section describes how the adaption criteria are calculated for each mesh edge.

- If the adaption criteria method is **Solution Variation**, then the adaption criteria, A_i, for a given mesh edge i of length l_i is calculated as:

$$A_i = \sum_j \frac{|\Delta\varphi_{ji}|}{N_{\varphi_j}\,|\Delta\varphi_j|} \tag{1–264}$$

  where φ_j is the j^{th} adaption variable (such as density, pressure, and so on), $\Delta\varphi_j$ is the global range of the variable φ_j over all the nodes (excluding those on wall boundary conditions for turbulent flow), $\Delta\varphi_{ij}$ is the difference between φ_j at one end of the edge and the other end, and N_{φ_j} is a scalar for adaption variable j to scale all the A_i to take values between 0 and 1.

- If the adaption criteria method is `Variation * Edge Length`, then the adaption criteria A_i for a given mesh edge i of length l_i is calculated as:

$$A_i = \sum_j \frac{l_i \ |\Delta\varphi_{ji}|}{N_{\varphi_j} \ |\Delta\varphi_j|} \tag{1–265}$$

  where: $\Delta\varphi_j$ is the global range of the j^{th} adaption variable φ_j over all the nodes (excluding those on wall boundary conditions for turbulent flow), $\Delta\varphi_{ji}$ is the difference between φ_j at one end of the edge and the other end, and N_{φ_j} is a length chosen to scale all the A_i to take values between 0 and 1.

- If you select more than one solution variable, then the adaption criteria are calculated for each variable at each edge, and the sum over all adaption variables is used.
- If you specify the adaption criteria to be an expression, then the expression is evaluated at each node, and the calculation of the adaption criteria follows as if this were another solution variable.

The edges that have the largest adaption criteria are marked for refinement, providing that this would not result in mesh edges that were shorter than the **Minimum Edge Length**, if this was specified.

1.11.2. Mesh Refinement Implementation in ANSYS CFX

There are two general methods for performing mesh adaption. **Incremental adaption** takes an existing mesh and modifies it to meet the adaption criteria. The alternative is **remeshing**, in which the whole geometry is remeshed at every adaption step according to the adaption criteria. In ANSYS CFX, incremental adaption is used because this is much faster; however, this imposes the limitation that the resulting mesh quality is limited by the quality of the initial mesh.

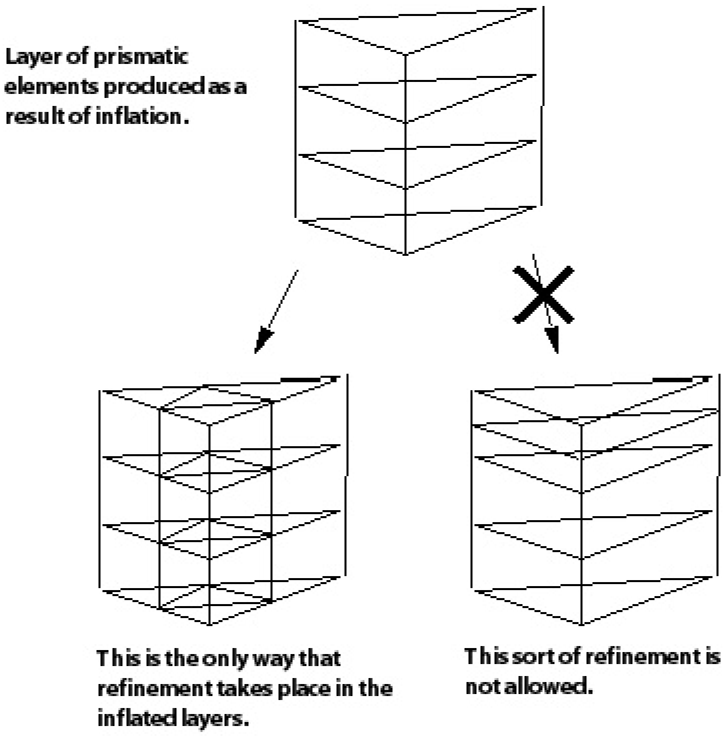
The particular implementation of incremental adaption that is adopted in ANSYS CFX is known as **hierarchical refinement** or **h-refinement**. Each adaption step consists of a structured refinement of an existing mesh. A sequence of refinements form a set of hierarchical levels.

In each mesh adaption step, each mesh edge that is marked for adaption (see the previous section) has an extra node placed half-way along it. The mesh elements that share this edge are then divided to use the new node, subject to the following:

- Neighboring elements must only differ by one refinement level. Hence, one mesh element cannot be divided twice if its neighbor has not been divided at all.
- Where possible, regular refinement of an element takes place. Regular refinement means that all the edges of an element are divided into two, and the element split accordingly. To make this possible, extra nodes may be added.
- No "hanging" nodes are allowed. This means that if an extra node is added to an edge, all the mesh elements that share that edge must be refined.
- Only certain types of elements are allowed in the refined mesh: tetrahedron, prism, pyramid and hexahedron.

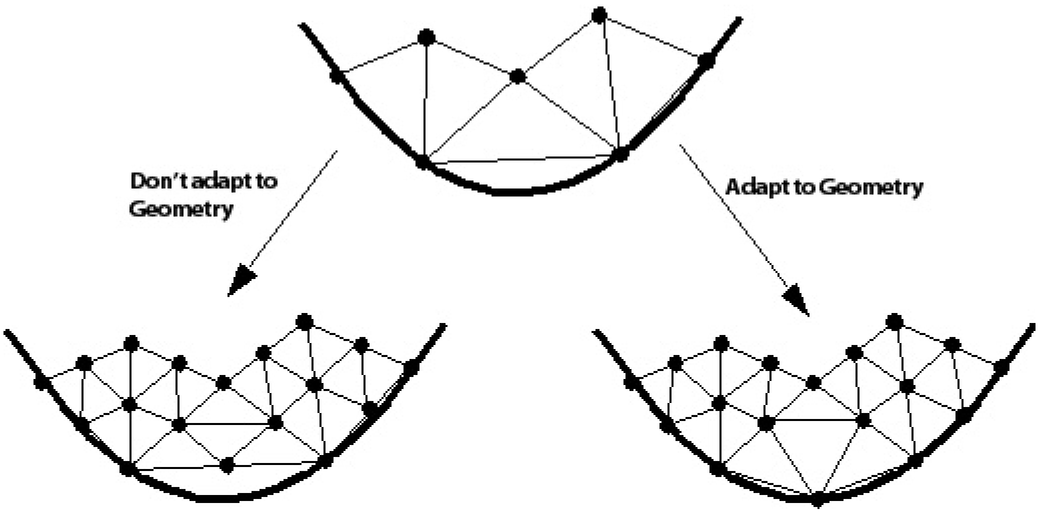
1.11.2.1. Adaption in Inflated Regions of the Mesh

In regions where inflation has taken place (so that there are prisms and some pyramid elements near wall boundary conditions), the mesh adaption avoids refining these elements in the direction perpendicular to the wall. Only edges on the interface between the inflated elements and the rest of the tetrahedral mesh are allowed to be marked for adaption. When the refinement of these edges takes place, the refinement propagates through the layers of prismatic elements to the wall boundary condition itself.



1.11.2.2. Adaption to the Original Geometry

Mesh adaption in ANSYS CFX does *not* have the capability of refining the surface mesh back to the original geometry. This means that nodes that are added to the surface of the problem are added onto the edges of the existing mesh rather than to the surfaces of the original geometry. An illustration of these two alternatives is shown below. Note that only the option shown on the left is available in ANSYS CFX.



Note

Support for adaption of the mesh to the original surface geometry is expected in future releases.

1.11.3. Mesh Adaption Limitations

Mesh Adaption in ANSYS CFX 14.0 is subject to the following limitations:

- Mesh adaptation is limited to allocate 2^31-1 words of 4 byte integer stack space; this limits the maximum problem size for mesh adaptation to approximately 80 million elements (structured) and 200 million elements (unstructured).
- Mesh adaption cannot be used in multi-domain simulations or in cases with external solver coupling. Mesh adaption also cannot be used for transient, mesh-motion, radiative-tracking, or particle-transport cases.
- The mesh may not be coarsened such that original mesh elements are combined. Mesh elements that have been refined in earlier adaption steps may be coarsened, back to their original mesh elements, but not beyond.
- You must specify how much adaption takes place by specifying the maximum number of nodes in the final mesh. You cannot specify that mesh adaption should take place until the mesh stops changing or until a particular solution criterion is satisfied.
- The method used is not well-suited to meshes with many high-aspect ratio elements, because it can only refine elements in an isotropic manner.
- The method cannot improve the quality of the existing mesh.

1.11.3.1. Mesh Adaption Tips

- More than one adaption step applied to a surface mesh with no underlying geometry can lead to convergence problems with transonic problems.
- If you set **Mesh Adaption Criteria Method** to `Solution Variation` without specifying **Minimum Edge Length**, then you may over-refine in regions where there are discontinuities. If you want to adapt meshes containing geometrically small features, you can exploit the adaption capability without limitation by not setting any minimum edge length.

- If you set **Mesh Adaption Criteria Method** to `Variation * Edge Length`, then you will not over-refine discontinuities. This method places an emphasis on adapting long edges with large variations of solution variable in preference to short edges with large variations of solution variable. This method will also refine long edges with a small variation of solution variable.
- Adaption criteria values for each node will be computed as the average value of all edges connected to the node, and these values will be stored in the results file for visualization purposes.
- If you find that the mesh adaption appears to have missed a discontinuity in a solution by refining the mesh in front of or behind the discontinuity, then your solution was not sufficiently converged before mesh adaption took place. As a solution containing a discontinuity develops, the location of such a discontinuity may move. If the solution is not sufficiently converged before mesh adaption takes place, then the mesh will be refined at the location of the discontinuity, which will move as convergence continues. A lower **Target Residual** can be set on the **Mesh Adaption/Advanced Parameters** tab to correct this problem.

1.12. Flow in Porous Media

Flow in porous media in ANSYS CFX can be calculated using either:

- A fluid domain together with a model for momentum loss. The effects of porosity are accounted for only through this loss term; all other terms in the governing equations are not changed. As a result, this formulation can be called the "superficial velocity formulation". For information on the momentum loss models available with this formulation, see *Momentum Sources* (p. 54).
- A porous domain that involves one or more fluids and an optional solid. Porosity modifies all terms in the governing equations as well as the loss term. As a result, this formulation can be called the "true velocity formulation" or the "full porous model". This method supports solid models (for example, for modeling thermal conductivity of the solid), and models for the interaction between the fluid and solid parts of the domain.

1.12.1. Full Porous Model

The full porous model is at once both a generalization of the Navier-Stokes equations and of Darcy's law commonly used for flows in porous regions. It can be used to model flows where the geometry is too complex to resolve with a grid. The model retains both advection and diffusion terms and can therefore be used for flows in rod or tube bundles where such effects are important.

In deriving the continuum equations, it is assumed that 'infinitesimal' control volumes and surfaces are large relative to the interstitial spacing of the porous medium, but small relative to the scales that you want to resolve. Thus, given control cells and control surfaces are assumed to contain both solid and fluid regions.

The **volume porosity** γ at a point is the ratio of the volume V' available to flow in an infinitesimal control cell surrounding the point, and the physical volume V of the cell. Hence:

$$V' = \gamma \, V \tag{1–266}$$

It is assumed that the vector area available to flow, $\boldsymbol{A}'$, through an infinitesimal planar control surface of vector area $\boldsymbol{A}$ is given by:

$$A' = \boldsymbol{K} \cdot \boldsymbol{A} \tag{1–267}$$

where: $\boldsymbol{K} = \left(K^{ij}\right)$ is a symmetric second rank tensor, called the **area porosity tensor**. Recall that the dot product of a symmetric rank two tensor with a vector is the vector $\boldsymbol{K} \cdot \boldsymbol{A}^i = K^{ij} A_j$. ANSYS CFX presently allows only $\boldsymbol{K}$ to be isotropic; that is, $K^{ij} = \gamma\delta^{ij}$.

The general scalar advection-diffusion equation in a porous medium becomes:

$$\frac{\partial}{\partial t}(\gamma \rho \Phi) + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U} \Phi) - \nabla \cdot (\Gamma \boldsymbol{K} \cdot \nabla \Phi) = \gamma S \tag{1–268}$$

In addition to the usual production and dissipation terms, the source term S will contain transfer terms from the fluid to the solid parts of the porous medium.

In particular, the equations for conservation of mass and momentum are:

$$\frac{\partial}{\partial t}\gamma \rho + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U}) = 0 \tag{1–269}$$

and:

$$\frac{\partial}{\partial t}(\gamma \rho \boldsymbol{U}) + \nabla \cdot \left(\rho (\boldsymbol{K} \cdot \boldsymbol{U}) \otimes \boldsymbol{U}\right) - \nabla \cdot \left(\mu_e \boldsymbol{K} \cdot \left(\nabla \boldsymbol{U} + (\nabla \boldsymbol{U})^{\mathrm{T}} - \frac{2}{3} \delta \nabla \cdot \boldsymbol{U}\right)\right)$$

$$= \gamma \boldsymbol{S}_M - \gamma \nabla p \tag{1–270}$$

where $\boldsymbol{U}$ is the true velocity, μ_e is the effective viscosity - either the laminar viscosity or a turbulent quantity, and $\boldsymbol{S}_M$ is a momentum source, which could be written as $-\boldsymbol{R} \cdot \boldsymbol{U}$ (where $\boldsymbol{R} = \left(R^{ij}\right)$ and represents a resistance to flow in the porous medium). This is in general a symmetric positive definite second rank tensor, in order to account for possible anisotropies in the resistance.

In the limit of large resistance, a large adverse pressure gradient must be set up to balance the resistance. In that limit, the two terms on the right-hand side of *Equation 1–270* (p. 79) are both large and of opposite sign, and the convective and diffusive terms on the left-hand side are negligible. Hence, *Equation 1–270* (p. 79) reduces to:

$$\boldsymbol{U} = -\boldsymbol{R}^{-1} \cdot \nabla p$$

Hence, in the limit of large resistance, you obtain an anisotropic version of Darcy's law, with permeability proportional to the inverse of the resistance tensor. However, unlike Darcy's law, you are working with the actual fluid velocity components $\boldsymbol{U}$, which are discontinuous at discontinuity in porosity, rather than the continuous averaged superficial velocity, $\boldsymbol{Q} = \boldsymbol{K} \cdot \boldsymbol{U}$

1.12.1.1. Heat Transfer Through the Fluid Only

Heat transfer can be modeled with an equation of similar form:

$$\frac{\partial}{\partial t}(\gamma \rho H) + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U} H) - \nabla \cdot (\Gamma_e \boldsymbol{K} \cdot \nabla H) = S^H \tag{1–271}$$

where Γ_e is an effective thermal diffusivity and S^H contains a heat source or sink to or from the porous medium.

1.12.1.2. Additional Variable Transfer Through the Fluid Only

For the fluid phases:

$$\frac{\partial \gamma\rho\varphi}{\partial t} + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U} \varphi) - \nabla \cdot (\Gamma_e \boldsymbol{K} \cdot \nabla \varphi) = S^\varphi \tag{1–272}$$

where Γ_e is an effective diffusivity.

1.12.1.3. Heat Transfer Through the Fluid and Solid

The current porous solid heat transfer formulation allows a finite temperature difference between the fluid phases and the solid phase. It is a non-thermal equilibrium model, therefore there are separate energy equations for each phase within the domain (N fluid phases plus one solid phase). In addition, it makes no assumption on the solid material properties.

For the fluid phases:

$$\frac{\partial \gamma\rho h}{\partial t} + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U} H) - \nabla \cdot (\Gamma_e \boldsymbol{K} \cdot \nabla H) = \gamma S_k^h + Q_{\mathrm{fs}} \tag{1–273}$$

For the solid phase:

$$\frac{\partial \gamma_{\mathrm{s}}\rho_{\mathrm{s}} C_{\mathrm{s}} T_{\mathrm{s}}}{\partial t} + \nabla \cdot (\rho \boldsymbol{K}_{\mathrm{s}} \cdot \boldsymbol{U}_{\mathrm{s}} C_{\mathrm{s}} T_{\mathrm{s}}) - \nabla \cdot (\lambda \boldsymbol{K}_{\mathrm{s}} \cdot \nabla T_{\mathrm{s}}) = \gamma_{\mathrm{s}} S_{\mathrm{s}}^T + Q_{\mathrm{sf}} \tag{1–274}$$

where $\gamma_s = 1 - \gamma$, and the interfacial heat transfer between the fluid and the solid, Q_{fs}, is determined using an overall heat transfer coefficient model using:

$$Q_{\mathrm{fs}} = -Q_{\mathrm{sf}} = h A_{\mathrm{fs}} (T_{\mathrm{s}} - T_{\mathrm{f}}) \tag{1–275}$$

h is the overall heat transfer coefficient between the fluid and the solid.

A_{fs} is the interfacial area density between the fluids and the solid. For multiphase flows, this concept can be split into a fluid-independent interfacial area density and a contact area fraction between the fluid and the solid:

$$A_{\mathrm{fs}} = A_{\alpha} \left(\frac{A}{V}\right) \tag{1–276}$$

where A_{α} represents the contact area fraction of fluid α with the solid. For single-phase flows, $A_{\alpha} = 1$.

1.12.1.4. Additional Variable Transfer Through the Fluid and Solid

For the fluid phases:

$$\frac{\partial \gamma\rho\varphi}{\partial t} + \nabla \cdot (\rho \boldsymbol{K} \cdot \boldsymbol{U}\, \varphi) - \nabla \cdot (\Gamma_e \boldsymbol{K} \cdot \nabla \varphi) = S^{\varphi} + T_{\mathrm{fs}} \tag{1–277}$$

For the solid phase:

$$\frac{\partial \gamma_{\mathrm{s}}\rho_{\mathrm{s}}\varphi_{\mathrm{s}}}{\partial t} + \nabla \cdot (\rho \boldsymbol{K}_{\mathrm{s}} \cdot \boldsymbol{U}_{\mathrm{s}}\, \varphi) - \nabla \cdot (\Gamma \boldsymbol{K}_{\mathrm{s}} \cdot \nabla \varphi) = S_{\mathrm{s}}^{\varphi} + T_{\mathrm{sf}} \tag{1–278}$$

where $\gamma_s = 1 - \gamma$, and the interfacial transfer between the fluid and the solid, Q_{fs}, is determined using an overall transfer coefficient model using:

$$T_{\mathrm{fs}} = -T_{\mathrm{sf}} = \tau A_{\mathrm{v}} (\varphi_{\mathrm{s}} - \varphi_{\mathrm{f}}) \tag{1–279}$$

τ is the overall additional variable transfer coefficient between the fluid and the solid.

A_{fs} is described in *Heat Transfer Through the Fluid and Solid* (p. 80).

1.12.1.5. Time-varying Porosity

For calculations without any heat or Additional Variable transfer through the solid, it is possible to use a porosity that varies with time. For calculations that do include heat or Additional Variable transfer through the solid, this is inadvisable because the term proportional to the rate of change of porosity is omitted from the equations within the solid.

1.12.2. Porous Momentum Loss Models

The momentum loss models available for porous domains are identical to the loss models available with the superficial velocity formulation. In particular, two options are available:

- *Isotropic Loss Model* (p. 54)
- *Directional Loss Model* (p. 55)

Note that, in these models, the velocity may refer to either true velocity or superficial velocity. Changing from one to the other leads to different coefficients. For fluid domains, the solver assumes that the coefficients multiply the superficial velocity, while for porous domains, you may choose either true or superficial velocity.

1.13. Wall and Boundary Distance Formulation

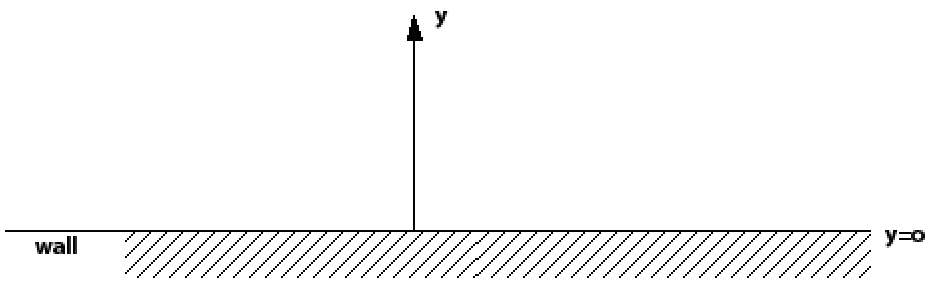
Wall and boundary distances are used in various functions that control the transition between near-wall and freestream turbulence models, and in the mesh deformation model. These distances are derived using the same formulation, which is described here.

The following topics are discussed:

- *1D Illustration of Concept* (p. 82)
- *Concept Generalized to 3D* (p. 83)

1.13.1. 1D Illustration of Concept

Consider the 1D case of a horizontal surface, with the y direction normal to the surface:



Consider the following Poisson equation for a variable ϕ:

$$\frac{d^2\phi}{dy^2} = -1 \qquad \textbf{(1–280)}$$

with a boundary condition of $\phi = 0$ at the wall.

Integrate once:

$$\frac{d\phi}{dy} = -y + C_1 \qquad \textbf{(1–281)}$$

Integrate again:

$$\phi = -0.5y^2 + C_1 y + C_2 \qquad \textbf{(1–282)}$$

Because $\phi = 0$ at $y = 0$, you can deduce that $C_2 = 0$. You also know from *Equation 1–280* (p. 82) that:

$$C_1 = \frac{d\phi}{dy} + y \tag{1–283}$$

Substituting into *Equation 1–282* (p. 82) for C_1 and C_2, you are left with a quadratic equation for y in terms of ϕ and $(d\phi/dy)$:

$$\phi = 0.5y^2 + \frac{d\phi}{dy}y \tag{1–284}$$

Rearranging this gives:

$$ay^2 + by + c = 0 \tag{1–285}$$

where $a = 0.5$, $b = \frac{d\phi}{dy}$, and $c = -\phi$

Solving this quadratic and choosing the positive root gives:

$$y = -\frac{d\phi}{dy} + \sqrt{\left(\frac{d\phi}{dy}\right)^2 + 2\phi} \tag{1–286}$$

1.13.2. Concept Generalized to 3D

Equation 1–280 (p. 82) becomes a diffusion-only transport equation with a uniform source term of unity:

$$\nabla^2\phi = -1 \tag{1–287}$$

Dirichlet conditions of $\phi = 0$ are applied on boundaries of interest (such as walls), and Neumann (or zero flux) conditions are applied on other boundaries. *Equation 1–286* (p. 83) is then evaluated, replacing $(d\phi/dy)$ with $\nabla\phi$ ($d\phi/dy$ is always positive) and y is interpreted as the distance to the nearest boundary where $\phi = 0$ was set. This gives the following expression for desired distance:

$$\text{Wall Distance} = -|\nabla\phi| + \sqrt{|\nabla\phi|^2 + 2\phi} \tag{1–288}$$

Because ϕ is always positive, the wall distance is also always positive.

1.14. Wall Condensation Theory

The wall condensation model may be used to model condensation of a condensable component of a variable composition mixture, such as steam and air. Condensation may occur on walls or fluid solid boundaries that are sufficiently cold to permit condensation onto a thin liquid film. It assumes that the primary resistance to heat transfer is caused by the boundary gradient of concentration of the condensable component, and that the liquid film is sufficiently thin that its contribution to heat transfer resistance is negligible. Condensation is modeled as a boundary mass sink of the condensable component, effectively removing the condensed mass of the condensable from the fluid domain. Flow in the liquid film is not modeled.

This chapter describes the following:

1.14.1. Wall Condensation Model

A laminar boundary layer model (Equation 1–298 in the *CFX-Solver Theory Guide*) is employed for laminar flow. A turbulent boundary layer model (Equation 1–304 in the *CFX-Solver Theory Guide*) is employed for turbulent flow. This makes use of the turbulent wall functions.

Although you see the Wall Condensation Model as a wall boundary condition for component mass fractions, the internal solver implementation extends the boundary source machinery rather than the boundary condition machinery. Wall condensation mass sinks are implemented as internally coded boundary sources which use the same infrastructure as user-defined boundary sources. The implementation is consistent with that of a user boundary mass source with:

Sink Option = Specified Mass Fractions and Local Temperature **(1–289)**

and with values of outgoing mass fractions to be assigned as follows:

Unique Condensable Component Mass Fraction = 1

Non-Condensable Component Mass Fractions = 0 **(1–290)**

The following topics are discussed:

1.14.1.1. Laminar Boundary Layer Model

In a laminar boundary layer, the molar fluxes J_A, J_B of a non-condensable component A and a condensable component B of a binary gaseous mixture through a plane parallel to, and at a distance y from, a wall, are given by:

$$J_A = J_m X_A - D_{AB} c_m \frac{\partial X_A}{\partial y} = 0 \qquad \text{(non-condensable)} \tag{1–291}$$

$$J_B = J_m X_B - D_{AB} c_m \frac{\partial X_B}{\partial y} \qquad \text{(condensable)} \tag{1–292}$$

D_{AB} is the binary diffusion coefficient, J_m is the molar convective flux of the mixture, c_m is the molar density of the mixture and X_A, X_B are the molar fractions of the non-condensable and condensable components, respectively.

Adding Equation 1–291 (p. 84) and Equation 1–292 (p. 84), using $X_A + X_B = 1$, gives $J_B = J_m$. Hence, substituting into (Equation 1–292 (p. 84)) and rearranging gives:

$$J_B = -D_{AB} \frac{c_m}{1 - X_B} \frac{\partial X_B}{\partial y} \tag{1–293}$$

Hence, integrating from the wall ($y=0$) to the edge of the boundary layer ($y=\delta$) gives:

$$\int_0^{\delta} J_B dy = D_{AB} c_m \ln\left(\frac{1 - X_B(\delta)}{1 - X_B(0)} \right)$$

Finally, assuming J_B= constant in the boundary layer gives:

$$J_B = \frac{D_{AB} c_m}{\delta} \ln\left(\frac{1 - X_B(\delta)}{1 - X_B(0)} \right) \tag{1–294}$$

It is convenient to express this result in terms of mass concentrations ρ_{A}, ρ_B, ρ_m, and mass fluxes M_{A}, M_B, M_m. Component mass concentrations are related to component molar concentrations and molecular weights W_{A}, W_B as follows:

$$\rho_{\mathrm{A}} = W_A c_A, \qquad \rho_B = W_B c_B \tag{1–295}$$

Also, mixture concentrations are related to component concentrations as follows:

$$\rho_m = \rho_A + \rho_{\mathrm{B}}, \qquad c_{\mathrm{m}} = c_A + c_B \tag{1–296}$$

From Equation 1–295 (p. 85) and Equation 1–296 (p. 85), we define a mixture molecular weight relating mixture mass and molar concentrations as follows:

$$\rho_{\mathrm{m}} = W_m c_m \quad \Rightarrow \quad W_m = \frac{W_A c_A + W_B c_B}{c_A + c_B} \tag{1–297}$$

Finally, we can substitute Equation 1–297 (p. 85) into Equation 1–294 (p. 85) to express the condensable component mass flux as follows:

$$M_B = W_B J_B = m_B \ln\left(\frac{1 - X_B(\delta)}{1 - X_B(0)} \right) \tag{1–298}$$

where the mass transfer coefficient for laminar flow, m_B, is given by:

$$m_B = \frac{W_B}{W_m} \frac{D_{AB} \rho_m}{\delta} \quad \textbf{(1–299)}$$

The condensable component molar fraction at the interface is determined by assuming that the vapor is in thermal equilibrium with the liquid film at the interface, and hence its partial pressure is equal to its saturation pressure at the interface temperature. In reality, the vapor at the edge of the boundary layer may be a supersaturated wet vapor, or mist. Only the dry part of the vapor will form the concentration gradient which drives the condensation mass flux. Hence, only the molar fraction of dry vapor must be used to determine the mass flux. This is achieved by clipping the molar fraction at the edge of the boundary layer as follows:

$$X_B(\delta) = \min(X_B(\delta), X_{Bsat}(\delta)) \quad \textbf{(1–300)}$$

where X_{Bsat} denotes the saturated molar fraction.

It is assumed that the liquid droplets suspended in the vapor are not transported to the interface, but merely act as a reservoir for the production of dry vapor to counter the depletion of the boundary layer by condensation.

1.14.1.2. Turbulent Boundary Layer Model

For turbulent flows, the concentration boundary layer is modeled using turbulent wall functions. Consequently, we use discrete versions of *Equation 1–291* (p. 84), *Equation 1–292* (p. 84) employing turbulent wall functions to model the mass fluxes in the turbulent boundary layer. Thus:

$$J_{Aw} = J_m X_{Aw} - T_M (X_{AP} - X_{Aw}) = 0 \qquad \text{(non-condensable)} \quad \textbf{(1–301)}$$

$$J_{Bw} = J_m X_{Bw} - T_M (X_{AP} - X_{Aw}) \qquad \text{(condensable)} \quad \textbf{(1–302)}$$

Here, *w* subscripts refer to wall quantities, *P* subscripts refer to near wall mesh points, and T_M is the wall multiplier determined by the form of the turbulent wall function ($T_M = c_m D_{AB} / \delta$ for laminar flow). Performing the same manipulations as in the previous section, we obtain:

$$J_{Bm} = -T_M \frac{X_{BP} - X_{Bw}}{1 - X_{Bw}} \quad \textbf{(1–303)}$$

As the multi-component flow model solves for mass fractions rather than molar fractions, it is more convenient to work with mass fluxes and mass fractions rather than molar fluxes and mole fractions. This gives the following formula for the condensation mass flux:

$$M_{Bw} = -T_M \frac{Y_{BP} - Y_{Bw}}{1 - Y_{Bw}} \quad \textbf{(1–304)}$$

Note that the turbulent wall function array is only defined at no-slip walls. It is not defined at free slip walls, specified stress walls, and non-overlap boundary conditions. Hence, for CFX Release 14, wall condensation is disallowed on these types of walls in the case of turbulent flows.

1.14.2. Condensation Heat Transfer (CHT)

At a fluid-solid boundary, the latent heat released by condensation to the liquid film is assumed to be absorbed by the solid phase at the interface. This gives a heat source to the solid side:

$$Q = -M_{BW} L \tag{1–305}$$

where L is the latent heat of vaporization.

At an isothermal wall boundary, the wall is assumed to constitute an infinite reservoir on which the effect of the condensation heat source is negligible, hence it is maintained at its constant temperature.

1.14.3. Specification of Secondary Fluxes

Strictly speaking, it is necessary for you to specify only secondary fluxes for incoming boundary mass sources. For this model of wall condensation, the corresponding boundary mass sources are outgoing. The solver implementation of the wall condensation model exploits the boundary mass source machinery. All secondary fluxes are computed using the local solution values of variables, with the single exception of component mass sources to the mass fraction equations. Consequently, for this model, all secondary source information is automatically included.

It is possible in principal to define other boundary mass sources in addition to the condensation sinks, provided that they are consistent with the internal options used for the wall condensation term. However, in the current implementation, additional mass sources on wall condensation boundaries are disallowed.

Chapter 2: Turbulence and Wall Function Theory

This chapter describes:

2.1. Turbulence Models

Turbulence consists of fluctuations in the flow field in time and space. It is a complex process, mainly because it is three dimensional, unsteady and consists of many scales. It can have a significant effect on the characteristics of the flow. Turbulence occurs when the inertia forces in the fluid become significant compared to viscous forces, and is characterized by a high Reynolds Number.

In principle, the Navier-Stokes equations describe both laminar and turbulent flows without the need for additional information. However, turbulent flows at realistic Reynolds numbers span a large range of turbulent length and time scales, and would generally involve length scales much smaller than the smallest finite volume mesh, which can be practically used in a numerical analysis. The Direct Numerical Simulation (DNS) of these flows would require computing power which is many orders of magnitude higher than available in the foreseeable future.

To enable the effects of turbulence to be predicted, a large amount of CFD research has concentrated on methods which make use of *turbulence models*. Turbulence models have been specifically developed to account for the effects of turbulence without recourse to a prohibitively fine mesh and direct numerical simulation. Most turbulence models are statistical turbulence model, as described below. The two exceptions to this in ANSYS CFX are the Large Eddy Simulation model and the Detached Eddy Simulation model.

- *Large Eddy Simulation Theory* (p. 126)
- *Detached Eddy Simulation Theory* (p. 132)

2.1.1. Statistical Turbulence Models and the Closure Problem

When looking at time scales much larger than the time scales of turbulent fluctuations, turbulent flow could be said to exhibit average characteristics, with an additional time-varying, fluctuating component. For example, a velocity component may be divided into an average component, and a time varying component.

In general, turbulence models seek to modify the original unsteady Navier-Stokes equations by the introduction of averaged and fluctuating quantities to produce the Reynolds Averaged Navier-Stokes (RANS) equations. These equations represent the mean flow quantities only, while modeling turbulence effects without a need for the resolution of the turbulent fluctuations. All scales of the turbulence field are being modeled. Turbulence models based on the RANS equations are known as Statistical Turbulence Models due to the statistical averaging procedure employed to obtain the equations.

Simulation of the RANS equations greatly reduces the computational effort compared to a Direct Numerical Simulation and is generally adopted for practical engineering calculations. However, the averaging procedure introduces additional unknown terms containing products of the fluctuating quantities, which act like additional stresses in the fluid. These terms, called 'turbulent' or 'Reynolds' stresses, are difficult to determine directly and so become further unknowns.

The Reynolds (turbulent) stresses need to be modeled by additional equations of known quantities in order to achieve "closure." Closure implies that there is a sufficient number of equations for all the unknowns, including the Reynolds-Stress tensor resulting from the averaging procedure. The equations used to close the system define the type of turbulence model.

2.1.1.1. Reynolds Averaged Navier-Stokes (RANS) Equations

As described above, turbulence models seek to solve a modified set of transport equations by introducing averaged and fluctuating components. For example, a velocity U_i may be divided into an average component, $\overline{U}_i$, and a time varying component, u_i.

$$U_i = \overline{U}_i + u_i \tag{2–1}$$

The averaged component is given by:

$$\overline{U}_i = \frac{1}{\Delta t} \int_t^{t+\Delta t} U_i dt \tag{2–2}$$

where Δt is a time scale that is large relative to the turbulent fluctuations, but small relative to the time scale to which the equations are solved. For compressible flows, the averaging is actually weighted by density (Favre-averaging), but for simplicity, the following presentation assumes that density fluctuations are negligible.

For transient flows, the equations are ensemble-averaged. This allows the averaged equations to be solved for transient simulations as well. The resulting equations are sometimes called URANS (Unsteady Reynolds Averaged Navier-Stokes equations).

Substituting the averaged quantities into the original transport equations results in the Reynolds averaged equations given below. For details, see Transport Equations (p. 25). In the following equations, the bar is dropped for averaged quantities, except for products of fluctuating quantities.

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j\right) = 0 \tag{2–3}$$

$$\frac{\partial \rho U_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_i U_j\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\tau_{ij} - \rho \overline{u_i u_j}\right) + S_M \tag{2–4}$$

where τ is the molecular stress tensor (including both normal and shear components of the stress).

The continuity equation has not been altered but the momentum and scalar transport equations contain turbulent flux terms additional to the molecular diffusive fluxes. These are the Reynolds stresses, $\rho \overline{u_i u_j}$. These terms arise from the nonlinear convective term in the un-averaged equations. They reflect the fact that convective transport due to turbulent velocity fluctuations will act to enhance mixing over and above that caused by thermal fluctuations at the molecular level. At high Reynolds numbers, turbulent velocity fluctuations occur over a length scale much larger than the mean free path of thermal fluctuations, so that the turbulent fluxes are much larger than the molecular fluxes.

The Reynolds averaged energy equation is:

$$\frac{\partial \rho h_{tot}}{\partial t} - \frac{\partial p}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j h_{tot}\right) = \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j} - \rho \overline{u_j h}\right) + \frac{\partial}{\partial x_j}\left[U_i \left(\tau_{ij} - \rho \overline{u_i u_j}\right) \right] + S_E \tag{2–5}$$

This equation contains an additional turbulence flux term, $\rho \overline{u_i h}$ compared with the instantaneous equation. For details on this, see *Equation 1–86* (p. 26). The $\frac{\partial}{\partial x_j}\left[U_j \left(\tau_{ij} - \rho \overline{u_i u_j}\right) \right]$ term in the equation is the viscous work term that can be included by enabling `Viscous Work` in CFX-Pre.

The mean Total Enthalpy is given by:

$$h_{tot} = h + \frac{1}{2} U_i U_i + k \tag{2–6}$$

Note that the Total Enthalpy contains a contribution from the turbulent kinetic energy, *k*, given by:

$$k = \frac{1}{2} \overline{u_i^2} \tag{2–7}$$

Similarly, the Additional Variable Φ may be divided into an average component, $\overline{\Phi}$, and a time varying component, φ. After dropping the bar for averaged quantities, except for products of fluctuating quantities, the Additional Variable equation becomes

$$\frac{\partial \rho \Phi}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \Phi\right) = \frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial \Phi}{\partial x_j} - \rho \overline{u_j \varphi}\right) + S_\Phi \tag{2–8}$$

where $\rho \overline{u_j \varphi}$ is the Reynolds flux.

Turbulence models close the Reynolds averaged equations by providing models for the computation of the Reynolds stresses and Reynolds fluxes. CFX models can be broadly divided into two classes: eddy viscosity models and Reynolds stress models.

2.2. Eddy Viscosity Turbulence Models

One proposal suggests that turbulence consists of small eddies which are continuously forming and dissipating, and in which the Reynolds stresses are assumed to be proportional to mean velocity gradients. This defines an "eddy viscosity model".

The eddy viscosity hypothesis assumes that the Reynolds stresses can be related to the mean velocity gradients and eddy (turbulent) viscosity by the gradient diffusion hypothesis, in a manner analogous to the relationship between the stress and strain tensors in laminar Newtonian flow:

$$-\rho \overline{u_i u_j} = \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} \delta_{ij} \left(\rho k + \mu_t \frac{\partial U_k}{\partial x_k} \right) \tag{2–9}$$

where μ_t is the eddy viscosity or turbulent viscosity, which must be modeled.

Analogous to the eddy viscosity hypothesis is the eddy diffusivity hypothesis, which states that the Reynolds fluxes of a scalar are linearly related to the mean scalar gradient:

$$-\rho \overline{u_i \varphi} = \Gamma_t \frac{\partial \Phi}{\partial x_i} \tag{2–10}$$

where Γ_t is the eddy diffusivity, and this has to be prescribed. The eddy diffusivity can be written as:

$$\Gamma_t = \frac{\mu_t}{Pr_t} \tag{2–11}$$

where Pr_t is the turbulent Prandtl number. Eddy diffusivities are then prescribed using the turbulent Prandtl number.

The above equations can express turbulent fluctuations in terms of functions of the mean variables only if the turbulent viscosity, μ_t, is known. Both the k-ε and k-ω two-equation turbulence models use this variable.

Subject to these hypotheses, the Reynolds averaged momentum and scalar transport equations become:

$$\frac{\partial \rho U_i}{\partial t} + \frac{\partial}{\partial x_j} \left(\rho U_i U_j \right) = -\frac{\partial p'}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\mu_{eff} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] + S_M \tag{2–12}$$

where S_M is the sum of the body forces, and μ_{eff} is the Effective Viscosity defined by:

$$\mu_{eff} = \mu + \mu_t \tag{2–13}$$

and p' is a modified pressure, defined by:

$$p' = p + \frac{2}{3}\rho k + \frac{2}{3}\mu_{eff}\frac{\partial U_k}{\partial x_k} \tag{2–14}$$

The last term in *Equation 2–14* (p. 93),

$$\frac{2}{3}\mu_{eff}\frac{\partial U_k}{\partial x_k}$$

involves the divergence of velocity. It is neglected in ANSYS CFX, although this assumption is strictly correct only for incompressible fluids.

The treatment of the second term in the right-hand side of *Equation 2–14* (p. 93) depends on the expert parameter `pressure value option`, which can take the following values:

Option	Description
1	When p is required (for example, to calculate material properties), it is derived from p' using *Equation 2–14* (p. 93).
2 This is the default.	When p is required (for example, to calculate material properties), it is approximated as being equal to p'. Option 2 is an approximation which may be numerically better-behaved than option 1.
3	p' is not computed using the term $(2/3)\ \rho\ k$. However, this term is incorporated directly in the momentum equation. This option is mathematically equivalent to option 1, but may differ numerically.

The Reynolds averaged energy equation becomes:

$$\begin{aligned}&\frac{\partial \rho h_{tot}}{\partial t} - \frac{\partial p}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j h_{tot}\right)\\ &= \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j} + \frac{\mu_t}{\Pr_t}\frac{\partial h}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left[U_i\left(\tau_{ij} - \rho \overline{u_i u_j}\right)\right] + S_E\end{aligned} \tag{2–15}$$

Note that although the transformation of the molecular diffusion term may be inexact if enthalpy depends on variables other than temperature, the turbulent diffusion term is correct, subject to the eddy diffusivity hypothesis. Moreover, as turbulent diffusion is usually much larger than molecular diffusion, small errors in the latter can be ignored.

Similarly, the Reynolds averaged transport equation for Additional Variables (non-reacting scalars) becomes:

$$\frac{\partial \rho \Phi}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \Phi\right) = \frac{\partial}{\partial x_j}\left[\left(\Gamma_\Phi + \frac{\mu_t}{\sigma_\Phi}\right)\frac{\partial \Phi}{\partial x_j}\right] + S_\Phi \tag{2–16}$$

Eddy viscosity models are distinguished by the manner in which they prescribe the eddy viscosity and eddy diffusivity.

2.2.1. The Zero Equation Model in ANSYS CFX

Very simple eddy viscosity models compute a global value for μ_{t} from the mean velocity and a geometric length scale using an empirical formula. Because no additional transport equations are solved, these models are termed 'zero equation.'

The zero equation model in ANSYS CFX uses an algebraic equation to calculate the viscous contribution from turbulent eddies. A constant turbulent eddy viscosity is calculated for the entire flow domain.

The turbulence viscosity is modeled as the product of a turbulent velocity scale, U_{t}, and a turbulence length scale, l_{t}, as proposed by Prandtl and Kolmogorov,

$$\mu_{\mathrm{t}} = \rho f_\mu U_{\mathrm{t}} l_{\mathrm{t}} \tag{2–17}$$

where f_μ is a proportionality constant. The velocity scale is taken to be the maximum velocity in the fluid domain. The length scale is derived using the formula:

$$l_{\mathrm{t}} = \left(V_D^{\frac{1}{3}}\right)/7 \tag{2–18}$$

where V_D is the fluid domain volume. This model has little physical foundation and is not recommended.

2.2.2. Two Equation Turbulence Models

Two-equation turbulence models are very widely used, as they offer a good compromise between numerical effort and computational accuracy. Two-equation models are much more sophisticated than the zero equation models. Both the velocity and length scale are solved using separate transport equations (hence the term 'two-equation').

The k-ε and k-ω two-equation models use the gradient diffusion hypothesis to relate the Reynolds stresses to the mean velocity gradients and the turbulent viscosity. The turbulent viscosity is modeled as the product of a turbulent velocity and turbulent length scale.

In two-equation models, the turbulence velocity scale is computed from the turbulent kinetic energy, which is provided from the solution of its transport equation. The turbulent length scale is estimated from two properties of the turbulence field, usually the turbulent kinetic energy and its dissipation rate. The dissipation rate of the turbulent kinetic energy is provided from the solution of its transport equation.

2.2.2.1. The k-epsilon Model in ANSYS CFX

k is the turbulence kinetic energy and is defined as the variance of the fluctuations in velocity. It has dimensions of (L^2 T^{-2}); for example, m^2/s^2. ε is the turbulence eddy dissipation (the rate at which the velocity fluctuations dissipate), and has dimensions of k per unit time (L^2 T^{-3}); for example, m^2/s^3.

The k-ε model introduces two new variables into the system of equations. The continuity equation is then:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j\right) = 0 \tag{2–19}$$

and the momentum equation becomes:

$$\frac{\partial \rho U_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_i U_j\right) = -\frac{\partial p'}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right] + S_M \tag{2–20}$$

where S_M is the sum of body forces, μ_{eff} is the effective viscosity accounting for turbulence, and p' is the modified pressure as defined in *Equation 2–14* (p. 93).

The k-ε model, like the zero equation model, is based on the eddy viscosity concept, so that:

$$\mu_{\text{eff}} = \mu + \mu_{\text{t}} \tag{2–21}$$

where μ_{t} is the turbulence viscosity. The k-ε model assumes that the turbulence viscosity is linked to the turbulence kinetic energy and dissipation via the relation:

$$\mu_{\text{t}} = C_\mu \rho \frac{k^2}{\varepsilon} \tag{2–22}$$

where C_μ is a constant. For details, see *List of Symbols* (p. 2).

The values of k and ε come directly from the differential transport equations for the turbulence kinetic energy and turbulence dissipation rate:

$$\frac{\partial (\rho k)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j k\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \rho\varepsilon + P_{kb} \tag{2–23}$$

$$\frac{\partial (\rho \varepsilon)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \varepsilon\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right] + \frac{\varepsilon}{k}\left(C_{\varepsilon 1} P_k - C_{\varepsilon 2}\rho\varepsilon + C_{\varepsilon 1} P_{\varepsilon b}\right) \tag{2–24}$$

where $C_{\varepsilon 1}, C_{\varepsilon 2}, \sigma_k$ and σ_ε are constants. For details, see *List of Symbols* (p. 2).

P_{kb} and $P_{\varepsilon b}$ represent the influence of the buoyancy forces, which are described below. P_k is the turbulence production due to viscous forces, which is modeled using:

$$P_k = \mu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\frac{\partial U_i}{\partial x_j} - \frac{2}{3}\frac{\partial U_k}{\partial x_k}\left(3\mu_t \frac{\partial U_k}{\partial x_k} + \rho k\right) \tag{2–25}$$

For incompressible flow, $(\partial U_k / \partial x_k)$ is small and the second term on the right side of *Equation 2–25* (p. 96) does not contribute significantly to the production. For compressible flow, $(\partial U_k / \partial x_k)$ is only large in regions with high velocity divergence, such as at shocks.

The term $3\,\mu_t$ in *Equation 2–25* (p. 96) is based on the "frozen stress" assumption [54]. This prevents the values of k and ε becoming too large through shocks, a situation that becomes progressively worse as the mesh is refined at shocks. The parameter Compressible Production (accessible on the **Advanced Control** part of the **Turbulence** section in CFX-Pre (see Turbulence in the *CFX-Pre User's Guide*)) can be used to set the value of the factor in front of μ_t, the default value is 3, as shown. A value of 1 will provide the same treatment as CFX-4.

In order to avoid the build-up of turbulent kinetic energy in stagnation regions, two production limiters are available. For details, see *Production Limiters* (p. 103).

2.2.2.2. Buoyancy Turbulence

If the full buoyancy model is being used, the buoyancy production term P_{kb} is modeled as:

$$P_{kb} = -\frac{\mu_t}{\rho\sigma_\rho} g_i \frac{\partial \rho}{\partial x_i} \tag{2–26}$$

and if the Boussinesq buoyancy model is being used, it is:

$$P_{kb} = \frac{\mu_t}{\rho\sigma_\rho}\rho\beta g_i \frac{\partial T}{\partial x_i} \tag{2–27}$$

This buoyancy production term is included in the k equation if the **Buoyancy Turbulence** option in CFX-Pre is set to `Production`, or `Production and Dissipation`. It is also included in the ε

equation if the option is set to `Production and Dissipation`. $P_{\varepsilon b}$ is assumed to be proportional to P_{kb} and must be positive, therefore it is modeled as:

$$P_{\varepsilon b} = C_3 \cdot \max\left(0, P_{kb}\right) \tag{2–28}$$

If the directional option is enabled, then $P_{\varepsilon b}$ is modified by a factor accounting for the angle ϕ between velocity and gravity vectors:

$$P_{\varepsilon b} = C_3 \cdot \max\left(0, P_{kb}\right) \cdot \sin\phi \tag{2–29}$$

Default model constants are given by:

Turbulence Schmidt Number σ_ρ:

- σ_ρ = 0.9 for Boussinesq buoyancy
- σ_ρ = 1 for full buoyancy model

Dissipation Coefficient, C_3 = 1

Directional Dissipation = Off

For omega based turbulence models, the buoyancy turbulence terms for the ω equation are derived from P_{kb} and $P_{\varepsilon b}$ according to the transformation $\varepsilon = \beta' \omega\, k$.

2.2.2.3. The RNG k-epsilon Model in ANSYS CFX

The RNG $k-\varepsilon$ model is based on renormalization group analysis of the Navier-Stokes equations. The transport equations for turbulence generation and dissipation are the same as those for the standard $k-\varepsilon$ model, but the model constants differ, and the constant $C_{\varepsilon 1}$ is replaced by the function $C_{\varepsilon 1\,\mathrm{RNG}}$.

The transport equation for turbulence dissipation becomes:

$$\begin{aligned} \frac{\partial(\rho\varepsilon)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \varepsilon\right) = {} & \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{\varepsilon RNG}}\right)\frac{\partial \varepsilon}{\partial x_j}\right] \\ & + \frac{\varepsilon}{k}\left(C_{\varepsilon 1RNG} P_k - C_{\varepsilon 2RNG}\rho\varepsilon + C_{\varepsilon 1RNG} P_{\varepsilon b}\right) \end{aligned} \tag{2–30}$$

where:

$$C_{\varepsilon 1\,\mathrm{RNG}} = 1.42 - f_\eta \tag{2–31}$$

and:

$$f_\eta = \frac{\eta\left(1 - \frac{\eta}{4.38}\right)}{\left(1 + \beta_{RNG}\,\eta^3\right)}$$

$$\eta = \sqrt{\frac{P_k}{\rho\, C_{\mu\, RNG}\, \epsilon}} \tag{2–32}$$

For details, see *List of Symbols* (p. 2).

2.2.2.4. The k-omega Model in ANSYS CFX

One of the advantages of the k-ω formulation is the near wall treatment for low-Reynolds number computations. The model does not involve the complex nonlinear damping functions required for the k-ε model and is therefore more accurate and more robust. A low-Reynolds k-ε model would typically require a near wall resolution of $y^+ < 0.2$, while a low-Reynolds number k-ω model would require at least $y^+ < 2$. In industrial flows, even $y^+ < 2$ cannot be guaranteed in most applications and for this reason, a new near wall treatment was developed for the k-ω models. It allows for smooth shift from a low-Reynolds number form to a wall function formulation.

The k-ω models assumes that the turbulence viscosity is linked to the turbulence kinetic energy and turbulent frequency via the relation:

$$\mu_t = \rho \frac{k}{\omega} \tag{2–33}$$

2.2.2.4.1. The Wilcox k-omega Model

The starting point of the present formulation is the k-ω model developed by Wilcox [11]. It solves two transport equations, one for the turbulent kinetic energy, k, and one for the turbulent frequency, ω. The stress tensor is computed from the eddy-viscosity concept.

k-equation:

$$\frac{\partial (\rho k)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j k\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \beta' \rho k \omega + P_{kb} \tag{2–34}$$

ω-equation:

$$\frac{\partial (\rho\omega)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \omega\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right] + \alpha\frac{\omega}{k}P_k - \beta\rho\omega^2 + P_{\omega b} \qquad \textbf{(2–35)}$$

In addition to the independent variables, the density, ρ, and the velocity vector, $\boldsymbol{U}$, are treated as known quantities from the Navier-Stokes method. P_k is the production rate of turbulence, which is calculated as in the k-ε model *Equation 2–25* (p. 96).

The model constants are given by:

$$\beta' = 0.09 \qquad \textbf{(2–36)}$$

$$\alpha = 5/9 \qquad \textbf{(2–37)}$$

$$\beta = 0.075 \qquad \textbf{(2–38)}$$

$$\sigma_k = 2 \qquad \textbf{(2–39)}$$

$$\sigma_\omega = 2 \qquad \textbf{(2–40)}$$

The unknown Reynolds stress tensor, $\rho\overline{u_i u_j}$, is calculated from:

$$-\rho\overline{u_i u_j} = \mu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\delta_{ij}\left(\rho k + \mu_t\frac{\partial U_k}{\partial x_k}\right) \qquad \textbf{(2–41)}$$

In order to avoid the build-up of turbulent kinetic energy in stagnation regions, two production limiters are available. For details, see *Production Limiters* (p. 103).

The buoyancy production term is included in the k-equation if the **Buoyancy Turbulence** option in CFX-Pre is set to **Production** or **Production and Dissipation**. The formulation is the same is given in *Equation 2–26* (p. 96) and *Equation 2–27* (p. 96).

The buoyancy turbulence terms for the ω-equation are derived from P_{kb} and $P_{\varepsilon b}$ according to the transformation P_{kb}.

The additional buoyancy term in the ω-equation reads:

$$P_{\omega b} = \frac{\omega}{k}\left(\left(\alpha + 1\right) C_3 \max\left(P_{kb}, 0\right) - P_{kb}\right) \qquad \textbf{(2–42)}$$

Here the first term on the right hand side, which comes from the ε-equation, is included only if the **Buoyancy Turbulence** option is set to **Production and Dissipation**. The second term, originating from the k-equation, is included with both **Production** and **Production and Dissipation** options.

If the directional option is enabled, then the first term on the right hand side is modified according to *Equation 2–29* (p. 97):

$$P_{\omega b} = \frac{\omega}{k}\left((\alpha + 1)\, C_3 \max\left(P_{kb}, 0\right) \cdot \sin\phi - P_{kb}\right) \tag{2–43}$$

2.2.2.5. The Baseline (BSL) k-Omega Model

The main problem with the Wilcox model is its well known strong sensitivity to freestream conditions (Menter [12]). Depending on the value specified for ω at the inlet, a significant variation in the results of the model can be obtained. This is undesirable and in order to solve the problem, a blending between the k-ω model near the surface and the k-ε model in the outer region was developed by Menter [9]. It consists of a transformation of the k-ε model to a k-ω formulation and a subsequent addition of the corresponding equations. The Wilcox model is thereby multiplied by a blending function F_1 and the transformed k-ε model by a function $1-F_1$. F_1 is equal to one near the surface and decreases to a value of zero outside the boundary layer (that is, a function of the wall distance). For details, see *Wall and Boundary Distance Formulation* (p. 82). At the boundary layer edge and outside the boundary layer, the standard k-ε model is therefore recovered.

Wilcox model:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j k\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{k1}}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \beta' \rho k \omega \tag{2–44}$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \omega\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{\omega 1}}\right)\frac{\partial \omega}{\partial x_j}\right] + \alpha_1 \frac{\omega}{k} P_k - \beta_1 \rho \omega^2 \tag{2–45}$$

Transformed $k - \varepsilon$ model:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j k\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{k2}}\right)\frac{\partial k}{\partial x_j}\right] + P_k - \beta' \rho k \omega \tag{2–46}$$

$$\begin{aligned} &\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j \omega\right) \\ &= \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_{\omega 2}}\right)\frac{\partial \omega}{\partial x_j}\right] + 2\rho \frac{1}{\sigma_{\omega 2}\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} + \alpha_2 \frac{\omega}{k} P_k - \beta_2 \rho \omega^2 \end{aligned} \tag{2–47}$$

Now the equations of the Wilcox model are multiplied by function F_1, the transformed k-ε equations by a function $1-F_1$ and the corresponding k- and ω- equations are added to give the BSL model. Including buoyancy effects the BSL model reads:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial}{\partial x_j}\left(\rho U_j k\right)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_{k3}}\right)\frac{\partial k}{\partial x_j}\right]+P_k-\beta' \rho k\omega+P_{kb} \tag{2–48}$$

$$\begin{aligned}&\frac{\partial(\rho\omega)}{\partial t}+\frac{\partial}{\partial x_j}\left(\rho U_j\omega\right)=\\&\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_{\omega 3}}\right)\frac{\partial \omega}{\partial x_j}\right]+(1-F_1)\,2\rho\frac{1}{\sigma_{\omega 2}\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}+\alpha_3\frac{\omega}{k}P_k-\beta_3\rho\omega^2+P_{\omega b}\end{aligned} \tag{2–49}$$

The coefficient α in the buoyancy production term $P_{\omega b}$ in *Equation 2–42* (p. 99) and *Equation 2–43* (p. 100) is also replaced by the new coefficient α_3.

The coefficients of the new model are a linear combination of the corresponding coefficients of the underlying models:

$$\Phi_3=F_1\,\Phi_1+(1-F_1)\,\Phi_2 \tag{2–50}$$

All coefficients are listed again for completeness:

$$\beta' = 0.09 \quad \textbf{(2–51)}$$

$$\alpha_1 = 5/9 \quad \textbf{(2–52)}$$

$$\beta_1 = 0.075 \quad \textbf{(2–53)}$$

$$\sigma_{k1} = 2 \quad \textbf{(2–54)}$$

$$\sigma_{\omega 1} = 2 \quad \textbf{(2–55)}$$

$$\alpha_2 = 0.44 \quad \textbf{(2–56)}$$

$$\beta_2 = 0.0828 \quad \textbf{(2–57)}$$

$$\sigma_{k2} = 1 \quad \textbf{(2–58)}$$

$$\sigma_{\omega 2} = 1/0.856 \quad \textbf{(2–59)}$$

2.2.2.6. The Shear Stress Transport (SST)

The k-ω based SST model accounts for the transport of the turbulent shear stress and gives highly accurate predictions of the onset and the amount of flow separation under adverse pressure gradients.

The BSL model combines the advantages of the Wilcox and the k-ε model, but still fails to properly predict the onset and amount of flow separation from smooth surfaces. The reasons for this deficiency are given in detail in Menter [9]. The main reason is that both models do not account for the transport of the turbulent shear stress. This results in an overprediction of the eddy-viscosity. The proper transport behavior can be obtained by a limiter to the formulation of the eddy-viscosity:

$$v_t = \frac{a_1 k}{\max\left(a_1 \omega, SF_2\right)} \quad \textbf{(2–60)}$$

where

$$v_t = \mu_t / \rho \quad \textbf{(2–61)}$$

Again F_2 is a blending function similar to F_1, which restricts the limiter to the wall boundary layer, as the underlying assumptions are not correct for free shear flows. S is an invariant measure of the strain rate.

2.2.2.6.1. Blending Functions

The blending functions are critical to the success of the method. Their formulation is based on the distance to the nearest surface and on the flow variables.

$$F_1 = \tanh\left(arg_1^{\,4}\right) \tag{2–62}$$

with:

$$arg_1 = \min\left(\max\left(\frac{\sqrt{k}}{\beta' \omega y}, \frac{500\, v}{y^2 \omega}\right), \frac{4 \rho k}{CD_{kw} \sigma_{\omega 2} y^2}\right) \tag{2–63}$$

where y is the distance to the nearest wall, v is the kinematic viscosity and:

$$CD_{k\omega} = \max\left(2\rho \frac{1}{\sigma_{\omega 2} \omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j}, 1.0 \times 10^{-10}\right) \tag{2–64}$$

$$F_2 = \tanh\left(arg_2^{\,2}\right) \tag{2–65}$$

with:

$$arg_2 = \max\left(\frac{2\sqrt{k}}{\beta' \omega y}, \frac{500\, v}{y^2 \omega}\right) \tag{2–66}$$

2.2.2.6.2. The Wall Scale Equation

During the solution of a simulation using the SST or BSL model, you will see a plot in the CFX-Solver Manager for **Wall Scale**. These models require the distance of a node to the nearest wall for performing the blending between k-ε and k-ω. Detailed information on the wall scale equation is available in *Wall and Boundary Distance Formulation* (p. 82).

2.2.2.7. Production Limiters

A disadvantage of standard two-equation turbulence models is the excessive generation of turbulence energy, P_{k}, in the vicinity of stagnation points. In order to avoid the build-up of turbulent kinetic energy in stagnation regions, a formulation of limiters for the production term in the turbulence equations is available.

The formulation follows Menter [9] and reads:

$$P_k = \min\left(P_k, C_{lim} \rho \varepsilon\right) \quad \textbf{(2–67)}$$

The coefficient C_{lim} is called `Clip Factor` and has a value of 10 for ω based models. This limiter does not affect the shear layer performance of the model, but has consistently avoided the stagnation point build-up in aerodynamic simulations.

Using the standard Boussinesq-approximation for the Reynolds stress tensor, the production term P_k can be expressed for incompressible flow as:

$$\begin{aligned} &P_k = \mu_t S^2 \\ &S = \sqrt{2 S_{ij} S_{ij}}, S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \end{aligned} \quad \textbf{(2–68)}$$

where S denotes the magnitude of the strain rate and S_{ij} the strain rate tensor.

Kato and Launder [128] noticed that the very high levels of the shear strain rate S in stagnation regions are responsible for the excessive levels of the turbulence kinetic energy. Because the deformation near a stagnation point is nearly irrotational, that is the vorticity rate Ω is very small, they proposed the following replacement of the production term:

$$\begin{aligned} &P_k = \mu_t S\, \Omega \\ &\Omega = \sqrt{2 \Omega_{ij} \Omega_{ij}}, \Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right) \end{aligned} \quad \textbf{(2–69)}$$

where Ω denotes the magnitude of the vorticity rate and Ω_{ij} the vorticity tensor. In a simple shear flow, S and Ω are equal. Therefore, formulation recovers in such flows, as seen in the first parts of *Equation 2–68* (p. 104) and *Equation 2–69* (p. 104).

The production limiters described above are available for the two ε based turbulence models and for the (k,ω)-, BSL- and SST-turbulence models. They are available in the **Advanced Control** settings of the **Turbulence Model** section in CFX-Pre. For details, see Turbulence: Option in the *CFX-Pre User's Guide*. The allowed options of the production limiter are `Clip Factor` and `Kato Launder`.

2.2.2.8. Curvature Correction for Two-Equation Models

One weakness of the eddy-viscosity models is that these models are insensitive to streamline curvature and system rotation. Based on the work of Spalart and Shur [191] a modification of the production term has been derived which allows to sensitize the standard two-equation models to these effects [192]. The empirical function suggested by Spalart and Shur [191] to account for these effects is defined by

$$f_{rotation} = (1 + c_{r1}) \frac{2r^*}{1 + r^*} \left[1 - c_{r3} \tan^{-1} (c_{r2} \tilde{r}) \right] - c_{r1} \tag{2–70}$$

It is used as a multiplier of the production term and has been limited in ANSYS CFX in the following way:

$$P_k \to P_k \cdot f_r \tag{2–71}$$

where

$$f_r = \max \left[0, 1 + C_{scale} \left(\tilde{f}_r - 1 \right) \right]$$

and

$$\tilde{f}_r = \max \left\{ \min (f_{rotation}, 1.25), 0 \right\}$$

The original function is limited in the range from 0.0 corresponding, for example, to a strong convex curvature (stabilized flow, no turbulence production) up to 1.25 (for example, strong concave curvature, enhanced turbulence production). The lower limit is introduced for numerical stability reasons, whereas the upper limit is needed to avoid overgeneration of the eddy viscosity in flows with a destabilizing curvature/rotation. The specific limiter 1.25 provided a good compromise for different test cases which have been considered with the SST model (for example, flow through a U-turn, flow in hydro cyclone, and flow over a NACA 0012 wing tip vortex [192]). The scaling coefficient C_{scale} has been introduced to allow the user to influence the effect of the curvature correction if needed for a specific flow. The default value of this scaling coefficient is 1.0 and can be changed on the **Fluid Models** tab of the **Domain** details view in CFX-Pre.

Assuming that all the variables and their derivatives are defined with respect to the reference frame of the calculation, which is rotating with a rate Ω^{rot}, the arguments r^* and $\tilde{r}$ of the function $f_{rotation}$ are defined in the following way:

$$r^* = \frac{S}{\Omega} \tag{2–72}$$

$$\tilde{r} = 2\Omega_{ik} S_{jk} \left[\frac{DS_{ij}}{Dt} + \left(\varepsilon_{imn} S_{jn} + \varepsilon_{jmn} S_{in} \right) \Omega_m^{Rot} \right] \frac{1}{\Omega D^3} \tag{2–73}$$

where the first term in brackets is equivalent to the 2nd velocity gradient (in this case the Lagrangian derivative of the strain rate tensor) and the 2nd term in the brackets is a measure of the system rotation. The strain rate and vorticity tensor are defined, respectively, using Einstein summation convention as

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \tag{2–74}$$

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right) + 2\varepsilon_{mji}\Omega_m^{rot} \tag{2–75}$$

where

$$S^2 = 2S_{ij}S_{ij}$$

$$\Omega^2 = 2\Omega_{ij}\Omega_{ij}$$

$$D^2 = \max\left(S^2, 0.09\omega^2\right)$$

and, where

$$\frac{DS_{ij}}{Dt}$$

are the components of the Lagrangian derivative of the strain rate tensor.

Finally, based on the performed tests, the empirical constants c_{r1}, c_{r2} and c_{r3} involved in *Equation 2–70* (p. 105) are set equal to 1.0, 2.0, and 1.0, respectively.

This curvature correction is available for ε and ω-based eddy-viscosity turbulence models ($k - \varepsilon$, RNG$k - \varepsilon$, k-ω, BSL, SST) as well as the DES-SST and SAS-SST turbulence models.

2.2.3. The Eddy Viscosity Transport Model

A very simple one-equation model has been developed by Menter [32] [33]. It is derived directly from the $k - \varepsilon$ model and is therefore named the $(k - \varepsilon)_{1\mathrm{E}}$ model:

$$\frac{\partial \rho \tilde{v}_t}{\partial t} + \frac{\partial \rho U_j \tilde{v}_t}{\partial x_j} = c_1 \rho \tilde{v}_t \boldsymbol{S} - c_2 \rho \left(\frac{\tilde{v}_t}{L_{vK}}\right)^2 + \left[\left(\mu + \frac{\rho \tilde{v}_t}{\sigma}\right)\frac{\partial \tilde{v}}{\partial x_j}\right] \tag{2–76}$$

where $\tilde{v}$ is the kinematic eddy viscosity, $\tilde{v}_t$ is the turbulent kinematic eddy viscosity and σ is a model constant.

The model contains a destruction term, which accounts for the structure of turbulence and is based on the von Karman length scale:

$$(L_{vK})^2 = \left| \frac{\boldsymbol{S}^2}{\frac{\partial \boldsymbol{S}}{\partial x_j} \frac{\partial \boldsymbol{S}}{\partial x_j}} \right| \qquad \textbf{(2–77)}$$

where **S** is the shear strain rate tensor. The eddy viscosity is computed from:

$$\mu_t = \rho \tilde{v}_t \qquad \textbf{(2–78)}$$

In order to prevent a singularity of the formulation as the von Karman length scale goes to zero, the destruction term is reformulated as follows:

$$\begin{aligned} E_{k-\varepsilon} &= \left(\frac{\tilde{v}_t}{L_{vK}} \right)^2 \\ E_{BB} &= \frac{\partial \tilde{v}_t}{\partial x_j} \frac{\partial \tilde{v}_t}{\partial x_j} \\ E_{1e} &= c_3 E_{BB} \tanh \left(\frac{E_{k-\varepsilon}}{c_3 E_{BB}} \right) \end{aligned} \qquad \textbf{(2–79)}$$

$$\frac{\partial \rho \tilde{v}_t}{\partial t} + \frac{\partial \rho U_j \tilde{v}_t}{\partial x_j} = c_1 D_1 \rho \tilde{v}_t \boldsymbol{S} - c_2 \rho E_{1e} + \left[\left(\mu + \frac{\rho \tilde{v}_t}{\sigma} \right) \frac{\partial \tilde{v}_t}{\partial x_j} \right] \qquad \textbf{(2–80)}$$

The coefficients are:

Coefficient	Value
c_1	0.144
c_2	1.86
c_3	7.0
A^+	13.5
κ	0.41
σ	1.0

By default, the model is solved in combination with the scalable wall function. For details, see *Scalable Wall Functions* (p. 140).

2.2.3.1. Low Reynolds Number Formulation

Low Reynolds formulation of the model is obtained by including damping functions. Near wall damping functions have been developed to allow integration to the surface:

$$D_1 = \frac{v_t + v}{\tilde{v}_t + v}$$
$$D_2 = 1 - \exp\left[- \left(\frac{\tilde{v}_t}{A^+ \kappa v} \right)^2 \right] \quad \text{(2–81)}$$

where D_2 is required to compute the eddy-viscosity which goes into the momentum equations:

$$\mu_t = \rho D_2 \tilde{v}_t \quad \text{(2–82)}$$

The low Reynolds formulation of the $(k - \varepsilon)_{1E}$ model requires a near wall mesh resolution of $y^+ < 1$.

2.3. Reynolds Stress Turbulence Models

These models are based on transport equations for all components of the Reynolds stress tensor and the dissipation rate. These models do not use the eddy viscosity hypothesis, but solve an equation for the transport of Reynolds stresses in the fluid. The Reynolds stress model transport equations are solved for the individual stress components.

Algebraic Reynolds stress models solve algebraic equations for the Reynolds stresses, whereas differential Reynolds stress models solve differential transport equations individually for each Reynolds stress component.

The exact production term and the inherent modeling of stress anisotropies theoretically make Reynolds Stress models more suited to complex flows; however, practice shows that they are often not superior to two-equation models.

The Reynolds averaged momentum equations for the mean velocity are:

$$\frac{\partial \rho U_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_i U_j\right) - \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right] = -\frac{\partial p''}{\partial x_i} - \frac{\partial}{\partial x_j}\left(\rho \overline{u_i u_j}\right) + S_{M_i} \quad \text{(2–83)}$$

where p'' is a modified pressure, S_{M_i} is the sum of body forces and the fluctuating Reynolds stress contribution is $\rho \overline{u_i u_j}$. Unlike eddy viscosity models, the modified pressure has no turbulence contribution and is related to the static (thermodynamic) pressure by:

$$p'' = p + \frac{2}{3}\mu \frac{\partial U_k}{\partial x_k} \quad \text{(2–84)}$$

In the differential stress model, $\rho \overline{u_i u_j}$ is made to satisfy a transport equation. A separate transport equation must be solved for each of the six Reynolds stress components of $\rho \overline{u_i u_j}$. The differential equation Reynolds stress transport is:

$$\frac{\partial \rho \overline{u_i u_j}}{\partial t} + \frac{\partial}{\partial x_k}\left(U_k \rho \overline{u_i u_j}\right) - \frac{\partial}{\partial x_k}\left(\left(\delta_{kl}\mu + \rho C_S \frac{k}{\varepsilon}\overline{u_k u_l}\right)\frac{\partial \overline{u_i u_j}}{\partial x_l}\right)$$
$$= P_{ij} - \frac{2}{3}\delta_{ij}\rho\varepsilon + \Phi_{ij} + P_{ij,b} \tag{2–85}$$

where P_{ij} and $P_{ij,b}$ are shear and buoyancy turbulence production terms of the Reynolds stresses respectively, Φ_{ij} is the pressure-strain tensor, and *C* is a constant.

Buoyancy turbulence terms $P_{ij,b}$ also take the buoyancy contribution in the pressure strain term into account and are controlled in the same way as for the $k - \varepsilon$ and $k - \omega$ model. See the discussion below *Equation 2–27* (p. 96).

2.3.1. The Reynolds Stress Model

The standard Reynolds Stress model in ANSYS CFX is based on the ε-equation. The CFX-Solver solves the following equations for the transport of the Reynolds stresses:

$$\frac{\partial \rho \overline{u_i u_j}}{\partial t} + \frac{\partial}{\partial x_k}\left(U_k \rho \overline{u_i u_j}\right) - \frac{\partial}{\partial x_k}\left(\left(\mu + \frac{2}{3}C_S \rho \frac{k^2}{\varepsilon}\right)\frac{\partial \overline{u_i u_j}}{\partial x_k}\right)$$
$$= P_{ij} - \frac{2}{3}\delta_{ij}\rho\varepsilon + \Phi_{ij} + P_{ij,b} \tag{2–86}$$

where φ_{ij} is the pressure-strain correlation, and P_{ij}, the exact production term, is given by:

$$P_{ij} = -\rho\overline{u_i u_k}\frac{\partial U_j}{\partial x_k} - \rho\overline{u_j u_k}\frac{\partial U_i}{\partial x_k} \tag{2–87}$$

The production due to buoyancy is

$$P_{ij,b} = B_{ij} - C_{buo}\left(B_{ij} - \frac{1}{3}B_{kk}\delta_{ij}\right) \tag{2–88}$$

where the second term represents the buoyancy contribution from the pressure-strain term (Launder [196]), and B_{ij} is given by

$$B_{ij} = g_i b_j + g_j b_i \tag{2–89}$$

If the Boussinesq buoyancy model is used, then b_i is modeled as

$$b_i = \frac{\mu_t}{\sigma_\rho}\beta\frac{\partial T}{\partial x_i} \tag{2–90}$$

where β is the thermal expansion coefficient. Otherwise the term is modeled as (full buoyancy model based on density differences)

$$b_i = -\frac{\mu_t}{\rho\sigma_\rho}\frac{\partial \rho}{\partial x_i} \tag{2–91}$$

As the turbulence dissipation appears in the individual stress equations, an equation for ε is still required. This now has the form:

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \frac{\partial}{\partial x_k}(\rho U_k \varepsilon) = \frac{\varepsilon}{k}(C_{\varepsilon 1}P_k - C_{\varepsilon 2}\rho\varepsilon + C_{\varepsilon 1}P_{\varepsilon b}) + \frac{\partial}{\partial x_k}\left[\left(\mu + \frac{\mu_t}{\sigma_{\varepsilon \mathrm{RS}}}\right)\frac{\partial \varepsilon}{\partial x_k}\right] \tag{2–92}$$

In these equations, the anisotropic diffusion coefficients of the original models have been replaced by an isotropic formulation, which increases the robustness of the Reynolds stress model.

The Reynolds Stress model is also available with anisotropic diffusion coefficients. In this case, the CFX-Solver solves the following equations for the transport of the Reynolds stresses:

$$\begin{aligned}&\frac{\partial \rho\overline{u_i u_j}}{\partial t} + \frac{\partial}{\partial x_k}\left(U_k\rho\overline{u_i u_j}\right) - \frac{\partial}{\partial x_k}\left(\left(\delta_{kl}\mu + \rho C_S\frac{k}{\varepsilon}\overline{u_k u_l}\right)\frac{\partial \overline{u_i u_j}}{\partial x_l}\right)\\&= P_{ij} - \frac{2}{3}\delta_{ij}\rho\varepsilon + \Phi_{ij} + P_{ij,b}\end{aligned} \tag{2–93}$$

where φ_{ij} is the pressure-strain correlation, and P_{ij}, the exact production term, is given by *Equation 2–87* (p. 109).

In this case the production due to buoyancy is for the Boussinesq approximation modeled as:

$$b_i = C_S\rho\beta\frac{k}{\varepsilon\sigma_\rho}\overline{u_i u_k}\frac{\partial T}{\partial x_k} \tag{2–94}$$

Otherwise the term is modeled as (full buoyancy model based on density differences):

$$b_i = -C_S\frac{k}{\varepsilon\sigma_\rho}\overline{u_i u_k}\frac{\partial \rho}{\partial x_k} \tag{2–95}$$

The equation for ε is:

$$\frac{\partial\ (\rho\,\varepsilon)}{\partial\,t} + \frac{\partial}{\partial\,x_k}(\rho\,U_k\,\varepsilon)$$
$$= \frac{\varepsilon}{k}(C_{\varepsilon 1}P_k - C_{\varepsilon 2}\rho\varepsilon + C_{\varepsilon 1}P_{\varepsilon b}) + \frac{\partial}{\partial\,x_k}\left[\left(\mu\,\delta_{k\,l} + c_\varepsilon\,\rho\,\frac{k}{\varepsilon}\,\overline{u_k\,u_l}\right)\frac{\partial\,\varepsilon}{\partial\,x_l}\right] \quad \textbf{(2–96)}$$

The model constants are listed below for each model.

2.3.1.1. Pressure-Strain Terms

One of the most important terms in Reynolds stress models is the pressure-strain correlation, φ_{ij}. It acts to drive turbulence towards an isotropic state by redistributing the Reynolds stresses.

The pressure strain term can be split into two parts:

$$\Phi_{ij} = \Phi_{ij,1} + \Phi_{ij,2} \quad \textbf{(2–97)}$$

where $\Phi_{ij,1}$ is the ‘slow’ term, also known as the return-to-isotropy term, and $\Phi_{ij,2}$ is called the ‘rapid’ term.

There are three varieties of the standard Reynolds stress models based on the ε-equation available. These are known as LRR-IP, LRR-QI and SSG. The LRR-IP and LRR-QI models were developed by Launder, Reece and Rodi [4]. In both models, the pressure-strain correlation is linear. “IP” stands for Isotropization of Production and is the simplest of the 3 models. The two terms read:

$$\Phi_{ij,1} = -C_1\rho\frac{\varepsilon}{k}\left(\overline{u_i u_j} - \frac{2}{3}\delta_{ij}k\right) \quad \textbf{(2–98)}$$

$$\Phi_{ij,2} = -C_2\left(P_{ij} - \frac{2}{3}P\delta_{ij}\right) \quad \textbf{(2–99)}$$

The values of the two coefficients are $C_1 = 1.8$ and $C_2 = 0.6$.P is given by $0.5P_{ii}$.

“QI” stands for Quasi-Isotropic and differs from the IP model in the formulation of the rapid term:

$$\begin{aligned}\Phi_{ij,2} = & -\frac{C_2+8}{11}\left(P_{ij} - \frac{2}{3}P\delta_{ij}\right) - \frac{30C_2-2}{55}\rho k\left(\frac{\partial\,U_i}{\partial\,x_j} + \frac{\partial\,U_j}{\partial\,x_i}\right) \\ & -\frac{8C_2-2}{11}\left(D_{ij} - \frac{2}{3}\delta_{ij}P\right)\end{aligned} \quad \textbf{(2–100)}$$

where D_{ij} is given by:

$$D_{ij} = -\rho \overline{u_i u_k} \frac{\partial U_k}{\partial x_j} - \rho \overline{u_j u_k} \frac{\partial U_k}{\partial x_i} \tag{2–101}$$

The SSG model was developed by Speziale, Sarkar and Gatski [5] and uses a quadratic relation for the pressure-strain correlation.

In order to compare the pressure-strain correlations for the three models, a general form can be derived based on the anisotropy tensor a_{ij} and the mean strain rate tensor and vorticity tensor, S_{ij} and Ω_{ij} respectively. The general form reads:

$$\Phi_{ij,1} = -\rho\varepsilon \left[C_{s1} a_{ij} + C_{s2} \left(a_{ik} a_{kj} - \frac{1}{3} a_{mn} a_{mn} \delta_{ij} \right) \right] \tag{2–102}$$

$$\begin{aligned} \Phi_{ij,2} = & -C_{r1} P a_{ij} + C_{r2} \rho k S_{ij} - C_{r3} \rho k S_{ij} \sqrt{a_{mn} a_{mn}} \\ & + C_{r4} \rho k \left(a_{ik} S_{jk} + a_{jk} S_{ik} - \frac{2}{3} a_{kl} S_{kl} \delta_{ij} \right) + C_{r5} \rho k \left(a_{ik} \Omega_{jk} + a_{jk} \Omega_{ik} \right) \end{aligned} \tag{2–103}$$

where

$$a_{ij} = \frac{\overline{u_i u_j}}{k} - \frac{2}{3} \delta_{ij} \tag{2–104}$$

$$S_{ij} = \frac{1}{2} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \tag{2–105}$$

$$\Omega_{ij} = \frac{1}{2} \left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i} \right) \tag{2–106}$$

This general form can be used to model linear and quadratic correlations by using appropriate values for the constants. The constants are listed in the table below for each model.

Model	$C_{\mu\mathrm{RS}}$	$\sigma_{\varepsilon\mathrm{RS}}$	c_s	c_ε	$c_{\varepsilon 1}$	$c_{\varepsilon 2}$
LRR-IP	0.1152	1.10	0.22	0.18	1.45	1.9
LRR-QI	0.1152	1.10	0.22	0.18	1.45	1.9
SSG	0.1	1.36	0.22	0.18	1.45	1.83

Model	C_{s1}	C_{s2}	C_{r1}	C_{r2}	C_{r3}	C_{r4}	C_{r5}
LRR-IP	1.8	0.0	0.0	0.8	0.0	0.6	0.6

Model	C_{s1}	C_{s2}	C_{r1}	C_{r2}	C_{r3}	C_{r4}	C_{r5}
LRR-QI	1.8	0.0	0.0	0.8	0.0	0.873	0.655
SSG	1.7	-1.05	0.9	0.8	0.65	0.625	0.2

Selection of the appropriate model is carried out on the **Fluid Models** panel of the **Domains** form in CFX-Pre. The following options correspond to the types of models listed above:

- Reynolds Stress Model - LRR-IP
- SSG Reynolds Stress Model - SSG
- QI Reynolds Stress Model - LRR-IQ

The coefficients of the pressure strain term can be specified in CFX-Pre in the general form for the LRR-QI and SSG models. The LRR-IP model has been added for completeness in the table above. In CFX-Pre the pressure strain coefficients of the LRR-IP are asked in the form as given in *Equation 2–98* (p. 111) and *Equation 2–99* (p. 111). For the Boussinesq buoyancy model the default value of σ_ρ is 0.9 for LRR-IP and LRR-QI and 2/3 for SSG. In the full buoyancy model based on density differences the default value of σ_ρ is 1.0 or all models.

2.3.2. Omega-Based Reynolds Stress Models

CFX provides two Reynolds Stress-ω models: the Omega Reynolds Stress and Baseline (BSL) Reynolds Stress models. The two models relate to each other in the same way as the two equation $k - \omega$ and BSL models. For details, see *The Baseline (BSL) k-Omega Model* (p. 100).

The Reynolds Stress-ω turbulence model, or SMC-ω model, is a Reynolds Stress model based on the ω-equation. The advantage of the ω-equation is that it allows for a more accurate near wall treatment with an automatic switch from a wall function to a low-Reynolds number formulation based on the grid spacing.

The modeled equations for the Reynolds stresses can be written as follows:

$$\frac{\partial \rho \overline{u_i u_j}}{\partial t} + \frac{\partial}{\partial x_k}\left(U_k \rho \overline{u_i u_j}\right) = P_{ij} - \frac{2}{3}\beta' \rho \omega k \delta_{ij} + \Phi_{ij} + P_{ij,b} + \frac{\partial}{\partial x_k}\left(\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial \overline{u_i u_j}}{\partial x_k}\right) \tag{2–107}$$

The production due to buoyancy is modeled in the same way as given in the *Equation 2–90* (p. 110) and *Equation 2–91* (p. 110). The default value of σ_ρ is 0.9 for the Boussinesq buoyancy model and 1.0 in the full buoyancy model based on density differences.

2.3.2.1. The Omega Reynolds Stress Model

The Omega Reynolds Stress Model uses the following equation for ω:

$$\frac{\partial\,(\rho\,\omega)}{\partial\,t}+\frac{\partial\,(U_k\,\rho\,\omega)}{\partial\,x_k}=\alpha\,\rho\,\frac{\omega}{k}\,P_k+P_{\omega b}-\beta\,\rho\,\omega^2+\frac{\partial}{\partial\,x_k}\left(\left(\mu+\frac{\mu_t}{\sigma}\right)\frac{\partial\,\omega}{\partial\,x_k}\right) \tag{2–108}$$

The following coefficients apply:

$$\begin{aligned}
&\sigma^*=2\\
&\sigma=2\\
&\beta=0.075\\
&\alpha=\frac{\beta}{\beta'}-\frac{\kappa^2}{\sigma\,(\beta')^{0.5}}=\frac{5}{9}
\end{aligned} \tag{2–109}$$

2.3.2.2. The BSL Reynolds Stress Model

The coefficients α and β of the ω-equation, as well as both the turbulent Prandtl numbers $\sigma\ *$ and σ, are blended between values from the two sets of constants, corresponding to the ω-based model constants and the ε-based model constants transformed to an ω-formulation:

$$\begin{aligned}
\frac{\partial\,(\rho\,\omega)}{\partial\,t}+\partial\left(U_k\,\rho\,\omega\right)= \quad &\alpha_3\,\frac{\omega}{k}\,P_k+P_{\omega b}-\beta_3\,\rho\,\omega^2+\frac{\partial}{\partial\,x_k}\left(\left(\mu+\frac{\mu_t}{\sigma_{\omega\,3}}\right)\frac{\partial\,\omega}{\partial\,x_k}\right)\\
&+\,(1-F_1)\;2\,\rho\,\frac{1}{\sigma_2\,\omega}\,\frac{\partial\,k}{\partial\,x_k}\,\frac{\partial\,\omega}{\partial\,x_k}
\end{aligned} \tag{2–110}$$

- Set 1 (SMC-ω zone):

$$\begin{aligned}
&\sigma_1^*=2\\
&\sigma_1=2\\
&\beta_1=0.075\\
&\alpha_1=\frac{\beta}{\beta'}-\frac{\kappa^2}{\sigma\,(\beta')^{0.5}}=0.553
\end{aligned} \tag{2–111}$$

  The value of β here corresponds to the $k\ -\ \omega$ model. For details, see *The Wilcox k-omega Model* (p. 98). The von Karman constant κ has a commonly used value of 0.41.

- Set 2 (SMC-ε zone):

$$\sigma_2^* = 1.0$$
$$\sigma_2 = 0.856$$
$$\beta_2 = 0.0828 \qquad \textbf{(2–112)}$$
$$\alpha_2 = \frac{\beta}{\beta'} - \frac{\kappa^2}{\sigma\,(\beta')^{0.5}} = 0.44$$

The blending of coefficients is done by smooth linear interpolation with the same weight function *F* as the one used in a cross-diffusion term of the ω-equation (*Equation 2–110* (p. 114)):

$$\varphi_3 = F \cdot \varphi_1 + (1 - F)\,\varphi_2 \qquad \textbf{(2–113)}$$

where $F = \tanh(\arg^4)$ with:

$$\arg = \min\left\{\max\left(\frac{\sqrt{k}}{\beta'\,\omega\,y}, \frac{500\,\upsilon}{y^2\,\omega}\right), \frac{4\,\rho\,k}{CD_{k\omega}\,\sigma_{k-\varepsilon}\,y^2}\right\} \qquad \textbf{(2–114)}$$

and

$$CD_{k\omega} = \max\left(2\,\rho\,\frac{1}{\sigma_{k-\varepsilon}\,\omega}\,\frac{\partial k}{\partial x_j}\,\frac{\partial \omega}{\partial x_j}, 10^{-10}\right) \qquad \textbf{(2–115)}$$

2.3.2.3. Pressure-Strain Correlation

The constitutive relation for the pressure-strain correlation is given by:

$$\begin{aligned}\Phi_{ij} = {} & \beta' C_1 \rho\omega\left(-\overline{u_i u_j} + \frac{2}{3}k\delta_{ij}\right) - \hat{a}\left(P_{ij} - \frac{2}{3}P\delta_{ij}\right) \\ & - \hat{\beta}\left(D_{ij} - \frac{2}{3}P\delta_{ij}\right) - \hat{\gamma}\rho k\left(S_{ij} - \frac{1}{3}S_{kk}\delta_{ij}\right)\end{aligned} \qquad \textbf{(2–116)}$$

The production tensor of Reynolds stresses is given by:

$$P_{ij} = -\overline{\rho u_i u_k}\frac{\partial U_j}{\partial x_k} - \rho\overline{u_j u_k}\frac{\partial U_i}{\partial x_k}; \qquad P = 0.5 P_{kk} \qquad \textbf{(2–117)}$$

The tensor D_{ij}, participating in the pressure-strain model *Equation 2–116* (p. 115), differs from the production tensor in the dot-product indices:

$$D_{ij} = -\rho \overline{u_i u_k} \frac{\partial U_k}{\partial x_j} - \rho \overline{u_j u_k} \frac{\partial U_k}{\partial x_i} \tag{2–118}$$

The turbulent viscosity in the diffusion terms of the balance equations (*Equation 2–107* (p. 113) and *Equation 2–108* (p. 114)) is calculated in the same way as in the Wilcox $k-\omega$ model. For details, see *The Wilcox k-omega Model* (p. 98).

$$\mu_T = \rho \frac{k}{\omega} \tag{2–119}$$

The coefficients for the model are:

$\beta' = 0.09$

$\hat{\alpha} = (8 + C_2)/11$

$\hat{\beta} = (8\ C_2 - 2)/11$

$\hat{\gamma} = (60\,C_2 - 4)/55$

$C_1 = 1.8$

$C_2 = 0.52$

2.3.2.4. Wall Boundary Condition

The $\mathrm{SMC}-\omega$ model is used in combination with the automatic wall treatment developed for the $k-\omega$ based models ($k-\omega$, BSL and SST). The formulation has been recalibrated to ensure a constant wall shear stress and heat transfer coefficient under variable near wall resolution.

2.3.3. Rotating Frame of Reference for Reynolds Stress Models

One of the advantages of Reynolds stress transport models, when compared to $k-\varepsilon$ and $k-\omega$ models, is their ability to simulate the additional anisotropy of the Reynolds stresses due to the Coriolis forces appearing in the rotating frame of reference.

The necessary additional source terms to account for the Coriolis forces have been implemented into ANSYS CFX for use with any of the available Reynolds stress transport models. These terms are described in a book by Wilcox [30], and in more detail in a paper by Launder, Tselepidakis and Younis [31].

If the flow equations are written in the frame of the coordinate system, which rotates relative to the steady inertial frame with a rate Ω^{rot} then one new source term G_{ij} has to be added directly to the right hand side of the transport equation for $\rho \overline{u_i u_j}$, *Equation 2–107* (p. 113):

$$G_{ij} = \rho \cdot \Omega_k^{rot} \cdot \left(\tau_{jm}\,\varepsilon_{ikm} + \tau_{im}\,\varepsilon_{jkm}\right) \quad \textbf{(2–120)}$$

where ε_{ijk} is a Levi-Chivita factor, equal to 1 if its indices {*i*,*j*,*k*} form an even permutation of {1,2,3}, equal to -1 for an odd permutation of {1,2,3}, and equal to 0 if any two indices are equal.

Besides the absolute velocity gradient tensor, participating in the production tensor *Equation 2–117* (p. 115) and in the model equation for the pressure-strain correlation *Equation 2–116* (p. 115), is written in the rotating frame as a sum of the strain rate tensor S_{ij}:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \quad \textbf{(2–121)}$$

and the vorticity tensor Ω_{ij}, which has an additional term due to system rotation:

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right) + \varepsilon_{ijk} \cdot \Omega_k^{rot} \quad \textbf{(2–122)}$$

This representation of the velocity gradient results in an apparent additional source term G_{ij} *Equation 2–120* (p. 117), coming from the production term ($-\mathrm{pP}_{ij}$). That is why in reference [31] the Coriolis source term G_{ij} differs from *Equation 2–120* (p. 117) by an additional factor of 2.

2.3.4. Explicit Algebraic Reynolds Stress Model

Explicit Algebraic Reynolds Stress Models (EARSM) represent an extension of the standard two-equation models. They are derived from the Reynolds stress transport equations and give a nonlinear relation between the Reynolds stresses and the mean strain-rate and vorticity tensors. Due to the higher order terms, many flow phenomena are included in the model without the need to solve transport equations. The EARSM allow an extension of the current ($k - \varepsilon$ and BSL) turbulence models to capture the following flow effects:

- Secondary flows
- Flows with streamline curvature and system rotation.

The implementation is based on the explicit algebraic Reynolds stress model of Wallin and Johansson [188]. Differences from the original formulation by Wallin and Johansson are explained in the following text. The current EARSM formulation can be used in CFX together with either the BSL $k - \omega$ or $k - \varepsilon$ model. It is recommended that the current EARSM formulation be used with the BSL model (BSL EARSM).

With EARSM, the Reynolds stresses are computed from the anisotropy tensor according to its definition (see *Equation 2–104* (p. 112)):

$$\overline{u_i u_j} = k\left(a_{ij} + \frac{2}{3}\delta_{ij}\right)$$

where the anisotropy tensor a_{ij} is searched as a solution of the following implicit algebraic matrix equation:

$$N\mathbf{a} = -A_1\mathbf{S} + (\mathbf{a\Omega} - \mathbf{\Omega a}) - A_2\left(\mathbf{aS} - \mathbf{Sa} - \frac{2}{3}tr\ \{\mathbf{aS}\}\right), \text{ with } \quad N = A_3 + A_4\,(P_k/\varepsilon) \quad \textbf{(2–123)}$$

The coefficients A_i in this matrix equation depend on the C_i-coefficients of the pressure-strain term in the underlying Reynolds stress transport model. Their values are selected here as A_1=1.245, A_2=0, A_3=1.8, A_4=2.25.

The values of A_2, A_3, and A_4 are the same as those used in the original model by Wallin and Johansson [188]. As for the value of A_1, it is increased from 1.2 to 1.245 in the course of calibrating EARSM for its use together with the BSL $k - \omega$ model.

S_{ij} and Ω_{ij} denote the non-dimensional strain-rate and vorticity tensors, respectively. They are defined as:

$$S_{ij} = \frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \quad \textbf{(2–124)}$$

$$\Omega_{ij} = \frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right) \quad \textbf{(2–125)}$$

where the time-scale τ is given by:

$$\tau = k/\varepsilon = 1/\left(C_\mu\omega\right), \quad C_\mu = 0.09 \quad \textbf{(2–126)}$$

In order to arrive at an explicit solution of the *Equation 2–123* (p. 118), the anisotropy tensor is expressed as a polynomial based on the strain rate and the vorticity tensors as follows:

$$\begin{aligned} a_{ij} &= \beta_1 S_{ij} + \beta_3\left(\Omega_{ik}\Omega_{kj} - \frac{1}{3}II_\Omega\delta_{ij}\right) + \beta_4\left(S_{ik}\Omega_{kj} - \Omega_{ik}S_{kj}\right) \\ &\quad + \beta_6\left(S_{ik}\Omega_{kl}\Omega_{lj} + \Omega_{ik}\Omega_{kl}S_{lj} - \frac{2}{3}IV\delta_{ij} - II_\Omega S_{ij}\right) \end{aligned} \quad \textbf{(2–127)}$$

The β-coefficients are evaluated to:

$$\beta_1 = -N/Q$$

$$\beta_3 = -12 \cdot IV / \left(N \cdot Q \cdot \left(2N^2 - II_\Omega\right)\right)$$

$$\beta_4 = -1/Q$$

$$\beta_6 = -6 \cdot N / \left(Q \cdot \left(2N^2 - II_\Omega \right) \right)$$

where the denominator Q is:

$$Q = \left(N^2 - 2II_\Omega \right) / A_1$$

The invariants, which appear in the formulation of the anisotropy tensor and the coefficients, are defined by:

$$II_S = S_{kl} S_{lk}$$

$$II_\Omega = \Omega_{kl} \Omega_{lk}$$

$$IV = S_{kl} \Omega_{lm} \Omega_{mk}$$

The model representation of the anisotropy tensor *Equation 2–127* (p. 118) and its coefficients β_i follows the original model by Wallin and Johansson [188] with two differences. First, the fourth order tensor polynomial contribution (the $\beta_9 \left(\mathbf{\Omega S \Omega}^2 - \mathbf{\Omega}^2 \mathbf{S \Omega} \right)$ term) is neglected in *Equation 2–127* (p. 118). Second, the tensor basis is slightly changed for convenience according to Apsley and Leschziner [219]. Although the tensor basis is changed, the model remains algebraically equivalent to the original model of Wallin and Johansson. The latter change results in correspondingly changed expressions for the coefficients β_i.

In three-dimensional flows, the equation to be solved for the function N is of sixth order and no explicit solution can be derived, whereas in two-dimensional mean flows the function N can be derived from a cubic equation, an analytic solution of which is recommended by Wallin and Johansson [188] also for three-dimensional cases:

$$N = \begin{cases} A_3/3 + \left(P_1 + \sqrt{P_2} \right)^{1/3} + \operatorname{sign}\left(P_1 - \sqrt{P_2} \right) \left| P_1 - \sqrt{P_2} \right|^{1/3} & \text{for} \quad P_2 \geq 0 \\ A_3/3 + 2 \left(P_1^2 - P_2 \right)^{1/6} \cos\left(\frac{1}{3} \arccos\left(\frac{P_1}{\sqrt{P_1^2 - P_2}} \right) \right) & \text{for} \quad P_2 < 0 \end{cases} \quad \textbf{(2–128)}$$

where

$$P_1 = \left(\frac{A_3^2}{27} + \frac{A_1 A_4}{6} II_S - \frac{2}{3} II_\Omega \right) \cdot A_3$$

$$P_2 = P_1^2 - \left(\frac{A_3^2}{9} + \frac{A_1 A_4}{3} II_S + \frac{2}{3} II_\Omega \right)^3$$

A diffusion correction, suggested by Wallin and Johansson [188], is also implemented in CFX as an option. This correction consists in the replacement of A_3 in the above written formulas for N, P_1 and P_2, through A_3':

$$A'_3 = A_3 + A_4 \cdot C_{Diff} \max\left(1 + \beta_1^{eq} II_S \, , \, 0\right)$$

where β_1^{eq} is the equilibrium value of β_1, that is a value of β_1 achieved under the source term equilibrium P_k=ε:

$$\beta_1^{eq} = -A_1 \frac{N^{eq}}{\left(N^{eq}\right)^2 - 2II_\Omega}, \quad N^{eq} = A_3 + A_4$$

For the parameter C_{Diff}, which scales the degree of the diffusion correction, Wallin and Johansson suggest a value of 2.2. In the current CFX implementation, it is set by default to zero, but this can be overridden.

In the original model by Wallin and Johansson [188], the diffusion terms in the transport equations for k and ε (or ω) were calculated using the effective eddy viscosity, $\mu_t^{eff} = C_\mu^{eff} k^2 / \varepsilon = \left(C_\mu^{eff} / C_\mu\right) \cdot k / \omega$, of EARSM, where $C_\mu^{eff} = -\beta_1 / 2$. The EARSM model, implemented in CFX, uses the standard eddy viscosity $\mu_t = C_\mu k^2 / \varepsilon = k / \omega$ for the diffusion terms. This model change helps to avoid the problems with the asymptotic behavior at the boundary layer edge, which were reported by Hellsten [190].

For the underlying $k - \varepsilon$ or BSL $k - \omega$ model, the standard coefficients are used.

2.3.4.1. Streamline Curvature and System Rotation

In order to account for effects of streamline curvature, the non-dimensional vorticity tensor is extended in the following way ([189], [190]):

$$\Omega_{ij} = \frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - C_{scale} \cdot \frac{\tau}{A_0} \Omega_{ij}^{cc} \quad \textbf{(2–129)}$$

where the contribution of the curvature correction based on the work of Wallin and Johansson [189] and Spalart and Shur [191] is given in the following way:

$$\Omega_{ij}^{cc} = -\varepsilon_{ijk} \omega_k^{S-S} \quad \textbf{(2–130)}$$

where

$$\omega_k^{S-S} = \frac{S_{pl} \dot{S}'_{lq} \varepsilon_{pqk}}{2II_S}$$

and

$$\dot{S}' = \frac{DS_{ij}}{Dt} + \left(\varepsilon_{imn} S_{jn} + \varepsilon_{jmn} S_{in}\right) \Omega_m^{rot}$$

In this formulation ε_{ijk} represents the Levi-Chivita factor, which is equal to 0 if two or more indices are equal, 1 if the indices {i,j,k} form an even permutation of {1,2,3} and –1 if they form an odd permutation. Ω_m^{rot} are the components of the coordinate system rotation vector. The coefficient A_0 has to be calibrated. The scaling coefficient C_{scale} has been introduced similar to the 'Curvature Correction of Two-Equation Models' in order to allow the user to influence the effect of the curvature correction term

$$\frac{\tau}{A_0} \Omega_{ij}^{cc}$$

if needed for a specific flow. The default value of this coefficient is 1.

In a rotational frame of reference, where the coordinate system rotates relative to a steady inertial frame, the vorticity tensor reads:

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) + \varepsilon_{ijk} \Omega_k^{rot} \tag{2–131}$$

Because the anisotropy tensor is computed in terms of nondimensional strain-rate and vorticity tensors, the nondimensional form of this term becomes:

$$\Omega_{ij} = \frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) + \tau \varepsilon_{ijk} \Omega_k^{rot} \tag{2–132}$$

It should be noted that there are significant robustness issues with the streamline curvature model described in *Equation 2–130* (p. 120). This is because one has the second velocity derivative (that is, the gradient of the strain rate) divided by the first velocity derivative (that is, the strain rate). Consequently, numerical noise in the second velocity derivative is amplified when the strain rate is small (for example, in the free stream outside of a shear layer). Because the curvature corrected Reynolds stresses enter directly in the momentum equation, this can create a feedback loop that results in increasing spikes in the Reynolds stresses which are unphysical and can even cause the solver to diverge. The solution to this problem turned out to be to compute two sets of Reynolds stresses with the EARSM, those with and without the curvature term in *Equation 2–129* (p. 120). The Reynolds stresses without the curvature term are used in the momentum equations, while the Reynolds stresses with the curvature term are used in the assembly of the production term in the k and ω (or ε) equations. Consequently, the change in turbulence production due to curvature is still captured, however the feedback loop that caused the spikes in the Reynolds stresses is avoided because the curvature term does not enter directly into the momentum equations. There is probably some loss in accuracy due to this treatment. However this is preferable to a model, which is not robust and cannot be used in industrial applications. The A_0 constant in *Equation 2–129* (p. 120). has been calibrated to a value of -0.4.

2.4. ANSYS CFX Transition Model Formulation

The full transition model is based on two transport equations, one for the intermittency and one for the transition onset criteria in terms of momentum thickness Reynolds number. It is called 'Gamma

Theta Model' and is the recommended transition model for general-purpose applications. It uses a new empirical correlation (Langtry and Menter) which has been developed to cover standard bypass transition as well as flows in low free-stream turbulence environments. This built-in correlation has been extensively validated together with the SST turbulence model ([101], [102], [103]) for a wide range of transitional flows. The transition model can also be used with the BSL or SAS-SST turbulence models as well.

It should be noted that a few changes have been made to this model compared to original version (that is, the CFX-5.7 formulation, [101],[102]) in order to improve the transition predictions. These include:

1. An improved transition onset correlation that results in improved predictions for both natural and bypass transition.
2. A modification to the separation induced transition modification that prevents it from causing early transition near the separation point.
3. Some adjustments of the model coefficients in order to better account for flow history effects on the transition onset location.

You should use the new formulation [103], although the original version of the transition model (that is, the CFX-5.7 formulation) can be recovered by specifying the optional parameter "`Transition Model Revision = 0`" in the CCL in the following way:

```
FLUID MODELS:
  …
  TURBULENCE MODEL:
    Option = SST
    TRANSITIONAL TURBULENCE:
      Option = Gamma Theta Model
      Transition Model Revision = 0
      TRANSITION ONSET CORRELATION:
        Option = Langtry Menter
      END
    END
  END
  …
END
```

The transport equation for the intermittency, γ, reads:

$$\frac{\partial\,(\rho\,\gamma)}{\partial t}+\frac{\partial\left(\rho\,U_j\,\gamma\right)}{\partial x_j}=P_{\gamma 1}-E_{\gamma 1}+P_{\gamma 2}-E_{\gamma 2}+\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\gamma}\right)\frac{\partial \gamma}{\partial x_j}\right] \qquad \textbf{(2–133)}$$

The transition sources are defined as follows:

$$P_{\gamma 1}=2\,F_{\text{length}}\,\rho\,S\;\left[\gamma\,F_{\text{onset}}\right]^{c_{\gamma 3}}\;;\;E_{\gamma 1}=P_{\gamma 1}\,\gamma \qquad \textbf{(2–134)}$$

where S is the strain rate magnitude. F_{length} is an empirical correlation that controls the length of the transition region. The destruction/relaminarization sources are defined as follows:

$$P_{\gamma 2}=\left(2\,c_{\gamma 1}\right)\rho\,\Omega\,\gamma\,F_{\text{turb}}\;;\;E_{\gamma 2}=c_{\gamma 2}\,P_{\gamma 2}\,\gamma \qquad \textbf{(2–135)}$$

where Ω is the magnitude of vorticity rate. The transition onset is controlled by the following functions:

$$Re_v = \frac{\rho\, y^2\, S}{\mu}\ ;\ R_T = \frac{\rho\, k}{\mu\, \omega} \tag{2–136}$$

$$\begin{aligned} F_{\text{onset }1} &= \frac{Re_v}{2.193 \cdot Re_{\theta c}} \\ F_{\text{onset }2} &= \min\left(\max\left(F_{\text{onset }1}, F^4_{\text{onset }1}\right), 2.0\right) \end{aligned} \tag{2–137}$$

$$\begin{aligned} F_{\text{onset }3} &= \max\left(1 - \left(\frac{R_T}{2.5}\right)^3, 0\right) \\ F_{\text{onset}} &= \max\left(F_{\text{onset }2} - F_{\text{onset }3}, 0\right) \\ F_{\text{turb}} &= e^{-\left(\frac{R_T}{4}\right)^4} \end{aligned} \tag{2–138}$$

$Re_{\theta c}$ is the critical Reynolds number where the intermittency first starts to increase in the boundary layer. This occurs upstream of the transition Reynolds number, $\widetilde{Re}_{\theta t}$, and the difference between the two must be obtained from an empirical correlation. Both the F_{length} and $Re_{\theta c}$ correlations are functions of $\widetilde{Re}_{\theta t}$.

The constants for the intermittency equation are:

$$\begin{aligned} c_{y1} &= 0.03 \\ c_{y2} &= 50 \\ c_{y3} &= 0.5 \\ \sigma_y &= 1.0 \end{aligned} \tag{2–139}$$

The modification for separation-induced transition is:

$$\begin{aligned} \gamma_{\text{sep}} &= \min\left(2 \cdot \max\left[\left(\frac{Re_v}{3.235\, Re_{\theta c}}\right) - 1, 0\right] F_{\text{reattach}}, 2\right) F_{\theta t} \\ F_{\text{reattach}} &= e^{-\left(\frac{R_T}{20}\right)^4} \\ \gamma_{eff} &= \max\left(\gamma, \gamma_{\text{sep}}\right) \end{aligned} \tag{2–140}$$

The model constants in *Equation 2–140* (p. 123) have been adjusted from those of Menter et al. [101] in order to improve the predictions of separated flow transition. The main difference is that the constant that controls the relation between Re_v and $Re_{\theta c}$ was changed from 2.193, its value for a Blasius boundary layer, to 3.235, the value at a separation point where the shape factor is 3.5 (see, for example Figure 2 in Menter et al. [101]). The boundary condition for γ at a wall is zero normal flux while for an inlet γ is equal to 1.0.

The transport equation for the transition momentum thickness Reynolds number, $\tilde{R}e_{\theta t}$, reads:

$$\frac{\partial\ (\rho \tilde{R}e_{\theta t})}{\partial t} + \frac{\partial \left(\rho U_j \tilde{R}e_{\theta t}\right)}{\partial x_j} = P_{\theta t} + \frac{\partial}{\partial x_j}\left[\sigma_{\theta t}\left(\left(\mu + \mu_t\right) \frac{\partial \tilde{R}e_{\theta t}}{\partial x_j}\right)\right] \quad \textbf{(2–141)}$$

The source term is defined as follows:

$$P_{\theta t} = c_{\theta t} \frac{\rho}{t} (Re_{\theta t} - \tilde{R}e_{\theta t})\ (1.0 - F_{\theta t})\ ;\ t = \frac{500\,\mu}{\rho\, U^2} \quad \textbf{(2–142)}$$

$$F_{\theta t} = \min\left(\max\left(F_{\text{wake}} \cdot e^{-\left(\frac{y}{\delta}\right)^4}, 1.0 - \left(\frac{\gamma - 1/50}{1.0 - 1/50}\right)^2\right), 1.0\right) \quad \textbf{(2–143)}$$

$$\theta_{BL} = \frac{\tilde{R}e_{\theta t}\,\mu}{\rho\, U}\ ;\ \delta_{BL} = \frac{15}{2}\theta_{BL}\ ;\ \delta = \frac{50\,\Omega\, y}{U} \cdot \delta_{BL} \quad \textbf{(2–144)}$$

$$Re_{\omega} = \frac{\rho\,\omega\, y^2}{\mu}\ ;\ F_{\text{wake}} = e^{-\left(\frac{Re_{\omega}}{1\times 10^5}\right)^2} \quad \textbf{(2–145)}$$

The model constants for the $\tilde{R}e_{\theta t}$ equation are:

$$c_{\theta t} = 0.03\ ;\ \sigma_{\theta t} = 2.0 \quad \textbf{(2–146)}$$

The boundary condition for $\tilde{R}e_{\theta t}$ at a wall is zero flux. The boundary condition for $\tilde{R}e_{\theta t}$ at an inlet should be calculated from the empirical correlation based on the inlet turbulence intensity.

The model contains three empirical correlations. $Re_{\theta t}$ is the transition onset as observed in experiments. This has been modified from Menter et al. [101] in order to improve the predictions for natural transition. It is used in *Equation 2–142* (p. 124). F_{length} is the length of the transition zone and goes into *Equation 2–134* (p. 122). $Re_{\theta c}$ is the point where the model is activated in order to match both $Re_{\theta t}$ and F_{length}; it goes into *Equation 2–137* (p. 123). At present, these empirical correlations are proprietary and are not given in this manual.

$$Re_{\theta t} = f\,(Tu, \lambda)\ ;\ F_{\text{length}} = f\,(\tilde{R}e_{\theta t})\,, Re_{\theta c} = f\,(\tilde{R}e_{\theta t}) \tag{2–147}$$

The first empirical correlation is a function of the local turbulence intensity, $T\ u$, and the Thwaites pressure gradient coefficient λ_{θ} defined as:

$$\lambda_{\theta} = \left(\frac{\theta^2}{\nu} \right) \frac{dU}{ds} \tag{2–148}$$

where dU/ds is the acceleration in the streamwise direction.

The transition model interacts with the SST turbulence model, as follows:

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}\left(\rho u_j k\right) = \tilde{P}_k - \tilde{D}_k + \frac{\partial}{\partial x_j}\left(\left(\mu + \sigma_k \mu_t\right) \frac{\partial k}{\partial x_j} \right) \tag{2–149}$$

$$\tilde{P}_k = \gamma_{eff} P_k\ ;\ \tilde{D}_k = \min\left(\max\left(\gamma_{eff}, 0.1\right), 1.0 \right) D_k \tag{2–150}$$

$$R_y = \frac{\rho y \sqrt{k}}{\mu}\ ;\ F_3 = e^{-\left(\frac{R_y}{120}\right)^8}\ ;\ F_1 = max\left(F_{1\,\text{orig}}, F_3\right) \tag{2–151}$$

where P_k and D_k are the original production and destruction terms for the SST model and $F_{1\ \text{orig}}$ is the original SST blending function. Note that the production term in the ω-equation is not modified. The rationale behind the above model formulation is given in detail in Menter et al. [101].

In order to capture the laminar and transitional boundary layers correctly, the grid must have a y^+ of approximately one. If the y^+ is too large (that is, > 5) then the transition onset location moves upstream with increasing y^+. You should use the High Resolution discretization scheme (which is a bounded second-order upwind biased discretization) for the mean flow, turbulence and transition equations.

Note

The default production limiter for the turbulence equations is the ‘Kato-Launder’ formulation when the transition model is used.

As outlined in CFX Transition Model in the *CFX-Solver Modeling Guide*, two reduced variants of the transition model are available beside the two-equation **Gamma Theta** transition model. A zero equation model, where the user can prescribe the intermittency directly as a CEL expression **Specified Intermittency**, and a one equation model, which solves only the intermittency equation using a user specified value of the transition onset Reynolds number **Gamma Model**.

2.4.1. Transition and Rough Walls

If the two-equation transition model is used together with rough walls, then an equivalent sand-grain roughness height must be specified at the wall and in addition the option **Roughness Correlation** must be selected on the **Fluid Models** tab of the **Domain** details view. Further information on the rough wall modifications can be found in *Treatment of Rough Walls* (p. 144).

2.5. Large Eddy Simulation Theory

This section outlines the theoretical details of the LES model in ANSYS CFX. Additional information on setting up an LES simulation is available, as well as modeling advice. For details, see The Large Eddy Simulation Model (LES) in the *CFX-Solver Modeling Guide*.

The rationale behind the large-eddy simulation technique is a separation between large and small scales. The governing equations for LES are obtained by filtering the time-dependent Navier-Stokes equations in the physical space. The filtering process effectively filters out the eddies whose scales are smaller than the filter width or grid spacing used in the computations. The resulting equations thus govern the dynamics of the large eddies.

A filtered variable is denoted in the following by an overbar and is defined by

$$\overline{\Phi}(\boldsymbol{x}) = \int_D \Phi(\boldsymbol{x}')\, G(\boldsymbol{x};\ \boldsymbol{x}')\, d\boldsymbol{x}' \tag{2–152}$$

where D is the fluid domain, and G is the filter function that determines the scale of the resolved eddies.

The unresolved part of a quantity Φ is defined by

$$\Phi' = \Phi - \overline{\Phi} \tag{2–153}$$

It should be noted that the filtered fluctuations are not zero:

$$\overline{\Phi'} \neq 0 \tag{2–154}$$

The discretization of the spatial domain into finite control volumes implicitly provides the filtering operation:

$$\overline{\Phi}(\boldsymbol{x}) = \frac{1}{V}\int_V \Phi(\boldsymbol{x}')\, d\boldsymbol{x}' \quad , \quad \boldsymbol{x}' \in V$$

where V is the control volume. The filter function $G(x;\ x')$ implied here is then

$$G(\boldsymbol{x};\ \boldsymbol{x'}) = \begin{cases} 1/V, \boldsymbol{x'} \in V \\ 0 \text{otherwise} \end{cases} \tag{2–155}$$

Filtering the Navier-Stokes equations leads to additional unknown quantities. In the following the theory will be outlined for the incompressible equations. The filtered incompressible momentum equation can be written in the following way:

$$\frac{\partial \overline{U}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{U}_i \overline{U}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left[v\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right)\right] - \frac{\partial \tau_{ij}}{\partial x_j} \tag{2–156}$$

where τ_{ij} denotes the subgrid-scale stress. It includes the effect of the small scales and is defined by

$$\tau_{ij} = \overline{U_i U_j} - \overline{U}_i \overline{U}_j \tag{2–157}$$

The large scale turbulent flow is solved directly and the influence of the small scales is taken into account by appropriate subgrid-scale (SGS) models. In ANSYS CFX an eddy viscosity approach is used which relates the subgrid-scale stresses τ_{ij} to the large-scale strain rate tensor $\overline{S}_{ij}$ in the following way:

$$-\left(\tau_{ij} - \frac{\delta_{ij}}{3}\tau_{kk}\right) = 2 v_{sgs} \overline{S}_{ij} \quad , \quad \overline{S}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right) \tag{2–158}$$

Unlike in RANS modelling, where the eddy viscosity v_{sgs} represents all turbulent scales, the subgrid-scale viscosity only represents the small scales.

It should be noted that the isotropic part of τ_{kk} is not modeled, but added to the filtered static pressure.

Three models are available to provide the subgrid-scale (SGS) viscosity v_{sgs}. You should use the wall-adapted local eddy-viscosity model by Nicoud and Ducros [200](LES WALE model) as a first choice. This is an algebraic model like the Smagorinsky model, but overcomes some known deficiencies of the Smagorinsky model: the WALE model produces almost no eddy-viscosity in wall-bounded laminar flows and is therefore capable to reproduce the laminar to turbulent transition. Furthermore, the WALE model has been designed to return the correct wall-asymptotic y^{+3}-variation of the SGS viscosity and needs no damping functions.

In addition to the WALE model, the Smagorinsky model [34] and the Dynamic Smagorinsky-Lilly model (Germano et al. [198], Lilly [199]) are available. The Dynamic Smagorinsky-Lilly model is based on the Germano-identity and uses information contained in the resolved turbulent velocity field to evaluate the model coefficient, in order to overcome the deficiencies of the Smagorinsky model. The model coefficient is no longer a constant value and adjusts automatically to the flow type. However this method needs explicit (secondary) filtering and is therefore more time consuming than an algebraic model.

2.5.1. Subgrid-Scale Models

2.5.1.1. Smagorinsky Model

The Smagorinsky model [34] is an algebraic model for the SGS viscosity $\mathbf{v}_{SGS}$. Based on dimensional analysis, the SGS viscosity can be expressed as:

$$v_{SGS} \propto l\, q_{SGS} \tag{2–159}$$

where l is the length scale of the unresolved motion (usually the grid size $\Delta = (\mathrm{Vol})^{1/3}$) and q_{SGS} is the velocity of the unresolved motion.

Based on an analogy to the Prandtl mixing length model, the velocity scale is related to the gradients of the filtered velocity:

$$q_{SGS} = \Delta |\overline{S}| \quad \text{where} \quad |\overline{S}| = \left(2\, \overline{S}_{ij}\, \overline{S}_{ij}\right)^{1/2} \tag{2–160}$$

This yields the Smagorinsky model [34] for the subgrid-scale (SGS) viscosity:

$$v_{SGS} = (C_S\, \Delta)^2\, |\overline{S}| \tag{2–161}$$

with C_S the Smagorinsky constant. The value of the Smagorinsky constant for isotropic turbulence with inertial range spectrum:

$$E(k) = C_k\, \varepsilon^{2/3}\, k^{-5/3} \tag{2–162}$$

is:

$$C_S = \frac{1}{\pi}\left(\frac{2}{3\, C_k}\right)^{3/4} = 0.18 \tag{2–163}$$

For practical calculations, the value of C_S is changed depending on the type of flow and mesh resolution. Its value is found to vary between a value of 0.065 (channel flows) and 0.25. Often a value of 0.1 is used and has been found to yield the best results for a wide range of flows. This is also the default value of C_S.

The coefficient C_S is, therefore, not a universal constant and this is the most serious shortcoming of this model. Furthermore, damping functions are needed close to walls.

2.5.1.1.1. Wall Damping

Close to walls, the turbulent viscosity can be damped using a combination of a mixing length minimum function, and a viscosity damping function f_μ:

$$\nu_{sgs} = \min\left(l_{mix}, f_\mu C_S \Delta\right)^2 \sqrt{2S_{ij}S_{ij}} \quad \textbf{(2–164)}$$

with:

$$l_{\text{mix}} = \kappa \cdot y_{\text{wall}} \quad \textbf{(2–165)}$$

C_S and κ are constants which you can set; their default values are 0.1 and 0.4 respectively.

By default, the damping function f_μ is 1.0. A Van Driest and a Piomelli like damping can be specified by the user. For the Van Driest case, the damping function is:

$$f_\mu = 1 - \exp(-\tilde{y}/A) \quad \textbf{(2–166)}$$

with A = 25. For the Piomelli case it is:

$$f_\mu = \sqrt{1 - \exp\left[(-\tilde{y}/A)^3\right]} \quad \textbf{(2–167)}$$

with A = 25. The normalized wall distance:

$$\tilde{y} = (y \cdot \tilde{u})/\boldsymbol{\nu} \quad \textbf{(2–168)}$$

is defined as a function of the calculated wall distance y, kinematic viscosity ν, and local velocity scale $\tilde{u}$.

The Van Driest or Piomelli wall damping can be switched on when the LES turbulence model is selected. The damping factor A is defaulted to 25.0.

2.5.1.2. WALE Model

The wall-adapted local eddy-viscosity model (Nicoud and Ducros [200]) is formulated locally and uses the following equation to compute the eddy-viscosity:

$$\nu_{sgs} = (C_w \Delta)^2 \frac{\left(S_{ij}^d S_{ij}^d\right)^{3/2}}{\left(\overline{S}_{ij}\overline{S}_{ij}\right)^{5/2} + \left(S_{ij}^d S_{ij}^d\right)^{5/4}} \quad \textbf{(2–169)}$$

where S_{ij}^d denotes the traceless symmetric part of the square of the velocity gradient tensor:

$$S^d_{ij} = \frac{1}{2}\left(\overline{g}^{\,2}_{ij} + \overline{g}^{\,2}_{ji}\right) - \frac{1}{3}\delta_{ij}\overline{g}^{\,2}_{kk} \tag{2–170}$$

and where $\overline{g}^{\,2}_{ij} = \overline{g}_{ik}\overline{g}_{kj}$, $\overline{g}_{ij} = \partial\,\overline{u}_i / \partial\,x_j$ and δ_{ij} is the Kronecker symbol. The tensor S^d_{ij} can be rewritten in terms of the strain-rate and vorticity tensors in the following way:

$$S^d_{ij} = \overline{S}_{ik}\overline{S}_{kj} + \overline{\Omega}_{ik}\overline{\Omega}_{kj} - \frac{1}{3}\delta_{ij}\left(\overline{S}_{mn}\overline{S}_{mn} - \overline{\Omega}_{mn}\overline{\Omega}_{mn}\right) \tag{2–171}$$

where the vorticity tensor $\overline{\Omega}_{ij}$ is given by:

$$\overline{\Omega}_{ij} = \frac{1}{2}\left(\frac{\partial\,\overline{U}_i}{\partial\,x_j} - \frac{\partial\,\overline{U}_j}{\partial\,x_i}\right) \tag{2–172}$$

The main advantages of the WALE model are the capability to reproduce the laminar to turbulent transition and the design of the model to return the correct wall-asymptotic y^{+3}-variation of the SGS viscosity. It offers therefore the same advantages as the Dynamic Smagorinsky-Lilly model, and at the same time does not require an explicit (secondary) filtering. The constant C_w has been calibrated using freely decaying isotropic homogeneous turbulence: the default value is 0.5 and is available in CFX-Pre.

2.5.1.3. Dynamic Smagorinsky-Lilly Model

Germano et al. [198] and subsequently Lilly [199]introduced a method for evaluating subgrid scale model coefficients using information contained in the resolved turbulent velocity field, in order to overcome the deficiencies of the Smagorinsky model. The model coefficient is no longer a constant value and adjusts automatically to the flow type. The basic idea behind the model is an algebraic identity, which relates subgrid scale stresses at two different filter widths. The filtering on the smaller filter width is given implicitly by the grid size, whereas the filtering on the larger filter width requires the use of an explicit filtering procedure. Because it is required that the filter works on unstructured grids, a volume-weighted averaging of the variables from neighboring element centers to the corresponding vertex is used.

The so-called Germano identity reads:

$$L_{ij} = T_{ij} - \left\{\tau_{ij}\right\} \tag{2–173}$$

where τ_{ij} represents the subgrid-scale (SGS) stress at scale Δ and T_{ij} the SGS stress at scale $\{\Delta\}$:

$$\tau_{ij} = \overline{U_i U_j} - \overline{U}_i \overline{U}_j \quad \textbf{(2–174)}$$

$$T_{ij} = \left\{\overline{U_i U_j}\right\} - \{\overline{U}_i\}\left\{\overline{U}_j\right\} \quad \textbf{(2–175)}$$

and $\{\ldots\}$ denotes secondary filtering of a quantity with $\{\Delta\} > \Delta$. The stress formulated by turbulent motions of scale intermediate between Δ and $\{\Delta\}$ reads:

$$L_{ij} = \left\{\overline{U}_i \overline{U}_j\right\} - \{\overline{U}_i\}\left\{\overline{U}_j\right\} \quad \textbf{(2–176)}$$

Although the Germano procedure can be applied with any SGS model, the Smagorinsky model has been used to compute the SGS stresses at the different filtering levels:

$$\tau_{ij} - \frac{\delta_{ij}}{3}\tau_{kk} = C_d\left(-2\Delta^2|\overline{S}_{ij}|\overline{S}_{ij}\right) = C_d m_{ij}^{sgs} \quad \textbf{(2–177)}$$

$$T_{ij} - \frac{\delta_{ij}}{3}T_{kk} = C_d\left(-2\{\Delta\}^2\left|\left\{\overline{S}_{ij}\right\}\right|\left\{\overline{S}_{ij}\right\}\right) = C_d m_{ij}^{test} \quad \textbf{(2–178)}$$

Using the Germano identity one obtains:

$$L_{ij}^a = L_{ij} - \frac{\delta_{ij}}{3}L_{kk} = C_d m_{ij}^{test} - \left\{C_d m_{ij}^{sgs}\right\} \quad \textbf{(2–179)}$$

The resulting system of *Equation 2–179* (p. 131) is over-determined and the coefficient C_d appears inside the secondary filter operation. In order to solve this system of equations, the error e_{ij} associated with the Smagorinsky model is defined in the following way:

$$e_{ij} = L_{ij}^a - C_d m_{ij}^{test} + \left\{C_d m_{ij}^{sgs}\right\} \quad \textbf{(2–180)}$$

Lilly [199] applied a least square approach to minimize the error. The coefficient C_d is taken out of the filtering procedure. This leads to

$$C_d = \frac{L_{ij}^a M_{ij}}{M_{ij} M_{ij}} \quad \textbf{(2–181)}$$

$$M_{ij} = m_{ij}^{test} - \left\{m_{ij}^{sgs}\right\} \quad \textbf{(2–182)}$$

Using the coefficient C_d, the eddy viscosity is obtained by:

$$v_{sgs} = C_d \Delta^2 |\overline{S}_{ij}| \tag{2–183}$$

The model coefficient C_d obtained using the dynamic Smagorinsky-Lilly model varies in time and space over a fairly wide range. To avoid numerical instability, a relaxation of C_d in time is applied and an upper and lower limit on the coefficient is imposed in the following way:

1. Lower bound in order to avoid negative viscosity:

$$C_d = \max\left(C_d, C_d^{\min}\right) \quad , C_d^{\min} = 0.0 \tag{2–184}$$

2. Upper bound:

$$C_d = \min\left(C_d, C_d^{\max}\right) \tag{2–185}$$

3. Relaxation in time:

$$\tilde{C}_d^n = \alpha C_d^n + (1-\alpha)\,\tilde{C}_d^{n-1} \quad \text{with} \quad \alpha = 0.01 \tag{2–186}$$

Freely decaying isotropic homogeneous turbulence has again been used to calibrate $C_d^{\max}$. The value of $C_d^{\max}$ should be in the range of 0.04 to 0.09. The default value is 0.04 and can be specified in CFX-Pre.

2.6. Detached Eddy Simulation Theory

This section outlines the theoretical details of the DES model in ANSYS CFX. Details on setting up a DES simulation and modeling advice are available. For details, see The Detached Eddy Simulation Model (DES) in the *CFX-Solver Modeling Guide*.

Experience has shown that the use of LES in boundary layer flows at high Re numbers is prohibitively expensive [57] and therefore not useful for many industrial flow simulations. On the other hand, turbulent structures can be resolved in massively separated regions, where the large turbulent scales are of the same dimension as the geometrical structure generating them (airfoil flaps, buildings, blunt trailing edges on turbine blades). DES is an attempt to combine elements of RANS and LES formulations in order to arrive at a hybrid formulation, where RANS is used inside attached and mildly separated boundary layers. Additionally, LES is applied in massively separated regions. While this approach offers many advantages, it is clearly not without problems, as the model has to identify the different regions automatically. DES models require a detailed understanding of the method and the grid generation requirements and should not be used as "black-box." You are advised to read the ANSYS CFX technical report [55] on the subject, which explains all details on grid generation requirements, zonal formulation and boundary conditions.

The ANSYS CFX version of the DES model is based on the SST formulation. The advantage of this combination is that the accurate prediction of turbulent boundary layers up to separation and in mildly separated regions carries over from the SST model. In addition, the SST model supports the formulation of a zonal DES formulation [56], which is less sensitive to grid resolution restrictions than the standard DES formulation, as proposed by Strelets [58]. Refer to the ANSYS CFX technical report [55] for details.

2.6.1. SST-DES Formulation Strelets et al.

The idea behind the DES model of Strelets [58] is to switch from the SST-RANS model to an LES model in regions where the turbulent length, L_t, predicted by the RANS model is larger than the local grid spacing. In this case, the length scale used in the computation of the dissipation rate in the equation for the turbulent kinetic energy is replaced by the local grid spacing, Δ.

$$\begin{aligned} &\varepsilon = \beta^* k \omega = k^{\frac{3}{2}}/L_t \rightarrow k^{\frac{3}{2}}/(C_{DES}\ \Delta) \quad \text{for} \quad (C_{DES}\ \Delta < L_t) \\ &\Delta = \max(\Delta_i) \ ; \mathrm{L}_t = (\sqrt{k})/\beta^* \omega \end{aligned} \tag{2–187}$$

The practical reason for choosing the maximum edge length in the DES formulation is that the model should return the RANS formulation in attached boundary layers. The maximum edge length is therefore the safest estimate to ensure that demand.

The DES modification of Strelets can be formulated as a multiplier to the destruction term in the k-equation:

$$\epsilon = \beta^* k \omega \rightarrow \beta^* k \omega \cdot F_{DES} \quad \text{with} \quad F_{DES} = \max\left(\frac{L_t}{C_{DES}\ \Delta}, 1\right) \tag{2–188}$$

with C_{DES} equal to 0.61, as the limiter should only be active in the $k - \varepsilon$ model region. The numerical formulation is also switched between an upwind biased and a central difference scheme in the RANS and DES regions respectively.

2.6.2. Zonal SST-DES Formulation in ANSYS CFX

The main practical problem with the DES formulation (both for the Spalart Allmaras and the standard SST-DES model) is that there is no mechanism for preventing the limiter from becoming active in the attached portion of the boundary layer. This will happen in regions where the local surface grid spacing is less than the boundary layer thickness $\Delta_s < c\ \delta$ with c of the order of one. In this case the flow can separate as a result of the grid spacing (grid-induced separation), which is undesirable. In order to reduce this risk, ANSYS CFX offers a zonal formulation of the DES formulation, based on the blending functions of the SST model [56]. The blending functions are functions of the wall distance. For details, see *Wall and Boundary Distance Formulation* (p. 82).

$$F_{DES-CFX} = \max\left(\frac{L_t}{C_{DES}\ \Delta}(1 - F_{SST}), 1\right) \ \text{with} F_{SST} = 0, F_1, F_2 \tag{2–189}$$

In case F_{SST} is set to 0, the Strelets model is recovered. The default option is $F_{SST} = F_2$, which offers the highest level of protection against grid-induced separation, but might also be less responsive in LES regions. For details, refer to the ANSYS CFX technical report [55].

2.6.3. Discretization of the Advection Terms

Following Strelets [58], the zonal DES model switches from a second-order upwind scheme in the RANS region to a central difference scheme in the LES region. The finite volume method approximates the advection term in the transport equation for a generic variable ϕ as:

$$\nabla \cdot \rho U \phi \approx \frac{1}{\Omega_{\mathrm{CV}}} \sum_{ip} \dot{m}_{ip} \phi_{ip} \tag{2–190}$$

where Ω_{CV} is the control volume value, the index ip denotes the integration points on the control volume faces, $\dot{m}_{ip}$ and ϕ_{ip} are the mass flow and the transported variable value estimated at the integration point ip. The value of ϕ_{ip} is obtained by interpolating from the surrounding grid nodes according to the selected discretization scheme. Blending between the upwind-biased scheme and the central scheme is achieved by combining the corresponding interpolation values $\phi_{ip,U}$ and $\phi_{ip,C}$ using the blending function σ:

$$\phi_{ip} = \sigma \cdot \phi_{ip,U} + (1-\sigma) \cdot \phi_{ip,C} \tag{2–191}$$

A specific form of the blending function σ is taken from Strelets [58] with only minor changes:

$$\sigma = \sigma_{\max} \cdot \tanh\left(A^{\mathrm{C_{H1}}}\right) \tag{2–192}$$

$$\begin{aligned} A &= C_{\mathrm{H2}} \cdot \max\left(\frac{C_{\mathrm{DES}}\,\Delta_{\max}}{L_{\mathrm{turb}}\,g} - 0.5, 0\right) \\ L_{\mathrm{turb}} &= \frac{\sqrt{k}}{c_\mu \omega} \\ g &= \max\left(\tanh\left(B^4\right), 10^{-10}\right) \end{aligned} \tag{2–193}$$

$$B = C_{\mathrm{H3}} \cdot \frac{\Omega \max(\Omega, S)}{\max\left(\frac{S^2+\Omega^2}{2,} 10^{-10}\right)} \tag{2–194}$$

with $\Delta_{\max}$ being the maximum neighboring grid edge, and the same constant values as in Strelets [58]:

$$\sigma_{max} = 1, C_{H1} = 3, C_{H2} = 1, C_{H3} = 2 \tag{2–195}$$

Note

For the SST-DES model, the blending function σ is additionally limited in order to prevent the grid-induced separation:

$$\sigma_{SST-DES} = \max\left(\sigma, BF1 \cdot F1, BF2 \cdot F2\right) \tag{2–196}$$

where F1 and F2 are the first and second SST blending function F_1 and F_2 ([129]), respectively. BF1 and BF2 are blend factors and are used in order to select one of the two blending functions. The default and the recommendation is to use the second SST blending function F2, therefore the corresponding default values are BF1 = 0 and BF2 = 1.

A limiter based on the Courant-number has been developed in order to avoid oscillations due to the central difference scheme, which can occur for medium and high Courant-numbers (this limiter is not given in the original formulation of Strelets [58]). The blending function σ is limited in the following way:

$$\sigma^{\lim} = \max\left[\sigma, 1 - \min\left(\frac{CFL^{\max}}{CFL}, 1\right)^{\mathrm{CFLEXP}}\right] \tag{2–197}$$

The default values of $CFL^{\max}$ and CFLEXP are 5.0 and 1.0, respectively.

2.6.4. Boundary Conditions

For LES simulations, unsteady fluctuations have to be specified in most cases at the inlet boundaries. This greatly complicates the use of LES in industrial flows, as the details of these fluctuations are generally not known. In DES simulations, it is in most cases possible to specify the same boundary conditions as for the SST model. This is particularly true if the default setting of the zonal model are used ([55]).

2.7. Scale-Adaptive Simulation Theory

The Scale-Adaptive Simulation (SAS) is an improved URANS formulation, which allows the resolution of the turbulent spectrum in unstable flow conditions. The SAS concept is based on the introduction of the von Karman length-scale into the turbulence scale equation. The information provided by the von Karman length-scale allows SAS models to dynamically adjust to resolved structures in a URANS simulation, which results in a LES-like behavior in unsteady regions of the flowfield. At the same time, the model provides standard RANS capabilities in stable flow regions.

In recent years the turbulent length-scale equation has been revisited by Menter and Egorov [130]. It was shown that the exact transport equation for the turbulent length-scale, as derived by Rotta [134], does actually introduce the second derivative of the velocity field and thereby L_{vK} into the turbulent scale equation. In Menter and Egorov [130], a two-equation turbulence model was presented using a $k - \nu_{\mathrm{t}}$ formulation, which can be operated in RANS and SAS mode. While the further development of this model is still ongoing, it was considered desirable to investigate if the SAS term in the $k - \nu_{\mathrm{t}}$

model could be transformed to existing two-equation models. The target two-equation model is the SST model and this leads to the formulation of the SST-SAS model.

The original version of the SST-SAS model (Menter and Egorov [131]) has undergone certain evolution and the latest model version has been presented in Egorov and Menter [197]. One model change is the use of the quadratic length scale ratio $(L/L_{vK})^2$ in the Equation 2–200 (p. 136) below, rather than the linear form of the original model version. The use of the quadratic length scale ratio is more consistent with the derivation of the model and no major differences to the original model version are expected. Another new model aspect is the explicitly calibrated high wave number damping to satisfy the requirement for an SAS model that a proper damping of the resolved turbulence at the high wave number end of the spectrum (resolution limit of the grid) must be provided. In the following the latest model version of the SST-SAS model (Egorov and Menter [197]) will be discussed, which is also the default version in ANSYS CFX.

The governing equations of the SST-SAS model differ from those of the SST RANS model [129] by the additional SAS source term Q_{SAS} in the transport equation for the turbulence eddy frequency ω:

$$\frac{\partial \rho k}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j k\right) = P_k - \rho c_\mu k\omega + \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] \tag{2–198}$$

$$\begin{aligned}&\frac{\partial \rho\omega}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho U_j\omega\right)\\ &= \alpha\frac{\omega}{k}P_k - \rho\beta\omega^2 + Q_{SAS} + \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right] + (1-F_1)\frac{2\rho}{\sigma_{\omega 2}}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}\end{aligned} \tag{2–199}$$

where $\sigma_{\omega 2}$ is the σ_ω value for the $k-\varepsilon$ regime of the SST model.

The additional source term Q_{SAS} reads for the latest model version Egorov and Menter [197]:

$$Q_{SAS} = \max\left[\rho\zeta_2\kappa S^2\left(\frac{L}{L_{vK}}\right)^2 - C\cdot\frac{2\rho k}{\sigma_\Phi}\max\left(\frac{1}{\omega^2}\frac{\partial \omega}{\partial x_j}\frac{\partial \omega}{\partial x_j}, \frac{1}{k^2}\frac{\partial k}{\partial x_j}\frac{\partial k}{\partial x_j}\right), 0\right] \tag{2–200}$$

This SAS source term originates from a term

$$U''\int_{-\infty}^{\infty} r_y\cdot\overline{u'(y)\,v'\left(y+r_y\right)}\,dr_y$$

in Rotta's transport equation for the correlation-based length scale, see Menter and Egorov [130]. Because the integral is zero in homogeneous turbulence, it should in general be proportional to a measure related to inhomogeneity. The second velocity derivative U'' was selected as this measure to ensure that the integral alone is modeled to zero at a constant shear rate, thus leading to U''^2 and ultimately to $(L/L_{vk})^2$ in the SAS source term (*Equation 2–200* (p. 136)).

This model version (Egorov and Menter [197], `Model Version=2007`) is used as default. In order to recover the original model formulation (Menter and Egorov [131]), the parameter `Model Version` must be set to `2005` directly in the CCL:

```
FLUID MODELS:
  ...
  TURBULENCE MODEL:
    Model Version = 2005
    Option = SAS SST
  END
  ...
END
```

The model parameters in the SAS source term *Equation 2–200* (p. 136) are

$$\zeta_2 = 3.51\,, \quad \sigma_\Phi = /3\,, \quad C =$$

Here L is the length scale of the modeled turbulence

$$L = \sqrt{k} \,/\, \left(c_\mu^{1/4} \cdot \omega\right) \tag{2–201}$$

and the von Karman length scale L_{vK} given by

$$L_{vK} = \frac{\kappa S}{|U''|} \tag{2–202}$$

is a three-dimensional generalization of the classic boundary layer definition

$$L_{vK}^{BL} = \kappa U'(y) \,/\, U''(y)$$

The first velocity derivative $U'(y)$ is represented in L_{vK} by S, which is a scalar invariant of the strain rate tensor S_{ij}:

$$S = \sqrt{S_{ij} S_{ij}}\,, \quad S_{ij} = \frac{1}{-}\left[\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right] \tag{2–203}$$

Note, that the same S also directly participates in Q_{SAS} (*Equation 2–200* (p. 136)) and in the turbulence production term $P_k = \mu_t S$.

The second velocity derivative $U''(y)$ is generalized to 3-D using the magnitude of the velocity Laplacian:

$$|U''| = \sqrt{\sum_{(i)} \left(\frac{\partial U_i}{\partial x_j \partial x_j} \right)} \quad \textbf{(2–204)}$$

As a result, L and L_{vK} are both equal to (κy) in the logarithmic part of the boundary layer, where κ=0.41 is the von Karman constant.

Beside the use of the quadratic length scale ratio (L/L_{vK}), the latest model version provides for the direct control of the high wave number damping. Two formulations are available:

1. The first formulation is the default and is realized by a lower constraint on the L_{vK} value in the following way:

$$L_{vK} = \max\left(\frac{\kappa S}{|U''|}, C_S \sqrt{\frac{\kappa\zeta}{\left(\beta/c_\mu\right) - \alpha}} \cdot \Delta \right) \quad , \quad \Delta = \Omega_{CV}^{1/3} \quad \textbf{(2–205)}$$

This limiter is proportional to the grid cell size Δ, which is calculated as the cubic root of the control volume size Ω_{CV}. The purpose of this limiter is to control damping of the finest resolved turbulent fluctuations. The structure of the limiter is derived from analyzing the equilibrium eddy viscosity of the SST-SAS model. Assuming the source term equilibrium (balance between production and destruction of the kinetic energy of turbulence) in both transport equations, one can derive the following relation between the equilibrium eddy viscosity μ_t^{eq}, L_{vK} and S:

$$\mu_t^{eq} = \rho \cdot \left(\sqrt{\left(\left(\beta/c_\mu\right) - \alpha \right) / \left(\kappa\zeta\right)} \cdot L_{vK} \right) \cdot S \quad \textbf{(2–206)}$$

This formula has a similar structure as the subgrid scale eddy viscosity in the LES model by Smagorinsky:

$$\mu_t^{LES} = \rho \cdot (C_S \cdot \Delta) \cdot S$$

Therefore it is natural to adopt the Smagorinsky LES model as a reference, when formulating the high wave number damping limiter for the SST-SAS model. The limiter, imposed on the L_{vK} value, must prevent the SAS eddy viscosity from decreasing below the LES subgrid-scale eddy viscosity:

$$\mu_t^{eq} \ge \mu_t^{LES} \quad \textbf{(2–207)}$$

Substitution of μ_t^{eq} and μ_t^{LES} in condition above (*Equation 2–207* (p. 138)) results in the L_{vK} limiter value used in *Equation 2–205* (p. 138). Similar to LES, the high wave number damping is a cumulative effect of the numerical dissipation and the SGS eddy viscosity. The model parameter C_S has been calibrated using decaying isotropic turbulence. The default value of C_S is 0.11 which provides nearly the same energy spectrum as the Smagorinsky LES model.

2. The second formulation limits the eddy viscosity directly:

$$\mu_t = \max\left(\mu_t^{SAS}, \mu_t^{LES}\right) \quad \textbf{(2–208)}$$

The LES-WALE model is used for the calculation of the LES eddy viscosity μ_t^{LES} because this model is suitable for transitional flows and does not need wall damping functions. This limiter can be turned on by setting the expert parameter `limit sas eddy viscosity = t`. The limiter (*Equation 2–205* (p. 138)) is then automatically turned off. The default value for the WALE model coefficient is 0.5 and can be specified by the expert parameter `limit sas eddy viscosity coef=0.5`.

Similar to the DES formulation, the SAS model also benefits from a switch in the numerical treatment between the steady and the unsteady regions. In DES, this is achieved by a blending function as proposed by Strelets [58], which allows the use of a second order upwind scheme with the CFX-Solver in RANS regions and a second order central scheme in unsteady regions. The blending functions are based on several parameters, including the grid spacing and the ratio of vorticity and strain rate, as explained below.

2.7.1. Discretization of the Advection Terms

The discretization of the advection is the same as that for the SST-DES model, beside the fact that no RANS-shielding is performed for the SAS-SST model. For details, see *Discretization of the Advection Terms* (p. 134).

2.8. Modeling Flow Near the Wall

This section presents the mathematical details of how flow near to a no-slip wall is modeled in ANSYS CFX. An introduction to near-wall flow, modeling details and guidelines on using wall functions are presented. For details, see Modeling Flow Near the Wall in the *CFX-Solver Modeling Guide*.

2.8.1. Mathematical Formulation

The wall-function approach in ANSYS CFX is an extension of the method of Launder and Spalding [13]. In the log-law region, the near wall tangential velocity is related to the wall-shear-stress, τ_ω, by means of a logarithmic relation.

In the wall-function approach, the viscosity affected sublayer region is bridged by employing empirical formulas to provide near-wall boundary conditions for the mean flow and turbulence transport equations. These formulas connect the wall conditions (for example, the wall-shear-stress) to the dependent variables at the near-wall mesh node which is presumed to lie in the fully-turbulent region of the boundary layer.

The logarithmic relation for the near wall velocity is given by:

$$u^+ = \frac{U_t}{u_\tau} = \frac{1}{\kappa}\ln\left(y^+\right) + C \quad \textbf{(2–209)}$$

where:

$$y^{+} = \frac{\rho \ \Delta y \ u_{\tau}}{\mu} \tag{2–210}$$

$$u_{\tau} = \left(\frac{\tau_{\omega}}{\rho} \right)^{1/} \tag{2–211}$$

u^{+} is the near wall velocity, u_{τ} is the friction velocity, U_t is the known velocity tangent to the wall at a distance of Δy from the wall, y^{+} is the dimensionless distance from the wall, τ_{ω} is the wall shear stress, κ is the von Karman constant and C is a log-layer constant depending on wall roughness (natural logarithms are used).

A definition of Δy in the different wall formulations is available in *Solver Yplus and Yplus* (p. 142).

2.8.1.1. Scalable Wall Functions

Equation 2–209 (p. 139) has the problem that it becomes singular at separation points where the near wall velocity, U_t, approaches zero. In the logarithmic region, an alternative velocity scale, u^* can be used instead of u_{τ}:

$$u^{*} = C_{\mu}^{1/4} k^{1/} \tag{2–212}$$

This scale has the useful property that it does not go to zero if U_t goes to zero. Based on this definition, the following explicit equation for u_{τ} can be obtained:

$$u_{\tau} = \frac{U_t}{\frac{1}{\kappa} \ln(y^{*}) + C} \tag{2–213}$$

The absolute value of the wall shear stress τ_{ω}, is then obtained from:

$$\tau_{\omega} = \rho \, u^{*} \, u_{\tau} \tag{2–214}$$

where:

$$y^{*} = (\rho \, u^{*} \, \Delta y) / \mu \tag{2–215}$$

and u^{*} is as defined earlier.

One of the major drawbacks of the wall-function approach is that the predictions depend on the location of the point nearest to the wall and are sensitive to the near-wall meshing; refining the mesh does not necessarily give a unique solution of increasing accuracy (Grotjans and Menter [10]). The problem of inconsistencies in the wall-function, in the case of fine meshes, can be overcome with the use of the **Scalable Wall Function** formulation developed by ANSYS CFX. It can be applied on arbitrarily fine meshes and allows you to perform a consistent mesh refinement independent of the Reynolds number of the application.

The basic idea behind the scalable wall-function approach is to limit the y^* value used in the logarithmic formulation by a lower value of $\tilde{y}^* = \max\,(y^*, 11.06)$ where 11.06 is the value of y^* at the intersection between the logarithmic and the linear near wall profile. The computed $\tilde{y}^*$ is therefore not allowed to fall below this limit. Therefore, all mesh points are outside the viscous sublayer and all fine mesh inconsistencies are avoided.

The boundary condition for the dissipation rate, ε, is then given by the following relation which is valid in the logarithmic region:

$$\varepsilon = \frac{\rho u^*}{\tilde{y}^* \mu}\frac{C_\mu^{3/4}}{\kappa}k^{3/} \tag{2–216}$$

It is important to note the following points:

- To fully resolve the boundary layer, you should put at least 10 nodes into the boundary layer.
- Do not use Standard Wall Functions unless required for backwards compatibility.
- The upper limit for y^+ is a function of the device Reynolds number. For example, a large ship may have a Reynolds number of 10^9 and y^+ can safely go to values much greater than 1000. For lower Reynolds numbers (for example, a small pump), the entire boundary layer might only extend to around $y^+ = 300$. In this case, a fine near wall spacing is required to ensure a sufficient number of nodes in the boundary layer.

If the results deviate greatly from these ranges, the mesh at the designated Wall boundaries will require modification, unless wall shear stress and heat transfer are not important in the simulation.

In most turbulent flows the turbulence kinetic energy is not completely zero and the definition for u^* given in *Equation 2–212* (p. 140) will give proper results for most cases. In flows with low free stream turbulence intensity, however, the turbulence kinetic energy can be very small and lead to vanishing u^* and therefore also to vanishing wall shear stress, If this situation happens, a lower limiter can be applied to u^* by setting the expert parameter `'ustar limiter = t'`. u^* will then be calculated using the following relation:

$$u^* = \max\left(C_\mu^{1/4}\sqrt{k}\,, Coef\,\cdot U_t/\tilde{y}^*\right) \tag{2–217}$$

The coefficient used in this relation can be changed by setting the expert parameter `'ustar limiter coef '`. The default value of this parameter is 0.01.

2.8.1.2. Solver Yplus and Yplus

In the solver output, there are two arrays for the near wall y^+ spacing. The definition for the Yplus variable that appears in the post processor is given by the standard definition of y^+ generally used in CFD:

$$y^+ = \frac{\sqrt{\tau_\omega/\rho} \cdot \Delta n}{v} \tag{2–218}$$

where Δn is the distance between the first and second grid points off the wall.

In addition, a second variable, Solver Yplus, is available which contains the y^+ used in the logarithmic profile by the solver. It depends on the type of wall treatment used, which can be one of three different treatments in ANSYS CFX. They are based on different distance definitions and velocity scales. This has partly historic reasons, but is mainly motivated by the desire to achieve an optimum performance in terms of accuracy and robustness:

- Standard wall function (based on $\Delta y = \Delta n\ /4$)
- Scalable wall function (based on $\Delta y = \Delta n\ /4$)
- Automatic wall treatment (based on $\Delta y = \Delta n$)

The scalable wall function y^+ is defined as:

$$y^+ = \max\ (y^*, 11.06) \qquad y^* = \frac{u^*\ \Delta n/4}{v} \tag{2–219}$$

and is therefore based on $^1/_4$ of the near wall grid spacing.

Note that both the scalable wall function and the automatic wall treatment can be run on arbitrarily fine meshes.

2.8.1.3. Automatic Near-Wall Treatment for Omega-Based Models

While the wall-functions presented above allow for a consistent mesh refinement, they are based on physical assumptions which are problematic, especially in flows at lower Reynolds numbers ($Re<10^5$), as the sublayer portion of the boundary layer is neglected in the mass and momentum balance. For flows at low Reynolds numbers, this can cause an error in the displacement thickness of up to 25%. It is therefore desirable to offer a formulation which will automatically switch from wall-functions to a low-*Re* near wall formulation as the mesh is refined. The $k-\omega$ model of Wilcox has the advantage that an analytical expression is known for ω in the viscous sublayer, which can be exploited to achieve this goal. The main idea behind the present formulation is to blend the wall value for ω between the logarithmic and the near wall formulation.

The automatic wall treatment allows a consistent y^+ insensitive mesh refinement from coarse grids, which do not resolve the viscous sublayer, to fine grids placing mesh points inside the viscous sublayer.

Note that for highly accurate simulations, like heat transfer predictions, a fine grid with y^+ around 1 is recommended.

The flux for the *k*-equation is artificially kept to be zero and the flux in the momentum equation is computed from the velocity profile. The equations are as follows:

Flux for the momentum equation, F_U:

$$F_U = -\rho\, u_\tau\, u^* \tag{2–220}$$

with:

$$u^* = \sqrt[4]{\left(\sqrt{\frac{\mu}{\rho}\left|\frac{\Delta U}{\Delta y}\right|}\right)^4 + \left(\sqrt{a_1 k}\right)^4} \tag{2–221}$$

$$u_\tau = \sqrt[4]{\left(u_\tau^{vis}\right)^4 + \left(u_\tau^{\log}\right)^4} \tag{2–222}$$

where

$$u_\tau^{vis} = \sqrt{\frac{\mu}{\rho}\left|\frac{\Delta U}{\Delta y}\right|}$$

and

$$u_\tau^{\log} = \frac{U}{1/\kappa \log\left(y^+\right) + C}$$

Flux for the *k*-equation:

$$F_k = 0 \tag{2–223}$$

In the ω-equation, an algebraic expression is specified instead of an added flux. It is a blend between the analytical expression for ω in the logarithmic region:

$$\omega_l = \frac{u^*}{a_l\, \kappa\, y} = \frac{1}{a_l\, \kappa\, \nu}\, \frac{u^*}{y^+} \tag{2–224}$$

and the corresponding expression in the sublayer:

$$\omega_s = \frac{6\, v}{\beta\,(\,\Delta y\,)} \qquad \textbf{(2–225)}$$

with Δy being the distance between the first and the second mesh point. In order to achieve a smooth blending and to avoid cyclic convergence behavior, the following formulation is selected:

$$\omega_\omega = \omega_s \sqrt{1 + \left(\frac{\omega_l}{\omega_s}\right)} \qquad \textbf{(2–226)}$$

While in the wall-function formulation, the first point is treated as being outside the edge of the viscous sublayer; in the low-*Re* mode, the location of the first mesh point is virtually moved down through the viscous sublayer as the mesh is refined. Note that the physical location of the first mesh point is always at the wall ($y = 0$). However, the first mesh point is treated as if it were Δy away from the wall. The error in the wall-function formulation results from this virtual shift, which amounts to a reduction in displacement thickness. Also, at very low Reynolds numbers this shift becomes visible when the solution is compared with a laminar calculation, because the shift is not needed in the laminar near wall treatment. This error is always present in the wall-function model. The shift is based on the distance between the first and the second mesh point $\Delta y = y_2 - y_1$ with y being the wall normal distance.

2.8.1.4. Treatment of Rough Walls

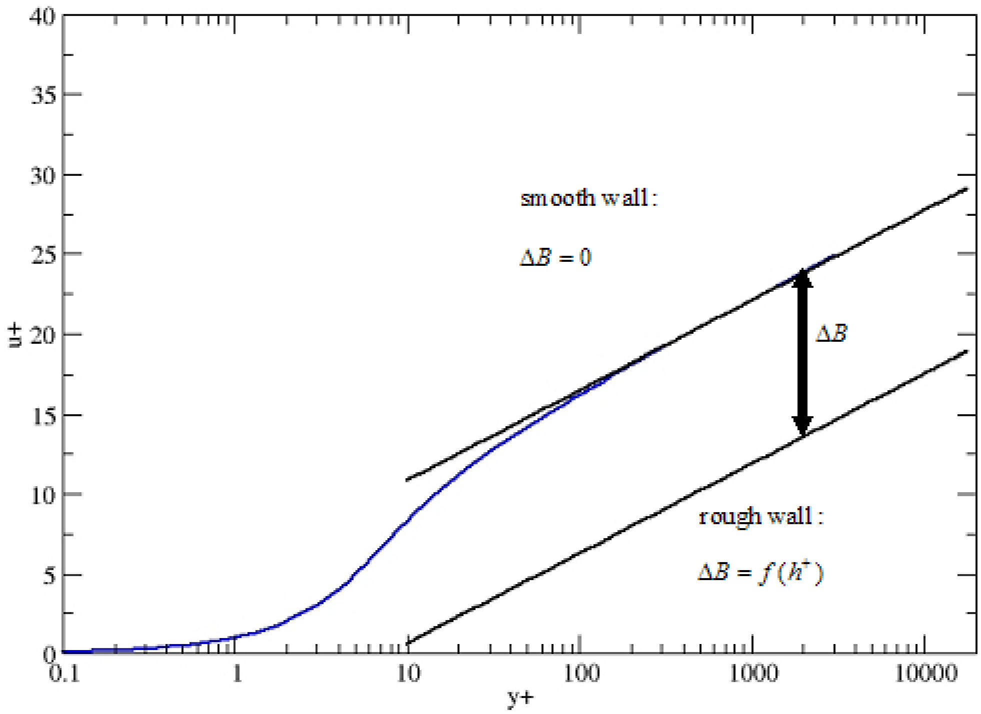
The near wall treatment which has been discussed above (Scalable Wall Functions, Automatic Wall Treatment) is appropriate when walls can be considered as hydraulically smooth. Surface roughness can have a significant effect on flows of engineering interest. Surface roughness typically leads to an increase in turbulence production near the wall. This in turn can result in significant increases in the wall shear stress and the wall heat transfer coefficients. For the accurate prediction of near wall flows, particularly with heat transfer, the proper modelling of surface roughness effects is essential for a good agreement with experimental data.

Wall roughness increases the wall shear stress and breaks up the viscous sublayer in turbulent flows. The figure below (*Figure 2.1* (p. 145)) shows the downward shift in the logarithmic velocity profile:

$$u^{+} = \frac{1}{\kappa}\ln\left(y^{+}\right) + B - \Delta B \qquad \textbf{(2–227)}$$

where B=5.2. The shift ΔB is a function of the dimensionless roughness height, h^{+}, defined as $h^{+} = h u_{\tau}/\upsilon.$

Figure 2.1 Downward Shift of the Logarithmic Velocity Profile

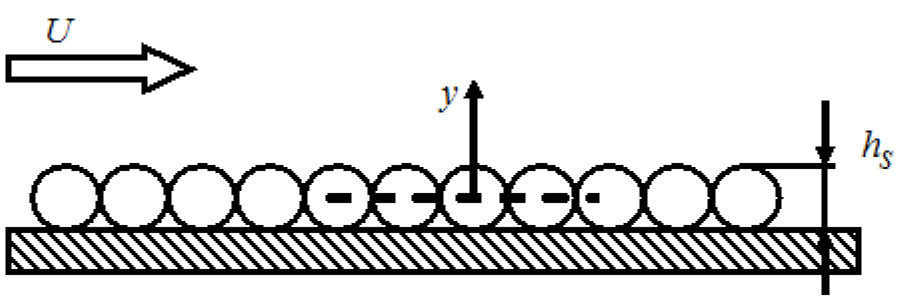


For sand-grain roughness, the downward shift can be expressed as:

$$\Delta B = \frac{1}{\kappa}\ln\left(1 + 0.3 h_s^{+}\right) \tag{2–228}$$

It has been shown that a technical roughness, which has peaks and valleys of different shape and size, can be described by an equivalent sand-grain roughness ([113], [77]). The corresponding picture is a wall with a layer of closely packed spheres, which have an average roughness height h_s (see *Figure 2.2* (p. 145)):

Figure 2.2 Equivalent Sand-Grain Roughness



Guidance to determine the appropriate equivalent sand-grain roughness height can be obtained from White [14], Schlichting [77] and Coleman et al. [193].

Depending on the dimensionless sand-grain roughness h_s^{+}, three roughness regimes can be distinguished:

- hydraulically smooth wall : $0 \le h_s^{+} \le 5$
- transitional-roughness regime: $5 \le h_s^{+} \le 70$
- fully rough flow: $h_s^{+} \ge 70$

The viscous sublayer is fully established near hydraulically smooth walls. In the transitional roughness regime the roughness elements are slightly thicker than the viscous sublayer and start to disturb it, so that in fully rough flows the sublayer is destroyed and viscous effects become negligible.

From the picture of the sand-grain roughness (note that *Figure 2.2* (p. 145) shows a two-dimensional cut of a three-dimensional arrangement), it can be assumed that the roughness has a blockage effect, which is about 50% of its height. The idea is to place the wall physically at 50% height of the roughness elements:

$$y = \max\left(y, h_s / \right) \tag{2–229}$$

This gives about the correct displacement caused by the surface roughness.

2.8.1.4.1. Rough Wall Treatment for Turbulence Models Based on the Dissipation Equation

The Scalable Wall Function approach neglects the viscous sublayer and limits the value used in the logarithmic formulation by a lower value of $y^{+} = \max\left(y^{*}, 11.06\right)$ (*Equation 2–219* (p. 142)). This limiter must be combined with the formulation *Equation 2–229* (p. 146) for the rough wall treatment and reads:

$$y^{+} = \max\left(y^{*}, h_S^{+} / , 11.06\right) \tag{2–230}$$

2.8.1.4.2. Automatic Rough Wall Treatment for Turbulence Models Based on the Omega Equation

The automatic wall treatment works well for smooth walls and has been extended for rough walls. However, any calibration of the model coefficients has to be performed for both y^{+} and h_s^{+} and is therefore more complex than for smooth walls. As outlined above, the viscous sublayer gets more and more disturbed with increasing values of h_s^{+} until it is destroyed in fully rough flows. Therefore, a blending between the viscous sublayer and the logarithmic region is not physical for large values of h_s^{+}. This fact supports the idea of placing the wall physically at 50% height of the roughness elements. Due to this shift, the viscous sublayer formulation has only an influence for small values of h_s^{+} in the automatic near wall treatment for rough walls. The blending between the viscous sublayer formulation and the wall function is described in detail in the report Lechner and Menter [194].

Because the automatic wall treatment works together with the transition model, it is the default rough wall treatment. This default **Automatic** setting for the **Wall Function** option can be set on the **Fluid Models** tab of the **Domain** details view in CFX-Pre. The CFX-Solver checks internally if a roughness height has been specified for a wall and then uses the proper wall treatment: when a roughness height exists, then the automatic rough wall treatment is used, otherwise the automatic smooth wall treatment.

2.8.1.4.3. Transition and Rough Walls

If the two-equation transition model (Gamma Theta Model) is used together with rough walls, then the option **Roughness Correlation** must be turned on in CFX-Pre on the **Fluid Models** tab in order to take the roughness effects into account in the transition model. Setting the sand-grain roughness height at a wall alone is not sufficient. The roughness correlation requires the geometric roughness height as input parameter because it turned out that for the transition process from laminar to turbulent flow, the geometric roughness height is more important than the equivalent sand-grain roughness height. The correlation is defined as

$$\mathrm{Re}_{\theta t,rough} = \tilde{R}e_{\theta t} \cdot f\ (h) \tag{2–231}$$

where the function f depends on the geometric roughness height h. This function is proprietary and therefore not given in this documentation. $\mathrm{Re}_{\theta t,rought}$ is then used in the correlations for F_{length} and $\mathrm{Re}_{\theta c}$ in *Equation 2–147* (p. 125) instead of the $\tilde{R}e_{\theta t}$ which represents the transition momentum thickness Reynolds number (for details, see *ANSYS CFX Transition Model Formulation* (p. 121)). In other words the roughness effect is taken into account by using a modified transition momentum thickness Reynolds number in the correlations for F_{length} and $\mathrm{Re}_{\theta c}$.

Consequently the shift in *Equation 2–229* (p. 146) was redefined as the minimum of either the geometric roughness height h or the sand grain roughness height h_s as follows:

$$y = \max\left(y, 0.5 \cdot \min\left(h, h_s\right)\right) \tag{2–232}$$

2.8.1.4.4. Wall Function Approach for Omega-Based Turbulence Models

The fact that the viscous sublayer is lost very quickly with increasing $h_s^{\ +}$ suggests to neglect the viscous sublayer in the formulation of a near wall treatment for rough walls and leads to a second rough wall treatment for ω-based turbulence models. It should only be used if the user knows that the influence of the viscous sublayer can be neglected ($h_s^{\ +}$ > 70) and if the flow is fully turbulent. It can only be enabled by extracting the CCL and changing the **Wall Function** from `Automatic` to `Scalable`.

Then scalable wall functions are used both for smooth and rough walls for the ω-based turbulence models. The idea is comparable to the scalable wall function approach for the $k-\varepsilon$ model, where the first grid point is shifted to the edge of the viscous sublayer and only the relations of the logarithmic region are used to derive the boundary conditions. In order to avoid a negative logarithm in the logarithmic law of the wall, a lower value of 11.06 is defined in the following manner:

$$y^{\ +} = \max\left(y^{\ *}, h_s^{\ +} / , 11.06\right) \tag{2–233}$$

where $y^{\ *} = \rho u^{\ *}\ \Delta y / \mu$.

It should be noted that this value is smaller than the lower limit in the scalable wall functions of the $k-\varepsilon$ model because for rough walls, the log layer extends to smaller values of $y^{\ +}$. Because the scalable wall function approach neglects the viscous sublayer, the diffusion coefficients in the Navier-Stokes and turbulence transport equations must no longer be computed as the sum of the laminar and turbulent viscosity. Instead the maximum value of the both is used. In the momentum equation for example the effective viscosity will be computed as

$$\mu_{eff} = \max\left(\mu, \mu_T\right) \tag{2–234}$$

Note that the logarithmic layer formulation is only correct if the molecular viscosity is not included in the equations. This is ensured by the above formulation.

2.8.1.5. Heat Flux in the Near-Wall Region

Heat flux at the wall can be modeled using the scalable wall function approach or the automatic wall treatment. Using similar assumptions as those above, the non-dimensional near-wall temperature profile follows a universal profile through the viscous sublayer and the logarithmic region. The non-dimensional temperature, T^+, is defined as:

$$T^+ = \frac{\rho\, c_p\, u^*\left(T_w - T_f\right)}{q_w} \tag{2–235}$$

where T_w is the temperature at the wall, T_f the near-wall fluid temperature, c_p the fluid heat capacity and q_w the heat flux at the wall. The above equation can be rearranged to get a simple form for the wall heat flux model:

$$q_w = \frac{\rho\, c_p\, u^*}{T^+}\left(T_w - T_f\right) \tag{2–236}$$

Turbulent fluid flow and heat transfer problems *without* conjugate heat transfer objects require the specification of the wall heat flux, q_w, or the wall temperature, T_w. The energy balance for each boundary control volume is completed by multiplying the wall heat flux by the surface area and adding to the corresponding boundary control volume energy equation. If the wall temperature is specified, the wall heat flux is computed from the equation above, multiplied by the surface area and added to the boundary energy control volume equation.

2.8.1.5.1. Scalable Wall Functions

For scalable wall functions, the non-dimensional temperature profile follows the log-law relationship:

$$T^+ = .1\ \ln\left(y^*\right) + \beta \tag{2–237}$$

where y^* is defined in *Equation 2–219* (p. 142) and β is computed from *Equation 2–239* (p. 149).

2.8.1.5.2. Automatic Wall Treatment

For the automatic wall function, the thermal boundary layer is modeled using the thermal law-of-the-wall function of B.A. Kader [15]. Then, the non-dimensional temperature distribution is modeled by blending the viscous sublayer and the logarithmic law of the wall. This can be modeled as:

$$T^{+} = Pr\, y^{*}\, e^{(-\Gamma)} + \left[.1 \ln(y^{*}) + \beta \right] e^{(-1/\Gamma)} \quad \textbf{(2–238)}$$

where:

$$\beta = \left(3.85\, Pr^{1/3} - 1.3 \right) + .1 \ln(Pr) \quad \textbf{(2–239)}$$

$$\Gamma = \frac{0.01\, (Pr\, y^{*})^{4}}{1 + 5\, Pr^{3}\, y^{*}} \quad \textbf{(2–240)}$$

Pr is the fluid Prandtl number, given by:

$$Pr = \frac{\mu\, c_p}{\lambda} \quad \textbf{(2–241)}$$

where λ is the fluid thermal conductivity.

2.8.1.5.3. Effect of Rough Walls

It can be shown for a flat plate with heat transfer that the wall heat flux is under-predicted if the logarithmic law of the wall for the temperature is used in the form which has been derived for smooth walls. Based on the idea that for $Pr = Pr_t = 1.0$ the velocity and temperature logarithmic laws of the wall are the same for smooth walls, the logarithmic law of the wall for the temperature has been extended in a similar way as the one for the velocity to account for the wall roughness effect. But instead of the roughness height Reynolds number h_s^{+}, the formulation is based on $Pr h_s^{+}$ due to the influence of the Prandtl number which has already been accounted for in the smooth wall formulation. A calibration constant has been introduced because the Kader logarithmic law of the wall is not strictly identical with the velocity log law for $Pr = Pr_t = 1.0$. It should also be noted that roughness elements introduce pressure-drag on the sides of the elements, which is not present in the thermal formulation. The coefficient by which the thermal layer has to be shifted is therefore expected to be smaller than one. The shift has been introduced in the rough wall treatment of both ε and ω-based turbulence models. The modification of the logarithmic law of the wall for the temperature reads:

$$T^{+}_{\log} = .1 \ln(\mathrm{Pr} \cdot y^{*}) + (3.85 \mathrm{Pr}^{1/3} - 1.3) - \Delta B_{th} \quad \textbf{(2–242)}$$

where

$$\Delta B_{th} = (1/0.41) \ln(1 + C 0.3 \mathrm{Pr} h_s^{+})$$

The coefficient C has been calibrated using the experimental data of Pimenta et al. [195] for a flat plate with heat transfer. The experiment was performed with air as fluid and it has been shown, that a value of 0.2 is a good choice in this case (Lechner and Menter [194]). Therefore the default value of C is 0.2. This coefficient is called **Energy Calibration Coefficient** and can be changed in the CCL if necessary in the following way:

```
FLUID MODELS:
  ...
  TURBULENCE MODEL:
    Option = SST
  END
  TURBULENT WALL FUNCTIONS:
    Option = Automatic
    Energy Calibration Coefficient = 0.2
  END
  ...
END
```

If the wall function approach for ω-based turbulence models is used, two additional points have to be considered.

1. Similar to *Equation 2–233* (p. 147) a lower limit for $\Pr \cdot y^{*}$, which appears in the equation of T^{+}, has been introduced. A value of 11.06 was chosen as lower limit:

$$\Pr \cdot y^{*} = \max\left(\Pr \cdot y^{*}, 11.06\right) \tag{2–243}$$

2. Because the scalable wall function approach neglects the viscous sublayer, the following relation is automatically used to compute the effective conductivity λ_{eff}:

$$\lambda_{eff} = \max\left(\lambda, \lambda_T\right) \tag{2–244}$$

2.8.1.5.4. Treatment of Compressibility Effects

With increasing Mach number (*Ma* > *3*), the accuracy of the wall-functions approach degrades, which can result in substantial errors in predicted shear stress, wall heat transfer and wall temperature for supersonic flows.

It has been found that the incompressible law-of-the-wall is also applicable to compressible flows if the velocity profile is transformed using a so-called "Van Driest transformation" [16]. The logarithmic velocity profile is given by:

$$\frac{U_{\text{comp}}}{u_{\tau}} = \frac{1}{\kappa} \ln\left(\frac{y^{*} y}{\nu}\right) + C \tag{2–245}$$

where $u_{\tau} = \left(\tau_w / \rho_w\right)^{1/}$, $y^{+} = u_{\tau}\left(y / \nu_w\right)$, $\kappa = 0.41$ and $C = 5.2$ The subscript *w* refers to wall conditions, and the subscript “comp” refers to a velocity defined by the following equation (transformation):

$$U_{comp} = \int \frac{\rho}{\rho_w} dU \tag{2–246}$$

Near a solid wall, the integrated near-wall momentum equation reduces to:

$$\tau = \tau_w \tag{2–247}$$

while the energy equation reduces to:

$$q = q_w + U\ \tau_w \tag{2–248}$$

Expressions for shear stress and heat flux applicable to the boundary layer region are:

$$\tau = \mu_t \frac{\partial U}{\partial y} \tag{2–249}$$

and

$$q = -\left(\frac{\mu_t c_p}{\mathrm{Pr}_t}\right)\left(\frac{\partial T}{\partial y}\right) \tag{2–250}$$

If *Equation 2–247* (p. 151), *Equation 2–249* (p. 151) and *Equation 2–250* (p. 151) are substituted into *Equation 2–248* (p. 151) and integrated, the resulting equation is:

$$T = T_w - \frac{Pr_t\, q_w\, U}{c_p\, \tau_w} - \frac{Pr_t\, U}{c_p} \tag{2–251}$$

which provides a relationship between the temperature and velocity profiles. Using the perfect gas law, and the fact that the pressure is constant across a boundary layer, *Equation 2–251* (p. 151) replaces the density ratio found in *Equation 2–246* (p. 150). Performing the integration yields the following equation for the "compressible" velocity:

$$U_{comp} = \sqrt{B}\left[\mathrm{asin}\left(\frac{A+U}{D}\right) - \mathrm{asin}\left(\frac{A}{D}\right)\right] \tag{2–252}$$

where:

$$A = q_w / \tau_w \tag{2–253}$$

$$B = \left(c_p\, T_w\right) / \left(Pr_t\right) \tag{2–254}$$

$$D = \sqrt{A + B} \tag{2–255}$$

The above derivation provides most of the equations necessary for the implementation of wall-functions which are applicable to compressible flows. These wall-functions are primarily implemented in two locations in ANSYS CFX: hydrodynamics and energy. First, consider the implementation in the hydrodynamics section. The equation for the wall-shear-stress is obtained after a slight rearrangement of Equation 2–245 (p. 150):

$$\frac{\tau_w}{\rho} = \frac{u^*}{\frac{1}{k}\ln\frac{u^* y}{v} + C} U_{\text{comp}} \tag{2–256}$$

This is similar to the low speed wall-function, except that U_{comp} now replaces U. The Van Driest transformation given by Equation 2–252 (p. 151), must now be performed on the near wall velocity. In the implementation, Equation 2–252 (p. 151) is re-written in the following form:

$$(Ec)^{1/}_{\text{comp}} = \left[\frac{}{Pr_t}\frac{T_w}{T}\right]^{1/}\left[\operatorname{asin}\left(\frac{B^* + Ec}{D^*}\right) - \operatorname{asin}\left(\frac{B^*}{D^*}\right)\right] \tag{2–257}$$

where:

$$\begin{aligned}
u^* &= C_\mu^{1/4} k^{1/} \\
\tau_w^* &= \rho\left(u^*\right) \\
Ec &= \frac{U}{c_p T} \\
B^* &= \frac{q_w}{c_p T \tau_w^*} U \\
D^* &= \left(B^* + \frac{}{Pr_t}\frac{T_w}{T} Ec\right)^{1/} \\
U_{\text{comp}} &= \left(c_p T\, (Ec)_{\text{comp}}\right)^{1/} \\
C &= 5. \\
Pr_t &= 0.9 \\
\frac{T_w}{T} &= 1 + Pr_t B^* + \frac{Pr_t}{} Ec
\end{aligned} \tag{2–258}$$

This completes the wall-function modifications for the hydrodynamics.

The relationship between wall heat transfer, near wall velocity and near wall temperature is given by Equation 2–251 (p. 151) (and also in rearranged form after Equation 2–257 (p. 152): the equation for T_w / T).

A dimensionless variable, θ^+, can be defined as:

$$\theta^+ = Pr_t \frac{U}{u^*} \tag{2–259}$$

and hence:

$$\theta^+ = Pr_t \frac{U}{u^*} = \frac{\rho\, c_p\, u^*}{q_w} \left(T_w - T - \frac{Pr_t\, U}{c_p} \right) \tag{2–260}$$

From *Equation 2–260* (p. 153) it is clear that knowing θ^+, the near wall velocity, the near wall temperature and the wall heat transfer, the wall temperature can be calculated (if instead the wall temperature is known, then the wall heat transfer is calculated). Before proceeding, it is instructive to consider the equation that is used in ANSYS CFX for low Mach number heat transfer. The equation is subsequently modified for use at higher Mach numbers. This unmodified equation is given by:

$$\theta^+ = (Pr\, U_1^+)\, e^{-\Gamma} + (Pr_t\, U^+)\, e^{-1/\Gamma} \tag{2–261}$$

where:

$$U_1^+ = y^+ \tag{2–262}$$

$$U^+ = \frac{1}{Pr_t}\left(.1 \ln(y^+) + \beta \right) \tag{2–263}$$

$$\Gamma = f_1 (Pr\, y^+) \tag{2–264}$$

$$\beta = f\,(Pr) \tag{2–265}$$

It can be seen that this equation blends between the linear and logarithmic near wall regions, and hence is more general than just using the logarithmic profile that is implied by *Equation 2–259* (p. 153). *Equation 2–261* (p. 153) has been extended for use in the compressible flow regime by interpreting the linear and logarithmic velocity profiles given above as “compressible” or Van Driest transformed velocities:

$$U^+_{\text{comp 1}} = y^+ \tag{2–266}$$

$$U^+_{\text{comp}} = \frac{1}{Pr_t}\left(.1 \ln(y^+) + \beta \right) \tag{2–267}$$

This change is consistent with *Equation 2–245* (p. 150). The “untransformed” velocities required by *Equation 2–261* (p. 153) can be obtained by applying the inverse Van Driest velocity transformation to these “compressible” velocities. The inverse transformation is given by:

$$U^{+} = \frac{1}{R} \sin\left(R\, U^{+}_{\mathrm{comp}}\right) - H\left(1 - \cos\left(R\, U^{+}_{\mathrm{comp}}\right)\right) \qquad \textbf{(2–268)}$$

where:

$$R = (u^{*}) / (\sqrt{B}) \qquad \textbf{(2–269)}$$

$$H = A / (u^{*}) \qquad \textbf{(2–270)}$$

and A and B are defined following *Equation 2–252* (p. 151) above. The reverse transformed velocities obtained from *Equation 2–268* (p. 154) are substituted into *Equation 2–261* (p. 153) to obtain the value of θ^{+}. Either the wall heat transfer or the wall temperature can then be obtained from *Equation 2–260* (p. 153).

The Van Driest transformation described above is used when the `High Speed (compressible) Wall Heat Transfer Model` is selected in CFX-Pre. This option is available only in combination with the Total Energy heat-transfer model.

2.8.1.6. Additional Variables in the Near Wall Region

The treatment of additional scalar variables in the near wall region is similar to that for heat flux.

Chapter 3: GGI and MFR Theory

This chapter provides an overview of the numerical methods used for GGI (general grid interface) in CFX.

A control surface approach is used to perform the connection across a GGI attachment or periodic condition. A physically based intersection algorithm is employed to provide the complete freedom to change the grid topology and physical distribution across the interface.

A general intersection algorithm permits connections to be successfully made, even when the resultant surfaces on either side of an interface do not physically “fit” together to form a well defined physical connection. In addition, an automatic surface trimming function is performed by the GGI algorithm, to account for mismatched surface extent. This means that a GGI attachment or periodic condition can be successfully defined where the surface on one side of the interface is larger (in extent) than the surface on the other side of the interface. The interface is constructed between the overlapping regions of the two sides of the interface.

Multiple Frames of Reference (MFR) allows the analysis of situations involving domains that are rotating relative to one another. For CFX, this feature focuses on the investigation of rotor/stator interaction for rotating machinery. Because MFR is based on the GGI technology, the most appropriate meshing style may be used for each component in the analysis.

When using a Reynolds stress turbulence model (see Reynolds Stress Model in the *CFX-Solver Manager User's Guide*) the contribution of the Reynolds stresses to the momentum equations at GGI interfaces is approximated by an eddy viscosity approach. This approximation influences only the discretization of the control volumes along the interface and is expected to be justifiable compared to the other approximations at the interface.

This chapter describes:

3.1. Interface Characteristics

The numerical algorithms employed, as well as the control surface treatment of the numerical fluxes across the interface, are designed and implemented in such a way as to provide for maximum robustness and accuracy. The treatment of the interface fluxes is fully implicit and fully conservative in mass, momentum, energy, scalars, etc. This means that the multigrid solver can be applied directly, without any penalty in terms of robustness or convergence rate, to problems involving GGI conditions. Each of the different types of GGI interfaces have the following attributes:

1. Strict conservation is maintained across the interface, for all fluxes of all equations (after accounting for changes in pitch).
2. The interface treatment is fully implicit, so that the presence of an interface does not adversely affect overall solution convergence.

3. The interface is applicable to incompressible, subsonic, transonic and supersonic flow conditions, and all model options within CFX (for example, turbulence models, multiphase models, mixture models, CHT, reaction, etc.).
4. The interface accounts internally for pitch change by scaling up or down (as required) the local flows as they cross the interface, for the case of frame change interfaces.
5. Any number of GGI connection conditions are possible within a computational domain.

The surface fluxes along each side of the interface are discretized in terms of nodal dependent variables, and in terms of control surface equations and control surface variables.

3.2. Numerics

If the case is transient rotor-stator, then the current relative position of each side of a sliding interface is first computed at the start of each timestep. Each frame change model then proceeds by discretizing the surface fluxes along each side of the interface in terms of nodal dependent variables, and in terms of control surface equations and control surface variables. Each interface surface flow is discretized using the standard flux discretization approach, but employs both the nodal dependent variables on the local side of the interface and control surface variables on the interface region. Balance equations within the interface region are generated for the interface variables from the flux contributions from both sides of the interface. These equations are called control surface equations (different from control volume equations) because they enforce a balance of surface flows over a given surface area.

In detail, the GGI connection condition is implemented as follows:

1. Define regions within the interface where the fluxes must balance: control surfaces. Within each control surface, identify new dependent variables. These are called interface variables. For a Stage interface, the balance is across the entire interface in the direction of rotation, with as many control surfaces perpendicular to the direction of rotation as the grid permits. For all other interfaces, the control surface balance is at the resolution of the interface grid.
2. Evaluate the fluxes at each interface location, by visiting all control volumes with surfaces exposed to the interface. Evaluate the surface flows using the `standard' approach taken for all interior flux evaluations for advection, diffusion, pressure in momentum, and mass flows. Use a combination of nodal dependent variables and the interface variables in these evaluations. For example consider advection; if the flow is into the interface control volume, the advected quantity is equated to the interface variable. If the flow is out of the interface control volume, the advected quantity is equated to the local nodal control volume variable. Below is a summary of all common flux discretizations at the interface:
   - **Advection**: Mass out is connected to the upstream (nodal) values, and mass in is connected to upstream (control surface) values.
   - **Diffusion**: A diffusion gradient is estimated using the regular shape function based gradient coefficients, but all dependence of the gradient estimate on nodes on the interface are changed to a dependence on interface variables.
   - **Pressure in momentum**: Evaluated using local nodal and control surface pressures and shape function interpolations.
   - **Local pressure gradient in mass redistribution**: This gradient is estimated using the regular shape function based gradient coefficients, but all dependence of the gradient estimate on nodal pressure on the interface is in terms of the interface pressure variable.
3. When a face is in contact with more than one control surface balance equation, discretize the fluxes at each integration point location in terms of generic interface unknowns, evaluate the flux N times (where N is the number of control surfaces in contact with the face), each time using a different control

surface variable and applying a weighting factor to the flow based on an `exposed fraction' basis. Each partial flow is accumulated in the control volume equation and in the relevant control surface equation.

4. Include each surface flow evaluation in two places: once in the interface control volume equation, and once in the adjacent control surface equation. Once all interface surfaces have been visited, the resulting equation set is as follows:

   - All interface control volume equations are complete. Each equation has coefficients to the usual neighboring nodal variables, as well as to interface variables.
   - All control surface equations are now complete. Each equation has coefficients to local interface variables as well as to nodal variables.

5. Solve the linear equation set to yield values of all nodal variables (from the control volume equations) and all interface variables (from the control surface equations).

Chapter 4: Transient Blade Row Modeling Theory

Using conventional methods for modeling transient rotor-stator applications, it is often the case that prohibitive computing resources are required to obtain detailed accurate simulations.

Figure 4.1 (p. 159) illustrates how conventional periodicity is inadequate when trying to use one passage to model a simulation that has a non-unity pitch ratio.

Figure 4.1 One passage periodicity cannot be applied

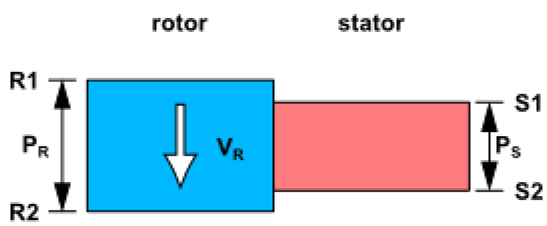


Figure 4.2 (p. 159) shows the workaround needed in using conventional programs to correctly capture the effect of non-unity pitch ratio. Simply put, copies need to be made on either side of the interface such that the resulting or effective pitch ratio resulting from the larger problem is unity. This can be very costly and might require the simulation of the entire wheel in turbo-machinery applications if the number of blades on one of the sides is a prime number.

Figure 4.2 Workaround using standard periodicity

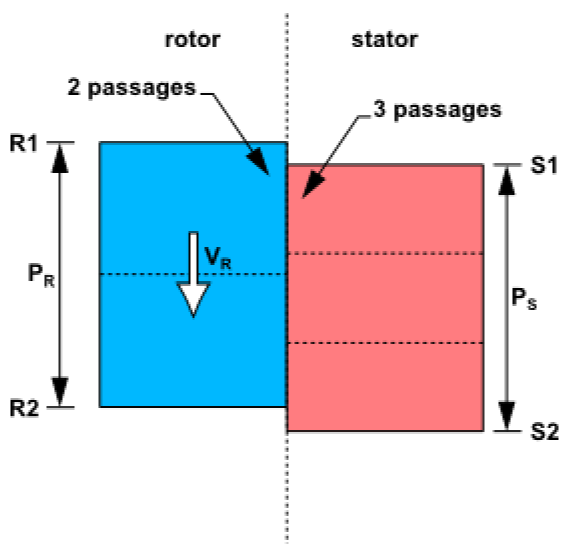
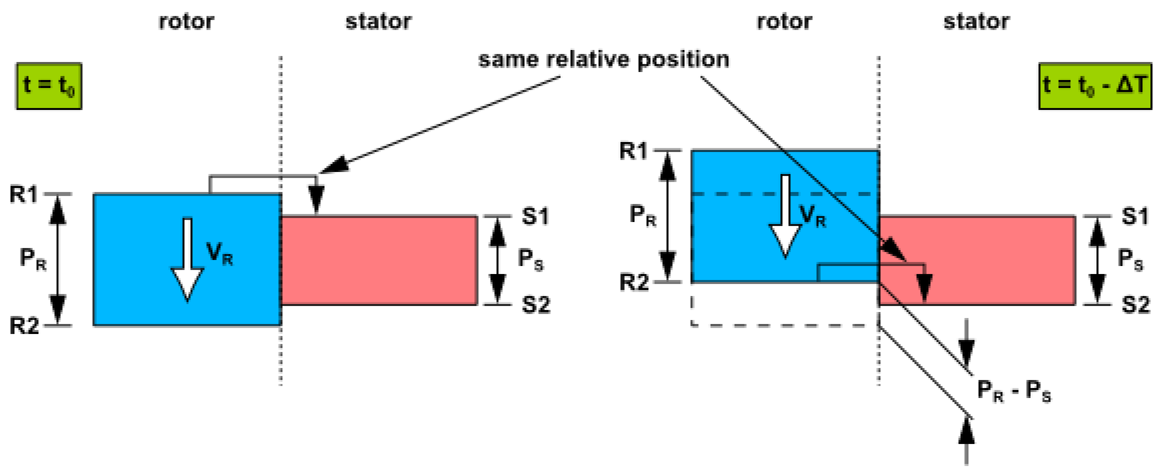


Figure 4.3 (p. 160) illustrates the common principle used in two of the Transient Blade Row methods: *Time Transformation Method* (p. 160) and *Fourier Transformation Method* (p. 162). Each of these methods applies in its own way what we have called here "phase shifted periodic boundary conditions". The basic principle is that the pitch-wise boundaries R1/R2 and S2/S2 are periodic to each other at different

instances in time. For example the relative position of R1 and S1 at t_0 is reproduced between sides R2 and S2 at an earlier time $t_0 - \Delta T$ where ΔT is defined by $(P_R - P_S) / V_R$. Here P_R and P_S are the rotor and stator pitches, respectively, and V_R is the velocity of the rotor.

Figure 4.3 Phase Shifted Periodic Boundary Conditions



4.1. Time Transformation Method

The Time Transformation method handles the problem of unequal pitch described above by transforming the time coordinates of each rotor and stator passages. Let the r, θ, and z coordinate axis represent the radial, tangential (pitchwise) and axial directions of the problem described in *Figure 4.3* (p. 160).

Mathematically, the condition of enforcing the flow spatial periodic boundary conditions on both rotor and stator passages, respectively, is given by:

$$U_{R1}(r,\theta,z,t) = U_{R2}(r,\theta+P_R,z,t-\Delta\mathrm{T}) \quad \textbf{(4–1)}$$

$$U_{S1}(r,\theta,z,t) = U_{S2}(r,\theta+P_S,z,t-\Delta\mathrm{T}) \quad \textbf{(4–2)}$$

Alternatively, we can apply the following set of space-time transformations [217] to the problem above as:

$$r' = r \quad \textbf{(4–3)}$$

$$\theta' = \theta \quad \textbf{(4–4)}$$

$$z' = z \quad \textbf{(4–5)}$$

$$t' = t - \lambda_{R,S}\,\theta \quad \textbf{(4–6)}$$

where $\lambda_{R,S} = \Delta T / P_{R,S}$.

The transformed set of equations now has regular spatial periodic boundary conditions as:

$$U_{R1}(r',\theta',z',t') = U_{R2}(r',\theta'+P_R,z',t') \quad \textbf{(4–7)}$$

$$U_{S1}(r',\theta',z',t') = U_{S2}(r',\theta'+P_S,z',t') \quad \textbf{(4–8)}$$

And the periodicity is maintained at any instant in time in the computational domain.

There are two important implications of working with the transformed system above instead of the conventional system:

- The equations that are solved are in the computational (r', θ', z', t') transformed space-time domain and need to be transformed back to physical (r, θ, z, t) domain before post-processing.
- The rotor and stator passages are marching at different time step sizes.

One way of understanding the difference in time step size stated in the second point is:

a) Each passage experiences a different period, i.e., the stator period is:

$$T_S = P_R / V_R \quad \textbf{(4–9)}$$

whereas the rotor period is:

$$T_R = P_S / V_R \quad \textbf{(4–10)}$$

b) The period in each passage must be discretized using an identical number of timesteps n. We can

therefore write the rotor and stator periods as:

$$T_S = n\ \Delta t_S \quad \textbf{(4–11)}$$

$$T_R = n\ \Delta t_R \quad \textbf{(4–12)}$$

where Δt_R and Δt_S represent the time step size for the rotor and stator, respectively.

c) Combining the two definitions of rotor and stator periods in a) and b) we have the time step sizes in the rotor and stator related by their pitch ratio as:

$$\frac{P_R}{P_S} = \frac{\Delta t_S}{\Delta t_R} \quad \textbf{(4–13)}$$

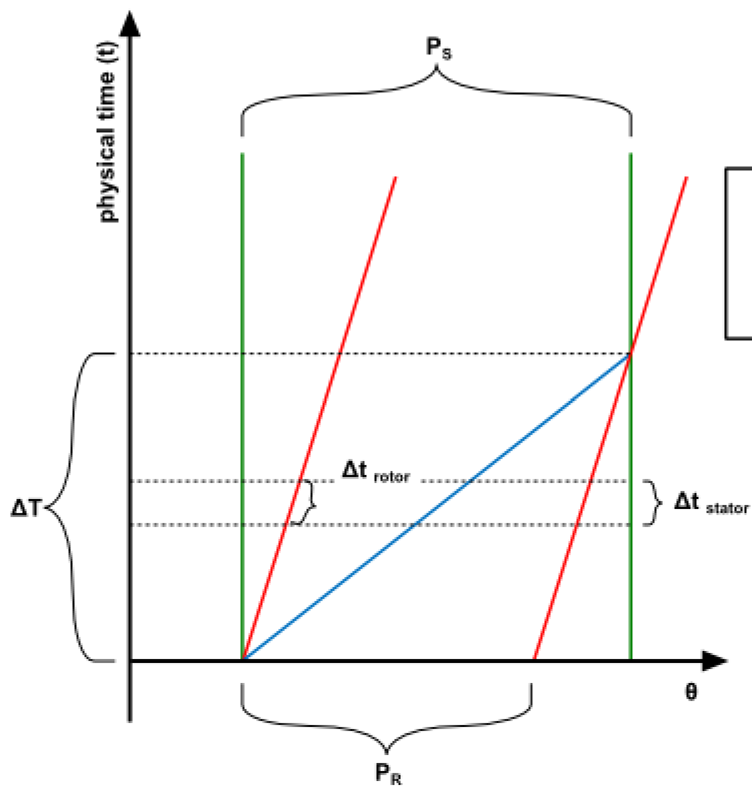
The simulation time step size set for the run is used in the stator passage(s) (Δt_S) and ANSYS CFX computes the respective rotor passage time step size (Δt_R) based on the rotor-stator interface pitch ratio as described above.

When the solution is transformed back to physical time, the elapsed simulation time is considered the stator simulation time.

The flow interaction across the rotor-stator interfaces is handled by the pitch scaling machinery already present in the Profile Transformation method.

Another way of seeing the difference in time step sizes is to look at the time vs. pitchwise direction plot for both stator and rotor domains. In *Figure 4.4* (p. 162) the rotor domain periodic boundaries are plotted with reference to the stator domain. One can notice that, even though the rotor pitch is smaller than the stator pitch in physical space, their boundaries match in the inclined computational space after time $t + \Delta T$. Furthermore, if we look at the time step size of stator and rotor, we see that they are different and related by the rotor-stator pitch ratio.

Figure 4.4 Rotor and stator periodic boundaries in space-time



For modeling information, see Transient Blade Row Modeling in the *CFX-Solver Modeling Guide*.

4.2. Fourier Transformation Method

The Fourier Transformation method applies the phase shifted boundary conditions illustrated in *Figure 4.3* (p. 160) in a more direct method than in the Time Transformation method.

It basically consists of attempting to directly "impose" a phase shift between the R1,R2 boundaries and the S1,S2 boundaries for the primitive variables of the problem. A direct method of applying the phase shift boundary method was proposed by Erdos [215] but his method requires storage of the signal for a compete period on all the R1/R2 S1/S2 boundaries including the interface shared by the rotor and stator sections. The Frequency Transformation method makes use of temporal Fourier series decomposition *Equation 4–14* (p. 163) to avoid storing the signal on the S1/S2 and R1/R2 boundaries. This method was initially introduced by He [218].

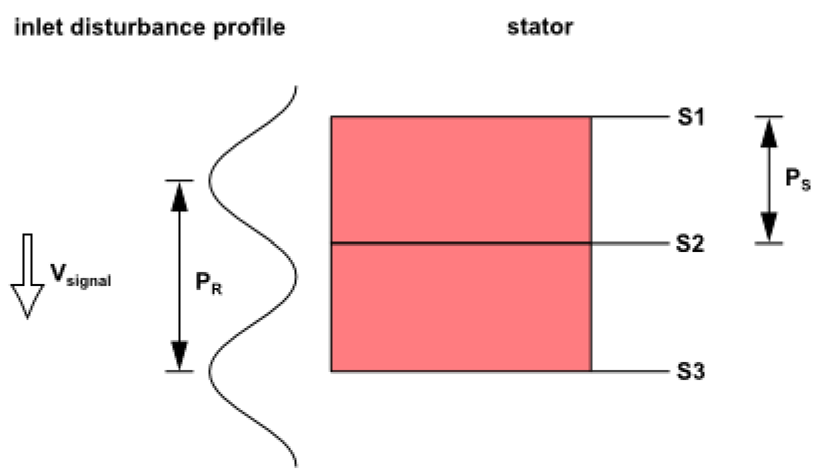
$$\hat{f}(t) = \sum_{m=-M}^{M} A_m e^{-j(\omega m t)} \qquad \textbf{(4–14)}$$

In this method the signal is decomposed into harmonics of the fundamental frequency $\omega / (2\pi)$ of each passage. For example in typical TRS problems this fundamental frequency is the inverse of the blade passing period T_{bpp}, where:

$$T_{\mathrm{bpp}} = \frac{2\pi}{w} \qquad \textbf{(4–15)}$$

The Fourier Transformation method in ANSYS CFX is implemented using a double passage method. The data is collected at the (implicit) interface (R2,S2) between the two passages, *Figure 4.5* (p. 163). While performance is case dependent, the double passage approach converges in fewer cycles than a single passage method, owing to the superior quality of the data update (Fourier data is collected at fully implicit flow variables, far from the actual periodic boundaries). The double passage Fourier Transformation approach converges very rapidly to a quasi-periodic state, requiring as few as 3-4 cycles in some cases.

Figure 4.5 Double Passage Method



In *Figure 4.5* (p. 163), the time shift is:

$$\Delta T = \frac{P_R - P_S}{V_{\mathrm{signal}}}$$

The signal $f_{S1}(t)$ on S1 is equal to the signal on S2 phase shifted by

$$+\Delta\mathrm{T}$$

$$f_{S1}(t) = f_{S2}(t + \Delta T) \approx \widehat{f}_{S2}(t + \Delta T)$$

The signal $f_{S3}(t)$ on S3 is equal to the signal on S2 phase shifted by

$-\Delta T$

:

$$f_{S3}(t) = f_{S2}(t - \Delta T) \approx \widehat{f}_{S2}(t - \Delta T)$$

Where $\hat{f}_{S2}\left(t\right)$ is the reconstructed signal at S2 using the accumulated Fourier coefficients on that same boundary.

For modeling information, see Transient Blade Row Modeling in the *CFX-Solver Modeling Guide*.

Chapter 5: Multiphase Flow Theory

This chapter discusses:

Two distinct multiphase flow models are available in CFX, an Eulerian-Eulerian multiphase model and a Lagrangian Particle Tracking multiphase model. This section describes Eulerian-Eulerian multiphase flow theory. For information about modeling Eulerian-Eulerian multiphase flow in ANSYS CFX, see "Multiphase Flow Modeling". Lagrangian Particle Tracking theory and modeling are described in *Particle Transport Theory* (p. 239) and "Particle Transport Modeling", respectively.

You should be familiar with the mathematical implementation of single-phase flow before reading this section. For details, see *Basic Solver Capability Theory* (p. 1).

5.1. Multiphase Notation

In addition to the notation given here, review the list of symbols. For details, see *List of Symbols* (p. 2).

Different phases of fluids are denoted using lowercase Greek letters α, β, γ, etc. In general, a quantity subscribed with α, β, γ, etc., refers to the value of the quantity for that particular phase. For example, the **volume fraction** of α is denoted r_α. Thus, the volume V_α occupied by phase α in a small volume V around a point of volume fraction r_α is given by:

$$V_\alpha = r_\alpha V \tag{5–1}$$

The total number of phases is N_P. The volume fraction of each phase is denoted r_α, where $\alpha = 1 \text{ to } N_P$.

It is important to distinguish between the **material density** and the **effective density** of a fluid α. The material density, ρ_α, is the density of the fluid if it is the only phase present, that is, the mass of α per unit volume of α. The effective density is then defined as:

$$\tilde{\rho}_\alpha = r_\alpha \rho_\alpha \qquad (5\text{–}2)$$

This is the actual mass per unit volume of phase α, given that phase α only occupies a fraction of the volume, that is, the mass of α per unit volume of the bulk fluid.

The **mixture density** is given by:

$$\rho_m = \sum_\alpha \rho_\alpha r_\alpha \qquad (5\text{–}3)$$

5.1.1. Multiphase Total Pressure

The total pressure in a multiphase simulation is defined as:

$$p_{\text{tot}} = p_{\text{stat}} + \sum_\alpha \frac{1}{2} r_\alpha \rho_\alpha \boldsymbol{U}_\alpha^2 \qquad (5\text{–}4)$$

This definition is used for both incompressible and compressible flows, whereas single phase flows treat total pressure differently depending on the simulation. For details, see *Total Pressure* (p. 16).

5.2. The Homogeneous and Inhomogeneous Models

Two different sub-models are available for Eulerian-Eulerian multiphase flow: the homogeneous model and the inter-fluid transfer (inhomogeneous) model.

5.2.1. The Inhomogeneous Model

5.2.1.1. Interfacial Area Density

Interfacial transfer of momentum, heat and mass is directly dependent on the contact surface area between the two phases. This is characterized by the interfacial area per unit volume between phase α and phase β, known as the interfacial area density, $A_{\alpha\beta}$. Note that it has dimensions of inverse length.

Interfacial transfer can be modeled using either the particle or mixture models. These essentially provide different algebraic prescriptions for the interfacial area density.

5.2.1.1.1. The Particle Model

The **Particle model** for interfacial transfer between two phases assumes that one of the phases is continuous (phase α) and the other is dispersed (phase β). The surface area per unit volume is then calculated by assuming that phase β is present as spherical particles of Mean Diameter d_β. Using this model, the interphase contact area is:

$$A_{\alpha\beta} = \frac{6\, r_\beta}{d_\beta} \qquad (5\text{–}5)$$

This simple model is modified for robustness purposes in two ways:

- r_β is clipped to a minimum volume fraction to ensure the area density does not go exactly to zero.
- For large r_β (that is, when the assumption of β being dispersed is invalid), the area density is decreased to reflect the fact that it should lead to zero as r_β tends to 1.

With these modifications, the area density for the particle model is implemented as

$$A_{\alpha\beta} = \frac{6\,\tilde{r}_\beta}{d_\beta} \tag{5–6}$$

where

$$\tilde{r}_\beta = \begin{cases} \max\left(r_\beta, r_{\min}\right) & \text{if}\left(r_\beta < r_{\max}\right) \\ \max\left(\dfrac{1-r_\beta}{1-r_{\max}}\, r_{\max}, r_{\min}\right) & \text{if}\left(r_\beta > r_{\max}\right) \end{cases} \tag{5–7}$$

By default, $r_{\max}$ and $r_{\min}$ take values of 0.8 and 10^{-7}, respectively. In some cases, it may be appropriate to use a different value for $r_{\min}$; for example, increasing it to 10^{-3} provides a very crude nucleation model for boiling a subcooled liquid. $r_{\min}$ is controlled by the parameter `Minimum Volume Fraction for Area Density`.

For non-drag forces, the solver uses a slightly different formulation of area density called the Unclipped Interfacial Area Density. In this formulation, the area density is permitted to go to zero, that is, $r_{\min} = 0$ in *Equation 5–7* (p. 167). In addition, the area density is reduced more aggressively as the dispersed phase volume fraction becomes large:

$$A_{\alpha\beta}^{\text{unclipped}} = f\, A_{\alpha\beta} \tag{5–8}$$

where:

$$f = \left(\frac{1-r_\beta}{1-r_{\beta'}}\right)^n,$$

$$n = 5 \quad \text{by default,and}$$

$$r_{\beta'} = \begin{cases} r_\beta & \text{if } \; r_\beta \le 0.25 \\ 0.393855 - 0.57142 r_\beta & \text{if } \; 0.25 < r_\beta \le 0.6 \\ 0.05 & \text{otherwise} \end{cases}$$

Non-dimensional interphase transfer coefficients may be correlated in terms of the particle **Reynolds number** and the fluid **Prandtl number**. These are defined using the particle mean diameter, and the continuous phase properties, as follows:

$$\mathrm{Re}_{\alpha\beta} = \frac{\rho_\alpha \; |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \; d_\beta}{\mu_\alpha} \tag{5–9}$$

$$\mathrm{Pr}_{\alpha\beta} = \frac{\mu_\alpha \, C_{P\,\alpha}}{\lambda_\alpha} \tag{5–10}$$

where μ_α, $C_{P\alpha}$ and λ_α are the viscosity, specific heat capacity and thermal conductivity of the continuous phase α.

5.2.1.1.2. The Mixture Model

This is a very simple model which treats both phases α, β symmetrically. The surface area per unit volume is calculated from

$$A_{\alpha\beta} = \frac{r_\alpha \, r_\beta}{d_{\alpha\beta}} \tag{5–11}$$

where $d_{\alpha\beta}$ is an interfacial length scale, which you must specify.

By way of example, suppose you have oil-water flow in which you may have either water droplets in continuous oil, or oil droplets in continuous water, in the limits $r_\alpha \to 0, r_\beta \to 0$ respectively. Then, a simple model for interfacial area density that has the correct behavior in these two limits is given by:

$$A_{\alpha\beta} = \frac{6 \, r_\alpha \, r_\beta}{r_\alpha \, d_\beta + r_\beta \, d_\alpha} \Rightarrow d_{\alpha\beta} = \frac{r_\alpha \, d_\beta + r_\beta \, d_\alpha}{6} \tag{5–12}$$

Non-dimensional interphase transfer coefficients may be correlated in terms of the **mixture Reynolds number** and **Prandtl number** defined as follows:

$$\mathrm{Re}_{\alpha\beta} = \frac{\rho_{\alpha\beta} \; |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \; d_{\alpha\beta}}{\mu_{\alpha\beta}} \tag{5–13}$$

$$\mathrm{Pr}_{\alpha\beta} = \frac{\mu \, C_{P\alpha\beta}}{\lambda_{\alpha\beta}} \tag{5–14}$$

where $\rho_{\alpha\beta}$, $\mu_{\alpha\beta}$, $C_{P\alpha\beta}$ and $\lambda_{\alpha\beta}$ are the density, viscosity, specific heat capacity and thermal conductivity of the mixture respectively, defined by:

$$\rho_{\alpha\beta} = r_\alpha \rho_\alpha + r_\beta \rho_\beta$$
$$\mu_{\alpha\beta} = r_\alpha \mu_\alpha + r_\beta \mu_\beta \tag{5–15}$$

5.2.1.1.3. The Free Surface Model

The free surface model attempts to resolve the interface between the fluids. If there are just two phases in the simulation, the following equation is used for interfacial area density:

$$A_{\alpha\beta} = |\nabla r_\alpha| \tag{5–16}$$

When more than two phases are present, this is generalized as follows:

$$A_{\alpha\beta} = \frac{2\ |\nabla r_\alpha|\ |\nabla r_\beta|}{|\nabla r_\alpha| + |\nabla r_\beta|} \tag{5–17}$$

5.2.2. The Homogeneous Model

In homogeneous multiphase flow, a common flow field is shared by all fluids, as well as other relevant fields such as temperature and turbulence. This allows some simplifications to be made to the multifluid model resulting in the homogeneous model.

For a given transport process, the homogeneous model assumes that the transported quantities (with the exception of volume fraction) for that process are the same for all phases, that is,

$$\varphi_\alpha = \varphi \quad 1 \le \alpha \le N_P \tag{5–18}$$

Because transported quantities are shared in homogeneous multiphase flow, it is sufficient to solve for the shared fields using bulk transport equations rather than solving individual phasic transport equations.

The bulk transport equations can be derived by summing the individual phasic transport equations (*Equation 5–202* (p. 212)) over all phases to give a single transport equation for φ:

$$\frac{\partial}{\partial t}(\rho\,\varphi) + \nabla \cdot (\rho\,\boldsymbol{U}\,\varphi - \Gamma\ \nabla\varphi) = S \tag{5–19}$$

where:

$$\rho = \sum_{\alpha=1}^{N_P} r_\alpha \rho_\alpha$$

$$\boldsymbol{U} = \frac{1}{\rho} \sum_{\alpha=1}^{N_P} r_\alpha \rho_\alpha \boldsymbol{U}_\alpha \tag{5–20}$$

$$\Gamma = \sum_{\alpha=1}^{N_P} r_\alpha \Gamma_\alpha$$

The homogeneous model does not need to be applied consistently to all equations. For example, the velocity field may be modeled as inhomogeneous, but coupled with a homogeneous turbulence model. Alternatively, a homogeneous velocity field may be coupled with inhomogeneous temperature fields. Homogeneous Additional Variables are also available in CFX.

5.3. Hydrodynamic Equations

The following is a summary of the equations of momentum and mass transfer for inhomogeneous and homogeneous multiphase flow in CFX. The equivalent for single-phase flow is also available. For details, see *Transport Equations* (p. 25).

5.3.1. Inhomogeneous Hydrodynamic Equations

The inhomogeneous hydrodynamic equations are as follows:

- *Momentum Equations* (p. 170)
- *Continuity Equations* (p. 171)
- *Volume Conservation Equation* (p. 171)
- *Pressure Constraint* (p. 172)

5.3.1.1. Momentum Equations

$$\begin{aligned} &\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha \boldsymbol{U}_\alpha\right) + \nabla \cdot \left(r_\alpha \left(\rho_\alpha \boldsymbol{U}_\alpha \otimes \boldsymbol{U}_\alpha\right)\right) \\ &= -r_\alpha \nabla p_\alpha + \nabla \cdot \left(r_\alpha \mu_\alpha \left(\nabla \boldsymbol{U}_\alpha + (\nabla \boldsymbol{U}_\alpha)^T\right)\right) \\ &+ \sum_{\beta=1}^{N_p} \left(\Gamma^+_{\alpha\beta} \boldsymbol{U}_\beta - \Gamma^+_{\beta\alpha} \boldsymbol{U}_\alpha\right) + \boldsymbol{S}_{M\alpha} + \boldsymbol{M}_\alpha \end{aligned} \tag{5–21}$$

- $\boldsymbol{S}_{M\alpha}$ describes momentum sources due to external body forces, and user defined momentum sources. For details, see *Sources* (p. 54).
- $\boldsymbol{M}_\alpha$ describes the interfacial forces acting on phase α due to the presence of other phases. Additional information for the models available for interfacial forces is available in *Interphase Momentum Transfer Models* (p. 173).
- The term:

$$\left(\Gamma^{+}_{\alpha\beta}\,\boldsymbol{U}_{\beta} - \Gamma^{+}_{\beta\,\alpha}\,\boldsymbol{U}_{\alpha}\right) \tag{5–22}$$

represents momentum transfer induced by interphase mass transfer. For details, see *Interphase Mass Transfer* (p. 216).

- The above momentum equations are valid for fluid phases only. For dispersed solid phases, additional terms are present representing additional stresses due to particle collisions.

5.3.1.2. Continuity Equations

$$\frac{\partial}{\partial t}\left(r_{\alpha}\rho_{\alpha}\right) + \nabla\cdot\left(r_{\alpha}\rho_{\alpha}\,\boldsymbol{U}_{\alpha}\right) = \boldsymbol{S}_{M\,S\,\alpha} + \sum_{\beta=1}^{N_p}\Gamma_{\alpha\beta} \tag{5–23}$$

- $\boldsymbol{S}_{M\,S\,\alpha}$ describes user specified mass sources. For details, see *Sources* (p. 54).
- $\Gamma_{\alpha\beta}$ is the mass flow rate per unit volume from phase β to phase α. This term only occurs if interphase mass transfer takes place. For details, see *Interphase Mass Transfer* (p. 216).

5.3.1.3. Volume Conservation Equation

This is simply the constraint that the volume fractions sum to unity:

$$\sum_{\alpha=1}^{N_P} r_{\alpha} = 1 \tag{5–24}$$

This equation may also be combined with the phasic continuity equations to obtain a transported volume conservation equation. Take *Equation 5–23* (p. 171), divide by phasic density, and sum over all phases. This yields:

$$\sum_{\alpha}\frac{1}{\rho_{\alpha}}\left(\frac{\partial}{\partial t}\left(r_{\alpha}\rho_{\alpha}\right) + \nabla\cdot\left(r_{\alpha}\rho_{\alpha}\,\boldsymbol{U}_{\alpha}\right)\right) = \sum_{\alpha}\frac{1}{\rho_{\alpha}}\left(\boldsymbol{S}_{M\,S\,\alpha} + \sum_{\beta=1}^{N_P}\Gamma_{\alpha\beta}\right) \tag{5–25}$$

Interpreting this equation is simpler if you consider the special case of incompressible phases with no sources, in which it simplifies to:

$$\sum_{\alpha}\nabla\cdot\left(r_{\alpha}\,\boldsymbol{U}_{\alpha}\right) = 0 \tag{5–26}$$

which requires the volume flows to have zero divergence. *Equation 5–25* (p. 171) is the volume continuity equation solved by the CFX-Solver.

5.3.1.4. Pressure Constraint

The complete set of hydrodynamic equations represent $4\ N_P + 1$ equations in the $5\ N_P$ unknowns U_α, V_α, W_α, r_α, p_α. You need $N_P - 1$ more equations to close the system. These are given by constraints on the pressure, namely that all phases share the same pressure field:

$$p_\alpha = p \quad \text{for all} \ \ \alpha = 1 \ ,\ldots\ldots, N_P \tag{5–27}$$

5.3.2. Homogeneous Hydrodynamic Equations

5.3.2.1. Momentum Equations

The homogeneous model for momentum transport assumes:

$$\boldsymbol{U}_\alpha = \boldsymbol{U}, 1 \le \alpha \le N_P \tag{5–28}$$

and is given by Equation 5–23 (p. 171) and the momentum equation:

$$\frac{\partial}{\partial t}(\rho\,\boldsymbol{U}) + \nabla \cdot \left(\rho\,\boldsymbol{U} \otimes \boldsymbol{U} - \mu\left(\nabla \boldsymbol{U} + (\nabla \boldsymbol{U})^T\right)\right) = \boldsymbol{S}_M - \nabla p \tag{5–29}$$

where:

$$\begin{aligned} \rho &= \sum_{\alpha=1}^{N_P} r_\alpha \rho_\alpha \\ \mu &= \sum_{\alpha=1}^{N_P} r_\alpha \mu_\alpha \end{aligned} \tag{5–30}$$

The following points should be noted:

- The interphase transfer terms have all canceled out.
- This is essentially a single phase transport equation, with variable density and viscosity.

5.3.2.2. Continuity Equations

The homogeneous continuity equation is the same as for full multiphase, Equation 5–23 (p. 171), except that $\boldsymbol{U}$ is not phase specific.

5.3.2.3. Volume Conservation Equations

The homogeneous volume conservation equation is the same as for full multiphase, Equation 5–24 (p. 171), except that $\boldsymbol{U}$ is not phase specific.

5.3.2.4. Pressure Constraint

The pressure constraint given above for full multiphase flow is also used for homogeneous multiphase flow.

5.4. Multicomponent Multiphase Flow

In multiphase multicomponent flows, transport equations for the mass fractions of components $Y_{A\,\alpha}$ are assumed to take a similar form those used for single-phase multicomponent flow:

$$\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha Y_{A\,\alpha}\right) + \nabla \cdot \left(r_\alpha \left(\rho_\alpha \boldsymbol{U}_\alpha Y_{A\,\alpha} - \rho_\alpha D_{A\,\alpha} \left(\nabla Y_{A\,\alpha}\right)\right)\right) = S_{A\alpha} \tag{5–31}$$

You should note that the molecular diffusion coefficients are given by $\rho_\alpha D_{A\,\alpha}$, where $D_{A\,\alpha}$ is the Kinematic Diffusivity.

Source terms in multicomponent multiphase flow behave the same as multicomponent mass sources, but on a per fluid basis. For details, see *Sources* (p. 54).

5.5. Interphase Momentum Transfer Models

The theory described in this section only applies to inhomogeneous multiphase flow. When using the homogeneous model, momentum transfer between phases is assumed to be very large.

Interphase momentum transfer, $\boldsymbol{M}_{\alpha\beta}$, occurs due to interfacial forces acting on each phase α, due to interaction with another phase β. The total force on phase α due to interaction with other phases is denoted $\boldsymbol{M}_\alpha$, and is given by:

$$\boldsymbol{M}_\alpha = \sum_{\beta \neq \alpha} \boldsymbol{M}_{\alpha\beta} \tag{5–32}$$

Note that interfacial forces between two phases are equal and opposite, so the net interfacial forces sum to zero:

$$\left(\boldsymbol{M}_{\alpha\beta} = -\boldsymbol{M}_{\beta\,\alpha}\right) \Rightarrow \sum_\alpha \boldsymbol{M}_\alpha = 0 \tag{5–33}$$

The total interfacial force acting between two phases may arise from several independent physical effects:

$$\boldsymbol{M}_{\alpha\beta} = \boldsymbol{M}_{\alpha\beta}^{D} + \boldsymbol{M}_{\alpha\beta}^{L} + \boldsymbol{M}_{\alpha\beta}^{LUB} + \boldsymbol{M}_{\alpha\beta}^{VM} + \boldsymbol{M}_{\alpha\beta}^{TD} + \boldsymbol{M}_S + \ldots. \tag{5–34}$$

The forces indicated above respectively represent the interphase drag force, lift force, wall lubrication force, virtual mass force, turbulence dispersion force and solids pressure force (for dense solid particle phases only).

CFX provides a wide range of physical models for these forces. These models are described in the following places:

- *Interphase Drag* (p. 174)
- *Lift Force* (p. 181)
- *Wall Lubrication Force* (p. 185)
- *Virtual Mass Force* (p. 184)
- *Interphase Turbulent Dispersion Force* (p. 187)
- *Solid Particle Collision Models* (p. 188)

For more information, see Multiphase Flow with Turbulence Dispersion Force in the *CFX Reference Guide*.

5.5.1. Interphase Drag

The following general form is used to model interphase drag force acting on phase α due to phase β:

$$\boldsymbol{M}_\alpha = c_{\alpha\beta}^{(d)} \left(\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right) \tag{5–35}$$

Note that $c_{\alpha\alpha} = 0$ and $c_{\alpha\beta} = c_{\beta\alpha}$. Hence, the sum over all phases of all interphase transfer terms is zero.

In this section, you will learn how the coefficients $c_{\alpha\beta}^{(d)}$ may be computed from a knowledge of dimensionless drag coefficients. The range of models available for drag coefficients is also described.

The total drag force is most conveniently expressed in terms of the dimensionless **drag coefficient**:

$$C_D = \frac{D}{\frac{1}{2}\rho_\alpha \left(U_\alpha - U_\beta\right)^2 A} \tag{5–36}$$

where ρ is the fluid density, $\left(U_\alpha - U_\beta\right)$ is the relative speed, D is the magnitude of the drag force and A is the projected area of the body in the direction of flow. The continuous phase is denoted by α and the dispersed phase is denoted by β.

5.5.2. Interphase Drag for the Particle Model

For spherical particles, the coefficients $c_{\alpha\beta}^{(d)}$ may be derived analytically. The area of a single particle projected in the flow direction, A_p, and the volume of a single particle V_p are given by:

$$A_p = \frac{\pi d^2}{4} \qquad V_p = \frac{\pi d^3}{6} \tag{5–37}$$

where d is the mean diameter. The number of particles per unit volume, n_p, is given by:

$$n_p = \frac{r_\beta}{V_p} = \frac{6\, r_\beta}{\pi\, d^3} \tag{5–38}$$

The drag exerted by a single particle on the continuous phase is:

$$\boldsymbol{D}_p = \frac{1}{2}\, C_D\, \rho_\alpha\, A_p\, |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \left(\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right) \tag{5–39}$$

Hence, the total drag per unit volume on the continuous phase is:

$$\boldsymbol{D}_{\alpha\beta} = n_p\, \boldsymbol{D}_p = \frac{3}{4}\, \frac{C_D}{d}\, r_\beta\, \rho_\alpha\, |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \left(\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right) \tag{5–40}$$

Comparing with the *Momentum Equations* (p. 170) for phase α, where the drag force per unit volume is:

$$\boldsymbol{D}_{\alpha\beta} = c_{\alpha\beta}^{(d)} \left(\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right) \tag{5–41}$$

you get:

$$c_{\alpha\beta}^{(d)} = \frac{3}{4}\, \frac{C_D}{d}\, r_\beta\, \rho_\alpha\, |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \tag{5–42}$$

which can be written as:

$$c_{\alpha\beta}^{(d)} = \frac{C_D}{8}\, A_{\alpha\beta}\, \rho_\alpha\, |\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha| \tag{5–43}$$

This is the form implemented in CFX.

The following section describes drag correlations specific to dispersed multiphase flow.

5.5.2.1. Sparsely Distributed Solid Particles

At low particle Reynolds numbers (the viscous regime), the drag coefficient for flow past spherical particles may be computed analytically. The result is Stokes' law:

$$C_D = \frac{24}{\mathrm{Re}}, \quad \mathrm{Re} \ll 1 \tag{5–44}$$

For particle Reynolds numbers, such as *Equation 5–9* (p. 168), that are sufficiently large for inertial effects to dominate viscous effects (the inertial or Newton's regime), the drag coefficient becomes independent of Reynolds number:

$$C_D = 0.44 , \quad 1000 \le \mathrm{Re} \le 1 - 2\ x\ 10^5 \tag{5–45}$$

In the transitional region between the viscous and inertial regimes, $0.1 < \mathrm{Re} < 1000$ for spherical particles, both viscous and inertial effects are important. Hence, the drag coefficient is a complex function of Reynolds number, which must be determined from experiment.

This has been done in detail for **spherical particles**. Several empirical correlations are available. The one available in CFX is due to Schiller and Naumann (1933) [6]. It can be written as follows:

5.5.2.1.1. Schiller Naumann Drag Model

$$C_D = \frac{24}{\mathrm{Re}}\left(1 + 0.15\ \mathrm{Re}^{0.687}\right) \tag{5–46}$$

CFX modifies this to ensure the correct limiting behavior in the inertial regime by taking:

$$C_D = \max\left(\frac{24}{\mathrm{Re}}\left(1 + 0.15\ \mathrm{Re}^{0.687}\right), 0.44\right) \tag{5–47}$$

5.5.2.2. Densely Distributed Solid Particles

5.5.2.2.1. Densely Distributed Solid Particles: Wen Yu Drag Model

$$\begin{gathered} C_D = r_c^{-1.65} \max\left(\frac{24}{\mathrm{Re}'}\left(1 + 0.15\ \mathrm{Re}'^{0.687}\right), 0.44\right) \\ \mathrm{Re}' = r_c\,\mathrm{Re} \end{gathered} \tag{5–48}$$

Note that this has the same functional form as the Schiller Naumann correlation, with a modified particle Reynolds number, and a power law correction, both functions of the continuous phase volume fraction r_c.

You may also change the **Volume Fraction Correction Exponent** from its default value of -1.65, if you want.

Note

Although the Wen Yu drag law implemented in ANSYS CFX follows the implementation by Gidaspow [18] and its subsequent wide use, this implementation of the drag law is, in fact, quite different from that given in the original Wen and Yu paper [181].

5.5.2.2.2. Densely Distributed Solid Particles: Gidaspow Drag Model

$$
\begin{aligned}
&C_D = C_D\,(\text{Wen Yu})\ , \quad r_c > 0.8 \\
&c_{\alpha\beta}^{(d)} = 150\,\frac{(1-r_c)^2\,\mu_c}{r_c\,d_p^2} + \frac{7}{4}\,\frac{(1-r_c)\,\rho_c\,|\boldsymbol{U}_c - \boldsymbol{U}_d|}{d_p}\ , \quad r_c < 0.8
\end{aligned}
\tag{5–49}
$$

This uses the Wen Yu correlation for low solid volume fractions $r_d < 0.2$, and switches to Ergun's law for flow in a porous medium for larger solid volume fractions.

Note that this is discontinuous at the cross-over volume fraction. In order to avoid subsequent numerical difficulties, CFX modifies the original Gidaspow model by linearly interpolating between the Wen Yu and Ergun correlations over the range $0.7 < r_c < 0.8$.

You may also change the **Volume Fraction Correction Exponent** of the Wen Yu part of the correlation from its default value of -1.65, if you want.

5.5.2.3. Sparsely Distributed Fluid Particles (Drops and Bubbles)

At sufficiently small particle Reynolds numbers (the viscous regime), fluid particles behave in the same manner as solid spherical particles. Hence, the drag coefficient is well approximated by the Schiller-Naumann correlation described above.

At larger particle Reynolds numbers, the inertial or distorted particle regime, surface tension effects become important. The fluid particles become, at first, approximately ellipsoidal in shape, and finally, spherical cap shaped.

In the spherical cap regime, the drag coefficient is well approximated by:

$$
C_D\,(\text{cap}) = \frac{8}{3} \tag{5–50}
$$

Several correlations are available for the distorted particle regime. CFX uses the Ishii Zuber [19] and Grace [35] correlations.

5.5.2.3.1. Sparsely Distributed Fluid Particles: Ishii-Zuber Drag Model

In the distorted particle regime, the drag coefficient is approximately constant, independent of Reynolds number, but dependent on particle shape through the dimensionless group known as the Eotvos number, which measures the ratio between gravitational and surface tension forces:

$$
Eo = \frac{g\,\Delta\rho\,d_p^2}{\sigma} \tag{5–51}
$$

Here, $\Delta\rho$ is the density difference between the phases, g is the gravitational acceleration, and σ is the surface tension coefficient.

The Ishii-Zuber correlation gives:

$$C_D\,(\text{ellipse}) = \frac{2}{3}\,Eo^{1/2} \qquad \textbf{(5–52)}$$

In this case, CFX automatically takes into account the spherical particle and spherical cap limits by setting:

$$\begin{aligned} C_D\,(\text{dist}) &= \min\,(\,C_D\,(\text{ellipse}),\, C_D\,(\text{cap})\,) \\ C_D &= \max\,(\,C_D\,(\text{sphere}),\, C_D\,(\text{dist})\,) \end{aligned} \qquad \textbf{(5–53)}$$

The Ishii Zuber Model also automatically takes into account dense fluid particle effects. For details, see *Densely Distributed Fluid Particles* (p. 179).

5.5.2.3.2. Sparsely Distributed Fluid Particles: Grace Drag Model

The Grace drag model is formulated for flow past a single bubble. Here the drag coefficient in the distorted particle regime is given by:

$$C_D\,(\text{ellipse}) = \frac{4}{3}\,\frac{g\,d}{U_T^2}\,\frac{\Delta\rho}{\rho_c} \qquad \textbf{(5–54)}$$

where the terminal velocity U_T is given by:

$$U_T = \frac{\mu_\text{c}}{\rho_\text{c}\,d_p}\,M^{-0.149}\,(J-0.857) \qquad \textbf{(5–55)}$$

where:

$$M = \frac{\mu_c^{\,4}\,g\,\Delta\rho}{\rho^2\,\sigma^3} = \text{Morton Number} \qquad \textbf{(5–56)}$$

and:

$$J = \begin{cases} 0.94H^{0.757} & 2 < H \le 59.3 \\ 3.42H^{0.441} & H > 59.3 \end{cases} \qquad \textbf{(5–57)}$$

$$H = \frac{4}{3}\,Eo\,M^{-0.149}\left(\frac{\mu_c}{\mu_\text{ref}}\right)^{-0.14} \qquad \textbf{(5–58)}$$

$\mu_\text{ref} = 0.0009\ \text{kg m}^{-1}\ \text{s}^{-1}$ is the molecular viscosity of water at 25°C and 1 bar.

In this case, CFX automatically takes into account the spherical particle and spherical cap limits by setting:

$$C_D(\text{dist}) = \min\left(C_D(\text{ellipse}), C_D(\text{cap})\right)$$
$$C_D = \max\left(C_D(\text{sphere}), C_D(\text{dist})\right) \quad \text{(5–59)}$$

5.5.2.4. Densely Distributed Fluid Particles

5.5.2.4.1. Densely Distributed Fluid Particles: Ishii-Zuber Drag Model

The Ishii Zuber [19] drag laws automatically take into account dense particle effects. This is done in different ways for different flow regimes.

In the **viscous regime**, where fluid particles may be assumed to be approximately spherical, the Schiller Naumann correlation is modified using a mixture Reynolds number based on a mixture viscosity.

5.5.2.4.2. Densely Distributed Fluid Particles: Dense Spherical Particle Regime (Ishii Zuber)

$$C_D(\text{sphere}) = \frac{24}{\text{Re}_m}\left(1 + 0.15\,\text{Re}_m^{0.687}\right)$$
$$\text{Re}_m = \frac{\rho_c\,|\boldsymbol{U}_d - \boldsymbol{U}_c|\,d_p}{\mu_m}$$
$$\frac{\mu_m}{\mu_c} = \left(1 - \frac{r_d}{r_{dm}}\right)^{-2.5\,r_{dm}\,\mu_*} \quad \text{(5–60)}$$
$$\mu_* = \frac{\mu_d + 0.4\,\mu_c}{\mu_d + \mu_c}$$

Here, r_{dm} is the user defined Maximum Packing value. This is defaulted to unity for a dispersed fluid phase.

In the **distorted particle** regime, the Ishii Zuber modification takes the form of a multiplying factor to the single particle drag coefficient.

5.5.2.4.3. Densely Distributed Fluid Particles: Dense Distorted Particle Regime (Ishii Zuber)

$$\begin{aligned} C_D(\text{ellipse}) &= E(r_d)\ C_{D\infty} \\ C_{D\infty} &= \frac{2}{3} Eo^{1/2} \\ E(r_d) &= \frac{\left(1 + 17.67 f(r_d)^{6/7}\right)}{18.67 f(r_d)} \\ f(r_d) &= \frac{\mu_c}{\mu_m}(1 - r_d)^{1/2} \end{aligned} \tag{5–61}$$

5.5.2.4.4. Densely Distributed Fluid Particles: Dense Spherical Cap Regime (Ishii Zuber)

$$\begin{aligned} C_D(\text{cap}) &= (1 - r_d)^2\ C_{D\infty} \\ C_{D\infty} &= \frac{8}{3} \end{aligned} \tag{5–62}$$

The Ishii Zuber correlation, as implemented in CFX, automatically selects flow regime as follows:

5.5.2.4.5. Densely Distributed Fluid Particles: Automatic Regime Selection (Ishii Zuber)

$$\begin{aligned} &C_D = C_D(\text{sphere}) && \text{if} \quad C_D(\text{sphere}) \geq C_D(\text{ellipse}) \\ &C_D = \min\left(C_D(\text{ellipse}), C_D(\text{cap})\right) && \text{if} \quad C_D(\text{sphere}) < C_D(\text{ellipse}) \end{aligned} \tag{5–63}$$

5.5.2.4.6. Densely Distributed Fluid Particles: Grace Drag Model

The Grace [35] drag model is formulated for flow past a single bubble. For details, see *Sparsely Distributed Fluid Particles (Drops and Bubbles)* (p. 177).

For high bubble volume fractions, it may be modified using a simple power law correction:

$$C_D = r_c^{\,p}\ C_{D\infty} \tag{5–64}$$

Here, $C_{D\infty}$ is the single bubble Grace drag coefficient. Advice on setting the exponent value for the power law correction is available in Densely Distributed Fluid Particles: Grace Drag Model in the *CFX-Solver Modeling Guide*.

5.5.3. Interphase Drag for the Mixture Model

In the mixture model, a non-dimensional **drag coefficient** C_D is defined as follows:

$$\boldsymbol{D}_{\alpha\beta} = C_D \rho_{\alpha\beta} A_{\alpha\beta} \left|\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right| \left(\boldsymbol{U}_\beta - \boldsymbol{U}_\alpha\right) \quad \textbf{(5–65)}$$

where $\boldsymbol{D}_{\alpha\beta}$ is the total drag exerted by phase β on phase α per unit volume.

The mixture density $\rho_{\alpha\beta}$ is given by:

$$\rho_{\alpha\beta} = r_\alpha \rho_\alpha + r_\beta \rho_\beta \quad \textbf{(5–66)}$$

and the interfacial area per unit volume $A_{\alpha\beta}$ is given by:

$$A_{\alpha\beta} = \frac{r_\alpha r_\beta}{d_{\alpha\beta}} \quad \textbf{(5–67)}$$

where $d_{\alpha\beta}$ is a user-specified mixture length scale.

5.5.4. Interphase Drag for the Free Surface Model

In the free surface model, interphase drag is calculated in the same way as for the mixture model (see *Interphase Drag for the Mixture Model* (p. 180)), except that the interfacial area density is given by:

$$A_{\alpha\beta} = \left|\nabla r_\alpha\right| \quad \textbf{(5–68)}$$

When more than two phases are present, this is generalized as follows:

$$A_{\alpha\beta} = \frac{2\,\left|\nabla r_\alpha\right|\,\left|\nabla r_\beta\right|}{\left|\nabla r_\alpha\right| + \left|\nabla r_\beta\right|} \quad \textbf{(5–69)}$$

[*The Free Surface Model* (p. 169)]

5.5.5. Lift Force

The lift force acts perpendicular to the direction of relative motion of the two phases.

CFX contains a model for the shear-induced lift force acting on a dispersed phase in the presence of a rotational continuous phase. This is given by:

$$\begin{aligned} \boldsymbol{F}_c^L &= -\boldsymbol{F}_d^L = r_d \rho_c C_L \left(\boldsymbol{U}_d - \boldsymbol{U}_c\right) \times \boldsymbol{\omega}_c \\ \boldsymbol{\omega}_c &= \text{curl } \boldsymbol{U}_c \end{aligned} \quad \textbf{(5–70)}$$

where C_L is a non-dimensional lift coefficient. In a rotating frame of reference with rotation vector $\boldsymbol{\Omega}$, the lift force is given by:

$$\boldsymbol{F}_c^L = -\boldsymbol{F}_d^L = r_d \rho_c C_L (\boldsymbol{U}_d^* - \boldsymbol{U}_c^*) \times (\boldsymbol{\omega}_c^* + 2\boldsymbol{\Omega})$$
$$\boldsymbol{\omega}_c^* = \text{curl } \boldsymbol{U}_c^* \tag{5–71}$$

Currently, ANSYS CFX has the following built-in lift models:

- *The Saffman Mei Lift Force Model* (p. 182)
- *The Legendre and Magnaudet Lift Force Model* (p. 183)
- *The Tomiyama Lift Force Model* (p. 183)

5.5.5.1. The Saffman Mei Lift Force Model

This model is applicable mainly to the lift force on spherical solid particles, though it could be applied to liquid drops that are not significantly distorted. It is a generalization of the older Saffman model, which was applicable to a lower range of particle Reynolds numbers than the Saffman Mei model.

The lift coefficient is correlated in terms of both particle Reynolds number and vorticity Reynolds numbers:

$$\mathrm{Re_p} = \frac{\rho_\mathrm{c} |\boldsymbol{U}_\mathrm{c} - \boldsymbol{U}_\mathrm{d}| d_\mathrm{p}}{\mu_\mathrm{c}},\ \mathrm{Re}_\omega = \frac{\rho_\mathrm{c} \omega_\mathrm{c} d_\mathrm{p}^{\ 2}}{\mu_\mathrm{c}},\ \omega_\mathrm{c} = |\nabla \times \boldsymbol{U}_\mathrm{c}| \tag{5–72}$$

Saffman (1965, 1968) [86][170] correlated the lift force for low Reynolds number flow past a spherical particle as follows:

$$\boldsymbol{F}_\mathrm{c} = -\boldsymbol{F}_\mathrm{d} = \frac{3}{2\pi} \frac{\sqrt{\nu_\mathrm{c}}}{d_\mathrm{p}\sqrt{|\nabla \times \boldsymbol{U}_\mathrm{c}|}} C'_\mathrm{L} r_\mathrm{d} \rho_\mathrm{c} (\boldsymbol{U}_\mathrm{d} - \boldsymbol{U}_\mathrm{c}) \times (\nabla \times \boldsymbol{U}_\mathrm{c} + 2\Omega) \tag{5–73}$$

where $C'_\mathrm{L} = 6.46$, and $0 \le \mathrm{Re_p} \le \mathrm{Re}_\omega \le 1$. By inspection, Saffman's lift coefficient is related to the one adopted in ANSYS CFX as follows:

$$C_\mathrm{L} = \frac{3}{2\pi\sqrt{\mathrm{Re}_\omega}} C'_\mathrm{L} \tag{5–74}$$

Saffman's correlation was generalized by Mei and Klausner (1994) [87] to a higher range of particle Reynolds numbers, as follows:

$$C'_\mathrm{L} = \begin{cases} 6.46 \cdot f\left(\mathrm{Re_p}, \mathrm{Re}_\omega\right) & \text{for:}\mathrm{Re_p} < 40 \\ 6.46 \cdot 0.0524 \cdot \left(\beta \mathrm{Re_p}\right)^{1/2} & \text{for:}40 < \mathrm{Re_p} < 100 \end{cases} \tag{5–75}$$

where $\beta = \left(1/2\right)\left(\mathrm{Re}_\omega / \mathrm{Re_p}\right)$, and $f\left(\mathrm{Re_p}, \mathrm{Re}_\omega\right) = \left(1 - 0.3314\beta^{1/2}\right) \cdot e^{-0.1\mathrm{Re}_p} + 0.3314\beta^{1/2}$.

5.5.5.2. The Legendre and Magnaudet Lift Force Model

This model, as developed by Legendre & Magnaudet (1998) [171], is applicable mainly to the lift force of small diameter spherical fluid particles, though it could be applied to non-distorted liquid drops and bubbles. In contrast to the lift force model of Saffman-Mei for rigid solid particles, it accounts for the momentum transfer between the flow around the particle and the inner recirculation flow inside the fluid particle as caused by the fluid friction/stresses at the fluid interface. Therefore the predicted lift force coefficients are about a factor of 2-5 smaller than for rigid solid particles.

The range of validity given by Legendre & Magnaudet (1998) is as follows:

$$0.1 \le \mathrm{Re_p} \le 500 \qquad Sr = 2\beta \le 1 \tag{5–76}$$

The lift force coefficient is then predicted by:

$$C_{\mathrm{L}} = \sqrt{\left(C_{\mathrm{L,low\ Re}}\right)^2 + \left(C_{\mathrm{L,highRe}}\right)^2} \tag{5–77}$$

where

$$C_{\mathrm{L,low\ Re}} = \frac{6}{\pi^2}\left(\mathrm{Re_p Sr}\right)^{-1/2} J'(\epsilon)$$

$$C_{\mathrm{L,high\ Re}} = \frac{1}{2}\frac{1+16\mathrm{Re_p}^{-1}}{1+29\mathrm{Re_p}^{-1}} \tag{5–78}$$

$$\epsilon = \sqrt{\frac{2\beta}{\mathrm{Re_p}}}$$

$$J'\left(\epsilon\right) = \frac{J(\infty)}{\left(1+0.2\epsilon^{-2}\right)^{3/2}}$$

$$J(\infty) = 2.55$$

5.5.5.3. The Tomiyama Lift Force Model

This is a model applicable to the lift force on larger-scale deformable bubbles in the ellipsoidal and spherical cap regimes. Like the Grace and Ishii-Zuber models for drag force, it depends on Eotvos number. Hence, it requires specification of the surface tension between the dispersed and continuous phases. Its main important feature is prediction of the cross-over point in bubble size at which particle distortion causes a reversal of the sign of the lift force to take place. The lift coefficient is given by (Tomiyama 1998) [172]:

$$C_L = \begin{cases} \min\left[0.288 \tanh\left(0.121 \mathrm{Re}_p, f\left(\mathrm{Eo}'\right)\right)\right] & \mathrm{Eo}' \le 4 \\ f\left(\mathrm{Eo}'\right) & 4 < \mathrm{Eo}' \le 10 \\ -0.27 & 10 < \mathrm{Eo}' \end{cases} \tag{5–79}$$

where:

$$f\left(\mathrm{Eo}'\right) = 0.00105 \mathrm{Eo}'^3 - 0.0159 \mathrm{Eo}'^2 - 0.0204 \mathrm{Eo}' + 0.474$$

Eo' is a modified Eotvos number, based on the long axis, d_H, of the deformable bubble:

$$\mathrm{Eo}' = \frac{g\left(\rho_L - \rho_G\right) d_H^2}{\sigma}$$

$$d_H = d_p\left(1 + 0.163 \mathrm{Eo}^{0.757}\right)^{1/3}$$

$$\mathrm{Eo} = \frac{g\left(\rho_L - \rho_G\right) d_p^2}{\sigma}$$

The correlation has been slightly modified from Tomiyama's original form, following Frank et al. (2004) [173], whereby the value of C_L for $\mathrm{Eo}' > 10$ has been changed to -0.27 to ensure a continuous dependence on modified Eotvos number. Also, some publications omit the exponent of Eotvos number in the formula for d_H. The formula adopted here is taken from Wellek et al.[174].

5.5.6. Virtual Mass Force

The virtual mass force is proportional to relative phasic accelerations as follows:

$$\boldsymbol{F}_c^{VM} = -\boldsymbol{F}_d^{VM} = r_d \rho_c C_{VM}\left(\frac{D_d \boldsymbol{U}_d}{D t} - \frac{D_c \boldsymbol{U}_c}{D t}\right) \tag{5–80}$$

In a rotating frame of reference with rotation vector $\boldsymbol{\Omega}$, the virtual mass force in terms of $\boldsymbol{U}^*$ is modified by Coriolis theorem, and is given by:

$$\boldsymbol{F}_c^{VM} = -\boldsymbol{F}_d^{VM} = r_d \rho_c C_{VM}\left(\frac{D_d^* \boldsymbol{U}_d^*}{Dt} - \frac{D_c^* \boldsymbol{U}_c^*}{Dt} + 2\,\Omega \times \left(\boldsymbol{U}_d^* - \boldsymbol{U}_c^*\right)\right) \tag{5–81}$$

The non-dimensional virtual mass coefficient $C_{VM} = 0.5$ for inviscid flow around an isolated sphere. In general, C_{VM} depends on shape and particle concentration C_{VM} may be specified by the user as a constant or a CEL expression.

5.5.7. Wall Lubrication Force

Under certain circumstances, for example, bubbly upflow in a vertical pipe, the dispersed phase is observed to concentrate in a region close to the wall, but not immediately adjacent to the wall. This effect may be modeled by the wall lubrication force, which tends to push the dispersed phase away from the wall.

Currently, ANSYS CFX has the following wall lubrication force models:

- *The Antal Wall Lubrication Force Model* (p. 185)
- *The Tomiyama Wall Lubrication Force Model* (p. 186)
- *The Frank Wall Lubrication Force Model* (p. 186)

5.5.7.1. The Antal Wall Lubrication Force Model

The Antal model, given by Antal et al. (1991) [88], computes the wall lubrication force as:

$$\boldsymbol{F}_{\mathrm{WL}} = -C_{\mathrm{WL}} r_G \rho_{\mathrm{L}} \left|\boldsymbol{U}_{\mathrm{L}} - \boldsymbol{U}_{\mathrm{G}}\right|^2 \boldsymbol{n}_{\mathrm{W}} \tag{5–82}$$

where

- $C_{\mathrm{WL}} = \max\left\{0, \frac{C_{\mathrm{W1}}}{d_{\mathrm{p}}} + \frac{C_{\mathrm{W2}}}{y_{\mathrm{W}}}\right\}$

  The non-dimensional coefficients are defaulted to $C_{\mathrm{W1}} = -0.01$ and $C_{\mathrm{W2}} = 0.05$. You can change these values.
- r_G is the gas volume fraction.
- ρ_{L} is the liquid density.
- $\boldsymbol{n}_{\mathrm{W}}$ is the unit normal pointing away from the wall.
- $\boldsymbol{U}_{\mathrm{L}} - \boldsymbol{U}_{\mathrm{G}}$ is the relative velocity difference between phases, in the plane of the nearby wall surface (that is, orthogonal to $\boldsymbol{n}_{\mathrm{w}}$).
- d_{p} is the dispersed phase mean diameter.
- y_w is the distance to the nearest wall.

Note that this force is only active in a thin layer adjacent to the wall; it is only active up to a cut-off distance of:

$$y_{\mathrm{w}} \le -\left(C_{\mathrm{W2}}/C_{\mathrm{W1}}\right) d_{\mathrm{p}}$$

where $y_w = 5d_{\mathrm{p}}$ with default values of C_{W1} and C_{W2}.

Hence, this force will only be activated on a sufficiently fine mesh, and grid convergence can be expected only on extremely fine meshes.

5.5.7.2. The Tomiyama Wall Lubrication Force Model

Tomiyama (1998) [172] modified the wall lubrication force formulation of Antal, based on the results of experiments with the flow of air bubbles in glycerin in a pipe. The modification is as follows:

$$C_{\mathrm{WL}} = C_{\mathrm{W}}\left(Eo \right)\frac{d_{\mathrm{p}}}{2}\left(\frac{1}{{y_{\mathrm{W}}}^{2}} - \frac{1}{\left(D - y_{\mathrm{W}} \right)^{2}} \right) \tag{5–83}$$

Here, D is the pipe diameter. Hence, although the model was found to be superior to Antal's [88] (Frank et al. 2004 [173]), it is restricted to flows in pipe geometries. As is the case for the Tomiyama lift force correlation, the coefficient $C_{\mathrm{W}}\left(Eo \right)$ is dependent on the Eotvos number, and hence on the surface tension between the two phases. Frank et al (2004) modified this correlation slightly to ensure continuous dependence of the wall lubrication coefficient on Eotvos number:

$$C_{\mathrm{W}}\left(Eo \right) = \begin{cases} 0.47 & Eo < 1 \\ e^{-0.933Eo + 0.179} & 1 \le Eo \le 5 \\ 0.00599Eo - 0.0187 & 5 < Eo \le 33 \\ 0.179 & 33 < Eo \end{cases} \tag{5–84}$$

5.5.7.3. The Frank Wall Lubrication Force Model

Frank et al. ([177], [173]) generalized the Tomiyama model to produce the Frank Wall Lubrication Force model, which has no dependence on pipe diameter, and is given by:

$$C_{\mathrm{wall}} = C_{\mathrm{W}}\left(\mathrm{Eo} \right) \cdot \max\left\{ 0, \frac{1}{C_{\mathrm{WD}}} \cdot \frac{1 - \dfrac{y_{\mathrm{W}}}{C_{\mathrm{WC}}d_{\mathrm{p}}}}{y_{\mathrm{W}} \cdot \left(\dfrac{y_{\mathrm{W}}}{C_{\mathrm{WC}}d_{\mathrm{p}}} \right)^{p-1}} \right\} \tag{5–85}$$

Note:

- $C_{\mathrm{W}}\left(Eo \right)$ preserves the same dependence on Eotvos number as the Tomiyama model.
- The cut-off coefficient, C_{WC}, determines the distance relative to the particle diameter over which the force is active. $C_{\mathrm{WC}} = 5$ gives the same range as the Antal model with default constants.
- The damping coefficient, C_{WD}, determines the relative magnitude of the force. $C_{\mathrm{WD}} = 100$ gives the same behavior as the Antal model with default constants. However, Frank et al. found that such high

damping of the wall lubrication force was not able to sufficiently counterbalance the Tomiyama lift force in the near wall region.

- The power-law constant, p, makes the force fall off with a variable potential law relationship:

  $F_{\mathrm{WL}} \sim 1/{y_{\mathrm{W}}}^{p}$. It is recommended that p be in the range: 1.5 to 2.

- In extensive validation exercises by Frank et al. [177], the following model constants were determined in order to produce the best possible agreement with experimental data for vertical bubbly flow in pipes: $C_{\mathrm{WC}} = 10$, $C_{\mathrm{WD}} = 6.8$, $p = 1.7$.

5.5.8. Interphase Turbulent Dispersion Force

The topics in this section include:

- *Favre Averaged Drag Model* (p. 187)
- *Lopez de Bertodano Model* (p. 187)

5.5.8.1. Favre Averaged Drag Model

CFX implements a model for turbulent dispersion force, based on the Favre average of the interphase drag force [90].

$$\boldsymbol{F}_c^{TD} = -\boldsymbol{F}_d^{TD} = C_{TD}\, C_{cd} \frac{v_{tc}}{\sigma_{tc}} \left(\frac{\nabla r_d}{r_d} - \frac{\nabla r_c}{r_c} \right) \qquad \textbf{(5–86)}$$

Here, C_{cd} is the momentum transfer coefficient for the interphase drag force. Hence, the model clearly depends on the details of the drag correlation used σ_{tc} is the turbulent Schmidt number for continuous phase volume fraction, currently taken to be 0.9.

C_{TD} is a user-modifiable CEL multiplier. Its default value is unity.

5.5.8.2. Lopez de Bertodano Model

The model of Lopez de Bertodano (1991) [20] was one of the first models for the turbulent dispersion force:

$$M_c^{TD} = -M_d^{TD} = -C_{TD}\, \rho_c\, k_c\, \nabla r_c \qquad \textbf{(5–87)}$$

Unfortunately, it is not possible to recommend universal values of C_{TD} for this model. C_{TD} values of 0.1 to 0.5 have been used successfully for bubbly flow with bubble diameters of order a few millimeters. However, values up to 500 have been required for other situations. See Lopez de Bertodano [21] and Moraga et al. [91].

This model is included in CFX for historical back compatibility with CFX. However, the more universal Favre Averaged Drag model is recommended for all situations where an appropriate value of C_{TD} is unknown.

5.6. Solid Particle Collision Models

This section presents the theoretical background of the solid particle collision models implemented in CFX.

The following topic(s) will be discussed:

5.6.1. Solids Stress Tensor

Additional stresses due to inter-particle collisions are modeled using a collisional solids stress tensor in the solid phase momentum equation only:

$$\tau_{s\,ij} = -P_s\,\delta_{ij} + \mu_s \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} - \frac{2}{3}\frac{\partial U_k}{\partial x_k}\delta_{ij} \right) + \zeta_s \frac{\partial U_k}{\partial x_k}\delta_{ij} \qquad \textbf{(5–88)}$$

Here, P_s denotes solids pressure, μ_s denotes solids shear viscosity, and ζ_s denotes solids bulk viscosity. There are two important classes of models for these quantities:

5.6.1.1. Empirical Constitutive Equations

There exist wide classes of models where the constitutive elements of the solids stress are specified using empirical constitutive relations. See, for example, (Enwald et al. [97]). In many of these, the solids pressure, shear and bulk viscosities are simple functions of the solid phase volume fraction.

5.6.1.2. Kinetic Theory Models for the Solids Stress Tensor

These are a class of models, based on the kinetic theory of gases, generalized to take into account inelastic particle collisions. In these models, the constitutive elements of the solids stress are functions of the solid phase granular temperature, defined to be proportional to the mean square of fluctuating solid phase velocity due to inter-particle collisions:

$$\Theta_s = \frac{1}{3}\,\overline{u_s'^{\,2}} \qquad \textbf{(5–89)}$$

In the most general kinetic theory models, the granular temperature is determined from a transport equation. However, in many circumstances, it is possible to ignore the transport terms, and determine granular temperature from the resulting algebraic equation.

5.6.2. Solids Pressure

5.6.2.1. Empirical Constitutive Equations

The most popular constitutive equations for solids pressure are due to (Gidaspow [18]). These actually specify the solids pressure gradient, rather than solids pressure directly:

$$P_s = P_s\left(r_s\right) \Rightarrow \nabla P_s = G\left(r_s\right)\ \nabla r_s \tag{5–90}$$

$$G\left(r_s\right) = G_0\, e^{c\left(r_s - r_{sm}\right)} \tag{5–91}$$

Where $G\left(r_s\right)$ is the **Elasticity Modulus**, G_0 is the **Reference Elasticity Modulus**, c is the **Compaction Modulus**, and r_{sm} is the **Maximum Packing Parameter**.

The Gidaspow model is implemented with an option for specifying the **Reference Elasticity Modulus** and **Compaction Modulus**. There is also an option to specify the **Elasticity Modulus** $G\left(r_s\right)$ directly. There is also an option to specify the solids pressure directly. This permits more general constitutive relations than those where the solids pressure is a function of volume fraction only.

5.6.2.2. Kinetic Theory Models for Solids Pressure

The kinetic theory model for solids pressure is similar to the equation of state for ideal gases, modified to take account of inelastic collisions, and maximum solid packing.

$$P_s = \rho_s\, r_s\, \Theta_s \left(1 + 2\left(1 + e\right)\ g_0\, r_s\right) \tag{5–92}$$

Here, e denotes the coefficient of restitution for solid-solid collisions, and $g_0\left(r_s\right)$ denotes the radial distribution function. Popular models for the radial distribution function are given by:

Gidaspow (1994) [18]:

$$g_0\left(r_s\right) = 0.6\left(1 - \left(r_s / r_{sm}\right)^{1/3}\right)^{-1} \tag{5–93}$$

Lun and Savage (1986) [100]:

$$g_0\left(r_s\right) = \left(1 - \left(r_s / r_{sm}\right)\right)^{-2.5\, r_{sm}} \tag{5–94}$$

Note that the radial distribution function tends to infinity as $r_s \rightarrow r_{sm}$. The singularity is removed in CFX by setting:

$$g_0\left(r_s\right) = C_0 + C_1\left(r_s - r_c\right) + C_2\left(r_s - r_c\right)^2 + C_3\left(r_s - r_c\right)^3,\ \left(r \geq r_c\right) \tag{5–95}$$

where $r_c = r_{sm} - 0.001$ and:

$$
\begin{aligned}
C_0 &= 1079 \\
C_1 &= 1.08 \times 10^6 \\
C_2 &= 1.08 \times 10^9 \\
C_3 &= 1.08 \times 10^{12}
\end{aligned}
\qquad (5–96)
$$

5.6.3. Solids Bulk Viscosity

5.6.3.1. Constitutive Equation Models

Most simple constitutive equation models ignore the solids bulk viscosity. However, it is possible for the user to specify a model for this, using a CEL expression, if required. Note that any solids volume fraction dependence must be included in the CEL expression.

5.6.3.2. Kinetic Theory Models for Solids Bulk Viscosity

There is general agreement on the correct kinetic theory form for the solids bulk viscosity (Lun et al. 1984 [99]).

$$
\zeta_s = \frac{4}{3} r_s^2 \rho_s d_p g_0 (1+e) \sqrt{\frac{\Theta}{\pi}} \qquad (5–97)
$$

5.6.4. Solids Shear Viscosity

5.6.4.1. Constitutive Equation Models

The simplest constitutive equation model for solids shear viscosity was presented by Miller and Gidaspow (1992) [106]. They successfully modeled gas-solid flow in a riser using a solids shear viscosity linearly proportional to the solids phase volume fraction.

$$
\mu_s = 5.35\ [\mathrm{Poise}] \times r_s \qquad (5–98)
$$

Note that their constant of proportionality is dimensional, and is likely to require modification for different fluid-solid material properties.

More complex models for solids shear stress allow the shear stress to become very large in the limit of maximum packing. A wide range of such models is discussed in the review article by Enwald et al [97].

It is possible for the user to implement any of these models, using a CEL expression for the solids shear viscosity. Note that any solids volume fraction dependence must be included in the CEL expression.

5.6.4.2. Kinetic Theory Models for Solids Shear Viscosity

Typically, the shear viscosity is expressed as a sum of at least two contributions: the kinetic and collisional contributions:

$$\mu_s = \mu_{s,col} + \mu_{s,kin} \tag{5–99}$$

There is wide agreement on the correct form of the collisional contribution. As in the kinetic theory of gases, it is proportional to the square root of the granular temperature:

$$\mu_{s,col} = \frac{4}{5} r_s^2 \rho_s d_p g_0 (1+e) \sqrt{\frac{\Theta}{\pi}} \tag{5–100}$$

However, there are many proposals in the literature for the correct form of the kinetic contribution. For example:

Gidaspow (1994) [18]:

$$\mu_{s,kin} = \frac{5\sqrt{\pi}}{48} \frac{\rho_s d_p}{(1+e)\, g_0} \left(1 + \frac{4}{5} (1+e)\, g_0 r_s \right)^2 \sqrt{\Theta} \tag{5–101}$$

Lun and Savage (1986) [100]:

$$\mu_{s,kin} = \frac{5\sqrt{\pi}}{96} \rho_s d_p \left(\frac{1}{\eta g_0} + \frac{8}{5} r_s \right) \left(\frac{1 + \frac{8}{5} \eta (3\eta - 2)\, r_s g_0}{2 - \eta} \right) \sqrt{\Theta} \tag{5–102}$$

Here, $\eta = \frac{1}{2}(1+e)$.

Kinetic contributions are omitted from ANSYS CFX.

5.6.5. Granular Temperature

ANSYS CFX is restricted to models where the granular temperature Θ_s is determined algebraically.

5.6.5.1. Algebraic Equilibrium Model

Θ_s may be specified directly by the user, or from the assumption of local equilibrium in a transport equation model. The latter is based on:

$$\text{Production} = \text{Dissipation} \Rightarrow \tau_{s\,ij} \frac{\partial U_i}{\partial x_j} = \gamma_s \tag{5–103}$$

where $\tau_{s\,ij}$ denotes the solids shear stress tensor *Equation 5–88* (p. 188), and:

$$\gamma_s = 3\,(1-e^2)\; r_s^2\,\rho_s\,g_0\,\Theta_s\left(\frac{4}{d_s}\left(\sqrt{\frac{\Theta_s}{\pi}} - \frac{\partial U_k}{\partial x_k}\right)\right) \tag{5–104}$$

Expand the production term:

$$\tau_{s\,ij}\frac{\partial U_i}{\partial x_j} = -P_s\frac{\partial U_k}{\partial x_k} + \mu_s\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\frac{\partial U_i}{\partial x_j} + \left(\zeta_s - \frac{2}{3}\mu_s\right)\left(\frac{\partial U_k}{\partial x_k}\right)^2 \tag{5–105}$$

$$\Rightarrow \text{Production} = -P_s D + \mu_s S^2 + \lambda_s D^2 \tag{5–106}$$

where:

$$\lambda_s = \zeta_s - \frac{2}{3}\mu_s \quad D = \frac{\partial U_k}{\partial x_k} \quad S^2 = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)^2 \tag{5–107}$$

In order to determine Θ_s from *Equation 5–103* (p. 191), it is useful to take into account the dependence of solids pressure and shear bulk viscosities on Θ_s. You have:

$$P_s \propto \Theta_s \qquad\qquad \mu_s, \zeta_s \propto \Theta_s^{1/2} \tag{5–108}$$

so you may write:

$$P_s = P_s^{(0)}\;\Theta_s \qquad\qquad \mu_s = \mu_s^{(0)}\;\Theta_s^{1/2} \tag{5–109}$$

$$\zeta_s = \zeta_s^{(0)}\;\Theta_s^{1/2} \qquad\qquad \lambda_s = \lambda_s^{(0)}\;\Theta_s^{1/2} \tag{5–110}$$

Hence, substituting into *Equation 5–106* (p. 192), you may express the production term in terms of granular temperature as follows:

$$\begin{aligned} \text{Production} &= \left(\lambda_s^{(0)}\;D^2 + \mu_s^{(0)}\;S^2\right)\Theta_s^{1/2} - P_s^{(0)}\;D\,\Theta_s \\ &= A_P\,\Theta_s^{1/2} - B_P\,\Theta_s \end{aligned} \tag{5–111}$$

where:

$$A_P = \lambda_s^{(0)}\;D^2 + \mu_s^{(0)}\;S^2 \ge 0 \qquad B_P = P_s^{(0)}\;D \tag{5–112}$$

Similarly, the dissipation term *Equation 5–104* (p. 192) may be simplified as follows:

$$\text{Dissipation} = A_D\, \Theta_s^{3/2} - B_D\, \Theta_s \tag{5–113}$$

where:

$$\begin{aligned} A_D &= \frac{4}{d_s \sqrt{\pi}}\, C_D \geq 0 \\ B_D &= C_D\, D \\ C_D &= 3\,(1-e^2)\; r_s^2\, \rho_s\, g_0 \geq 0 \end{aligned} \tag{5–114}$$

Equating *Equation 5–111* (p. 192) and *Equation 5–113* (p. 193), and dividing by $\Theta_s^{1/2}$ gives a quadratic equation for Θ_s:

$$A_D\, \Theta_s + (B_P - B_D)\; \Theta_s^{1/2} - A_P = 0 \tag{5–115}$$

Note that A_D is strictly positive if, and only if, the coefficient of restitution is strictly less than unity. In this case, in view of the fact that $A_P \geq 0$, *Equation 5–115* (p. 193) has a unique positive solution:

$$\Theta_s^{1/2} = \frac{B_D - B_P + \sqrt{(B_D - B_P)^2 + 4\, A_D\, A_P}}{2\, A_D} \tag{5–116}$$

The Algebraic Equilibrium model has the flaw that unphysically large granular temperatures can be generated in regions of low solid particle volume fraction. To circumvent this, specify an upper bound for the granular temperature. The square of the mean velocity scale is a reasonable estimate for this.

5.6.5.2. Zero Equation Model

The Zero Equation Model implements the simpler algebraic model of Ding and Gidaspow [98].

$$\Theta_s = \frac{1}{15\,(1-e)}\, d_s^2\, S^2 \tag{5–117}$$

5.7. Interphase Heat Transfer

In the multiphase model, there are separate enthalpy and temperature fields for each phase.

5.7.1. Phasic Equations

The multiphase version of the total energy equation generalizes *Equation 1–86* (p. 26):

$$\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha h_{\alpha,\mathrm{tot}}\right) - r_\alpha \frac{\partial p}{\partial t} + \nabla \bullet \left(r_\alpha \rho_\alpha \boldsymbol{U}_\alpha h_{\alpha,\mathrm{tot}}\right)$$
$$= \nabla \bullet (r_\alpha \lambda_\alpha \nabla T_\alpha) + r_\alpha \nabla \bullet (\boldsymbol{U}_\alpha \bullet \tau_\alpha) + S_{E\alpha} + Q_\alpha \tag{5–118}$$
$$+ \sum_{\beta=1}^{N_p} \left(\Gamma_{\alpha\beta}^{+} h_{\beta s,\mathrm{tot}} - \Gamma_{\beta\alpha}^{+} h_{\alpha s,\mathrm{tot}}\right)$$

The multiphase version of the thermal energy equation for static enthalpy (incompressible and low speed compressible flows only) generalizes *Equation 1–93* (p. 27):

$$\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha e_\alpha\right) + \nabla \cdot \left(r_\alpha \rho_\alpha \boldsymbol{U}_\alpha e_\alpha\right)$$
$$= \nabla \cdot (r_\alpha \lambda_\alpha T_\alpha) + r_\alpha \tau_\alpha : \nabla \boldsymbol{U}_\alpha + S_{E\alpha} + Q_\alpha \tag{5–119}$$
$$+ \Sigma_{\beta=1}^{N_p} \left(\Gamma_{\alpha\beta}^{+} e_{\beta s} - \Gamma_{\beta a}^{+} e_{\alpha s}\right)$$

where:

- h_α, T_α, λ_α, denote the static enthalpy, the temperature, and the thermal conductivity of phase α.
- e_α represents the internal energy of phase α, although the variable is still called Static Enthalpy in CFD-Post.
- $S_{E\alpha}$ describes external heat sources. For details, see *Sources* (p. 54).
- Q_α denotes interphase heat transfer to phase α across interfaces with other phases. For details, see *Inhomogeneous Interphase Heat Transfer Models* (p. 194).
- The term

  $$\left(\Gamma_{\alpha\beta}^{+} h_{\beta s} - \Gamma_{\beta a}^{+} h_{\alpha s}\right) \tag{5–120}$$

  represents heat transfer induced by interphase mass transfer. For details, see *Interphase Mass Transfer* (p. 216).

5.7.2. Inhomogeneous Interphase Heat Transfer Models

Interphase heat transfer occurs due to thermal non-equilibrium across phase interfaces. The total heat per unit volume transferred to phase α due to interaction with other phases is denoted Q_α, and is given by:

$$Q_\alpha = \sum_{\beta \neq \alpha} Q_{\alpha\beta} \tag{5–121}$$

where:

$$Q_{\alpha\beta} = -Q_{\beta\alpha} \Rightarrow \sum_{\alpha} Q_{\alpha} = 0 \tag{5–122}$$

5.7.2.1. Overall Heat Transfer Coefficients

Heat transfer across a phase boundary is usually described in terms of an overall **heat transfer coefficient** $h_{\alpha\beta}$, which is the amount of heat energy crossing a unit area per unit time per unit temperature difference between the phases. Thus, the rate of heat transfer, $Q_{\alpha\beta}$, per unit time across a phase boundary of interfacial area per unit volume $A_{\alpha\beta}$, from phase β to phase α, is:

$$Q_{\alpha\beta} = h_{\alpha\beta} A_{\alpha\beta} \left(T_{\beta} - T_{\alpha}\right) \tag{5–123}$$

This may be written in a form analogous to momentum transfer:

$$Q_{\alpha\beta} = c_{\alpha\beta}^{(h)} \left(T_{\beta} - T_{\alpha}\right) \tag{5–124}$$

where the volumetric heat transfer coefficient, $c_{\alpha\beta}^{(h)}$, is modeled using the correlations described below.

5.7.2.2. Particle Model Correlations

For particle model, the volumetric heat transfer coefficient is modeled as:

$$c_{\alpha\beta}^{(h)} = h_{\alpha\beta} A_{\alpha\beta} \tag{5–125}$$

Hence, the interfacial area per unit volume and the heat transfer coefficient $h_{\alpha\beta}$ are required.

More information on interfacial area density calculation is available.

It is often convenient to express the heat transfer coefficient in terms of a dimensionless Nusselt number:

$$h = \frac{\lambda\,\mathrm{Nu}}{d} \tag{5–126}$$

In the particle model, the thermal conductivity scale λ is taken to be the thermal conductivity of the continuous phase, and the length scale d is taken to be the mean diameter of the dispersed phase:

$$h_{\alpha\beta} = \frac{\lambda_{\alpha}\,\mathrm{Nu}_{\alpha\beta}}{d_{\beta}} \tag{5–127}$$

For laminar forced convection around a spherical particle, theoretical analysis shows that $\mathrm{Nu} = 2$. For a particle in a moving incompressible Newtonian fluid, the Nusselt number is a function of the particle Reynolds number Re and the surrounding fluid Prandtl number $\mathrm{Pr} = \mu_{\alpha} C_{P\alpha}/\lambda_{\alpha}$.

Additional information on models in ANSYS CFX is available in Particle Model Correlations for Overall Heat Transfer Coefficient in the *CFX-Solver Modeling Guide*. Some additional details for the Interface Flux model are provided below.

- **Interface Flux**

  The heat flux coefficients for both fluids and the interfacial heat flux value, F_{12}, from Fluid 1 to Fluid 2 (Fluid 1 is the fluid to appear on the left of the Fluid Pairs list) are specified. F_{12} is the rate of heat transfer per unit time per unit interfacial area from phase 1 to phase 2. Hence, the heat transferred to fluid 2 from fluid 1 per unit volume is given by:

$$Q_{21} = -Q_{12} = A_{12} F_{12} \tag{5–128}$$

  F_{12} may be given as a constant or an expression.

  Typically, F_{12} will be a function of the fluid 1 and fluid 2 temperature fields, and possibly other variables. In this case, the user may accelerate convergence of the coupled solver by also specifying optional fluid 1 and fluid 2 heat flux coefficients.

$$h_1 \approx \frac{\partial F_{12}}{\partial T_1} \geq 0 \,, \quad h_2 \approx \frac{\partial F_{21}}{\partial T_2} \geq 0 \tag{5–129}$$

  The solver takes the absolute value of these flux coefficients to ensure that they are positive. This is required for numerical stability. The partial derivatives need not be computed exactly; it is sufficient for the specified coefficients to simply approximate the partial derivatives. Specification of heat flux coefficients only affects the convergence rate to the solution of the coupled heat transfer equations, it does not affect the accuracy of the converged solution.

  For example, the simple model using a heat transfer coefficient multiplied by a bulk temperature difference my be recovered using:

$$F_{12} = -F_{21} = h\,(T_1 - T_2)\,, \quad h_1 = h_2 = h$$

5.7.2.3. Mixture Model Correlations

When using the mixture model, the Nusselt number is defined in terms of a mixture conductivity scale and the mixture length scale:

$$h_{\alpha\beta} = \frac{\lambda_{\alpha\beta}\,\mathrm{Nu}_{\alpha\beta}}{d_{\alpha\beta}}\,, \quad \lambda_{\alpha\beta} = r_\alpha \lambda_\alpha + r_\beta \lambda_\beta \tag{5–130}$$

For details, see Mixture Model Correlations for Overall Heat Transfer Coefficient in the *CFX-Solver Modeling Guide*.

5.7.2.4. The Two Resistance Model

There are special situations where the use of an overall heat transfer coefficient is not sufficient to model the interphase heat transfer process. A more general class of models considers separate heat transfer processes either side of the phase interface. This is achieved by using two heat transfer coefficients defined on each side of the phase interface.

Defining the sensible heat flux to phase α from the interface as:

$$q_\alpha = h_\alpha (T_s - T_\alpha) \quad \textbf{(5–131)}$$

and the sensible heat flux to phase β from the interface as:

$$q_\beta = h_\beta \left(T_s - T_\beta\right) \quad \textbf{(5–132)}$$

where h_α and h_β are the phase α and phase β heat transfer coefficients respectively. T_s is interfacial temperature, and it is assumed to be the same for both phases.

The fluid-specific Nusselt number is defined as:

$$\mathrm{Nu}_\alpha = \frac{h_\alpha d_{\alpha\beta}}{\lambda_\alpha} \quad \textbf{(5–133)}$$

where λ_α is the thermal conductivity of fluid α, and $d_{\alpha\beta}$ is the interfacial length scale (the mean particle diameter for the Particle Model, and the mixture length scale for the Mixture Model).

In the absence of interphase mass transfer, you must have overall heat balance $q_\alpha + q_\beta = 0$. This condition determines the interfacial temperature:

$$T_s = \frac{h_\alpha T_\alpha + h_\beta T_\beta}{h_\alpha + h_\beta} \quad \textbf{(5–134)}$$

It also determines the interphase heat fluxes in terns of an overall heat transfer coefficient:

$$q_\alpha = -q_\beta = h_{\alpha\beta}\left(T_\beta - T_\alpha\right), \quad \frac{1}{h_{\alpha\beta}} = \frac{1}{h_\alpha} + \frac{1}{h_\beta} \quad \textbf{(5–135)}$$

Hence, in the absence of interphase mass transfer, the two resistance model is somewhat superfluous, as it may be implemented using a user-specified overall heat transfer coefficient.

It is possible to specify a zero resistance condition on one side of the phase interface. This is equivalent to an infinite fluid specific heat transfer coefficient $h_\alpha \rightarrow \infty$. Its effect is to force the interfacial temperature to be the same as the phase temperature, $T_s = T_\alpha$.

Modeling advice is available in Two Resistance Model for Fluid Specific Heat Transfer Coefficients in the *CFX-Solver Modeling Guide*.

5.7.3. Homogeneous Heat Transfer in Multiphase Flow

For all transport processes other than heat transfer, the shared field is the same as the transported quantity in the equation. However, in the case of heat transfer, it is temperature which is shared but enthalpy which is transported. Hence, CFX does not solve a bulk enthalpy equation, but rather solves a separate enthalpy equation for each fluid with a large interphase heat transfer term which forces the temperature fields to be the same.

The equations solved are identical to the phasic equations for full multiphase described above. For homogeneous heat transfer model, the interphase heat transfer coefficient is not modeled by any of the correlations used in full multiphase. Instead it is chosen to be large relative to the other transported processes in the equation, thereby ensuring the phasic temperatures are the same.

5.8. Multiple Size Group (MUSIG) Model

The MUSIG (Multiple Size Group) model has been developed to handle polydispersed multiphase flows. By polydispersed, it means that the dispersed phase has a large variation in size. One of the attributes of polydispersed multiphase flow is that the different sizes of the dispersed phases interact with each other through the mechanisms of breakup and coalescence. Population balance is a well-established method for calculating the size distribution of a polydispersed phase, including breakup and coalescence effects. MUSIG provides a framework in which the population balance method can be incorporated into three-dimensional CFD calculations.

5.8.1. Model Derivation

5.8.1.1. Population Balance Equation

The starting point for the MUSIG model is the population balance equation. Let $n\left(v,t\right)$ represent the number density of particles of mass m at time t. Then the population balance equation is:

$$\frac{\partial}{\partial t}n\left(m,t\right)+\frac{\partial}{\partial x^{i}}\left(U^{i}\left(m,t\right)\,n\left(m,t\right)\right)=B_B-D_B+B_C-D_C \tag{5–136}$$

where B_B, D_B, B_C, and D_C respectively represent the birth rate due to breakup of larger particles, the death rate due to breakup into smaller particles, the birth rate due to coalescence of smaller particles, and the death rate due to coalescence with other particles. These rates may further be expressed as:

$$B_B = \int_m^\infty g(\varepsilon\ ;\ m)\ n(\varepsilon,t)\ d\varepsilon \tag{5–137}$$

$$D_B = n(m,t)\ \int_0^m g(m\ ;\ \varepsilon)\ d\varepsilon \tag{5–138}$$

$$B_C = \frac{1}{2}\int_0^m Q(m-\varepsilon\ ;\ \varepsilon)\ n(m-\varepsilon,t)\ n(m,t)\ d\varepsilon \tag{5–139}$$

$$D_C = n(m,t)\ \int_0^\infty Q(m\ ;\ \varepsilon)\ n(\varepsilon,t)\ dt \tag{5–140}$$

where $g(m\ ;\ \varepsilon)$ represents the specific breakup rate (the rate at which particles of mass m break into particles of mass ε and $m-\varepsilon$) and $Q(m\ ;\ \varepsilon)$ represents the specific coalescence rate (the rate at which particles of mass m coalesce with particles of mass ε to form particles of mass $m+\varepsilon$.

5.8.1.2. Size Fraction Equations

The next step of the MUSIG model is to discretize *Equation 5–136* (p. 198) into size groups, or bins. Let N_i represent the number density of size group i:

$$N_i(t)\ = \int_{m_{i-1/2}}^{m_{i+1/2}} n(m,t)\ dm \tag{5–141}$$

Also define the mass and volume fraction of size group i to be m_i and r_i, respectively, and recognize that $m_i N_i = \rho_i r_i$. Now integrate *Equation 5–136* (p. 198) over the bin size dimension and multiply by m_i to give:

$$\frac{\partial\ (m_i N_i(t)\ )}{\partial t} + \frac{\partial}{\partial x^i}\left(U_i^i(t)\ N_i(t)\ \right) = S_i \tag{5–142}$$

or:

$$\frac{\partial\ \left(\rho_i r_i\right)}{\partial t} + \frac{\partial}{\partial x^i}\left(\rho_i r_i U_i^i\right) = S_i \tag{5–143}$$

Defining the size fraction $f_i = r_i/r_d$, this equation may also be written as:

$$\frac{\partial}{\partial t}\left(\rho_i r_d f_i\right) + \frac{\partial}{\partial x^i}\left(\rho_i r_d U_i^i f_i\right) = S_i \quad (5–144)$$

which is the size fraction equation used by the MUSIG model. A further simplification is to assume that all size groups share the same density ρ_d and velocity u_d^i yielding the homogeneous MUSIG model:

$$\frac{\partial}{\partial t}\left(\rho_d r_d f_i\right) + \frac{\partial}{\partial x^i}\left(\rho_d r_d U_d^i f_i\right) = S_i \quad (5–145)$$

5.8.1.3. Source Terms

The contribution of the birth rate due to breakup of larger particles to the source term in Equation 5–145 (p. 200) is:

$$\begin{aligned} B_{Bi} &= m_i \int_{m_{i-1/2}}^{m_{i+1/2}} B_B \, dm \\ &= m_i \sum_{j>i} g\left(m_j ; m_i\right) N_j \\ &= \rho_d r_d \left(\sum_{j>i} g\left(m_j ; m_i\right) f_j \right) \end{aligned} \quad (5–146)$$

Similarly, the contribution of the death rate due to breakup into smaller particles is:

$$D_{Bi} = \rho_d r_d \left(f_i \sum_{j<1} g\left(m_i ; m_j\right) \right) \quad (5–147)$$

Note that the total source due to breakup is zero when summed over all size groups:

$$\sum_i \left(B_{Bi} - D_{Bi}\right) = 0 \quad (5–148)$$

For the discretized coalescence sources, you must define the coalescence mass matrix X_{jki} as the fraction of mass due to coalescence between groups j at t which goes into group i:

$$X_{jki} = \begin{cases} \dfrac{\left(m_j + m_k\right) - m_{i-1}}{m_i - m_{i-1}} & \text{if} \quad m_{i-1} < m_j + m_k < m_i \\ \dfrac{m_{i+1} - \left(m_j + m_k\right)}{m_{i+1} - m_i} & \text{if} \quad m_i < m_j + m_k < m_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{5–149}$$

The contribution of the birth rate due to coalescence of smaller particles to the source term in *Equation 5–145* (p. 200) is:

$$\begin{aligned} B_{Ci} &= m_i \int_{m_{i-1/2}}^{m_{i+1/2}} B_C \, dm \\ &= m_i \left(\frac{1}{2} \sum_{j \le i} \sum_{k \le i} Q\left(m_j \, ; \, m_k\right) X_{jki} N_j N_k \right) \\ &= \left(\rho_d r_d\right)^2 \left(\frac{1}{2} \sum_{j \le i} \sum_{k \le i} Q\left(m_j \, ; \, m_k\right) X_{jki} f_j f_k \frac{m_j + m_k}{m_j m_k} \right) \end{aligned} \tag{5–150}$$

Similarly, the contribution of the death rate due to coalescence into larger groups is:

$$D_{Ci} = \left(\rho_d r_d\right)^2 \left(\sum_j Q\left(m_i \, ; \, m_j\right) f_i f_j \frac{1}{m_j} \right) \tag{5–151}$$

Note that this formulation for the coalescence source terms guarantees that the total source to coalescence is zero when summed over all size groups:

$$\sum_i \left(B_{Ci} - D_{Ci}\right) = 0 \tag{5–152}$$

This follows from the requirement that $Q\left(m_i \, ; \, m_j\right) = Q\left(m_d \, ; \, m_i\right)$ together with the following property of the mass matrix for all j, k:

$$\sum_i X_{jki} = 1 \tag{5–153}$$

5.8.2. Size Group Discretization

We now consider how the size distribution is discretized into a predefined set of size groups. Note that the group sizes are represented by mass rather than diameter or volume, because this leads to simpler equations for variable-density fluids. For setup convenience, however, the size group discretization is controlled by user-defined diameters; these diameters are converted to masses using the fluid density (for constant density fluids) or a user-specified reference density (for variable density fluids). The diameter and mass of a particular group are related by:

$$m = \rho_d \frac{\pi d^3}{6} \tag{5–154}$$

There are three built-in recipes for creating size groups from a user-defined minimum and maximum diameter: equal mass, equal diameter, and geometric.

5.8.2.1. Equal Mass Discretization

In this case, the minimum and maximum mass are calculated from the minimum and maximum diameter using *Equation 5–154* (p. 202). The mass of group i is calculated using:

$$m_i = m_{\min} + \Delta m \left(i - \frac{1}{2} \right) \tag{5–155}$$

$$\Delta m = \frac{m_{\max} - m_{\min}}{N} \tag{5–156}$$

5.8.2.2. Equal diameter discretization

In this case, the diameter of group i is calculated from:

$$d_i = d_{\min} + \Delta d \left(i - \frac{1}{2} \right) \tag{5–157}$$

$$\Delta d = \frac{d_{\max} - d_{\min}}{N} \tag{5–158}$$

and the group mass is calculated from *Equation 5–154* (p. 202).

5.8.2.3. Geometric Mass Discretization

In this case, a geometrical progression for mass as follows:

$$m_i = m_{\min} + \frac{3}{4} \frac{m_{\max} - m_{\min}}{2^{N-i}} \tag{5–159}$$

5.8.2.4. Comparison

The following table compares diameters for the three discretization options for a polydispersed fluid having $d_{\min} = 0 \text{ mm}$ and $d_{\max} = 2 \text{ mm}$.

The equal mass distribution is weighted towards the largest bubbles the most, the geometric distribution toward the smallest bubbles, and the equal diameter distribution is in between.

Group	Equal Mass	Equal Diameter	Geometric Mass
1	585	50	22.5
2	843	150	28.4
3	1000	250	35.8
4	1118	350	45.1
5	1216	450	56.8
6	1300	550	71.5
7	1375	650	90.1
8	1442	750	114
9	1503	850	143
10	1560	950	180
11	1613	1050	227
12	1663	1150	286
13	1710	1250	360
14	1754	1350	454
15	1796	1450	572
16	1837	1550	721
17	1875	1650	908
18	1912	1750	1144
19	1948	1850	1442
20	1983	1950	1817

5.8.3. Breakup Models

Breakup kernels are often expressed as a function of the breakup fraction:

$$f_{BV} = \frac{m_j}{m_i} \tag{5–160}$$

CFX-Solver supports two breakup models.

5.8.3.1. Luo and Svendsen Model

Luo and Svendsen [61] developed a theoretical model for the breakup of drops and bubbles in turbulent suspensions. The model is based on the theory of isotropic turbulence and probability. The breakup kernel is modeled as:

$$g\left(m_i\,;\,f_{BV}\,m_i\right) = 0.923\,F_B\,(1 - r_d)\left(\frac{\varepsilon_c}{d_i^2}\right)^{1/3}\int_{\xi_{\min}}^{1}\frac{(1+\xi)^2}{\xi^{11/3}}\,e^{-\chi}\,d\xi \tag{5–161}$$

where:

$$\chi = \frac{12\left(f_{BV}^{2/3} + \left(1 - f_{BV}\right)^{2/3} - 1\right)\sigma}{\beta\,\rho_c\,\varepsilon_c^{2/3}\,d_i^{5/3}\,\xi^{11/3}} \tag{5–162}$$

ξ is the dimensionless size of eddies in the inertial subrange of isotropic turbulence. The lower limit of the integration is given by:

$$m = \frac{\pi}{6}\rho_d d^3 \tag{5–163}$$

where

$$\eta = \left(\frac{1}{\varepsilon_c}\,\nu_c^3\right)^{1/4} \tag{5–164}$$

In addition, F_B is a calibration coefficient, $\beta = 2$, ε_c is the continuous-phase eddy dissipation rate, ν_c is the continuous-phase kinematic viscosity, and σ is the surface tension coefficient.

5.8.3.2. User-Defined Models

A custom model for the breakup kernel $g\left(m_i\,;\,m_j\right)$ may also be provided. The model may be a CEL expression or User Routine and may be a function of the diameter and/or mass of groups i and j as well as any fluid variable.

5.8.4. Coalescence Models

CFX-Solver supports two coalescence models.

5.8.4.1. Prince and Blanch Model

The model of Prince and Blanch [62] assumes that the coalescence of two bubbles occurs in three steps. First, the bubbles collide trapping a small amount of liquid between them. This liquid film then drains until the liquid film separating the bubbles reaches a critical thickness. The film them ruptures and the bubbles join together.

The coalescence kernel is therefore modeled by a collision rate of two bubbles and a collision efficiency relating to the time required for coalescence:

$$Q\left(m_i\ ;\ m_j\right) = \left(\theta_{ij}^T + \theta_{ij}^B + \theta_{ij}^S\right)\ \eta_{ij} \tag{5–165}$$

The collision efficiency is modeled by comparing the time required for coalescence t_{ij} with the actual contact time during the collision τ_{ij}:

$$\eta_{ij} = e^{-t_{ij}/\tau_{ij}} \tag{5–166}$$

$$t_{ij} = \left(\frac{\rho_c\, r_{ij}^3}{16\,\sigma}\right)^{1/2} \ln\left(\frac{h_0}{h_f}\right) \tag{5–167}$$

$$\tau_{ij} = \frac{r_{ij}^{2/3}}{\varepsilon_c^{1/3}} \tag{5–168}$$

where h_0 is the initial film thickness, h_f is the critical film thickness when rupture occurs, and r_{ij} is the equivalent radius:

$$r_{ij} = \left(\frac{1}{2}\left(\frac{1}{r_i} + \frac{1}{r_j}\right)\right)^{-1} \tag{5–169}$$

The turbulent contributions to collision frequency are modeled as:

$$\theta_{ij}^T = F_{CT}\, S_{ij}\left(u_{ti}^2 + u_{tj}^2\right)^{1/2} \tag{5–170}$$

where the cross-sectional area of the colliding particles is defined by:

$$S_{ij} = \frac{\pi}{4}\left(d_i + d_j\right)^2 \tag{5–171}$$

the turbulent velocity is given by:

$$u_{ti} = \sqrt{2}\ \varepsilon_c^{1/3}\ d_i^{1/3} \tag{5–172}$$

and F_{CT} is a calibration factor. The buoyancy contribution to collision frequency is modeled as:

$$\theta_{ij}{}^{B} = F_{C\,B}\, S_{ij}\ \left| U_{rj} - U_{ri} \right| \tag{5–173}$$

where:

$$U_{ri} = \sqrt{\frac{2.14\,\sigma}{\rho_c\, d_i} + 0.505\; g\; d_i} \tag{5–174}$$

and F_{CB} is a calibration factor.

The shear contribution to collision frequency is currently neglected.

5.8.4.2. User-Defined Models

A custom model for the coalescence rate kernel $Q\left(m_i\ ;\ m_j\right)$ may also be provided. The model may be a CEL expression or User Routine involving the diameter and/or mass of groups i and j as well as any fluid variable. Note that the model must give symmetric coalescence rates ($Q\left(m_i\ ;\ m_j\right) = Q\left(m_j\ ;\ m_i\right)$).

5.9. The Algebraic Slip Model

Models for algebraic slip were first introduced by Ishii [59] Manninen and Taivassalo [60], provide a more general formulation which forms the basis for the implementation in CFX.

5.9.1. Phasic Equations

A starting point is to review the equations for multiphase flow. The continuity equation for phase α is:

$$\frac{\partial\left(\rho_\alpha r_\alpha\right)}{\partial t} + \frac{\partial\left(\rho_\alpha r_\alpha u_\alpha^{\,i}\right)}{\partial x^{i}} = 0 \tag{5–175}$$

and the momentum equation is:

$$\frac{\partial\left(\rho_\alpha r_\alpha u_\alpha^{\,i}\right)}{\partial t} + \frac{\partial\left(\rho_\alpha r_\alpha u_\alpha^{\,j} u_\alpha^{\,i}\right)}{\partial x^{j}} = -r_\alpha \frac{\partial p}{\partial x^{i}} + \frac{\partial\left(r_\alpha \tau_\alpha^{ji}\right)}{\partial x^{j}} + r_\alpha \rho_\alpha g^{i} + M_\alpha^{i} \tag{5–176}$$

where M_α^{i} represents momentum transfer with other phases.

5.9.2. Bulk Equations

A bulk continuity equation is derived by summing Equation 5–175 (p. 206) over all phases:

$$\frac{\partial \rho_m}{\partial t} + \frac{\partial \left(\rho_m u_m^i\right)}{\partial x^i} = 0 \qquad \textbf{(5–177)}$$

and a bulk momentum equation by summing *Equation 5–176* (p. 206) over all phases:

$$\frac{\partial \left(\rho_m u_m^i\right)}{\partial t} + \frac{\partial \left(\rho_m u_m^j u_m^i\right)}{\partial x^j} = -\frac{\partial p}{\partial x^i} + \frac{\partial \left(\tau_m^{ji} + \tau_D^{ji}\right)}{\partial x^j} + \rho_m g^i \qquad \textbf{(5–178)}$$

where:

$$\begin{aligned}
\rho_m &= \sum_\alpha r_\alpha \rho_\alpha \\
\rho_m u_m^i &= \sum_\alpha r_\alpha \rho_\alpha u_\alpha^i \\
\tau_m &= \sum_\alpha r_\alpha \tau_\alpha \\
\tau_D^{ji} &= -\sum_\alpha r_\alpha \rho_\alpha \left(u_\alpha^i - u_m^i\right) u_\alpha^j
\end{aligned} \qquad \textbf{(5–179)}$$

5.9.3. Drift and Slip Relations

You now define the slip velocity to be the phasic velocity relative to the continuous phase:

$$u_{S\alpha}^i = u_\alpha^i - u_c^i \qquad \textbf{(5–180)}$$

and the drift velocity as:

$$u_{D\alpha}^i = u_\alpha^i - u_m^i \qquad \textbf{(5–181)}$$

The slip and drift velocities are related by:

$$u_{D\alpha}^i = u_{S\alpha}^i - \sum_\alpha Y_\alpha u_{S\alpha}^i \qquad \textbf{(5–182)}$$

With these relationships, the phasic continuity equation may be written in terms of mass fraction and drift velocity as:

$$\frac{\partial \left(\rho_m Y_\alpha\right)}{\partial t} + \frac{\partial}{\partial x^i}\left(\rho_m Y_\alpha \left(u_m^i + u_{D\alpha}^i\right)\right) = 0 \tag{5–183}$$

5.9.4. Derivation of the Algebraic Slip Equation

The phasic and bulk momentum equations are first transformed to nonconservative form by combining with the phasic and bulk continuity equations. The phasic momentum equation then becomes:

$$\rho_\alpha r_\alpha \frac{\partial u_\alpha^i}{\partial t} + \rho_\alpha r_\alpha u_\alpha^j \frac{\partial u_\alpha^i}{\partial x^j} = -r_\alpha \frac{\partial p}{\partial x^i} + \frac{\partial \left(r_\alpha \tau_\alpha^{ji}\right)}{\partial x^j} + r_\alpha \rho_\alpha g^i + M_\alpha^i \tag{5–184}$$

and the bulk momentum equation becomes:

$$\rho_m \frac{\partial u_m^i}{\partial t} + \rho_m u_m^j \frac{\partial u_m^i}{\partial x^j} = -\frac{\partial p}{\partial x^i} + \frac{\partial \left(\tau_m^{ji} + \tau_D^{ji}\right)}{\partial x^j} + \rho_m g^i \tag{5–185}$$

Equation 5–184 (p. 208) and Equation 5–185 (p. 208) are combined to eliminate the pressure gradient term, yielding:

$$\begin{aligned} M_\alpha^i = {} & r_\alpha \left(\rho_\alpha \frac{\partial u_{D\alpha}^i}{\partial_t} + \left(\rho_\alpha - \rho_m\right)\frac{\partial u_m^i}{\partial t}\right) \\ & + r_\alpha \left(\rho_\alpha u_\alpha^j \frac{\partial u_\alpha^i}{\partial x^j} - \rho_m u_m^j \frac{\partial u_m^i}{\partial x^j}\right) \\ & - \frac{\partial \left(r_\alpha \tau_\alpha^{ji}\right)}{\partial x^j} + r_\alpha \frac{\partial \left(\tau_m^{ji} + \tau_D^{ji}\right)}{\partial x^j} \\ & - r_\alpha \left(\rho_\alpha - \rho_m\right) g^i \end{aligned} \tag{5–186}$$

Several assumptions are now made:

1. The dispersed phase is assumed to instantaneously reach its terminal velocity, so the transient term on the drift velocity is neglected.
2. The approximation is made that:

$$u_\alpha^j \frac{\partial u_\alpha^i}{\partial x^j} \approx u_m^j \frac{\partial u_m^i}{\partial x^j} \tag{5–187}$$

3. The viscous stresses and apparent diffusion stresses are neglected.

With these approximations, Equation 5–186 (p. 208) simplifies to:

$$M_{\alpha}^{i} = r_{\alpha}\left(\rho_{\alpha} - \rho_{m}\right)\left(\frac{\partial u_{m}^{i}}{\partial t} + u_{m}^{j}\frac{\partial u_{m}^{i}}{\partial x^{j}} - g^{i}\right) \tag{5–188}$$

In addition, it is assumed that the interphase momentum transfer is due only to drag and that the particles are spherical:

$$M_{\alpha}^{i} = -\frac{3}{4}\frac{r_{\alpha}\rho_{c}}{d_{p}} C_{D} \left|u_{S\alpha}\right| u_{S\alpha}^{i} \tag{5–189}$$

which leads to the following closed relationship for the slip velocity:

$$\left|u_{S\alpha}\right| u_{S\alpha}^{i} = -\frac{4}{3}\frac{d_{p}}{\rho_{c} C_{D}}\left(\rho_{\alpha} - \rho_{m}\right)\left(\frac{\partial u_{m}^{i}}{\partial t} + u_{m}^{j}\frac{\partial u_{m}^{i}}{\partial x^{j}} - g^{i}\right) \tag{5–190}$$

Note that, for rotating reference frames, the apparent accelerations are automatically included by taking the derivative of the absolute frame velocity rather than relative frame velocity on the right-hand-side.

The effect of τ_{D}^{ji} in the bulk momentum equation is neglected in the current implementation.

5.9.5. Turbulence Effects

In `Eulerian-Eulerian` multiphase, the averaging process used ensures that turbulence effects do not affect the continuity equation; instead, they appear as an apparent turbulent dispersion force in the momentum equation.

In the ASM formulation outlined here, turbulent dispersion forces are not considered in the derivation of slip velocity. Instead, turbulent dispersion is modeled using the same turbulence model as for multicomponent flows:

$$\rho_{\alpha}\overline{Y_{\alpha}'' u_{\alpha}^{i''}} = \frac{\mu_{t}}{Pr_{t}}\frac{\partial Y_{\alpha}}{\partial x^{i}} \tag{5–191}$$

5.9.6. Energy Equation

For multicomponent fluids, the energy equation has an additional term corresponding to enthalpy transport by species velocities. For standard transported components, this velocity is modeled using Fick’s law. For ASM components, this term uses the drift velocity.

The following term is therefore added to the right-hand side of *Equation 2–5* (p. 91):

$$-\frac{\partial}{\partial x^j}\left[\sum_{\alpha}^{N_c}\rho_m Y_\alpha u_{D\alpha}{}^j h_\alpha\right] \quad \textbf{(5–192)}$$

5.9.7. Wall Deposition

If desired, the dispersed phase which accumulates on a boundary may be removed from the calculation. A sink for the ASM mass fraction is defined accordingly. The implicit assumption is that this mass is replaced by an equal mass of the ballast (constraint) component. The energy equation also removes the enthalpy of the ASM species and adds the enthalpy of the ballast species.

5.10. Turbulence Modeling in Multiphase Flow

This section describes the extension of the single-phase turbulence models to multiphase simulations. Documentation that describes the theory of single-phase turbulence models should be read before continuing with this section. For details, see *Turbulence Models* (p. 89).

5.10.1. Phase-Dependent Turbulence Models

Phase dependent turbulence models can be used in conjunction with the inhomogeneous model (particle and mixture models) only.

5.10.1.1. The Eddy Viscosity Hypothesis

The eddy viscosity hypothesis is assumed to hold for each turbulent phase. Diffusion of momentum in phase α is governed by an effective viscosity:

$$\mu_{\alpha eff}=\mu_\alpha+\mu_{t\alpha} \quad \textbf{(5–193)}$$

5.10.1.2. Algebraic Models

5.10.1.2.1. Zero Equation Model

The default zero-equation model uses a formula based on geometric length scale and the mean solution velocity. It is correlated for single-phase turbulent pipe flow. The turbulence viscosity is modeled as the product of a turbulent velocity scale, $U_{t\alpha}$, and a turbulence length scale, $l_{t\alpha}$, as proposed by Prandtl and Kolmogorov:

$$\mu_{t\alpha}=\rho_\alpha f_\mu U_{t\alpha} l_{t\alpha} \quad \textbf{(5–194)}$$

where f_μ is a proportionality constant. The velocity scale is calculated to be the maximum velocity in phase α. If you specify a value for the velocity scale, it will be used for all phases. The length scale is derived using the formula:

$$l_{t\alpha} = \frac{V_D^{1/3}}{7} \tag{5–195}$$

where V_D is the fluid domain volume.

5.10.1.2.2. Dispersed Phase Zero Equation Model

$$v_{td} = \frac{v_{tc}}{\sigma} \Rightarrow \mu_{td} = \frac{\rho_d}{\rho_c} \frac{\mu_{tc}}{\sigma} \tag{5–196}$$

The parameter σ is a turbulent Prandtl number relating the dispersed phase kinematic eddy viscosity v_{td} to the continuous phase kinematic eddy viscosity v_{tc}.

In situations where the particle relaxation time is short compared to turbulence dissipation time scales, you may safely use the default value $\sigma = 1$. If the particle relaxation time is long compared to turbulence dissipation time scales, it may be better to use a value of $\sigma > 1$. This is highly model dependent. Several models are available in the literature.

5.10.1.3. Two-Equation Models

For the k-ε model, the turbulent viscosity is modeled as:

$$\mu_{t\alpha} = c_\mu \rho_\alpha \left(\frac{k_\alpha^2}{\varepsilon_\alpha} \right) \tag{5–197}$$

The transport equations for k and ε in a turbulent phase are assumed to take a similar form to the single-phase transport equations:

$$\begin{aligned} &\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha k_\alpha\right) + \nabla \cdot \left(r_\alpha \left(\rho_\alpha \boldsymbol{U}_\alpha k_\alpha - \left(\mu + \frac{\mu_{t\alpha}}{\sigma_k} \right) \nabla k_\alpha \right) \right) \\ &\qquad = r_\alpha \left(P_\alpha - \rho_\alpha \varepsilon_\alpha \right) + T_{\alpha\beta}^{(k)} \end{aligned} \tag{5–198}$$

$$\begin{aligned} &\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha \varepsilon_\alpha\right) + \nabla \bullet \left(r_\alpha \rho_\alpha U_\alpha \varepsilon_\alpha - \left(\mu + \frac{\mu_{t\alpha}}{\sigma_\varepsilon} \right) \nabla \varepsilon_\alpha \right) \\ &= r_\alpha \frac{\varepsilon_\alpha}{k_\alpha} \left(C_{\varepsilon 1} P_\alpha - C_{\varepsilon 2} \rho_\alpha \varepsilon_\alpha \right) + T_{\alpha\beta}^{\varepsilon} \end{aligned} \tag{5–199}$$

Definitions of the terms are available. For details, see *The k-epsilon Model in ANSYS CFX* (p. 95).

The additional terms $T_{\alpha\beta}^{(k)}$ and $T_{\alpha\beta}^{(\varepsilon)}$ represent interphase transfer for k and ε respectively. These are omitted in CFX, though they may be added as user sources.

Other two equation turbulence models are treated in a similar way.

5.10.1.4. Reynolds Stress Models

The multiphase versions of Reynolds stress models are equivalent to the single phase version, with all flux and volumetric source terms multiplied by volume fractions. Single phase version information is available. For details, see *Reynolds Stress Turbulence Models* (p. 108).

By default, no additional exchange terms are added, though they may be added as user sources.

5.10.2. Turbulence Enhancement

Sato [22] successfully modeled **Particle Induced Turbulence** this for bubbly flow using an enhanced continuous phase eddy viscosity:

$$\mu_{tc} = \mu_{ts} + \mu_{tp} \tag{5–200}$$

where μ_{ts} is the usual **Shear Induced Eddy Viscosity**, and μ_{tp} is an additional **Particle Induced Eddy Viscosity**:

$$\mu_{tp} = C_{\mu p}\,\rho_c\, r_d\, d_p\, \left|\boldsymbol{U}_d - \boldsymbol{U}_c\right| \tag{5–201}$$

The variable $C_{\mu p}$ has a value of 0.6.

5.10.3. Homogeneous Turbulence for Multiphase Flow

In homogeneous multiphase flow, bulk turbulence equations are solved which are the same as the single phase equations, except that the mixture density and the mixture viscosity are used. Single phase turbulence model information is available in *Turbulence Models* (p. 89).

For inhomogeneous multiphase flow, it is possible to solve a single turbulence field in a similar way to homogeneous flow. For details, see Homogeneous Turbulence in Inhomogeneous Flow in the *CFX-Solver Modeling Guide*.

5.11. Additional Variables in Multiphase Flow

When Φ exists in phase α, the corresponding field variable is denoted Φ_α. If it obeys a transport equation, it is given by:

$$\frac{\partial}{\partial t}\left(r_\alpha \rho_\alpha \varphi_\alpha\right) + \nabla \cdot \left(r_\alpha \rho_\alpha \boldsymbol{U}_\alpha \varphi_\alpha\right) - \nabla \cdot \left(r_\alpha \left(\rho_\alpha D_\alpha^{(\varphi)} + \frac{\mu_{t\alpha}}{Sc_{t\alpha}}\right) \nabla \varphi_\alpha\right) = S_\alpha^{(\Phi)} + T_\alpha^{(\Phi)} \tag{5–202}$$

Note:

- φ_α is the conserved quantity per unit mass of phase α.

- Φ_α is the conserved quantity per unit volume of phase α, where $\Phi_\alpha = \rho_\alpha \varphi_\alpha$
- $D_\alpha^{(\Phi)}$ is the Kinematic Diffusivity for the scalar in phase α. This may be set for each Additional Variable and each phase separately.
- $S_\alpha^{(\Phi)}$ is the external volumetric source term in phase α, with units of conserved quantity per unit volume per unit time.
- $T_\alpha^{(\Phi)}$ represents the total source to φ_α due to interphase transfer across interfaces with other phases.
- Although the turbulent Schmidt number, $Sc_{t\alpha}$, is shown to apply to phase α only, it currently cannot be specified on a phase-specific basis in CFX.

As in single phase, Diffusive Transport and Poisson equations are also available for multiphase Additional Variables). For details, see *Additional Variables* (p. 51). Diffusive transport equations exclude the advection term, turbulent diffusion term, and contributions due to interphase mass transfer from *Equation 5–202* (p. 212). Poisson equations exclude these terms as well as the transient term.

5.11.1. Additional Variable Interphase Transfer Models

It is possible for an Additional Variable Φ_α to be coupled to a different Additional Variable Ψ_β across a phase interface between fluids α and β.

The total source to Φ_α per unit volume due to interaction with other phases is given by:

$$T_\alpha^{(\Phi)} = \sum_{\beta \neq \alpha} T_{\alpha\beta}^{(\Phi)} \tag{5–203}$$

where:

$$T_{\alpha\beta}^{(\Phi)} = -T_{\beta\alpha}^{(\Phi)} \Rightarrow \sum_{\alpha} T_\alpha^{(\Phi)} = 0 \tag{5–204}$$

The simplest models for interphase transfer between Φ_α and Ψ_β take the driving force to be proportional to the difference in bulk Additional Variable values across the phase interface:

$$T_{\alpha\beta}^{(\Phi)} = c_{\alpha\beta}^{(\varphi)} \left(\psi_\beta - \varphi_\alpha\right) \tag{5–205}$$

$$T_{\alpha\beta}^{(\Phi)} = c_{\alpha\beta}^{(\Phi)} \left(\Psi_\beta - \Phi_\alpha\right) \tag{5–206}$$

The first of these is used if the Additional Variable is defined per unit mass. The latter is used if the Additional Variable is defined per unit volume.

The coefficients $c_{\alpha\beta}^{(\Phi)}$ are defined by analogy with heat transfer. For details, see *Inhomogeneous Interphase Heat Transfer Models* (p. 194).

Transfer of an Additional Variable across a phase boundary is described by an **additional variable transfer coefficient** $\tau_{\alpha\beta}$. It is the amount of Φ_α crossing a unit area per unit time per unit difference in Φ_α across the phase boundary. Thus:

$$T_{\alpha\beta}^{(\Phi)} = \tau_{\alpha\beta} A_{\alpha\beta} \left(\psi_\beta - \varphi_\alpha \right) , \quad \text{for specific variables} \tag{5–207}$$

$$T_{\alpha\beta}^{(\Phi)} = \tau_{\alpha\beta} A_{\alpha\beta} \left(\Psi_\beta - \Phi_\alpha \right) , \quad \text{for volumetric variables} \tag{5–208}$$

So, you have:

$$c_{\alpha\beta}^{(\Phi)} = \tau_{\alpha\beta} A_{\alpha\beta} \tag{5–209}$$

It is often convenient to express the Additional Variable transfer coefficient in terms of a dimensionless Sherwood number Sh, analogous to the Nusselt number in heat transfer.

$$\tau = \frac{\Gamma\ \mathrm{Sh}}{d} \tag{5–210}$$

The diffusivity scale Γ is the kinematic diffusivity D for a volumetric variable, and the dynamic diffusivity $\rho\, D$ for a specific variable.

5.11.1.1. Particle Model Correlations

In the particle model, the diffusivity scale Γ is that of the continuous phase, and the length scale d is the mean diameter of the dispersed phase:

$$\tau_{\alpha\beta} = \frac{\Gamma_\alpha \mathrm{Sh}_{\alpha\beta}}{d_\beta} \tag{5–211}$$

For laminar forced convection around a spherical particle, theoretical analysis shows that $\mathrm{Sh} = 2$. For a particle in a moving incompressible Newtonian fluid, the Sherwood number is a function of the particle Reynolds number Re and the Additional Variable Prandtl number $\mathrm{Pr} = \mu_\alpha / (\rho\, D)_\alpha$.

Details on the models available in CFX for Additional Variable transfer are available. For details, see Particle Model Correlations in the *CFX-Solver Modeling Guide*. Some additional details for the Interface Flux model are provided below.

- **Interface Flux**

  The user specifies directly the interfacial flux F_{12} from Additional Variable Φ_1 in fluid 1 to Additional Variable Ψ_2 in fluid 2 of a specified fluid pair. This is the rate of Additional Variable transfer per unit time per unit interfacial area from phase 1 to phase 2. Hence, the amount of Additional Variable transferred to fluid 2 from fluid 1 per unit volume is given by:

$$T_{21} = -T_{12} = A_{12} F_{12} \quad \textbf{(5–212)}$$

F_{12} may be given as a constant or an expression.

Typically, F_{12} will be a function of the fluid 1 and fluid 2 Additional Variable fields, and possibly other variables. In this case, the user may accelerate convergence of the coupled solver by also specifying optional fluid 1 and fluid 2 Additional Variable flux coefficients.

$$\begin{aligned} c_1 &\approx \frac{\partial F_{12}}{\partial \Phi_1} \geq 0\,, \quad c_2 \approx \frac{\partial F_{21}}{\partial \Psi_2} \geq 0\,, \quad \text{if volumetric} \\ c_1 &\approx \frac{\partial F_{12}}{\partial \varphi_1} \geq 0\,, \quad c_2 \approx \frac{\partial F_{21}}{\partial \psi_2} \geq 0\,, \quad \text{if specific} \end{aligned} \quad \textbf{(5–213)}$$

The solver takes the absolute value of these flux coefficients to ensure that they are positive. This is required for numerical stability. The partial derivatives need not be computed exactly; it is sufficient for the specified coefficients to simply approximate the partial derivatives. Specification of Additional Variable flux coefficients affects only the convergence rate to the solution of the coupled transfer equations; it does not affect the accuracy of the converged solution.

For example, the simple model using a transfer coefficient multiplied by bulk specific Additional Variable differences may be recovered using:

$$F_{12} = -F_{21} = c\left(\varphi_1 - \psi_2\right)\,, \quad c_1 = c_2 = c \quad \textbf{(5–214)}$$

5.11.1.2. Mixture Model Correlations

If you are using the mixture model, the Sherwood number is defined in terms of a mixture diffusivity scale and the mixture length scale:

$$\tau_{\alpha\beta} = \frac{\Gamma_{\alpha\beta}\,\mathrm{Sh}_{\alpha\beta}}{d_{\alpha\beta}}\,, \quad \Gamma_{\alpha\beta} = r_\alpha \Gamma_\alpha + r_\beta \Gamma_\beta \quad \textbf{(5–215)}$$

5.11.2. Homogeneous Additional Variables in Multiphase Flow

Homogeneous Additional Variables are assumed to have the same values for all phases, that is,

$$\Phi_\alpha = \Phi \;\; 1 \leq \alpha \leq N_p \quad \textbf{(5–216)}$$

and are described by the following bulk transport equation:

$$\frac{\partial}{\partial t}(\rho\,\varphi) + \nabla\cdot(\rho\,\boldsymbol{U}\,\varphi - \Gamma\,\nabla\varphi) = S \tag{5–217}$$

where:

$$\rho = \sum_{\alpha=1}^{N_P} r_\alpha\,\rho_\alpha,\ \boldsymbol{U} = \frac{1}{\rho}\sum_{\alpha=1}^{N_P} r_\alpha\,\rho_\alpha\,\boldsymbol{U}_\alpha,\ \Gamma = \sum_{\alpha=1}^{N_P} r_\alpha\,\Gamma_\alpha \tag{5–218}$$

Homogeneous diffusive transport equations exclude the advection term from *Equation 5–217* (p. 216), while homogeneous Poisson equations exclude the advection and transient terms.

Homogeneous Additional Variables are available for both homogeneous and inhomogeneous flow. However, in most cases, the homogeneous assumption is inappropriate for volumetric Additional Variables. For this reason, only specific Additional Variables are permitted to be homogeneous in ANSYS CFX.

5.12. Sources in Multiphase Flow

The following topics will be discussed:

5.12.1. Fluid-specific Sources

Fluid-specific sources are very similar to those in single phase, except that they are implemented on a phase basis. For details, see *Sources* (p. 54). You may need to multiply a multiphase source by the volume fraction. For details, see Sources in Multiphase Flow in the *CFX-Solver Modeling Guide*.

5.12.2. Bulk Sources

Sources in multiphase flow often have the property that they scale with volume fraction, that is, as the volume fraction source goes to 0, the source also goes to 0. Bulk sources satisfy this property. They are applied at a fluid-independent level, and are therefore automatically added to the equations of all fluids for which they are relevant.

5.13. Interphase Mass Transfer

Interphase mass transfer occurs when mass is carried from one phase into another. It is applicable to both the inhomogeneous and homogeneous multiphase models. For details, see Interphase Mass Transfer in the *CFX-Solver Modeling Guide*.

Mass transfer is represented by sources in the phasic continuity equations:

$$\frac{\partial}{\partial t}\left(r_\alpha\,\rho_\alpha\right) + \nabla\cdot\left(r_\alpha\,\rho_\alpha\,\boldsymbol{U}_\alpha\right) = \boldsymbol{S}_\alpha + \Gamma_\alpha \tag{5–219}$$

- $\boldsymbol{S}_\alpha$ describes user specified mass sources. For details, see *Sources* (p. 54).

- Γ_α is the mass source per unit volume into phase α due to interphase mass transfer. This is expressed as follows:

$$\Gamma_\alpha = \sum_{\beta = 1}^{N_p} \Gamma_{\alpha\beta} \tag{5–220}$$

- $\Gamma_{\alpha\beta}$ is the mass flow rate per unit volume from phase β to phase α. You must have:

$$\Gamma_{\alpha\beta} = -\Gamma_{\beta\alpha} \Rightarrow \sum_{\alpha = 1}^{N_p} \Gamma_\alpha = 0 \tag{5–221}$$

As it is important to keep track of the direction of mass transfer processes, it is convenient to express $\Gamma_{\alpha\beta}$ as follows:

$$\Gamma_{\alpha\beta} = \Gamma^+_{\alpha\beta} - \Gamma^+_{\beta\alpha} \tag{5–222}$$

- The term: $\Gamma^+_{\alpha\beta} > 0$ represents the *positive* mass flow rate per unit volume from phase β to phase α.

For mass transfer processes across a phase interphase, it is useful to express the volumetric mass sources in terms of mass fluxes:

$$\Gamma_{\alpha\beta} = \dot{m}_{\alpha\beta} A_{\alpha\beta} \tag{5–223}$$

- $\dot{m}_{\alpha\beta}$ is the mass flow rate per unit interfacial area from phase β to phase α, and $A_{\alpha\beta}$ is the interfacial area density between the phases.

As interfacial area is commonly proportional to volume fraction, this permits automatic linearization of mass transfer terms relative to volume fraction.

By default, $A_{\alpha\beta}$ is the clipped area density, which does not go to zero as volume fraction goes to zero. Using a clipped area density is sometimes useful because it serves as a crude nucleation model for some kinds of mass transfer. The droplet condensation model has an explicit nucleation model and therefore does not use the clipped area density. In some other cases, using the clipped area density can prevent obtaining a mass-conserving solution. If so, using the unclipped area density may give better results. This is possible by setting the expert parameter `ipmt area density clip = f`.

5.13.1. Secondary Fluxes

The mass source terms affecting the continuity equations, $\Gamma_{\alpha\beta}$ are referred to as **primary mass sources**. Clearly, if mass transfer occurs between phases, then this also induces additional sources and sinks between all other transport equations. These are referred to as **secondary sources**, or **secondary fluxes**, due to mass transfer.

The *default* form of the secondary source terms for a transported variable φ_α is:

$$S_{M\alpha} = \sum_{\beta=1}^{N_p} \left(\Gamma^{+}_{\alpha\beta}\, \varphi_\beta - \Gamma^{+}_{\beta\alpha}\, \varphi_\alpha \right) \quad \textbf{(5–224)}$$

That is, mass transfer from a phase β into a phase α carries the bulk conserved quantity φ_β into phase α. This is referred to as an **upwind formulation**, as the upwinded value is carried out of the outgoing phase, and into the incoming phase.

This default formulation is modified in certain circumstances, in order to take account of discontinuities in the transported variable at the phase interface, for example, enthalpy in the case of phase change. For details, see *The Thermal Phase Change Model* (p. 222).

5.13.2. User Defined Interphase Mass Transfer

For advanced applications, it is possible to specify directly the interphase mass transfer sources.

If the **interphase mass flux** $\dot{m}_{\alpha\beta}$ between any pair of phases is specified, the volumetric mass source is computed internally by multiplying by interfacial area density:

$$\Gamma_{\alpha\beta} = \dot{m}_{\alpha\beta}\, A_{\alpha\beta} \quad \textbf{(5–225)}$$

- Alternatively, you may directly specify the volumetric mass source, or **interphase mass flow**, $\Gamma_{\alpha\beta}$.

In both cases, all transport equations are automatically assigned default secondary sources of the upwinded form:

$$S_{M\alpha\beta} = \left(\Gamma^{+}_{\alpha\beta}\, \varphi_\beta - \Gamma^{+}_{\beta\alpha}\, \varphi_\alpha \right) \quad \textbf{(5–226)}$$

This default form of secondary sources may be overridden in CCL.

5.13.3. General Species Mass Transfer

Consider the interphase mass transfer of a component *A* which is present in two phases α and β. There are several different but related variables for measuring the concentration of component *A* in a mixture that contains it.

$$\begin{aligned} c_{A\alpha} &= \text{Molar Concentration of Component A in phase } \alpha \\ \rho_{A\alpha} &= \text{MassConcentration of Component A in phase } \alpha \\ X_{A\alpha} &= \text{Mole Fraction ofComponent A in phase } \alpha \\ Y_{A\alpha} &= \text{Mass Fraction of Component A in phase } \alpha \end{aligned} \quad \textbf{(5–227)}$$

These are related at follows:

$$\rho_{A\alpha} = M_A\, c_{A\alpha}$$
$$X_{A\alpha} = \frac{c_{A\alpha}}{c_\alpha} \tag{5–228}$$
$$Y_{A\alpha} = \frac{\rho_{A\alpha}}{\rho_\alpha}$$

5.13.3.1. Equilibrium Models

Consider two phases α, β, which contain a component *A*, and consider the situation where the component *A* is in dynamic equilibrium between the phases. Typically, the concentrations of *A* in each phase are *not* the same. However, there is a well defined **equilibrium curve** relating the two concentrations at equilibrium. This is often, though not always, expressed in terms of mole fractions:

$$X_{A\alpha} = f\left(X_{A\beta}\right) \tag{5–229}$$

For binary mixtures, the equilibrium curve depends on temperature and pressure. For multicomponent mixtures, it also depends on mixture composition. The equilibrium curve is in general monotonic and nonlinear. Nevertheless, it is convenient to quasi-linearize the equilibrium relationship as follows:

$$X_{A\alpha} = K^{x}_{A\alpha\beta} X_{A\beta} \quad K^{x}_{A\alpha\beta} = \text{mole fraction equilibrium ratio} \tag{5–230}$$

Other useful forms of the equilibrium relationship are:

$$Y_{A\alpha} = K^{y}_{A\alpha\beta}\, Y_{A\beta} \quad K^{y}_{A\alpha\beta} = \text{mass fraction equilibrium ratio}$$
$$c_{A\alpha} = K^{c}_{A\alpha\beta}\, c_{A\beta} \quad K^{c}_{A\alpha\beta} = \text{molar concentration equilibrium ratio} \tag{5–231}$$
$$\rho_{A\alpha} = K^{\rho}_{A\alpha\beta}\, \rho_{A\beta} \quad K^{\rho}_{A\alpha\beta} = \text{mass concentration equilibrium ratio}$$

The various equilibrium ratios are related as follows:

$$K^{\rho}_{A\alpha\beta} = K^{c}_{A\alpha\beta} = \frac{c_\alpha}{c_\beta} K^{x}_{A\alpha\beta} = \frac{\rho_\alpha}{\rho_\beta} K^{y}_{A\alpha\beta} \tag{5–232}$$

5.13.3.1.1. Raoult's Law

In gas-liquid systems, equilibrium relationships are most conveniently expressed in terms of the partial pressure of the solute in the gas phase.

It is well known that, for a pure liquid *A* in contact with a gas containing its vapor, dynamic equilibrium occurs when the partial pressure of the vapor *A* is equal to its saturated vapor pressure at the same temperature.

Raoult's law generalizes this statement for the case of an ideal liquid mixture in contact with a gas. It states that the partial pressure $P_{A\,g}$ of solute gas *A* is equal to the product of its saturated vapor pressure $P_{A\,\mathrm{sat}}$ at the same temperature, and its mole fraction in the liquid solution X_A.

$$P_{A\,g} = P_{A\,\mathrm{sat}} X_{A\,l} \tag{5–233}$$

If the gas phase is ideal, then Dalton's law of partial pressures gives:

$$P_{A\,g} = X_{A\,g} P_g \tag{5–234}$$

and Raoult's law in terms of a mole fraction equilibrium ratio:

$$P_{A\,g} = P_{A\,\mathrm{sat}} X_{A\,l} \tag{5–235}$$

5.13.3.1.2. Henry's Law

In the case of a gaseous material *A* dissolved in a non-ideal liquid phase, Raoult's law needs to be generalized. Henry's law states that a linear relationship exists between the mole fraction of *A* dissolved in the liquid and the partial pressure of *A* in the gas phase. This is:

$$P_{A\,g} = H^x X_{A\,l} \tag{5–236}$$

H^x is Henry's constant for the component *A* in the liquid l. It has units of pressure, and is known empirically for a wide range of material pairs, especially for common gases dissolved in water. It is strongly dependent on temperature.

Henry's law may also be combined with Dalton's law in order to express it in terms of a mole fraction equilibrium ratio:

$$X_{A\,g} = K^x X_{A\,l}, \quad K^x = \frac{H^x_{A\,l}}{P_g} \tag{5–237}$$

Unfortunately, there is no common convention on the definition of Henry's constant. Another definition in common use relates the partial pressure in the gas to the molar *concentration* in the liquid:

$$P_{A\,g} = H^c c_{A\,l} \tag{5–238}$$

The two definitions of Henry's constant are related by:

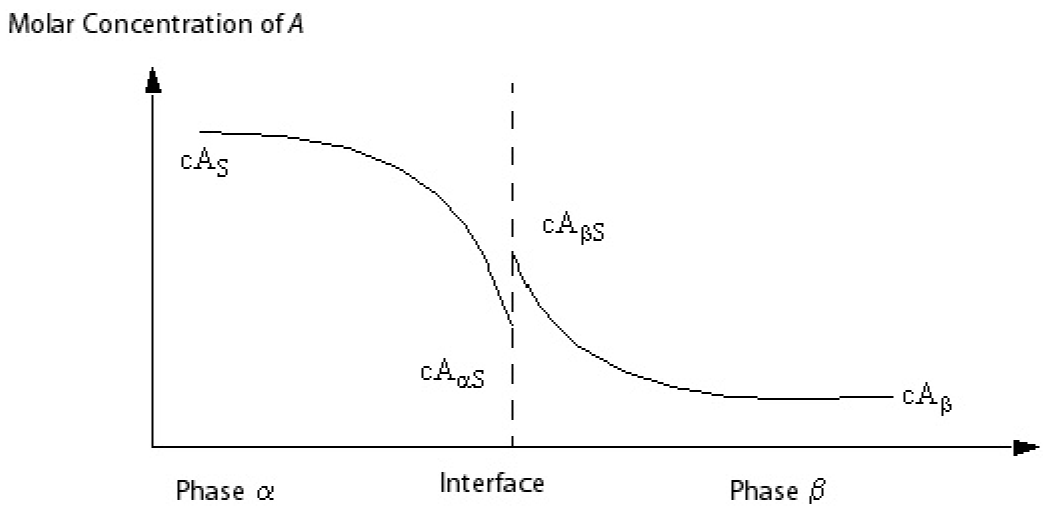
$$H^x X_{A\,l} = H^c c_{A\,l} \quad \textbf{(5–239)}$$

5.13.3.2. Two Resistance Model with Negligible Mass Transfer

Due to the presence of discontinuities in concentration at phase equilibrium, it is, in general, not possible to model multicomponent mass transfer using a single overall mass transfer coefficient. Instead, it is necessary to consider a generalization of the Two Resistance Model previously discussed for heat transfer. For details, see *The Two Resistance Model* (p. 196).

Consider a species *A* dissolved in two immiscible phases α and β. The basic assumption is that there is no resistance to mass transfer at the phase interface, and hence the equilibrium conditions prevail at the phase interface.

Also, it is assumed in this section that the total mass transfer due to species transfer is sufficiently small that primary mass sources to the phasic continuity equations are neglected. Hence, secondary mass fluxes to the species mass transfer equations are also ignored.



You model the component mass transfer using mass transfer coefficients k_α^c, k_β^c, defined on either side of the phase interface. These are usually defined so that driving forces are defined in terms of molar concentration differences. Thus, the molar flux of *A* to phase α from the interface is:

$$n_{A\,\alpha} = k_\alpha^c \left(c_{A\,\alpha\,s} - c_{A\,\alpha} \right) \quad \textbf{(5–240)}$$

and the molar flux of *A* to phase β from the interface is:

$$n_{A\,\beta} = k_\beta^c \left(c_{A\,\beta\,s} - c_{A\,\beta} \right) \quad \textbf{(5–241)}$$

Multiplying through by the molar mass of *A*, this determines the mass fluxes as follows:

Mass flux of *A* to phase α from the interface:

$$m_{A\,\alpha} = k_\alpha^c \left(\rho_{A\,\alpha\,s} - \rho_{A\,\alpha} \right)$$

Mass flux of A to phase β from the interface:

$$m_{A\beta} = k_\beta^c \left(\rho_{A\beta s} - \rho_{A\beta} \right) \tag{5–242}$$

In order to eliminate the unknown interface values, *Equation 5–241* (p. 221) and *Equation 5–242* (p. 222) must be supplemented by the equilibrium relation between $\rho_{\alpha s}$ and $\rho_{\beta s}$. This is most conveniently expressed using the molar concentration equilibrium ratio:

$$c_{A\alpha s} = K^c c_{A\beta s} \Rightarrow \rho_{A\alpha s} = K^c \rho_{A\beta s} \tag{5–243}$$

The mass balance condition:

$$m_{A\alpha} + m_{A\beta} = 0 \tag{5–244}$$

may now be combined with the quasi-linearized equilibrium relationship *Equation 5–243* (p. 222) to determine the interface mass concentrations:

$$\frac{\rho_{A\alpha s}}{K^c} = \rho_{A\beta s} = \frac{k_\alpha^c \rho_{A\alpha} + k_\beta^c \rho_{A\beta}}{K^c k_\alpha^c + k_\beta^c} \tag{5–245}$$

These may be used to eliminate the interface values in *Equation 5–241* (p. 221) and *Equation 5–242* (p. 222) in order to express the interfacial mass fluxes in terms of the phasic mass concentrations:

$$m_{A\alpha} = -m_{A\beta} = k_{\alpha\beta}^c \left(K^c \rho_{A\beta} - \rho_{A\alpha} \right) , \quad \frac{1}{k_{\alpha\beta}^c} = \frac{1}{k_\alpha^c} + \frac{K^c}{k_\beta^c} \tag{5–246}$$

5.13.4. The Thermal Phase Change Model

This model describes phase change induced by interphase heat transfer; it may be used to simulate boiling and condensation, or melting and solidification. For example, it may be used to model condensation of saturated vapor bubbles in sub-cooled liquid, or evaporation of saturated bubbles in superheated liquid.

This section discusses the theory. For modeling information, see Thermal Phase Change Model.

It is essential to consider the heat transfer processes on each side of the phase interface. Hence, the Two Resistance model for interphase heat transfer must be used in conjunction with the Thermal Phase Change model. For details, see *The Two Resistance Model* (p. 196).

Recall that, in this case, the sensible heat flux to phase α from the interface is:

$$q_\alpha = h_\alpha \left(T_s - T_\alpha \right) \tag{5–247}$$

and the sensible heat flux to phase β from the interface is:

$$q_\beta = h_\beta \left(T_s - T_\beta \right) \quad \textbf{(5–248)}$$

- h_α and h_β are the phase α and phase β heat transfer coefficients respectively.
- The interfacial temperature is determined from considerations of **thermodynamic equilibrium**. Ignoring effects of surface tension on pressure, assume $T_s = T_{\text{sat}}$, the **saturation temperature**.

Note that this is in contrast to the case where there is no mass transfer. In that case, the interfacial temperature is determined from the sensible heat balance $q_\alpha + q_\beta = 0$.

In the case of interphase mass transfer, the interphase mass transfer is determined from the total heat balance, as follows.

Total heat flux to phase α from the interface:

$$Q_\alpha = q_\alpha + \dot{m}_{\alpha\beta} H_{\alpha s} \quad \textbf{(5–249)}$$

Total heat flux to phase β from the interface:

$$Q_\beta = q_\beta - \dot{m}_{\alpha\beta} H_{\beta s} \quad \textbf{(5–250)}$$

- $\dot{m}_{\alpha\beta}$ denotes mass flux into phase α from phase β.
- $H_{\alpha s}$ and $H_{\beta s}$ represent interfacial values of enthalpy carried into and out of the phases due to phase change, see below for details.

The total heat balance $Q_\alpha + Q_\beta = 0$ now determines the interphase mass flux:

$$\dot{m}_{\alpha\beta} = \frac{q_{\alpha\beta} + q_{\beta\alpha}}{H_{\beta s} - H_{\alpha s}} \quad \textbf{(5–251)}$$

The secondary heat flux term must be modified in order to take account of the discontinuity in static enthalpy due to latent heat between the two phases. This is achieved using a modification of the upwind formulation *Equation 5–226* (p. 218), due to Prakash [53]. In this formulation, the bulk fluid enthalpy is carried out of the outgoing phase, as in the default upwind formulation. However, the saturation enthalpy is carried into the incoming phase. Thus:

$$\begin{aligned} \dot{m}_{\alpha\beta} > 0 &\Rightarrow H_{\alpha s} = H_{\alpha\,\text{sat}}\,,\ H_{\beta s} = H_\beta \\ \dot{m}_{\alpha\beta} < 0 &\Rightarrow H_{\alpha s} = H_\alpha\,,\ H_{\beta s} = H_{\beta\,\text{sat}} \end{aligned} \quad \textbf{(5–252)}$$

This leads to a formulation which is stable both physically and numerically. It implies that the denominator of *Equation 5–251* (p. 223) is non-zero, being greater than or equal to the latent heat: $L = H_{\beta\,\text{sat}} - H_{\alpha\,\text{sat}}$.

5.13.4.1. Wall Boiling Model

Wall boiling starts when the wall temperature achieves a temperature which is sufficiently large to initiate the activation of wall nucleation sites. This activation temperature is typically a few degrees above the saturation temperature. However, at this stage, the average temperature of the liquid in the vicinity of the heated wall is still well below the saturation temperature, hence in the sub-cooled boiling regime.

Evaporation starts in the microscopic cavities and crevices, which are always present on the solid surface. Liquid becomes supersaturated locally in these nucleation sites, leading to the growth of vapor bubbles at the sites. The bubbles become detached from the sites when they are sufficiently large that external forces (inertial, gravitational, or turbulent) exceed the surface tension forces that keep them attached to the wall. As the bubbles depart from the wall, they are displaced by superheated liquid in the vicinity of the nucleation sites, after which the nucleation site is free to create another bubble. In regions of the wall not affected by bubble growth, the wall heat transfer to the liquid may be described by single phase convective heat transfer.

It is clear that detailed physics of bubble growth is very complex, and occurs on very small length scales in the vicinity of the wall. It is unrealistic to model the detailed physics in a phase-averaged Eulerian multiphase model. It is also unrealistic to resolve the small length scales with ultra-fine meshes. The model described here is a so-called mechanistic model, which aims to model the important physical sub-processes using engineering correlations. The model is a sub-grid scale model, in the sense that the complex physics is assumed to take place in a vicinity very close to the wall which is smaller than the mesh resolution at the wall.

The first and most well-known model of this kind was formulated by Kurul and Podowski (1991) [159], from the Rensselaer Polytechnic Institute. It is known as the RPI model. In this model, a number of the sub-models of the overall mechanistic model were taken from correlations originally developed for exploitation in one-dimensional thermo-hydraulic simulation methods. This model was implemented in CFX-4, with the near-wall distance taking the place of the centerline wall distance in the one-dimensional models. Unfortunately, this lead to results that were strongly mesh-dependent.

In the implementation in CFX-5.7.1, Egorov et al. formulated modifications of the one-dimensional correlations with the aim of restoring mesh independence to the results. This modification of the original RPI model is described here.

The following topics are discussed in this section:

- *Partitioning of the Wall Heat Flux* (p. 224)
- *Sub-models for the Wall Boiling Model* (p. 226)

For more information about the wall boiling model and its usage, see Wall Boiling Model in the *CFX-Solver Modeling Guide*.

5.13.4.1.1. Partitioning of the Wall Heat Flux

A fundamental feature of the mechanistic model of wall nucleation is the algorithm for deciding how the wall heat flux is to be partitioned amongst the separated physical processes of evaporation and sensible heating of the liquid phase. In regions of the wall not influenced by nucleation sites, it is sufficient to consider the wall heat flux as contributing solely to single-phase liquid convective heat transfer. However, in the vicinity of the nucleation sites, some of the heat contributes to vapor production, and the remainder to super heating of the liquid phase as it displaces the rising bubbles. This latter process is known as *quenching*.

Hence, the mechanistic heat partitioning model has the following general structure:

$$Q_{\mathrm{w}} = Q_{\mathrm{l}} + Q_{\mathrm{e}}$$

where Q_{l} is consumed for heating the sub-cooled liquid, and Q_{e} is the evaporation heat flux. Q_{l} is further assumed to be partitioned into two sub-processes:

$$Q_{\mathrm{l}} = Q_{\mathrm{c}} + Q_{\mathrm{q}}$$

where Q_{c} is the heat flux corresponding to convective heat transfer, and Q_q is the heat flux corresponding to quenching. Hence, in total, there is a three-way partition:

$$Q_{\mathrm{w}} = Q_{\mathrm{c}} + Q_{\mathrm{q}} + Q_{\mathrm{e}}$$

In this model, vapor is assumed to be saturated everywhere, and no part of the wall heat flux is arranged for superheating of the vapor phase.

The heat partitioning model considers the whole wall surface as being separated into two fractions:

- Fraction A_2 is influenced by the vapor bubbles, formed on the wall.
- Fraction A_1 is the rest of the wall surface, with $A_1 = 1 - A_2$.

The wall area fraction A_1 represents the part of the wall surface that does not 'feel' the presence of the vapor phase. The wall heat flux for this part of the surface is modeled in a similar way as for the single-phase flow of pure liquid, by using the turbulent wall function procedure in the case of the turbulent flow. This part of Q_{l}, called the convective heat flux Q_c, can be correlated for turbulent flow as:

$$Q_{\mathrm{c}} = A_1\, h_{\mathrm{c}}\, (T_{\mathrm{w}} - T_{\mathrm{l}})$$

where T_{w} is the temperature of the solid wall, T_{l} is the temperature of the liquid at the wall, and h_{c} is the turbulent heat transfer coefficient, which depends on the velocity field and on the near-wall grid cell size. Kurul and Podowski (1991) [159] modeled h_{c} using a one-dimensional Stanton number correlation. Egorov and Menter (2004) [163] modeled h_{c} using the turbulent wall function.

The wall area fraction A_2 represents the remaining part of the wall surface, which exchanges heat with both phases. The already mentioned heat flux Q_{e} comes from this part of the surface and is consumed for evaporation of the initially sub-cooled liquid:

$$Q_{\mathrm{e}} = \dot{m}\left(h_{\mathrm{g,sat}} - h_{\mathrm{l}}\right) \tag{5–253}$$

where $\dot{m}$ is the evaporation mass transfer rate per unit wall area, and $h_{\mathrm{g,sat}}$ and h_{l} are the specific enthalpies of the saturated vapor and sub-cooled liquid, respectively.

A part of the heat flux to the liquid, coming from the wall area fraction A_2, is transported between the bubble departure and the next bubble formation at the same nucleation site. This additional mechanism of heating the liquid phase is called quenching, and is modeled as:

$$Q_{\mathrm{q}} = A_2\, h_{\mathrm{q}}\, (T_{\mathrm{w}} - T_{\mathrm{l}})$$

The area fraction values A_1 and A_2 play an important role in the heat-partitioning model. They are related to the nucleation site density per unit wall area n and to the influence area of a single bubble forming at the wall nucleation site. The latter value is modeled by introducing the bubble departure diameter value d_{w}, which can generally serve as a length scale of the wall boiling mechanism. The RPI model assumes that the diameter of the bubble influence zone is twice as large as d_{w}, therefore the non-dimensional area fraction of the bubble influence is:

$$A_2 = \min\left(\pi\, d_{\mathrm{w}}^2 \cdot n, 1\right) \qquad \textbf{(5–254)}$$

Here the implicitly assumed factor of 2 for the diameter of the bubble influence can be user-specified.

It should be noted that, for the evaporation rate $\dot{m}$ and corresponding heat flux Q_{e} (*Equation 5–253* (p. 225)), the upper limit of *Equation 5–254* (p. 226) for bubble influence area fraction does not apply. Instead of using a factor of A_2 as defined by *Equation 5–254* (p. 226), the $\dot{m}$ value is directly modeled as being proportional to the nucleation site density n.

5.13.4.1.2. Sub-models for the Wall Boiling Model

As mentioned in the previous section, the two most important parameters governing the heat partitioning model are the nucleation site density n and the bubble departure diameter d_{w}. In the RPI wall-boiling model they are correlated to the wall superheat $\Delta T_{\mathrm{sup}} = T_{\mathrm{wall}} - T_{\mathrm{sat}}$ and to the near-wall liquid subcooling $\Delta T_{\mathrm{sub}} = T_{\mathrm{sat}} - T_{\mathrm{liq}}$, respectively.

The following topics will be discussed:

5.13.4.1.2.1. Wall Nucleation Site Density

The model for wall nucleation site density adopted in the RPI model is that of Lemmert and Chawla (1977) [164]:

$$n\left[m^{-2}\right] = \left(m\left(\Delta T_{\mathrm{sup}}\,[K]\right)\right)^{p} \qquad m = 210, \quad p = 1.805$$

Note that the wall superheat in the above equation cannot be negative. Its negative value means that the wall temperature drops below the saturation temperature, where there is no boiling and the heat partitioning model is not used.

This model was implemented in CFX-4 with parameter $m = 185$, giving an overall correction factor of 0.8. Egorov and Menter (2004) [163], conjecturing that this was a deliberate alteration by the RPI group, related to the corresponding factor in the bubble waiting time model. Egorov and Menter also reformulated the correlation as follows:

$$n = n_{\mathrm{ref}} \left(\Delta T_{\mathrm{sup}} / \Delta T_{\mathrm{ref}} \right)^{p}$$

where $n_{\mathrm{ref}} = 0.8 \times 9.922\mathrm{E}{+}5 \quad \left[\mathrm{m}^{-2} \right]$ and $\Delta T_{\mathrm{ref}} = 10 \quad [\mathrm{K}]$. This formulation avoids fractional powers of physical dimensions, hence is more amenable to the use of different unit systems.

Alternative correlations have been proposed for nucleation site density by Kocamustafaogullari and Ishii (1983) [160].

5.13.4.1.2.2. Bubble Departure Diameter

Kurul and Podowski (1991) [159] adopted the correlation for bubble departure diameter due to Tolubinski and Kostanchuk (1970) [165]:

$$d_{\mathrm{w}} = \min\left(d_{\mathrm{ref}} \cdot \exp\left(-\frac{\Delta T_{\mathrm{sub}}}{\Delta T_{\mathrm{ref}}} \right), d_{\mathrm{max}} \right) \tag{5–255}$$

The parameters of the model are dimensional ($d_{\mathrm{max}} = 1.4 \quad [\mathrm{mm}]$, $d_{\mathrm{ref}} = 0.6 \quad [\mathrm{mm}]$, $\Delta T_{\mathrm{ref}} = 45 \quad [\mathrm{K}]$) and are chosen to fit pressurized water data. Hence the model is clearly not universal. Negative liquid sub-cooling is possible here, where it means onset of the bulk boiling. Limiting the bubble departure diameter applies to this situation and prevents d_{w} from growing too high.

Note that the model is strongly dependent on a liquid temperature scale. In the original experimental data, this was taken as the pipe center-line temperature. CFX-4 used the cell-center value in near wall cells, but this proved to give mesh-dependent results. Egorov and Menter restored mesh independence by using the logarithmic form of the wall function to estimate the liquid temperature, T_{l}, at a fixed value of $y^{+} = 250$. ANSYS CFX uses this method, with a user-modifiable value of y^{+}.

5.13.4.1.2.3. Bubble Detachment Frequency

The computation of the evaporation rate $\dot{m}$ requires an additional model parameter, namely the frequency of the bubble detachment from the nucleation site. The model adopted by Kurul and Podowski is that due to Cole (1960) [166]:

$$f = \sqrt{\frac{4g\left(\rho_{1} - \rho_{\mathrm{g}}\right)}{3C_{\mathrm{D}}\, d_{\mathrm{w}}\, \rho_{1}}}$$

Note that, due to its dependence on gravity, this correlation is taken from pool boiling. It is simply estimated as the bubble rise velocity divided by the bubble departure diameter. The drag coefficient factor C_{D} was taken to be unity by Ceumern-Lindenstjerna (1977) [169].

5.13.4.1.2.4. Bubble Waiting Time

Kurul and Podowski (1991) [159] employed the model of Tolubinski and Kostanchuk (1970) [165]. This fixes the waiting time between departures of consecutive bubbles at 80% of the bubble detachment period:

$$t_{\mathrm{w}} = \frac{0.8}{f} \qquad \textbf{(5–256)}$$

The numerator in this equation is adjustable in ANSYS CFX.

5.13.4.1.2.5. Area Influence Factors

Recall from *Partitioning of the Wall Heat Flux* (p. 224) that Kurul and Podowski assumed a diameter of influence of a nucleating bubble equal to twice the bubble departure diameter d_{w}. Encoding this as a user-modifiable parameter F_2 (default value = 2), the area fraction of the bubble influence is given by:

$$A_2 = \min\left(\frac{\pi F_2^2 d_{\mathrm{w}}^2}{4} \cdot n, 1\right)$$

The area fraction A_1 subjected to single phase liquid convective heat transfer is limited from below by a small value, so its actual form is:

$$A_1 = \max\left(10^{-4}, 1 - A_2\right)$$

5.13.4.1.2.6. Convective Heat Transfer

As discussed in *Partitioning of the Wall Heat Flux* (p. 224), single phase convective heat transfer to the liquid phase is modeled using the turbulent wall function (Egorov and Menter 2004 [163]).

$$Q_c = A_1\, h_c\, (T_{\mathrm{w}} - T_{\mathrm{l}})$$

This replaces the mesh dependent Stanton Number correlation originally employed by Kurul and Podowski (1991) [159].

5.13.4.1.2.7. Quenching Heat Transfer

As discussed in *Partitioning of the Wall Heat Flux* (p. 224), quenching heat transfer to the liquid phase in the area of influence of the vapor phase is modeled using a quenching heat transfer coefficient:

$$Q_{\mathrm{q}} = A_2\, h_{\mathrm{q}}\, (T_{\mathrm{w}} - T_{\mathrm{l}})$$

In order to close the model, the quenching heat transfer coefficient h_q, participating in the quenching heat flux to liquid, must be defined. This value depends on the waiting time between the bubble departure and the next bubble formation.

With this value, the quenching heat transfer coefficient is correlated as:

$$h_q = 2\lambda_l f\sqrt{\frac{t_w}{\pi a_l}}$$

where $a_l = \lambda_l / \left(C_{Pl}\rho_l\right)$ is the liquid temperature conductivity coefficient. (Mikic and Rohsenow 1969 [167], Del Valle and Kenning 1985 [168]).

As for the case of bubble departure diameter, Egorov and Menter (2004) [163] used the logarithmic form of the wall function to estimate the liquid temperature T_l at a fixed y^+ value of 250.

5.13.4.1.2.8. Evaporation Rate

Knowing the bubble departure frequency, as well as the bubble size and the nucleation site density, one can obtain the evaporation rate as a product of the bubble mass, the detachment frequency and the site density:

$$\dot{m} = \frac{\pi d_w^3}{6}\rho_g f n$$

This was the form adopted by Kurul and Podowski (1991) [159].

Egorov and Menter expressed the evaporation rate in terms the non-limited area fraction A_2':

$$A_2' = \frac{\pi F_2^2 d_w^2}{4} \cdot n$$

(unlike A_2, A_2' can exceed 1). In this case, the evaporation rate obtains the form:

$$\dot{m} = A_2' \cdot \frac{1}{6} d_w \rho_g f$$

In the final form of the evaporation rate, the area fraction factor A_2' is limited by:

$$\dot{m} = \min\left(A_2', A_{2,\max}'\right) \cdot \frac{1}{6} d_w \rho_g f, \qquad A_{2,\max}' = 5 \tag{5–257}$$

The last limiting procedure is not used if the bubble departure diameter is below $1\ \mu\mathrm{m}$. In this case the non-limited form *Equation 5–257* (p. 229) is used. The estimated value of 5 for $A_{2,\max}'$ is the upper limit for the nucleation sites area fraction, taking into account the effect of overlapping neighboring sites, operating out of phase. The functional form of this parameter is given in terms of the departure frequency and waiting time:

$$A_{2,\max}' = \frac{1}{1 - f \cdot t_w}$$

which gives 5 with the correlation *Equation 5–256* (p. 228) for t_w.

5.13.4.1.3. Determination of the Wall Heat Flux Partition

The wall heat flux partition discussed in *Partitioning of the Wall Heat Flux* (p. 224), together with the user's choice of submodels described in *Sub-models for the Wall Boiling Model* (p. 226), gives the total heat flux as a complex nonlinear function of the wall superheat $\Delta T_{\mathrm{sup}} = T_w - T_{sat}$

$$Q_w\left(\Delta T_{\mathrm{sup}}\right) = Q_c\left(\Delta T_{\mathrm{sup}}\right) + Q_q\left(\Delta T_{\mathrm{sup}}\right) + Q_e\left(\Delta T_{\mathrm{sup}}\right)$$

In the case of an isothermal wall, the three components of the heat flux are computed explicitly as functions of the given wall superheat. However, in the case of a wall with specified total heat flux, or specified wall heat transfer coefficient and outside temperature, the above equation determines ΔT_{sup} implicitly as a function of the total heat supplied to the wall. In this case, ΔT_{sup} is computed by solving the nonlinear equation $Q_{total} = Q_w\left(\Delta T_{\mathrm{sup}}\right)$ using the binary chop method. The resulting value of ΔT_{sup} is subsequently used to determine the three individual components of the wall heat partition.

Important

If you want to use the **User Specified** option for any of the sub-models described in *Sub-models for the Wall Boiling Model* (p. 226), and if you want any of these to be functions of the wall superheat, then it is important to use the correct name for this variable. As the wall boiling model is described as a phase-pair model, this variable is a phase-pair based variable named Tsuperheat. For example:

ΔT_{sup} = water | steam.Tsuperheat

It is not appropriate to use a phase-specific temperature field at the wall, such as water.Temperature, to determine ΔT_{sup} as this variable does not participate in the binary chop algorithm to determine the wall superheat.

5.13.5. Cavitation Model

The tendency for a flow to cavitate is characterized by the cavitation number, defined as:

$$C_a = \frac{p - p_v}{\frac{1}{2}\rho U^2} \qquad \textbf{(5–258)}$$

where p is a reference pressure for the flow (for example, inlet pressure), p_v is the vapor pressure for the liquid, and the denominator represents the dynamic pressure. Clearly, the tendency for a flow to cavitate increases as the cavitation number is decreased.

Cavitation is treated separately from thermal phase change, as the cavitation process is typically too rapid for the assumption of thermal equilibrium at the interface to be correct. In the simplest cavitation models, mass transfer is driven by purely mechanical effects, namely liquid-vapor pressure differences, rather than thermal effects. Current research is directed towards models which take both effects into account.

In CFX, the Rayleigh Plesset model is implemented in the multiphase framework as an interphase mass transfer model. User defined models can also be implemented.

For cavitating flow, the homogeneous multiphase model is typically used.

5.13.5.1. The Rayleigh Plesset Model

The Rayleigh-Plesset equation provides the basis for the rate equation controlling vapor generation and condensation. The Rayleigh-Plesset equation describing the growth of a gas bubble in a liquid is given by:

$$R_B \frac{d^2 R_B}{dt^2} + \frac{3}{2}\left(\frac{dR_B}{dt}\right)^2 + \frac{2\,\sigma}{\rho_f R_B} = \frac{p_v - p}{\rho_f} \tag{5–259}$$

where R_B represents the bubble radius, p_v is the pressure in the bubble (assumed to be the vapor pressure at the liquid temperature), p is the pressure in the liquid surrounding the bubble, ρ_f is the liquid density, and σ is the surface tension coefficient between the liquid and vapor. Note that this is derived from a mechanical balance, assuming no thermal barriers to bubble growth. Neglecting the second order terms (which is appropriate for low oscillation frequencies) and the surface tension, this equation reduces to:

$$\frac{dR_B}{dt} = \sqrt{\frac{2}{3}\frac{p_v - p}{\rho_f}} \tag{5–260}$$

The rate of change of bubble volume follows as:

$$\frac{dV_B}{dt} = \frac{d}{dt}\left(\frac{4}{3}\,\pi\,R_B^3\right) = 4\,\pi\,R_B^2\sqrt{\frac{2}{3}\frac{p_v - p}{\rho_f}} \tag{5–261}$$

and the rate of change of bubble mass is:

$$\frac{dm_B}{dt} = \rho_g \frac{dV_B}{dt} = 4\,\pi\,R_B^2\,\rho_g\sqrt{\frac{2}{3}\frac{p_v - p}{\rho_f}} \tag{5–262}$$

If there are N_B bubbles per unit volume, the volume fraction r_g may be expressed as:

$$r_g = V_B\,N_B = \frac{4}{3}\,\pi\,R_B^3\,N_B \tag{5–263}$$

and the total interphase mass transfer rate per unit volume is:

$$\dot{m}_{fg} = N_B \frac{dm_B}{dt} = \frac{3\, r_g \rho_g}{R_B} \sqrt{\frac{2}{3} \frac{p_v - p}{\rho_f}} \tag{5–264}$$

This expression has been derived assuming bubble growth (vaporization). It can be generalized to include condensation as follows:

$$\dot{m}_{fg} = F \frac{3\, r_g \rho_g}{R_B} \sqrt{\frac{2}{3} \frac{|p_v - p|}{\rho_f}} \operatorname{sgn}\left(p_v - p\right) \tag{5–265}$$

where F is an empirical factor which may differ for condensation and vaporization, designed to account for the fact that they may occur at different rates (condensation is usually much slower than vaporization). For modeling purposes the bubble radius R_B will be replaced by the nucleation site radius R_{nuc}.

Despite the fact that *Equation 5–265* (p. 232) has been generalized for vaporization and condensation, it requires further modification in the case of vaporization.

Vaporization is initiated at nucleation sites (most commonly non-condensible gases). As the vapor volume fraction increases, the nucleation site density must decrease accordingly, because there is less liquid. For vaporization, r_g in *Equation 5–265* (p. 232) is replaced by $r_{nuc}\left(1 - r_g\right)$ to give:

$$\dot{m}_{fg} = F \frac{3\, r_{nuc}\left(1 - r_g\right) \rho_g}{R_{nuc}} \sqrt{\frac{2}{3} \frac{|p_v - p|}{\rho_f}} \operatorname{sgn}\left(p_v - p\right) \tag{5–266}$$

where r_{nuc} is the volume fraction of the nucleation sites. *Equation 5–265* (p. 232) is maintained in the case of condensation.

To obtain an interphase mass transfer rate, further assumptions regarding the bubble concentration and radius are required. The Rayleigh-Plesset cavitation model implemented in CFX uses the following defaults for the model parameters:

- $R_{nuc} = 1\ \mu\text{m}$
- $r_{nuc} = 5\ \text{E} - 4$
- $F_{vap} = 50$
- $F_{\text{cond}} = 0.01$

For an illustration and a validation of the Rayleigh-Plesset cavitation model see Bakir et al. [148]

5.13.5.2. User Defined Cavitation Models

Additional information on creating a user defined model is available. For details, see User Defined Cavitation Models in the *CFX-Solver Modeling Guide*.

When using a user defined cavitation model, the CFX-Solver will perform generic linearizations for the volume fraction and volume continuity equations to help stability and convergence. The saturation pressure is used by the CFX-Solver in linearizing the cavitation rate against pressure.

5.13.6. The Droplet Condensation Model

The Droplet Condensation Model is useful for situations where a dry (or near-saturation) two-phase flow undergoes rapid pressure reduction leading to nucleation and subsequent droplet condensation. It is also useful to model additional condensation when droplets are already present in significant quantities. Typical applications include low-pressure steam turbines, in which context this model is also referred to as the Nonequilibrium Steam (NES) model. Such flows are typically transonic. The droplet phase can enter through the inlet or appear through various nucleation mechanisms, including homogeneous (volumetric) and heterogeneous (surface) nucleation. Presently, only one homogeneous nucleation model is available for selection based on Classical theory. Heterogeneous and alternate homogeneous nucleation models can be supplied using the available user defined option.

The Droplet Condensation Model differs from the Thermal Phase Change model (see *The Thermal Phase Change Model* (p. 222)) in that the droplet diameter is calculated as part of the model rather than as a user input. This leads to improved accuracy. In order to do this a transport equation for droplet number must also be solved, which has as its source term nucleation contributions. This model differs from the Equilibrium Phase Change model (see Equilibrium Phase Change Model in the *CFX-Solver Modeling Guide*) in that it does not assume the flow to instantaneously reach equilibrium conditions, and therefore implicitly includes losses due to thermodynamic irreversibility.

In the following discussion, we consider a homogeneous multiphase system, in which the droplets move with the same velocity as the continuous phase. However, the model has been generalized to inhomogeneous systems as well.

The system of equations involves one continuous phase and any number of dispersed (condensed) phases. The condensed phases travel at the speed of the continuous phase. Any combination of condensed phases can exist in the solution so that for the continuous phase, mass conservation becomes:

$$\frac{\partial \rho_c r_c}{\partial t} + \frac{\partial}{\partial x_i}\left(\rho_c u_i r_c\right) = -\sum_{i=1}^{nd} (S_d + m^* r_c J_d) \tag{5–267}$$

where the mass sources are summed over the nd condensed phases. The condensed phases can change size by condensation or evaporation.

For a condensed phase, d mass conservation is:

$$\frac{\partial \rho_d r_d}{\partial t} + \frac{\partial}{\partial x_i}\left(\rho_d u_i r_d\right) = S_d + m^* r_c J_d \tag{5–268}$$

where each dispersed phase has a corresponding number equation of the form:

$$\frac{\partial \rho_d N_d}{\partial t} + \frac{\partial}{\partial x_\mathrm{i}}\left(\rho_d u_\mathrm{i} N_d\right) = \rho_d r_\mathrm{c} J_\mathrm{d} \quad \textbf{(5–269)}$$

and J_d is the nucleation model with units defined as the number of droplets generated per unit time per unit volume of vapor and m^* is the nucleated droplet mass based on the critical radius r^*. Note that the droplets are transported with the mixture velocity because no slip is assumed between the phases. The usual constraint applies for the volume fractions where:

$$r_\mathrm{c} + \sum_{\mathrm{i}=1}^{nd} r_d = 1 \quad \textbf{(5–270)}$$

In addition, global continuity and momentum equations are also solved as described in *Homogeneous Hydrodynamic Equations* (p. 172). The continuous phase energy equation, in total enthalpy form, is

$$\frac{\partial \rho_\mathrm{c} r_\mathrm{c} H_\mathrm{c}}{\partial t} + \frac{\partial}{\partial x_\mathrm{i}}\left(\rho_\mathrm{c} r_\mathrm{c} u_\mathrm{i} H_\mathrm{c}\right) = (-r_\mathrm{c}) \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_\mathrm{i}}\left(\Gamma_\mathrm{t} r_\mathrm{c} \frac{\partial T}{\partial x_\mathrm{i}}\right) + S_\mathrm{H} \quad \textbf{(5–271)}$$

where a similar energy transport equation can be applied for the dispersed phases, either in the context of small or large droplets as will be discussed subsequently. The Droplet Condensation model can be used for both small and large droplets. However, small and large droplets use different models for heat transfer and phase change. For large droplets, the heat transfer and phase change models described by the Thermal Phase Change model should be used. For small droplets (less than 1 μm) the Small Droplet heat transfer model is appropriate; it sets the droplet temperature to

$$T_\mathrm{d} = T_\mathrm{s}(P) - T_\mathrm{sc} \frac{R_\mathrm{d}^*}{R_\mathrm{d}} \quad \textbf{(5–272)}$$

where T_s refers to the saturation temperature, T_sc refers to the supercooling level in gas phase, R_d is the droplet radius, and R_d^* is the critical radius at formation of the dispersed phase.

For small droplets, the interphase heat and mass transfer models are also modified to include the influence of the Knudsen (*Kn*) number on the Nusselt number. The Kn dependence is required because droplet sizes vary significantly from the initial nucleated radius (in a non-continuum regime) in the range of angstroms. The droplet growth rate is:

$$\frac{dR_\mathrm{d}}{dt} = \frac{k_\mathrm{c}}{R_\mathrm{d} \rho_\mathrm{d} (1 + c\, Kn)} \left(\frac{T_\mathrm{d} - T_\mathrm{g}}{h_\mathrm{g} - h_\mathrm{p}}\right) \quad \textbf{(5–273)}$$

which is subsequently used to compute the interphase mass transfer rate in conjunction with an interfacial area density to be described later in this section.

The source of droplets into the domain is based on a nucleation model, which for classical nucleation models has the form of:

$$J = A \exp\left(-\frac{\Delta G^*}{k\, T_g} \right) \tag{5–274}$$

where A is a constant determined by the particular nucleation model, G^* is the Gibbs free energy change at the critical radius conditions, k is Boltzmann's constant, and T_g is the supercooled vapor temperature. To compute the Gibbs free energy change a property database must be used that evaluates supercooled state properties. This requires an equation of state for the vapor phase amenable to extrapolation into regions within the saturation zone. The IAPWS and Redlich Kwong equations of state satisfy this requirement. A user defined nucleation model can also be applied, allowing for different homogeneous or heterogeneous nucleation models to be employed.

The interaction between the phases by mass transfer depends on calculating the droplet diameter, which, if a monodispersed distribution is assumed for droplets with a common origin, can be determined from the droplet number, N_d. The relevant equation is then:

$$R_d = \left(\frac{3\, r_d}{4\, \pi\, N_d} \right)^{\frac{1}{3}} \tag{5–275}$$

with an interfacial area density defined as:

$$\beta_d = \frac{3\, r_d}{R_d} \tag{5–276}$$

The interfacial mass transfer term can then be computed with the known droplet growth rate and the interfacial area density:

$$S_d = \rho_d\, \beta_d\, \frac{d R_d}{dt} \tag{5–277}$$

which can be used to obtain the heat transfer term:

$$S_H = -S_d\, H_u + \beta_d\, Q_d \tag{5–278}$$

where H_u is an upwinded total enthalpy, its value either for the continuous or dispersed phase depending on the direction of interphase mass transfer. In addition, Q_d is the heat transfer (per unit area) between the dispersed and continuous phase based on:

$$Q_d = \frac{k_c}{R_d\,(1 + c\, Kn)} \left(T_d - T_g \right) \tag{5–279}$$

Since the droplets in condensing systems are generally quite small (less than 1 $\mu\, m$), it is assumed that the droplet temperature is uniform (a zero resistance model between the droplet surface and its internal temperature). This implies that almost all of the heat transfer either comes from the continuous phase during evaporation or goes into it during condensation.

The Nusselt (Nu) number underlying *Equation 5–273* (p. 234) and *Equation 5–279* (p. 235) is corrected to account for droplet sizes that span a wide Knudsen number (Kn) range (from free-molecular to continuum). The Nusselt number applied is:

$$Nu = \frac{2}{(1 + c\,Kn)} \tag{5–280}$$

where c is an empirical factor set to 3.18.

5.14. Free Surface Flow

This section describes free surface flow theory, which is the most common application of homogeneous multiphase flow.

5.14.1. Implementation

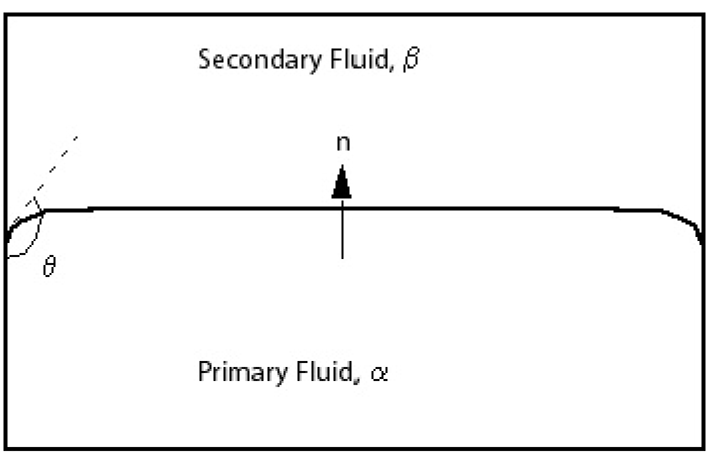
The implementation of free surface flow in CFX is essentially the same as multiphase (homogeneous or inhomogeneous) along with some special discretization options to keep the interface sharp. These include:

- A compressive differencing scheme for the advection of volume fractions in the volume fraction equations.
- A compressive transient scheme for the volume fraction equations (if the problem is transient).
- Special treatment of the pressure gradient and gravity terms to ensure that the flow remain well behaved at the interface.

5.14.2. Surface Tension

The surface tension model used in CFX is based on the Continuum Surface Force model of Brackbill et al. [27]. This models the surface tension force as a volume force concentrated at the interface, rather than a surface force. Consider the free surface interface shown in the figure below:

Figure 5.1 Free Surface Interface



Define a Primary Fluid α (the liquid phase) and a Secondary Fluid β (usually a gas phase). The surface tension force given by the Continuum Surface Force model is:

$$\boldsymbol{F}_{\alpha\beta} = \boldsymbol{f}_{\alpha\beta}\,\delta_{\alpha\beta} \tag{5–281}$$

where:

$$\boldsymbol{f}_{\alpha\beta} = -\sigma_{\alpha\beta}\,\kappa_{\alpha\beta}\,\boldsymbol{n}_{\alpha\beta} + \nabla_s\,\sigma \tag{5–282}$$

$$\delta_{\alpha\beta} = |\nabla r_{\alpha\beta}| \tag{5–283}$$

where σ is the surface tension coefficient, $\boldsymbol{n}_{\alpha\beta}$ is the interface normal vector pointing from the primary fluid to the secondary fluid (calculated from the gradient of a smoothed volume fraction), ∇_S is the gradient operator on the interface and κ is the surface curvature defined by:

$$\kappa_{\alpha\beta} = \nabla \cdot \boldsymbol{n}_{\alpha\beta} \tag{5–284}$$

The two terms summed on the right hand side of *Equation 5–282* (p. 237) reflect the normal and tangential components of the surface tension force respectively. The normal component arises from the interface curvature and the tangential component from variations in the surface tension coefficient (the Marangoni effect).

The $\delta_{\alpha\beta}$ term is often called the interface delta function; it is zero away from the interface, thereby ensuring that the surface tension force is active only near to the interface.

When the interface between the two fluids intersects a wall, it is possible to account for wall adhesion by specifying the contact angle, θ, which the interface makes with the wall through the primary fluid. The interface normal vector used for the calculations of both curvature and the surface tension force must satisfy the wall contact angle.

Chapter 6: Particle Transport Theory

Particle transport modeling is a type of multiphase model, where particulates are tracked through the flow in a Lagrangian way, rather than being modeled as an extra Eulerian phase. The full particulate phase is modeled by just a sample of individual particles. The tracking is carried out by forming a set of ordinary differential equations in time for each particle, consisting of equations for position, velocity, temperature, and masses of species. These equations are then integrated using a simple integration method to calculate the behavior of the particles as they traverse the flow domain.

This chapter describes:

The following section describes the methodology used to track the particles.

6.1. Lagrangian Tracking Implementation

Within the particle transport model, the total flow of the particle phase is modeled by tracking a small number of particles through the continuum fluid. The particles could be solid particles, drops or bubbles.

The application of Lagrangian tracking in CFX involves the integration of particle paths through the discretized domain. Individual particles are tracked from their injection point until they escape the domain or some integration limit criterion is met. Each particle is injected, in turn, to obtain an average of all particle tracks and to generate source terms to the fluid mass, momentum and energy equations. Because each particle is tracked from its injection point to final destination, the tracking procedure is applicable to steady state flow analysis.

The following sections describe the methodology used to track the particles.

6.1.1. Calculation of Particle Variables

The particle displacement is calculated using forward Euler integration of the particle velocity over timestep, δt.

As $\left(d\boldsymbol{x}_p / dt \right) = \boldsymbol{U}_p$, the particle displacement is given as:

$$\boldsymbol{x}_p^n = \boldsymbol{x}_p^o + \boldsymbol{U}_p^o \delta t \tag{6–1}$$

where the superscripts *o* and *n* refer to old and new values respectively, and $\boldsymbol{U}_p^o$ is the initial particle velocity. In forward integration, the particle velocity calculated at the start of the timestep is assumed to prevail over the entire step. At the end of the timestep, the new particle velocity is calculated using the analytical solution to the particle momentum equation:

$$m_p \frac{d\boldsymbol{U}_p}{dt} = \boldsymbol{F}_{all} \tag{6–2}$$

$\boldsymbol{F}_{all}$ is the sum of all forces acting on a particle. This equation is an example of the generic transport equation:

$$\frac{d\phi_p}{dt} = \frac{\left(\phi_f - \phi_p\right)}{\tau} + R \tag{6–3}$$

where ϕ is a generic transported variable, subscript f indicates the value of the variable in the surrounding fluid, τ is a linearization coefficient and R is a general nonlinear source. The analytical solution of the generic transport equation above can be written as:

$$\phi_p = \phi_f + \left(\phi_p^0 - \phi_f\right) exp\left(-\frac{\delta t}{\tau}\right) + \tau R\left(1 - exp\left(-\frac{\delta t}{\tau}\right)\right) \tag{6–4}$$

The fluid properties are taken from the start of the timestep.

In the calculation of forces and values for τ and R, many fluid variables, such as density, viscosity and velocity are needed at the position of the particle. These variables are always obtained accurately by calculating the element in which the particle is traveling, calculating the computational position within the element, and using the underlying shape functions of the discretization algorithm to interpolate from the vertices to the particle position.

6.1.2. Interphase Transfer Through Source Terms

Just as the fluid affects the particle behavior through forces and, for example, convective heat transfer, there is a counteracting influence of the particle on the fluid. This effect is termed as *coupling* between phases. If the fluid is allowed to influence trajectories but particles do not affect the fluid, then the interaction is termed one-way coupling. If the particles also affect the fluid behavior, then the interaction is termed two-way coupling.

The flow prediction of the two phases in one-way coupled systems is relatively straightforward. The fluid flow field may be calculated irrespective of the particle trajectories. One-way coupling may be an acceptable approximation in flows with low dispersed phase loadings where particles have a negligible influence on the fluid flow.

Two-way coupling requires that the particle source terms are included in the fluid momentum equations. The momentum sources could be due to turbulent dispersion forces or drag. The particle source terms are generated for each particle as they are tracked through the flow. Particle sources are applied in the control volume that the particle is in during the timestep.

The particle sources to the fluid momentum equations are obtained by solving transport equations for the sources. The generic equation for particle sources is:

$$\frac{dS_P}{dt} = C_S \phi_P + R_S \quad \textbf{(6–5)}$$

Where $C_S \phi_P$ are the contributions from the particles that are linear in the solution variable and R_S contains all other contributions, including when appropriate a mass transfer term $-\left(dm_p/dt\right)\phi_p$. This equation has the same form as the general particle transport and is solved in the same way as outlined above.

The source, *S*, to be added to the continuous phase is then multiplied by the number flow rate for that particle, which is the mass flow rate divided by the mass of the particle.

In CFX, the particle source terms are recalculated each time particles are injected. The source terms are then retained in memory in order that they may be applied each time the fluid coefficients are calculated. Thus, the particle sources may be applied even though particles have not been injected in the current flow calculation.

6.2. Momentum Transfer

Consider a discrete particle traveling in a continuous fluid medium. The forces acting on the particle that affect the particle acceleration are due to the difference in velocity between the particle and fluid, as well as to the displacement of the fluid by the particle. The equation of motion for such a particle was derived by Basset, Boussinesq and Oseen for a rotating reference frame:

$$m_P \frac{d\boldsymbol{U}_P}{dt} = \boldsymbol{F}_{\mathrm{D}} + \boldsymbol{F}_{\mathrm{B}} + \boldsymbol{F}_{\mathrm{R}} + \boldsymbol{F}_{\mathrm{VM}} + \boldsymbol{F}_P + \boldsymbol{F}_{\mathrm{BA}} \quad \textbf{(6–6)}$$

which has the following forces on the right hand side:

- $\boldsymbol{F}_{\mathrm{D}}$: drag force acting on the particle.
- $\boldsymbol{F}_{\mathrm{B}}$: buoyancy force due to gravity.
- $\boldsymbol{F}_{\mathrm{R}}$: forces due to domain rotation (centripetal and Coriolis forces).
- $\boldsymbol{F}_{\mathrm{VM}}$: virtual (or added) mass force. This is the force to accelerate the virtual mass of the fluid in the volume occupied by the particle. This term is important when the displaced fluid mass exceeds the particle mass, such as in the motion of bubbles.
- $\boldsymbol{F}_P$: pressure gradient force. This is the force applied on the particle due to the pressure gradient in the fluid surrounding the particle caused by fluid acceleration. It is only significant when the fluid density is comparable to or greater than the particle density.
- $\boldsymbol{F}_{\mathrm{BA}}$: Basset force or history term which accounts for the deviation in flow pattern from a steady state. This term is not implemented in CFX.

The left hand side of *Equation 6–6* (p. 241) can be modified due to the special form of the virtual mass term (see *Virtual or Added Mass Force* (p. 244)) which leads to the following form of the particle velocity:

$$\frac{d\boldsymbol{U}_P}{dt} = \frac{1}{m_P + \frac{C_{\mathrm{VM}}}{2} m_{\mathrm{F}}} \left(\boldsymbol{F}_{\mathrm{D}} + \boldsymbol{F}_{\mathrm{B}} + \boldsymbol{F}'_{\mathrm{VM}} + \boldsymbol{F}_P \right) + \frac{1}{m_P} \boldsymbol{F}_{\mathrm{R}} \quad \textbf{(6–7)}$$

Only a part of the virtual mass term, $\boldsymbol{F}'_{\mathrm{VM}}$, remains on the right hand side. The particle and fluid mass values are given by:

$$m_P = \frac{\pi}{6} d_P^3 \rho_P \quad \text{and} \quad m_F = \frac{\pi}{6} d_P^3 \rho_F \quad \textbf{(6–8)}$$

with the particle diameter d_P as well as the fluid and particle densities ρ_F and ρ_P. The ratio of the original particle mass and the effective particle mass (due to the virtual mass term correction) is stored in

$$\begin{aligned} R_{\mathrm{VM}} &= \frac{m_P}{m_P + \frac{C_{\mathrm{VM}}}{2} m_{\mathrm{F}}} = \frac{\rho_P}{\rho_P + \frac{C_{\mathrm{VM}}}{2} \rho_{\mathrm{F}}} \\ 1 - R_{\mathrm{VM}} &= \frac{\frac{C_{\mathrm{VM}}}{2} \rho_{\mathrm{F}}}{\rho_P + \frac{C_{\mathrm{VM}}}{2} \rho_{\mathrm{F}}} \end{aligned} \quad \textbf{(6–9)}$$

Using R_{VM}, *Equation 6–7* (p. 242) can be written as

$$\frac{d\boldsymbol{U}_P}{dt} = \frac{R_{\mathrm{VM}}}{m_P} \left(\boldsymbol{F}_{\mathrm{D}} + \boldsymbol{F}_{\mathrm{B}} + \boldsymbol{F}'_{\mathrm{VM}} + \boldsymbol{F}_P \right) + \frac{1}{m_P} \boldsymbol{F}_{\mathrm{R}} \quad \textbf{(6–10)}$$

Each term on the right hand side of *Equation 6–10* (p. 242) can potentially be linearized with respect to the particle velocity variable $\boldsymbol{U}_P$, leading to the following equation for each term:

$$\boldsymbol{T} = \boldsymbol{R} + C_{\mathrm{lin}} \boldsymbol{U}_P \quad \textbf{(6–11)}$$

The following sections show the contribution of all terms to the right hand side values $\boldsymbol{R}$ and the linearization coefficient C_{lin}.

6.2.1. Drag Force on Particles

The aerodynamic drag force on a particle is proportional to the slip velocity, $\boldsymbol{U}_{\mathrm{s}}$, between the particle and the fluid velocity:

$$\boldsymbol{F}_{\mathrm{D}} = \frac{1}{2}\, C_{\mathrm{D}}\, \rho_{\mathrm{F}}\, A_{\mathrm{F}}\, |\boldsymbol{U}_{\mathrm{S}}|\, \boldsymbol{U}_{\mathrm{S}} = \frac{1}{2}\, C_{\mathrm{D}}\, \rho_{\mathrm{F}}\, A_{\mathrm{F}}\, |\boldsymbol{U}_{\mathrm{F}} - \boldsymbol{U}_{P}|\, (\boldsymbol{U}_{\mathrm{F}} - \boldsymbol{U}_{P}) \tag{6–12}$$

where C_{D} is the drag coefficient and A_{F} is the effective particle cross section. The drag coefficient, C_{D}, is introduced to account for experimental results on the viscous drag of a solid sphere. The coefficient is calculated in the same way as for Eulerian-Eulerian multiphase flow.

Using the new variable $D = \left(C_{\mathrm{D}}\, \rho_{\mathrm{F}}\, A_{\mathrm{F}}\, |\boldsymbol{U}_{\mathrm{S}}| \right) / 2$, this leads to the following contribution to the right hand side and linearization coefficient of *Equation 6–7* (p. 242):

$$\boldsymbol{R} = \frac{D}{m_P + \frac{C_{\mathrm{VM}}}{2}\, m_{\mathrm{F}}}\, \boldsymbol{U}_{\mathrm{F}} = \frac{R_{\mathrm{VM}}}{m_P}\, D\, \boldsymbol{U}_{\mathrm{F}} \tag{6–13}$$

$$C_{\mathrm{lin}} = -\frac{D}{m_P + \frac{C_{\mathrm{VM}}}{2}\, m_{\mathrm{F}}} = -\frac{R_{\mathrm{VM}}}{m_P}\, D \tag{6–14}$$

The particle momentum source due to drag is calculated from the following equation:

$$\frac{d\boldsymbol{S}}{dt} = -\boldsymbol{F}_{\mathrm{D}} = -\frac{1}{2}\, C_{\mathrm{D}}\, \rho_{\mathrm{F}}\, A_{\mathrm{F}}\, |\boldsymbol{U}_{\mathrm{S}}|\, \boldsymbol{U}_{\mathrm{S}} \tag{6–15}$$

The source, $\boldsymbol{S}$, added to the continuous phase is then multiplied with the number rate for that particle.

6.2.2. Buoyancy Force on Particles

The buoyancy force is the force on a particle immersed in a fluid. The buoyant force is equal to the weight of the displaced fluid and is given by

$$\boldsymbol{F}_{\mathrm{B}} = (m_P - m_{\mathrm{F}})\, \boldsymbol{g} = m_P \left(1 - \frac{\rho_{\mathrm{F}}}{\rho_P} \right) \boldsymbol{g} = \frac{\pi}{6}\, d_P^3 \left(\rho_P - \rho_{\mathrm{F}} \right) \boldsymbol{g} \tag{6–16}$$

where $\boldsymbol{g}$ is the gravity vector.

This leads to the following contribution to the right hand side of *Equation 6–7* (p. 242):

$$\boldsymbol{R} = \frac{\rho_P - \rho_{\mathrm{F}}}{\rho_P + \frac{C_{\mathrm{VM}}}{2}\, \rho_{\mathrm{F}}}\, \boldsymbol{g} = R_{\mathrm{VM}} \left(1 - \frac{\rho_{\mathrm{F}}}{\rho_P} \right) \boldsymbol{g} \tag{6–17}$$

The buoyancy force has no contribution to the particle momentum source into the continuous phase.

6.2.3. Rotation Force

In a rotating frame of reference, the rotation term is an intrinsic part of the acceleration and is the sum of Coriolis and centripetal forces:

$$\boldsymbol{F}_{\mathrm{R}} = m_P \left(-2\,\boldsymbol{\Omega}\times\boldsymbol{U}_P - \boldsymbol{\Omega}\times\boldsymbol{\Omega}\times\boldsymbol{r}_P \right) \tag{6–18}$$

As described elsewhere (see *Pressure Gradient Force* (p. 245) and *Virtual or Added Mass Force* (p. 244)), the implemented rotation term also contains contributions from the pressure gradient and the virtual mass force due to the domain rotation which leads to the following final contribution of the rotation term to the right hand side of *Equation 6–7* (p. 242).

$$\begin{aligned}
\boldsymbol{R} = & -2\,\boldsymbol{\Omega}\times\boldsymbol{U}_P - \boldsymbol{\Omega}\times\boldsymbol{\Omega}\times\boldsymbol{r}_P \\
& -\left(1 - R_{\mathrm{VM}}\right)\left(-2\,\boldsymbol{\Omega}\times\boldsymbol{U}_{\mathrm{F}} - \boldsymbol{\Omega}\times\boldsymbol{\Omega}\times\boldsymbol{r}_P \right) \quad \text{(virtual mass force)} \\
& -R_{\mathrm{VM}}\frac{\rho_{\mathrm{F}}}{\rho_P}\left(-2\,\boldsymbol{\Omega}\times\boldsymbol{U}_{\mathrm{F}} - \boldsymbol{\Omega}\times\boldsymbol{\Omega}\times\boldsymbol{r}_P \right) \quad \text{(pressure gradient term)}
\end{aligned} \tag{6–19}$$

Using the substitutions:

$$\begin{aligned}
\boldsymbol{R}_{\mathrm{Cor,P}} &= -2\,\boldsymbol{\Omega}\times\boldsymbol{U}_P \\
\boldsymbol{R}_{\mathrm{Cor,F}} &= -2\,\boldsymbol{\Omega}\times\boldsymbol{U}_{\mathrm{F}} \\
\boldsymbol{R}_{\mathrm{Cent}} &= -\boldsymbol{\Omega}\times\boldsymbol{\Omega}\times\boldsymbol{r}_P
\end{aligned}$$

Equation 6–19 (p. 244) can be written as:

$$\boldsymbol{R} = \boldsymbol{R}_{\mathrm{Cor,P}} + \left(C - 1\right)\,\boldsymbol{R}_{\mathrm{Cor,F}} + C\,\boldsymbol{R}_{\mathrm{Cent}} \tag{6–20}$$

where

- $C = 1$ if neither pressure gradient nor virtual mass force is taken into account
- $C = R_{\mathrm{VM}}$ if only virtual mass force is taken into account
- $C = 1 - \rho_{\mathrm{F}}/\rho_P$ if only pressure gradient force is taken into account
- $C = R_{\mathrm{VM}}\left(1 - \rho_{\mathrm{F}}/\rho_P\right)$ if pressure gradient and virtual mass forces are taken into account

The rotation force has no contribution to the particle momentum source into the continuous phase.

6.2.4. Virtual or Added Mass Force

This force is caused by the fact that the particle has to accelerate some of the surrounding fluid, leading to an additional drag of the following form:

$$\boldsymbol{F}_{\mathrm{VM}} = \frac{C_{\mathrm{VM}}}{2} m_{\mathrm{F}} \left(\frac{d\boldsymbol{U}_{\mathrm{F}}}{dt} - \frac{d\boldsymbol{U}_{P}}{dt} \right) \tag{6–21}$$

If the virtual mass force is included, the coefficient C_{VM} is normally set to 1. However, when the virtual mass force is not included, then C_{VM} effectively has the value zero, and R_{VM} is equal to 1. The second part of the right hand side contains the particle velocity deviation, hence, it can be brought to the left hand side of the particle momentum equation (see *Equation 6–7* (p. 242)), leading to a modified effective mass. Considering only steady state flows, the remaining term can be written as

$$\boldsymbol{F}'_{\mathrm{VM}} = \frac{C_{\mathrm{VM}}}{2} m_{\mathrm{F}} \left(\boldsymbol{U}_{\mathrm{F}} \nabla \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{R}_{\mathrm{F}} \right) \tag{6–22}$$

leading to the following contribution to the right hand side of *Equation 6–7* (p. 242):

$$\boldsymbol{R} = \left(1 - R_{\mathrm{VM}}\right) \left(\boldsymbol{U}_{\mathrm{F}} \nabla \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{R}_{\mathrm{F}} \right) \tag{6–23}$$

with

$$\boldsymbol{R}_{\mathrm{F}} = -2\, \boldsymbol{\Omega} \times \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{\Omega} \times \boldsymbol{\Omega} \times \boldsymbol{r}_{P} \tag{6–24}$$

necessary if the particles are solved in a rotating system. The $\boldsymbol{R}_{\mathrm{F}}$ term is shifted to the rotation term (see *Equation 6–19* (p. 244)), therefore, the implemented virtual mass term reduces to

$$\boldsymbol{R} = \left(1 - R_{\mathrm{VM}}\right) \left(\boldsymbol{U}_{\mathrm{F}} \nabla \boldsymbol{U}_{\mathrm{F}} \right) \tag{6–25}$$

The particle momentum source due to the virtual mass force is computed from:

$$\frac{d\boldsymbol{S}}{dt} = -\boldsymbol{F}_{\mathrm{VM}} \tag{6–26}$$

6.2.5. Pressure Gradient Force

The pressure gradient force results from the local fluid pressure gradient around the particle and is defined as:

$$\boldsymbol{F}_{P} = -\frac{m_{\mathrm{F}}}{\rho_{\mathrm{F}}} \nabla p \tag{6–27}$$

This force is only important if large fluids pressure gradients exist and if the particle density is smaller than or similar to the fluid density. Neglecting diffusive and source terms in the steady state momentum equation, the pressure gradient can be replaced by the velocity gradient. Assuming a constant fluids density, the pressure gradient force can be written as

$$\boldsymbol{F}_P = m_{\mathrm{F}} \left(\boldsymbol{U}_{\mathrm{F}} \ \nabla \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{R}_{\mathrm{F}} \right) = m_P \frac{\rho_{\mathrm{F}}}{\rho_P} \left(\boldsymbol{U}_{\mathrm{F}} \ \nabla \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{R}_{\mathrm{F}} \right) \tag{6–28}$$

leading to the following contribution to the right hand side of *Equation 6–7* (p. 242):

$$\boldsymbol{R} = R_{\mathrm{VM}} \frac{\rho_{\mathrm{F}}}{\rho_P} \left(\boldsymbol{U}_{\mathrm{F}} \ \nabla \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{R}_{\mathrm{F}} \right) \tag{6–29}$$

with

$$\boldsymbol{R}_{\mathrm{F}} = -2 \, \boldsymbol{\Omega} \times \boldsymbol{U}_{\mathrm{F}} - \boldsymbol{\Omega} \times \boldsymbol{\Omega} \times \boldsymbol{r}_P \tag{6–30}$$

necessary if the particles are solved in a rotating system. The $\boldsymbol{R}_{\mathrm{F}}$ term is shifted to the rotation term (see *Equation 6–19* (p. 244)), therefore, the implemented pressure gradient term reduces to

$$\boldsymbol{R} = R_{\mathrm{VM}} \frac{\rho_{\mathrm{F}}}{\rho_P} \boldsymbol{U}_{\mathrm{F}} \ \nabla \boldsymbol{U}_{\mathrm{F}} \tag{6–31}$$

The particle momentum source due to the pressure gradient force is computed from:

$$\frac{d\boldsymbol{S}}{dt} = -\boldsymbol{F}_{\mathrm{P}} \tag{6–32}$$

6.2.6. Turbulence in Particle Tracking

The calculation of the instantaneous fluid velocity, v_f in *Equation 6–6* (p. 241), depends on the flow regime and the type of particle tracking desired (mean or with turbulent dispersion). In laminar flows or in flows where mean particle tracking is calculated, v_f is equal to the mean local fluid velocity, $\overline{v_f}$, surrounding the particle. The path of a particle is deterministic (that is, there is a unique path for a particle injected at a given location in the flow).

In turbulent tracking, the instantaneous fluid velocity is decomposed into mean, $\overline{v_f}$, and fluctuating, v_f', components. Now particle trajectories are not deterministic and two identical particles, injected from a single point, at different times, may follow separate trajectories due to the random nature of the instantaneous fluid velocity. It is the fluctuating component of the fluid velocity which causes the dispersion of particles in a turbulent flow.

The model of turbulent dispersion of particles that is used, which is due to Gosman and Ioannides [147], assumes that a particle is always within a single turbulent eddy. Each eddy has a characteristic fluctuating velocity, v_f', lifetime, τ_e, and length, l_e. When a particle enters the eddy, the fluctuating velocity for that eddy is added to the local mean fluid velocity to obtain the instantaneous fluid velocity used in *Equation 6–19* (p. 244). The turbulent fluid velocity, v_f', is assumed to prevail as long as the particle/eddy interaction time is less than the eddy lifetime and the displacement of the particle relative to the eddy is less than the eddy length. If either of these conditions is exceeded, the particle is assumed to be entering a new eddy with new characteristic v_f', τ_e, and l_e.

The turbulent velocity, eddy and length and lifetime are calculated based on the local turbulence properties of the flow:

$$v_f' = \Gamma\,(2k/3)^{0.5}$$
$$l_e = \frac{C_\mu^{3/4}\,k^{3/2}}{\varepsilon} \qquad \textbf{(6–33)}$$

$$\tau_e = l_e / (2k/3)^{1/2} \qquad \textbf{(6–34)}$$

where k and ε are the local turbulent kinetic energy and dissipation, respectively, and C_μ is a turbulence constant. The factor $C_\mu^{3/4}$ was chosen to relate the characteristic length scale to the eddy dissipation length [39]. The variable Γ is a normally distributed random number which accounts for the randomness of turbulence about a mean value. Because of this randomness, each component of the fluctuating velocity (u', v', w') may have a different value in each eddy.

6.2.7. Turbulent Dispersion

If turbulent particle dispersion is enabled, you will need to track a much larger number of particles (usually an order of magnitude higher) because a stochastic method is used. This will greatly increase computational time; therefore, this is most often performed as a post-process where there is only one particle iteration.

If turbulent dispersion is used in an iterative situation, it may not be possible to achieve complete convergence because of the stochastic nature of the sources to the continuous phase, although the random number generator used in determining the eddies is reset on each particle iteration.

6.3. Heat and Mass Transfer

The following topics will be discussed:

6.3.1. Heat Transfer

The rate of change of temperature is governed by three physical processes: convective heat transfer, latent heat transfer associated with mass transfer, and radiative heat transfer.

The convective heat transfer Q_C is given by:

$$Q_C = \pi \, d \, \lambda \, \mathrm{Nu} \left(T_G - T_p \right) \tag{6–35}$$

where λ is the thermal conductivity of the fluid, T_G and T_p are the temperatures of the fluid and of the particle, and Nu is the Nusselt number given by:

$$\mathrm{Nu} = 2 + 0.6 \, \mathrm{Re}^{0.5} \left(\mu \, \frac{C_P}{\lambda} \right)^{\frac{1}{3}} \tag{6–36}$$

where C_P is the specific heat of the fluid, and μ and λ are the fluid dynamic viscosity and thermal conductivity respectively.

For cases including multiphase reactions, the convective heat transfer has a blowing correction based on the rate of mass transfer from the particle:

$$Q_C = Q_C \, \zeta \, \frac{e^{-\zeta}}{1 - e^{-\zeta}} \tag{6–37}$$

where ζ is given by

$$\zeta = C_p \, \frac{\left| \frac{dm}{dt} \right|}{\pi \, d \, \lambda \, \mathrm{Nu}} \tag{6–38}$$

and $\frac{dm}{dt}$ is the total mass transfer rate of the particle. This modification can be omitted by using the expert parameter setting: `pt heat transfer blowing correction = f`.

The heat transfer associated with mass transfer Q_M is given by the relation:

$$Q_M = \sum \frac{dm_C}{dt} \, V \tag{6–39}$$

where the sum is taken over all components of the particle for which heat transfer is taking place. The latent heat of vaporization V is temperature dependent, and is obtained directly from the MATERIALS information for the liquid in the particle and its vapor.

The radiative heat transfer, Q_R, for a particle with diameter d_p, uniform temperature T_p, and emissivity ε_p, is given by:

$$Q_R = \varepsilon_p \pi \, d_p^{\,2} \left(I - \sigma \, n^2 \, T_p^{\,4} \right) \tag{6–40}$$

where I is the Radiation Intensity on the particle surface at the location of the particle, n is the Refractive Index of the fluid, and σ is the Stefan-Boltzmann constant. An equivalent amount of heat can be removed from the radiation field.

The rate of change of temperature for the particle is then obtained from:

$$\sum \left(m_C \, C_P \right) \frac{dT}{dt} = Q_C + Q_M + Q_R \tag{6–41}$$

where the sum in this equation is taken over all components of the particle including those not affected by mass transfer.

The energy source to the continuous fluid is obtained from:

$$\frac{dS}{dt} = -Q_C \tag{6–42}$$

6.3.2. Simple Mass Transfer

Each component of mass being transferred between the continuous and particle phases satisfies the equation:

$$\frac{dm_C}{dt} = -\pi \, d_P \, \rho_F \, D_F \, \mathrm{Sh} \left(E \cdot Y_{P,C} - Y_{F,C} \right) \tag{6–43}$$

In this equation, m_C is the mass of the constituent in the particle, $Y_{P,C}$ is the mass fraction of component C in the particle, $Y_{F,C}$ is the mass fraction of component C in the surrounding fluid, E is the equilibrium mass fraction ratio, $\rho_F \, D_F$ is the dynamic diffusivity of the mass fraction in the continuum, and Sh is the Sherwood number given by:

$$\mathrm{Sh} = 2 + 0.6 \, \mathrm{Re}^{0.5} \left(\frac{\mu}{\rho \, D} \right)^{\frac{1}{3}} \tag{6–44}$$

If no value is set for the equilibrium mass fraction E, a value of 1 is used.

The simple model assumes that the mass transfer is driven only by concentration differences. While this may be appropriate in some situations, such as solids having a small moisture content, it does not adequately account for the vapor pressure dependence on particle temperature, which is imported for evaporating liquids. In these situations, the liquid evaporation model, presented below, is more appropriate.

The mass source to the continuous fluid is obtained from:

$$\frac{dS}{dt} = -\frac{dm_C}{dt} \tag{6–45}$$

6.3.3. Liquid Evaporation Model

The liquid evaporation model is a model for particles with heat and mass transfer. The model uses two mass transfer correlations depending on whether the droplet is above or below the boiling point.

The boiling point is determined through an Antoine equation (vapor pressure equation that describes the relation between vapor pressure and temperature for pure components, see Antoine Equation in the *CFX-Solver Modeling Guide*) that is given by

$$p_{vap} = p_{\text{scale}} \exp\left(A - \frac{B}{T_P + C}\right) \tag{6–46}$$

where A, B and C are user-supplied coefficients. The particle is boiling if the vapor pressure, p_{vap}, is greater than the ambient gas pressure, p_{ambient}.

When the particle is above the boiling point, the mass transfer is determined by:

$$\frac{dm_P}{dt} = -\frac{Q_C + Q_R}{V} \tag{6–47}$$

where V is the latent heat of evaporation of the particle component and where Q_C and Q_R are the convective and radiative heat transfers, respectively.

When the particle is below the boiling point, the mass transfer is given by:

$$\frac{dm_P}{dt} = \pi\, d_P\, \rho\, D\, \text{Sh}\, \frac{W_C}{W_G} \ln\left(\frac{1 - X_s^V}{1 - X_{\text{vap}}^V}\right) \tag{6–48}$$

where d_P is the droplet diameter, ρD is the dynamic diffusivity of the component in the continuum and Sh is the Sherwood number (see *Equation 6–44* (p. 249)). W_C and W_G are the molecular weights of the vapor and the mixture in the continuous phase, X_s^V is the equilibrium vapor mole fraction of the evaporating component at the droplet surface, and X_{vap}^V is the mole fraction of the evaporating component in the gas phase.

The mass source to the continuous fluid is obtained from:

$$\frac{dS}{dt} = -\frac{dm_P}{dt} \tag{6–49}$$

6.3.3.1. Extension of the Liquid Evaporation Model

The Liquid Evaporation model described in the previous section is limited to being applied to a single component of a particle. Under the following assumptions, the Liquid Evaporation model can be extended to multi-component evaporation:

1. Particle liquids are entirely miscible in all proportions
2. Intermolecular forces between different particle components are of equal strength

These assumptions imply that the liquids in a multi-component particle form an ideal mixture. By definition an ideal mixture is one which obeys Raoult's law (where the vapor pressure of an ideal mixture is dependent on the vapor pressure of each chemical component and the mole fraction of the component present in the mixture).

6.3.3.1.1. Examples of ideal mixtures

There is actually no such thing as an ideal mixture. However, some liquid mixtures get fairly close to being ideal. These are mixtures of closely similar substances.

Commonly quoted examples include:

- hexane and heptane
- benzene and methylbenzene

The more dissimilar the nature of the substances, the more strongly the solution is expected to deviate from ideality.

6.3.3.2. Determination of the Total Vapor Pressure of an Ideal Mixture

Based on the assumption of an ideal mixture, the total vapor pressure, p_{total}, can be calculated based on the assumption of Raoult's law

$$p_{\text{total}} = p_{\text{vap},1} + p_{\text{vap},2} + ... = \sum_i X_i^L p_{\text{vap},i}^{\text{pure}}\left(T_p\right) \tag{6–50}$$

where $p_{\text{vap},i}^{\text{pure}}\left(T_p\right)$ is the vapor pressure of a pure component i and X_i^L is the mole fraction of component i in the liquid phase. The vapor pressure of a pure component, $p_{\text{vap},i}^{\text{pure}}$ is determined from an Antoine equation. This equation has the following form:

$$p_{\text{vap},i}^{\text{pure}}(T_P) = p_{\text{scale}} \exp\left(A_i - \frac{B_i}{T_P + C_i}\right) \tag{6–51}$$

where A_i, B_i and C_i are the Antoine coefficients of binary mixture i.

Inserting *Equation 6–51* (p. 251) into *Equation 6–50* (p. 251) allows the calculation of the total vapor pressure of the mixture as follows:

$$p_{\text{total}} = \sum_i X_i^L p_{\text{scale},i} \exp\left(A_i - \frac{B_i}{T_P + C_i} \right) \quad \textbf{(6–52)}$$

It can be seen that the total vapor pressure is a function of the material properties of the components, the particle temperature, as well as the instantaneous composition of the evaporating droplet.

A particle is assumed to be boiling if the total vapor pressure, p_{total}, of all evaporating components is equal to or larger than the ambient gas pressure, p_{ambient}:

$$p_{\text{total}} \geq p_{\text{ambient}}$$

If the total vapor pressure is below the ambient gas pressure, the particle is assumed to not boil.

6.3.3.3. Diffusion Regime (Non-Boiling Particles)

The particle evaporates in the diffusion regime if the particle is below the boiling point.

The mass transfer of a single particle component can be derived from the mass conservation equation and can be written in the following form (see, for example, Abramzon and Sirignano [201] and Sazhin [202])

$$\frac{dm_{P,i}}{dt} = \pi\, d_P\, \rho\, D_i\, \text{Sh}\, \frac{W_{C,i}}{W_G} \ln\left(\frac{1 - X_{s,i}^V}{1 - X_{\text{vap},i}^V} \right) \quad \textbf{(6–53)}$$

where d_P is the droplet diameter, ρD_i is the dynamic diffusivity of the component i in the continuum, Sh is the Sherwood number, and where $W_{C,i}$ and W_G are the molecular weights of the vapor of component i and the mixture in the continuous phase. $X_{\text{vap},i}^V$ is the mole fraction of the vapor of component i in the continuum, $X_{s,i}^V$ is the equilibrium vapor mole fraction of component i at the droplet surface.

For the calculation of the molar concentration of component i on the particle surface, $X_{s,i}^V$, again the assumption of Raoult's law is applied:

$$X_{s,i}^V = \frac{p_{\text{vap},i}}{p_{\text{ambient}}} = X_i^L \frac{p_{\text{vap},i}^{\text{pure}}}{p_{\text{ambient}}} \quad \textbf{(6–54)}$$

The total mass flow rate is computed as the sum of the mass flow rates of all evaporating components:

$$\frac{dm_P}{dt} = \sum_i \frac{dm_{P,i}}{dt} = \sum_i \pi\, d_P\, \rho\, D_i\, \mathrm{Sh}\, \frac{W_{C,i}}{W_G} \ln\left(\frac{1 - X_{s,i}^V}{1 - X_{\mathrm{vap},i}^V} \right) \tag{6–55}$$

6.3.3.4. Boiling Particles

A particle is boiling if the total vapor pressure, p_{total}, is greater than or equal to the ambient gas pressure p_{ambient}.

When the particle reaches the boiling point, the mass transfer is determined by the convective and radiative heat transfer:

$$\sum_j \left(\frac{dm_j}{dt} V_j \right) = -\left(Q_C + Q_R \right) \tag{6–56}$$

where V_j is the latent heat of evaporation of component j, and where Q_C and Q_R are the convective and radiative heat transfer, respectively.

In order to determine the evaporation rate of a particular particle component, an additional assumption is required. It is assumed that the evaporation rate of each component is proportional to the ratio of the vapor pressure of this component to the total vapor pressure:

$$\frac{dm_i/dt}{\sum_j \left(dm_j/dt \right)} = \frac{p_{vap,i}}{p_{total}} \tag{6–57}$$

The component vapor pressure is computed from Raoult's law:

$$\frac{p_{vap,i}}{p_{total}} = \frac{X_i^L p_{vap,i}^{pure}}{p_{total}} = X_{s,i}^V \tag{6–58}$$

Rearranging Equation 6–57 (p. 253):

$$\frac{dm_i}{dt} = X_{s,i}^V \sum_j \left(dm_j/dt \right) \tag{6–59}$$

and substituting it into Equation 6–56 (p. 253) gives the total mass transfer rate:

$$\sum_j \left(dm_j/dt\right) = -\frac{\left(Q_C+Q_R\right)}{\sum_j \left(X^V_{s,j}\, V_j\right)} \tag{6–60}$$

Using *Equation 6–59* (p. 253) and *Equation 6–60* (p. 254) the following equation for the mass transfer of component i can be derived:

$$\frac{dm_i}{dt} = -\frac{X^V_{s,i}}{\sum_j \left(X^V_{s,j}\, V_j\right)}\left(Q_C+Q_R\right) = -\frac{(Q_C+Q_R)}{\hat{V}_i} \tag{6–61}$$

with the modified latent heat (averaged with the molar concentration):

$$\hat{V}_i = \frac{1}{X^V_{s,i}}\sum_j \left(X^V_{s,j}\, V_j\right) \tag{6–62}$$

Note that this modified latent heat is only used for the computation of the mass transfer rates in *Equation 6–61* (p. 254).

6.3.4. Oil Evaporation/Combustion

The oil combustion model uses the particle transport model to track evaporating oil droplets, which are modeled in a very similar way to the liquid evaporation model, and uses the eddy dissipation model for the combustion of the volatile gases in the gas phase.

- *The Eddy Dissipation Model* (p. 306)
- *Liquid Evaporation Model* (p. 250)

6.3.4.1. Light Oil Modification

The light oil modification bases the physical parameters used in the Reynolds number, the Nusselt number, and the Sherwood number on the gas assumed to be in the boundary layer of the droplet. This, in turn, depends upon the Antoine equation (that is, if the drop is boiling, the gas in the boundary layer is all volatiles). In the other extreme, the gas in the boundary layer consists entirely of the local gas mixture.

6.3.5. Reactions

The only reactions allowed are generic Arrhenius reactions, and some specific char reactions for coal combustion. The reactions determine the rate of change of species in the particle and the continuous phase, and the amount of heat released in the two phases.

6.3.5.1. Arrhenius Reactions

For the Arrhenius reactions, all reactants must be in the particle, but products can be in both the particle and the continuous phase. The rate of reaction is

$$R = k \cdot V \cdot \prod_{\alpha = \text{Reactants}} \left(\frac{m_\alpha}{V} \right)^{n_\alpha}$$

where m_α is the mass of reactant α, V is the particle volume, n_α is the order of reactant α, and

$$k = A \cdot {T_P}^m \cdot \exp\left(-T_A / T_P \right) \tag{6–63}$$

where A has units of

$$[kg^{1-n} m^{3(n-1)} K^{-m} s^{-1}], \quad n = \sum n_\alpha \tag{6–64}$$

Note that the Arrhenius expression is evaluated at the *particle* temperature, T_P

The rate of change of mass of a reactant A is then $-mc_A * R$, and of a product B is $+mc_B * R$, where mc is the mass coefficient of that species. The reaction orders n_α and the temperature exponent m may be fractional numbers, and not necessarily integers.

6.3.5.2. Heat Release

For any reaction, the heat release depends upon the reference specific enthalpies of the various reactants and products, if these are all provided. Alternatively, the heat release of a reaction can be specified directly and is then given by

$$H = -\left(\frac{dm}{dt} \right)_{\text{Ref}} L \tag{6–65}$$

where Ref refers to the reference material, and L is the latent heat at the particle temperature.

6.3.5.3. Char Oxidation

6.3.5.3.1. Field

The char reaction is determined by both the rate of diffusion to the surface and the rate of chemical reaction at the surface. The surface reaction is assumed to be first-order in the oxygen mole fraction.

The rate of diffusion of oxygen per unit area of particle surface is given by $k_d(X_g-X_S)$, where X_g is the mole fraction of oxygen in the furnace gases far from the particle boundary layer and X_S is the mole fraction of oxygen at the particle surface. The value of k_d is given by:

$$k_d = \frac{D_{ref}}{r_p} \left(\frac{T_p + T_g}{2\, T_{ref}} \right)^\alpha \frac{P_A}{P} \tag{6–66}$$

where:

- r_p is the particle radius

- T_P is the particle temperature
- T_g is the far-field gas temperature
- P is the local pressure
- P_A is atmospheric pressure
- D_{ref} is the dynamic diffusivity (recommended value is 1.8e-5 [kg m^-1 s^-1])
- T_{ref} is the reference temperature (recommended value is 293 [K])
- α is the exponent with value 0.75

The char oxidation rate per unit area of particle surface is given by $k_c X_S$. The chemical rate coefficient k_c is given by:

$$k_c = A_c T_p{}^n \exp\left(-\frac{T_c}{T_p} \right) \tag{6–67}$$

where:

- The parameters A_c and T_c depend on the type of coal, and are specified as input parameters.
- The default value of n is 0.0.
- For this model, k_d and k_c are in units of [kg m^-2 s^-1],
- Recommended values for A_c and T_c are 497 [kg m^-2 s^-1] and 8540 K [80].

  By equating the diffusion rate and the chemical reaction rate to eliminate X_S, the overall char reaction rate of a particle is given by:

$$\frac{dm_c}{dt} = \left(k_d^{-1} + k_c^{-1} \right)^{-1} X_g 4\pi R_p^2 \frac{P}{P_A} \tag{6–68}$$

and is controlled by the smaller of the rates k_d and k_c.

6.3.5.3.2. Gibb

The oxidation mechanism of carbon can be characterized by the parameter φ so that oxides are produced according to the equation:

$$\varphi\ \mathrm{C} + \mathrm{O}_2 \rightarrow 2\,(\varphi - 1)\ \mathrm{CO} + (2 - \varphi)\ \mathrm{CO}_2 \tag{6–69}$$

The value of φ is assumed to depend on the particle temperature T_P:

$$\frac{2(\varphi-1)}{2-\varphi} = A_S \exp\left(-\frac{T_S}{T_p}\right) \quad \textbf{(6–70)}$$

where the constants are given by Gibb as A_S = 2500 and T_S = 6240 K.

By solving the oxygen diffusion equation analytically, the following equation is obtained for the rate of decrease in the char mass m_c:

$$\frac{dm_c}{dt} = -\frac{3\varphi}{1-\varepsilon}\frac{M_c}{M_{O2}}\frac{\rho_\infty}{\rho_c}\left(k_1^{-1} + (k_2+k_3)^{-1}\right)^{-1} m_c \quad \textbf{(6–71)}$$

The far field oxygen concentration ρ_∞ is taken to be the time-averaged value obtained from the gas phase calculation, and ρ_c is the density of the char. Physically, k_1 is the rate of external diffusion, k_2 is the surface reaction rate, and k_3 represents the rate of internal diffusion and surface reaction. These are defined as follows:

$$k_1 = \frac{D}{r_p^2} \quad \textbf{(6–72)}$$

where D is the external diffusion coefficient of oxygen in the surrounding gas. The coefficient is calculated in the same way as for the Field model, except in this model, kinematic diffusivity is used instead of dynamic diffusivity:

$$D = \frac{D_{ref}}{\rho_{\mathrm{fluid}}}\left(\frac{T_p+T_g}{2\,T_{ref}}\right)^{\alpha} \quad \textbf{(6–73)}$$

where:

- D_{ref} is the dynamic diffusivity (recommended value is 1.8e-5 [kg m^-1 s^-1])
- T_{ref} is the reference temperature (recommended value is 293 [K])
- α is the exponent with value 0.75.

$$k_2 = (1-\varepsilon)\,\frac{k_c}{R} \quad \textbf{(6–74)}$$

where k_c is the carbon oxidation rate, defined by the modified Arrhenius equation

$$k_c = A_c T_p \exp\left(-\frac{T_c}{T_p} \right) \tag{6–75}$$

The default values of the model constants are A_c = 14 [m s^-1 K^-1] and T_c = 21580 K. Further:

$$k_3 = k_c (\beta \coth\beta - 1) / (\beta^2 a) \tag{6–76}$$

where:

$$\beta = R \left(\frac{k_c}{D_p \varepsilon a} \right)^{0.5} \tag{6–77}$$

The pore diffusivity, D_p, is computed from external diffusivity, D, according to:

$$D_p = \mathrm{effic} \times D \tag{6–78}$$

Note that the units of k_1, k_2 and k_3 for this model are s^{-1}, and the units for k_c in *Equation 6–75* (p. 258) differ from those in *Equation 6–67* (p. 256) in the Field model. "effic" is the so-called "internal diffusion efficiency" and describes the effective internal diffusion coefficient of oxygen within the particle pores. It is a user-accessible CCL parameter.

6.3.5.4. Radiative Preheating

The stability of a pulverized coal flame depends on the feedback of heat from the flame zone to the raw coal as it emerges from the burner. In general, this preheating is supplied by a combination of convective heating by recirculating product gases, as well as by the absorption of radiation. CFX calculates the radiative heating using *Equation 6–40* (p. 249).

The value of the particle emissivity ε_p is expected to change as pyrolysis proceeds (that is, it varies depending upon the mass fractions of coal and char). The present model assumes a linear variation in ε_p from the raw coal value ε_p(coal) to the value for char ε_p(char). That is:

$$\varepsilon_p = \left(1 - f_v\right) \varepsilon_p (\mathrm{coal}) + f_v \varepsilon_p (\mathrm{char}) \tag{6–79}$$

where f_v is the fractional yield of volatiles. Typical values for ε_p are 1 for coal and 0.6 for char.

6.3.5.5. Coal Combustion

6.3.5.5.1. Coal Combustion - Gas Phase

Coal combustion is calculated by combining a particle transport calculation of the coal particles with an eddy dissipation calculation for the combustion of the volatile gases in the gas phase. Two separate gases are given off by the particles, the volatiles and the char products that come from the burning of carbon within the particle.

Gas phase combustion is modeled by means of regular single phase reactions. A transport equation is solved in the fluid for each material given off by the particles. The 'volatiles' may be either a pure substance, a fixed composition mixture, or several independent materials.

6.3.5.5.2. Coal Decomposition

Pulverized coal particles are treated in CFX as non-interacting spheres with internal reactions, heat transfer and full coupling of mass, momentum and energy with the gaseous phase. The combustion of a coal particle is a two stage process: the devolatilization of the raw coal particle followed by the oxidation of the residual char to leave incombustible ash. The devolatilization is usually modeled as a single step or a two step process. The char oxidation is modeled either as a global reaction, or using an analytical solution for the diffusion and reaction of oxygen within the pores of the char particle.

The devolatilization and char oxidation processes can occur on time scales of order milliseconds which are several orders of magnitude less than the typical residence time of the particle in the furnace. Large variations in time scales can result in numerically stiff equations, which can cause accuracy problems with explicit integration algorithms. The CFX implementation of the particle transport model bases its timestep on the reaction rate to ensure that the solution has the required accuracy.

6.3.5.5.3. Devolatilization

Devolatilization can be modeled by one or more reaction steps using the generic Arrhenius multiphase reactions capability, although normally the process is represented by one or two reaction steps. The simpler model is the single reaction model of Badzioch and Hawksley [78]. If, at time t, the coal particle consists of mass fractions C_O of raw (that is, unreacted) coal, C_{ch} of residual char, and C_A of ash, then the rate of conversion of the raw coal is:

$$\frac{dC_O}{dt} = -k_V C_O \qquad \textbf{(6–80)}$$

and the rate of production of volatiles in the gas phase is given by:

$$\frac{dV}{dt} = Y\, k_V C_O \qquad \textbf{(6–81)}$$

where Y is the actual yield of volatiles (that is, the proximate yield multiplied by a factor to correct for the enhancement of yield due to rapid heating), so that the rate of char formation is:

$$\frac{dC_{ch}}{dt} = (1 - Y)\, k_V C_O \qquad \textbf{(6–82)}$$

The rate constant k_V is expressed in Arrhenius form as:

$$k_V = A_V \exp\left(-\frac{E_V}{R\ T_P} \right) \tag{6–83}$$

where T_P is the temperature of coal particles (assumed uniform), R is the gas constant, and A_V and E_V are constants determined experimentally for the particular coal.

Often the volatiles yield of a particular type of coal is known only from laboratory proximate analysis, where the heating rate is low and the volatiles escaping from the coal may undergo secondary reactions including cracking and carbon deposition on solid surfaces. It has been found experimentally that volatiles released from pulverized coal particles widely dispersed in a gas and heated quickly to typical furnace temperatures can produce a yield greater by as much as a factor of two than the proximate value. If the single reaction model is used, it is difficult to know which data to use because the coal particles experience a wide range of temperatures as they disperse in the furnace.

Bituminous coals generally have a volatiles yield that depends strongly on temperature and heating rates. In such cases, it is important to take account of this dependence, for example by using a multiple reaction model of devolatilization. As an alternative to the single reaction model, it is better to use the model of Ubhayakar et al. [79] in which two reactions with different rate parameters and volatiles yields compete to pyrolyse the raw coal. The first reaction dominates at lower particle temperatures and has a lower yield Y_1 than the yield Y_2 of the second reaction, which dominates at higher temperatures. As a result, the final yield of volatiles will depend on the temperature history of the particle, will increase with temperature, and will lie somewhere between Y_1 and Y_2. In this model, the mass fraction of combustible material (the raw coal) is specified as the mass fraction of volatiles because all this material could be converted to volatiles.

Again it is assumed that a coal particle consists of mass fraction C_O of raw coal, C_{ch} of residual char after devolatilization has occurred, and C_A of ash. The rate constants k_1 and k_2 of the two reactions determine the rate of conversion of the raw coal:

$$\frac{dC_O}{dt} = -\ (k_1 + k_2)\ C_O \tag{6–84}$$

the rate of volatiles production is given by:

$$\frac{dV}{dt} = (\ Y_1 k_1 + Y_2 k_2\ )\ C_O \tag{6–85}$$

and so the rate of char formation is:

$$\frac{dC_{ch}}{dt} = \Big(\ (1 - Y_1)\ k_1 + (1 - Y_2)\ k_2 \Big)\ C_O \tag{6–86}$$

The initial value of C_O is equal to (1-C_A). The value of Y_1, and parameters A_1 and E_1 which define k_1 in the Arrhenius equation, analogous to *Equation 6–83* (p. 260), are obtained from proximate analysis of the coal. Y_2, A_2 and E_2 are obtained from high temperature pyrolysis. Note that the yields are defined on a dry ash-free (DAF) basis.

Typically, the coal particle will swell due to the gas release during the devolatilization phase. The model assumes that the particle diameter changes in proportion to the volatiles released and the fractional increase in the mean diameter of the coal particle after complete devolatilization must be specified as input to the model. The particle diameter change due to swelling is calculated using the following relation:

$$\frac{d}{dt}d_p = C_s\, d_{0,p}\, \frac{\dot{m}_{ref}}{m_{ref,0}} \qquad \textbf{(6–87)}$$

where:

- d_p is the current particle diameter
- C_s is the swelling coefficient
- $d_{0,p}$ is the particle diameter at the start of devolatilization
- $\dot{m}_{ref}$ is the rate of change of mass of the reference material
- $m_{ref,0}$ is the mass of the reference material at the start of the devolatilization

When the swelling coefficient is equal to 0.0, the particle diameter stays constant during devolatilization.

6.3.6. Hydrocarbon Fuel Analysis Model

The ‘Hydrocarbon Fuel Analysis’ model enables you to define all the properties of a solid or liquid hydrocarbon fuel in a user-friendly way. The solver uses the provided information to derive the parameters of related objects, such as the initial particle mass fractions, material properties of the volatiles released, and stoichiometric or mass coefficients of reactions.

The primary input data corresponds 1-to-1 to what, typically, is available from standard analysis of the solid or liquid fuel:

- Heating value (higher or lower heating value)
- Proximate analysis (mass fractions of ash, moisture, fixed carbon, and volatiles)
  - As Received: 1 = ash + moisture + fixed carbon + volatiles
  - Dry Ash Free: 1 = fixed carbon + volatiles
- Ultimate analysis (mass fractions of carbon, hydrogen, oxygen, nitrogen, sulfur, and chlorine)
  - As Received: 1 = ash + moisture + carbon + hydrogen + oxygen + nitrogen + sulfur + chlorine
  - Dry Ash Free: 1 = carbon + hydrogen + oxygen + nitrogen + sulfur + chlorine.

Some additional input data is required by the solver, for which the default values should be appropriate in many cases:

- Volatiles yield enhancement: Ratio of actual yield under rapid heating to that determined at a slow heating rate in the proximate analysis.
- Average molar mass of volatiles released. Three options are available:
  - `Automatic`: Computes average molar mass from volatiles elementary composition, assuming a mixture of CH4, CO, H2 and H2O

  - `Value`: Use value specified in fuel analysis
  - `Use Material Definition`: Use value defined in volatiles fuel material.
- Reference conditions for heating value (temperature and pressure)
- Moisture latent heat in case of higher heating value specified:
  - `Automatic`: Standard value (2.4423 [MJ/kg])
  - `Value`: User value for given temperature and pressure.

The above data are used to derive the following quantities:

- Initial mass fractions for particle (ash, char and raw combustible).

  Note that initial char mass fraction will typically be zero, as char is produced from raw combustible during pyrolysis.
- Fuel volatiles material properties:
  - Average molar mass
  - Specific reference enthalpy (heating value)
  - Carbon, hydrogen and oxygen content.
- Stoichiometric coefficients for gas phase reactions:
  - Fuel volatiles oxidation
  - NO reburn by fuel.
- Mass coefficients for multiphase reactions:
  - Devolatilization (decomposition of raw combustible into char and volatiles)
  - Char oxidation.

These calculations are performed using a model fuel determined by the fuel analysis data. The model fuel has all chlorine removed but accounts for the oxygen needed to oxidize sulfur to SO_2. Nitrogen is included into the model fuel if the multiphase reactions are setup to release HCN to the gas phase, otherwise, the fuel nitrogen is removed.

The total amount of material released to the gas phase during devolatilization is the actual volatiles yield plus the moisture. Carbon, hydrogen and oxygen content of the volatiles are computed from ultimate analysis, which in turn defines the stoichiometric coefficients in the gas phase reactions involving the volatiles material. When the fuel nitrogen model is enabled, corrections are made in order to account for the carbon and hydrogen released as HCN.

6.4. Basic Erosion Models

The following topics will be discussed:

6.4.1. Model of Finnie

The wear of a wall due to the erosive effect of particle impacts is a complex function of particle impact, particle and wall properties. For nearly all metals, erosion is found to vary with impact angle and velocity according to the relationship [63]:

$$E = k\ V_P^n f\ (\gamma) \tag{6–88}$$

where E is a dimensionless mass, V_P is the particle impact velocity and $f\ (\gamma)$ is a dimensionless function of the impact angle. The impact angle is the angle in radians between the approaching particle track and the wall. The value of the exponent, n, is generally in the range 2.3 to 2.5 for metals.

Finnie's model of erosive wear [64] relates the rate of wear to the rate of kinetic energy of impact of particles on the surface, using $n = 2$:

$$E = k\ V_P^2 f\ (\gamma) \tag{6–89}$$

where:

$$\begin{aligned} f\ (\gamma) &= \frac{1}{3}\cos^2\gamma & \text{if } \tan\gamma > \frac{1}{3} \\ f\ (\gamma) &= \sin\ (2\gamma) - 3\sin^2\gamma & \text{if } \tan\gamma \le \frac{1}{3} \end{aligned} \tag{6–90}$$

6.4.1.1. Implementation in CFX

In CFX, the need to adjust the dimension of k to obtain a non-dimensional erosion factor is overcome by specifying:

$$E = \left(\frac{V_P}{V_0}\right)^n f\ (\gamma) \tag{6–91}$$

where V_0 is equal to $\left(\frac{1}{\sqrt[n]{k}}\right)$ and defaults to 1 [m/s] in CFX. Typical values for steel are 590 m/s.

6.4.2. Model of Tabakoff and Grant

In the erosion model of Tabakoff and Grant, the erosion rate E is determined from the following relation:

$$E = k_1 f\ (\gamma)\ V_P^2 \cos^2\gamma \left[1 - R_T^2\right] + f\ (V_{PN}) \tag{6–92}$$

where:

$$f(\gamma) = \left[1 + k_2 k_{12} \sin\left(\gamma \frac{\pi/2}{\gamma_0}\right)\right]^2 \tag{6–93}$$

$$R_T = 1 - k_4 V_P \sin\gamma \tag{6–94}$$

$$f(V_{PN}) = k_3 (V_P \sin\gamma)^4 \tag{6–95}$$

$$k_2 = \begin{cases} 1.0 & \text{if } \gamma \le 2\gamma_0 \\ 0.0 & \text{if } \gamma > 2\gamma_0 \end{cases} \tag{6–96}$$

Here E is the dimensionless mass (mass of eroded wall material divided by the mass of particle). V_P is the particle impact velocity. γ is the impact angle in radians between the approaching particle track and the wall, γ_0 being the angle of maximum erosion. k_1 to k_4, k_{12} and γ_0 are model constants and depend on the particle/wall material combination.

6.4.2.1. Implementation in CFX

The model constants in the original formulation of Tabakoff and Grant are only valid for particle velocities specified in feet per second [ft/s]. The Tabakoff formulation is modified in CFX as outlined below:

$$E = f(\gamma) \left(\frac{V_P}{V_1}\right)^2 \cos^2\gamma \left(1 - R_T^2\right) + f(V_{PN}) \tag{6–97}$$

where:

$$f(\gamma) = \left[1 + k_2 k_{12} \sin\left(\gamma \frac{\pi/2}{\gamma_0} \right) \right]^2 \quad \textbf{(6–98)}$$

$$R_T = 1 - \frac{V_P}{V_3} \sin\gamma \quad \textbf{(6–99)}$$

$$f(V_{PN}) = \left(\frac{V_P}{V_2} \sin\gamma \right)^4 \quad \textbf{(6–100)}$$

$$k_2 = \begin{cases} 1.0 & \text{if } \gamma \le 2\gamma_0 \\ 0.0 & \text{if } \gamma > 2\gamma_0 \end{cases} \quad \textbf{(6–101)}$$

Note

In CFX, the erosive wear is calculated as grams of eroded material per gram of colliding particles (not milligrams of eroded material per gram of colliding particles, as may be found in literature).

6.4.2.1.1. Mapping of CFX to Original Tabakoff Constants

To make the model more general, the model is rewritten so that all model constants have a dimension of velocity. The following list shows the link between the constants of the original model and those in CFX:

6.4.2.1.2. Constants

Value	Dimensions	CFX-Pre Variable
k_{12}	(dimensionless)	K12 Constant
k_2	(dimensionless)	
V_1	[Velocity]	Reference Velocity 1
V_2	[Velocity]	Reference Velocity 2
V_3	[Velocity]	Reference Velocity 3
γ_0	[deg]	Angle of Maximum Erosion

where:

$$V_1 = 1/\sqrt{k_1} \tag{6–102}$$

$$V_2 = 1/\left(\sqrt[4]{k_3}\right) \tag{6–103}$$

$$V_3 = 1/k_4 \tag{6–104}$$

6.4.3. Overall Erosion Rate and Erosion Output

The erosion of a wall due to a particle is computed from the following relation:

$$\text{ErosionRate} = E * \dot{N} * m_P \tag{6–105}$$

as an individual representative particle represents many actual particles. Here m_P is the mass of the particle and $\dot{N}$ is its number rate. The overall erosion of the wall is then the sum over all particles. This gives an erosion rate in kg/s, and an erosion rate density variable in the res file and post-processor in kg/s/m^2. Note that this erosion rate is only a qualitative guide to erosion, unless precise values for the model constants are known.

6.5. Spray Breakup Models

Spray processes play an important role in many technical systems and industrial applications. Examples are spray painting or fuel injection systems.

The following sections give an overview about the spray breakup models that are implemented in ANSYS CFX 14.0:

6.5.1. Primary Breakup/Atomization Models

The main task of primary breakup (or atomizer) models is to determine starting conditions for the droplets that leave the injection nozzle. These conditions are:

- Initial particle radius
- Initial particle velocity components
- Initial spray angle

The radius, velocity, and spray angle parameters are influenced by the internal nozzle flow (cavitation and turbulence induced disturbances), as well as by the instabilities on the liquid-gas interface.

The following primary breakup models are available in CFX:

- *Blob Method* (p. 267)
- *Enhanced Blob Method* (p. 267)
- *LISA Model* (p. 269)

- *Turbulence Induced Atomization* (p. 273)

For the Blob, Enhanced Blob, and the LISA model, the initial injection spray angle must be specified explicitly, while for the Turbulence Induced Atomization model, the spray angle is computed within the model.

A large variety of approaches of different complexities are documented in literature. For a comprehensive model overview, see Baumgarten et al. [120].

6.5.1.1. Blob Method

This is one of the simplest and most popular approaches to define the injection conditions of droplets. In this approach, it is assumed that a detailed description of the atomization and breakup processes within the primary breakup zone of the spray is not required. Spherical droplets with uniform size, $D_p = D_{\mathrm{nozzle}}$, are injected that are subject to aerodynamic induced secondary breakup.

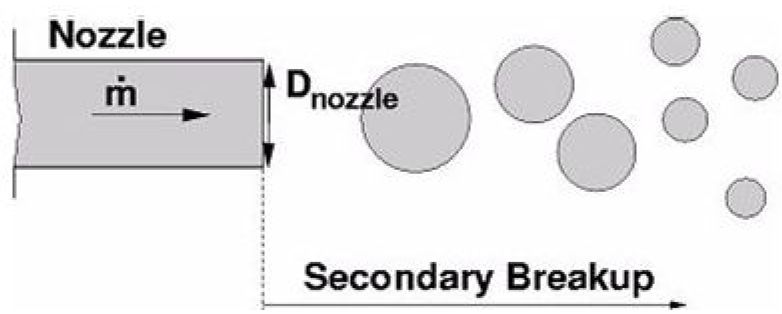
Assuming non-cavitating flow inside the nozzle, it is possible to compute the droplet injection velocity by conservation of mass:

$$U_{P,inital}(t) = \frac{\dot{m}_{nozzle}(t)}{A_{nozzle}\,\rho_P} \qquad \textbf{(6–106)}$$

A_{nozzle} is the nozzle cross-section and $\dot{m}_{\mathrm{nozzle}}$ the mass flow rate injected through the nozzle.

The spray angle is either known or can be determined from empirical correlations. The blob method does not require any special settings and it is the default injection approach in CFX.

Figure 6.1 Blob Method



6.5.1.2. Enhanced Blob Method

Kuensberg et al. [110] have suggested an enhanced version of the blob-method. Similar to the blob-method, it is assumed that the atomization processes need not be resolved in detail. Contrary to the standard blob-method, this method enables you to calculate an effective injection velocity and an effective injection particle diameter taking into account the reduction of the nozzle cross section due to cavitation.

During the injection process, the model determines if the flow inside the nozzle is cavitating or not and dynamically changes the injection particle diameter and the particle injection velocity. The decision, whether the flow is cavitating or not, is based on the value of the static pressure at the vena contracta, P_{vena}, that is compared to the vapor pressure, P_{vapor}.

$$P_{\text{vena}}(t) = P_1 - \frac{\rho_P}{2} U_{\text{vena}}^2(t) \tag{6–107}$$

with:

$$U_{\text{vena}}(t) = \frac{U_{\text{mean}}(t)}{C_c} \tag{6–108}$$

C_c is the coefficient of contraction that depends on nozzle geometry factor, such as nozzle length versus nozzle diameter or the nozzle entrance sharpness [114].

If P_{vena} is higher than the vapor pressure, P_{vapor}, the flow remains in the liquid phase and the injection velocity is set equal to U_{mean}.

$$U_{\text{mean}}(t) = \frac{\dot{m}(t)}{A_{\text{nozzle}}\,\rho_P} \tag{6–109}$$

The initial droplet diameter is equal to the nozzle diameter, D_{nozzle}.

However, if P_{vena} is lower than P_{vapor}, it is assumed that the flow inside the nozzle is cavitating and the new effective injection velocity, U_{eff}, and injection diameter, D_{eff}, are computed from a momentum balance from the vena contracta to the nozzle exit (2):

$$U_{eff}(t) = \frac{A_{\text{nozzle}}}{\dot{m}(t)} \left(P_{\text{vapor}} - P_2\right) + U_{\text{vena}}(t) \tag{6–110}$$

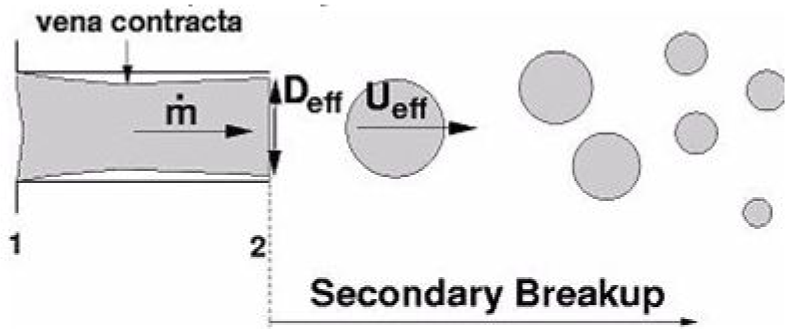
and:

$$D_{eff}(t) = \sqrt{\frac{4\,A_{eff}(t)}{\pi}} \tag{6–111}$$

with:

$$A_{eff}(t) = \dot{m} / \left(U_{eff}(t)\ \rho_P \right) \tag{6–112}$$

Figure 6.2 Enhanced-blob Method



6.5.1.2.1. Input Parameters for the Enhanced Blob Method

The following information is required for the enhanced blob method:

- Contraction coefficient due to cavitation inside the injection nozzle.
- Injection total pressure of the liquid - This information is required to compute the static pressure of the liquid at the vena contracta.
- Injection total temperature of the liquid - This information is required to compute the static temperature of the liquid at the vena contracta, which is required to determine the fluid vapor pressure from an Antoine equation (homogeneous binary mixture).
- Vapor pressure of the particle fluid

  Two options are available to compute the material vapor pressure: `Automatic` and `Particle Material Vapor Pressure`.

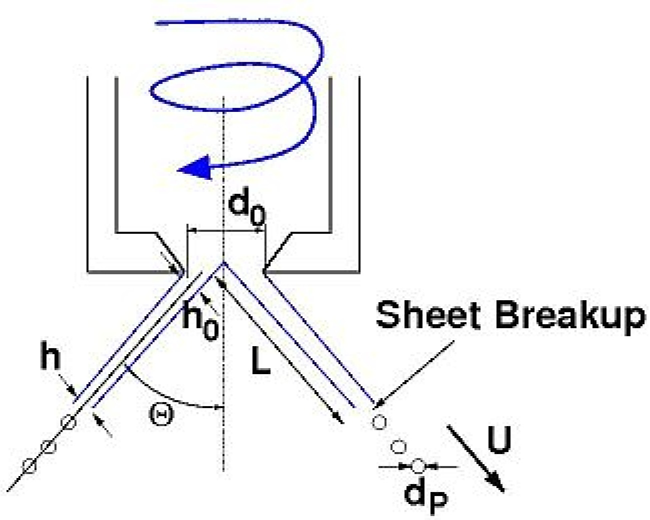
  - To specify the material vapor pressure directly, use the `Particle Material Vapor Pressure` option.
  - To have CFX compute the particle material vapor pressure from a homogeneous binary mixture, use the `Automatic` option.
- Normal distance of the pressure probe from the injection center - The fluid pressure at this position (marked as position **2** in *Figure 6.2* (p. 269)) will be used to determine the acceleration of the liquid from the vena contract to the injection nozzle outlet.

6.5.1.3. LISA Model

The LISA (Linearized Instability Sheet Atomization) model is able to simulate the effects of primary breakup in pressure-swirl atomizers as described in this section and presented in detail by Senecal et al. [126].

In direct-injection spark ignition engines, pressure swirl atomizers are often used in order to establish hollow cone sprays. These sprays are typically characterized by high atomization efficiencies. With pressure swirl injectors, the fuel is set into a rotational motion and the resulting centrifugal forces lead to a formation of a thin liquid film along the injector walls, surrounding an air core at the center of the injector. Outside the injection nozzle, the tangential motion of the fuel is transformed into a radial component and a liquid sheet is formed. This sheet is subject to aerodynamic instabilities that cause it to break up into ligaments.

Figure 6.3 Pressure Swirl Atomizer



Within the LISA model, the injection process is divided into two stages:

- *Film Formation* (p. 270)
- *Sheet Breakup and Atomization* (p. 271)

6.5.1.3.1. Film Formation

Due to the centrifugal motion of the liquid inside the injector, a liquid film along the injector walls is formed. The film thickness, h_0, at the injector exit can be expressed by the following relation:

$$\dot{m} = \pi \, \rho_P \, u \, h_0 \, (d_0 - h_0) \quad \textbf{(6–113)}$$

where $\dot{m}$ is the mass flow rate through the injector, ρ_P is the particle density, and d_0 is the injector exit diameter. The quantity u is the axial velocity component of the film at the injector exit and depends on internal details of the injector. In CFX, u is calculated using the approach of Han et al. [127]. It is assumed that the total velocity, U, is related to the injector pressure by the following relation:

$$U = k_{\mathrm{v}} \sqrt{\frac{2 \, \Delta p}{\rho_P}} \quad \textbf{(6–114)}$$

where Δp is the pressure difference across the injector and k_{v} is the discharge coefficient, which is computed from:

$$k_{\mathrm{v}} = \max \left[0.7 \, , \frac{4 \, \dot{m}}{\pi \, d_0^2 \, \rho_P \, \cos\theta} \sqrt{\frac{\rho_P}{2 \, \Delta p}} \right] \quad \textbf{(6–115)}$$

Assuming that Δp is known, the total injection velocity can be computed from *Equation 6–114* (p. 270). The axial film velocity component, u, is then derived from:

$$u = U\cos\Theta \tag{6–116}$$

where Θ is the spray angle, which is assumed to be known. At this point, the thickness, h_0, and axial velocity component of the liquid film are known at the injector exit. Note that the computed film thickness, h_0, will be equal to half the injector nozzle diameter if the discharge coefficient, k_{v}, is larger than 0.7. The tangential component of velocity ($w = U\sin\Theta$) is assumed to be equal to the radial velocity component of the liquid sheet downstream of the nozzle exit. The axial component of velocity is assumed to remain constant.

6.5.1.3.2. Sheet Breakup and Atomization

After the liquid film has left the injector nozzle, it is subject to aerodynamic instabilities that cause it to break up into ligaments. The theoretical development of the model is given in detail by Senecal et al. [126] and is only briefly repeated here.

The model assumes that a two-dimensional, viscous, incompressible liquid sheet of thickness $2\,h$ moves with velocity U through a quiescent, inviscid, incompressible gas medium. A spectrum of infinitesimal disturbances is imposed on the initially steady motion, and is given in terms of the wave amplitude, η, as:

$$\eta = \eta_0\, e^{i k x + \omega t} \tag{6–117}$$

where η_0 is the initial wave amplitude, $k = (2\,\pi)/\lambda$ is the wave number, $\omega = \omega_{\mathrm{r}} + i\,\omega_{\mathrm{i}}$ is the complex growth rate, x is the streamwise film coordinate, and t is the time. The most unstable disturbance, Ω, has the largest value of ω_{r} and is assumed to be responsible for sheet breakup.

As derived by Senecal et al. [126], Ω can be computed by finding the maximum of the following equation:

$$\omega_{\mathrm{r}} = -\frac{2\,\nu_P\,k^2\tanh(k\,h)}{\tanh(k\,h)+Q} + \frac{\sqrt{4\,\nu_P^2\,k^4\,(\tanh(k\,h))^2 - Q^2\,U^2\,k^2 - (\tanh(k\,h)+Q)\left(\dfrac{\sigma\,k^3}{\rho_P} - Q\,U^2\,k^2\right)}}{\tanh(k\,h)+Q} \tag{6–118}$$

here $Q = \rho_{\mathrm{g}}/\rho_P$, that is, the ratio of local gas density ρ_{g} to particle densities ρ_{p}, and σ is the surface tension coefficient. Once Ω is known, the breakup length, L, and the breakup time, τ, are given by:

$$L=\frac{U}{\Omega}\ln\left(\frac{\eta_\mathrm{b}}{\eta_0}\right) \tag{6–119}$$

$$\tau=\frac{1}{\Omega}\ln\left(\frac{\eta_\mathrm{b}}{\eta_0}\right) \tag{6–120}$$

where η_b is the critical wave amplitude at breakup and $\ln\left(\eta_\mathrm{b}/\eta_0\right)$ is an empirical sheet constant with a default value of 12.

The unknown diameter, d_1, of the ligaments at the breakup point is obtained from a mass balance. For wavelengths that are long compared to the sheet thickness ($\mathrm{We}_\mathrm{p}<27/16$; here the Weber Number, We_p, is based on half the film thickness and the gas density), d_1 is given by:

$$d_1=\sqrt{\frac{f_l\,h}{K_s}} \tag{6–121}$$

using the long wave ligament factor, f_l, of 8.0 (default) and K_s is the wave number corresponding to the maximum growth rate, Ω.

For wavelengths that are short compared to the sheet thickness, d_1 is given by:

$$d_1=\sqrt{\frac{f_s\,h}{K_s}} \tag{6–122}$$

In this case, the short wave ligament factor, f_s, of 16.0 (default) is used and K_s is given by:

$$K_s=\frac{\rho_\mathrm{g}\,U^2}{2\,\sigma} \tag{6–123}$$

The most probable droplet diameter that is formed from the ligaments is determined from:

$$d_0=1.88\,d_1(1+3\,\mathrm{Oh})^{1/6} \tag{6–124}$$

using the droplet diameter size factor of 1.88 (default) and Oh is the particle Ohnesorge number that is defined as:

$$\mathrm{Oh} = \frac{\sqrt{\mathrm{We}_p}}{\mathrm{Re}_p} \quad \textbf{(6–125)}$$

where We_p is the Weber Number based on half the film thickness and the gas density. Re_p is the Reynolds Number based on the slip velocity.

6.5.1.3.3. User Input Data for the LISA Model

Some of the settings for the simulation of primary breakup using the LISA model are:

- Injection pressure difference, Δp
- Nozzle geometry (outer nozzle diameter d_0, injection half cone angle Θ)
- Normal distance of the density probe from the injection center:

  The LISA model requires knowledge of *downstream density*. The solver finds this density by probing the flow-field at a user-defined distance downstream of the injection location. This distance is a user input given via `Density Probe Normal Distance` option in CFX-Pre.
- Swirl Definition

  To add swirl to the flow of particles that stream from the injection region, select the **Swirl Definition** check box and then specify a non-dimensional value to impart a tangential velocity. The value that you specify is multiplied by the magnitude of the axial velocity (the axial velocity being directed normal to the injector inlet area, with a magnitude that satisfies mass continuity) in order to establish the (single) value for tangential velocity, to be applied over the entire injector inlet area. The direction of swirl depends on whether you specify a positive or negative value; the positive swirl direction is established by using the right-hand rule and the injector axis.
- Particle mass flow rate, $\dot{m}$

For a complete list of *required* and *optional* settings for the LISA model and primary breakup models in general, see Settings for Particle Primary Breakup in the *CFX-Pre User's Guide*.

6.5.1.4. Turbulence Induced Atomization

This atomization model is based on the modification of the Huh and Gosman model as suggested by Chryssakis (see [155, 156]). It accounts for turbulence induced atomization and can also be used to predict the initial spray angle. The model assumes that turbulent fluctuations within the injected liquid produce initial surface perturbations, which grow exponentially by the aerodynamic induced Kelvin-Helmholtz mechanism and finally form new droplets. The initial turbulent fluctuations at the nozzle exit are estimated from simple overall mass, momentum and energy balances. Droplets are injected with an initial diameter equal to the nozzle diameter, D_{nozzle}. It is assumed that unstable Kelvin-Helmholtz waves grow on their surfaces, which finally break up the droplet. The effects of turbulence are introduced by assuming, that the atomization length scale, L_A, is proportional to the turbulence length scale and that the atomization time scale, t_A, can be written as a linear combination of the turbulence time scale (from nozzle flow) and the wave growth time scale (external aerodynamic forces). At breakup, child droplets also inherit a velocity component normal to their parents' velocity direction. This is different to the model suggested by Chryssakis, where child droplets did not get this velocity component.

The initial spray angle is computed from the following relation:

$$\tan\left(\Theta/2\right) = \frac{L_A/t_A}{U_{\text{inj}}} \quad \textbf{(6–126)}$$

The model does not take effects of cavitation into account, but assumes that the turbulence at the nozzle exit completely represents the influence of the nozzle characteristics on the atomization process.

6.5.1.4.1. User Input Data for Turbulence Induced Atomization

For the simulation of primary breakup using turbulence induced atomization, the following input data is needed:

- Particle mass flow rate, $\dot{m}$

  Injection pressure difference, Δp
- Normal distance of the density probe from the injection center
- Nozzle length/diameter ratio - This is required for the calculation of the average turbulence kinetic energy and dissipation rates.
- Nozzle discharge coefficient - This is the ratio of mass flow through nozzle to theoretically possible mass flow and can be specified optionally.

6.5.1.5. Usage and Restrictions for Primary Breakup Models

Primary breakup models are used to determine starting conditions for the droplets that leave the injection nozzle. To accomplish this task, the models use averaged nozzle outlet quantities, such as liquid velocity, turbulent kinetic energy, location and distribution of liquid and gas zones, as input for the calculation of initial droplet diameters, breakup times and droplet injection velocities. Because all detailed information is replaced by quasi 1D data, the following restrictions will apply:

- Primary breakup model support will only be available for particle injection regions and not for boundary condition.
- On particle injection regions, primary breakup will only be available for cone type injection, and only if the nozzle cross sectional area is larger than 0.

6.5.2. Secondary Breakup Models

The following sections give an overview about the currently available secondary breakup models in CFX. Further information about the breakup models can be found in the provided references.

6.5.2.1. Breakup Regimes

The breakup of a liquid jet into droplets is caused by a combination of different mechanisms: turbulence within the liquid phase, implosion of cavitation bubbles and external aerodynamic forces acting on the liquid jet. Depending on the injection parameters such as the relative velocity between liquid and gas, the liquid and gas densities and the liquid viscosity and surface tension the contribution of each of the above mechanisms to the spray breakup varies.

Breakup regimes are typically classified in terms of the dimensionless numbers: Weber Number (We) and Ohnesorge number (Oh), as given by:

$$\mathrm{We} = \frac{\rho_F\, V_{\mathrm{slip}}^2\, d_P}{\sigma} \tag{6–127}$$

$$\mathrm{Oh} = \frac{\mu}{\sqrt{\rho_P\, \sigma\, d_P}} \tag{6–128}$$

where the subscript P refers to the droplet (particle) and the subscript F refers to the surrounding fluid. Sometimes, the Weber number is defined using the particle radius, r_P.

If a droplet is exposed to a gas flow, significant deformation starts at a Weber number of unity. Above a certain value of the Weber number, the droplet deformation leads to breakup. Typically, the following breakup regimes are observed [117]:

- Vibrational breakup: $\mathrm{We} < 12$
- Bag breakup: $12 < \mathrm{We} < 50$
- Bag-and-stamen breakup: $50 < \mathrm{We} < 100$
- Sheet stripping: $100 < \mathrm{We} < 350$
- Catastrophic breakup: $350 < \mathrm{We}$

6.5.2.2. Numerical Approach to Breakup Modeling

For the numerical simulation of droplet breakup, a so-called statistical breakup approached is used in CFX. In this framework, it is assumed that if a droplet breaks up into child droplets, the particle diameter is decreased accordingly to the predictions of the used breakup model. The particle number rate is adjusted so that the total particle mass remains constant (mass of parent droplet = Σ mass of child droplets). Using this assumption, it is not required to generate and track new droplets after breakup, but to continue to track a single representative particle.

6.5.2.3. Reitz and Diwakar Breakup Model

This model ([115]) distinguishes between two breakup regimes: bag breakup and stripping breakup. Breakup occurs if a critical particle Weber number has been exceeded. Independent of the breakup regime, it is assumed that during breakup the following relation describes the reduction of the particle radius:

$$\frac{dr_P}{dt} = \frac{-\,(r_P - r_{\mathrm{stable}})}{t_{br}} \tag{6–129}$$

r_p is the droplet radius prior to breakup, r_{stable} is the new radius for the stable droplet and t_{br} is the characteristic breakup time. Values for r_{stable} and t are calculated from the equations given in the following section:

6.5.2.3.1. Bag Breakup

$(\mathrm{We} > \mathrm{We}_{crit})$:

$$t_{br} = C_1 \sqrt{\frac{\rho_P\, r_P^3}{2\,\sigma}} \tag{6–130}$$

and

$$r_{\mathrm{stable}} = \frac{6\,\sigma}{\rho_F\, V_{\mathrm{slip}}^2} \tag{6–131}$$

6.5.2.3.2. Stripping Breakup

$(\mathrm{We}/\sqrt{Re} > C_{s1})$:

$$t_{br} = C_2 \frac{r}{V_{\mathrm{slip}}} \sqrt{\frac{\rho_P}{\rho_F}} \tag{6–132}$$

and

$$r_{\mathrm{stable}} = \frac{\sigma^2}{2\,\rho_F^2\, V_{\mathrm{slip}}^3\, v} \tag{6–133}$$

The model constants C_1, C_2, We_{crit} and C_{s1} of the Reitz & Diwakar breakup model are accessible via CCL. The standard values of the constants are given in *Table 6.1: Reitz and Diwakar breakup model constants and their default values* (p. 276).

Table 6.1 Reitz and Diwakar breakup model constants and their default values

Constant	Value	CCL Name
C_1	π	Time Factor for Bag Breakup
C_2	20	Time Factor for Stripping
We_{crit}	6.0	Critical Weber Number for Bag
C_{s1}	0.5	Weber Number Factor for Stripping

6.5.2.4. Schmehl Breakup Model [112]

In the Schmehl model, the droplet deformation and breakup times are based on experimental findings of Hsiang et al. [109] and Pilch et al. [117]. It can be shown that irrespective of the breakup regime, the time to deform a particle from a sphere into a disk shape is approximately constant, and is given by:

$$t_i = 1.6 \ * \ t^* \tag{6–134}$$

with the characteristic time t^* given as:

$$t^* = \frac{d_P}{V_{\text{slip}}} \sqrt{\frac{\rho_P}{\rho_F}} \tag{6–135}$$

The second phase of breakup, which is characterized by further distortion of the droplet to its final destruction, is modeled by the following correlations:

$$\frac{t_{br}}{t^*} = \begin{cases} 6\,(\mathrm{We} - 12)^{-0.25} & 12 \le \mathrm{We} < 18 \\ 2.45\,(\mathrm{We} - 12)^{0.25} & 18 \le \mathrm{We} < 45 \\ 14.1\,(\mathrm{We} - 12)^{-0.25} & 45 \le \mathrm{We} < 351 \\ 0.766\,(\mathrm{We} - 12)^{0.25} & 351 \le \mathrm{We} < 2670 \\ 5.5 & 2670 \le \mathrm{We} \end{cases} \tag{6–136}$$

Note that the last two breakup regimes are an extension to the originally proposed model by Schmehl and are based on experimental findings given by [117].

For large Ohnesorge numbers $(\mathrm{Oh} > 1)$, the following correlation is used:

$$\frac{t_{br}}{t^*} = 4.5 \left(1 \ + \ 1.2 \ \mathrm{Oh}^{0.74}\right) \tag{6–137}$$

6.5.2.4.1. Breakup Process for Schmehl Breakup Model

The breakup process used by the Schmehl breakup model can be summarized as follows:

- Breakup can only occur if the droplet Weber number, We, is larger than the critical Weber number, We_{crit}, where:

$$\mathrm{We}_{crit} = 12\,(1 + 1.077\mathrm{Oh}^{1.6}) \tag{6–138}$$

- Depending on the Weber number regime at the beginning of the breakup process, the following droplet breakup scenarios are possible:
  - Bag Breakup Regime - This is characterized by:

    $12\,(1 + 1.077\mathrm{Oh}^{1.6}) < \mathrm{We} < 20\,(1 + 1.2\mathrm{Oh}^{1.5})$

    → In this regime, the breakup occurs in the second half of the t_i to t_{br} time frame.

    A random breakup time is computed.

Child droplet sizes and normal velocity of children after breakup are computed.

If the particle life time exceeds the breakup time, then breakup occurs.

- Multimode Regime - This is characterized by $20\,(1+1.2\mathrm{Oh}^{1.5}) < \mathrm{We} < 32\,(1+1.5\mathrm{Oh}^{1.4})$.
  - → In this regime, the breakup occurs in the second half of the t_i to t_{br} time frame.

    A random breakup time is computed.

    Child droplet sizes and normal velocity of children after breakup are computed.

    If the particle life time exceeds the breakup time, then breakup occurs.
- Shear Breakup Regime - This is characterized by $32\,(1+1.5\mathrm{Oh}^{1.4}) < \mathrm{We}$.
  - → In this regime, the breakup can occur any time between t_i and t_{br} time frame.

    A random breakup time is computed.

    Child droplet sizes and normal velocity of children after breakup are computed.

    If the particle life time exceeds the breakup time, then breakup occurs.

Droplet size after breakup for the Bag Breakup and Multimode regimes is computed from the following relation:

$$d_{32} = d_p \left(1.5\mathrm{Oh}^{0.2}\mathrm{We}_{corr}^{-0.25}\right) \tag{6–139}$$

with:

$$\mathrm{We}_{corr} = \frac{\mathrm{We}}{\left(1+1.077\mathrm{Oh}^{1.6}\right)} \tag{6–140}$$

In the Shear Breakup regime, the child droplet size, $d_{32,red}$, is computed as:

$$d_{32,red} = \frac{4\,d_{32}\,d_c}{5d_c - d_{32}} \tag{6–141}$$

where d_c is the maximum stable diameter computed from the critical Weber number, We_{crit},

with:

$$d_c = \frac{\mathrm{We}_{crit}\,\sigma}{\rho_F V_{\mathrm{slip}}^2} \tag{6–142}$$

Except for the Shear Breakup regime, all child droplets inherit their parent's velocity and an additional velocity component, V_N, which is given by:

$$|V_N| = \frac{d_{p,\max} - d_0}{2\,(t_{br} - t_i)} \qquad \textbf{(6–143)}$$

with $d_{p,\max} \sim 6 d_0$ at $t = t_{br}$, and d_0 being the droplet diameter before breakup.

The velocity component V_N is assumed to be in a plane normal to the parent droplet velocity direction. Its circumferential orientation within this plane cannot be specified and is chosen randomly. The total child droplet velocity is then determined from:

$$\vec{V}_{P,\mathrm{new}} = \vec{V}_{P,\mathrm{old}} + \vec{V}_N \qquad \textbf{(6–144)}$$

6.5.2.5. Taylor Analogy Breakup (TAB) Model [116]

O'Rourke and Amsden proposed the so-called TAB model that is based on the Taylor analogy. Within the Taylor analogy, it is assumed that the droplet distortion can be described as a one-dimensional, forced, damped, harmonic oscillation similar to the one of a spring-mass system. In the TAB model, the droplet deformation is expressed by the dimensionless deformation $y = 2\,(x/r)$, where x describes the deviation of the droplet equator from its underformed position (see *Figure 6.4* (p. 281)). Assuming that the droplet viscosity acts as a damping force and the surface tension as a restoring force, it is possible to write the equation of deformation motion as:

$$\ddot{y} = \frac{5\,\mu_p}{\rho_P\, r^2}\,\dot{y} + \frac{8\,\sigma}{\rho_P\, r^3}\,y = \frac{2\,\rho_g\, V_{\mathrm{slip}}^2}{3\,\rho_g\, r^2} \qquad \textbf{(6–145)}$$

Integration of this equation leads to the following time-dependent particle distortion equation:

$$y\,(t) = \mathrm{We_C} + e^{-t/t_\mathrm{D}} \left[\left(y_0 - \mathrm{We_C}\right) \cos\omega t + \left(\frac{y_0}{\omega} + \frac{y_0 - \mathrm{We_C}}{\omega t_\mathrm{D}} \right) \sin\omega t \right] \qquad \textbf{(6–146)}$$

with:

$$t_{\mathrm{D}} = \frac{2\,\rho_P\,r^2}{C_{\mathrm{d}}\,\mu_P} \tag{6–147}$$

$$\omega^2 = \frac{C_{\mathrm{k}}\,\sigma}{\rho_P\,r^3} - \frac{1}{t_d^2} \tag{6–148}$$

$$\mathrm{We_C} = \mathrm{We}\,\frac{C_{\mathrm{f}}}{C_{\mathrm{k}}\,C_{\mathrm{b}}} \tag{6–149}$$

y_0 and $\dot{y}_0$ are the initial values of distortion and distortion rate of change. For the TAB model, y_0 and $\dot{y}_0$ are typically taken as zero.

During particle tracking, *Equation 6–107* (p. 268) is solved for the dimensionless particle distortion. Breakup only occurs if the particle distortion *y* exceeds unity, which means that the deviation of the particle equator from its equilibrium position has become larger than half the droplet radius.

The Sauter mean radius of the child droplets after breakup is calculated from the following expression:

$$\frac{r_{\mathrm{P,Parent}}}{r_{\mathrm{P,Child}}} = \left[\,1 + 0.4\,K + \frac{\rho_{\mathrm{P}}\,r^3_{\mathrm{P,Parent}}}{\sigma}\,\dot{y}_0^2\left(\frac{6\,K - 5}{120}\right)\right] \tag{6–150}$$

that is based on the conservation of surface energy and energy bound in the distortion and oscillation of the parent droplet and surface energy and kinetic energy of the child droplets.

The TAB model has been used to determine the normal velocity of the child droplets after breakup. At the time of breakup, the equator of the parent droplet moves at a velocity of $V_{\mathrm{N}} = C_{\mathrm{v}}\,C_{\mathrm{b}}\,r\,\dot{y}$ in a direction normal to the parent droplet path. This velocity is taken as the normal velocity component of the child droplets and the spray angle θ can be determined from:

$$\tan\frac{\theta}{2} = \frac{V_{\mathrm{N}}}{V_{\mathrm{slip}}} \tag{6–151}$$

After breakup of the parent droplet, the deformation parameters of the child droplet are set to $y(0) = \dot{y}(0) = 0$.

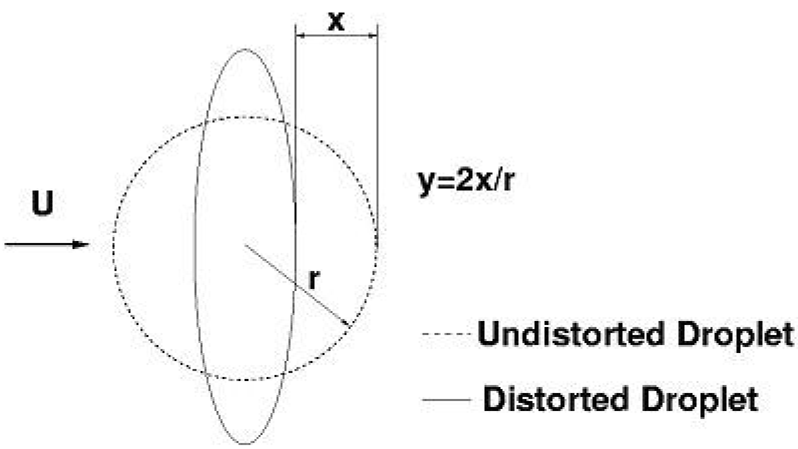
The following user accessible model constants are available for the TAB breakup model.

Table 6.2 TAB Breakup Model Constants and their Default Values

Constant	Value	CCL Name
C_b	0.5	Critical Amplitude Coefficient

C_d	5.0	Damping Coefficient
C_f	1/3	External Force Coefficient
C_k	8.0	Restoring Force Coefficient
C_v	1.0	New Droplet Velocity Factor
K	10/3	Energy Ratio Factor

Figure 6.4 Particle distortion for the TAB model



6.5.2.6. ETAB [106]

The enhanced TAB model uses the same droplet deformation mechanism as the standard TAB model, but uses a different relation for the description of the breakup process. It is assumed that the rate of child droplet generation, $dn(t)/(dt)$, is proportional to the number of child droplets:

$$\frac{d}{dt}n(t) = 3\,K_{br}\,n(t) \tag{6–152}$$

The constant K_{br}, depends on the breakup regime and is given as:

$$K_{br} = \begin{cases} k_1\,\omega & \mathrm{We} \le \mathrm{We}_t \\ k_2\,\omega\,\sqrt{\mathrm{We}} & \mathrm{We} > \mathrm{We}_t \end{cases} \tag{6–153}$$

with We_t being the Weber number that divides the bag breakup regime from the stripping breakup regime. We_t is set to default value of 80. Assuming a uniform droplet size distribution, the following ratio of child to parent droplet radii can be derived:

$$\frac{r_{P,\mathrm{Child}}}{r_{P,\mathrm{Parent}}} = e^{-K_{br} t} \tag{6–154}$$

After breakup of the parent droplet, the deformation parameters of the child droplet are set to $y(0) = \dot{y}(0) = 0$. The child droplets inherit a velocity component normal to the path of the parent droplet with a value:

$$V_N = A\ \dot{x} \tag{6–155}$$

where A is a constant that is determined from an energy balance consideration:

$$A^2 = 3\left[1 - \frac{r_{P,\mathrm{Parent}}}{r_{P,\mathrm{Child}}} + 5\ C_D\ \mathrm{We}/72\right]\frac{\omega^2}{\dot{y}^2} \tag{6–156}$$

with:

$$\omega^2 = \frac{C_\mathrm{k}\ \sigma}{\rho_P\ r_{P,\mathrm{Parent}}} \tag{6–157}$$

and C_D being the parent droplet drag coefficient at breakup.

It has been observed that the TAB model often predicts a ratio of child to parent droplet that is too small. This is mainly caused by the assumption, that the initial deformation parameters $y(0)$ and $\dot{y}(0)$ are zero upon injection, which may lead to far too short breakup times. The largely underestimated breakup times in turn lead to an underprediction of global spray parameters such as the penetration depth, as well as of local parameters such as the cross-sectional droplet size distribution. To overcome this limitation Tanner [107] proposed to set the initial value of the rate of droplet deformation, $\dot{y}(0)$, to the largest negative root of *Equation 6–107* (p. 268):

$$\dot{y}(0) = \left[1 - \mathrm{We_C}\left(1 - \cos\omega t_{bu}\right)\right]\frac{\omega}{\sin\omega t_{bu}} \tag{6–158}$$

while keeping the initial value of the droplet deformation, $y(0) = 0$. t_{bu} is determined from the following equation:

$$t_{bu} = C\sqrt{\frac{\rho_P}{\rho_F}}\ \frac{d_{P,0}}{V_{P,0}} \tag{6–159}$$

with C = 5.5

The effect of setting $\dot{y}(0)$ to a negative number is to delay the first breakup of the large initial droplets and to extend their life span, which results in a more accurate simulation of the jet breakup.

In addition to the TAB model constants, the following user accessible model constants are available for the ETAB breakup model:

Table 6.3 ETAB Breakup Model Constants and Their Default Values

Constant	Value	CCL Name
K_1	2/9	ETAB Bag Breakup Factor
K_2	2/9	Stripping Breakup Factor
We_t	80	Critical Weber Number for Stripping Breakup

6.5.2.7. Cascade Atomization and Breakup Model (CAB)

A further development of the ETAB model, is the so-called Cascade Atomization and Breakup Model (CAB). Identical to the ETAB model, the following equation is used to determine the child droplet size after breakup:

$$\frac{r_{P,\mathrm{Child}}}{r_{P,\mathrm{Parent}}} = e^{-K_{br} t} \tag{6–160}$$

the main difference being the definition of the breakup constant K_{br}:

$$K_{br} = \begin{cases} k_1\,\omega & 5 < \mathrm{We} < 80 \\ k_2\,\omega\,\sqrt{\mathrm{We}} & 80 < \mathrm{We} < 350 \\ k_3\,\omega\,\mathrm{We}^{3/4} & 350 < \mathrm{We} \end{cases} \tag{6–161}$$

where

$$k_2 = k_1 \frac{\sqrt{1 - \frac{1}{2}\left(\frac{C_k C_b}{C_f We_t}\right)}}{\mathrm{acos}\left(1 - \frac{C_k C_b}{C_f We_t}\right)} \tag{6–162}$$

and

$$k_3 = k_2 / We_{t2}^{\ 1/4} \tag{6–163}$$

The model constants C_b, C_f and C_k have the same values as given for the TAB model. For details see *Table 6.2: TAB Breakup Model Constants and their Default Values* (p. 280).

In addition to the TAB model constants, the following user accessible model constants are available for the CAB breakup model:

Table 6.4 CAB Breakup Model Constants and Their Default Values

Constant	Value	CCL Name
k_1	0.05	CAB Bag Breakup Factor
We_t	80	Critical Weber Number for Stripping Breakup
We_{t2}	350	Critical Weber Number for Catastrophic Breakup

6.5.3. Dynamic Drag Models

Many particle drag models assume that the droplet remains spherical throughout the domain. However, this is not always the case and the particle shape may be distorted significantly. In the extreme case, the particle shape will approach that of a disk. The drag coefficient is highly dependent on the particle shape and it is therefore desirable to modify the standard drag laws to account for the effects of droplet distortion.

In CFX, the following models are implemented that modify the drag coefficient depending on the particle distortion:

6.5.3.1. Liu [108]

The drag coefficient is assumed to vary linearly between that of a sphere and that of a disk:

$$C_{d,\mathrm{droplet}} = C_{d,\mathrm{sphere}}\,(1 + 2.632\,y) \quad \textbf{(6–164)}$$

$$0 < y < 1 \quad \textbf{(6–165)}$$

y is a measure of the particle distortion. If the droplet is not distorted $y = 0$, then the drag coefficient of a sphere will be obtained. If the particle is maximally distorted $y = 1$, then the drag coefficient of a disk will be obtained. This drag model is only available for the TAB, ETAB and CAB breakup models.

Note

The Liu drag coefficient modification is activated by default for the TAB, ETAB, and CAB breakup models.

6.5.3.2. Schmehl [105]

The droplet deformation due to external aerodynamic forces leads to a change in the drag coefficient of the droplet that is assumed to vary between the two limiting geometries, a sphere and a disk. The drag coefficient of the droplet is given as:

$$C_{d,\mathrm{droplet}} = f\ C_{d,disc} + (1 - f)\ C_{d,sphere} \quad \textbf{(6–166)}$$

with:

$$C_{d,\mathrm{sphere}} = 0.36 + 5.48\ \mathrm{Re}^{-0.573} + \frac{24}{\mathrm{Re}} \quad \textbf{(6–167)}$$

$$C_{d,\mathrm{disc}} = 1.1 + \frac{64}{\pi\ \mathrm{Re}} \quad \textbf{(6–168)}$$

and:

$$f = 1 - E^2 \quad \textbf{(6–169)}$$

The droplet aspect ratio, E, and the particle distortion, y, are related as follows:

$$E = 1/y^3 \quad \textbf{(6–170)}$$

here E is 1 for a sphere and 0 for the distorted particle (disc).

This drag model is available for the Schmehl breakup model only. It is activated by default.

6.6. Particle Collision Model

The following topics will be discussed:

6.6.1. Introduction to the Particle Collision Model

Highly loaded gas-particle flows are commonly simulated by the two-fluid model where interactions between particles are computed using the Kinetic Theory of Dense Gases. In classical Euler-Lagrange modeling, the equations of motion of individual particles are solved without considering collisions between particles because the presence of other particles is not taken into account.

The *particle-particle collision model* (PPCM) in ANSYS CFX takes inter-particle collisions and their effects on the particle and gas phase into consideration. The model implemented into ANSYS CFX is the stochastic particle-particle collision model by Oesterlé & Petitjean [151] that has been extended by Frank, Th. [149], Hussmann, B. et al. [150] and Sommerfeld [152].

6.6.1.1. Background Information

Standard Lagrange simulations are constrained to dilute gas-particle flows because particles are treated as being independent of their neighbors. Activating the particle collision model facilitates the application of the Lagrange model to dense gas-solid flows with a high mass-loading while the particle volume fraction is still low. Hence, dense multiphase flows in which contact forces between particles preponderate over aerodynamic forces exerted by the fluid, such as in fluidized beds or hoppers, are excluded. This is because the model is limited to binary collisions which dominate, if the average distance between two particles is much greater than their diameters.

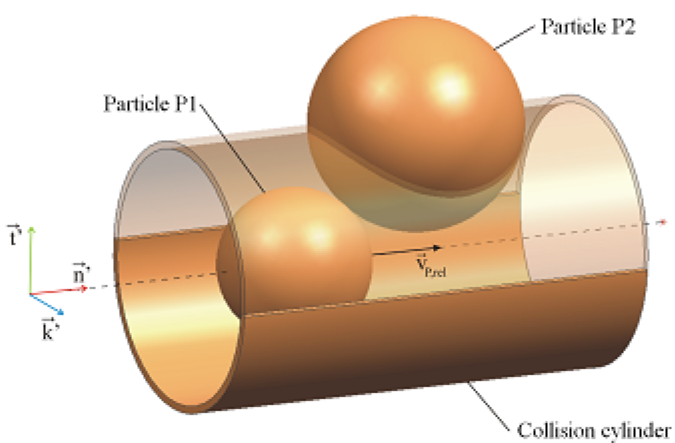
Both the Euler-Euler multiphase model and the "classical" Euler-Lagrange model can approximate dense gas-solid flows only roughly. The *LPTM-PPCM* (p. 285) model implemented in ANSYS CFX expands the Euler-Lagrange model by a stochastic inter-particle collision model by allowing the so-called *four-way coupling*. Hereby, the mutual influence of gas and particles is accounted for as well as the mutual interaction of (spherical) particles by means of binary collisions.

6.6.2. Implementation of a Stochastic Particle-Particle Collision Model in ANSYS CFX

The main idea behind the stochastic collision model is the creation of a *virtual collision partner*, which is done by using local size and velocity distributions of the droplet phase. Hence, the virtual droplet is a representative of the local droplet population. This approach avoids the need to know the locations of neighboring droplets and the time consuming search for collision partners in order to decide whether a collision takes place or not. Only the droplet size- and velocity-distribution functions must be stored for each computational element, see Sommerfeld, M. (1996) [152]. The following illustration shows a virtual collision partner, Particle P2, whose position is determined using the following considerations:

- Stochastic determination
- Probability equally distributed over cross section

Figure 6.5 Position of the Virtual Collision Partner



6.6.2.1. Implementation Theory

For the calculation of the instantaneous velocity of the virtual collision partner, a partial correlation of the turbulent fluctuation velocities between the real and the virtual particle is taken into account, as proposed by Sommerfeld, M. (2001) [154]. The correlation is a function of the turbulent Stokes number St_t, which is the ratio of the aerodynamic relaxation time and a characteristic eddy lifetime, the latter provided by the turbulence model. Small particles being able to follow the gas flow easily have Stokes numbers below unity; the Stokes numbers of large inertial particles exceed unity.

Sommerfeld's correlation function,

$$R(\mathrm{St}_t) = \exp(-0.55\mathrm{St}_t^{0.4}) \tag{6–171}$$

which was adapted to LES-data of a homogeneous isotropic turbulence field by Lavieville, J., E. Deutsch, and O. Simonin (1995) [153], is used to determine the fluctuating velocity of the virtual collision partner as given below:

$$v'_{2,i} = R(\mathrm{St}_t)\, v'_{1,i} + \sigma_{P,i}\sqrt{1 - R(\mathrm{St}_t)^2}\,\xi \tag{6–172}$$

where the index 1 stands for the real particle and index 2 for the virtual particle, index i represents the three coordinate directions and the prime indicates a fluctuating part of the velocity. $\sigma_{P,i}$ is the i-th component of the mean fluctuation velocity in the control volume. ξ is a Gaussian random number with zero mean and a standard deviation of unity. It represents the uncorrelated part of the fluctuation velocity of the virtual particle. Its instantaneous velocity is the sum of the fluctuating part described above and the local mean value. The collision frequency is then determined in analogy to the Kinetic Theory of Gases [151], by the following equation:

$$f_c = \frac{\pi}{4}(d_{P1} + d_{P2})^2 |\vec{v}_1 - \vec{v}_2| n_P \tag{6–173}$$

where $\vec{v}_1$and $\vec{v}_2$ represent the instantaneous velocities of the real particle 1 and its collision partner 2. The diameter of the latter is sampled from a Gaussian distribution around the local average value. The collision probability is a simple function of the collision frequency and the Lagrangian time step and is calculated as follows:

$$P_c = 1 - \exp(-f_c \Delta t) \tag{6–174}$$

The time step can be altered in the collision subroutine to ensure accuracy and stability of the calculation by limiting it to $\Delta t < 0.05/f_c$. This allows for at most one binary collision per time step, as derived by Sommerfeld, M. [152].

A uniformly distributed random number, Ψ, is then generated and compared to the collision probability P_c. If $P_c > \Psi$, the inter-particle collision is calculated deterministically. For $P_c < \Psi$, no collision occurs and the velocity components of the real particle remain unchanged. In case of a collision, the location of the virtual particle is determined in a stochastic way. The calculation of the position of the collision partner relative to the real particle is done in a local coordinate system. The position is sampled randomly from a uniform distribution on the collision cylinder cross section and a distance of the center point according to the sum of the two particle radii. Subsequently, the position of the virtual particle is transformed back to the global system. A more detailed description is given by Frank, Th. (2002) [149].

At this stage, information on location, size and velocity of the virtual collision partner is known. The next step is to determine the change in the velocity components caused by the collision. To identify the post-collision velocities, the momentum transferred between the particles has to be determined. For this purpose, it is again suitable to use a local coordinate system, different to the one mentioned above and fixed to the real particle. To identify the post-collision velocities, the momentum $\vec{J}$ transferred between the particles has to be determined. Here a distinction is made between a sliding and a non-sliding collision, if particle rotation is accounted for which affects the tangential components of $\vec{J}$.

During a non-sliding collision, the relative movement at the point of contact ceases; whereas during a sliding collision, relative motion of the contact surfaces is maintained under the influence of sliding friction.

6.6.2.1.1. Particle Collision Coefficients Used for Particle-Particle Collision Model

Besides the coefficient of restitution e considering the losses normal to the plane of contact, the coefficients of sliding and static friction have to be supplied for the particle material, if particle rotation is taken into account. Hence, in the case of rotating particles, a decision between the two collision modes is made based on the coefficient of static friction. The respective components of the transferred momentum are determined and the post-collision velocity components are calculated in the local coordinate system. Finally these values are transformed back to the global coordinate system and passed to the Lagrangian solver. Because the computational particles (parcel) represent a number of real particles, it is assumed that all the real particles inside the parcel collide with the same number of virtual particles.

The parameters used for the particle collision model are outlined below:

Sommerfeld Collision Model

- `Coefficient of Restitution`: Enter a numerical quantity or CEL based expression to specify the value of coefficient of restitution for inter-particle collisions. A value of '1.0' means a fully elastic collision, while a value of '0.0' would result in an inelastic collision.
- `Static Friction Coefficient` and `Kinetic Friction Coefficient`: Enter a numerical quantity or CEL based expression to specify values of coefficients of friction for inter-particle collisions.

  See Implementation Theory in the *CFX-Solver Theory Guide* for more information on setting up `Coefficient of Restitution`, `Static Friction Coefficient`, and `Kinetic Friction Coefficient`.

User Defined

This option is available only if you have created a particle user routine to set up the model. Specify the name of **Particle User Routine** and select input arguments and type of particle variables returned to the user routine from the **Arguments** and **Variable List** drop-down list, respectively. See Particle User Routines in the *CFX-Pre User's Guide* for information on setting up a particle user routine.

The friction coefficient values are dependent on particle material and should be obtained from experimental investigations. As another example, a suitable (static and kinetic) friction coefficient for the collision of steel particles can be assumed to be equal to 0.15, while a value of about 0.4 is mentioned in literature for glass particles. Furthermore, these values are dependent on the surface roughness of the particle material and the degree of sphericity of the particle material in the flow and are therefore subject to uncertainty.

6.6.2.1.2. Particle Variables Used for Particle-Particle Collision Model

The calculation of particle collisions with the `Sommerfeld collision model` uses instantaneous and averaged fluid and particle quantities as well as the following additional quantities:

- Particle number density
- Turbulent Stokes number
- Standard deviation of particle diameter and particle velocity
- Size of the integration time step of the Lagrange particle solver (User Fortran only)

6.6.2.1.2.1. Particle Number Density

The particle number density, n_P, describes the number of particles per unit volume and is calculated as follows:

$$n_P = \frac{\sum \dot{N} \Delta t}{Vol_{CV}} \quad \textbf{(6–175)}$$

In this expression, the sum is taken over all particles and all time-steps taken in the control volume of each vertex. Here Δt is the particle integration time-step and $\dot{N}$ is the number rate for the particle. Vol_{CV} is the volume of the control volume associated with the vertex.

6.6.2.1.2.2. Turbulent Stokes Number

The turbulent Stoke number, St_t, is used for the calculation of the fluctuating velocity of the collision partner of a droplet. The turbulent Stokes number is defined as the ratio of the aerodynamic relaxation time, τ_R, and a characteristic eddy lifetime, t_E.

$$\mathrm{St}_t = \frac{\tau_R}{t_E} \quad \textbf{(6–176)}$$

With

$$\tau_R = \frac{\rho_P \, d_P^2}{18 \mu_F} \quad \textbf{(6–177)}$$

and

$$t_E = 0.3 \frac{k}{\varepsilon} \quad \textbf{(6–178)}$$

6.6.2.1.2.3. Standard Deviation of Particle Quantities

For the calculation of the collision partner, instantaneous, mean and the standard deviation of particle quantities are required. The standard deviations of the following variables are calculated and are available for solver internal use, as well as for Particle User Fortran and post-processing.

- Particle Velocity
- Particle Temperature
- Particle Diameter
- Particle Number Rate

The standard deviation of a particle variable, ϕ, is calculated within the "vertex variable" machinery and uses the following definition:

$$\phi_{rms} = \sqrt{\overline{\phi^2} - \overline{\phi}^2} \quad \textbf{(6–179)}$$

For information on vertex and RMS particle variables, see Particle Field Variables in the *CFX Reference Guide*.

6.6.2.1.2.4. Integration Time Step Size (User Fortran Only)

The particle integration time-step is passed to the collision model routine that is provided by the user. The time-step size is required to calculate the collision frequency for that particular integration.

Note

The particle integration time-step can be accessed by the user but it can not be changed. The only exception is made for the particle collision model when using Particle User Fortran. In this case, the solver allows users to overwrite the `Particle Integration Timestep` variable. This extension is required, as the integration time-step computed from the particle tracker is typically 2 to 4 orders of magnitude larger than the one computed (and usually needed) by the collision model routine. This ensures an accurate calculation of the droplet collision in regions with high particle concentration. For details, see *Implementation of a Stochastic Particle-Particle Collision Model in ANSYS CFX* (p. 286).

Name	Meaning	Dimension
Particle Integration Timestep	Size of the current integration time-step	[Time]
Particle Turbulent Stokes Number	Turbulent Particle Stokes number	[ ]
Particle Number Density	Particle number density	[1/Length^3]
RMS Particle Number Rate	Standard deviation of the particle number rate	[1/Time]
RMS Velocity	Standard deviation of particle velocity	[Velocity]
RMS Mean Particle Diameter	Standard deviation of particle diameter	[Length]

6.6.3. Range of Applicability of Particle-Particle Collision Model

The aim of the *LPTM-PPCM* (p. 285) is to raise one of the restrictions for a Lagrangian particle tracking model, namely to extend the range of applicability to higher concentrations of the dispersed particulate phase. A fluid particle flow can either be characterized by the ratio of the mean time between particle-particle collisions and the particle relaxation time:

$$\tau_C \Big/ \tau_P = \frac{1}{n\pi d_P^2 v_r} \Bigg/ \frac{\rho_P d_P^2}{18\mu_C} \quad \textbf{(6–180)}$$

or by the inter-particle spacing defined as the ratio of the distance between two particles in the flow and the particle diameter:

$$\frac{L}{d_P} = \sqrt[3]{\frac{\pi}{6\alpha_P}} \quad \textbf{(6–181)}$$

A fluid-particle flow is particle-particle collision dominated, if the ratio $\tau_C / \tau_P < 1$, which means that the time between adjacent particle-particle collisions is too small in order to allow the particle to get accelerated by aerodynamic forces to its normal slip velocity with the carrier fluid until the next particle-particle collision occurs. This fairly well corresponds to an inter-particle spacing of less than 10 and the fluid-particle flow is called a dense flow.

In the case of dense fluid-particle flow, it is necessary to additionally account for the direct particle-to-particle momentum transfer. The PPCM model, in accordance with Oesterlé & Petitjean and Sommerfeld, accounts for this inter-particle momentum transfer by making the assumption that only binary particle collisions occur in the intermediate regime between dilute fluid-particle flows and flow regimes in packed and fluidized beds. For the latter conditions it is not sufficient to assume binary particle collisions and therefore the PPCM model is not applicable to flow regimes, where multiple particles collide at the same time or stay in direct frictional contact.

Also see Requirements for the Applicability of Particle-Particle Collision Model in the *CFX-Solver Modeling Guide*.

6.6.4. Limitations of Particle-Particle Collision Model in ANSYS CFX

General model limitations and underlying assumptions are mentioned in *Range of Applicability of Particle-Particle Collision Model* (p. 290) and in [149], [150] and [152]. When using the particle-particle collision model in ANSYS CFX, the following limitations apply:

- Particle collisions only take place between particles of the same type.
- Particle collisions are currently only supported for fully coupled particles.
- Collisions are taken into account in transient simulations only after the second time step, because the PPCM relies on volume-averaged particle properties, which are available only after the completed first time step. Thus, averaged particle properties used in the PPCM for the generation of the virtual collision partner in binary particle collisions lag behind by one time step. With $\Delta t \rightarrow 0$, the model converges to the correct particle property representation for the virtual collision partner in the PPCM, but for large integration time steps the particle phase information seen by the tracked particle in the PPCM might be inaccurate.

Also see Requirements for the Applicability of Particle-Particle Collision Model in the *CFX-Solver Modeling Guide*.

6.7. Particle-Wall Interaction

The following topics will be discussed:

6.7.1. Introduction to Particle-Wall Interaction

Particle-wall interaction involves complex physics and not all aspects are well understood. Dimensional analysis shows that droplet-wall interaction depends on the wall temperature, wall material and roughness, impact angle and impact velocity, the existence of a wall film, and various other parameters.

ANSYS CFX uses advanced particle-wall interaction and quasi-static wall film models to address some of the shortcomings found in ANSYS CFX 11.0 and previous releases (see the background information). The following advanced particle-wall interaction models are available in ANSYS CFX:

- Elsaesser particle-wall interaction model - This model accounts for all of the above listed influencing factors during the particle droplet reflection.
- Stick-to-wall model - This model enforces all particles that hit a wall to become part of the wall film.

For details on the implementation of quasi-static wall film in ANSYS CFX, see *Quasi Static Wall Film Model* (p. 298).

6.7.1.1. Background Information

In ANSYS CFX 11.0 and all previous releases, a very simplistic approach is used to describe the process of particles colliding with walls: It is assumed that during the collision process, particles exchange momentum only with the wall. During such collisions, the momentum loss is controlled by constant or time-dependent coefficients of restitution.

In the case of a fully inelastic collision, the particle is *collected* at a wall; however, the corresponding particle mass or energy is no longer available for the simulation. It is also not possible to explicitly account for the effects of wall roughness or wall temperature during the particle-wall interaction.

6.7.2. The Elsaesser Particle-Wall Interaction Model

The Elsaesser model is used to describe the outcome of a drop impact on a wall. The model distinguishes between three impact regimes, which are characterized by their wall temperature and the existence of a wall film.

This model is implemented in ANSYS CFX as per Elsässer [175], however, with one exception that the double droplet reflection used in the original formulation of Elsässer is neglected.[1]

This section uses the following notation for the remaining part of the model description:

- Incident parcel is designated with index “0”; whereas, “1” and consecutive integers are used for the reflected parcels.
- Angles are always defined with respect to the wall tangent.
- The wall can be dry or wetted with the restriction that the wall film fluid is identical to the particle fluid.

6.7.2.1. Classification of Impact Regimes

The impact regime classification is based on the work of Bai and Gosman [176], who defined two limiting temperature to distinguish between the following three temperature regimes:

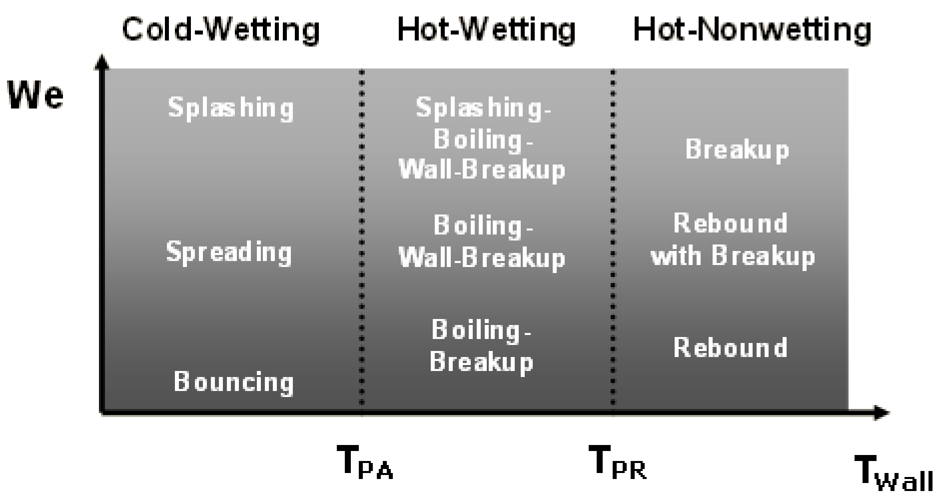
- Cold wall with wall film (cold-wetting)

[1]The governing equations used in the formulation of Elsaesser particle-wall interaction model in ANSYS CFX differ in some coefficient values from those presented in Elsässer [175].

- Hot wall with wall film (hot-wetting)
- Hot wall without wall film (hot-nonwetting)

The limiting temperatures are called *pure adhesion* temperature, T_{PA}, and *pure rebound* temperature, T_{PR}. The pure adhesion and the pure rebound temperatures are functions of the wall and particle material combination. Further dependencies (such as particle velocity, surface roughness, drop size and initial temperature) probably play an important role for dynamic impact but are neglected (usually done in most models), as the influences are not well known. Only the material properties and the ambient pressure are thus considered. The correlations used by Elsässer [175] to compute rebound and adhesion temperatures are strictly valid only for a typical internal combustion engine application, which involve the interaction of gasoline droplet on a wall material composed of aluminum.

Figure 6.6 Particle-wall interaction as a function of the incoming particle Weber number (We) and the wall temperature (T_{Wall})

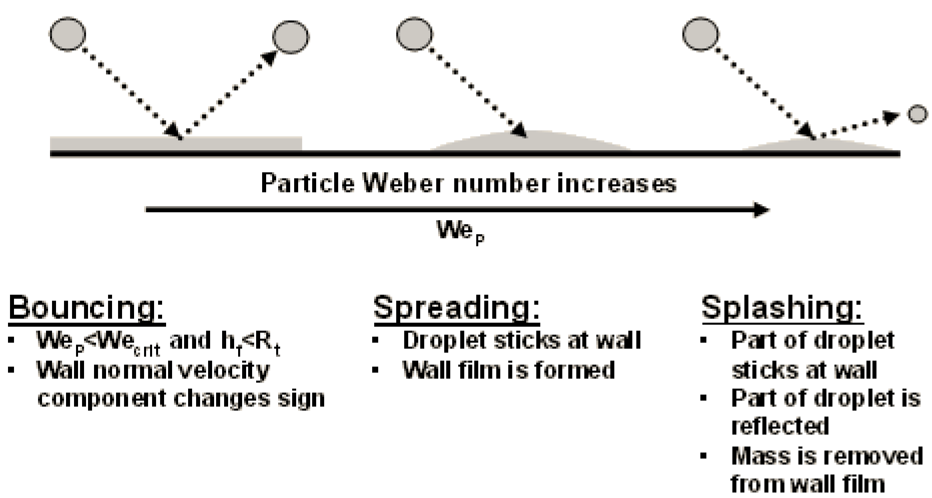


6.7.2.1.1. Cold Wall with Wall Film ($T_{Wall} < T_{PA}$)

In this temperature regime, also known as cold-wetting, the following three impingement options are modeled (see *Figure 6.7* (p. 294)):

- If the incoming particle Weber number (We_P) is below a critical Weber number (We_{crit}) and a wall film exists, then the impinging droplet is reflected.
- If the incoming particle Weber number is above a critical Weber number, then the incoming particle sticks to the wall and forms a wall film (in the particle solver, this means that the particle type is changed from *regular particle* to *wall particle*).
- If the incoming Weber number is increased further, then splashing occurs. In this case, part of the incoming droplet sticks to the wall and only part of it is reflected back into the gas phase. In the current release of ANSYS CFX, mass removal from the wall film is neglected.

Figure 6.7 Particle-wall interactions for cold-wet walls

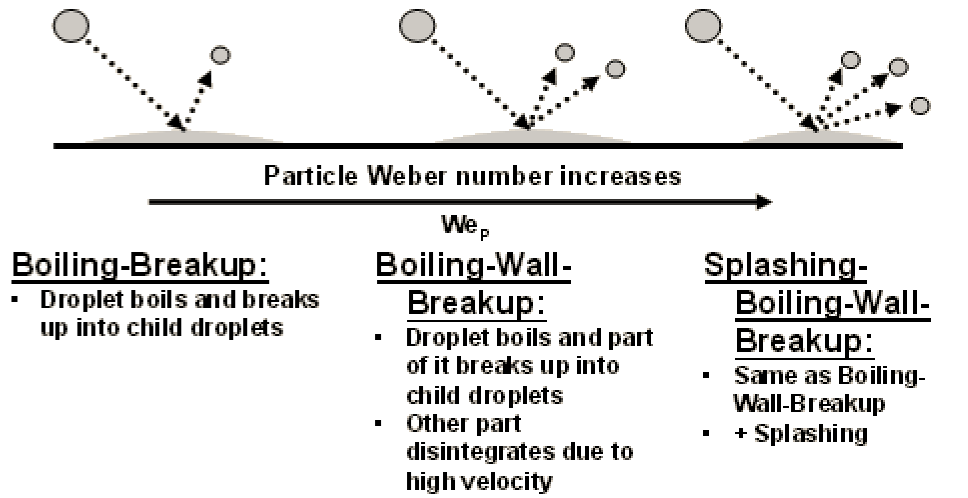


6.7.2.1.2. Hot Wall with Wall Film ($T_{PA} < T_{Wall} < T_{RA}$)

In this temperature regime, also known as hot-wetting, the following impingement options are modeled (see *Figure 6.8* (p. 295)):

- If the incoming particle Weber number is below a critical Weber number and a wall film exists, then the impinging droplet is reflected.
- If the incoming particle Weber number is above a critical Weber number and a wall film exists already, then the incoming particle sticks to the wall and its mass is transferred to the wall film.
- If the incoming particle Weber number is above a critical Weber number, then boiling and splashing are possible and are allowed to occur independently of each other. Also, a part of the impinging droplet is transferred into the wall film.
- If the incoming Weber number is increased further, then splashing occurs. In this case, part of the incoming droplet sticks to the wall and only part of it is reflected back into the gas phase. The mass removal from the wall film is neglected.

Figure 6.8 Particle-wall interactions for hot wall with wall film

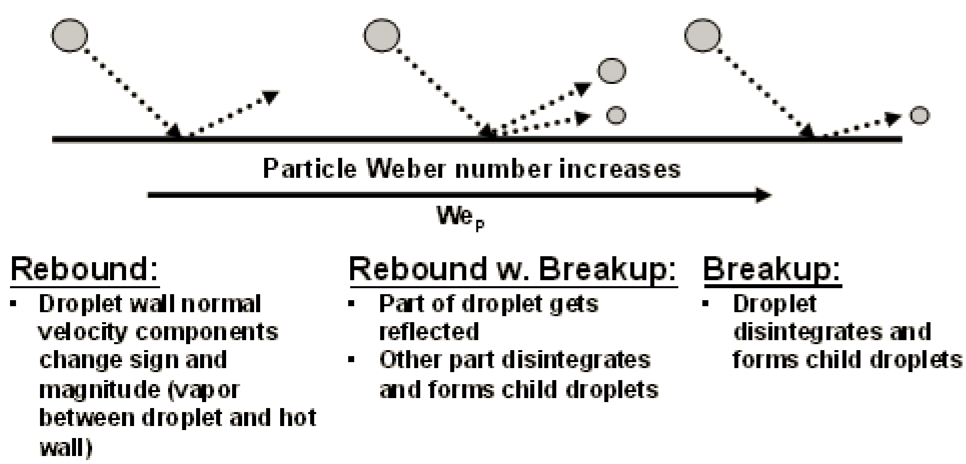


6.7.2.1.3. Hot Wall Without Wall Film ($T_{Wall} > T_{Ra}$)

In this temperature regime, also known as hot-nonwetting, no wall film can exist and the following impingement options are modeled (see *Figure 6.9* (p. 295)):

- If the incoming particle Weber number is below a critical Weber number, then the impinging droplet is reflected from the wall.
- If the incoming particle Weber number is above a critical Weber number, then part of the incoming droplet is reflected from the wall. The other part of the droplet disintegrates and forms child droplets that are also reflected off the wall.
- If the incoming Weber number is increased further, then pure breakup occurs (that is, the incoming droplets complete disintegrates and form child droplets that reflect from the wall).

Figure 6.9 Particle-wall interaction for hot walls without wall film



6.7.2.2. Wall Roughness

Mean roughness values in internal combustion engine applications are of the same order as the drop diameter. Therefore, a numerical consideration of the roughness cannot be neglected.

The Elsaesser model assumes an idealized 2D roughness profile in the impact plane of the drop. It is presented as neighboring isosceles triangles with angle α and a height of R_t.

If a wall film exists, an effective film thickness, h_{fr}, between roughness elements is calculated from:

$$h_{fr} = \sqrt{2 R_t h_f} \qquad \textbf{(6–182)}$$

h_f is the local film thickness. A non-dimensional roughness height, γ, is computed as:

$$\gamma = \frac{R_t}{d_P} \qquad \textbf{(6–183)}$$

with d_p being the droplet diameter.

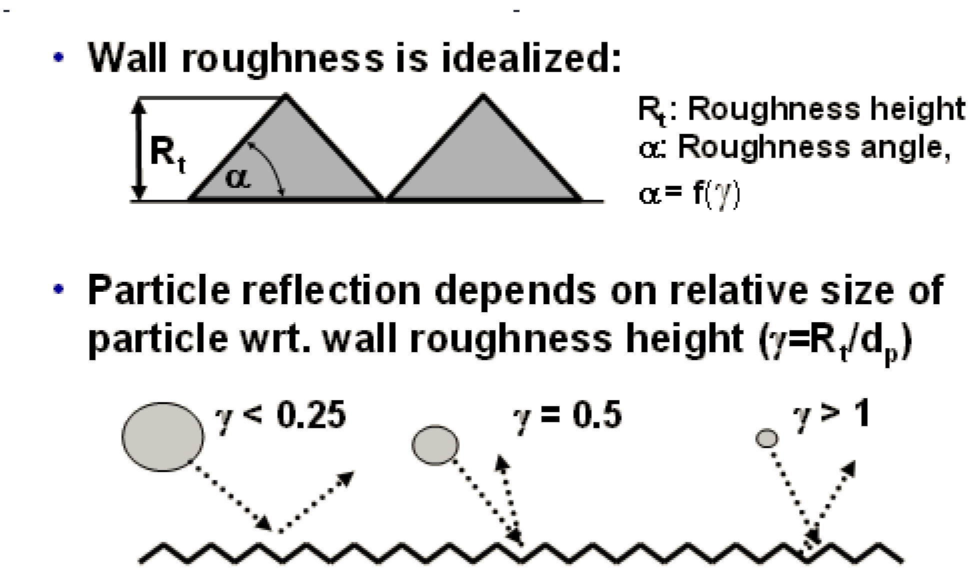
The virtual roughness angle, α, is determined based on h_{fr} and γ as follows:

$\gamma \le 0.25$ and $h_{fr} > R_t$	$\alpha = 0°$
$0.25 < \gamma \le 1.0$ or $h_{fr} \le R_t$	$\alpha = -60°\gamma^2 + 135°\gamma - 30°$
$\gamma > 1$ and $h_{fr} \le R_t$	$\alpha = 45°$

Particle with a non-dimensional size of larger than $R_t = 0.25$ are unaffected by wall roughness, while for smaller droplets the reflection is computed based on the virtual roughness angle, α.

Note that contrary to the originally proposed model of Elsässer [175], multiple reflections within one roughness element are not considered.

Figure 6.10 Treatment of rough walls



6.7.2.3. Range of Applicability, Input Data and Restrictions

The Elsaesser particle-wall interaction model is mainly targeted towards the application in internal combustion engines, because most of the correlations used in the model are strictly only valid for gasoline type of fuels and assume that the wall material is aluminum. The model uses over 50 different model constants internally.

To run a simulation using the Elsaesser wall interaction model, the following input is required:

- Wall material density
- Wall material heat capacity
- Wall material thermal conductivity

The particle solver automatically uses any roughness information you provided to simulate the droplet reflection.

The Elsaesser model uses the *child droplet generation* capability of ANSYS CFX. This feature is turned on automatically, if the Elsaesser (or any other) advanced particle-wall interaction model is selected. Particles that have hit the wall and have been changed into *wall particles* are reported separately in the particle fate section. The wall film is represented by *wall particles* and it is assumed that this film does not move due to external forces. The wall film can exchange mass and energy with its surrounding, but no momentum.

6.7.3. Stick-to-Wall Model

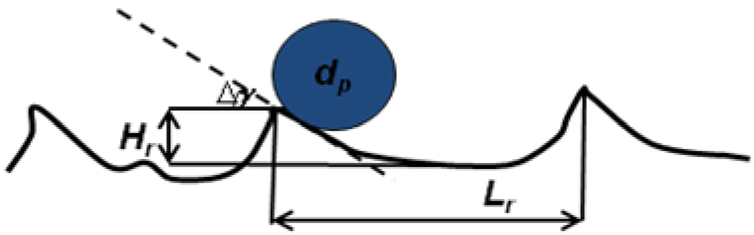
The model enforces all particles that hit a wall to become part of the wall film, regardless of their impact velocity or impact angle. Also, particles that are collected on a wall are able to interact with their surroundings by exchanging mass and energy (for example, during the droplet evaporation).

6.7.4. The Sommerfeld-Frank Rough Wall Model (Particle Rough Wall Model)

The majority of industrially important disperse multiphase-flows are confined flows, such as flows in cyclone separators or in pneumatic conveying pipe systems. The motion of large particles, which is dominated by inertia, is strongly influenced by the confinement. Studies of the wall-collision process have shown that irregularities due to wall-roughness and deviation of particle shape from a sphere play important roles [208], [209], [210].

To account for the influence of wall roughness on the particle-wall collision, the irregular bouncing model of Frank [211] is used. This model is based on the virtual wall model that was proposed by Sommerfeld [212].

Figure 6.11 Particle-Wall Interaction at a rough wall. Definition of $\Delta\gamma$ and characteristic roughness element length scales



In this model, it is assumed that the reflection of a particle at a rough wall can be modeled as the reflection of the particle at a virtual wall that is inclined with an angle of γ_w relative to the real wall. The inclination angle γ_w is a function of the wall roughness scale and the particle diameter.

In the model of Frank [4] the inclination angle, γ_w, is sampled from a Gaussian distribution with a mean value of 0° and a standard deviation of $\Delta\gamma$. The standard deviation depends on the particle diameter d_p and the roughness parameters. It is estimated as:

$$\Delta\gamma = \begin{cases} \arctan\dfrac{2\,\Delta H_r}{L_r} & if \quad d_p < \dfrac{L_r}{\sin\left(\arctan\dfrac{2H_r}{L_r}\right)} \\ \arctan\dfrac{2H_r}{L_r} & if \quad d_p > \dfrac{L_r}{\sin\left(\arctan\dfrac{2H_r}{L_r}\right)} \end{cases} \tag{6–184}$$

where:

- L_r is the mean cycle of roughness (the average distance between peaks of wall material)
- H_r is the mean roughness height
- ΔH_r is the standard deviation of the roughness height

6.8. Quasi Static Wall Film Model

In ANSYS CFX, a so called *quasi static* wall film model is implemented that neglects the wall film movement due to external forces, such as shear stress, gravity, pressure forces, etc. In this model, wall particles interact only with their surroundings via mass transfer (evaporation) or heat transfer (wall conduction, convection).

The following topics will be discussed:

6.8.1. Assumptions

The following main assumptions apply to the wall-film model:

- The wall film thickness is thin and the wall film does not alter the geometry.
- The existence of a wall film has no influence on the turbulence generation in the wall boundary layer.
- Film particles are in direct contact to the wall and heat transfer from wall to film takes place by conduction.
- Film particles originating from different particle types do not mix (that is, fuel droplets that hit oil-covered walls do not interact with each other). In ANSYS CFX, the particle solver checks that only one particle type can form a wall film on a domain boundary.
- In the non-flooded regime wall particles keep their spherical shape.
- The simulation is assumed to be turbulent.
- The wall film model can be used only in transient simulations.
- The wall film model cannot be used together with dispersed Eulerian particles.

6.8.2. Determination of Flooded Regime

By default, it is assumed that a wall face is covered by a wall film (*flooded*) as soon as the first wall particle is created on that particular face.

6.8.3. Energy Transfer to and from the Wall Film

The total heat transfer to a single wall film droplet is found from the following energy balance:

$$mc_p\frac{dT}{dt}=Q_{cond}+Q_{conv}+Q_{evap}+Q_{rad} \tag{6–185}$$

6.8.3.1. Conductive Heat Transfer

Q_{cond} is the heat conducted from the wall, as given by:

$$Q_{cond}=h\,A_P\left(T_w-T_p\right) \tag{6–186}$$

where $h=\lambda_P/\Delta$ is the conductive heat transfer coefficient and A_P is the wall area covered by the particle. This term is always included in the energy equation for wall particles.

6.8.3.1.1. Non-flooded Regime

In the non-flooded regime, Δ is assumed to be equal to the drop diameter. The wall contact area is computed from $A_p = 0.25 \pi d_p^2$.

6.8.3.1.2. Flooded Regime

In the flooded regime, Δ is set equal to the wall film thickness, δ_f. The wall contact area is computed from $A_p = Vol_p / \delta_f$, with Vol_p being the volume of the particle and δ_f the wall film thickness that was computed at the previous time step.

6.8.3.2. Convective Heat Transfer

The heat transferred from the gas to the film, Q_{conv}, is given by:

$$Q_{conv} = h\, A_P \,(T_g - T_p) \quad \textbf{(6–187)}$$

where A_p is the area covered by a wall particle, T_g is the conservative gas temperature at the particle position, and h is the film heat transfer coefficient.

This term is always included in the energy equation for wall particles.

6.8.3.2.1. Non-flooded Regime

For the non-flooded regime, A_p and h are computed as:

$$\begin{aligned} A_p &= \pi d_p^2 \\ h &= Nu \lambda_F / d_P \end{aligned} \quad \textbf{(6–188)}$$

The Nusselt number used for the calculation of h is computed using the Ranz-Marshall correlation for a sphere:

$$Nu = 2 + 0.6 * \sqrt{Re}\, Pr^{1/3} \quad \textbf{(6–189)}$$

6.8.3.2.2. Flooded Regime

In the flooded regime, h is set equal to the transfer coefficient for energy computed by the flow solver.

Note

The energy transfer coefficient is only available for turbulent flows. A physics check is added to the tracker setup phase to make sure that you set the flow type to *turbulent*.

6.8.3.3. Calculation of the Average Wall Film Temperature

The average wall film temperature (that is, the temperature of all wall particles associated to a boundary vertex) is computed by using the particle vertex machinery, by adding up contributions of *wall particles* (particles that already exist and new particles that turn into wall particles). This is done as a post-processing step after all particles have been tracked.

6.8.3.4. Evaporation from Film

Q_{evap} accounts for the energy that is removed from the wall film particle, as it evaporates into the surrounding medium:

$$Q_{evap} = \dot{m}_p V \tag{6–190}$$

V is the latent heat of vaporization, $\dot{m}_p$ is the rate of evaporation.

6.8.3.4.1. Non-flooded Regime (Non-boiling)

The mass transfer rate for a single droplet is computed using the *Liquid Evaporation Model*. For particles below the boiling point, the following relation is used:

$$\dot{m}_P = h_m A_p \left(\frac{W_P}{W_F} \log\left[\frac{1 - X_P}{1 - X_S} \right] \right) \tag{6–191}$$

The transfer coefficient, h_m, is defined as:

$$h_m = Sh \rho_F D_F / d_P \tag{6–192}$$

The Sherwood number is calculated using the Ranz-Marshall correlation for a sphere as:

$$Sh = 2 + 0.6 * \sqrt{\mathrm{Re}}\ (v_F / D_F)^{1/3}$$

6.8.3.4.2. Flooded Regime (Non-boiling)

The mass transfer rate of a particle component, p, into the coupled Euler phase can be determined by considering species mass balances on control surfaces on either side of the phase interface:

$$\dot{m}_P = Y_S \dot{m}_P + J_P \tag{6–193}$$

Y_S is the mass fraction of the particle component at the film surface, and J_P is the surface diffusive mass flux.

The surface mass flux can be expressed in terms of a mass transfer coefficient, k, as:

$$J_p = k\ (Y_s - Y) \tag{6–194}$$

Y is the mass fraction of the volatile particle component as obtained from the flow field solution. The transfer coefficient, k, is set equal to the transfer coefficient for scalars as computed by the flow solver.

Inserting Equation 6–194 (p. 302) into Equation 6–193 (p. 301) and solving for $\dot{m}_p$, gives:

$$\dot{m}_P = k\frac{Y_S - Y}{1 - Y_S} \tag{6–195}$$

6.8.3.4.3. Flooded and Non-flooded Regime (Boiling Particles)

For particles above the boiling point this relation is used:

$$\dot{m}_P = \frac{-\left(Q_{cond} + Q_{conv} + Q_{rad}\right)}{V} \tag{6–196}$$

where Q_{cond} and Q_{conv} is given by Equation 6–186 (p. 299) and Equation 6–187 (p. 300) respectively. Q_{rad} is computed in the same way as for regular particles.

6.8.4. Mass Transfer to and from the Wall Film

Mass transfer due to evaporation is computed from Equation 6–191 (p. 301), Equation 6–195 (p. 302) or Equation 6–196 (p. 302) depending on the regime (viz., flooded or non-flooded) and the state of the particle (viz., boiling or non-boiling). Mass transfer from impinging droplets is computed by simply adding the droplet mass to the film mass on a particular wall face.

6.8.5. Wall Film Thickness

The wall film thickness on a given (control volume) wall face is computed from the volume fraction, r_{pw}, of all wall particles in a control volume, times the 'control volume' volume, Vol_{cv}, divided by the local control volume face area, $A_{w,cv}$:

$$\delta_f = \frac{r_{pw} Vol_{cv}}{A_{w,cv}} \tag{6–197}$$

The generalization to control volume sectors or element faces is straightforward.

6.8.6. Wall Film in Moving Mesh Applications

Even though the wall film is assumed to be not moving due to external forces, there may be situations, where the wall film is either moving with the wall (for example, piston top) or moving relative to the grid (for example, cylinder wall). The second situation occurs due to the moving mesh approach used for simulations with changing geometries.

Both situations are handled with the current approach without user interference. The distinction whether wall particles move with a wall or move relative to the underlying mesh is based on the value of the so called *Wall Velocity*, which is given as:

For the moving wall:

Velocity of mesh (V_{mesh}) = Velocity of wall (V_{wall}) = Velocity of wall-particle ($V_{wall\text{-}particle}$)

For the stationary wall:

Velocity of mesh (V_{mesh}) $\neq$ 0, Velocity of wall (V_{wall}) = Velocity of wall-particle ($V_{wall\text{-}particle}$) = 0

6.8.6.1. Wall Film Moving with Mesh

The situation is identical to the case, where wall particles are located at a non-moving wall.

When a particle has hit a wall face, the topological data of the impact is known and is stored in the particle database. Because the particle does not move, there is no need to update this data during the simulation.

6.8.6.2. Wall Film Moving Relative to Underlying Mesh

This case requires the update of the wall particle topology data during the simulation, as wall particles may cross elements due to the specified mesh movement. The relocalization of wall particles is currently done after all particles have been tracked. Due to performance reasons the update is not done while tracking wall particles. Relocalizing wall particles at the end of a fluid time step therefore puts an upper limit onto the maximum allowed fluid time step, because for accurate simulations wall particle should not cross more than one element per fluid time step. If a too large time step is used, wall particles will put their heat and mass sources into the last known element.

6.8.7. User Control for Heat and Mass Transfer Terms of Wall Particles

In ANSYS CFX, the coupling of heat and mass transfer terms of wall particles into the coupled Eulerian phase can be controlled, similar to the coupling control for regular particles. However, it is not possible to control the coupling on a term-by-term basis, but the coupling is applicable on the *global* equation level. This means that if the particle energy equation is run in a *fully coupled* mode, then this will not only apply to regular particles, but also to wall bounded particles (and the wall film they represent). Likewise, if the coupling of the energy equation is set to *one-way coupled*, then this also applies to the wall particles of that particular particle type. The coupling control for wall particles is essentially identical to what is currently done for regular particles. The same applies to the particle mass transfer equation.

Chapter 7: Combustion Theory

This chapter covers the implementation of the combustion models in the CFX-Solver and extends the ideas covered in *Multicomponent Flow* (p. 44). You should be familiar with the multicomponent flow chapter before reading this section.

This chapter describes:

The following sections outline the basis of the implementation of combustion modeling in CFX. First, the transport equations for energy and the components are revisited, then the chemical reaction rate computation is described, and finally, the computation of the rate of progress of a chemical reaction is explained in the context of the Eddy Dissipation and Finite Rate Chemistry Models. Extinction is modeled by setting the reaction rate locally to zero.

7.1. Transport Equations

Combustion models in CFX use the same algorithm used for Multicomponent Fluid with the addition of a source/sink term due to chemical reactions. The equation of transport for component *I* with mass fraction, Y_I is then:

$$\frac{\partial\ (\rho\ Y_I)}{\partial t}+\frac{\partial \left(\rho\ u_j\ Y_I\right)}{\partial x_j}=\frac{\partial}{\partial x_j}\left(\Gamma_{I_{eff}}\frac{\partial\ Y_I}{\partial x_j}\right)+S_I \tag{7–1}$$

where the source term S_I is due to the chemical reaction rate involving component *I*.

7.2. Chemical Reaction Rate

In general, chemical reactions can be described in terms of K elementary reactions involving N_C components that can be written as:

$$\left[\sum_{I=A,B,C,\ldots}^{N_C} \nu'_{kI} I \Leftrightarrow \sum_{I=A,B,C,\ldots}^{N_C} \nu''_{kI} I \right] \tag{7–2}$$

where ν_{kI} is the stoichiometric coefficient for component I in the elementary reaction k. The stoichiometric coefficients for reactants are denoted by a single dash, ν'_{kI}, and those for products by a double dash, ν''_{kI}.

The rate of production/consumption, S_I, for component I can be computed as the sum of the rate of progress for all the elementary reactions in which component I participates:

$$S_I = W_I \sum_{k=1}^{K} (\nu''_{kI} - \nu'_{kI}) \; R_k \tag{7–3}$$

where R_k is the elementary reaction rate of progress for reaction k, which in CFX can be calculated using the Eddy Dissipation Model or/and the Finite Rate Chemistry Model. W_I is the molar mass of component I.

7.3. Fluid Time Scale for Extinction Model

When the model for flame extinction at high turbulence is activated, local extinction occurs when .$\tau_t < \tau_c$.

While τ_c is directly specified, the turbulence time scale is computed from the CFD solution fields. One possibility is to apply the Kolmogorov time scale:

$$\tau_{\text{Kolmogorov}} = \sqrt{\frac{\nu}{\varepsilon}} \tag{7–4}$$

An alternative is to use the mixing time scale:

$$\tau_{\text{mixing}} = \frac{k}{\varepsilon} \tag{7–5}$$

7.4. The Eddy Dissipation Model

The eddy dissipation model is based on the concept that chemical reaction is fast relative to the transport processes in the flow. When reactants mix at the molecular level, they instantaneously form products. The model assumes that the reaction rate may be related directly to the time required to mix reactants at the molecular level. In turbulent flows, this mixing time is dominated by the eddy properties and, therefore, the rate is proportional to a mixing time defined by the turbulent kinetic energy, *k, and dissipation,* ε.

$$\text{rate} \propto \frac{\varepsilon}{k} \quad \textbf{(7–6)}$$

This concept of reaction control is applicable in many industrial combustion problems where reaction rates are fast compared to reactant mixing rates.

In the Eddy Dissipation model, the rate of progress of elementary reaction *k*, is determined by the smallest of the two following expressions:

7.4.1. Reactants Limiter

$$R_k = A\,\frac{\varepsilon}{k}\,\min\left(\frac{[I]}{v'_{kI}}\right) \quad \textbf{(7–7)}$$

where [*I*] is the molar concentration of component *I* and *I* only includes the reactant components.

7.4.2. Products Limiter

$$R_k = A\,B\,\frac{\varepsilon}{k}\left(\frac{\sum_P (I)\;W_I}{\sum_P v''_{kI}\,W_I}\right) \quad \textbf{(7–8)}$$

where P loops over all product components in the elementary reaction k.

The products limiter is disabled when the model coefficient B is set to a negative value. For both single step and multi-step reaction schemes, it is turned off by default, (set to -1), but may be turned on by explicitly setting the model coefficient B to a positive value (although this is not recommended for multistep reaction schemes).

7.4.3. Maximum Flame Temperature Limiter

Optionally, a maximum flame temperature may be applied for the Eddy Dissipation model. The reaction rate is smoothly blended to zero when the specified upper temperature limit is approached. This is implemented by an additional bound added to the minimum condition in the EDM reaction rate:

$$R_{k,MFT} = A\,\frac{\varepsilon}{k}\,C_{MFT} \quad \textbf{(7–9)}$$

where:

$$C_{MFT} = \max\{(T_{\max} - T), 0\,[K]\} \cdot \frac{\rho C_P}{\Delta H_R} \qquad \textbf{(7–10)}$$

C_{MFT} may be interpreted as a virtual concentration, which vanishes if the temperature is equal to the maximum flame temperature. C_P is the specific heat capacity of the fluid mixture at constant pressure and ΔH_R is the reaction heat release per mole.

7.5. The Finite Rate Chemistry Model

The Finite Rate Chemistry model, as implemented in CFX, assumes that the rate of progress of elementary reaction *k* can be reversible only if a backward reaction is defined. Therefore, the rate of progress R_k, is computed as:

$$R_k = \left(F_k \prod_{I=A,B,\ldots}^{N_C} [I]^{r'_{kI}} - B_k \prod_{I=A,B,\ldots}^{N_C} [I]^{r''_{kI}} \right) \qquad \textbf{(7–11)}$$

where [*I*] is the molar concentration of component *I* and F_k and B_k are the forward and backward rate constants respectively.

r represent the reaction order of component *I* in the elementary reaction *k*. This reaction order is equal to the stoichiometric coefficient for elementary reactions, but it can be different for certain global reactions.

The only built-in formula for the forward and backward rate constants assumes an Arrhenius temperature dependence as:

$$F_k = A_k\, T^{\beta_k} \exp\left(-\frac{E_k}{R\,T} \right) \qquad \textbf{(7–12)}$$

$$B_k = A_k\, T^{\beta_k} \exp\left(-\frac{E_k}{R\,T} \right) \qquad \textbf{(7–13)}$$

where:

- A_k is pre-exponential factor
- β_k is the temperature exponent (dimensionless)
- E_k is the activation energy
- *T* is the absolute temperature

R_k can also be specified directly without using relations.

Separate sets of coefficients A_k, β_k and E_k are applied to forward and backward rates.

7.5.1. Third Body Terms

In cases that a third body is needed for the reaction to occur, the rate of progress described earlier is scaled by:

$$\sum_{i=1}^{N} \alpha_{ki} \ [I_i] \qquad \textbf{(7–14)}$$

where α_{ki} is the relative participation of component I_i in reaction k.

For those components that have a high probability to participate in the reaction, the coefficient α_{ki} is higher than those that rarely participate or do not participate at all ($\alpha_{ki} = 0$). If the third body term is present in a reaction, CFX assumes the default efficiency for all components (**Option** = `Default`). Efficiency factors can be specified for individual species by setting **Option** to `Efficiency Factor List` and listing the components as well as their efficiency factors in the `Materials List` and `Efficiency Factor List` parameters, respectively. If the `Efficiency Factor List` option is effective, the default still applies to those components that are not listed.

7.6. The Combined Eddy Dissipation/Finite Rate Chemistry Model

The effective reaction rate, for the combined model, is computed to be the minimum of the Finite Chemistry Rate and the Eddy Dissipation rate.

The Theory documentation for this model is the same as for others:

- *The Eddy Dissipation Model* (p. 306)
- *The Finite Rate Chemistry Model* (p. 308)

7.7. Combustion Source Term Linearization

The combustion source terms may have a dominant influence on the solution of the scalar and energy equations. Thus, it is important to treat the combustion source terms carefully in order to obtain robust convergence of the fluid flow.

A property of multicomponent fluids is that the mass fraction of any given component is bounded between 0 and 1. Combustion tends to drive reactant concentrations towards the lower limit and product concentrations toward the upper limit. If the timestep is large, the combustion sources may cause scalars to exceed these bounds. Thus, the sources may need to be moderated to maintain physically realistic mass fractions.

The combustion sources in CFX have been linearized to prevent the formation of negative mass fractions. Consider the solution of component *I* with the source term, *R*, which is calculated from *Equation 7–15* (p. 309) in each control volume. To prevent the mass fraction of *I* from exceeding the bounds of 0 to 1, the source term is calculated according to:

$$\text{source} = \left(\frac{R_I - |R_I|}{2\, I^*} \right) I + \left(\frac{R_I + |R_I|}{2\,(1 - I)^{**}} \right) (1 - I) \qquad \textbf{(7–15)}$$

where $I^* = \max\,(\delta, I)$ and where $(1 - I)^{**} = \max\,(\delta, 1 - I)$

and ε is a small number (set to 10^{-6}). The combustion reaction rate should approach 0 whenever any of the reactant or product mass fractions approach 0. If the source is positive (for products), then the first term on the right hand side of *Equation 7–15* (p. 309) is zero and the source is:

$$\text{source} = R_I \quad \text{if} \quad I \geq 1 - \delta \tag{7–16}$$

otherwise:

$$\text{source} = R_I \frac{(1 - I)}{\delta} \tag{7–17}$$

Thus as $\text{products} \rightarrow 1 . \text{source} \rightarrow 0$

If the source is negative (reactants), then the second term on the right hand side of *Equation 7–15* (p. 309) is zero and the source is:

$$\text{source} = R_I \qquad \text{if} \qquad I \geq \delta \tag{7–18}$$

otherwise:

$$\text{source} = \frac{(R_I I)}{\delta} \tag{7–19}$$

Thus as $\text{reactant} \rightarrow 0 . \text{source} \rightarrow 0$

This treatment of combustion sources allows larger timesteps to be used in calculating a steady state solution than would be possible without the linearization.

7.8. The Flamelet Model

The Flamelet concept [37] for non premixed combustion describes the interaction of chemistry with turbulence in the limit of fast reactions (large Damköhler number). The combustion is assumed to occur in thin sheets with inner structure called Flamelets. The turbulent flame itself is treated as an ensemble of laminar Flamelets which are embedded into the flow field.

The Flamelet model is a non equilibrium version of the classical "Burke-Schumann" limit. It adds new details to the simulation of combustion processes compared to other common combustion models for the price of the solution of only two scalar equations in the case of turbulent flow. An arbitrary number of intermediates may be specified as long as their laminar chemistry is known.

The main advantage of the Flamelet model is that even though detailed information of molecular transport processes and elementary kinetic reactions are included, the numerical resolution of small length and time scales is not necessary. This avoids the well-known problems of solving highly nonlinear kinetics in fluctuating flow fields and makes the method very robust. Only two scalar equations have to be solved independent of the number of chemical species involved in the simulation. Information of laminar model flames are pre-calculated and stored in a library to reduce computational time. On the other hand, the model is still restricted by assumptions like fast chemistry or the neglecting of different Lewis numbers of the chemical species.

The coupling of laminar chemistry with the fluctuating turbulent flow field is done by a statistical method. The PDF used can in principle be calculated at every point in the flow field by solving a PDF transport equation as shown by Pope and many others. The most often mentioned advantage of this method is that the nonlinear chemical source term needs no modeling. Even though the method avoids some modeling which is necessary if using moment closure, it still requires modeling of some of the most important terms, in particular, the fluctuating pressure gradient term and the molecular diffusion term. If combustion occurs in thin layers as assumed here, the molecular diffusion term is closely coupled to the reaction term and the problem of modeling the chemical source term is then shifted towards modeling the diffusion term.

However, there is no source term in the mixture fraction equation, which is the principal transport equation in the Flamelet model. Therefore, a presumed beta-PDF, which is a commonly accepted choice, is used here. Additionally, this avoids the extremely large computational efforts of calculating the PDF in 3D with a Monte Carlo method.

The following list outlines the assumptions made to derive the Flamelet model:

- Fast Chemistry
- Unity Lewis numbers for all species, $Le_i = 1$
- Combustion is in the Flamelet Regime
- Two feed system, that is, fluid composition at boundaries must be pure "fuel," pure "oxidizer" or a linear blend of them.
- Diffusion flames. For premixed or partially premixed combustion, the Flamelet model can be combined with a model for reaction progress. For details, see *Burning Velocity Model (Premixed or Partially Premixed)* (p. 316).

Fluid properties, including temperature and density, are computed from the mean composition of the fluid in the same way as for other combustion models, such as the Eddy Dissipation model.

The Flamelet model as implemented in CFX can be applied for non-adiabatic configurations. The only limitation is that changes in the composition of the fluid due to different temperature and pressure levels are not accounted for. However, the effect of heat loss and pressure on density and temperature is taken into account. For heat losses occurring in many combustion devices, the influence of heat losses on composition is sufficiently small to be neglected.

In a large number of industrial combustion devices, pure non-premixed combustion is less present than premixed or partly premixed combustion. In CFX, a model for premixed and partially premixed combustion is available, which involves the Flamelet model as a sub-model. For details, see *Burning Velocity Model (Premixed or Partially Premixed)* (p. 316).

7.8.1. Laminar Flamelet Model for Non Premixed Combustion

A diffusion flame is characterized by the diffusion of reactants into the flame front. While convective and diffusive time scales are of the same order of magnitude, the chemical time scales are much smaller for typical combustion processes of interest. Several approaches to treat chemical reactions have been developed and tested during the last decades.

The assumption of local chemical equilibrium has often been used in modeling the fast chemistry regime. For hydrocarbon flames, however, the assumption of local chemical equilibrium results in an over-prediction of intermediates like CO and H_2. This suggests that non equilibrium effects are important in modeling these flames. Further essential non equilibrium effects are flame extinction, lift-off and blow-out.

Another well known approach is the flame sheet model of Burke and Schuman often characterized as 'mixed is burned'. Here, only the mixture of the reactants is calculated and the chemistry is treated as infinite fast and complete when mixing is complete. Therefore, combustion occurs in an infinitely thin sheet at the surface of stoichiometric mixture. Again non equilibrium effects are not taken into account, which are important if the strongly varying time scales of the turbulent flow fields approach those of the chemical reactions.

Linan [42] was the first who incorporated non equilibrium effects in diffusion flames. He analyzed the inner structure of the thin laminar flame sheet, referred to here as a Flamelet using an asymptotic description with a large Damköhler number as the expansion parameter. The Damköhler number is the ratio of flow to chemical time scales:

$$D_a = \frac{t_t}{t_c} \tag{7–20}$$

Linan's method is similar to Prandtl's boundary layer theory. The inner layer of the thin reaction sheet with well defined structure is called "Flamelet" from now on. A more simple description of flamelets is possible by using the mixture fraction, which is the sum of all elementary mass fractions:

$$Z_j = \sum_{i=1}^{n} \frac{a_{ij} M_j}{M_i} Y_i \tag{7–21}$$

which have its origin in a system consisting of fuel inlet (labeled 1) and oxidizer inlet (labeled 2). Here Y_i is the mass fraction of species i, Z_j the mass fraction of a chemical element j (such as C or H), M the molecular mass, and a_{ij} the number of elements j in the molecule i

$$Z = \frac{Z_{\text{Fuel}}}{Z_{\text{Fuel},1}} = 1 - \frac{Z_{\text{Oxidizer}}}{Z_{\text{Oxidizer},2}} \tag{7–22}$$

Assuming equal diffusivities and heat capacities for all chemical components a conservation equation for the mixture fraction Z can be derived by summing all species conservation equations, and the chemical source terms therefore cancel exactly. The mixture fraction is not influenced by chemical reactions because it deals with elements rather than molecules, and elements are not affected by chemistry.

$$\frac{\partial\ (\rho Z)}{\partial t} + \frac{\partial\ (\rho Z)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\rho D \frac{\partial Z}{\partial x_j}\right) \tag{7–23}$$

The isosurface $Z = Z_{st}$ determines the location of stoichiometric mixture. To be able to describe the location of the flamelets anywhere in the flow field, a new coordinate system is introduced here. One of its coordinates is locally perpendicular to the surface of stoichiometric mixture.

The transformation is shown for the temperature equation as an example. The 3 terms at the right hand side represent chemical reactions, radiation and the transient pressure gradient, respectively. The last is important for combustion involving fast changing pressure such as in closed burning chambers of a piston engine.

$$\rho \frac{\partial T}{\partial t} + \rho\, \upsilon_\alpha \frac{\partial T}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha}\left(\rho D \frac{\partial T}{\partial x_\alpha}\right) = \sum_{k=1}^{r} \frac{Q_k}{c_p}\omega_k + \frac{q_R}{c_p} + \frac{1}{c_p}\frac{\partial P}{\partial t} \tag{7–24}$$

After applying the following transformation rules:

$$\frac{\partial}{\partial t} = \frac{\partial}{\partial \tau} + \frac{\partial Z}{\partial t}\frac{\partial}{\partial Z} \tag{7–25}$$

$$\frac{\partial}{\partial x_k} = \frac{\partial}{\partial Z_k} + \frac{\partial Z}{\partial x_k}\frac{\partial}{\partial Z}\,(k = 2, 3) \tag{7–26}$$

$$\frac{\partial}{\partial x_1} = \frac{\partial Z}{\partial x_1}\frac{\partial}{\partial Z} \tag{7–27}$$

and using the assumption of a constant Lewis number:

$$Le = \frac{\lambda}{\rho\, c_p D} \tag{7–28}$$

you obtain the temperature equation in the form:

$$\begin{aligned}
&\rho\left(\frac{\partial T}{\partial \tau} + u_2\frac{\partial T}{\partial Z_2}u_3\frac{\partial T}{\partial Z_3}\right) - \frac{\partial\,(\rho D)}{\partial x_2}\frac{\partial T}{\partial Z_2} - \frac{\partial\,(\rho D)}{\partial x_3}\frac{\partial T}{\partial Z_3} \\
&- \rho D\left[\left(\frac{\partial Z}{\partial x_\alpha}\right)^2\frac{\partial^2 T}{\partial Z^2} + 2\frac{\partial Z}{\partial x_2}\frac{\partial^2 T}{\partial Z \partial Z_2} + 2\frac{\partial Z}{\partial x_3}\frac{\partial^2 T}{\partial Z \partial Z_3} + \frac{\partial^2 T}{\partial Z_2^2} + \frac{\partial^2 T}{\partial Z_3^2}\right] \\
&= \frac{1}{\rho c_p}\sum_{i=1}^{n} h_i \dot{m}_i + \frac{q_R}{c_p} + \frac{1}{c_p}\frac{\partial P}{\partial t}
\end{aligned} \tag{7–29}$$

Because the Flamelet is assumed to be thin, only gradients normal to the surface of stoichiometric mixture are large, and all terms without a second derivative in respect to the mixture fraction Z can be neglected. When this is done formally by introducing a stretched coordinate, it turns out that the remaining equation is essentially one dimensional. The same arguments apply for all other equations as well:

$$\frac{\partial T}{\partial t} - \frac{\chi_{st}}{2}\frac{\partial^2 T}{\partial Z^2} = \frac{1}{\rho\, c_p}\sum_{i=1}^{n} h_i \dot{m}_i + \frac{q_R}{c_p} + \frac{1}{c_p}\frac{\partial P}{\partial t} \tag{7–30}$$

Non equilibrium effects - the influence of the outer flow field on the inner reaction zone - are described by the scalar dissipation rate χ_{st} at stoichiometric mixture.

$$\chi_{st} = 2\, D_{st}\, (\nabla Z)^{2}_{st} \tag{7–31}$$

It represents a reciprocal residence time which is increased by stretch effects of the flow field and reduced by diffusion. At a critical value of $\chi = \chi_q$ the flame shows a threshold behavior and extinguishes. The stretch in physical space leads to a reaction zone which is so thin that the production of heat cannot balance the heat loss caused by conduction. The temperature drops to unburnt values and the reactions freeze. `Freeze' means they are at lower temperatures and are much slower than the fluid time scales.

The important conclusion of this derivation is that flamelet structures in the presence of fast chemistry can be described by one dimensional model flames. This will be used to model reacting turbulent flow fields.

7.8.2. Coupling of Laminar Flamelet with the Turbulent Flow Field

In the turbulent flow field, the Favre averaged (tilde superscript) mixture fraction equation is solved:

$$\frac{\partial\, (\overline{\rho}\, \tilde{Z})}{\partial t} + \frac{\partial \left(\overline{\rho}\, \overline{u}_j\, \tilde{Z}\right)}{\partial x_j} = \frac{\partial}{\partial x_j} \left\{ \left(\overline{\mu} + \frac{\mu_t}{\sigma_Z} \right) \frac{\partial \tilde{Z}}{\partial x_j} \right\} \tag{7–32}$$

The Favre averaging is extensively explained in the theory documentation of CFX. Statistical information on the mixture fraction is obtained from the variance of Z.

$$\frac{\partial \left(\overline{\rho}\, \widetilde{Z''^{2}} \right)}{\partial t} + \frac{\partial \left(\overline{\rho}\, \tilde{u}_j\, \widetilde{Z''^{2}} \right)}{\partial x_j} = \frac{\partial}{\partial x_j} \left\{ \left(\overline{\mu} + \frac{\mu_t}{\sigma_{Z''^2}} \right) \frac{\partial \widetilde{Z''^{2}}}{\partial x_j} \right\} + 2\, \frac{\mu_t}{\sigma_Z} \left(\frac{\partial \tilde{Z}}{\partial x_j} \right)^{2} - \overline{\rho}\, \tilde{\chi} \tag{7–33}$$

The structure of this equation is similar to the mixture fraction equation itself except for last two terms on the right hand side. The first source term is the production and the second source term models the dissipation of the variance. Here, χ stands for the scalar dissipation rate and is modeled in turbulent flow using the empirical relation:

$$\tilde{\chi} = C_\chi \frac{\tilde{\varepsilon}}{\tilde{k}}\, Z''^{2} \tag{7–34}$$

It includes the effects of strain as well as mixture fraction fluctuations. The standard set of model coefficients in CFX is $\sigma_Z = 0.9$, $\sigma_{Z''^2} = 0.9$ and $C_\chi = 2.0$

The mean composition of the fluid is computed as a function of mean mixture fraction, mixture fraction variance and scalar dissipation rate by look-up in a flamelet library:

$$\tilde{Y}_i = \tilde{Y}_i\left(\tilde{Z}, \widetilde{Z''^2}, \tilde{\chi}_{st}\right) = \int_0^1 Y_i\left(Z, \tilde{\chi}_{st}\right) \cdot P_{\tilde{Z}, \widetilde{Z''^2}}(Z)\ \partial Z \tag{7–35}$$

The integration over the probability density function (PDF) P is not carried out during the CFD calculation, but is part of the generation process for the Flamelet library. For details, see *Flamelet Libraries* (p. 315). The CFD solver looks up the preintegrated values from the library.

In principle, many types of PDF could be applied, but the most commonly agreed choice is the Beta-PDF. The shape of P is presumed to be that of a beta function (β-function):

$$P_{Z, \widetilde{Z''^2}}(Z) = \frac{Z^{a-1}(1-Z)^{b-1}}{\int_0^1 \psi^{a-1}(1-\psi)^{b-1}\,d\psi} \tag{7–36}$$

$$a = \tilde{Z}\left(\frac{Z(1-\tilde{Z})}{\widetilde{Z''^2}} - 1\right), b = (1-\tilde{Z})\left(\frac{\tilde{Z}(1-\tilde{Z})}{\widetilde{Z''^2}} - 1\right) \tag{7–37}$$

The Beta-PDF is used for the Flamelet libraries shipped with CFX, and for libraries created with CFX-RIF. For details, see CFX-RIF in the *CFX-Solver Modeling Guide*.

Note that for the table look-up, the solver is actually using χ instead of χ_{st}, i.e., the local value of the scalar dissipation rate is applied instead of the value at stoichiometric mixture fraction. This is exact only for stoichiometric mixture or for vanishing variance of mixture fraction (perfectly premixed case). However, many radicals of interest, for example, OH radicals, have significant concentrations only at stoichiometry and in its surrounding, which makes this approximation acceptable. In principle, χ_{st} could be derived from the solution fields, but this would introduce errors, too, because it would require either additional modeling or averaging over the computational domain.

7.8.3. Flamelet Libraries

A flamelet library provides the mean species mass fractions as functions of mean mixture fraction, variance of mixture fraction and scalar dissipation rate:

$$\tilde{Y}_i = \tilde{Y}_i\left(\tilde{Z}, \widetilde{Z''^2}, \tilde{\chi}_{st}\right) \tag{7–38}$$

A separate Flamelet library is required for each fuel and each combination of fuel/oxidizer inlet temperatures and pressure level. The libraries shipped with CFX are listed in *Table 7.1: Flamelet libraries included with CFX* (p. 316). The files reside in the subdirectory: `<CFXROOT>/etc/reactions-extra/flamelet/`

where <CFXROOT> is the location of your CFX installation.

Table 7.1 Flamelet libraries included with CFX

Library file name	**Fuel/Oxidizer**	T_{Fuel}	T_{Oxid}	**Pressure**
H2_Air_Tf300K_Tox300K_p1bar.fll	hydrogen/air	298 [K]	298 [K]	1 [bar]
CH4_Air_Tf298K_Tox298K_p1bar.fll	Methane/air	300 [K]	300 [K]	1 [bar]

If more libraries are needed, these can be created with CFX-RIF. For details, see CFX-RIF in the *CFX-Solver Modeling Guide*.

Note

The flamelet libraries shipped with ANSYS CFX are provided to illustrate the flamelet model and the partially premixed model capabilities. Flamelet libraries calculated using CFX-RIF may consider a larger number of components and a more detailed reaction mechanism. As a result, the calculated mixture composition based on flamelet libraries calculated using CFX-RIF may be more accurate.

7.9. Burning Velocity Model (Premixed or Partially Premixed)

The model for premixed or partially premixed combustion can be split into two independent parts:

- Model for the progress of the global reaction: Burning Velocity Model (BVM), also called Turbulent Flame Closure (TFC)
- Model for the composition of the reacted and non-reacted fractions of the fluid: Laminar Flamelet with PDF

The mass fractions in the non-reacted fraction of the fluid, $Y_{i,\mathrm{fresh}}$, are obtained by linear blending of fuel and oxidizer compositions. The species mass fractions in the burned fraction of the fluid, $Y_{i,\mathrm{burned}}$, are computed by applying the Flamelet model.

7.9.1. Reaction Progress

A single progress variable, c, is used to describe the progress of the global reaction:

$$\mathrm{Fuel} + \mathrm{Oxidizer} \rightarrow \mathrm{Products} \tag{7–39}$$

The composition of the fluid is determined by blending the compositions of the non-reacted state (fresh gases) and the reacted state (burned gases), where $c = 0$ corresponds to fresh materials and $c = 1$corresponds to fully reacted materials.

In turbulent flow, a bimodal distribution of c is assumed. At any given time and position in space the fluid is considered to be either fresh materials or fully reacted. This assumption is justified if the chemical reaction is fast compared to the integral turbulent time scales of the flow.

Then, the averaged reaction progress variable, $\tilde{c}$, is the probability for the instantaneous state of the fluid being reacted. The mean species composition of the fluid is computed according to:

$$\tilde{Y}_i = (1 - \tilde{c})\ \tilde{Y}_{i,\text{fresh}} + \tilde{c}\ \tilde{Y}_{i,\text{burned}} \tag{7–40}$$

For example, if $\tilde{c} = 0.6$, then the fluid at the given position will be fully reacted during 60% of time and non-reacted and non-reacted during the remaining 40% of time.

The reaction progress variable is computed by solving a transport equation:

$$\frac{\partial(\bar{\rho}\,\tilde{c})}{\partial t} + \frac{\partial\left(\bar{\rho}\,\tilde{u}_j\,\tilde{c}\right)}{\partial x_j} = \frac{\partial}{\partial x_j}\left[\left(\overline{\rho D} + \frac{\mu_t}{\sigma_c}\right)\frac{\partial \tilde{c}}{\partial x_j}\right] + \overline{\omega_c} \tag{7–41}$$

The default value of the turbulent Schmidt number σ_c for the reaction progress variable is $\sigma_c = 0.9$.

In the limits of pure fuel and pure oxidizer, the reaction progress is not well defined because burnt and unburnt conditions correspond to the same physical state:

$$\tilde{Y}_{\text{i,burned}}(\tilde{Z}=0) = \tilde{Y}_{\text{i,fresh}}(\tilde{Z}=0) \tag{7–42}$$

$$\tilde{Y}_{\text{i,burned}}(\tilde{Z}=1) = \tilde{Y}_{\text{i,fresh}}(\tilde{Z}=1) \tag{7–43}$$

This poses the issue of which boundary value to specify for $\tilde{c}$ when the mixture is either pure fuel or pure oxidizer. Even though different values for $\tilde{c}$ correspond to the same mixture composition, it is still important to impose the proper value on the boundary, because it controls the combustion regime in the domain after mixing has occurred:

- $\tilde{c} = 0$ corresponds with premixed combustion
- $\tilde{c} = 1$ corresponds with non-premixed combustion (diffusion flame)

In most cases, $\tilde{c} = 0$ is appropriate for fuel inlets (in fact, if $\tilde{c} = 1$ for fuel the Flamelet model could be used and not solve for $\tilde{c}$ at all). For oxidizer inlets, the appropriate boundary value depends on the mixing process in the domain:

- $\tilde{c} = 0$ for oxidizer premixing with fuel (e.g., primary air)
- $\tilde{c} = 1$ for oxidizer mixing with products (e.g., secondary air)

However, which case applies may not be known prior to the simulation. In case of a flow split, it may even be that both cases occur for the same oxidizer inlet. In this situation, the artificial distinction between "burnt air" ($\tilde{c} = 1$) and "fresh air" ($\tilde{c} = 0$) may cause unphysical behavior such as even fuel being generated by mixing of products with fresh air. For example, mixing "fresh air" ($\tilde{c} = 0$, $\tilde{Z} = 0$) with products ($\tilde{c} = 1$, $\tilde{Z} = Z_{st}$) would result in a mixture that is only partially burnt ($\tilde{c} = \alpha$, $\tilde{Z} = \alpha \cdot Z_{st}$). The reaction progress of the resulting mixture equals the fraction of products $\alpha < 1$, equivalent to fuel being re-established. This is unphysical. The correct behavior would be obtained by mixing products with “burnt air” ($\tilde{c} = 1, \tilde{Z} = 0$).

7.9.2. Weighted Reaction Progress

In order to overcome the boundary value issue for reaction progress on oxidizer inlets, the weighted reaction progress, F, is introduced:

$$\tilde{F} = \tilde{Z} \cdot (1 - \tilde{c}) \tag{7–44}$$

Because $\tilde{Z} = 0$ for pure oxidizer, the weighted reaction progress is well defined to be $\tilde{F} = 0$. Linear combination of the transport equations for $\tilde{Z}$ and $\tilde{c}$ yields the following transport equation for the weighted reaction progress:

$$\begin{aligned} &\frac{\partial (\bar{\rho} \tilde{F})}{\partial t} + \frac{\partial \left(\bar{\rho} \tilde{u}_j \tilde{F}\right)}{\partial x_j} \\ &= \frac{\partial}{\partial x_j} \left[\left(\overline{\rho D} + \frac{\mu_t}{\sigma_F} \right) \frac{\partial \tilde{F}}{\partial x_j} \right] + 2 \left(\overline{\rho D} + \frac{\mu_t}{\sigma_F} \right) \left(\frac{\partial \tilde{Z}}{\partial x_j} \cdot \frac{\partial \tilde{c}}{\partial x_j} \right) - \tilde{Z}\, \overline{\omega_c} \end{aligned} \tag{7–45}$$

The turbulent Schmidt number for the weighted reaction progress by default is $\sigma_F = 0.9$. Because of the formal derivation of the transport equation, its solution will be equivalent to solving for reaction progress directly. In other words, introducing the weighted reaction progress overcomes the boundary value issue for reaction progress without changing the model. Reaction progress can be restored from mixture fraction and the weighted reaction progress according to

$$\tilde{c} = (\tilde{Z} - \tilde{F}) / \tilde{Z} \tag{7–46}$$

The recipe for $\tilde{c}$ is ill-posed for $\tilde{Z} = 0$. Therefore, numerical treatment is implemented in the limit of pure oxidizer to smoothly blend both the reaction progress and its gradient to zero:

$$\tilde{c} \rightarrow 0, \frac{\partial \tilde{c}}{\partial x_j} \rightarrow 0 \text{ as } \tilde{Z} \rightarrow 0$$

Default for ANSYS CFX 14.0 is to solve for weighted reaction progress. Expert control (CCL) is available for reverting to the standard reaction progress equation:

```
FLUID MODELS:
  Option = Burning Velocity Model
  REACTION PROGRESS VARIABLE:
    Option = Reaction Progress
    # default: Weighted Reaction Progress
  END
END
```

7.10. Burning Velocity Model (BVM)

The burning velocity model (BVM), also known as turbulent flame closure (TFC), is used to close the combustion source term for reaction progress.

$$\overline{\omega_{\mathrm{c}}} = \overline{S_{\mathrm{c}}} - \frac{\partial}{\partial x_{\mathrm{j}}}\left((\overline{\rho D}) \frac{\partial \tilde{c}}{\partial x_{\mathrm{j}}} \right) \tag{7–47}$$

$$\overline{S_{\mathrm{c}}} = \overline{\rho_{\mathrm{u}}}\, S_{\mathrm{T}}\, |\nabla \tilde{c}| \tag{7–48}$$

Where $\overline{\rho_{\mathrm{u}}}$ is the density of the unburnt mixture. Note that the molecular diffusion term for the reaction progress is removed from the transport equation. The diffusive exchange of species and energy, which makes the flame proceed in space, is already accounted for by the source term $\overline{S_{\mathrm{c}}}$. However, turbulent transport is a convective process and is modeled using the Eddy Diffusivity approximation.

The model is completed with a closure for the turbulent burning velocity s_T. Accordingly, this type of model is called Turbulent Burning Velocity Model (BVM). The concept has two significant advantages compared to models based on molecular reaction rates:

1. In a given configuration, s_T typically varies by only 1 order of magnitude. In contrast, molecular reaction rates occurring in combustion of hydro-carbonates typically vary in the computational domain by several orders of magnitude.
2. s_T can be measured directly in experiments, i.e., data is available for the quantity that is modeled.

Further, the burning velocity directly determines target quantities of a simulation, such as flame position. Thus, it is easier to derive and fine-tune accurate models for the burning velocity model than to do so for approaches based on molecular reaction rates.

7.10.1. Equivalence Ratio, Stoichiometric Mixture Fraction

Referring to the unburnt mixture, the equivalence ratio φ describes the ratio of fuel relative to the amount of fuel that potentially could be burnt with the available oxidizer. For stoichiometric mixture, the equivalence ratio is defined to be $\varphi = 1$, i.e., the amount of fuel and oxidizer match such that they could be burnt with neither fuel nor oxidizer left behind. $\varphi > 1$ indicates fuel-rich mixtures (excess fuel), and $\varphi < 1$ indicates fuel-lean mixtures (excess oxidizer). The limits are $\varphi = 0$ for pure oxidizer and $\varphi = \infty$ (infinite) for pure fuel.

When the stoichiometric mixture fraction Z_{st} is known, the local equivalence ratio can be computed from mixture fraction Z according to

$$\varphi = \frac{Z}{1 - Z} \cdot \frac{1 - Z_{\mathrm{st}}}{Z_{\mathrm{st}}} \tag{7–49}$$

The stoichiometric mixture fraction Z_{st} depends on the fuel and the oxygen content in the oxidizer and is a property of the flamelet library.

- Value

  This option enables you to specify directly the stoichiometric mixture fraction used for calculating the equivalence ratio.

- Reactants

  For this option, the reactants and their stoichiometric coefficients for a representative global reaction are specified. For example, for a single component fuel,

$$v_{\mathrm{F}}\ \mathrm{Fuel} + v_{\mathrm{Ox}}\ \mathrm{O}_2 \rightarrow (\mathrm{products}) \tag{7–50}$$

  or for the generic form,

$$\sum_{\alpha} v_{\alpha} X_{\alpha} \rightarrow (\mathrm{products}) \tag{7–51}$$

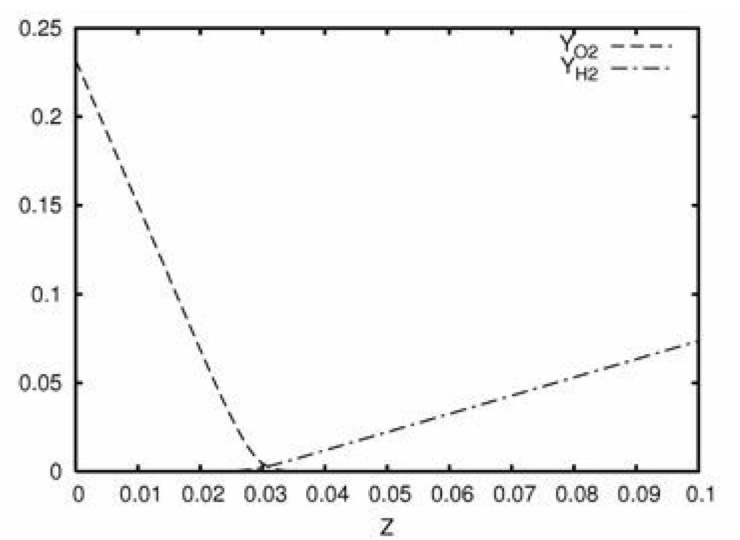
  The stoichiometric mixture fraction is computed using the reactants stoichiometric coefficients and the corresponding species mass fractions in the fuel and oxidizer streams, respectively. The species mass fractions in the fuel and in the oxidizer are obtained from the flamelet library.

- Automatic

  This option derives the stoichiometric mixture fraction from the flamelet library requiring no additional information. The numerical procedure is described below.

Figure 1 shows the qualitative behavior of mass fractions for fuel and oxygen plotted over mixture fraction (hydrogen/air in the example).

Figure 7.1 Oxygen Mass Fraction Over Mixture Fraction



One can observe that the oxygen concentration is approximately linear below and above the stoichiometric mixture fraction: linear decay on the lean side and constantly zero on the fuel side. Obviously, the curvature of the curve is close to zero except near the sharp bend at stoichiometric mixture fraction. This observation is generalized to establish the following procedure

The stoichiometric mixture fraction approximated by the point of maximum curvature for oxygen mass fraction, or Z_c, is

$$Z_{st} \approx Z_c, \quad \frac{\partial^2 Y_{O_2}(Z_c)}{\partial Z^2} = \max_{0 \le Z \le 1} \left[\frac{\partial^2 Y_{O_2}(Z)}{\partial Z^2} \right] \quad \textbf{(7–52)}$$

This is a heuristic approach and only provides an approximation. It is recommended to check for plausibility of the calculated value, which is reported to the solver output file. For details, see CFX-Solver Output File (Combustion Runs) in the *CFX-Solver Manager User's Guide*.

7.11. Laminar Burning Velocity

The laminar burning velocity, s_L, is a property of the combustible mixture. It is defined as the speed of the flame front relative to the fluid on the unburnt side of the flame. The burning velocity relative to the burnt fluid will be higher by a factor equal to the expansion ratio, $s_L^{burnt} = s_L \cdot \rho_u / \rho_b$.

Physically, the laminar burning velocity depends on the fuel, the equivalence ratio, the temperature of the unburnt mixture (preheating) and on pressure. Depending on the configuration in the simulation, it may be possible sometimes to neglect preheat and pressure dependencies. However, for partially premixed combustion, it is very important to account for the dependency on equivalence ratio. Specifically, the flammability limits have to be obeyed.

7.11.1. Value

User-defined laminar burning velocity, s_L. It is recommended to account for dependency on fuel/oxidizer ratio by making the expression depend on equivalence ratio or mixture fraction. Properties of the unburnt mixture (temperature, density, specific heat capacity, thermal conductivity) may be used in the expression in order to account for preheating or mixture dependency.

7.11.2. Equivalence Ratio Correlation

The equivalence ratio correlation follows the approach by Metghalchi and Keck [124], expressing the laminar burning velocity as a base value at reference conditions, $s_{L,0}$, multiplied by correction factors for preheat and pressure dependencies:

$$s_L = s_L^0 \cdot \left(\frac{T_u}{T_{ref}} \right)^{\alpha} \cdot \left(\frac{p}{p_{ref}} \right)^{\beta} \quad \textbf{(7–53)}$$

The exponents for preheat dependency and for pressure dependency are quadratic polynomials in equivalence ratio:

$$\alpha = a_0 + a_1 \varphi + a_2 \varphi^2 \tag{7–54}$$

$$\beta = b_0 + b_1 \varphi + b_2 \varphi^2 \tag{7–55}$$

When all three coefficients are set to zero, then the preheat dependency or the pressure dependency is disabled. For reference burning velocity, the following options are available:

- *Fifth Order Polynomial* (p. 322)
- *Quadratic Decay* (p. 322)
- *Beta Function* (p. 322)

All three options specify the flammability limits for fuel-lean and for fuel-rich mixtures, $\varphi_{\mathrm{flam,l}}$ and $\varphi_{\mathrm{flam,r}}$. The burning velocity is set to zero if the local equivalence ratio is out of these bounds.

7.11.2.1. Fifth Order Polynomial

The laminar burning velocity may be specified as a polynomial up to fifth order:

$$s_{\mathrm{L}}^{0} = s_0 \varphi + s_1 \varphi + s_2 \varphi^2 + s_3 \varphi^3 + s_4 \varphi^4 + s_5 \varphi^5 \tag{7–56}$$

This polynomial is evaluated on a specified fit range, $\varphi_{\mathrm{fit,l}} \le \varphi \le \varphi_{\mathrm{fit,r}}$. Outside this range, the burning velocity is modeled to linearly decay to zero at the flammability limit.

7.11.2.2. Quadratic Decay

For quadratic decay the maximum laminar burning velocity at reference conditions, s_{max}^{0}, and the corresponding equivalence ratio, φ_{max}, are given. For smaller or larger equivalence ratio the burning velocity is modeled to decrease according to a quadratic decay coefficient, C_{decay}:

$$s_{\mathrm{L}}^{0} = s_{0}^{\mathrm{max}} - C_{\mathrm{decay}} \left(\varphi - \varphi_{\mathrm{max}} \right)^2 \tag{7–57}$$

This quadratic function is evaluated on a specified fit range, $\varphi_{\mathrm{fit,l}} \le \varphi \le \varphi_{\mathrm{fit,r}}$. Outside this range, the burning velocity is modeled to linearly decay to zero at the flammability limit.

7.11.2.3. Beta Function

The beta function correlation sets the maximum laminar burning velocity, s_{0}^{max}, and the corresponding equivalence ratio, φ_{max}. A beta function is used to model the velocity decay to zero at the fuel-lean or fuel-rich flammability limit:

$$s_{\mathrm{L}}^{0} = s_{0}^{\mathrm{max}} \cdot \left(\frac{\varphi - \varphi_{\mathrm{flam,l}}}{\varphi_{\mathrm{max}} - \varphi_{\mathrm{flam,l}}} \right)^{a} \cdot \left(\frac{\varphi_{\mathrm{flam,r}} - \varphi}{\varphi_{\mathrm{flam,r}} - \varphi_{\mathrm{max}}} \right)^{b}$$

where $a = 2 \cdot \frac{\varphi_{\mathrm{max}} - \varphi_{\mathrm{flam,l}}}{\varphi_{\mathrm{flam,r}} - \varphi_{\mathrm{flam,l}}}$ and $b = 2 \cdot \frac{\varphi_{\mathrm{flam,r}} - \varphi_{\mathrm{max}}}{\varphi_{\mathrm{flam,r}} - \varphi_{\mathrm{flam,l}}}$.

7.11.2.4. Residual Material Dependency

In order to account for the residual material, an optional correction factor is multiplied to the laminar burning velocity:

$$s_L = s_L^0 \cdot \left(\frac{T_u}{T_{\mathrm{ref}}} \right)^{\alpha} \cdot \left(\frac{p}{p_{\mathrm{ref}}} \right)^{\beta} \cdot (1 - \gamma X_{EGR}) \qquad \textbf{(7–58)}$$

where X_{EGR} denotes the molar fraction (volume fraction) of the residual material and γ is a coefficient function of the equivalence ratio:

$$\gamma = c_0 + c_1 \varphi + c_2 \varphi^2 \qquad \textbf{(7–59)}$$

In combination with the residual material model, the values ofα, β and γ are calculated using the equivalence ratio $\varphi \,|_{fresh}$ conditioned to the ‘fresh’ mixture without residual material.

The residual material dependency in the laminar burning velocity correlation is optional. The default residual material dependency coefficient γ is constant zero, that is, no residual material dependency.

7.11.2.5. Metghalchi and Keck

The correlation by Metghalchi and Keck is based on the equivalence ratio correlation with quadratic decay described above. Predefined sets of coefficients are provided for several hydrocarbon fuels. The fuel type is characterized by the number of carbon atoms in the fuel molecule, here called the fuel carbon index. *Table 7.2: Fuel Dependent Coefficients for Metghalchi and Keck Laminar Burning Velocity Correlation* (p. 323) lists the coefficients for methane, propane and iso-octane (gasoline), respectively.

Table 7.2 Fuel Dependent Coefficients for Metghalchi and Keck Laminar Burning Velocity Correlation

Carbon Index	Fuel	s_0^{max} [m/s]	C_{decay} [m/s]	φ_{max}	$\varphi_{\mathrm{flam,l}}$	$\varphi_{\mathrm{flam,r}}$
1	Methane	0.35	1.387	1.06	0.533	1.68
3	Propane	0.342	1.387	1.08	0.536	2.50
8	Iso-octane	0.263	0.847	1.13	0.601	3.80

For fuels with other carbon indices the coefficients are obtained by linear interpolation of the provided values. Fit range, preheat dependency and pressure dependency are modeled independent of the fuel (*Table 7.3: Common Coefficients for Metghalchi and Keck Laminar Burning Velocity Correlation* (p. 324)).

Table 7.3 Common Coefficients for Metghalchi and Keck Laminar Burning Velocity Correlation

$\varphi_{\mathrm{fit,l}}$	$\varphi_{\mathrm{fit,r}}$	a_0	a_1	a_2	b_0	b_1	b_2
0.7	1.4	2.98	-0.8	0	-0.38	0.22	0

7.12. Turbulent Burning Velocity

For turbulent flow, the effective or turbulent burning velocity s_{T} will differ from the laminar burning velocity, s_{L}. Typically turbulence will increase the burning velocity, because wrinkling of the flame front results in an increased effective flame surface. At very high turbulence, the opposite effect may occur, leading to a decrease in the effective burning velocity because of local extinction. A model is required to describe the turbulent burning velocity as a function of laminar burning velocity and turbulence quantities.

The burning velocity is defined relative to the unburnt fluid. Relative to the burnt fluid, it will be higher by a factor equal to the fluid expansion ratio, $s_{\mathrm{T}}^{\mathrm{burnt}} = s_{\mathrm{T}} \cdot \rho_{\mathrm{u}} / \rho_{\mathrm{b}}$.

7.12.1. Value

This option can be used to implement user models for turbulent burning velocity. Typically, this will include expressions using laminar burning velocity and turbulence quantities.

7.12.2. Zimont Correlation

The closure developed by Zimont et al. [38] [40] [41] is used for the turbulent burning velocity:

$$s_T = A\, G\, u'^{3/4}\, s_L^{1/2}\, \lambda_u^{-1/4}\, l_t^{1/4} \tag{7–60}$$

The leading factor, A, is a modeling coefficient that has the universal value $A = 0.5$ (default), with the exception of H2/Air flames where $A = 0.6$ is recommended [41].

The stretching factor, G, accounts for reduction of the flame velocity due to large strain rate (large dissipation rate of the turbulent kinetic energy). This effect is modeled in terms of the probability for turbulence eddy dissipation, ε, being larger than a critical value $\varepsilon_{\mathrm{cr}}$. For $\varepsilon > \varepsilon_{\mathrm{cr}}$, flamelet extinction takes place, while for $\varepsilon < \varepsilon_{\mathrm{cr}}$, the stretching effect is ignored completely. Assuming a lognormal distribution for ε, the stretching factor is given by:

$$G = \frac{1}{2} \operatorname{erfc}\left[-\frac{1}{\sqrt{2}\,\sigma} \left(\ln\left(\varepsilon_{\mathrm{cr}} / \tilde{\varepsilon}\right) + \frac{\sigma}{2} \right) \right] \tag{7–61}$$

where erfc denotes the complimentary error function and $\sigma = \mu_{\mathrm{str}} \ln\left(l_t / \eta \right)$ is the standard deviation of the distribution of ε, with μ_{str} being an empirical model coefficient (default $\mu_{\mathrm{str}} = 0.28$).

λ_u is the thermal conductivity of the unburned mixture. The turbulent flame speed closure model is completed with the following models for integral velocity fluctuations level:

$$u' = \sqrt{\frac{2}{3} k} \quad \textbf{(7–62)}$$

integral turbulent length scale:

$$l_t = k^{3/2}/\varepsilon \quad \textbf{(7–63)}$$

and Kolmogorov length scale.

$$\eta = v^{3/4}/\varepsilon^{1/4} \quad \textbf{(7–64)}$$

The critical dissipation rate, $\varepsilon_{\mathrm{cr}}$, is computed from a specified critical velocity gradient, g_{cr}, and the kinematic viscosity of the fluid, v, according to:

$$\varepsilon_{\mathrm{cr}} = 15\, v\, g_{cr}^2 \quad \textbf{(7–65)}$$

For steady laminar flow the critical velocity gradient for quenching, g_{cr}, can be obtained numerically. However, for turbulent flows, the critical value must be larger than in laminar cases because the smallest turbulent eddies, which are responsible for the largest strain rates, do not persist long enough to quench a flame front locally. Furthermore, different model problems may result in significant variation of the critical values obtained. For these reasons, the quenching critical velocity gradient has to be tuned for industrial simulations. In fact, it is the only significant parameter for tuning the model.

Theory or numerical modeling can suggest a range of physically plausible values of .g_{cr}. For example, the inverse of the chemical time scale of the reaction, τ_{ch}, scaled by a factor in the range 0.1 to 1.0 is a reasonable starting point. For gas turbine combustion chambers (burning a lean methane/air mixture) values in the range:

$$g_{\mathrm{cr}} = 6000 \left[s^{-1} \right] \text{to}\, 10000 \left[s^{-1} \right]$$

depending on the configuration, have been used successfully [40] [41] It should be noted that these recommended values are for atmospheric temperature and pressure.

Table 7.4 Default Model Coefficients for the Zimont Turbulent Burning Velocity Correlation

Parameter	Default Value
A	0.5
μ_{str}	0.28
g_{cr}	10000 s^{-1}

7.12.3. Peters Correlation

The Peters correlation for turbulent burning velocity was originally developed for the G-equation combustion model (N. Peters [37]). For CFX, it was adopted to be used with the burning velocity model.

$$s_T = s_L\,(1 + \sigma_t) \tag{7–66}$$

where $\sigma_t = -A\frac{l}{l_F} + \sqrt{\left(A\frac{l}{l_F}\right)^2 + B\frac{u'l}{s_L l_F}}$, $u' = \sqrt{\frac{2}{3}k}$, $A = \frac{a_4 b_3^2}{2 b_1}$, and $B = a_4 b_3^2$.

Turbulence effects are modeled as a function of the ratio between the integral turbulence length scale, l, and the laminar flame thickness, l_F.

$$l = a_1 \frac{u'^3}{\varepsilon} \tag{7–67}$$

$$l_F = \frac{D}{s_L} = \frac{\left(\lambda / c_p\right)_u}{\rho_u\, s_L} \tag{7–68}$$

The diffusion coefficient for the flame is computed from molecular properties of the unburnt mixture indicated by the subscript u. The default values for the model constants are listed in *Table 7.5: Default Model Coefficients for Peters Turbulent Burning Velocity Correlation* (p. 326).

Table 7.5 Default Model Coefficients for Peters Turbulent Burning Velocity Correlation

Coefficient	Default Value
a_1	0.37
a_4	0.78
b_1	2.0
b_3	1.0

7.12.4. Mueller Correlation

The correlation by Mueller et al. follows an approach similar to the Peters correlation. It accounts for variation of the average laminar burning velocity due to turbulent fluctuations of mixture fraction and accounts for extinction under high scalar dissipation rates. However, the Mueller correlation uses a different correlation for turbulence dependency:

$$s_T = s_L \cdot (1 + \sigma_t) \cdot \left(1 - \alpha\,\frac{\overline{\chi}}{\chi_q}\right), \sigma_t = b_1\,\frac{u'}{s_{L,st}} + b_2\sqrt{\frac{u'}{s_{L,st}}}, u' = \sqrt{\frac{2}{3}k}$$

Where $s_{\mathrm{L,st}} = s_{\mathrm{L}}(Z_{\mathrm{st}})$ is the laminar burning velocity at stoichiometric mixture fraction (equivalence ratio $\varphi = 1$). The laminar burning velocity is integrated over a presumed PDF accounting for fluctuations of mixture fraction.

$$\overline{s_{\mathrm{L}}} = \int_0^1 \left(P(Z) \cdot s_{\mathrm{L}}(Z) \right) dZ \qquad \textbf{(7–69)}$$

The presumed PDF for mixture fraction is described in *Coupling of Laminar Flamelet with the Turbulent Flow Field* (p. 314). *Table 7.6: Default Model Coefficients for Mueller Turbulent Burning Velocity Correlation* (p. 327) lists the default values for the model coefficients.

Table 7.6 Default Model Coefficients for Mueller Turbulent Burning Velocity Correlation

Coefficient	Default Value
b_1	1.5
b_4	0.8
χ_{q}	1.0 [1/s]
α	1.0

7.13. Extended Coherent Flame Model (ECFM)

The Extended Coherent Flame Model (ECFM) shares the framework for premixed or partially premixed combustion with the Burning Velocity Model (BVM). Transport equations are solved for mean mixture fraction (*Equation 7–32* (p. 314)), variance of mixture fraction (*Equation 7–33* (p. 314)) and for either reaction progress (*Equation 7–41* (p. 317)) or weighted reaction progress (*Equation 7–45* (p. 318)).

In order to describe the intensity and the location of the reaction zone the flame surface density Σ is introduced, which is defined as the area of flame surface per unit volume. The physical units are that of an inverse length (for example, [1/m]). The reaction source in *Equation 7–41* (p. 317)or *Equation 7–45* (p. 318) is defined in terms of flame surface density as:

$$\overline{\omega_c} = \overline{\rho}^{u} s_L \Sigma \qquad \textbf{(7–70)}$$

The flame surface density is calculated by solving a transport equation:

$$\frac{\partial \Sigma}{\partial t} + \frac{\partial}{\partial x_j}\left(\tilde{u}_j \Sigma\right) = \frac{\partial}{\partial x_j}\left[\left(\overline{\rho D} + \frac{\mu_t}{\sigma_\Sigma}\right)\frac{\partial\left(\Sigma / \overline{\rho}\right)}{\partial x_j}\right] + \left(P_1 + P_2 + P_3\right)\Sigma - D \qquad \textbf{(7–71)}$$

The source terms and their physical meanings are as follows:

- Flame surface production by turbulent stretch

$$P_1 = \alpha K_t \tag{7–72}$$

- Flame surface production by mean flow dilatation

$$P_2 = \frac{2}{3} \frac{\partial \tilde{u}_j}{\partial x_j} \tag{7–73}$$

- Thermal expansion and curvature

$$P_3 = \frac{2}{3} \frac{\overline{\rho}^u}{\overline{\rho}^b} s_L \frac{1 - \overline{c}}{\overline{c}} \Sigma \tag{7–74}$$

- Destruction due to consumption

$$D = \beta s_L \frac{\Sigma^2}{1 - \overline{c}} \tag{7–75}$$

where the typical (default) values for α and β are 1.6 and 1.0, respectively.

Note that the P_3 production term and the destruction term are formulated in terms of the volumetric reaction progress $\overline{c}$, in contrast to the specific reaction progress $\tilde{c}$ (Favre-average):

$$\overline{c} = \frac{\overline{\rho}^u}{\overline{\rho}^b} \cdot \frac{\tilde{c}}{\frac{\overline{\rho}^u}{\rho^b} \cdot \tilde{c} + (1 - \tilde{c})} \tag{7–76}$$

The flame surface density Σ is a volumetric quantity, i.e. flame surface area per unit volume. In order to be able to use the generic machinery for solving transport the specific flame surface density, $\tilde{S}$, is introduced as an auxiliary variable:

$$\Sigma = \overline{\rho} \tilde{S} \tag{7–77}$$

which can be rewritten as

$$\tilde{S} = \frac{\Sigma}{\overline{\rho}} \tag{7–78}$$

The specific flame surface density is the flame surface area per unit mass. Rewriting Equation 7–71 (p. 327)in terms of $\tilde{S}$ leads to:

$$\frac{\partial\,(\bar{\rho}\tilde{S})}{\partial t}+\frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_j\tilde{S}\right)=\frac{\partial}{\partial x_j}\left[\left(\overline{\rho D}+\frac{\mu_t}{\sigma_\Sigma}\right)\frac{\partial\tilde{S}}{\partial x_j}\right]+(P_1+P_2+P_3)\,\Sigma-D \qquad \textbf{(7–79)}$$

The volumetric transport *Equation 7–71* (p. 327) and the specific transport *Equation 7–79* (p. 329) are equivalent. The specific form has the advantage that it has the same structure as other transport equations, which are also in specific form, and therefore *Equation 7–79* (p. 329) may be solved using the same discretization schemes and numerical algorithms. The flame surface density Σ becomes a dependent variable and is calculated according to *Equation 7–77* (p. 328).

7.13.1. Turbulent Flame Stretch

The turbulent flame stretch accounts for the increase of flame surface area due to stretching and wrinkling of the flame front under turbulence. For the Extended Coherent Flame Model this effect is described by the Intermittent Turbulent Net Flame Stretch (ITNFS) model proposed by Meneveau and Poinsot [186]:

$$K_t=\alpha_0\Gamma_K\left(\left(\frac{u'}{s_L^0}\right),\left(\frac{l_t}{\delta_L^0}\right)\right)\frac{\varepsilon}{k} \qquad \textbf{(7–80)}$$

where s_L^0 is the laminar flame speed, δ_L^0 is the laminar flame thickness, $u'=\sqrt{(2/3)\,k}$ is the mean turbulent velocity fluctuation and $l_t=k^{3/2}/\varepsilon$ is the integral length scale of turbulence, and,

$$\alpha_0=1.0$$

$$\log_{10}(\Gamma_K)=-\frac{1}{(s+0.4)}\cdot\exp(-(s+0.4))+(1-\exp(-(s+0.4)))\left(\sigma_1\left(\frac{u'}{s_L^0}\right)s-0.11\right)$$

$$s=\log_{10}\left(\frac{l_t}{\delta_L^0}\right)$$

$$\sigma_1\left(\frac{u'}{s_L^0}\right)=\frac{2}{3}\left(1-\frac{1}{2}\exp\left(-\left(\frac{u'}{s_L^0}\right)^{1/3}\right)\right)$$

7.13.2. Laminar Flame Thickness

The thickness of the laminar flame has to be provided in order to close the ITNFS model. For ideal gases Blint's correlation is a reasonable approximation:

$$\delta_L^0 = \delta \cdot C_B \left(\frac{T^b}{T^u} \right)^\beta = \frac{\lambda^u}{\rho^u C_p^u s_L} \cdot C_B \left(\frac{T^b}{T^u} \right)^\beta \quad \textbf{(7–81)}$$

Where C_B is the **Thickness Multiplier** with a default value of 2.0, β is the **Temperature Exponent** with a default value of 0.7 and δ is the 'diffusive' flame thickness, which is scaled by a correction factor accounting for the thermal expansion ratio in order to obtain the effective laminar flame thickness. Additional information on laminar flame thickness calculation can be found in the literature, for example in [187].

7.13.3. Wall Quenching Model

Quenching of the flame at walls is modeled by a simple wall flux model [182]. The model assumes that the boundary layer for the flame surface density Σ is thin such that the flame is unaffected by the wall until it contacts it. When a flamelet is touching the wall it will be quenched, resulting in a destruction of flame surface density at the wall.

The probability of a flamelet touching the wall is proportional to the fluctuation of the velocity component normal to the wall, u', multiplied by a factor $1/2$ in order to account for a 50% chance for the fluctuation being directed towards the wall. This picture leads to a wall transfer coefficient equal to $u'/2$, or the equivalent wall flux for flame surface density:

$$F_{\Sigma,wall} = -\frac{u'}{2} \Sigma_{nw} \quad \textbf{(7–82)}$$

where Σ_{nw} is the flame surface density at the internal near-wall boundary element center node. The sign of the flux is defined to be positive for flux in, and negative for flux going out.

7.14. Residual Material Model

The standard mixture fraction/reaction progress model assumes a two-stream mixing process of fuel and oxidizer. The residual material model extends this framework by mixing with a third stream, the residual or ballast material. The global mixing and reaction scheme is:

$$\text{Fuel} + \text{Oxidizer} + \text{Residual} \rightarrow \text{Products} + \text{Residual} \quad \textbf{(7–83)}$$

The residual material does not participate in the combustion directly, but it does interact with the mixture and the propagation of the flame. The most significant effects are:

- Lower peak temperature (dilution)
- Slow down in laminar burning velocity
- Different mixture composition in chemical equilibrium because of lower temperature level and chemical interaction of residual with products

For the dependency of the laminar burning velocity on the residual material concentration see also section *Equivalence Ratio Correlation* (p. 321).

7.14.1. Exhaust Gas Recirculation

In principle the residual material could have an arbitrary composition. The **Exhaust Gas** model in ANSYS CFX restricts the residual material to be products originating from the same fuel and oxidizer as for the combustion. Restricting the residual material to exhaust gas offers two advantages:

1. The interaction of the residual material with the products can be modeled using a single-mixture fraction chemistry library (that is, the same flamelet libraries can be used as without residual material).
2. Transformation of newly generated products into residual for simulation of subsequent engine cycles.

For many practical applications it is appropriate to assume that the residuals are recirculated exhaust gas. Exhaust gas recirculation (EGR) may be either externally by guiding the exhaust gas through a duct or pipe into the fresh mixture, or internally by recirculation or incomplete load exchange.

It is still necessary to distinct between products and residuals, because a fresh mixture containing residual material behaves differently than a partially-reacted mixture in the flamefront. The residual material will be less reactive, due to lower temperature and lower radical concentrations, such that no flame develops when merely mixing residuals with fuel and oxidizer.

7.14.2. Principal Variables and Transport Equation

The mixture is described as a three-stream system of fuel, oxidizer and residual material. The corresponding mass fractions neglecting any reaction sources (as for mixture fraction) are fuel tracer, Tr_F, oxidizer tracer, Tr_{Ox}, and residual tracer or EGR tracer, Σ. The tracer variables obey the following obvious constraints, here written for the turbulent means (the same relations apply for the laminar/instantaneous quantities):

$$\tilde{T}r_F + \tilde{T}r_{Ox} + \tilde{T}r_{EGR} = 1 \qquad \textbf{(7–84)}$$

where

$$0 \le \tilde{T}r_F , \tilde{T}r_{Ox} , \tilde{T}r_{EGR} \le 1$$

The fuel tracer and mixture fraction are very closely related quantities. The difference is that the fuel tracer refers to the mass of fresh fuel, while the mixture fraction does additionally include a contribution from the residual materials. The fuel tracer is equal to mixture fraction if, and only if, the residual mass fraction is zero.

Without loss of generality the residual material is defined to be stoichiometric,

$$\tilde{Z}\,|_{EGR} = Z_{st}$$

which establishes the following relation between the overall mixture fraction and the tracer variables:

$$\tilde{Z} = \tilde{T}r_F + \tilde{T}r_{EGR} \cdot Z_{st} \tag{7–85}$$

$$\tilde{T}r_{EGR} = \frac{\tilde{Z} - \tilde{T}r_F}{Z_{st}} \tag{7–86}$$

Using *Equation 7–84* (p. 331) and *Equation 7–85* (p. 332) it is sufficient to solve for two of the four variables $\tilde{Z}$, $\tilde{T}r_F$, $\tilde{T}r_{Ox}$ and $\tilde{T}r_{EGR}$. In addition to the mixture fraction equation discussed in section *The Flamelet Model* (p. 310), a transport equation is solved for the Favre-averaged fuel tracer:

$$\frac{\partial}{\partial}\left(\bar{\rho}\,\tilde{T}r_F\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}\,\tilde{u}_j\,\tilde{T}r_F\right) = \frac{\partial}{\partial x_j}\left[\left(\overline{\rho D} + \frac{\mu_t}{\sigma_Z}\right)\frac{\partial\,\tilde{T}r_F}{\partial x_j}\right] \tag{7–87}$$

The equation is identical to that for the mean mixture fraction, except for boundary and initial conditions.

When the residual material model is applied in combination with the weighted reaction progress model (for details, see *Weighted Reaction Progress* (p. 318)), the reaction progress is calculated based on the fuel tracer instead of mixture fraction. The relation between weighted reaction progress and reaction progress (*Equation 7–44* (p. 318) and *Equation 7–46* (p. 318)) then becomes the following:

$$\tilde{c} = \left(\tilde{T}r_F - \tilde{F}\right) / \tilde{T}r_F \tag{7–88}$$

where

$$\tilde{F} = \tilde{T}r_F \cdot \left(1 - \tilde{c}\right)$$

7.14.3. Mixture Composition

The residual material corresponds to stoichiometric products and its composition can be obtained by lookup in the Flamelet library for stoichiometric conditions. The species mass fractions in the fresh mixture with residual material are computed from mixture fraction and fuel tracer as follows:

$$\tilde{Y}_{i,fresh} = \left(1 - \tilde{T}r_{EGR}\right) \cdot Y_i^{\text{mix}}\left(\tilde{Z}\,|_{fresh}\right) + \tilde{T}r_{EGR} \cdot \tilde{Y}_i^{\text{flamelet}}\left(\tilde{Z}_{st}, \overline{Z''^2} = 0, \bar{\chi} = 0\ [1/\text{s}]\right) \tag{7–89}$$

where $\tilde{Z}\,|_{fresh}$ is the mixture fraction conditional to the combustible fraction of the mixture, that is, the mixture without the residual material, see *Equation 7–94* (p. 333).

The mass fraction in the total mixture is calculated by blending the fresh and burned fractions in the same way as without the residual material model:

$$\tilde{Y}_i = (1-\tilde{c}) \cdot \tilde{Y}_{i,fresh} + \tilde{c} \cdot \tilde{Y}_i^{\text{flamelet}}\left(\tilde{Z}, \overline{Z''^2}, \overline{\chi}\right) \tag{7–90}$$

The species mass fractions for the flamelet solution, $\tilde{Y}_i^{\text{flamelet}}$, and for inert mixing, Y_i^{mix}, are provided by the flamelet library.

7.14.4. Reinitialization for Subsequent Engine Cycles

When running an Internal Combustion Engine simulation with multiple subsequent cycles, the combustion products will become the residual material in the following cycle. Therefore, the 'new' products must be transformed into residual material. This is achieved by reinitializing fuel tracer and reaction progress at specified times:

$$Tr_F^{new} = \tilde{c}^{old} \cdot \max\left(0, \frac{\tilde{Z} - Z_{st}}{1 - Z_{st}}\right) + \left(1 - \tilde{c}^{old}\right) \cdot Tr_F^{old} \tag{7–91}$$

and

$$\tilde{c}^{new} = 0 \tag{7–92}$$

In case of the weighted reaction progress model, the second assignment is equivalent to:

$$\tilde{F}^{new} = \tilde{Z} \tag{7–93}$$

In the expression for the new fuel tracer after reinitialization, Tr_F^{new}, the first term accounts for the excess fuel in the burnt fraction of the mixture, and the second term accounts for the fuel in the fresh fraction.

Note that in the limit of zero scalar dissipation rate, $\overline{\chi} = 0$, the reinitialization leaves the component mass fractions effectively unchanged.

7.14.5. Equivalence Ratio and Conditional Fresh/Residual Mixtures

The `Equivalence Ratio` variable in the results file and in CEL is conditioned to the 'fresh' mixture. It is calculated for the fuel and oxidizer with the residual material removed from the mixture:

$$\tilde{Z}\,|_{fresh} = \frac{\tilde{Tr}_F}{\tilde{Tr}_F + \tilde{Tr}_{Ox}} = \frac{\tilde{Tr}_F}{1 - \tilde{Tr}_{EGR}} \tag{7–94}$$

$$\varphi\,|_{fresh} = \frac{1 - Z_{st}}{Z_{st}} \cdot \frac{\tilde{Z}\,|_{fresh}}{1 - \tilde{Z}\,|_{fresh}} \tag{7–95}$$

Without loss of generality the residual material is assumed to be stoichiometric. Mixture fraction and equivalence ratio conditional to the residual fraction are by definition

$$\tilde{Z}\,|_{resid} = Z_{st}$$

and

$$\varphi\,|_{resid} = 1$$

, respectively. Fuel-rich or fuel-lean residual material can be modeled as a combination of stoichiometric residuals with either excess fuel or excess oxidizer in the fresh mixture. For the purpose of specifying boundary conditions or initial values, the primitive variables can be calculated from the generalized conditional mixture fractions, allowing

$$\tilde{Z}\,|_{resid} \neq Z_{st}$$

according to:

$$\tilde{Z} = (1 - \tilde{Y}_{resid}) \cdot \tilde{Z}\,|_{fresh} + \tilde{Y}_{resid} \cdot \tilde{Z}\,|_{resid} \tag{7–96}$$

$$\varphi\,|_{fresh} = \frac{1 - Z_{st}}{Z_{st}} \cdot \frac{\tilde{Z}\,|_{fresh}}{1 - \tilde{Z}\,|_{fresh}} \tag{7–97}$$

where $\tilde{Y}_{resid}$ denotes the mass fraction of non-stoichiometric residual material, and

$$\tilde{Z}\,|_{fresh}$$

and

$$\tilde{Z}\,|_{resid}$$

are the conditional mixture fractions for the fresh and the residual mixture. Note that

$$\tilde{Y}_{resid} = \tilde{T}r_{EGR}$$

if, and only if,

$$\tilde{Z}\,|_{resid} = Z_{st}$$

. Conditional equivalence ratios can be transformed into conditional mixture fractions and vice versa according to:

$$\tilde{Z}\,|_{fresh} = \frac{Z_{st} \cdot \varphi\,|_{fresh}}{Z_{st} \cdot \varphi\,|_{fresh} + 1 - Z_{st}} \quad (7\text{–}98)$$

$$\tilde{Z}\,|_{resid} = \frac{Z_{st} \cdot \varphi\,|_{resid}}{Z_{st} \cdot \varphi\,|_{resid} + 1 - Z_{st}} \quad (7\text{–}99)$$

$$\varphi\,|_{fresh} = \frac{1 - Z_{st}}{Z_{st}} \cdot \frac{\tilde{Z}\,|_{fresh}}{1 - \tilde{Z}\,|_{fresh}} \quad (7\text{–}100)$$

$$\varphi\,|_{resid} = \frac{1 - Z_{st}}{Z_{st}} \cdot \frac{\tilde{Z}\,|_{resid}}{1 - \tilde{Z}\,|_{resid}} \quad (7\text{–}101)$$

7.15. Spark Ignition Model

The purpose of the spark ignition is two-fold: First, it is required in order to provide the appropriate conditions to start the combustion at time and location of the spark. Second, the initial size of the spark volume may be too small to be resolved by the mesh. Therefore, a model is needed in order to describe the initial growth of the spark kernel

The current model assumes that the burnt region around the spark initially grows as a ball. During this phase the radius of the spark kernel, r_{K}, is computed solving a zero-dimensional initial value problem (IVP). The radius at ignition, t_{ignite}, is defined by the initial spark volume, V_{initial}:

$$\frac{d}{dt} r_{\mathrm{K}}(t) = \frac{\rho_{\mathrm{u}}}{\rho_{\mathrm{b}}} \cdot s_{\mathrm{T,k}} \quad (7\text{–}102)$$

where $r_{\mathrm{K}}\left(t_{\mathrm{ignite}}\right) = \sqrt[3]{\frac{3\,V_{\mathrm{initial}}}{4\,\pi}}$

The spark kernel radius is mapped onto the three-dimensional flow field by averaging the reaction progress over the so-called phantom region. The phantom region is a ball of radius equal to the transition radius, r_{trans}, and center equal to the spark center, $\overline{x}_{\mathrm{spark}}$. While solving for the kernel radius, the reaction progress variable is algebraically set:

$\tilde{c} = \left(\frac{r_{\mathrm{K}}}{r_{\mathrm{trans}}}\right)^3$ for $|\overline{x} - \overline{x}_{\mathrm{spark}}| \le r_{\mathrm{trans}}$

$\tilde{c} = 0$ for $|\overline{x} - \overline{x}_{\mathrm{spark}}| > r_{\mathrm{trans}}$

The initial value problem is solved until the kernel radius reaches the transition radius, r_{trans}, specified by the user. At this point, the IVP solver is stopped and transition to the principal combustion model is made (i.e., switch to the burning velocity model).

The growth rate for the kernel radius is the turbulent burning velocity with a modification accounting for high curvature while the kernel is small:

$$s_{\mathrm{T,k}} = max\left(s_{\mathrm{L}}, s_{\mathrm{T}} - \frac{2}{r_{\mathrm{k}}} D_{\mathrm{t}}\right) \tag{7–103}$$

where $D_{\mathrm{t}} = \frac{C_{\mathrm{u}}}{\sigma_{\mathrm{t}}} \cdot \frac{k^2}{\varepsilon}$

The IVP solver uses quantities averaged over the phantom region for laminar and turbulent burning velocities, s_{L} and s_{T}, turbulence quantities, k and ε, and densities of the burnt and the fresh mixture, ρ_{b} and ρ_{u}.

7.16. Autoignition Model

A flammable mixture residing at sufficiently high temperature will ignite without further interaction after some temporal delay. This phenomenon is called 'autoignition' or 'selfignition'. The underlying process is fuel reforming and buildup of intermediate species and radicals. In the initial stage of combustion this is accompanied by only a small fraction of the total heat release ('low' temperature combustion). When the radicals and intermediates have reached critical concentrations, the chemistry transitions to the 'high' temperature regime with significant heat release.

The time expiring until the transition to the high temperature regime occurs, the ignition delay time, is determined by detailed chemistry with dozens of species and hundreds of reactions involved. For CFD it is impractical to calculate this process in all details. Instead, the progress of fuel reforming and radical buildup is correlated with the elapsed fraction of the delay time, $\tilde{R}$. A transport equation is solved in order to account for variations of the ignition delay time:

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{R}) + \frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_j\tilde{R}\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_t}\right)\frac{\partial \tilde{R}}{\partial x_j}\right] + \frac{\bar{\rho}}{\tau_{ign}} \tag{7–104}$$

The production term $\bar{\rho}/\tau_{ign}$ is disabled when no fuel is available, such that there is no further growth behind the flame front. The elapsed fraction of ignition delay time, $\tilde{R}$, may also be interpreted as a normalized radical concentration.

Autoignition is modeled to occur when the scalar exceeds a threshold, $\tilde{R} \geq 1$. In other words, reaction occurs if, and only if, the delay time has expired. This leads to a distinction in two kinds of autoignition models: 'Ignition Delay' and 'Knock'.

The physical information about ignition delay is provided by the ignition delay time, τ_{ign}, which is the primary input parameter to the model.

7.16.1. Ignition Delay Model

The ignition delay model is applicable when the principal combustion models would burn too early otherwise. It is available for combustion models for non-premixed combustion:

- Eddy Dissipation (EDM)
- Laminar Flamelet with PDF

The ignition delay model suppresses the reaction locally until the delay time has expired. Prior to that the reaction rate is set to zero (Eddy Dissipation) or the reaction progress is set to ‘fresh’ material (Flamelet).

7.16.2. Knock Model

The knock model is applicable when the principal combustion model would not ignite otherwise. This is the case for combustion models for premixed or partially-premixed combustion:

- Burning Velocity Model (BVM)
- Extended Coherent Flame Model (ECFM)

When autoignition (knock) occurs, a source term is added for reaction progress (BVM) or for flame surface density (ECFM), respectively. The magnitude of the source is defined by the Knock Reaction Rate, which by default is a first order Arrhenius reaction. For the BVM the following source is added locally to the reaction progress equation:

$$S_{c,knock} = \bar{\rho}\,(1-\tilde{c})\,\cdot A \cdot \exp\left(-E_{A,knock}/\,(RT)\right)\cdot \delta\,(\tilde{R}-1) \quad \textbf{(7–105)}$$

where $\delta\,(\tilde{R}-1)$ denotes a step function that is zero for $\tilde{R} < 1$ and one for $\tilde{R} > 1$, A_{knock} is the **Pre Exponential Factor** with a default value of $8.6 \cdot 10^9 s^{-1}$ and $E_{A,knock}$ is the **Activation Energy** with a default value of 15078 J/mol.

In order to improve the numerical robustness of the model, the knock reaction rate is limited to consume only a certain fraction α of the available fuel per timestep. The specified fraction is obeyed accurately when using the first order time discretization scheme, and it is approximate for the second order scheme. The default limit is $\alpha = 0.6$, that is no more than 60% of the fuel locally available may burnt by knock per timestep.

For the ECFM, the flame surface density is locally pulled towards a value equivalent to the knock reaction rate by adding a corresponding ‘Dirichlet’ source term:

$$S_{\Sigma,knock} = C\,\cdot\,(\Sigma_{knock} - \Sigma)\,\cdot \delta\,(\tilde{R}-1) \quad \textbf{(7–106)}$$

where

$$\Sigma_{knock} = S_{c,knock}/\left(\bar{\rho}_u s_L\right)$$

and, where C is a large coefficient (for example, 10^6).

7.16.3. Ignition Delay Time

The ignition delay time, τ_{ign}, is a function of local conditions, namely the fuel, pressure and temperature. The user may either specify a custom correlation by CEL or chose from the built-in empirical correlations.

7.16.3.1. Douaud and Eyzat

The ignition delay time of gasoline can be modeled using the empirical correlation by Douaud and Eyzat [183]:

$$\tau_{ign}^{gasoline} = 18.69\ [\mathrm{ms}] \cdot \left(\frac{ON}{100}\right)^{3.4017} \left(\frac{p}{98066.5\ [\mathrm{Pa}]}\right)^{-1.7} \exp\left(\frac{3800\ [K]}{T}\right) \tag{7–107}$$

where the octane number, ON, is a user input parameter for the model. The correlation has been calibrated for the octane number range $80 \leq ON \leq 100$.

7.16.3.2. Hardenberg and Hase

The ignition delay time of Diesel fuel can be modeled using the empirical correlation by Hardenberg and Hase [185]:

$$\tau_{ign}^{Diesel} = \frac{0.36 + 0.22\ [\mathrm{s/m}] \cdot MPS}{6 \cdot RPM} \cdot \exp\left[E_A\left(\frac{1}{RT} - \frac{1}{17190\ [\mathrm{J/mol}]}\right) + \left(\frac{21.2\ [\mathrm{bar}]}{p - 12.4\ [\mathrm{bar}]}\right)^{0.63}\right] \tag{7–108}$$

where

$$E_A = \frac{618840\ [\mathrm{J/mol}]}{CN + 25}$$

The user input parameters are the mean piston speed, MPS, the engine velocity in rounds per minute, RPM, and the cetane number of the fuel, CN. The apparent activation energy, E_A, has been calibrated for cetane number in the range $0 \leq CN \leq 100$.

7.17. Phasic Combustion

For multiphase simulations, phasic combustion describes combustion within each individual phase. The combustion models are solved in exactly the same way as for a single phase simulation.

- *The Eddy Dissipation Model* (p. 306)
- *The Finite Rate Chemistry Model* (p. 308)
- *The Combined Eddy Dissipation/Finite Rate Chemistry Model* (p. 309)
- *The Flamelet Model* (p. 310)
- *Burning Velocity Model (Premixed or Partially Premixed)* (p. 316)

7.18. NO Formation Model

The NO formation model is fully integrated into the CFX reaction and combustion module. This provides complete control to the NO model by means of common interface (`REACTION` object) on one hand;

on the other hand the generic reaction system profits from the extensions implemented for the NO model (Arrhenius with Temperature PDF reaction rates, variable pre-exponential factor for Arrhenius rates).

7.18.1. Formation Mechanisms

The formation of NOx is a complicated process involving several different mechanisms which are termed:

- Thermal NO
- Prompt NO
- Fuel Nitrogen
- N_2O
- Reburn (destruction of NO)

Reactions for the first three formation paths (thermal, prompt and fuel) and for NO reburn are predefined in the `REACTIONS` database. It is possible to add reactions for other mechanisms, or modify the provided mechanisms, using the **Reaction** details view in CFX-Pre (or manually in the CCL commands file).

7.18.1.1. Thermal NO

The thermal NO mechanism is a predominant source of NOx in gas flames at temperatures above 1800 K. The NO is formed from the combination of free radical O and N species, which are in abundance at high temperatures. The two-step mechanism, referred to as the Zeldovich mechanism, is thought to dominate the process:

$$\mathrm{O} + \mathrm{N}_2 \rightarrow \mathrm{NO} + \mathrm{N} \tag{7–109}$$

$$\mathrm{N} + \mathrm{O}_2 \rightarrow \mathrm{NO} + \mathrm{ON} \tag{7–110}$$

In sub or near stoichiometric conditions, a third reaction may be important:

$$\mathrm{OH} + \mathrm{N} \rightarrow \mathrm{NO} + \mathrm{H} \tag{7–111}$$

When this step is included with the first two, it is referred to as the extended Zeldovich mechanism.

The rates of each of these three reactions (using the subscripts 1,2,3 to describe the three reactions) are expressed as [43]:

$$k_1 = \left(1.8 \cdot 10^{11}\right) \exp\left(- \frac{38370}{T} \right) \quad \textbf{(7–112)}$$

$$k_2 = (6.4 \cdot 10^{9}) \exp\left(- \frac{3162}{T} \right) \quad \textbf{(7–113)}$$

$$k_3 = 3.0 \cdot 10^{13} \quad \textbf{(7–114)}$$

When multiplied by the concentrations of the reactants, they yield rates in terms of $[\mathrm{kmol}/m^3/s]$, which can be converted to a volumetric mass source term.

The first step tends to be rate limiting, producing both an NO and N radical species. The N radical is assumed to be oxidized by reaction *Equation 7–110* (p. 339) in the Zeldovich mechanism and also by reaction *Equation 7–111* (p. 339) in the extended Zeldovich mechanism. Either way, these second oxidation reactions are assumed to be fast and if Reaction *Equation 7–109* (p. 339) occurs, then two NO molecules will be formed. The thermal NO formation in $kg/m^3/s$, $S_{\mathrm{NO,\,thermal}}$, is therefore related to the rate of reaction *Equation 7–109* (p. 339):

$$S_{\mathrm{NO,\,thermal}} = 2\, W_{\mathrm{NO}}\, k_{\mathrm{thermal}} \left[O \right] \left[N_2 \right] \quad \textbf{(7–115)}$$

$$k_{\mathrm{thermal}} = k_1 \quad \textbf{(7–116)}$$

Here, W_{NO} denotes the molecular mass of NO. Thus, if the molar concentrations [O] and [N_2] of O radicals and N_2 are known, the thermal NO mechanism can be calculated.

When using the Laminar Flamelet model, almost always the O radical concentration can be taken without further assumptions from the solution because the model predicts it directly. However, when using the Eddy Dissipation model (EDM) and/or the Finite Rate Chemistry model (FRC), O radical concentrations usually are not known directly but must be derived from other quantities. Here, the O radical concentration is estimated from the molecular oxygen dissociation,

$$\frac{1}{2}\, \mathrm{O}_2 \Leftrightarrow \mathrm{O} \quad \textbf{(7–117)}$$

(Westenberg, 1975):

$$[\,\mathrm{O}\,] = 12567 \left[kmol^{1/2} m^{-3/2} K^{1/2} \right] \cdot T^{-1/2} \exp\left(-31096 K / T \right) [\,\mathrm{O}_2\,]^{1/2} \quad \textbf{(7–118)}$$

By substitution, the effective source term for NO then reads:

$$S_{\mathrm{NO,thermal,O_2}} = W_{\mathrm{NO}}\, k_{\mathrm{thermal}}\, [\mathrm{O}]^{1/2} \left[\, \mathrm{N}_2 \right] \tag{7–119}$$

$$k_{\mathrm{thermal,0_2}} = 4.52414 \cdot 10^{15} \left[m^{3/2}\, kmol^{-1/2}\, K^{1/2}\, s^{-1} \right] \cdot T^{-1/2} \cdot \exp(69466\ \mathrm{K/T}) \tag{7–120}$$

7.18.1.2. Prompt NO

At temperatures lower than 1800 K, hydrocarbon flames tend to have an NO concentration that is too high to be explained with the Zeldovich mechanisms. Hydrocarbon radicals can react with molecular nitrogen to form HCN, which may be oxidized to NO under lean flame conditions.

$$\mathrm{CH} + \mathrm{N}_2 \rightarrow \mathrm{HCN} + \mathrm{N} \tag{7–121}$$

$$\mathrm{HCN} + \mathrm{O}_2 \rightarrow \mathrm{NO} + \ldots \tag{7–122}$$

The complete mechanism is very complicated. However, De Soete (see also Peters and Weber, 1991) proposed a single reaction rate to describe the NO source by the Fenimore mechanism, $S_{\mathrm{NO,prompt}}$

$$S_{\mathrm{NO,prompt}} = W_{\mathrm{NO}}\, k_{\mathrm{prompt}}\, [\,\mathrm{O}_2\,]^{1/2}\, [\,\mathrm{N}_2\,]\, [\,\mathrm{Fuel}\,] \left(\frac{W}{\rho} \right)^{3/2} \tag{7–123}$$

$$k_{\mathrm{prompt}} = A_{\mathrm{prompt}} \exp\left(- T_{A,\mathrm{prompt}} / T \right) \tag{7–124}$$

W_{NO} and W denote molar mass of NO and the mean molar mass of the mixture, respectively. The Arrhenius coefficients depend on the fuel. (De Soete, 1974) proposed the following values:

Methane fuel

$$\begin{aligned} A_{\mathrm{prompt}} &= 6.4 \cdot 10^{6} \left[1/s \right] \\ T_{A,\mathrm{prompt}} &= 36510 \left[K \right] \end{aligned} \tag{7–125}$$

Acetylene fuel

$$A_{\text{prompt}} = 1.2 \cdot 10^6 \left[1/s \right]$$
$$T_{A,\text{prompt}} = 30215 \left[K \right] \quad \textbf{(7–126)}$$

7.18.1.3. Fuel Nitrogen

The fuel nitrogen model assumes that nitrogen is present in the fuel by means of HCN. HCN is modeled to either form or destroy NO depending on the local conditions in the mixture, with HCO acting as an intermediate species. The mechanism consists of three reaction steps:

1. HCN-NO formation: $\mathrm{HCN} + \mathrm{O}_2 \rightarrow \mathrm{HCO} + \mathrm{NO}$
2. HCN-NO destruction: $\mathrm{HCN} + \mathrm{NO} \rightarrow \mathrm{HCO} + \mathrm{N}_2$
3. HCO oxidation: $\mathrm{HCO} + \frac{3}{4}\mathrm{O}_2 \rightarrow \mathrm{CO}_2 + \frac{1}{2}\mathrm{H}_2\mathrm{O}$

The reaction rates in [mol/s] for the reactions are, respectively:

$$R_1 = 3.5 \cdot 10^{10} \left[s^{-1} \right] \cdot [\,\mathrm{HCN}\,] \cdot \left(X_{\mathrm{O}_2} \right)^{\alpha} \cdot \exp\left(- \left(67 \frac{kcal}{mol} \right) / (R\,T) \right) \quad \textbf{(7–127)}$$

where $\alpha = 1 / \left(1 + \left(X_{\mathrm{O}_2} / 0.007 \right)^2 \right)$

$$R_2 = 3 \cdot 10^{12} \left[s^{-1} \right] \cdot \frac{W}{\rho} \cdot [\,\mathrm{HCN}\,]\,[\,\mathrm{NO}\,] \cdot \exp\left(- \left(60 \frac{kcal}{mol} \right) / (R\,T) \right) \quad \textbf{(7–128)}$$

$$R_3 = 8.3 \cdot 10^6 \left[s^{-1} \right] \cdot [\mathrm{HCO}]^{-0.3} \cdot [\,\mathrm{O}_2\,]^{1.3} \cdot \exp\left(- \left(30 \frac{kcal}{mol} \right) / (R\,T) \right) \quad \textbf{(7–129)}$$

where X_{O_2} denotes the molar fraction of O_2 and W denotes the mean molar mass of the mixture.

7.18.1.4. NO Reburn

Under fuel rich conditions, when the amount of oxygen available is not sufficient to oxidize all of the fuel, the excess fuel may lead to reduction of NO. This process can be described by a global reaction:

$$\mathrm{NO} + v_F \mathrm{Fuel} \rightarrow \frac{1}{2}\ \mathrm{N}_2 + v_{\mathrm{CO}_2} \mathrm{CO}_2 + v_{\mathrm{H_2O}}\ \mathrm{H}_2\mathrm{O} \tag{7–130}$$

One should keep in mind that this is only a global representation. The physical process actually occurring is much more complicated. The real process involves many intermediate components appearing during combustion of the fuel, e.g., CHx radicals, which attack the NO.

The stoichiometric coefficients for fuel, carbon dioxide and water vapor are fuel-dependent. For a given fuel they can easily be derived from the element balance. For methane, the global NO reburn reaction is:

$$\mathrm{NO} + \frac{1}{4}\mathrm{CH}_4 \rightarrow \frac{1}{2}\ \mathrm{N}_2 + \frac{1}{4}\mathrm{CO}_2 + \frac{1}{2}\ \mathrm{H}_2\mathrm{O} \tag{7–131}$$

The reaction rate will be fuel dependent. For coal volatiles, the reaction rate defaults to:

$$R_{\mathrm{return}} = 2.72 \cdot 10^6 \left[s^{-1} \right] \cdot \frac{W}{\rho} \cdot [\mathrm{NO}]\ [\mathrm{Fuel}] \cdot \exp\left((-9460\ [K])/T \right) \tag{7–132}$$

The same reaction rate is applied for the predefined NO reburn reaction for methane.

7.18.1.5. Turbulence Effects

The above reaction rates are applicable to laminar flow, premixed chemistry. In turbulence systems, fluctuations can have a dominant impact on the NO formation rate. For both the thermal and prompt NO mechanisms, there is a strong dependence of the rates on the temperature due to their high activation energy. Thus, temperature fluctuations, particularly positive fluctuations, can dramatically increase the NO formed in flames. These temperature fluctuations are included in CFX using a statistical approach.

In order to determine the mean rate for NO formation, a presumed probability density function (presumed PDF) method is used to compute the weighted average of the reaction rate:

$$\tilde{k} = \frac{1}{T_u - T_l} \int_{T_l}^{T_u} k(T) \cdot P(T)\ dT \tag{7–133}$$

This integration is carried out separately for each reaction step. For simplicity, the subscripts (thermal or prompt) have been omitted. The integration range for temperature, $[T_l\ ;\ T_u]$, is the range of temperatures occurring. The default for NO reactions is the range [300 K; 2300 K], but this may be modified by you on a per reaction scope.

The probability density function (PDF) P is computed from mean temperature, $\tilde{T}$, and the variance of temperature, $\widetilde{T''^2}$. The shape of P is presumed to be that of a beta function (β-function):

$$P(T) = \frac{\phi^{a\,(-1)}\,(1-\phi)^{\,b-1}}{\int_0^1 \psi^{a-1}\,(1-\psi)^{\,b-1}\,d\psi} \tag{7–134}$$

Where:

$$a = \tilde{\phi}\left(\frac{\tilde{\phi}\left(1-\tilde{\phi}\right)}{g} - 1\right), b = \left(1-\tilde{\phi}\right)\left(\frac{\tilde{\phi}\left(1-\tilde{\phi}\right)}{g} - 1\right) \tag{7–135}$$

and:

$$\phi = \frac{T - T_l}{T_u - T_l},\ \tilde{\phi} = \frac{\tilde{T} - T_l}{T_u - T_l}, g = \frac{\tilde{T''}}{(T_u - T_l)^{\,2}} \tag{7–136}$$

For vanishing temperature variance (vanishing temperature fluctuations), the beta function is approaching to a single Dirac peak (delta-function). In the limit, the integrated reaction rate is the same as for that for the standard Arrhenius rate. For very large fluctuation, the beta function goes towards a double Dirac peak, and for small to medium temperature variance the shape of the PDF is similar to that of a Gaussian distribution.

Arrhenius reaction rates integrated over a PDF for temperature is not limited to NO formation but may be used for any reaction.

7.18.1.6. Temperature Variance Equation

For the temperature variance, $\tilde{T''}^{2}$, that is needed from constructing the probability density function (PDF) used for the temperature integration, the following transport equation is solved:

$$\frac{\partial\,\rho\,\tilde{T''}^{2}}{\partial t} + \frac{\partial\,\bar{\rho}\,\tilde{u}_j\,\tilde{T''}^{2}}{\partial x_j} = \frac{\partial}{\partial x_j}\left\{\left(\bar{\mu} + \frac{\bar{\mu}_t}{\mathrm{Pr}_t}\right)\frac{\partial\,\tilde{T''}^{2}}{\partial x_j}\right\} + C_{\mathrm{prod}}\,\frac{\mu_t}{\mathrm{Pr}_t}\left(\frac{\partial\,\tilde{\bar{T}}}{\partial x_j}\right)^{2} - C_{\mathrm{diss}}\,\bar{\rho}\,\frac{\bar{\epsilon}}{\tilde{k}}\,\tilde{T''}^{2} \tag{7–137}$$

Default values for the model coefficients are $C_{\mathrm{prod}} = 2.0$ and $C_{\mathrm{diss}} = 2.0$.

The above equation is missing some physical aspects namely the production of temperature fluctuations due to heat release by chemical reaction. Heat release is fluctuating, too, because of turbulent fluctuations of the reactants. However, in the current model temperature variance is only needed as input to another model: The construction of a probability density function with presumed shape. For this purpose, the above equation provides sufficient accuracy.

For convenience of setting up a case, the temperature variance equation can be run without specifying BC or IC data. In the first release, these are not offered by the user interface (CFX-Pre). If absent, zero fluctuations are assumed at inlets, openings and walls with specified temperature. At walls with specified heat flux or transfer coefficient, the default BC for temperature variance is zero flux.

7.18.1.7. Model Control

Because the model for NO formation is implemented by means of `REACTION` objects, you have full control of all aspects of the model. CFX provides the same flexibility for the NO model as for the generic combustion and reaction system.

7.18.1.7.1. Adjusting Model Coefficients

It is possible for a user to modify any coefficient of the NO model. Because the NO formation reactions are defined by means of `REACTION` objects, they may be edited either in CFX-Pre in the CCL Editor or in the CCL text file (commands file).

The model parameter that is most likely to need adjusting is the temperature integration range for the presumed PDF. It is specified by the `Lower Temperature` and `Upper Temperature` parameters. These appear in the `FORWARD REACTION RATE` object when **Option** is set to `Arrhenius with Temperature PDF`. The predefined reaction schemes come with the range set to the interval [300 K, 2300 K]. It is recommended to adjust this to the maximum temperature range occurring. For many systems, this temperature range is defined by the minimum inlet temperature and the adiabatic flame temperature.

7.18.1.7.2. User Defined NO Formation Mechanisms

The NO formation model can be extended to a user's specific needs by adding appropriate reactions for NO, for example, fuel nitrogen or the N_2O reaction path. The procedure is the same as for any other reaction. A user probably may want to chose the `Arrhenius with Temperature PDF` option for reactions rate in order to account for turbulent fluctuations of temperature.

It is also possible to select each NO formation path individually (thermal, prompt, fuel nitrogen, reburn, or possibly user defined). Simply add only those reactions to the mixture material that you want to account for.

7.19. Chemistry Post-Processing

The components and reactions that are post-processed are one-way coupled to the main simulation. This means that there is no effect on mixture properties or heat release. The list of components and reactions for post-processing are specified by the user.

Post-processing components:

- No contribution to mixture properties (density, static enthalpy)
- No contribution to total mass

Post processing reactions:

- Generate sources for components that are also post-processed
- Generate no source for regular components (i.e., not post-processed ones)
- No heat release

The above model assumptions are justified if the component mass fractions and reaction turnover are small relative to the bulk mixture. A typical application is the simulation of pollutants, e.g., NO formation. The transport equations for post-processing components are solved at the end of the flow analysis after the solving of the main equations. When the post-processing reactions require a temperature variance equation (reaction rate option is `Arrhenius with Temperature PDF`) and the variance equation was not solved during the analysis, then temperature variance will be post-processed as well. In this case, the temperature variance equation will be solved first to convergence or maximum number of iterations and is then followed by the post-processing components group.

Note

For steady state cases, chemistry post-processing is not performed at the end of the flow analysis if the termination of the flow analysis was due to a user-defined solver interrupt condition. For details, see Interrupt Control in the *CFX-Pre User's Guide*.

7.20. Soot Model

In the Magnussen soot model (Magnussen and Hjertager [46], it is assumed that soot is formed from a gaseous fuel in two stages, where the first stage represents formation of radical nuclei, and the second stage represents soot particle formation from these nuclei. Transport equations are solved for the specific concentration of radical nuclei, X_N[mol/kg], and for the soot mass fraction, $\tilde{Y}_s$ [kg/kg]:

$$\frac{\partial\,(\bar{\rho}\,\tilde{X}_N)}{\partial t}+\frac{\partial\left(\bar{\rho}\,\tilde{u}_j\,\tilde{X}_N\right)}{\partial x_j}=\left\{\left(\bar{\mu}+\frac{\mu_t}{Pr_t}\right)\frac{\partial\tilde{X}_N}{\partial x_j}\right\}+\tilde{S}_{\text{nuclei},f}+\tilde{S}_{\text{nuclei},c} \qquad \textbf{(7–138)}$$

$$\frac{\partial\,(\bar{\rho}\,\tilde{Y}_S)}{\partial t}+\frac{\partial\left(\bar{\rho}\,\tilde{u}_j\,\tilde{Y}_S\right)}{\partial x_j}=\left\{\left(\bar{\mu}+\frac{\mu_t}{Pr_t}\right)\frac{\partial\tilde{Y}_S}{\partial x_j}\right\}+\tilde{S}_{\text{soot},f}+\tilde{S}_{\text{soot},c} \qquad \textbf{(7–139)}$$

The modeling procedure can be grouped into three independent parts:

1. Formation of nuclei and soot particles following the models of Tesner et al., [45]
2. Combustion of nuclei and soot particles
3. Magnussen's Eddy Dissipation Concept (EDC) for modeling the effect of turbulence on mean reaction rates.

The soot model can be used in either single phase or multiphase flow (MPF) configurations. In multiphase calculations, however, the soot variables cannot be a separate phase but must be assigned to one of the fluids.

7.20.1. Soot Formation

Formation of nuclei and soot particles is computed following the empirical models of Tesner et al. [45]. The source terms are formulated in terms of particle number concentrations for nuclei:

$$C_N = \rho \cdot A \cdot X_N \qquad [part/m^3] \tag{7–140}$$

and soot particles:

$$C_S = \rho \cdot \frac{Y_S}{m_p} \qquad [part/m^3] \tag{7–141}$$

Where $A = 6.02214199 \cdot 10^{23}$ [part/mol] is Avogadro's number and:

$$m_p = \rho_{\text{soot}}\, \pi\, d^3/6 \tag{7–142}$$

is the mass of a soot particle ρ_{soot} and d are the density and the mean diameter of the soot particles, respectively. With the above definitions, the source terms for nuclei $\tilde{S}_{\text{nuclei},f}$ and soot formation $\tilde{S}_{\text{soot},f}$ can be modeled as (Tesner et al. [45]):

$$\tilde{S}_{\text{nuclei},f} = n_0 + (f - g)\ C_N - g_0\, C_N\, C_S \tag{7–143}$$

$$\tilde{S}_{\text{soot},f} = m_p\, (a - b\ C_S)\ C_N \tag{7–144}$$

In the nuclei equation, the spontaneous formation of radical nuclei from the fuel, n_0, is modeled using the Arrhenius approach,

$$n_0 = a_0 f_c\, \rho\, Y_{\text{fuel}} \exp\left(-T_{A,0}/T \right) \tag{7–145}$$

where f_c is the mass fraction of carbon in the fuel material. f is a linear branching coefficient, g is a linear termination coefficient, and g_0 is a coefficient of linear termination or radical nuclei on soot particles. In the soot equation, a and b are constants. The default values for all of the soot model parameters are summarized in the following table:

Parameter	Default Value by [part]	Default value by [mol]
ρ_{soot}	$2000\ [kg/m^3]$	$2000\ [kg/m^3]$
d	$1.785 \cdot 10^{-8}\ \left[\mathrm{m}\right]$	$1.785 \cdot 10^{-8}\ \left[\mathrm{m}\right]$
a_0	$1.35 \cdot 10^{37}$ [part/kg/s]	$2.24 \cdot 10^{13}$ [mol/kg/s]
f_c	Fuel dependent (methane 12/16, acetylene 24/26)	Fuel dependent (methane 12/16, acetylene 24/26)
$T_{A,0}$	$90000\ [\mathrm{K}]$	$90000\ [\mathrm{K}]$
$f - g$	$100\ [1/\mathrm{s}]$	$100\ [1/\mathrm{s}]$

Parameter	Default Value by [part]	Default value by [mol]
g_0	$1.0 \cdot 10^{-15}$ $[m^3/s/part]$	$6.02 \cdot 10^{8}$ $[m^3/s/mol]$
a	$1.0 \cdot 10^{5}$ [1/s]	$1.0 \cdot 10^{5}$ [1/s]
b	$8.0 \cdot 10^{-14}$ $[m^3/s/part]$	$4.82 \cdot 10^{10}$ $[m^3/s/mol]$

Most references list the coefficients using the absolute particle number [part] in the physical dimensions. However, for numerical, reasons CFX is using [mol] instead. To convert from the [part] system into the [mol] system, coefficients g_0 and b have to be multiplied by Avogadro's number

$A = 6.02214199 \cdot 10^{23}$[part/mol], and the coefficient a_0 needs to be divided by A. All other coefficients remain unchanged.

7.20.2. Soot Combustion

The mean rates of combustion of nuclei, $\tilde{S}_{\text{nuclei},c}$, and soot particles, $\tilde{S}_{\text{soot},c}$ are calculated from the fuel consumption reaction rate, $\tilde{S}_{\text{fuel}}$ in [kg/m^3/s], as:

$$\tilde{S}_{\text{nuclei},c} = \tilde{S}_{\text{fuel}} \frac{\tilde{X}_N}{\tilde{Y}_{\text{fuel}}} \quad \textbf{(7–146)}$$

$$\tilde{S}_{\text{soot},c} = \tilde{S}_{\text{fuel}} \frac{\tilde{Y}_S}{\tilde{Y}_{\text{fuel}}} \quad \textbf{(7–147)}$$

This is equivalent to posing the same assumptions to nuclei and soot particles combustion that were made for fuel consumption. For example, when the Eddy Dissipation model is applied for the fuel consumption reaction, combustion of nuclei and soot particles is assumed to operate at the fast chemistry limit. No special model for turbulence effects is required for the nuclei and soot combustion rates as this is already accounted for in the computation of the fuel reaction rate.

7.20.3. Turbulence Effects

The above reaction rates are applicable to laminar flow. In turbulent systems, fluctuations can have a significant impact on nuclei and soot formation because of the nonlinearity of the respective source terms. To account for the effect of turbulence on soot formation, the Eddy Dissipation Concept (EDC) developed by Magnussen [44] is applied.

The Eddy Dissipation Concept (EDC) is a reactor concept that identifies a reactor, where the combustion of fuel takes place, related to the fine structures in turbulence. This reactor is treated as a homogeneous reactor exchanging mass and energy with the surrounding fluid. The fraction of mass that is contained in the fine structures is determined from the turbulence quantities.

Local balance equations are solved to compute the temperature and the concentrations of nuclei and soot particles in the fine structures and in the surrounding fluid, respectively. The mean source terms are then computed assuming a bimodal distribution of the fluid between the fine structures and the surrounding. For a detailed explanation of the Eddy Dissipation Concept, see (Magnussen, 1989).

The EDC procedure need not be applied to the soot combustion terms, because turbulence is already accounted for in the computation of the fuel reaction rate. For details, see *Soot Combustion* (p. 348).

Chapter 8: Radiation Theory

This chapter contains a simple summary of the theory of thermal radiation and the algorithms used in CFX:

Details on modeling radiation in CFX are available in Radiation Modeling in the *CFX-Solver Modeling Guide*.

8.1. Radiation Transport

The goal of radiation modeling is to solve the radiation transport equation, obtain the source term, S, for the energy equation, and the radiative heat flux at walls, among others quantities of interest. You should restrict yourself to coherent time-independent radiation processes. This is normally a very good approximation of situations likely to be met in industrial applications because the time scale for radiation to come into local equilibrium is very short and the temperatures are relatively low.

The spectral radiative transfer equation (RTE) can be written as:

$$\frac{dI_v(\boldsymbol{r},\boldsymbol{s})}{ds} = \left(-(K_{av}+K_{sv})\, I_v(\boldsymbol{r},\boldsymbol{s}) + K_{av} I_b(v,T) + \frac{K_{sv}}{4\pi}\int_{4\pi} dI_v(\boldsymbol{r},\boldsymbol{s}')\, \Phi(\boldsymbol{s}\cdot\boldsymbol{s}')\, d\Omega' + S \right) \tag{8–1}$$

where:

- v = frequency
- **r** = *position vector*
- **s** = *direction vector*
- s = path length
- K_a = absorption coefficient
- K_s = scattering coefficient
- I_b= Blackbody emission intensity

- I_ν= Spectral radiation intensity which depends on position (r) and direction (s)
- T = local absolute temperature
- Ω = solid angle
- Φ = in-scattering phase function
- S = radiation intensity source term

The RTE is a first order integro-differential equation for I_ν in a fixed direction, **s**. To solve this equation within a domain, a boundary condition for I_ν is required. The following are the boundary conditions currently supported in CFX:

- Diffusely emitting and reflecting opaque boundaries

$$I_\nu(\boldsymbol{r}_w, \boldsymbol{s}) = \varepsilon_\nu(r_w)\; I_b(\nu, T) + \frac{\rho_w(\boldsymbol{r}_w)}{\pi} \int_{n \cdot s' < 0} I_\nu(\boldsymbol{r}_w, \boldsymbol{s}')\; |\boldsymbol{n} \cdot \boldsymbol{s}'|\; d\Omega' \qquad \textbf{(8–2)}$$

where ε_ν=spectral emissivity.

- Diffusely emitting and specularly reflecting boundaries

$$\begin{aligned} I_\nu(\boldsymbol{r}_w, \boldsymbol{s}) = \; & \varepsilon_\nu(\boldsymbol{r}_w)\; I_b(\nu, T) + \frac{\rho_\nu^d(\boldsymbol{r}_w)}{\pi} \int_{n \cdot \boldsymbol{s}' < 0} I_\nu(\boldsymbol{r}_w, \boldsymbol{s}')\; |\boldsymbol{n} \cdot \boldsymbol{s}'|\; d\Omega' \\ & + \rho_\nu^s(\boldsymbol{r}_w)\; I_\nu\left(\boldsymbol{r}_w, \boldsymbol{s}_s\right) \end{aligned} \qquad \textbf{(8–3)}$$

where:

- ρ_ν^d=diffuse reflectivity= $(1 - \varepsilon_\nu)$ *diffuse fraction
- ρ_ν^s=specular reflectivity= $(1 - \varepsilon_\nu)$ *(1-diffuse fraction)
- ρ_ν=spectral reflectivity=$\rho_\nu^d + \rho_\nu^s$= $(1 - \varepsilon_\nu)$
- $\boldsymbol{s}_s$=specular direction
- Semi-transparent walls (Monte Carlo only)

Due to the dependence on 3 spatial coordinates, 2 local direction coordinates, *s*, and frequency, the formal solution of the radiative transfer equation is very time consuming and usually accomplished by the use of approximate models for the directional and spectral dependencies. For directional approximations, CFX includes `Rosseland`, `P-1`, `Discrete Transfer` and `Monte Carlo`. For spectral approximations, CFX includes: `Gray`, `Multiband` and `Weighted Sum of Gray Gases`.

8.1.1. Blackbody Emission

The energy spectrum for radiation emitted by a blackbody is represented by:

$$E_b(v,T) = \frac{2\pi v^2}{c^2} \frac{n^2 hv}{e^{(hv/k_B T)} - 1} \left(W m^{-2} Hz^{-1} \right) = \pi I_b(v,T) \tag{8–4}$$

where:

- n = refractive index
- h = Planck's constant
- c = speed of light in vacuum
- k_B = Boltzmann's constant

Writing $x = hv/k_B T$, you have:

$$E_b(x,T) = n^2 \sigma T^4 \left(\frac{h}{k\ T} \right) \left(\frac{15}{\pi^4} \frac{x^3}{e^x - 1} \right) \tag{8–5}$$

where σ = Stefan-Boltzmann constant:

$$\sigma \equiv \frac{2\pi^5 k^4}{15 h^3 c^2} \tag{8–6}$$

The total blackbody emission is simply the integral of E_b over all frequencies:

$$E_b(T) = \int_0^\infty E_b(x,T)\ dx = n^2 \sigma T^4 \tag{8–7}$$

Note that the blackbody emission is proportional to the fourth power of the temperature and because of this strong dependence, radiation is usually unimportant or totally dominant for heat transfer.

The sun, for example, is approximately a blackbody at a temperature of 5700 K. The spectrum peaks in the yellow part of the visible spectrum.

Combustion temperatures are typically 1000 - 2000 K, with spectrum peaks in the near infra-red range. Note that the peak of the spectrum as a function of wavelength is at:

$$hv \sim 4\ k_B\ T \tag{8–8}$$

8.1.2. Quantities of Interest

The spectral radiative heat flux, q_v, passing through a surface at some location **r** with a unit vector normal **n** is:

$$q_v^R(\boldsymbol{r},\boldsymbol{n}) = \int (\boldsymbol{s}\cdot\boldsymbol{n})\; I_v(\boldsymbol{r},\boldsymbol{s})\; d\Omega_{\boldsymbol{s}} \tag{8–9}$$

Integrating the equation of transfer over solid angles, the divergence of the spectral radiative heat flux is given by:

$$\left(-\nabla\cdot q_v^R\right) = K_a\,(G_v - 4\,E_{bv}) \tag{8–10}$$

where G_v is the spectral incident radiation, given by:

$$G_v \equiv \int I_v d\Omega_s \tag{8–11}$$

The total radiative flux is obtained by integrating *Equation 8–10* (p. 354) over the spectrum:

$$\nabla\cdot q^R = \int_0^\infty q_v^R\,dv = \int_0^\infty K_{av}\,G_v\,dv - 4\int_0^\infty K_{av}\,E_{bv}\,dv \tag{8–12}$$

In the case of pure scattering, $K_a = 0$. Therefore $\nabla\cdot q^R = 0$, as it should because in this case no energy is lost from the radiation field; clearly this is also true in thermodynamic equilibrium.

8.1.2.1. Optical Thickness

Optical thickness is a dimensionless quantity that represents the ability of a given path length of gas to attenuate radiation of a given wavelength. Optical thickness is given by:

$$\tau = \int_0^S K_\lambda(S^*)\; dS^* \tag{8–13}$$

where τ is the optical thickness (or opacity) of the layer of thickness S and is a function of all the values of K_λ between 0 and S. A large value of τ means large absorption of radiation. Note that this definition of optical thickness is different from that traditionally found in the optics literature where the optical thickness is a property of the material.

8.1.3. Radiation Through Domain Interfaces

If radiation is included through conducting solids, then usually the difference in refractive indices between the fluid and solid determines the amount of reflection and refraction that occurs. The probability of being reflected is given by Fresnels' equation

$$\rho = \frac{1}{2}\left[\frac{\tan^2(\theta_1 - \theta_2)}{\tan^2(\theta_1 + \theta_2)} + \frac{\sin^2(\theta_1 - \theta_2)}{\sin^2(\theta_1 + \theta_2)}\right] \tag{8–14}$$

The fraction of the electromagnetic wave that is reflected normally depends on the polarization of the photon. CFX does not keep track of photon polarizations. Assuming that the photons are randomly polarized, then it is sufficient to take the average result. The two extreme polarizations are termed transverse electric (TE) and transverse magnetic (TM), and describe the orientations of the electric and magnetic vectors relative to the interface.

For the TE mode, the ratio of reflected to incident wave amplitude (E_E) is given by:

$$E_E = \frac{\frac{n_1}{n_2}\cos\theta - \cos\varphi}{\frac{n_1}{n_2}\cos\theta + \cos\varphi} \tag{8–15}$$

and for the TM mode the ratio of reflected to incident wave amplitude (E_M) is given by:

$$E_M = \frac{\frac{n_1}{n_2}\cos\varphi - \cos\theta}{\frac{n_1}{n_2}\cos\varphi + \cos\theta} \tag{8–16}$$

where θ and φ are the incident and refracted angles, and n_1 and n_2 are the refractive indices of the two media.

The probability of being reflected is determined by the energy flow at the interface which is proportional to the square of the wave amplitude. Thus, the average reflection coefficient is given as:

$$0.5\left(E_E^2 + E_M^2\right) \tag{8–17}$$

and the probability of being transmitted is:

$$1 - 0.5\left(E_E^2 + E_M^2\right) \tag{8–18}$$

No absorption takes place at the interface, so the probability of transmission plus reflection is always one. If the photon is transmitted, then the angle of refraction is determined by Snells' law:

$$\frac{\sin\varphi}{\sin\theta} = \frac{n_2}{n_1} \tag{8–19}$$

CFX performs these calculations at every radiation element boundary, although, in most cases, there is no change of refractive index.

8.2. Rosseland Model

The Rosseland approximation is a simplification of the Radiative Transport Equation (RTE) for the case of optically thick media. It introduces a new diffusion term into the original energy transport equation with a strongly temperature-dependent diffusion coefficient.

A good source for the simplification of the Radiation Transport Equation for the optically thick limit can be seen in Siegel and Howe [23]. The total radiative heat flux in an optically thick, and linearly anisotropic scattering medium can be written as:

$$q_r = -\int_0^{\infty} \frac{4}{3\beta - AK_s} \nabla E_{b\nu}\, d\nu \tag{8–20}$$

where β is the extinction coefficient (i.e., absorption plus scattering).

When the Rosseland Approximation is introduced into the energy transport equation, the conduction and radiative heat flux can be combined as:

$$q = q_c + q_r \tag{8–21}$$

$$= -\,(k + k_r)\ \nabla T \tag{8–22}$$

$$\text{where} \qquad k_r = -\frac{16\,\sigma\, n^2\, T^3}{3\,\beta} \tag{8–23}$$

where k is the thermal conductivity and k_r is the "total radiative conductivity." *Equation 8–21* (p. 356) is called upon to calculate the temperature field in the energy equation.

8.2.1. Wall Treatment

The Rosseland approximation is not valid near walls. Therefore, a special boundary condition must be specified when heat conduction is comparable to radiation heat transfer. It has been proposed [24] that a temperature slip boundary condition should be in this region. From [24] the heat flux at the wall, $q_{r,w}$ is given by:

$$q_{r,w} = \frac{-\sigma\left(T_w^4 - T_g^4\right)}{\psi} \tag{8–24}$$

where ψ is the slip co-efficient, T_w is the wall temperature and T_g is the gas temperature at the wall. ψ is computed as the solution of:

$$\psi = \frac{3}{4\,\pi} \int_0^1 \operatorname{atan}\frac{1}{\varphi_v}\, d\varphi \quad \textbf{(8–25)}$$

where φ_v is given by:

$$\varphi_v = \frac{1}{\pi} \left[\frac{\beta\, k}{8\, \varphi^3\, \sigma\, T_w^3} - \frac{2}{\varphi} - \ln\frac{1-\varphi}{1+\varphi} \right] \quad \textbf{(8–26)}$$

8.3. The P1 Model

The Differential Approximation or P1 is also a simplification of the Radiation Transport Equation, which assumes that the radiation intensity is isotropic or direction independent at a given location in space. The full form of the radiant energy equation and the derivation of the P1 model for radiation are given in Modest [8]. Only a brief summary will be given here.

The spectral radiative heat flux in the diffusion limit for an emitting, absorbing, and linearly scattering medium, can be computed as:

$$q_{rv} = -\frac{1}{3\,(K_{av} - K_{sv}) - AK_{sv}}\, \nabla G_v \quad \textbf{(8–27)}$$

The equation for the spectral incident radiation that results from substituting the above terms into the radiation transport equation:

$$-\nabla \cdot \left(\frac{1}{3\,(K_{av} - K_{sv}) - A\, K_{sv}}\, \nabla G_v \right) = K_{av}\,(E_{bv} - G_v) \quad \textbf{(8–28)}$$

where A is the linear anisotropy coefficient.

8.3.1. Wall Treatment

Assuming that the radiation intensity arriving at and leaving a wall are directionally independent, the boundary condition for Equation 8–28 (p. 357) at walls is:

$$\boldsymbol{n} \cdot \boldsymbol{q}_{rv} = -\frac{1}{3\,(K_{av} - K_{sv}) - A\, K_{sv}}\, \frac{\partial\, G_v}{\partial\, n^+} = \frac{\varepsilon_v}{2\,(2 - \varepsilon_v)}\, \left[E_{bv} - G_v\right]_w \quad \textbf{(8–29)}$$

where $\boldsymbol{n}$ is the unit vector outward normal to the wall, n^+ is a distance coordinate in the same direction, and w represents the value at the wall.

8.4. Discrete Transfer Model

The implementation of the Discrete Transfer model in CFX assumes that the scattering is isotropic; therefore, Equation 8–1 (p. 351) can be simplified as:

$$\frac{d\,I_v(\boldsymbol{r},\boldsymbol{s})}{ds} = -\,(K_{av}+K_{sv})\,I_v(\boldsymbol{r},\boldsymbol{s}) + K_a I_b(v,T) + \frac{K_{sv}}{4\pi}\int_{4\pi} I_v(\boldsymbol{r},\boldsymbol{s}')\,d\Omega' + S \tag{8–30}$$

Assuming that the system is reasonably homogeneous, so that:

$$I_v(\boldsymbol{r}) \sim I_v(\boldsymbol{r}+d\boldsymbol{r}) \quad q_v^R(\boldsymbol{r}) \sim q_v^R(\boldsymbol{r}+d\boldsymbol{r}) \tag{8–31}$$

the approach is then to solve for the intensity, I_v, along rays leaving from the boundaries using the equation of transfer:

$$I_v(\boldsymbol{r},\boldsymbol{s}) = I_{vo}\exp(-\,(K_{av}+K_{sv})\,s) + I_{bv}(1-\exp(-K_a s)) + K_{sv}\,\overline{I_v} \tag{8–32}$$

where:

I_{vo} = Radiation Intensity leaving the boundary

I_v = Mean Radiation Intensity

Then, integrate I over solid angle at discrete points to get the spectral incident radiation, G and the radiative heat flux, q_r and use the homogeneity assumption to extend the solution to the entire domain. Non-linearities in the system due to scattering, diffuse reflection, or temperature dependency of radiation quantities is overcome by iteration.

Because the objective of thermal radiation modeling is to obtain the total volumetric absorption and emission, additional calculations are still needed. For the Gray spectral model, the calculation is done once for a unique radiation intensity field. For the Multiband and Weighted Sum of Gray Gases, the solution must be computed for each spectral band/ gray gas and a final integration to obtain the total radiation quantities is required. Under the assumption of coherent radiation field, ie., the solution at a given frequency is independent of that at all other frequencies.

8.5. Monte Carlo Model

The Monte Carlo model assumes that the intensity is proportional to the differential angular flux of photons and you can think of the radiation field as a photon gas. For this gas, K_a is the probability per unit length that a photon is absorbed at a given frequency. Therefore, the mean radiation intensity, I is proportional to the distance traveled by a photon in unit volume at $\boldsymbol{r}$, in unit time.

Similarly q_v^R is proportional to the rate of incidence of photons on the surface at $\boldsymbol{r}$, because volumetric absorption is proportional to the rate of absorption of photons.

By following a typical selection of photons and tallying, in each volume element, the distance traveled, you can obtain the mean total intensity.

By following a typical selection of photons and tallying, in each volume element, the distance times the absorption coefficient, you can obtain the mean total absorbed intensity.

By following a typical selection of photons and tallying, in each volume element, the distance times the scattering coefficient, you can obtain the mean total scattered intensity.

By also tallying the number of photons incident on a surface and this number times the emissivity, you obtain the mean total radiative flux and the mean absorbed flux.

Note that no discretization of the spectrum is required because differential quantities are not usually important for heat transfer calculations. Providing that the spectral (Multiband or Weighted Sum of Gray Gases) selection is done properly, the Monte Carlo tallying automatically integrates over the spectrum.

8.5.1. Monte Carlo Statistics

The Monte Carlo radiation solver computes the standard deviation error based on Poisson statistics. The user-specified number of histories is divided into several groups. Histories are selected from each group and their physical interactions (emission, absorption, reflection, and so on) are tracked through the domain. At the end of the calculation, each group provides values for the quantities of interest, such as irradiation heat flux or absorbed radiation. The mean value, and standard deviation, of each quantity of interest are computed from the groups. A normalized standard deviation is computed by dividing the standard deviation by the mean value.

8.6. Spectral Models

The radiation intensity field is also a function of the spectrum as shown in *Equation 8–1* (p. 351). In order to make the spectral dependence tractable in conjunction with a flow calculation, CFX supports three different models for the spectral dependency of the radiative transfer equation: Gray, Multiband and Weighted Sum of Gray Gases.

8.6.1. Gray

The Gray model assumes that all radiation quantities are nearly uniform throughout the spectrum, consequently the radiation intensity is the same for all frequencies. Then, the dependency of *Equation 8–1* (p. 351) on frequency can be dropped.

This implies that only one radiative transfer equation must be solved and that all total radiation quantities and their spectral counterpart are the same.

8.6.2. Multiband Model

For this model, the spectrum is sub-divided into N spectral bands of finite width where radiative quantities are nearly uniform or can be averaged without losing accuracy. These bands should span the thermal radiation section of the spectrum. It is assumed that the value at a given spectral band is represented by the spectral band midpoint value in frequency domain.

CFX assumes that the main spectral variable is frequency because it is independent of the material refractive index and it will facilitate the setup of multidomain problems. Other spectral variables, such wavelength and wavenumber would be available for vacuum only.

Then, the radiative transfer equation is integrated within is spectral band and a modified RTE is obtained:

$$\frac{d\ \Delta v\ \bar{I}_{\Delta v}(\boldsymbol{r},\boldsymbol{s})}{ds} = -\overline{(K_{av}+K_{sv})}\ \Delta v\ I_{\Delta v}(\boldsymbol{r},\boldsymbol{s}) + \overline{K}_a F_{\Delta v} I_b(T) + \frac{\overline{K}_{sv}}{4\pi}\int_{4\pi} \Delta v\ I_{\Delta v}(\boldsymbol{r},\boldsymbol{s}')\ \Phi(\boldsymbol{s}\cdot\boldsymbol{s}')\ d\Omega' + S \quad \text{(8–33)}$$

for $\Delta v\ I_{\Delta v}$, where the emission within the spectral band is weighted by:

$$F_{\Delta v} = \int_{v_1}^{v_2} E_b(v,T)\ dv = \int_0^{v_2} E_b(v,T)\ dv - \int_0^{v_1} E_b(v,T)\ dv \quad \text{(8–34)}$$

After solving one RTE per spectral band, total radiation intensity can be computed as:

$$I(r,s) = \sum_1^N \Delta v\ I_{\Delta v}(r,s) \quad \text{(8–35)}$$

This immediately suggests that for an N-band model, N times as much work is required as for a gray, I-band model. In the case of the Discrete Transfer model, for small N this turns out not to be true because the tracking of the rays through the geometry is a major one-off overhead.

This model can be used in conjunction with all available radiation models.

8.6.3. Weighted Sum of Gray Gases

The radiative absorption and emission from a gas can be characterized by the emissivity as a function of temperature and *pL*, that is the product of the partial pressure and the path length. In the context of typical combustion systems, the dominant emitters of radiation are carbon dioxide and water vapor (although hydrocarbons, CO and SO_2 also make a minor contribution). Hottel and Sarofim [48] have published emissivity charts for CO_2 and H_2O that have been obtained by a combination of measurement and extrapolation. These plots show that emissivity is strongly dependent on pL and also has a weaker dependence on the gas temperature. This functional dependence can be accurately correlated by assuming that the emissivity arises as the result of independent emission from a sufficient number of gray gases:

$$\varepsilon_g = \sum_{i=1}^{N_g} a_{gi}\left(1 - e^{-k_i L p}\right) \quad \text{(8–36)}$$

Because emissivity must be proportional to absorptivity by Kirchoffs' law, it follows that ε_g must approach unity as $pL \to \infty$. This imposes a constraint on the gray gas weights or amplitudes:

$$\sum_{i=1}^{N_g} a_{gi} = 1 \tag{8–37}$$

Also the requirement that ε_g is a monotonically increasing function of pL is satisfied if all the a_{gi} are positive.

If the number of gray gases, N_g, is large, then a_{gi} may be thought of as the fraction of the energy spectrum, relative to the blackbody energy, for which the absorption coefficient is approximately k_i. Then, the methodology described for the Multiband model can be used directly.

8.6.3.1. Weighted Sum of Gray Gases Model Parameters

Hadvig [49] has published charts of emissivity of combined CO_2-H_2O mixtures, for mixtures with different relative proportions of CO_2 and H_2O. For the case of natural gas combustion, it can be shown that the proportions of water vapor and carbon dioxide in the products of combustion is such that partial pressure ratio, p_{H_2O} / p_{CO_2} is approximately equal to 2. Similarly, this ratio is 1 for oils and other fuels with the empirical formula, $(CH_2)_x$. Most other hydrocarbon fuels have combustion products with a p_{H_2O} / p_{CO_2} ratio lying between 1 and 2. Starting from the charts of Hottel and Sarofim (1967) [48] for CO_2 and H_2O and applying their correction factor for mixtures, Hadvig has evaluated the emissivity of a gas mixture with p_{H_2O} / p_{CO_2} = 1 and 2 and presented the results as a function of T_g and pL. Leckner [50] has also published emissivity data, based on integrating the measured spectral data for CO_2 and H_2O, which is in reasonable agreement with the Hottel charts where the charts are based on measured data.

Taylor and Foster (1974) [51] have integrated the spectral data and constructed a multigray gas representation:

$$\varepsilon_g = \sum_{i=1}^{4} a_{gi}\left(T_g\right) \left[1 - e^{-K_i\left(p_{H_2O} + p_{CO_2}\right) L}\right] \tag{8–38}$$

where the a_{gi} are represented as linear functions of T_g:

$$a_{gi} = b_{1i} + 10^{-5}\, b_{2i}\, T_g \tag{8–39}$$

As well as CO_2 and H_2O, the model developed by Beer, Foster and Siddall [52] takes into account the contribution of CO and unburnt hydrocarbons, e.g., methane (CH_4) which are also significant emitters of radiation. These authors generalize the parameterization of the absorption coefficients as follows:

$$K_i\left(p_{H_2O} + p_{CO_2}\right) \rightarrow K_i\left(p_{H_2O} + p_{CO_2} + p_{CO}\right) + K_{HCi}\, p_{HC} \tag{8–40}$$

where p_{CO} is the partial pressure of CO and p_{HC} is the total partial pressure of all hydrocarbon species.

The values of b_{1i}, b_{2i} [K^{-1}], K_i [m^{-1} atm^{-1}] and K_{HCi} [m^{-1} atm^{-1}] are given in *Table 8.1: Gray gas emissivity parameters for a carbon dioxide / water vapor / hydrocarbon mixture.* (p. 362), together with a similar correlation for N_g = 3, derived by Beer, Foster and Siddall [52], and suitable defaults for N_g = 2 or 1 (single gray gas) representations.

Table 8.1 Gray gas emissivity parameters for a carbon dioxide / water vapor / hydrocarbon mixture.

N_g	i	Gaseous Fuels p_{H2O}/p_{CO2} = 2				Oils p_{H2O}/p_{CO2} = 1			
		b_{1i}	b_{2i}	k_i	k_{HCi}	b_{1i}	b_{2i}	k_i	k_{HCi}
1	1	1	0	1	0	1	0	1	0
2	1	0.437	7.13	0	3.85	0.486	8.97	0	3.41
	2	0.563	-7.13	1.88	0	0.514	-8.97	2.5	0
3	1	0.437	7.13	0	3.85	0.486	8.97	0	3.41
	2	0.390	-0.52	1.88	0	0.381	-3.96	2.5	0
	3	1.173	-6.61	68.83	0	0.133	-5.01	109	0
4	1	0.364	4.74	0	3.85	0.4092	7.53	0	3.41
	2	0.266	7.19	0.69	0	0.284	2.58	0.91	0
	3	0.252	-7.41	7.4	0	0.211	-6.54	9.4	0
	4	0.118	-4.52	80	0	0.0958	-3.57	130	0

Note

To satisfy the requirement that the a_i factors sum to unity, the b_1 factors must sum to 1.0 and the b_2 factors must sum to 0.

Chapter 9: Electromagnetic Hydrodynamic Theory

Computational Fluid Dynamics is based on the conservation of mass, momentum, energy, and chemical species. Computational Electromagnetics (CEM) is based on:

- Maxwell's equations and further relations
- Material relations for polarization (permittivity), conductivity, and magnetization (permeability).

The interaction of CEM and CFD is through body forces that result due to interactions between electromagnetic fields and fluid motion.

Electromagnetic hydrodynamics is available to enable the modeling of:

- The electric potential for conducting materials (including magnetic induced currents)
- User-defined volumetric, point, and boundary sources
- Resistive heating in energy equations
- Conjugate transfer between fluid and solid domains
- Variable electrical conductivity (via CEL).

9.1. Electromagnetic Models

The solution of electromagnetic hydrodynamic problems at the continuum level requires the solution of Maxwell's equations. The fundamental equations are:

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{9–1}$$

$$\nabla \times H = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \tag{9–2}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{9–3}$$

$$\nabla \cdot \mathbf{D} = \rho_e \tag{9–4}$$

The Continuity equation (conservation of electric charge), which is a combination of the equations above, is:

$$\nabla \cdot \mathbf{J} = \frac{\partial \rho_e}{\partial t} \tag{9–5}$$

In the equations above:

- $\mathbf{B}$ is the magnetic induction
- $\mathbf{D}$ is the electric displacement
- $\mathbf{E}$ is the electric field
- $\mathbf{H}$ is the magnetic field
- $\mathbf{J}$ is the current density
- ρ_e is the electric charge density

To close this system of equations, constitutive relationships and a suitable set of boundary conditions must be included.

9.1.1. Constitutive Relationships

The electric field $\mathbf{E}$ and the electric displacement are related through the polarization vector $\mathbf{P}$ as:

$$\mathbf{D} = \varepsilon_0 \mathbf{E} + \mathbf{P} \tag{9–6}$$

where ε_0 is the permittivity of vacuum. For linear materials (typically diamagnetic or paramagnetic), the polarization vector can be written as

$$\mathbf{P} = \varepsilon_0 \chi_e \mathbf{E} \tag{9–7}$$

where χ_e is the electric susceptibility, which can be a function of the electric field $\mathbf{E}$.

A similar relationship is used to relate the magnetic induction $\mathbf{B}$ to the magnetic field $\mathbf{H}$, and the magnetization vector $\mathbf{M}$.

$$\mathbf{B} = \mu_0 (\mathbf{H} + \mathbf{M}) \tag{9–8}$$

where μ_0 is the permeability of vacuum. For linear materials, the magnetization vector $\mathbf{M}$ can written as

$$\mathbf{M} = \chi_m \mathbf{H} \tag{9–9}$$

where χ_m is the magnetic susceptibility.

Finally, for the current density $\mathbf{J}$ can be written as

$$\mathbf{J} = \sigma (\mathbf{E} + \mathbf{U} \times \mathbf{B}) \tag{9–10}$$

where σ is the electrical conductivity.

9.1.2. Magnetohydrodynamics (MHD)

Magnetohydrodynamics (MHD) is the study of the interactions between magnetic fields and electrically conducting fluids. The body (Lorentz) force acts on the fluid when electric current ($\mathbf{j}$) flows at an angle to the imposed magnetic field ($\mathbf{B}$). The resistive heating is measured in Joules.

For a particle:

$$\mathbf{F}_{emag} = q\,(\mathbf{E} + \mathbf{U} \times \mathbf{B}) \tag{9–11}$$

where q is the charge on the particle.

For a continuous fluid:

$$\mathbf{F}_{emag} = \mathbf{J} \times \mathbf{B} \tag{9–12}$$

where:

$$\mathbf{J} = \sigma\,(\mathbf{E} + \mathbf{U} \times \mathbf{B}) \tag{9–13}$$

9.1.3. Electrohydrodynamics (EHD)

Electrohydrodynamics (EHD) is the study of the interactions between magnetic fields and electrically charged fluids (species, particles). The body (Coulomb) force acts on the charged fluid along the imposed electric field ($\mathbf{E}$).

$$\mathbf{F}_{emag} = q\mathbf{E} \tag{9–14}$$

9.1.4. Ferrohydrodynamics (FHD)

Ferrohydrodynamics (FHD) is the study of interactions between magnetic fields and magnetically polarizable fluids (that is, ferrofluids). The body (Kelvin) force acts on the fluid magnetic dipoles along to the magnetic field ($\mathbf{H}$).

$$\mathbf{F}_{emag} = \mu_0\,(\mathbf{M} \cdot \nabla)\,\mathbf{H} \tag{9–15}$$

9.1.5. Electromagnetic Basics: Potential Formulation in ANSYS CFX

The electric potential (φ) for electric field ($\mathbf{E}$) is:

$$\mathbf{E} = -\nabla\varphi - \frac{\partial \mathbf{A}}{\partial t} \quad \textbf{(9–16)}$$

$$\nabla \cdot \mathbf{J} = \nabla \cdot \left[-\sigma\nabla\varphi + \sigma(\mathbf{U}\times\mathbf{B}) - \sigma\frac{\partial \mathbf{A}}{\partial t} \right] = 0 \quad \textbf{(9–17)}$$

The magnetic vector potential ($\mathbf{A}$) for magnetic induction ($\mathbf{B}$) is:

$$\mathbf{B} = \nabla\times\mathbf{A} \quad \textbf{(9–18)}$$

$$\nabla\times\mathbf{H} = \nabla\times\left[\frac{1}{\mu}\nabla\times\mathbf{A} \right] = -\sigma\nabla\varphi + \sigma(\mathbf{U}\times\mathbf{B}) - \sigma\frac{\partial \mathbf{A}}{\partial t} \quad \textbf{(9–19)}$$

9.1.6. Boundary Conditions

The boundary conditions are:

- $\mathbf{n}\times(\mathbf{E}_1 - \mathbf{E}_2) = 0$
- $\mathbf{n}\cdot(\mathbf{D}_1 - \mathbf{D}_2) = \rho_s$
- $\mathbf{n}\times(\mathbf{H}_1 - \mathbf{H}_2) = \mathbf{J}_s$
- $\mathbf{n}\cdot(\mathbf{B}_1 - \mathbf{B}_2) = 0$

9.1.7. Transformed Equations

In order to reduce the number of partial differential equations and transform them into something similar to the general transport equation within the ANSYS CFX solver, a formulation based on the electric scalar potential φ, and the magnetic vector potential $\mathbf{A}$ is used. The following vector identities are used:

$$\nabla\cdot(\nabla\times\mathbf{\Psi}) = 0 \quad \textbf{(9–20)}$$

$$\nabla\times(\nabla\Phi) = 0 \quad \textbf{(9–21)}$$

$$\nabla\times(\nabla\times\mathbf{\Psi}) = \nabla\cdot(\nabla\Psi) + \nabla(\nabla\cdot\Psi) \quad \textbf{(9–22)}$$

The electric and magnetic vector potentials are defined as:

$$\mathbf{B} = \nabla \times \mathbf{A} \tag{9–23}$$

$$\mathbf{E} = -\nabla \varphi - \frac{\partial \mathbf{A}}{\partial t} \tag{9–24}$$

Equation 9–1 (p. 363) and *Equation 9–3* (p. 363) are automatically satisfied:

$$\nabla \times \mathbf{E} = \nabla \times \left(-\nabla \varphi - \frac{\partial \mathbf{A}}{\partial t} \right) = 0 - \frac{\partial \nabla \times \mathbf{A}}{\partial t} = -\frac{\partial \mathbf{B}}{\partial t} \tag{9–25}$$

$$\nabla \cdot (\nabla \times \mathbf{A}) = 0 \tag{9–26}$$

9.1.8. Conductive Media Approximation

Starting with *Equation 9–5* (p. 363) and substituting the constitutive equations:

$$\begin{aligned} &\nabla \cdot \mathbf{J} = \nabla \cdot (\sigma (\mathbf{E} + \mathbf{U} \times \mathbf{B})) = \\ &\nabla \cdot \left(-\sigma \nabla \varphi - \sigma \frac{\partial \mathbf{A}}{\partial t} + \sigma \mathbf{U} \times \mathbf{B} \right) = \frac{\partial \rho_e}{\partial t} \end{aligned} \tag{9–27}$$

This can be rearranged as:

$$\frac{\partial \rho_e}{\partial t} - \nabla \cdot (-\sigma \nabla \varphi) = -\sigma \frac{\partial \mathbf{A}}{\partial t} + \sigma \mathbf{U} \times \mathbf{B} \tag{9–28}$$

Assuming a quasi-steady model (that is, neglecting transient terms)

$$\nabla \cdot (-\sigma \nabla (\varphi)) = -\sigma \mathbf{U} \times \mathbf{B} \tag{9–29}$$

Similarly for *Equation 9–3* (p. 363):

$$\nabla \times \mathbf{H} = \sigma (\mathbf{E} + \mathbf{U} \times \mathbf{B}) + \frac{\partial \mathbf{D}}{\partial t} \tag{9–30}$$

Using the constitutive relation between the Magnetic Induction ($\mathbf{B}$) and the Magnetic Field ($\mathbf{H}$), in addition to the quasi-steady approximation:

$$\nabla \times \left(\frac{1}{\mu} \nabla \times \mathbf{A} \right) = \sigma \left(\mathbf{E} + \mathbf{U} \times \left(\nabla \times \mathbf{A} \right) \right) \tag{9–31}$$

Using vector identities (*Equation 9–22* (p. 366)) assuming uniform magnetic permeability, and enforcing the Coulomb gauge that $\nabla \cdot \mathbf{A} = 0$, the transport equation for the magnetic vector potential becomes:

$$\nabla \cdot \left(\frac{1}{\mu} \nabla \mathbf{A} \right) = \sigma \left(\mathbf{E} + \mathbf{U} \times \left(\nabla \times \mathbf{A} \right) \right) \tag{9–32}$$

Note that the left-hand side of the transport equation for the magnetic vector potential has only transport by diffusion, which is balanced by the sources on the right-hand side.

9.2. Fluid Dynamics Model

For reference, the transport equations for the fluid dynamics model expressed in vector notation (including electromagnetic sources) are presented below.

Conservation of Mass

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \tag{9–33}$$

Conservation of Linear Momentum

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot \left(\rho \mathbf{U} \otimes \mathbf{U} \right) = -\nabla p + \nabla \cdot \tau + \mathbf{S}_M + \mathbf{F}_{emag} + \rho f \tag{9–34}$$

where $\mathbf{F}_{emag}$ denotes the sources of momentum arising from the electromagnetic forces and is given by:

$$\mathbf{F}_{emag} = q \ \mathbf{E} + \mathbf{J} \times \mathbf{B} + \mathbf{f}_{add} \tag{9–35}$$

Conservation of Energy

$$\frac{\partial (\rho h_{\mathrm{tot}})}{\partial t} - \frac{\partial p}{\partial t} + \nabla \cdot (\rho \boldsymbol{U} h_{\mathrm{tot}}) = \nabla \cdot (\lambda \nabla T) + \nabla \cdot (\boldsymbol{U} \cdot \tau) + \boldsymbol{U} \cdot \boldsymbol{S}_M + \boldsymbol{S}_E \tag{9–36}$$

Where:

- h_{tot} is the total enthalpy, related to the static enthalpy $h(T, p)$ by:

$$h_{tot} = h + \frac{1}{2} \boldsymbol{U}^2 \quad \textbf{(9–37)}$$

- The term $\nabla \cdot (\boldsymbol{U} \cdot \tau)$ represents the work due to viscous stresses and is called the viscous work term.
- The term $\boldsymbol{U} \cdot \boldsymbol{S}_M$ represents the work due to external momentum sources and is currently neglected.
- The term $\boldsymbol{S}_M$ represents the momentum source.
- The term $\mathbf{S}_E$ represents the energy source due to resistive heating and is given by $\mathbf{J} \cdot E$.

Chapter 10: Rigid Body Theory

This chapter contains a summary of the theory of rigid body and the algorithms used in ANSYS CFX:

10.1. Equations of Motion of Rigid Body

The equations of motion of a rigid body can be written as:

$$\frac{d\mathbf{P}}{dt} = \mathbf{F} \tag{10–1}$$

$$\frac{d\boldsymbol{\pi}}{dt} = \mathbf{m} \tag{10–2}$$

These equations state that for a rigid body undergoing translation and rotation the rate of change of linear and angular momentum, $\mathbf{P}$ and $\boldsymbol{\pi}$, respectively, of the rigid body is equal to the applied force and torque, $\mathbf{F}$ and $\mathbf{m}$, respectively, acting on the body.

10.2. Rigid Body Solution Algorithms

The rigid body solution algorithm uses a linear momentum solver for the translation of the rigid body and a separate angular momentum solver for the rotation of the rigid body. There are two choices for the angular momentum solver (a First Order Backward Euler solver and a Simo Wong solver).

The remainder of this section describes:

10.2.1. Translational Equations of Motion

The equation of motion for a translating center of mass $\mathbf{x}$, *Equation 10–1* (p. 371), can be expressed as:

$$m\ddot{\mathbf{x}} = \mathbf{F} \tag{10–3}$$

where m is the mass and $\mathbf{F}$ represents the sum of all forces which include aerodynamic, weight of rigid body, spring and/or explicit external force.

Expanding the sum of all forces, *Equation 10–3* (p. 371) may be written as:

$$m\ddot{\mathbf{x}} = \mathbf{F} = \mathbf{F}_{Aero} + m\mathbf{g} - k_{Spring}(\mathbf{x} - \mathbf{x}_{so}) + \mathbf{F}_{Ext} \tag{10–4}$$

where $\mathbf{F}_{Aero}$ is the aerodynamic force, $\mathbf{g}$ is the gravity, k_{Spring} is the linear spring constant and $\mathbf{F}_{Ext}$ is all other external forces acting on the body.

10.2.1.1. Newmark Integration

The linear momentum solver makes use of the Newmark integration scheme. The conventional Newmark integration scheme [203] can be expressed as:

$$h_{n+1} = h_n + \Delta t_n \dot{h}_n + \Delta t_n^2 \left(\left(\frac{1}{2} - \beta \right) \ddot{h}_n + \beta \ddot{h}_{n+1} \right) \tag{10–5}$$

$$\dot{h}_{n+1} = \dot{h}_n + \Delta t_n \left((1 - \gamma) \ddot{h}_n + \gamma \ddot{h}_{n+1} \right) \tag{10–6}$$

where t is time, h is representative of an entity such as spatial or angular position, and the subscripts n and $n+1$ represent the known solution at time t and unknown solution at time $t + \Delta t$, respectively.

The Newmark integration depends on two real parameters β and γ. These parameters are directly linked to accuracy and stability of the Newmark time integration scheme.

In typical applications of the Newmark method β and γ are chosen to be ¼ and ½, respectively. This choice of parameters corresponds to a trapezoidal rule which results in a second order accurate scheme which is also unconditionally stable in linear analyses. The linear momentum equations are solved using these parameters.

For fixed values of the external forces, the linear momentum solver advances the solution over time using the following procedure:

1. Expressing *Equation 10–4* (p. 372) as

$$m\ddot{\mathbf{x}}_{n+1} = \mathbf{F} = \mathbf{F}_{Aero} + m\mathbf{g} - k_{Linear}(\mathbf{x}_{n+1} - \mathbf{x}_{so}) + \mathbf{F}_{Ext} \tag{10–7}$$

   and substituting for $\ddot{\mathbf{x}}_{n+1}$ into *Equation 10–5* (p. 372) where h is replaced by $\mathbf{x}$ and rearranging, $\mathbf{x}_{n+1}$ is determined from:

$$\mathbf{x}_{n+1} = \frac{\frac{m}{\beta \Delta t^2}\mathbf{x}_n + \frac{m}{\beta \Delta t}\dot{\mathbf{x}}_n + m\left(\frac{1}{2\beta} - 1\right)\ddot{\mathbf{x}}_n + \mathbf{F}_{Aero} + m\mathbf{g} + k_{Linear}\mathbf{x}_{so} + \mathbf{F}_{Ext}}{\frac{m}{\beta \Delta t^2} + k_{Linear}} \tag{10–8}$$

2. From *Equation 10–5* (p. 372) where h is replaced by $\mathbf{x}$ and rearranging, $\ddot{\mathbf{x}}_{n+1}$ is determined from:

$$\ddot{\mathbf{x}}_{n+1} = \frac{1}{\beta \Delta t^2} (\mathbf{x}_{n+1} - \mathbf{x}_n) - \frac{1}{\beta \Delta t} \dot{\mathbf{x}}_n - 1 (\frac{1}{2\beta} - 1) \ddot{\mathbf{x}}_n \quad \textbf{(10–9)}$$

3. From Equation 10–6 (p. 372) where h is replaced by $\mathbf{x}$ and rearranging, $\dot{\mathbf{x}}_{n+1}$ is determined from:

$$\dot{\mathbf{x}}_{n+1} = \dot{\mathbf{x}}_n + \Delta t ((1 - \gamma) \ddot{\mathbf{x}}_n + \gamma \ddot{\mathbf{x}}_{n+1}) \quad \textbf{(10–10)}$$

   where t is time.

Note that iteration of the above procedure is required to account for dependencies of the forces on the position of the rigid body.

10.2.2. Rotational Equations of Motion

The equation of motion for a rotating rigid body can be rewritten in body-fixed coordinates (about the center of mass) as:

$$\dot{\boldsymbol{\theta}} \times \dot{\boldsymbol{\theta}} + \mathbf{I}\ddot{\boldsymbol{\theta}} = \mathbf{m} \quad \textbf{(10–11)}$$

where the mass moment of inertia matrix, $\mathbf{I}$, is defined as:

$$\mathbf{I}=\begin{bmatrix} \int((y-y_G)^2+(z-z_G)^2)\,dm & -\int(x-x_G)(y-y_G)\,dm & -\int(x-x_G)(z-z_G)\,dm \\ -\int(x-x_G)(y-y_G)\,dm & \int((x-x_G)^2+(z-z_G)^2)\,dm & -\int(y-y_G)(z-z_G)\,dm \\ -\int(x-x_G)(z-z_G)\,dm & -\int(y-y_G)(z-z_G)\,dm & \int((x-x_G)^2+(y-y_G)^2)\,dm \end{bmatrix} \quad \textbf{(10–12)}$$

In the mass moment of inertia matrix shown in Equation 10–12 (p. 373), the center of mass is given by (x_G, y_G, z_G), and dm is a differential element of mass.

Equation 10–11 (p. 373) represents the spatial coordinate form for the classical Euler's equation for the rigid body in body-fixed coordinates.

Also, in Equation 10–11 (p. 373) $\mathbf{m}$ is the total moment from all the separate contributions including spring and other external moments:

$$\mathbf{m} = \mathbf{m}_{Aero} - k_{Rotate} (\boldsymbol{\theta} - \boldsymbol{\theta}_{so}) + \mathbf{m}_{Ext} \quad \textbf{(10–13)}$$

where $\mathbf{m}_{Aero}$ is the aerodynamic torque, k_{Rotate} is the rotational spring constant and $\mathbf{m}_{Ext}$ is all other external torques acting on the body.

10.2.2.1. First Order Backward Euler

From Equation 10–11 (p. 373) the first order backward Euler algorithm begins with:

$$\ddot{\boldsymbol{\theta}}_{n+1} = \mathbf{I}^{-1}(\mathbf{m}_{n+1} - \dot{\boldsymbol{\theta}}_n \times \mathbf{I}\dot{\boldsymbol{\theta}}_n) \tag{10–14}$$

Using the first order backward Euler algorithm, $\dot{\boldsymbol{\theta}}_{n+1}$ is given by:

$$\dot{\boldsymbol{\theta}}_{n+1} = \ddot{\boldsymbol{\theta}}_{n+1}\Delta t + \dot{\boldsymbol{\theta}}_n \tag{10–15}$$

where $\ddot{\boldsymbol{\theta}}_{n+1}$ is determined from *Equation 10–14* (p. 374)

Equation 10–15 (p. 374) can be expressed in terms of $\boldsymbol{\phi}$, angles in the global frame (see 206), as:

$$\dot{\boldsymbol{\phi}}_{n+1} = \boldsymbol{\Gamma}^{-1}\dot{\boldsymbol{\theta}}_{n+1} \tag{10–16}$$

where

$$\boldsymbol{\Gamma} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \tag{10–17}$$

and where θ and ϕ are **Euler Angle Y** and **Euler Angle X**, respectively. For details on Euler Angles see Rigid Body Motion in the *CFX-Solver Modeling Guide*.

Integrating in global coordinates using the first order backward Euler algorithm:

$$\boldsymbol{\phi}_{n+1} = \Delta t\dot{\boldsymbol{\phi}}_{n+1} + \boldsymbol{\phi}_n \tag{10–18}$$

where $\dot{\boldsymbol{\phi}}_{n+1}$ is determined from *Equation 10–16* (p. 374)

Note that in addition to providing only a first order integration of the rotational momentum the first order backward Euler method does not exactly conserve total angular momentum and is subject to potential of encountering Gimbal lock [205].

10.2.2.2. Simo Wong Algorithm

The Simo Wong (ALGO_C1 variant) algorithm is a second order time stepping algorithm that exactly conserves energy and enforces conservation of total angular momentum. It uses a modified Newmark integration scheme with constants of the Newmark method given by $\beta = ½$ and $\gamma = 1$ to realize the second order accurate integration. A Quaternion representation is employed to avoid the potential of Gimbal lock and the use of convected incremental body fixed coordinates is adopted to ensure conservation of total angular momentum. For more information, see Simo Wong [204].

Chapter 11: Discretization and Solution Theory

This chapter describes:

11.1. Numerical Discretization

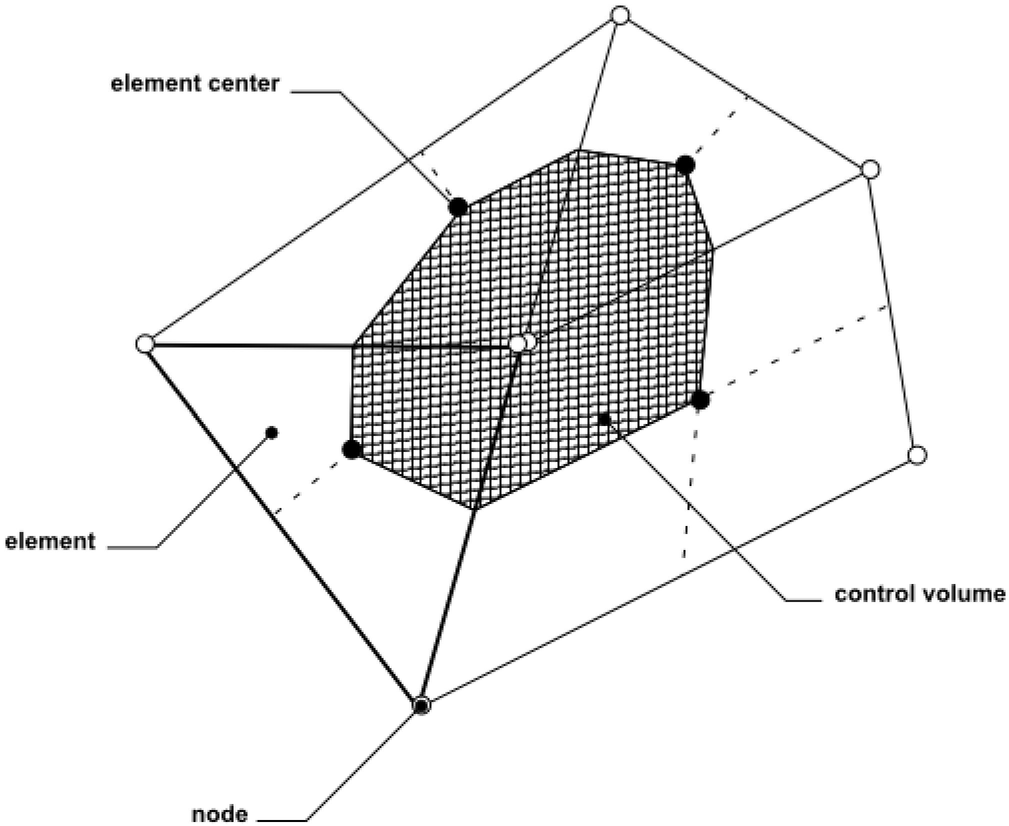
Analytical solutions to the Navier-Stokes equations exist for only the simplest of flows under ideal conditions. To obtain solutions for real flows, a numerical approach must be adopted whereby the equations are replaced by algebraic approximations that can be solved using a numerical method.

11.1.1. Discretization of the Governing Equations

ANSYS CFX uses an element-based finite volume method, which first involves discretizing the spatial domain using a mesh. The mesh is used to construct finite volumes, which are used to conserve relevant quantities such as mass, momentum, and energy. The mesh is three dimensional, but for simplicity we will illustrate this process for two dimensions.

The figure below shows a typical two-dimensional mesh. All solution variables and fluid properties are stored at the nodes (mesh vertices). A control volume (the shaded area) is constructed around each mesh node using the median dual (defined by lines joining the centers of the edges and element centers surrounding the node).

Figure 11.1 Control Volume Definition



To illustrate the finite volume methodology, consider the conservation equations for mass, momentum, and a passive scalar, expressed in Cartesian coordinates:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial \boldsymbol{x}_j}\left(\rho \boldsymbol{U}_j\right) = 0 \tag{11–1}$$

$$\frac{\partial}{\partial t}(\rho \boldsymbol{U}_i) + \frac{\partial}{\partial \boldsymbol{x}_j}\left(\rho \boldsymbol{U}_j \boldsymbol{U}_i\right) = -\frac{\partial P}{\partial \boldsymbol{x}_i} + \frac{\partial}{\partial \boldsymbol{x}_j}\left(\mu_{eff}\left(\frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{U}_j}{\partial \boldsymbol{x}_i}\right)\right) \tag{11–2}$$

$$\frac{\partial}{\partial t}(\rho \varphi) + \frac{\partial}{\partial \boldsymbol{x}_j}\left(\rho \boldsymbol{U}_j \varphi\right) = \frac{\partial}{\partial \boldsymbol{x}_j}\left(\Gamma_{eff}\left(\frac{\partial \varphi}{\partial \boldsymbol{x}_j}\right)\right) + S_\varphi \tag{11–3}$$

These equations are integrated over each control volume, and Gauss' Divergence Theorem is applied to convert volume integrals involving divergence and gradient operators to surface integrals. If control volumes do not deform in time, then the time derivatives can be moved outside of the volume integrals and the integrated equations become:

$$\frac{d}{dt}\int_V \rho\, dV + \int_s \rho\, \boldsymbol{U}_j\, dn_j = 0 \tag{11–4}$$

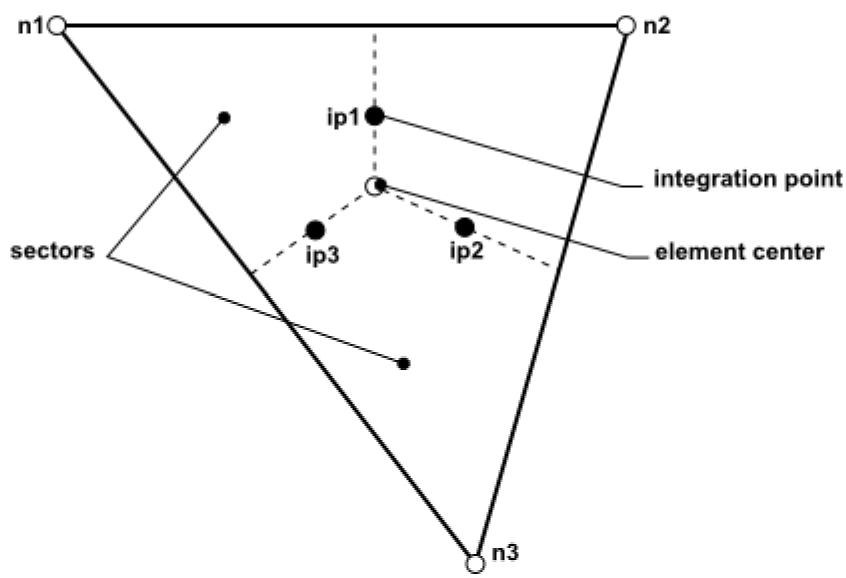
$$\frac{d}{dt}\int_V \rho\, \boldsymbol{U}_i\, dV + \int_s \rho\, \boldsymbol{U}_j\, \boldsymbol{U}_i\, dn_j = -\int_s P\, dn_j + \int_s \mu_{eff}\left(\frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{U}_j}{\partial \boldsymbol{x}_i}\right) dn_j + \int_V S_{\boldsymbol{U}_i}\, dV \tag{11–5}$$

$$\frac{d}{dt}\int_V \rho\, \varphi\, dV + \int_s \rho\, \boldsymbol{U}_j\, \varphi\, dn_j = \int_s \Gamma_{eff}\left(\frac{\partial \varphi}{\partial \boldsymbol{x}_j}\right) dn_j + \int_V S_\varphi\, dV \tag{11–6}$$

where *V* and *s* respectively denote volume and surface regions of integration, and dn_j are the differential Cartesian components of the outward normal surface vector. The volume integrals represent source or accumulation terms, and the surface integrals represent the summation of the fluxes. Note that changes to these equations need some generalization to account for mesh deformation. For details, see *Mesh Deformation* (p. 388).

The next step in the numerical algorithm is to discretize the volume and surface integrals. To illustrate this step, consider a single element like the one shown below.

Figure 11.2 Mesh Element



Volume integrals are discretized within each element sector and accumulated to the control volume to which the sector belongs. Surface integrals are discretized at the integration points (*ipn*) located at the center of each surface segment within an element and then distributed to the adjacent control volumes. Because the surface integrals are equal and opposite for control volumes adjacent to the integration points, the surface integrals are guaranteed to be locally conservative.

After discretizing the volume and surface integrals, the integral equations become:

$$V\left(\frac{\rho-\rho^{o}}{\Delta t}\right)+\sum_{ip}\dot{m}_{ip}=0 \tag{11–7}$$

$$V\left(\frac{\rho\,\boldsymbol{U}_i-\rho^{o}\,\boldsymbol{U}_i^{o}}{\Delta t}\right)+\sum_{ip}\dot{m}_{ip}\,(\boldsymbol{U}_i)_{ip}=\sum_{ip}(P\,\Delta n_i)_{ip}+\sum_{ip}\left(\mu_{eff}\left(\frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j}+\frac{\partial \boldsymbol{U}_j}{\partial \boldsymbol{x}_i}\right)\Delta n_j\right)_{ip}+\overline{S_{\boldsymbol{U}_i}}\,V \tag{11–8}$$

$$V\left(\frac{\rho\,\varphi-\rho^{o}\,\varphi^{o}}{\Delta t}\right)+\sum_{ip}\dot{m}_{ip}\,\varphi_{ip}=\sum_{ip}\left(\Gamma_{eff}\frac{\partial \varphi}{\partial \boldsymbol{x}_j}\Delta n_j\right)_{ip}+\overline{S_{\varphi}}\,V \tag{11–9}$$

where $\dot{m}_{ip}=\left(\rho\,\boldsymbol{U}_j\,\Delta n_j\right)_{ip}$, V is the control volume, Δt is the time step, Δn_j is the discrete outward surface vector, the subscript *ip* denotes evaluation at an integration point, summations are over all the integration points of the control volume, and the superscript ^{o} refers to the old time level. Note that the First Order Backward Euler scheme has been assumed in these equations, although a second order scheme (discussed later) is usually preferable for transient accuracy.

11.1.1.1. Order Accuracy

Many discrete approximations developed for CFD are based on series expansion approximations of continuous functions (such as the Taylor series). The order accuracy of the approximation is determined by the exponent on the mesh spacing or time step factor of the largest term in the truncated part of the series expansion, which is the first term excluded from the approximation. Increasing the order-accuracy of an approximation generally implies that errors are reduced more quickly with mesh or time step size refinement. Unfortunately, in addition to increasing the computational load, high-order approximations are also generally less robust (that is, less numerically stable) than their low-order counterparts. ANSYS CFX uses second order accurate approximations as much as possible. The role of error is discussed further in *Discretization Errors* (p. 394).

11.1.1.2. Shape Functions

Solution fields and other properties are stored at the mesh nodes. However, to evaluate many of the terms, the solution field or solution gradients must be approximated at integration points. ANSYS CFX uses finite-element shape functions (unless otherwise noted) to perform these approximations. Finite-element shape functions describe the variation of a variable φ varies within an element as follows:

$$\varphi=\sum_{i=1}^{N_{\mathrm{node}}}N_i\,\varphi_i \tag{11–10}$$

where N_i is the shape function for node *i* and φ_i is the value of φ at node *i*. The summation is over all nodes of an element. Key properties of shape functions include:

$$\sum_{i=1}^{N_{\text{node}}} N_i = 1 \tag{11–11}$$

$$\text{At node j,} \quad N_i = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{11–12}$$

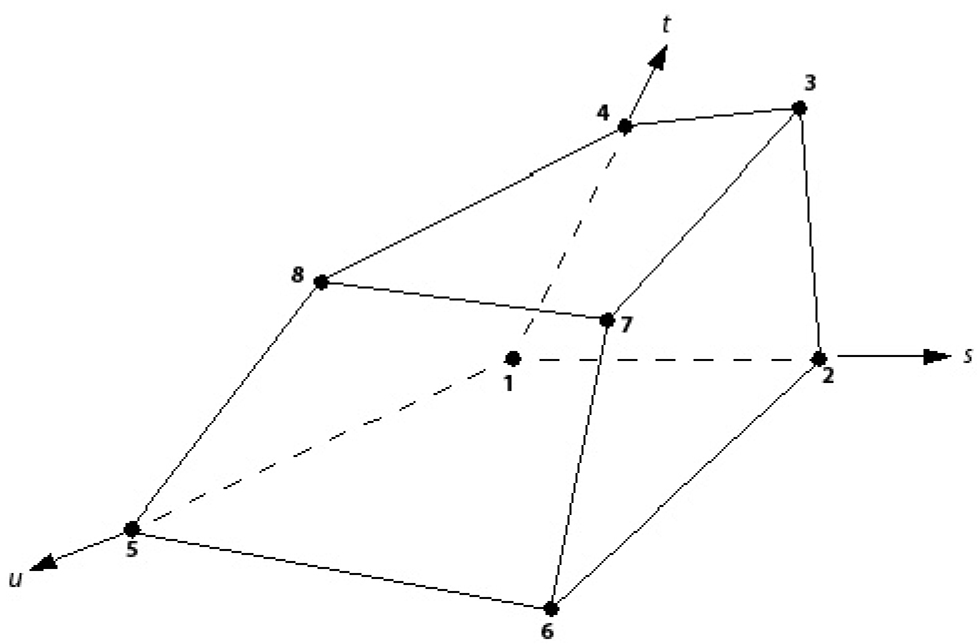
The shape functions used in ANSYS CFX are linear in terms of parametric coordinates. They are used to calculate various geometric quantities as well, including ip coordinates and surface area vectors. This is possible because *Equation 11–10* (p. 378) also holds for the coordinates:

$$y = \sum_{i=1}^{N_{\text{node}}} N_i y_i \tag{11–13}$$

The tri-linear shape functions for each supported mesh element are given below:

11.1.1.2.1. Hexahedral Element

Figure 11.3 Hexahedral Element

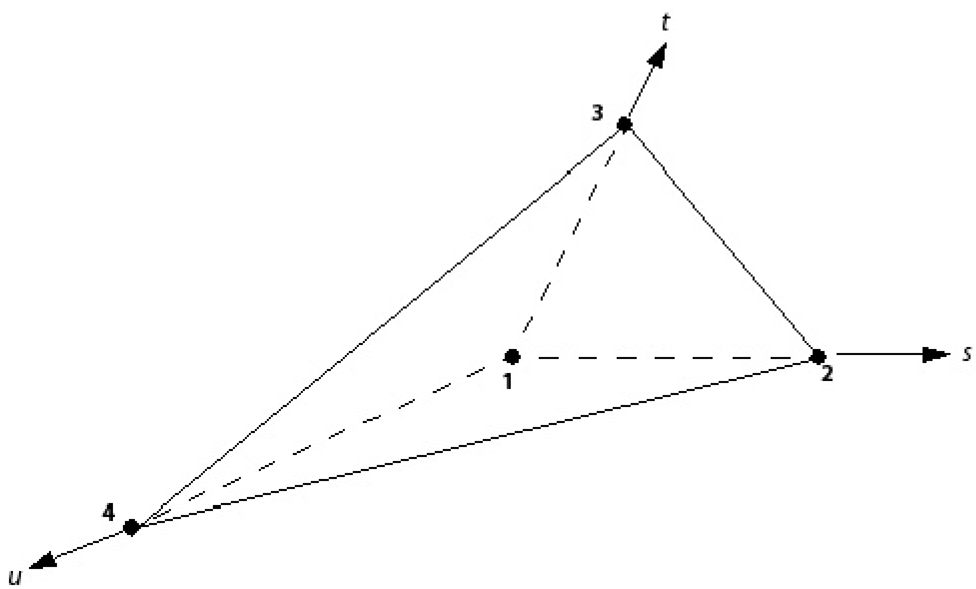


The tri-linear shape functions for the nodes are:

$$
\begin{aligned}
N_1(s,t,u) &= (1-s)\,(1-t)\,(1-u) \\
N_2(s,t,u) &= s\,(1-t)\,(1-u) \\
N_3(s,t,u) &= s\,t\,(1-u) \\
N_4(s,t,u) &= (1-s)\,t\,(1-u) \\
N_5(s,t,u) &= (1-s)\,(1-t)\,u \\
N_6(s,t,u) &= s\,(1-t)\,u \\
N_7(s,t,u) &= s\,t\,u \\
N_8(s,t,u) &= (1-s)\,t\,u
\end{aligned}
\tag{11–14}
$$

11.1.1.2.2. Tetrahedral Element

Figure 11.4 Tetrahedral Element

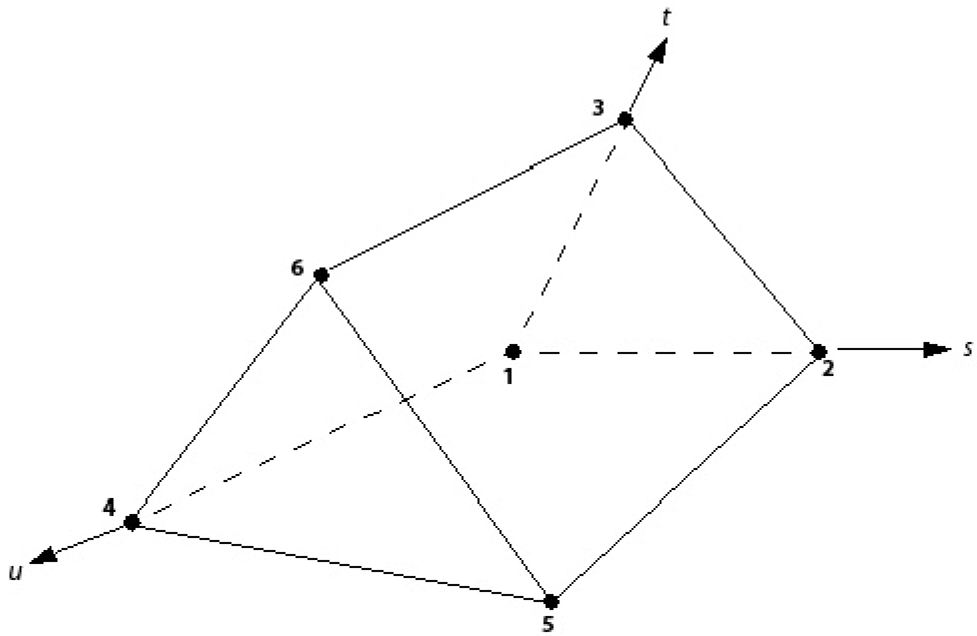


The tri-linear shape functions for the nodes are:

$$N_1(s,t,u) = 1 - s - t - u$$
$$N_2(s,t,u) = s$$
$$N_3(s,t,u) = t \tag{11–15}$$
$$N_4(s,t,u) = u$$

11.1.1.2.3. Wedge Element

Figure 11.5 Wedge Element

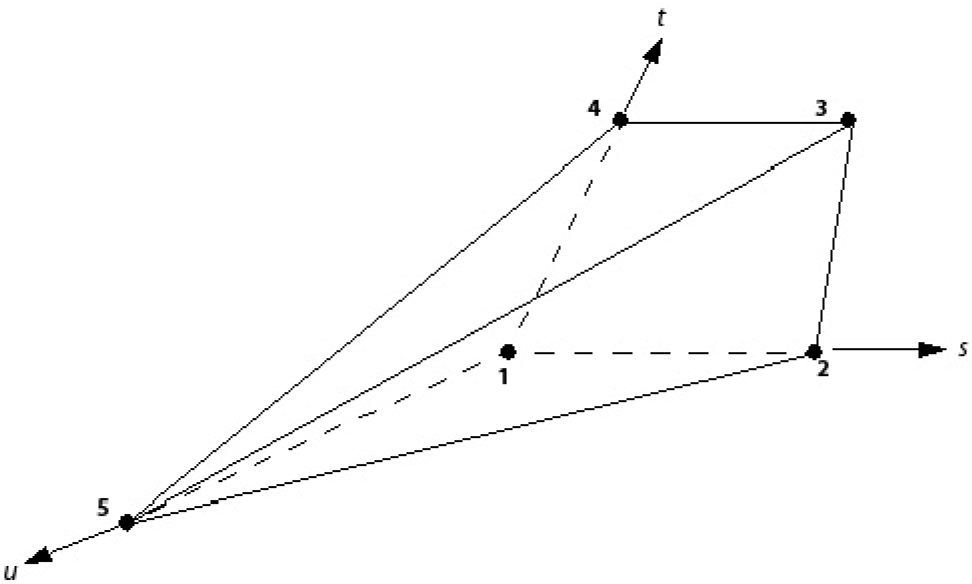


The tri-linear shape functions for the nodes are:

$$
\begin{aligned}
N_1(s,t,u) &= (1-s-t)(1-u) \\
N_2(s,t,u) &= s(1-u) \\
N_3(s,t,u) &= t(1-u) \\
N_4(s,t,u) &= (1-s-t)\,u \\
N_5(s,t,u) &= s\,u \\
N_6(s,t,u) &= t\,u
\end{aligned}
\tag{11–16}
$$

11.1.1.2.4. Pyramid Element

Figure 11.6 Pyramid Element



The tri-linear shape functions for the nodes are:

$$
\begin{aligned}
N_1(s,t,u) &= (1-s)(1-t)(1-u) \\
N_2(s,t,u) &= s(1-t)(1-u) \\
N_3(s,t,u) &= s\,t\,(1-u) \\
N_4(s,t,u) &= (1-s)\,t\,(1-u) \\
N_5(s,t,u) &= u
\end{aligned}
\tag{11–17}
$$

11.1.1.3. Control Volume Gradients

In a few situations, gradients are required at nodes. ANSYS CFX uses a form of the Gauss' divergence theorem to evaluate these control volume gradients:

$$\nabla \varphi = \frac{1}{V} \sum_{ip} (\varphi \, \Delta \vec{n})_{ip} \tag{11–18}$$

where $\Delta \vec{n}$ is the outward surface vector at *ip*.

This formula requires that φ be evaluated at integration points using finite-element shape functions.

11.1.1.4. Advection Term

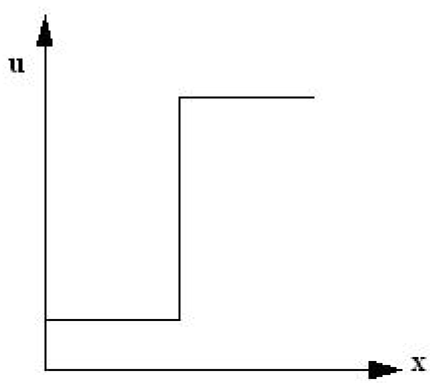
The advection term requires the integration point values of φ to be approximated in terms of the nodal values of φ. The advection schemes implemented in ANSYS CFX can be cast in the form:

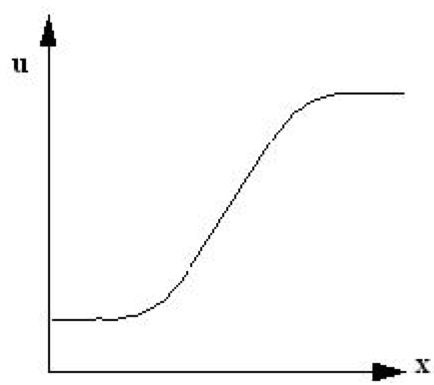
$$\varphi_{ip} = \varphi_{up} + \beta \, \nabla \varphi \cdot \Delta \vec{r} \tag{11–19}$$

where φ_{up} is the value at the upwind node, and $\vec{r}$ is the vector from the upwind node to the *ip*. Particular choices for β and $\nabla \varphi$ yield different schemes as described below.

11.1.1.4.1. 1st Order Upwind Differencing Scheme

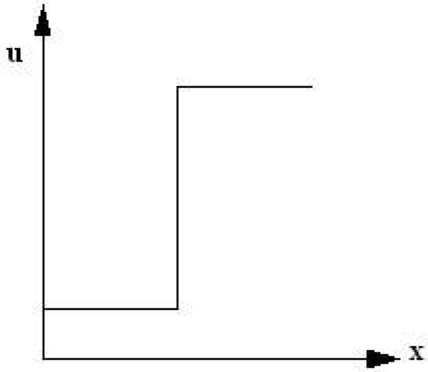
A value of $\beta = 0$ yields a first order Upwind Difference Scheme (UDS). This scheme is very robust, but it will introduce diffusive discretization errors that tend to smear steep spatial gradients as shown below:

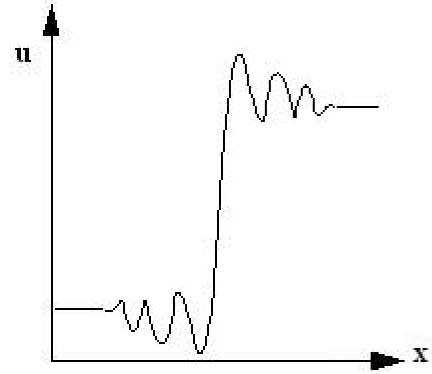




11.1.1.4.2. Specified Blend Factor

By choosing a value for β between 0 and 1, and by setting $\nabla \varphi$ equal to the average of the adjacent nodal gradients, the discretization errors associated with the UDS are reduced. The quantity $\beta \, \nabla \varphi \cdot \Delta \vec{r}$, called the Numerical Advection Correction, may be viewed as an anti-diffusive correction applied to the upwind scheme. The choice $\beta = 1$ is formally second-order-accurate in space, and the resulting discretization will more accurately reproduce steep spatial gradients than first order UDS. However, it is unbounded, and may introduce dispersive discretization errors that tend to cause non-physical oscillations in regions of rapid solution variation as shown below.





11.1.1.4.3. Central Difference Scheme

With the central difference scheme (CDS), β is set to 1 and $\nabla\varphi$ is set to the local element gradient. An alternative interpretation is that φ_{ip} is evaluated using the tri-linear shape functions:

$$\varphi_{ip} = \sum_n N_n \left(s_{ip}, t_{ip}, u_{ip} \right) \varphi_n \qquad \textbf{(11–20)}$$

The resulting scheme is also second-order-accurate, and shares the unbounded and dispersive properties of the Specified Blend Factor scheme. An additional undesirable attribute is that CDS may suffer from serious decoupling issues. While use of this scheme is not generally recommended, it has proven both useful for LES-based turbulence models.

11.1.1.4.4. Bounded Central Difference Scheme

The central differencing scheme described above is an ideal choice in view of its low numerical diffusion. However, it often leads to unphysical oscillations in the solution fields. In order to avoid these oscillations, the bounded central difference (BCD) scheme can be used as the advection scheme.

The bounded central difference scheme is essentially based on the normalized variable diagram (NVD) approach [213] together with the convection boundedness criterion (CBC) [214]. It uses the central difference scheme wherever possible, but blends to the first-order upwind scheme when the CBC is violated.

The bounded central difference scheme is suitable both for the LES and the SAS or DES turbulence models. For SAS and DES, the bounded central difference scheme then replaces the numerical blending between the central differencing scheme and the specified differencing scheme. See *Discretization of the Advection Terms* (p. 134) for more details regarding the numerical blending.

11.1.1.4.5. High Resolution Scheme

The High Resolution Scheme uses a special nonlinear recipe for β at each node, computed to be as close to 1 as possible without introducing new extrema. The advective flux is then evaluated using the values of β and $\nabla\varphi$ from the upwind node. The recipe for β is based on the boundedness principles used by Barth and Jesperson [28]. This methodology involves first computing a $\varphi_{\min}$ and $\varphi_{\max}$ at each node using a stencil involving adjacent nodes (including the node itself). Next, for each integration point around the node, the following equation is solved for β to ensure that it does not undershoot $\varphi_{\min}$ or overshoot $\varphi_{\max}$:

$$\varphi_{ip} = \varphi_{up} + \beta \ \nabla \varphi \cdot \Delta \vec{r} \tag{11–21}$$

The nodal value for β is taken to be the minimum value of all integration point values surrounding the node. The value of β is also not permitted to exceed 1. This algorithm can be shown to be Total Variation Diminishing (TVD) when applied to one-dimensional situations.

11.1.1.5. Diffusion Terms

Following the standard finite-element approach, shape functions are used to evaluate spatial derivatives for all the diffusion terms. For example, for a derivative in the x direction at integration point *ip*:

$$\frac{\partial \varphi}{\partial x}\Big|_{ip} = \sum_n \frac{\partial N_n}{\partial x}\Big|_{ip} \ \varphi_n \tag{11–22}$$

The summation is over all the shape functions for the element. The Cartesian derivatives of the shape functions can be expressed in terms of their local derivatives via the Jacobian transformation matrix:

$$\begin{bmatrix} \dfrac{\partial N}{\partial x} \\ \dfrac{\partial N}{\partial y} \\ \dfrac{\partial N}{\partial z} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial s} & \dfrac{\partial y}{\partial s} & \dfrac{\partial z}{\partial s} \\ \dfrac{\partial x}{\partial t} & \dfrac{\partial y}{\partial t} & \dfrac{\partial z}{\partial t} \\ \dfrac{\partial x}{\partial u} & \dfrac{\partial y}{\partial u} & \dfrac{\partial z}{\partial u} \end{bmatrix}^{-1} \begin{bmatrix} \dfrac{\partial N}{\partial s} \\ \dfrac{\partial N}{\partial t} \\ \dfrac{\partial N}{\partial u} \end{bmatrix} \tag{11–23}$$

The shape function gradients can be evaluated at the actual location of each integration point (that is, true tri-linear interpolation), or at the location where each *ip* surface intersects the element edge (that is, linear-linear interpolation). The latter formulation improves solution robustness at the expense of locally reducing the spatial order-accuracy of the discrete approximation.

11.1.1.6. Pressure Gradient Term

The surface integration of the pressure gradient in the momentum equations involves evaluation of the expression:

$$\left(P \ \Delta n_{ip}\right)_{ip} \tag{11–24}$$

The value of P_{ip} is evaluated using the shape functions:

$$P_{ip} = \sum_n N_n \left(s_{ip}, t_{ip}, u_{ip}\right) \ P_n \tag{11–25}$$

As with the diffusion terms, the shape function used to interpolate P can be evaluated at the actual location of each integration point (that is, true tri-linear interpolation), or at the location where each ip surface intersects the element edge (that is, linear-linear interpolation). By default, linear-linear interpolation is used unless the flow involves buoyancy, in which case tri-linear interpolation is used for improved accuracy.

11.1.1.7. Mass Flows

The discrete mass flow through a surface of the control volume, denoted by $\dot{m}_{ip}$, is given by:

$$\dot{m}_{ip} = \left(\rho\, \boldsymbol{U}_j\, \Delta n_j\right)_{ip} \tag{11–26}$$

This expression must be discretized carefully to lead to proper pressure-velocity coupling and to accurately handle the effects of compressibility, as discussed below.

11.1.1.7.1. Pressure-Velocity Coupling

ANSYS CFX uses a co-located (non-staggered) grid layout such that the control volumes are identical for all transport equations. As discussed by Patankar [118], however, naïve co-located methods lead to a decoupled (checkerboard) pressure field. Rhie and Chow [2] proposed an alternative discretization for the mass flows to avoid the decoupling, and this discretization was modified by Majumdar [119] to remove the dependence of the steady-state solution on the time step.

A similar strategy is adopted in ANSYS CFX. By applying a momentum-like equation to each integration point, the following expression for the advecting (mass-carrying) velocity at each integration point is obtained:

$$\boldsymbol{U}_{i,ip} = \overline{\boldsymbol{U}}_{i,ip} + f_{ip}\left(\frac{\partial p}{\partial x_i}\bigg|_{ip} - \overline{\frac{\partial p}{\partial x_i}}\bigg|_{ip}\right) - c_{ip} f_{ip}\left(\boldsymbol{U}^{o}_{i,ip} - \overline{\boldsymbol{U}}^{o}_{i,ip}\right) \tag{11–27}$$

$$f_{ip} = \frac{d_{ip}}{1 - c_{ip}\, d_{ip}} \tag{11–28}$$

$$d_{ip} = -\frac{V}{A} \tag{11–29}$$

A = approximation to the central coefficient of momentum equation, excluding the transient term

$$c_{ip} = \frac{\rho}{\Delta t} \tag{11–30}$$

The overbars indicate averaging of adjacent vertex values to the integration point, while the ^{o} superscript denotes values at the previous time step.

The naïve discretization, given simply by averaging the adjacent vertex velocities to the integration point, is augmented by a high-order pressure variation that scales with the mesh spacing. In particular, when substituted into the continuity equation, the expression

$$f_{ip}\left(\frac{\partial p}{\partial x_i}\Big|_{ip} - \overline{\frac{\partial p}{\partial x_i}}\Big|_{ip}\right) \quad \textbf{(11–31)}$$

becomes a fourth derivative of pressure that scales with $(\Delta x)^3$. This expression represents a spatially third-order accurate term, and is sometimes also called the pressure-redistribution term. This term is usually significantly smaller than the average of vertex velocities, especially as the mesh is refined to reasonable levels.

In some cases, the pressure-redistribution term can produce apparently significant spurious velocity fields. This may occur when a strong pressure gradient is required to balance a body force, S_i, such as buoyancy or porous drag. In these cases, the Rhie Chow discretization may lead to velocity wiggles at locations where the body force is discontinuous (for example., at a free surface interface, or at the boundary of a porous region). The wiggles are greatly reduced or eliminated by redistributing the body force as follows:

$$f_{ip}\left(\left(\frac{\partial p}{\partial x_i} - S_i\right)_{ip} - \left(\overline{\frac{\partial p}{\partial x_i}} - \overline{S}_i\right)_{ip}\right) \quad \textbf{(11–32)}$$

11.1.1.7.2. Compressibility

Mass flow terms in the mass conservation equation involve a product of the density and the advecting velocity. For compressible flows, the discretization of this product is made as implicit as possible with the use of the following Newton-Raphson linearization:

$$(\rho \boldsymbol{U})^n A \approx \rho^n \boldsymbol{U}^o A + \rho^o \boldsymbol{U}^n A - \rho^o \boldsymbol{U}^o A \quad \textbf{(11–33)}$$

Here, the superscripts n and o respectively indicate the current and previous iterates. This results in an active linearization involving both the new density and velocity terms for compressible flows at any Mach number. The value of ρ^n is linearized in terms of pressure as

$$\rho^n = \rho^o + \frac{\partial \rho}{\partial p}\Big|_T (p^n - p^o) \quad \textbf{(11–34)}$$

11.1.1.8. Transient Term

For control volumes that do not deform in time, the general discrete approximation of the transient term for the nth time step is:

$$\frac{\partial}{\partial t}\int_V \rho \varphi \, dV \approx V \, \frac{(\rho \varphi)^{n+\frac{1}{2}} - (\rho \varphi)^{n-\frac{1}{2}}}{\Delta t} \quad \textbf{(11–35)}$$

where values at the start and end of the time step are assigned the superscripts $^{n+½}$ and $^{n-½}$, respectively.

With the First Order Backward Euler scheme, the start and end of time step values are respectively approximated using the old and current time level solution values. The resulting discretization is:

$$\frac{\partial}{\partial t}\int_V \rho\,\varphi\,dV = V\left(\frac{\rho\,\varphi-\rho^{o}\,\varphi^{o}}{\Delta t}\right) \tag{11–36}$$

It is robust, fully implicit, bounded, conservative in time, and does not have a time step size limitation. This discretization is, however, only first-order accurate in time and will introduce discretization errors that tend to diffuse steep temporal gradients. This behavior is similar to the numerical diffusion experienced with the Upwind Difference Scheme for discretizing the advection term.

With the Second Order Backward Euler scheme, the start and end of time step values are respectively approximated as:

$$(\rho\,\varphi)^{n-\frac{1}{2}} = (\rho\,\varphi)^{o} + \frac{1}{2}\left((\rho\,\varphi)^{o} - (\rho\,\varphi)^{oo}\right) \tag{11–37}$$

$$(\rho\,\varphi)^{n+\frac{1}{2}} = (\rho\,\varphi) + \frac{1}{2}\left((\rho\,\varphi) - (\rho\,\varphi)^{o}\right) \tag{11–38}$$

When these values are substituted into the general discrete approximation, *Equation 11–35* (p. 387), the resulting discretization is:

$$\frac{\partial}{\partial t}\int_V \rho\,\varphi\,dV \approx V\,\frac{1}{\Delta t}\left(\frac{3}{2}(\rho\,\varphi) - 2(\rho\,\varphi)^{o} + \frac{1}{2}(\rho\,\varphi)^{oo}\right) \tag{11–39}$$

This scheme is also robust, implicit, conservative in time, and does not have a time step size limitation. It is second-order accurate in time, but is not bounded and may create some nonphysical solution oscillations. For quantities such as volume fractions, where boundedness is important, a modified Second Order Backward Euler scheme is used instead.

11.1.1.9. Mesh Deformation

The integral conservation equations presented above in *Equation 11–4* (p. 377) must be modified when the control volumes deform in time. These modifications follow from the application of the Leibnitz Rule:

$$\frac{d}{dt}\int_{V(t)} \varphi\,dV = \int_V \frac{\partial \varphi}{\partial t}\,dV + \int_s \varphi\,\boldsymbol{W}_j\,dn_j \tag{11–40}$$

where $\boldsymbol{W}_j$ is the velocity of the control volume boundary.

As before, the differential conservation equations are integrated over a given control volume. At this juncture, the Leibnitz Rule is applied, and the integral conservation equations become:

$$\frac{d}{dt}\int_{V(t)} \rho \, dV + \int_{s} \rho \left(\boldsymbol{U}_j - \boldsymbol{W}_j \right) dn_j = 0 \tag{11–41}$$

$$\begin{aligned} &\frac{d}{dt}\int_{V(t)} \rho \, \boldsymbol{U}_i \, dV + \int_{s} \rho \left(\boldsymbol{U}_j - \boldsymbol{W}_j \right) \boldsymbol{U}_i \, dn_j \\ &= -\int_{s} P \, dn_j + \int_{s} \mu_{eff} \left(\frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{U}_j}{\partial \boldsymbol{x}_i} \right) dn_j + \int_{V} S_{\boldsymbol{U}_i} \, dV \end{aligned} \tag{11–42}$$

$$\frac{d}{dt}\int_{V(t)} \rho \, \varphi \, dV + \int_{s} \rho \left(\boldsymbol{U}_j - \boldsymbol{W}_j \right) \varphi \, dn_j = \int_{s} \Gamma_{eff} \left(\frac{\partial \varphi}{\partial \boldsymbol{x}_j} \right) dn_j + \int_{V} S_{\varphi} \, dV \tag{11–43}$$

The transient term accounts for the rate of change of storage in the deforming control volume, and the advection term accounts for the net advective transport across the control volume's moving boundaries.

Erroneous sources of conserved quantities will result if the Geometric Conservation Law (GCL):

$$\frac{d}{dt}\int_{V(t)} dV = \int_{s} \boldsymbol{W}_j \, dn_j \tag{11–44}$$

is not satisfied by the discretized transient and advection terms. The GCL simply states that for each control volume, the rate of change of volume must exactly balance the net volume swept due to the motion of its boundaries. The GCL is satisfied by using the same volume recipes for both the control volume and swept volume calculations, rather than by approximating the swept volumes using the mesh velocities.

11.1.2. The Coupled System of Equations

The linear set of equations that arise by applying the finite volume method to all elements in the domain are discrete conservation equations. The system of equations can be written in the form:

$$\sum_{nb_i} a_i^{nb} \, \varphi_i^{nb} = b_i \tag{11–45}$$

where φ is the solution, b the right hand side, a the coefficients of the equation, i is the identifying number of the control volume or node in question, and nb means "neighbor", but also includes the central coefficient multiplying the solution at the ith location. The node may have any number of such neighbors, so that the method is equally applicable to both structured and unstructured meshes. The set of these, for all control volumes constitutes the whole linear equation system. For a scalar equation (for example, enthalpy or turbulent kinetic energy), a_i^{nb}, φ_i^{nb} and b_i are each single numbers. For the coupled, 3D mass-momentum equation set, they are a (4 x 4) matrix or a (4 x 1) vector, which can be expressed as:

$$a_i^{nb} = \begin{bmatrix} a_{uu} & a_{uv} & a_{uw} & a_{up} \\ a_{vu} & a_{vv} & a_{vw} & a_{vp} \\ a_{wu} & a_{wv} & a_{ww} & a_{wp} \\ a_{pu} & a_{pv} & a_{pw} & a_{pp} \end{bmatrix}_i^{nb} \tag{11–46}$$

and

$$\varphi_i^{nb} = \begin{bmatrix} u \\ v \\ w \\ p \end{bmatrix}_i^{nb} \tag{11–47}$$

$$b_i = \begin{bmatrix} b_u \\ b_v \\ b_w \\ b_p \end{bmatrix}_i \tag{11–48}$$

It is at the equation level that the coupling in question is retained and at no point are any of the rows of the matrix treated any differently (for example, different solution algorithms for momentum versus mass). The advantages of such a coupled treatment over a non-coupled or segregated approach are several: robustness, efficiency, generality, and simplicity. These advantages all combine to make the coupled solver an extremely powerful feature of any CFD code. The principal drawback is the high storage needed for all the coefficients.

11.2. Solution Strategy - The Coupled Solver

Segregated solvers employ a solution strategy where the momentum equations are first solved, using a guessed pressure, and an equation for a pressure correction is obtained. Because of the 'guess-and-correct' nature of the linear system, a large number of iterations are typically required in addition to the need for judiciously selecting relaxation parameters for the variables.

ANSYS CFX uses a coupled solver, which solves the hydrodynamic equations (for u, v, w, p) as a single system. This solution approach uses a fully implicit discretization of the equations at any given time step. For steady state problems, the time-step behaves like an 'acceleration parameter', to guide the approximate solutions in a physically based manner to a steady-state solution. This reduces the number of iterations required for convergence to a steady state, or to calculate the solution for each time step in a time-dependent analysis.

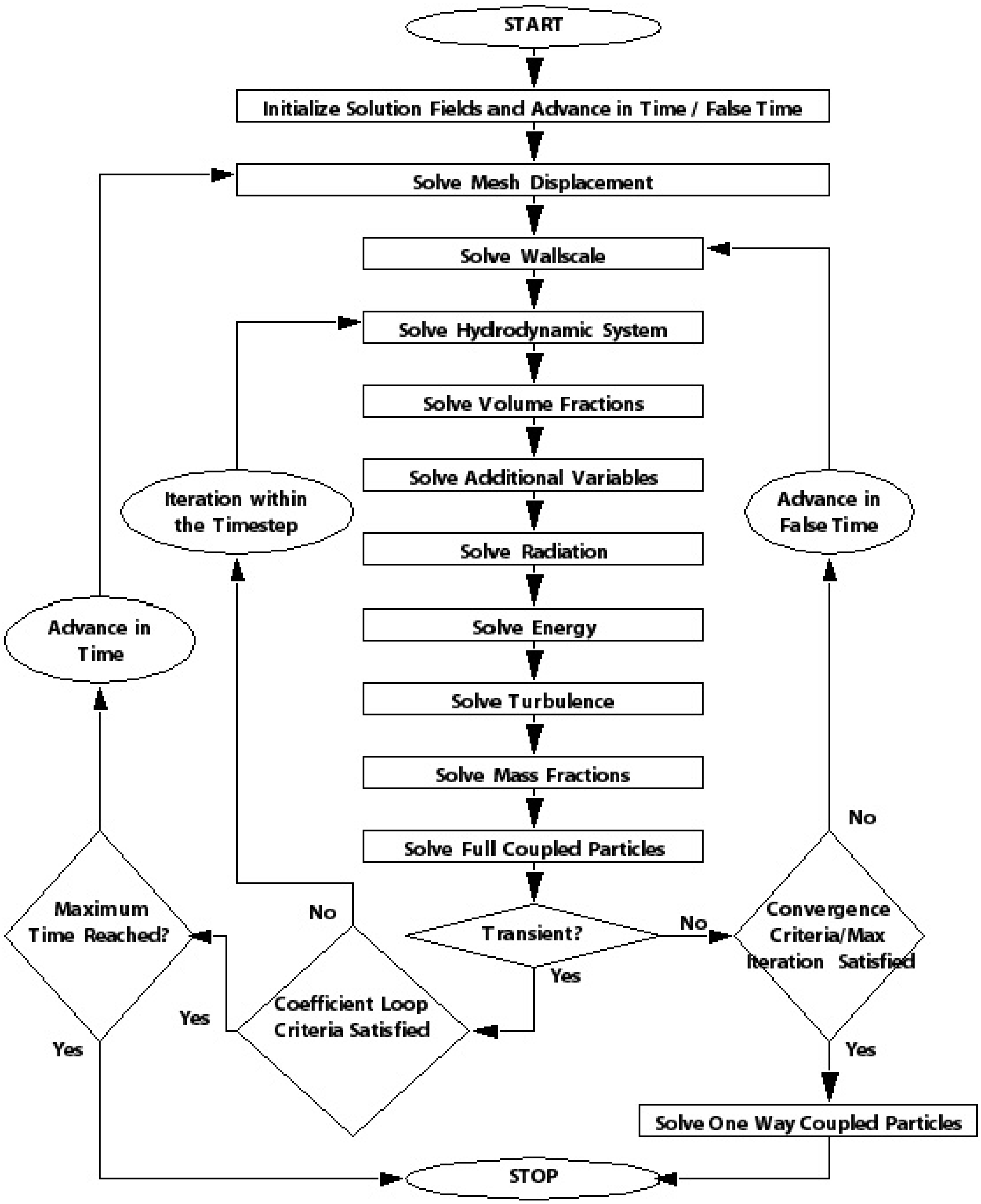
11.2.1. General Solution

The flow chart shown below illustrates the general field solution process used in the CFX-Solver.

The solution of each set of field equations shown in the flow chart consists of two numerically intensive operations. For each time step:

1. Coefficient Generation: The nonlinear equations are linearized and assembled into the solution matrix.
2. Equation Solution: The linear equations are solved using an Algebraic Multigrid method.

When solving fields in the CFX-Solver, the outer (or time step) iteration is controlled by the physical time scale or time step for steady and transient analyses, respectively. Only one inner (linearization) iteration is performed per outer iteration in steady state analyses, whereas multiple inner iterations are performed per time step in transient analyses.



11.2.2. Linear Equation Solution

ANSYS CFX uses a Multigrid (MG) accelerated Incomplete Lower Upper (ILU) factorization technique for solving the discrete system of linearized equations. It is an iterative solver whereby the exact solution of the equations is approached during the course of several iterations.

The linearized system of discrete equations described above can be written in the general matrix form:

$$[A]\ [\varphi] = [b] \tag{11–49}$$

where $[A]$ is the coefficient matrix, $[\varphi]$ the solution vector and $[b]$ the right hand side.

The above equation can be solved iteratively by starting with an approximate solution, φ^n, that is to be improved by a correction, φ', to yield a better solution, φ^{n+1}, that is,

$$\varphi^{n+1} = \varphi^n + \varphi' \tag{11–50}$$

where φ' is a solution of:

$$\dot{A}\ \varphi' = r^n \tag{11–51}$$

with r^n, the residual, obtained from:

$$r^n = b - A\ \varphi^n \tag{11–52}$$

Repeated application of this algorithm will yield a solution of the desired accuracy.

By themselves, iterative solvers such as ILU tend to rapidly decrease in performance as the number of computational mesh elements increases. Performance also tends to rapidly decrease if there are large element aspect ratios present.

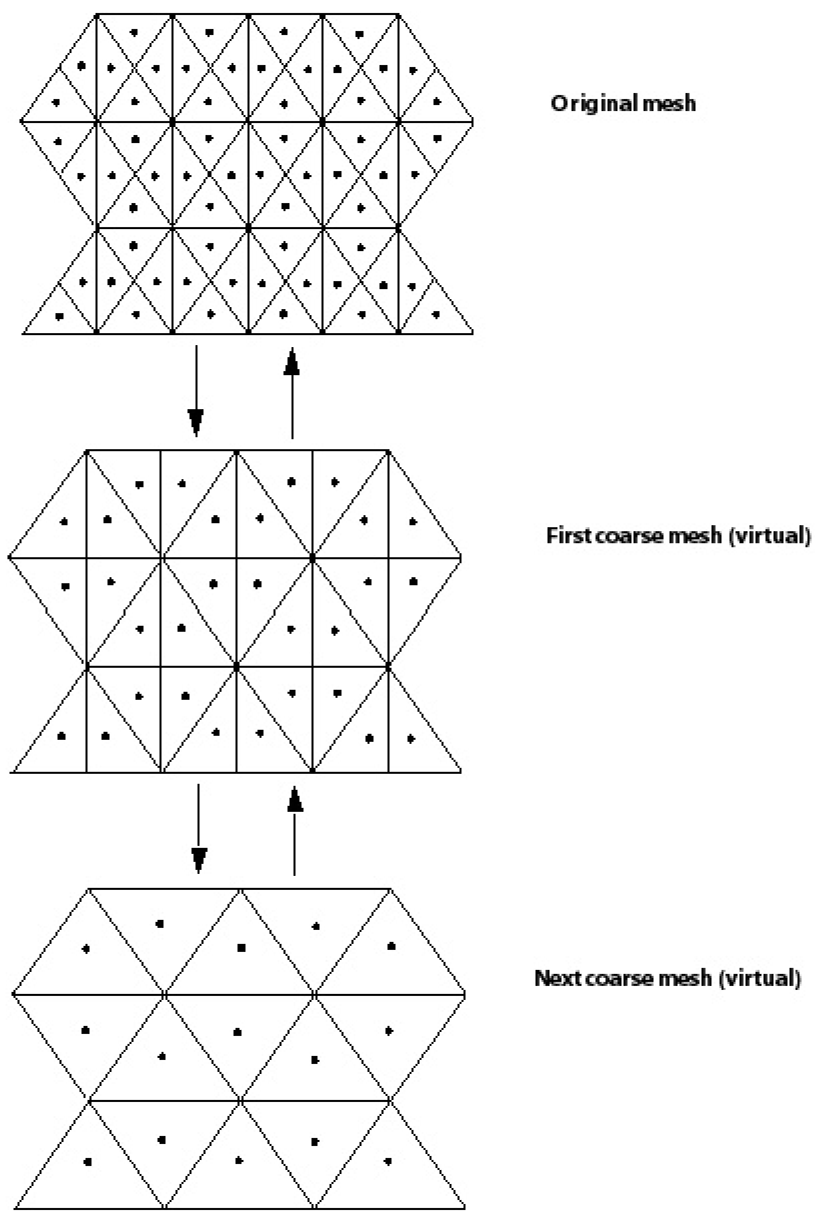
11.2.2.1. Algebraic Multigrid

The convergence behavior of many matrix inversion techniques can be greatly enhanced by the use of a technique called 'multigrid'. The multigrid process involves carrying out early iterations on a fine mesh and later iterations on progressively coarser virtual ones. The results are then transferred back from the coarsest mesh to the original fine mesh.

From a numerical standpoint, the multigrid approach offers a significant advantage. For a given mesh size, iterative solvers are efficient only at reducing errors that have a wavelength of the order of the mesh spacing. So, while shorter wavelength errors disappear quite quickly, errors with longer wavelengths, of the order of the domain size, can take an extremely long time to disappear. The Multigrid Method bypasses this problem by using a series of coarse meshes such that longer wavelength errors appear as shorter wavelength errors relative to the mesh spacing. To prevent the need to mesh the geometry using a series of different mesh spacings, ANSYS CFX uses Algebraic Multigrid.

Algebraic Multigrid [25] forms a system of discrete equations for a coarse mesh by summing the fine mesh equations. This results in virtual coarsening of the mesh spacing during the course of the iterations, and then re-refining the mesh to obtain an accurate solution. This technique significantly improves the convergence rates. Algebraic Multigrid is less expensive than other multigrid methods because the discretization of the nonlinear equations is performed only once for the finest mesh.

ANSYS CFX uses a particular implementation of Algebraic Multigrid called Additive Correction. This approach is ideally suited to the CFX-Solver implementation because, it takes advantage of the fact that the discrete equations are representative of the balance of conserved quantities over a control volume. The coarse mesh equations can be created by merging the original control volumes to create larger ones as shown below. The diagram shows the merged coarse control volume meshes to be regular, but in general their shape becomes very irregular. The coarse mesh equations thus impose conservation requirements over a larger volume and in so doing reduce the error components at longer wavelengths.



11.2.3. Residual Normalization Procedure

As described above, the raw residual, [*r*], is calculated as the imbalance in the linearized system of discrete equations. The raw residuals are then normalized for the purpose of solution monitoring and to obtain a convergence criteria. An overview of the normalization procedure is given below.

For each solution variable, φ, the normalized residual is given in general by:

$$\left[\tilde{r}_\varphi\right] = \frac{\left[r_\varphi\right]}{a_p\,\Delta\varphi} \tag{11–53}$$

where r_φ is the raw residual control volume imbalance, a_p is representative of the control volume coefficient, and $\Delta\varphi$ is a representative range of the variable in the domain. The exact calculation of a_p and $\Delta\varphi$ is not simple and is not presented here. However, some important notes are:

1. The normalized residuals are independent of the initial guess.
2. a_p is the central coefficient of the discretized control volume equation and therefore includes relevant advection, diffusion, source linearization, and other terms.
3. For steady state simulations, the time step is used only to underrelax the equations and is therefore excluded from the normalization procedure. This ensures that the normalized residuals are independent of the time step. The transient term is included in a_p for transient simulations.
4. For multiphase, if equations are coupled through an interphase transfer process (such as interphase drag or heat transfer), the residuals are normalized by the bulk a_p.

11.3. Discretization Errors

There are often differences between the exact analytical solution of the modeled differential equations (see *Equation 11–1* (p. 376)), and the fully converged solution of their discrete representations (see *Equation 11–4* (p. 377)). These differences are referred to as discretization errors.

Like the principal variables being solved for, errors in these values are both generated by localized sources and propagated (that is, amplified, advected, diffused) throughout the solution domain. Localized sources of error result from the high-order terms that are *excluded* from the discrete approximations of terms in the modeled equations. Conversely, error propagation results from the form of the terms that are *included* in the discrete approximations. Both error sources and propagation are affected by the solution and mesh distributions, as discussed in *Controlling Error Sources* (p. 394) and *Controlling Error Propagation* (p. 395).

11.3.1. Controlling Error Sources

Reducing the source of solution error (that is, the magnitude of terms excluded in the discrete approximations) is critical if accurate numerical solutions are desired. The two most effective strategies for accomplishing this are to increase the order-accuracy of discrete approximations (for example, using the high resolution rather than the upwind difference advection scheme) and/or to reduce the mesh spacing in regions of rapid solution variation. The former strategy is discussed above (see *Advection Term* (p. 383)), and implications of the latter are now considered.

Finely spaced isotropic mesh distributions are ideal, but they are often not tractable. A more useful strategy for reducing sources of solution error is to generate anisotropic meshes with fine spacing in directions of most rapid solution variation and relatively coarse spacing in other directions. This is exemplified by typical boundary layer meshes that are compressed in the direction of most rapid solution variation (that is, normal to the wall).

It is important to realize, however, that sources of solution error are also affected by poor geometrical mesh quality (see Measures of Mesh Quality in the *CFX-Solver Modeling Guide*). In particular, the error source contributions due to the discretization of transient/storage, diffusion, source and Rhie-Chow redistribution terms increase with mesh anisotropy. This is why, for example, high orthogonality and low expansion factors are recommended in boundary layer meshes where diffusive transport dominates.

11.3.2. Controlling Error Propagation

Controlling the transport and, more importantly, the amplification of error, is also critical, given that sources of solution error can often only be reduced and not eliminated. Little can be done to reduce error transport because error is subject to the same physical processes (such as, advection or diffusion) as the conserved quantities. Fortunately, the amplification of error is more easily controlled.

Errors are amplified by strong unphysical influences in the discrete form of the modeled equations. These unphysical influences will lead to convergence difficulties, and in extreme cases, complete divergence. Similar to error sources, error amplification is controlled through the choice of discretization and/or the mesh distribution.

As highlighted above, (unphysical) negative influences may be introduced into the discretization through the finite-element shape functions, and these influences may grow as geometrical mesh quality (see Measures of Mesh Quality in the *CFX-Solver Modeling Guide*) deteriorates. Tri-linear shape functions, for example, are more susceptible to negative influences than linear-linear shape functions. Nevertheless, tri-linear shape functions are used as much as possible due to their improved accuracy, and linear-linear shape functions are used whenever solution robustness is critical.

Geometrical mesh quality is important regardless of the shape functions used. An excellent example of this occurs when mesh elements are folded. If the mesh is sufficiently folded, control volumes become negative and the balance between the transient/storage and flow terms is rendered physically invalid. The amount of a conserved quantity within a control volume will actually decrease given a net flow into the control volume.

Index

T

U

V

W

Z