**Tampere University of Technology Modal Aerosol Model
for Computational Fluid Dynamics (CFD-TUTMAM) 2.0 Manual**
Miska Olin

September 15, 2017

CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

Latin symbols

$\mathcal{A}_{j,i}$	species i activity in a particle of mode j
C_{c}	slip correction coefficient
C_i	concentration of species i in the vapor phase
D	particle diffusion coefficient
d_1	the lowest particle diameter in a power law mode
d_2	the highest particle diameter in a power law mode
D_{coag}	cut diameter for coagulational transfer
D_i	gas species i diffusion coefficient
$\bar{D}_{j,k}$	k^{th} moment averaged diffusion coefficient of distribution j
d_{p}	particle diameter
D_{t}	turbulent diffusion coefficient
$D_{\phi,j,k,\mathrm{eff}}$	effective diffusion coefficient of UDS $\phi_{j,k}$
f_{HC}	fraction of condensing hydrocarbons
GR	condensational growth rate
i	index for particle species
iF	index for gas species in fluid
J	nucleation rate
k	order of the moment
K	coagulation kernel
k_{B}	the Boltzmann constant
K_i	Kelvin factor for species i
$k_{j,\mathrm{h2o}}$	condensation multiplier for water equilibrium computation for mode j
k_{nucl}	nucleation coefficient
Kn	Knudsen number
Kn_{c}	Knudsen number for coagulation
Kn_i	Knudsen number for species i
L_i	species i latent heat of condensation
m_{c}	nucleation cluster mass
$m_{\mathrm{c},i}$	nucleation cluster mass: species i part
$M_{j,1,i}$	species i part of 1st moment in mode j
$\mathcal{M}_i$	species i molar mass
$M_{j,k}$	k^{th} moment in mode j
m_{p}	particle mass
$m_{\mathrm{p},i}$	mass of species i in a particle
$n_{\mathrm{conc},i}$	nucleation exponent for species i

n_{GH}	Gauss-Hermite quadrature level
n_j	mode j density function
N_j	particle number concentration of a distribution j
n_{species}	number of particle species
$n_{\mathrm{vappres},i}$	nucleation exponent for saturation vapor pressure of species i
p	pressure
p_i°	species i saturation vapor pressure
$p_{i,\infty}$	species i partial pressure in fluid
$p_{j,i,\mathrm{p}}$	equilibrium vapor pressure of species i on a particle of mode j
R	universal gas constant
RH	relative humidity
t	time
T	temperature
$T_{\mathrm{p},j}$	mode j particle temperature
$\mathbf{u}$	flow velocity vector
$\bar{v}_i$	average thermic velocity of gas molecule i
$\bar{v}_{\mathrm{p}}$	average thermic velocity of a particle
w_i	weights for Gauss-Hermite quadrature
$x_{j,\mathrm{h2o,p}}^{\mathrm{eq}}$	water mole fraction in a particle of mode j in water equilibrium
$x_{j,i,\mathrm{p}}$	species i mole fraction in a particle of mode j
$Y_{\mathrm{HC,condensed}}$	fraction of already condensed hydrocarbons
Y_{iF}	mass fraction of gas species iF in fluid

Greek symbols

α	slope parameter of a power law distribution
β_c	Fuchs-Sutugin correction factor for coagulation
$\beta_{\mathrm{M},i}$	Fuchs-Sutugin correction factor for species i mass transport
β_{T}	Fuchs-Sutugin correction factor for heat transport
γ_{cond}	condensational transfer factor
$\gamma_{j,i}$	species i activity coefficient in mode j
$\gamma_{j,ph}$	phase ph activity coefficient in mode j
η	fluid dynamic viscosity
κ_{f}	thermal conductivity of the fluid
$\kappa_{j,\mathrm{h2o}}$	correction factor for water equilibrium computation for mode j
λ_{gas}	gas mean free path
λ_i	vapor i mean free path
ρ_{f}	fluid density
$\rho_{\mathrm{p},j}$	particle density in mode j
ρ_{ph}	phase density

σ_{ph}	surface tension of phase ph
$\phi_{j,k}$	value for UDS of moment k of mode j

Abbreviations

CFF	Custom Field Function
CMD	count median diameter
GSD	geometric standard deviation
UDM	User-Defined Memory
UDS	User-Defined Scalar
URF	Under-Relaxation Factor

1. CAPABILITIES AND LIMITATIONS

CFD-TUTMAM is an aerosol dynamics model used with CFD-modelling. Particle transport, diffusion, nucleation, condensation, coagulation, and vapor wall condensation can be simulated with the model. Table 1.1 shows the main limitations of the model, and Tab. 1.2 shows the modelling platform where CFD-TUTMAM is tested to be functioning.

Table 1.1: *Limitations of CFD-TUTMAM.*

Aerosol distribution	PL (power law), LN (log-normal), PL+LN
Number of distributions	$1 - 16$
Particle number concentration N_{tot}	high
Particle mass concentration M_1	$M_1 \ll \rho_{\text{f}}$
Particle morphology	spherical
Immiscible phases in a particle	$1 - 48$
Miscible species in a particle	$1 - 48$
Particle diameter d_{p}	$0.1\text{nm} - 1\mu\text{m}$
Temperature T	no limitations
Pressure p	no limitations

Table 1.2: *Tested modelling platform for CFD-TUTMAM.*

CFD-software	ANSYS FLUENT
Fluent versions	15.0.7, 16.1.0, 17.1.0
Fluent environments	serial, parallel
Geometry	3d, 2d, axisymmetric
Numerical precision	single, double
Operating systems	Windows 7 64bit, Unix 64bit
C-Compiler	MS Visual Studio 2010, Unix C-compiler

2. THEORY

2.1 Aerosol distributions

Aerosol distribution can modelled as one or more modes. All the modes are represented as log-normal (LN) distributions by default, but the first mode can be represented as power law (PL) distribution also. If PL distribution exists and there are at leat one LN distribution, particle tranfer from the PL distribution to the fist LN distribution can be modelled. Figure 2.1 represents a sample of a LN distribution. LN distribution has the following density
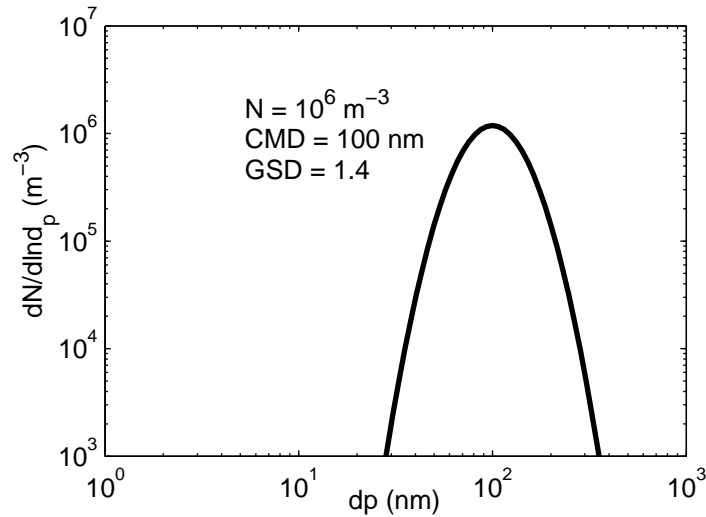

Figure 2.1: *A log-normal aerosol distribution.*

function

$$n_j = \left.\frac{\mathrm{d}N}{\mathrm{d}\ln d_\mathrm{p}}\right|_{\mathrm{LN}} = \frac{N_j}{\sqrt{2\pi}\,\ln GSD_j}\exp\left[-\frac{\ln^2\left(\frac{d_\mathrm{p}}{CMD_j}\right)}{\ln^2 GSD_j}\right], \tag{2.1}$$

where N_j is the total particle number concentration of mode j, $\mathrm{d}N$ is the particle number concentration in a logarithmic size range $\mathrm{d}\ln \mathrm{d}_\mathrm{p}$, CMD_j is the count median diameter, GSD_j is the geometric standard deviation, and d_p is the particle diameter (Hinds, 1999).

Figure 2.2 represents samples of PL distributions and of a combined PL+LN distribution. PL distribution has the following density function
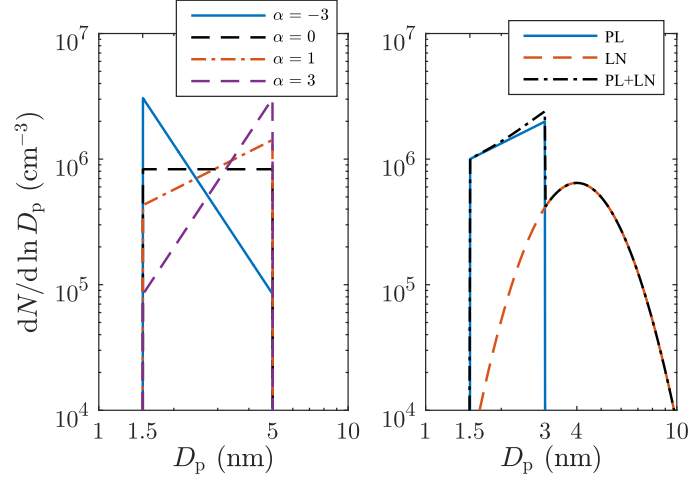
Figure 2.2: *Left pane represents examples of the power law distribution with different values of α. Right plane represents the combination of power law and log-normal distributions. (Olin et al., 2016)*

$$n_j = \left.\frac{\mathrm{d}N}{\mathrm{d}\ln d_\mathrm{p}}\right|_\mathrm{PL} = \begin{cases} \frac{N_j \alpha}{d_2^\alpha - d_1^\alpha} d_\mathrm{p}^\alpha, & d_1 \le d_\mathrm{p} \le d_2,\ \alpha \ne 0 \\[2mm] \frac{N_j}{\ln(d_2/d_1)}, & d_1 \le d_\mathrm{p} \le d_2,\ \alpha = 0 \\[2mm] 0, & \text{otherwise} \end{cases} \tag{2.2}$$

where α is the slope parameter, d_1 is the smallest diameter, and d_2 is the largest diameter of the the PL distribution (Olin et al., 2016).

2.2 Moments of a distribution

2.2.1 Single component case

A log-normal distribution function (Eq. 2.1) requires 3 parameters (N_j, CMD, GSD) to represent the particle concentration $\mathrm{d}N$ for a size d_p. Therefore, modelling of a log-normal distribution requires 3 variables and 3 equations to be modelled. A power law distribution requires 4 parameters (N_j, α, d_1, d_2) of which d_1 has a constant value. Therefore, 3 variables are required for the PL distribution also. These variables are the kth moments of a distribution j that are defined as[1]

$$M_{j,k} = \int_{-\infty}^{\infty} m_\mathrm{p}^k n_j\, \mathrm{d}\ln d_\mathrm{p}, \tag{2.3}$$

---

[1]In literature, it can be defined also with d_p^k instead of m_p^k.

where m_{p} is particle mass. The 3 moments [and their units] used in CFD-TUTMAM are

$$
\begin{array}{ll}
M_{j,0} & [1/\mathrm{m}^3] \\
M_{j,\frac{2}{3}} & [\mathrm{kg}^{\frac{2}{3}}/\mathrm{m}^3] \\
M_{j,1} & [\mathrm{kg}/\mathrm{m}^3]
\end{array}
\tag{2.4}
$$

which correspond to number concentration, surface area concentration (times $6^{-\frac{2}{3}}\pi^{-\frac{1}{3}}\rho_{\mathrm{p}}^{\frac{2}{3}}$), and mass concentration, respectively.

When 3 moments are known, the 3 parameters of a log-normal distribution can be calculated by (Whitby and McMurry, 1997)

$$
\begin{aligned}
N_j &= M_{j,0} \\
GSD_j &= \exp\left(\sqrt{\ln\frac{M_{j,1}^{\frac{2}{3}}M_{j,0}^{\frac{1}{3}}}{M_{j,\frac{2}{3}}}}\right) \\
CMD_j &= \left(\frac{6M_{j,1}}{\pi\rho_{\mathrm{p},j}M_{j,0}}\right)^{\frac{1}{3}}e^{-1.5\ln^2 GSD_j},
\end{aligned}
\tag{2.5}
$$

where $\rho_{\mathrm{p},j}$ is particle density. Reconstruction of the parameters of PL distribution from 3 moments is done either using the Levenberg-Marquardt iteration algorithm or using an interpolation table as described in Olin et al. (2016) . In the PL+LN model, totally 6 moments are modelled.

2.2.2 Multicomponent case

However, particles can be multicomponent, and thus the composition must be solved. CFD-TUTMAM assumes that the composition of particles in a specific location is equal in every particle, regardless of the particle size. In a multicomponent case, $M_{j,1}$ is divided to different species moments

$$
M_{j,1} = \sum_{i=0}^{n_{\mathrm{species}}} M_{j,1,i},
\tag{2.6}
$$

where i is the index for particle species. Therefore, the total amount of moments to be modelled is $(2 + n_{\mathrm{species}}) \cdot n_{\mathrm{modes}}$.

2.3 User-Defined Scalars

The moments are modelled in Fluent by using Fluent's User-Defined Scalars (UDSs), here denoted with a letter ϕ. They [and their units] are defined as

$$
\begin{aligned}
\phi_{j,0} &= M_{j,0}/\rho_{\mathrm{f}} \quad [1/\mathrm{kg}] \\
\phi_{j,\frac{2}{3}} &= M_{j,\frac{2}{3}}/\rho_{\mathrm{f}} \quad [\mathrm{kg}^{-\frac{1}{3}}] \\
\phi_{j,1,i} &= M_{j,1,i}/\rho_{\mathrm{f}} \quad [1]
\end{aligned} \tag{2.7}
$$

where ρ_{f} is fluid density.

2.4 General transport equation for UDSs

Aerosol dynamics is modelled as the following transport equation

$$
\frac{\partial(\rho_{\mathrm{f}}\phi_{j,k})}{\partial t} = \frac{\partial M_{j,k}}{\partial t} = convection_{j,k} + diffusion_{j,k} + nucleation_{j,k} + condensation_{j,k}
$$
$$
+ coagulation_{j,k}^{\mathrm{intra}} + coagulation_{j,k}^{\mathrm{inter}} + transfer_{j,k}^{\mathrm{PL+LN}}, \quad (2.8)
$$

which is solved by Fluent for all UDSs. Aerosol dynamics is connected to fluid by nucleation[2] and condensation (and evaporation), and therefore, the following equation is used for all condensing gas species iF

$$
\frac{\partial(\rho_{\mathrm{f}}Y_{iF})}{\partial t} = (\text{Equations from Fluent}) - nucleation_{0,1,i} - \sum_{j=0}^{n_{\mathrm{modes}}} condensation_{j,1,i}, \quad (2.9)
$$

where Y_{iF} is the mass fraction of gas species iF.

2.4.1 Convection

Convection for particles is modelled as they were travelling with the fluid, and the velocities of particles and fluid are always equal. Convection term in Eq. 2.8 is

$$
convection_{j,k} = -\nabla \cdot (\rho_{\mathrm{f}}\phi_{j,k}\mathbf{u}), \tag{2.10}
$$

where $\mathbf{u}$ is the flow velocity vector.

2.4.2 Diffusion

Diffusion of the UDSs is modelled with the following equation

$$
diffusion_{j,k} = \nabla \cdot (\rho_{\mathrm{f}}D_{\phi,j,k,\mathrm{eff}}\nabla\phi_{j,k}), \tag{2.11}
$$

---

[2]Nucleation is modelled only for the first mode ($j = 0$).

where $D_{\phi,j,k,\text{eff}}$ is the effective diffusion coefficient for UDS $\phi_{j,k}$, which is defined as

$$D_{\phi,j,k,\text{eff}} = \bar{D}_{j,k} + D_{\text{t}}, \tag{2.12}$$

where $\bar{D}_{j,k}$ is the kth moment averaged diffusion coefficient of a particle distribution j, and D_{t} is the turbulent diffusion coefficient.

The kth moment averaged diffusion coefficient is defined as

$$\begin{aligned} \bar{D}_{j,k} &= \frac{\int_{-\infty}^{\infty} m_{\text{p}}^k \, D \, n_j \, \mathrm{d}\ln d_{\text{p}}}{\int_{-\infty}^{\infty} m_{\text{p}}^k \, n_j \, \mathrm{d}\ln d_{\text{p}}} \\ &= \frac{1}{M_{j,k}} \int_{-\infty}^{\infty} m_{\text{p}}^k \, D \, n_j \, \mathrm{d}\ln d_{\text{p}}, \end{aligned} \tag{2.13}$$

where D is the diffusion coefficient of the particle:

$$D = \frac{k_{\text{B}} T C_{\text{c}}}{3\pi \eta d_{\text{p}}}, \tag{2.14}$$

where k_{B} is the Boltzmann constant, T is temperature, C_{c} is slip correction coefficient, and η is dynamic viscosity of the fluid. The slip correction coefficient is (Allen and Raabe, 1985)

$$C_{\text{c}} = 1 + \frac{\lambda_{\text{gas}}}{d_{\text{p}}} \left[2.34 + 1.05 \exp\left(-0.39 \frac{d_{\text{p}}}{\lambda_{\text{gas}}} \right) \right], \tag{2.15}$$

where λ_{gas} is the gas mean free path: (Willeke, 1976)

$$\lambda_{\text{gas}} = 66.5\,\text{nm} \cdot \left(\frac{T}{293.15\,\text{K}} \right) \left(\frac{101325\,\text{Pa}}{p} \right) \left(\frac{1 + 110.4\,\text{K}/293.15\,\text{K}}{1 + 110.4\,\text{K}/T} \right). \tag{2.16}$$

The integrand in Eq. 2.13 is complex and the integral cannot be solved analytically. CFD-TUTMAM provides two approaches to solve it: (1) computing the integral using Gauss-Hermite quadrature and (2) using parametrisation for C_{c} to achieve an analytically solvable integral. The Gauss-Hermite quadrature integration technique is basically a numerical integration and it is described in Sec. 2.5. Parametrisation for slip correction coefficient is (Olin, 2013)

$$C_{\text{c}}(d_{\text{p}}, T) = C_1 \, d_{\text{p}}^2 + C_2 \, d_{\text{p}} + C_3 + C_4 \, d_{\text{p}}^{-1}, \tag{2.17}$$

where the coefficients C_i for this temperature and particle size range are

$$\begin{aligned} C_1 &= -1.15 \cdot 10^{11} \\ C_2 &= -25.61\,T + 3.447 \cdot 10^5 \\ C_3 &= -1.451 \cdot 10^{-4}\,T + 0.7301 \\ C_4 &= 1.044 \cdot 10^{-9}\,T - 8.884 \cdot 10^{-8}. \end{aligned} \tag{2.18}$$

Using the parametrisation, the integrand in Eq. 2.13 becomes polynomial, and thus the integral can be solved analytically. The error produced by parametrisation at the temperature range of 250K – 1800K and the particle size range of the model at atmosperic pressure is below 6%, but it provides a shorter computation time compared to Gauss-Hermite integration. C_c approaches unity with increasing particle size. Even a more short computation time is obtained by using a constant value for C_c.

2.4.3 Nucleation

The nucleation terms in Eq. 2.8 and Eq. 2.9 are

$$
\begin{aligned}
nucleation_{0,0} &= J \\
nucleation_{0,\frac{2}{3}} &= J m_c^{\frac{2}{3}} \\
nucleation_{0,1,i} &= J m_{c,i},
\end{aligned}
\tag{2.19}
$$

where J is nucleation rate, m_c is the mass of the cluster formed by nucleation, and $m_{c,i}$ is the part i of the cluster. Nucleation rate and the cluester composition can be modelled using parametrized classical nucleation theory for sulfuric acid-water nucleation (Vehkamäki et al., 2002, 2003) or using a more general nucleation scheme (Olin et al., 2015)

$$
J = k_{\text{nucl}} \prod_{i=0}^{n_{\text{species}}} \frac{C_i^{n_{\text{conc},i}}}{[p_i^\circ(T)]^{n_{\text{vappres},i}}},
\tag{2.20}
$$

where k_{nucl} is nucleation coefficient, C_i is concentration of species i in the vapor phase, $p_i^\circ(T)$ is saturation vapor pressure of species i at temperature T, $n_{\text{conc},i}$ is nucleation exponent for species i, and $n_{\text{vappres},i}$ is exponent for saturation vapor pressure of species i. In general nucleation scheme, the cluster composition is defined as the constant number of molecules of each species in the cluster. Also some broadening of the cluster distribution in the general nucleation scheme is used by replacing $m_c^{\frac{2}{3}}$ in Eq. (2.19) with $m_c^{\frac{2}{3}} \exp\left(-\ln^2 GSD_c\right)$ where GSD_c is the geometric standard deviation of the cluster distribution formed.

2.4.4 Condensation and evaporation

The condensation terms in Eq. 2.8 and Eq. 2.9 are

$$
\begin{aligned}
condensation_{j,0} &= 0 \\
condensation_{j,\frac{2}{3}} &= \frac{2}{3} \int_{-\infty}^{\infty} m_p^{-\frac{1}{3}} \frac{\partial m_p}{\partial t} n_j \, \mathrm{d}\ln d_p \\
condensation_{j,1,i} &= \int_{-\infty}^{\infty} \frac{\partial m_{p,i}}{\partial t} n_j \, \mathrm{d}\ln d_p,
\end{aligned}
\tag{2.21}
$$

where $\frac{\partial m_{\mathrm{p}}}{\partial t}$ is the mass growth rate of the particle, and $\frac{\partial m_{\mathrm{p},i}}{\partial t}$ is the mass growth rate of particle species i (Barrett and Clement, 1988; Lehtinen and Kulmala, 2003)

$$\frac{\partial m_{\mathrm{p},i}}{\partial t} = \frac{2\pi\mathcal{M}_i}{RT}\left(D_p + D_i\right)\left(d_{\mathrm{p}} + d_i\right)\beta_{\mathrm{M},i}\left(p_{i,\infty} - p_{j,i,\mathrm{p}}\right)\left(1 + \frac{p_{i,\infty} - p_{j,i,\mathrm{p}}}{2p}\right), \qquad (2.22)$$

where $\mathcal{M}_i$, D_i, and d_i are molar mass, diffusion coefficient, and molecular diameter of species i, respectively. R is the universal gas constant, $\beta_{\mathrm{M},i}$ is the Fuchs-Sutugin correction factor for species i mass transport, $p_{i,\infty}$ is the partial pressure of species in fluid, $p_{j,i,\mathrm{p}}$ is the equilibrium vapor pressure of species i on the particle of mode j, and p is the absolute pressure of the fluid.

The equilibrium vapor pressure is calculated with

$$p_{j,i,\mathrm{p}} = \mathcal{A}_{j,i}\,K_i\,p_i^\circ = x_{j,i,\mathrm{p}}\,\gamma_{j,i}\,\gamma_{j,ph}\,K_i\,p_i^\circ, \qquad (2.23)$$

where $\mathcal{A}_{j,i} = x_{j,i,\mathrm{p}}\,\gamma_{j,i}\,\gamma_{j,ph}$ is the activity, $x_{j,i,\mathrm{p}}$ is the mole fraction, $\gamma_{j,i}$ is the activity coefficient, K_i is the Kelvin factor, and p_i° is the saturation vapor pressure of species i on the particle. $\gamma_{j,ph}$ is the phase activity coefficient of the phase ph where species i belongs to. The phase activity coefficient in single phase simulation is always unity; but modelling with two phases, the model desribed in Olin et al. (2015) can be used. The Kelvin factor is defined as

$$K_i = \exp\left(\frac{4\sigma_{ph}\mathcal{M}_i}{RT\rho_{ph}d_{\mathrm{p}}}\right), \qquad (2.24)$$

where σ_{ph} and ρ_{ph} are the surface tension and the liquid density of the phase ph, respectively. In the case of condensation, Eq. 2.22 becomes positive, and in the case of evaporation, it becomes negative.

If the latent heat release of condensation is modelled, particle temperature increases (condensation) or decreases (evaporation). Therefore, Eq. 2.23 needs to be calculated with particle temperature (Barrett and Clement, 1988)

$$T_{\mathrm{p},j} = T + \frac{\sum_{i=0}^{n_{\mathrm{species}}}\left(L_i\,condensation_{j,1,i}\right)}{2\pi\,CMD_j\,\kappa_{\mathrm{f}}\,N_j\,\beta_{\mathrm{T}}}, \qquad (2.25)$$

where L_i is the latent heat of condensation of species i, κ_{f} is the thermal conductivity of the fluid, and β_{T} is the Fuchs-Sutugin correction factor for heat transport. In this case, condensation term is not explicitly defined, thus $condensation_{j,1,i}$ in Eq. 2.25 is obtained from the previous iteration step.

The Fuchs-Sutugin correction factors are (Fuchs and Sutugin, 1970)

$$\beta_{\mathrm{M},i} = \frac{1 + Kn_i}{1 + 1.71 Kn_i + 1.333 Kn_i^2} \tag{2.26}$$

$$\beta_{\mathrm{T}} = \frac{1 + Kn}{1 + 1.71 Kn + 1.333 Kn^2}, \tag{2.27}$$

where Kn_i and Kn are the Knudsen numbers

$$Kn_i = \frac{2\lambda_i}{d_i + d_{\mathrm{p}}} \tag{2.28}$$

$$Kn = \frac{2\lambda_{\mathrm{gas}}}{d_{\mathrm{p}}}, \tag{2.29}$$

where λ_i is the vapor mean free path

$$\lambda_i = \frac{3(D_i + D_{\mathrm{p}})}{\sqrt{\bar{v}_i^2 + \bar{v}_{\mathrm{p}}^2}}. \tag{2.30}$$

$\bar{v}_i$ and $\bar{v}_{\mathrm{p}}$ are the average thermic velocities of the gas molecule i and of the particle

$$\bar{v}_i = \sqrt{\frac{8RT}{\pi M_i}} \tag{2.31}$$

$$\bar{v}_{\mathrm{p}} = \sqrt{\frac{8k_{\mathrm{B}}T}{\pi m_{\mathrm{p}}}}. \tag{2.32}$$

In Eq. 2.26, the accommodation coefficient is assumed unity. If D_i is modelled in a way that the relative humidity has an effect on its value (e.g., sulfuric acid forms hydrates of different size depending on the humidity), the average thermic velocity for the gas molecule will be modelled using the formula

$$\bar{v}_i = \frac{D_i(RH)}{D_i(RH = 0)} \sqrt{\frac{8RT}{\pi M_i}}. \tag{2.33}$$

Equation 2.21 is computed by the Gauss-Hermite (LN) or Gauss-Olin (PL) quadrature integration method or by numerical integration described in Sec. 2.5.

Water equilibrium computation

In the case of condensation of water vapor, condensation can be very rapid at least in exhaust simulations containing sulfuric acid also. Therefore, the mesh should be very dense to achieve a converged solution. Conversely, CFD-TUTMAM provides another formulation for water vapor condensation, in which an equilibrium state is assumed. (Olin et al., 2015)

The mass growth rate of a particle by water vapor $\frac{\partial m_{\mathrm{p,h2o}}}{\partial t}$ is connected to another species,

such as sulfuric acid, by

$$\frac{\partial m_{\mathrm{p,h2o}}}{\partial t} = \kappa_{j,\mathrm{h2o}} \, k_{j,\mathrm{h2o}} \, \frac{\partial m_{\mathrm{p,h2so4}}}{\partial t}, \tag{2.34}$$

where $\kappa_{j,\mathrm{h2o}}$ is a correction factor and $k_{j,\mathrm{h2o}}$ is a condensation multiplier defined as

$$k_{j,\mathrm{h2o}}(T, RH, \gamma_{j,ph}, d_{\mathrm{p}}) = \frac{x^{\mathrm{eq}}_{j,\mathrm{h2o,p}}}{1 - x^{\mathrm{eq}}_{j,\mathrm{h2o,p}}}, \tag{2.35}$$

where $x^{\mathrm{eq}}_{j,\mathrm{h2o,p}}(T, RH, \gamma_{j,ph}, d_{\mathrm{p}})$ is the mole fraction of water in the equilibrium state. The value of d_{p} is the diameter of the average mass of a distribution.

The correction factor $\kappa_{j,\mathrm{h2o}}$ is obtained from UDM. After every iteration step, the value of $\kappa_{j,\mathrm{h2o}}$ is updated so that all the particles[3] in every computational cell is in water equilibrium.

Diesel exhaust hydrocarbon model

The wide range of different hydrocarbons in diesel exhaust can be modelled using only one species. The diesel exhaust hydrocarbon model corrects the partial pressure of hydrocarbons $p_{\mathrm{HC},\infty}$ to

$$(f_{\mathrm{HC}} - Y_{\mathrm{HC,condensed}})p_{\mathrm{HC},\infty}, \tag{2.36}$$

where $f_{\mathrm{HC}}(T, p_{\mathrm{HC},\infty})$ is the fraction of condensing hydrocarbons and $Y_{\mathrm{HC,condensed}}$ is the fraction of already condensed hydrocarbons. (Olin et al., 2015)

2.4.5 Coagulation

The coagulation model includes intramodal coagulation inside a particle mode and intermodal coagulation between different modes.

Intramodal coagulation term in Eq. 2.8 is modelled according to Whitby and McMurry (1997):

$$\begin{aligned}
coagulation^{\mathrm{intra}}_{j,0} &= -\frac{1}{2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(d_{\mathrm{p},1}, d_{\mathrm{p},2}) \, n_j(d_{\mathrm{p},1}) \, n_j(d_{\mathrm{p},2}) \, \mathrm{d}\ln d_{\mathrm{p},1} \, \mathrm{d}\ln d_{\mathrm{p},2} \\
coagulation^{\mathrm{intra}}_{j,\frac{2}{3}} &= -\frac{1}{2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[2m^{\frac{2}{3}}_{\mathrm{p},1} - (m_{\mathrm{p},1} + m_{\mathrm{p},2})^{\frac{2}{3}} \right] \\
&\quad \times K(d_{\mathrm{p},1}, d_{\mathrm{p},2}) \, n_j(d_{\mathrm{p},1}) \, n_j(d_{\mathrm{p},2}) \, \mathrm{d}\ln d_{\mathrm{p},1} \, \mathrm{d}\ln d_{\mathrm{p},2} \\
coagulation^{\mathrm{intra}}_{j,1} &= 0 \, ,
\end{aligned} \tag{2.37}$$

where K is the coagulation kernel

$$K(d_{\mathrm{p},1}, d_{\mathrm{p},2}) = 2\pi \beta_{\mathrm{c}}(d_{\mathrm{p},1} + d_{\mathrm{p},2})(D_{\mathrm{p},1} + \mathrm{p}, D_2), \tag{2.38}$$

---

[3]Actually, only the particle with the diameter of average mass.

where β_c is the transition regime correction factor for coagulation. The correction factor can be calculated either with the Fuchs-Sutugin formula (Eq. 2.26) or with Dahneke's formula (Dahneke, 1983)

$$\beta_c = \frac{1 + Kn_c}{1 + 2Kn_c + 2Kn_c^2}. \tag{2.39}$$

In both formulae, Kn is replaced by

$$Kn_c(d_{p,1}, d_{p,2}) = \frac{Kn_c'\left(D_{p,1} + p, D_2\right)}{(d_{p,1} + d_{p,2})\sqrt{\bar{v}_{p,1}^2 + \bar{v}_{p,2}^2}}, \tag{2.40}$$

where $Kn_c' = 6$ using the Fuchs-Sutugin formula and $Kn_c' = 4$ using Dahneke's formula. A robustness model can be used with intramodal coagulation that uses analytical solution of coagulation within a single computational cell to prevent unphysical solutions with very high particle concentrations. The robustness model multiplies coagulation terms with

$$\frac{\dfrac{1}{1 + coagulation_{j,0}^{\text{intra}}\Delta t/N_j}}{\dfrac{1}{1 + coagulation_{j,\frac{2}{3}}^{\text{intra}}\Delta t/M_{\frac{2}{3}}}}, \tag{2.41}$$

where Δt is the time that particles comsume in the computational cell.

Intermodal coagulation term in Eq. 2.8 is

$$coagulation_{j,0}^{\text{inter}} = - \sum_{j'=j+1}^{n_{\text{modes}}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(d_{\text{p},1}, d_{\text{p},2}) \, n_{j'}(d_{\text{p},1}) \, n_j(d_{\text{p},2}) \, \mathrm{d}\ln d_{\text{p},1} \, \mathrm{d}\ln d_{\text{p},2}$$

$$coagulation_{j,\frac{2}{3}}^{\text{inter}} = \sum_{j'=0}^{j-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[(m_{\text{p},1} + m_{\text{p},2})^{\frac{2}{3}} - m_{\text{p},2}^{\frac{2}{3}} \right]$$

$$\times K(d_{\text{p},1}, d_{\text{p},2}) \, n_{j'}(d_{\text{p},1}) \, n_j(d_{\text{p},2}) \, \mathrm{d}\ln d_{\text{p},1} \, \mathrm{d}\ln d_{\text{p},2}$$

$$- \sum_{j'=j+1}^{n_{\text{modes}}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} m_{\text{p},2}^{\frac{2}{3}}$$

$$\times K(d_{\text{p},1}, d_{\text{p},2}) \, n_{j'}(d_{\text{p},1}) \, n_j(d_{\text{p},2}) \, \mathrm{d}\ln d_{\text{p},1} \, \mathrm{d}\ln d_{\text{p},2} \qquad (2.42)$$

$$coagulation_{j,1,i}^{\text{inter}} = \sum_{j'=0}^{j-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} m_{\text{p},1,i}$$

$$\times K(d_{\text{p},1}, d_{\text{p},2}) \, n_{j'}(d_{\text{p},1}) \, n_j(d_{\text{p},2}) \, \mathrm{d}\ln d_{\text{p},1} \, \mathrm{d}\ln d_{\text{p},2}$$

$$- \sum_{j'=j+1}^{n_{\text{modes}}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} m_{\text{p},2,i}$$

$$\times K(d_{\text{p},1}, d_{\text{p},2}) \, n_{j'}(d_{\text{p},1}) \, n_j(d_{\text{p},2}) \, \mathrm{d}\ln d_{\text{p},1} \, \mathrm{d}\ln d_{\text{p},2}.$$

The robustness model is not used with intermodal coagulation.

The double integrals in Eq. 2.37 and Eq. 2.42 are computed by the Gauss-Hermite (LN) or Gauss-Olin (PL) quadrature integration method or by numerical integration described in Sec. 2.5.

2.4.6 Vapor wall condensation

In addition to Fluent's own wall boundary conditions for vapors (specified mass fraction or zero flux), two other methods are provided in CFD-TUTMAM. The first method is based on the assumption that the vapor condenses on the wall if its saturation is exceeded. If the saturation rate of a vapor is over unity, the mass fraction is set to zero on the wall (condensation). Conversely, at the saturation rate below unity, zero flux is set to the wall (no condensation). The second method is based on the activity of the liquid formed to the walls. It provides a smooth behavior of the wall condensation, but it can be used only if the water equilibrium computation is used.

2.4.7 Transfer from PL mode to LN mode

Particles in the PL mode can be transferred to a LN mode using the coagulational and condensation transfer (Olin et al., 2016). In the case of the coagulational transfer, resultant particles larger than d_2 formed by self-coagulation in the PL mode ($j = 0$) are transferred to the first LN mode ($j = 1$). In the case of the condensational transfer, some of the largest particles in the PL mode are transferred to the first LN mode. Mathematically it is expressed as

$$transfer_{j,k}^{\mathrm{PL+LN}} = (-1)^{j+1} \left[\frac{1}{2} \int_{-\infty}^{\infty} \int_{\ln D_{\mathrm{coag}}}^{\infty} [m_{\mathrm{p},1} + m_{\mathrm{p},2}]^k \ K(d_{\mathrm{p},1}, d_{\mathrm{p},2}) \cdot \right.$$
$$n_0(d_{\mathrm{p},1}) \, n_0(d_{\mathrm{p},2}) \, \mathrm{d} \ln d_{\mathrm{p},1} \, \mathrm{d} \ln d_{\mathrm{p},2}$$
$$\left. + \gamma_{\mathrm{cond}} \frac{3}{\alpha + 2} \left(1 - \left(\frac{d_1}{d_2} \right)^{\alpha+2} \right) GR \, m_2^k \, d_2^{-1} \, n_0(d_2) \right], \qquad (2.43)$$

where

$$\ln D_{\mathrm{coag}} = \left(d_2^3 - d_{\mathrm{p},2}^3 \right)^{1/3} \qquad (2.44)$$

and

$$GR = \frac{\mathrm{d}d_{\mathrm{p}}}{\mathrm{d}t} = \frac{2}{\pi \rho_{\mathrm{p}} d_2^2} \frac{\mathrm{d}m}{\mathrm{d}t} \qquad (2.45)$$

and γ_{cond} the condensational transfer factor (ranges from 0 to 1). When the robustness model is used in the intramodal coagulation, it is also used in the coagulational transfer.

2.5 Numerical integration methods

In the Gauss-Hermite quadrature integration method, an integral of type

$$\int_{-\infty}^{\infty} f(x) \, e^{-x^2} \mathrm{d}x \qquad (2.46)$$

is computed numerically by converting it to

$$\sum_{i=1}^{n_{\mathrm{GH}}} w_i \, f(x_i), \qquad (2.47)$$

where n_{GH} is the level of the quadrature, w_i are the total weights, and x_i are the abscissas obtained from Hermite polynomials. The Hermite polynomials for quadrature levels 5, 7, 9, and 11 are stored in CFD-TUTMAM. The higher the level is, the more accurate solution it provides, but the computation time becomes also longer. In CFD-TUTMAM,

$$x = \frac{\ln \left(\frac{d_{\mathrm{p}}}{CMD} \right)}{\sqrt{2} \ln GSD} \qquad (2.48)$$

is used for x.

In the Gauss-Olin quadrature integration method, an integral of type

$$\int_{\ln d_1}^{\ln d_2} f(d_\mathrm{p}) \, d_\mathrm{p}^\alpha \, \mathrm{d} \ln d_\mathrm{p} \tag{2.49}$$

is computed numerically by converting it to

$$\sum_{i=1}^{4} w_i \left(\alpha \ln \frac{d_2}{d_1} \right) f(d_i). \tag{2.50}$$

Numerical integration should be used with the PL distribution when $\alpha < 0.5$ (condensation) or when $d > 3$ (coagulation), because the error caused by the Gauss-Olin quadrature increases with those values. An integral of type

$$\int_{\ln d_1}^{\ln d_2} f(d_\mathrm{p}) \, n_j \, \mathrm{d} \ln d_\mathrm{p} \tag{2.51}$$

is computed numerically as follows

$$\sum_{i=1}^{n_\mathrm{bins}} f(d_\mathrm{p,i}) \, n_j(d_\mathrm{p,i}) \, \mathrm{d} \ln d_\mathrm{p} \tag{2.52}$$

where $d_\mathrm{p,i}$ is the geometric mean diameter of the size bin; diameters spaced logarithmically to n_bins bins.

3.   USING CFD-TUTMAM

3.1   Installing CFD-TUTMAM

Extract `CFD-TUTMAM.zip` to the folder where your case files are.

3.2   Configuring CFD-TUTMAM

Some of the configuring is done by making changes to the source code files, but some is done through the CFD-TUTMAM GUI menu in Fluent. Source code files to be altered are:

```
tutmam_settings.h
tutmam_material_func.c
```

Settings in the CFD-TUTMAM GUI can be changed inside the case and the settings data are saved to `.cas` file. You can make different versions for those two files in the same folder by adding suffixes to their names, e.g.:

```
tutmam_settingsCase1.h
tutmam_material_funcMaterialSet1.c
```

The suffixes are then told to the CFD-TUTMAM GUI, when the code is to be compiled.

The following guide shows an example of particle formation settings. The case in the example is related to sulfuric acid-water particles represented with the PL+LN distribution.

3.2.1   Settings in source code files

1. Open `tutmam_settings.h`. The settings you can change there are `#define` macros.

2. Change the number of modes to:

   ```
   #define nTutmamModes 2
   ```

3. Change the number of particle phases to:

   ```
   #define nTutmamPhases 1
   ```

4. Figure out your table of species and the amount of the condensing species. Table 3.1 shows an example of the table of species. The IDs in Fluent should correspond to

Table 3.1: *Example table of species, with 3 gas species and 2 particle species.*

species name	ID in Fluent	ID in CFD-TUTMAM
index in source code	`iFluentSpecies`	`iSpecies`
index in this manual	iF	i
h2so4	0	0
h2o	1	1
n2	2	

the order of species in Fluent's material names list. The order of the IDs in CFD-TUTMAM is your choice, but the IDs must always start from 0. Now you have a vector of condensing species, which in this example case is [h2so4, h2o].

5. Change the number of condensing species to:

   ```
   #define nTutmamSpecies 2
   ```

6. Change the number of Fluent species to:

   ```
   #define nFluentSpecies 3
   ```

7. Change the following vectors:

   ```
   #define molarMasses 0.098079,0.018015
   #define fluentSpeciesIds 0,1
   #define phaseIds 0,0
   #define initialYSpeciesP {0.3,0.7},{0.3,0.7}
   ```

   where `molarMasses` represents the molar masses of the condensing species, `fluentSpeciesIds` the IDs in Fluent, `phaseIds` the IDs of the particle phases and `initialYSpeciesP` the mass fractions in initialization, in the order that corresponds to your vector of condensing species (one -bracket set per mode).

8. Set

   ```
   #define powerLawDistribution 0
   ```

   if power law distribution is to be modelled. Otherwise it is -1.

9. Next, there are settings for the User-Defined Memory (UDM) and User-Defined Scalar (UDS) locations for CFD-TUTMAM. There should be no need to change them.

10. The minimum values for the concentrations and the boundaries for *CMD* can be changed, but the default values should be ok.

11. You can change initial distribution parameters and particle density values to correspond to your case:

```
#define initialNumberConc 1.0e10,1.0e10
#define initialCMD 20.0e-9,20.0e-9
#define initialGSD 1.3,1.3
#define initialRhoParticle 1200.0,1200.0
```

The initial parameters are used, when the domain is initialized in Fluent. Separate the values for different modes with a comma.

12. You can change the maximum value of particle and fluid temperature difference (K) in:

```
#define maxParticleTemperatureDifference 100.0
```

but the default value should be fine.

13. You can change the settings related to water equilibrium calculation:

```
#define minCMDForKappaCalculation 1.5e-9
#define convergenceCriteriumForKappaCalculation 0.01
#define maxIterationsInKappaCalculation 20
#define uRFKappaCalculation 0.5
#define uRFKappa 0.8
```

`minCMDForKappaCalculation` is the minimum CMD diameter for which water equilibrium calculation is performed. `convergenceCriteriumForKappaCalculation` denotes how far water equilibrium calculation is iterated (0.1 means that when the equilibrium mole fraction of water has an error of less than 10%, it is fine). `maxIterationsInKappaCalculation` is the maximum amount of iterations in water equilibrium calculation. `uRFKappaCalculation` is the under-relaxation factor in iterative water equilibrium calculation. `uRFKappa` is the under-relaxation factor for κ in means of Fluent's iterations.

14. You can change the settings related to power law distribution:

```
#define whenAlphaIsZero 0.001
#define minAlpha -5.0
#define maxAlpha 5.0
#define powerLawD1 1.15e-9
#define maxD2 100.0e-9
#define uRFPowerLawParameters 0.99
```

whenAlphaIsZero 0.001 means that when $\alpha < 0.001$, functions where $\alpha = 0$ are used. minAlpha and maxAlpha are the boundaries for calculation of α. powerLawD1 is the parameter d_1 for power law distribution that is assumed constant. maxD2 is the maximum value of d_2. uRFPowerLawParameters is the under-relaxation factor for the parameters α and d_2 in the means of Fluent's iterations.

15. Change the directions of coagulation to:

```
#define coagulationFromModeToMode {1,1},{1,1}
```

where 0 denotes no coagulation and 1 denotes coagulation.

16. Change the laws for wall condensation to:

```
#define wallCondensationLaws 2,2
```

where 0 denotes no condensation, 1 denotes condensation only if saturation exceeded, and 2 denotes full condensation.

17. Save the file.

18. Open tutmam_material_func.c file.

19. Make the changes in the following material property functions corresponding to your table of species:

```
particle_bulk_density(int iSpecies)
particle_phase_density
    (real temp, int ph, const real *massFractionsInPhase)
diffusion_coefficient_gas
    (real temp, real pressure, int iFluentSpecies, real relativeHumidity)
latent_heat(int iSpecies)
phase_activity(real volumeFractionInParticle)
activation_coefficient
    (real temp, int iSpecies, const real *moleFractionsInPhase)
saturation_vapor_pressure(real temp, int iSpecies)
molecule_diameter(int iSpecies)
surface_tension(real temp, int ph, const real *moleFractionsInPhase)
eq_mole_fraction_water(real temp, real rh, real dp)
diesel_exhaust_hc_fraction(real temp, real partialPressureHC)
```

20. Save the file.

3.3   Launching CFD-TUTMAM

1. Start Fluent normally or by selecting `tutmam.gui` as a journal file. The journal file loads the CFD-TUTMAM GUI menus.

2. (OPTIONAL) Type the command

   ```
   f rj tutmam.gui
   ```

   in Fluent, if you didn't select the journal file when launching Fluent.

3. Compile the source code by typing the suffixes, e.g,:

   ```
   Case1
   MaterialSet1
   ```

   to the text boxes[1] and by pressing CFD-TUTMAM > Compilation > 2. Compile, or by compiling it through Fluent's menus, or by compiling it in your own way. Compiling the library without the GUI button requires that all `*.c` and `*.h` files are selected, the name of the library is `tutmamudf`, and suffixes are not used. However, compilation is not needed when the source code is not changed.

4. Open a previously defined CFD-TUTMAM case or build a new one.

3.4   Building a new case with CFD-TUTMAM

3.4.1   The first solution by modelling diffusion and nucleation only

1. Open a previously defined non-CFD-TUTMAM case or build a new one. The case has to have at least the Energy and Species Transport models included to work with CFD-TUTMAM.

2. You can solve the case without CFD-TUTMAM at first, or you can load CFD-TUTMAM now and solve the case with it.

3. Press CFD-TUTMAM > Compilation > 3. Load or load `tutmamudf` manually through Fluent's menus. This loads the CFD-TUTMAM libraries.

4. Select all the available hook functions

   ```
   tutmam_initialize_automatically::tutmamudf
   tutmam_save_udm_variables_automatically::tutmamudf
   ```

---

[1]If you leave the boxes empty, files without suffixes are used.

to Fluent's Define > User-Defined > Function Hooks > Initialization / Execute at End.

5. Press CFD-TUTMAM > General settings > Show UDS and UDM mapping. The IDs for UDSs and UDMs will be printed, and you need to check how many UDS and UDM slots you need to open. You need to open at least so many slots that the highest IDs are included. Remember that the first slot in Fluent is zero. For example, if the highest ID is 11, you need to open 12 slots. The slots can be opened in Fluent's Define > User-Defined > Scalars / Memory > Number of User-Defined Scalars/Memory[2] Locations.

6. Select `diff_uds::tutmamudf` for the UDS Diffusivity of your mixture in Fluent's Materials editor.

7. You can select `diff_gas::tutmamudf` for Mass Diffusivity of the mixture in Fluent's Materials editor to use `diffusion_coefficient_gas` function defined in `CFD-TUTMAM_material_func.c` file as the diffusion coefficients for the gas species. You can also define it in another way in Fluent, but CFD-TUTMAM still uses the `diffusion_coefficient_gas` function internally.

8. Remember to change Fluent's default constant value for the mixture viscosity to a temperature dependent function, such as the `sutherland` function, because the value of the mixture viscosity is commonly used in the CFD-TUTMAM code.

9. Set boundary conditions for aerosol distributions in Fluent's Boundary Conditions > zone > UDS.

10. Check the fluid zone ID in Fluent's Cell Zone Conditions and set that number to CFD-TUTMAM > General settings > Fluid zone ID.

11. The following steps (12–16) can be skipped in most cases, if you want to use the default values for them. However, you should check the values at some point.

12. Set the level of Gauss-Hermite quadrature integration in CFD-TUTMAM > General settings > Gauss-Hermite quad. level. The integration method is described in Sec. 2.5. The higher the level is, the more accurate solution it provides, but computation time becomes also longer.

13. In CFD-TUTMAM > Aerosol distribution, you can set the minimum and maximum values for the geometric standard deviation (GSD) of the distribution. The minimum value must be > 1. If you model the PL distribution, select the method of how the parameters of the PL distribution are calculated from the moments: Levenberg-Marquardt

---

[2]Node Memory Locations are not needed

iteration algorithm (default) or interpolation table. Interpolation table might be slower and more inaccurate, but more robust. Levenberg-Marquardt algorithm should provide the parameters accurately and fast if the initial guess is good. Initial guess is updated after every iteration. The best practise is to calculate first using the interpolation table and switch to Levenberg-Marquardt iteration algorithm after that.

14. In CFD-TUTMAM > Aerosol process control, you can select the processes to be included in the model. By default, only diffusion is selected, and a good practice may be starting the calculation only with diffusion (and nucleation) and selecting the other processes after some iterations. When the other processes are selected, you can deselect the `Compute` checkbox if you want to stop the computation of the process, but not to deselect the process itself. This can be useful, e.g., when you decide that condensation is already computed enough and you want to compute coagulation alone.

15. In CFD-TUTMAM > Diffusion, you can select the calculation of the slip correction coefficient by maintaining the selection in `Calculate slip corr. coeff.` checkbox. By deselecting it, a constant value will be used as a slip correction coefficient, which you can define in `Constant slip corr. coeff.` entry. If the slip correction coefficient is to be calculated, you can use the Diffusion integral parametrization that results in shorter computation time but also a more inaccurate solution. Remember that the parametrization is developed for the temperature range of 250K – 1800K and for atmospheric pressure.

16. If your model includes turbulence modelling, you can model particle turbulent diffusion by selecting CFD-TUTMAM > Diffusion > Particle turbulent diffusion. In that case, you need to define Particle turbulent Schmidt number or use the default value of 0.7.

17. Open CFD-TUTMAM > Nucleation panel.

18. Set under-relaxation factor (URF) for the nucleation process.

19. Set correction factor for nucleation rate.

20. Select nucleation law: (1) homogeneous binary h2so4-h2o nucleation using parametrizations of Vehkamäki et al. (2002, 2003) developed from classical nucleation theory, (2) the general nucleation scheme as in Eq. 2.20. In the general nucleation scheme, the following setting need to be defined: Nucleation species flags are 1 if the species participates in nucleation and 0 if not. Nucleation exponents are $n_{\mathrm{conc},i}$ in Eq. 2.20 and sat. vap. pres. exponents are $n_{\mathrm{vappres},i}$ in Eq. 2.20. Set also the number of molecules in a cluster formed by nucleation. The number of molecules defines the mass of the newly formed clusters, and the corresponding diameters should be little higher than the `minCMD` and `powerLawD1` values in `tutmam_settings.h` file. Producing clusters

having a fixed mass only using the general form of nucleation rate, can cause numerical difficulties in the calculation of distribution parameters from the moments of a distribution. To avoid that, some broadening of the cluster distribution can be taken into account. This is done by giving a small GSD value in the nucleation panel. The default value of 1.01 should be enough small to avoid these numerical difficulties.

21. Select the source terms

    ```
    udf nucleation::tutmamudf
    ```

    for all nucleating species and for UDSs of the first mode in Fluent's Cell Zone Conditions > fluid > Source Terms. The checkbox `Source Terms` must be selected to access the selection of the source terms.

22. At this point, you should run some iterations before selecting other aerosol processes. If there are already some iterations run in the case and you want to maintain the solution, you can start computing aerosol modelling by initializing only the UDS and UDM fields by executing

    ```
    tutmam_initialize_manually::tutmamudf
    ```

    in Fluent's Define > User-Defined > Execute on Demand. Initializing the whole solution can be done in Fluent's Solution Initialization > Initialize. This initializes both the fluid variables and the UDS and UDM variables.

23. Calculate the solution with your preferred settings for the Solution Methods / Controls. In some cases, a divergence error occurs at this point. In that case, try first to patch the variable, that the error message mentioned, to zero in the fluid zone. This can be done in Fluent's Solution Inialization > Patch. After that, do initialization again as explained in the previous step. Other errors may also occur here due to Fluent's bugs; in that case, you can try to save the case (and data), exit Fluent, and open Fluent and the case (and data) file again.

3.4.2 Solution with other aerosol processes

Condensation

1. Open CFD-TUTMAM > General settings panel.

2. If you want to model a PL distribution, set the level of numerical integration for condensation to 1 (only Gauss-Olin quadrature), 2 (Gauss-Olin quadrature when $\alpha < 0.5$, numerical integration otherwise), or 3 (only numerical integration).

3. If numerical integration is to be used, set the amount of bins to 200 (default) or to anything else.

4. Open CFD-TUTMAM > Condensation panel.

5. You can set Under-Relaxation Factors (URFs) individually for the specific particle species. The `URF for surface area moment condensation` entry is the URF for the $\frac{2}{3}^{\text{th}}$ moment commonly. The `URF vector for mass moment condensation` entry contains the vector of URFs for the different particle species in the order of your table of condensing species. The delimiters in the vector can be `,` `;` `_` or an empty space. By default, the URFs are 1, but in case of rapid condensation, lower values are needed to prevent divergence of the solution.

6. Selecting the `Latent heat effect included in condensation` checkbox, Eq. 2.25 will be used to compute particle temperature. Otherwise, the approximation where particle temperature is always equal to fluid temperature is used.

7. Select the species that are condensing by setting flags to 1 in `Condensing species flags`.

8. Select Kelvin effect if you want it.

9. Select condensing directions as follows: (1) condensation only, (-1) evaporation only, (0) both.

10. Select Phase activity model if you want it.

11. If PL+LN model is to be modelled, set intermodal condensation factor (γ_{cond}) to a value between 0 and 1.

12. If you want water equilibrium model, select the checkbox of that. Select also the Fluent ID for the species of which condensation rate is connected to water. Select also the boundaries for κ.

13. If you want diesel exhaust hydrocarbon fraction model, select the checkbox of that. Select also the Fluent ID for hydrocarbon species.

14. Select the source terms

   `udf condensation::tutmamudf`

   to the condensing gas species and to all the User Scalars[3] in Fluent's Cell Zone Conditions > fluid > Source Terms. The checkbox `Source Terms` must be selected to access the selection of the source terms.

   ---
   [3]Actually, it is not needed for the 0th moment UDSs, because it returns 0 for that.

15. Select a checkbox in CFD-TUTMAM > Aerosol process control > Condensation and maintain the `Compute` checkbox selected.

16. Calculate the solution.

Coagulation

1. Open CFD-TUTMAM > General settings panel.

2. If you want to model a PL distribution, set the level of numerical integration for coagulation to 1 (only Gauss-Olin quadrature), 2 (Gauss-Olin quadrature when $d < 3$, numerical integration otherwise), or 3 (only numerical integration).

3. If numerical integration is to be used, set the amount of bins to 20 (default) or to anything else.

4. Open CFD-TUTMAM > Coagulation panel.

5. Select transition regime correction law as follows: (1) Eq. 2.26, (2) Eq. 2.39.

6. If you want intramodal coagulation, select the checkbox of that. Coagulation computing is time consuming due to double integrals. You can select how often coagulation is computed in the `Iteration skip` box. If the value is 1, coagulation is computed at every iteration. If the value is 2, it is computed at every second iteration, etc. During the skipped iteration, the previously computed coagulation terms are used in the calculation. URF for intramodal coagulation process can be set also. In the case of rapid coagulation and a too coarse mesh, the URF must be low, but it can be up to unity in some cases. If unphysical solutions are encountered, select robustness model (Eq. 2.41 is used). Selecting the robustness model here, it will be selected also for the coagulational transfer from the PL distribution to the LN distribution.

7. If you want intermodal coagulation, select the checkbox of that. The settings as in intramodal coagulation cannot be used in intermodal coagulation.

8. Select the source terms

```
udf intra_coagulation::tutmamudf
udf inter_coagulation::tutmamudf
```

to all the User Scalars in Fluent's Cell Zone Conditions > fluid > Source Terms. The checkbox `Source Terms` must be selected to access the selection of the source terms.

9. Select the checkbox CFD-TUTMAM > Aerosol process control > Coagulation and maintain the `Compute` checkbox selected.

10. Calculate the solution.

Transfer from PL mode to LN mode

1. Select the source terms

   `udf transfer_pl2ln::tutmamudf`

   to all the User Scalars in Fluent's Cell Zone Conditions > fluid > Source Terms. The checkbox `Source Terms` must be selected to access the selection of the source terms. These are used only for modes 0 and 1 and only if both, PL and LN, modes are to be modelled.

2. Select the checkboxes CFD-TUTMAM > Aerosol process control > Self-coagulational transfer and Intermodal condensation. Choose also the under-relaxation factor to transfer process to URF box. The default value of 0.5 should be ok. If you encounter numerical difficulties after selecting the coagulational transfer, you can test the robustness model for coagulation. That can be selected in CFD-TUTMAM > Coagulation panel.

3. Calculate the solution.

Vapor wall condensation

1. Select the source terms

   `udf wall_condensation::tutmamudf`

   to all the gas species in Fluent's Boundary Conditions > wall > Species to walls you want to be condensing [4]. The source terms must be present there in every species shown regardless of the wall condensation laws of them.

2. Calculate the solution.

3.4.3 Launching an existing CFD-TUTMAM case

The only thing you need to do when you start a new Fluent process and open an existing CFD-TUTMAM case, is loading the journal `tutmam.gui`. The journal can be loaded by selecting it in Fluent's launcher or by typing the command

`f rj tutmam.gui`

in Fluent. The journal can also be loaded after loading the case file. Do not load the journal, if CFD-TUTMAM menu is already seen in Fluent's menu bar.

---

[4]Note that a wall-condensing species must be a particle-condensing species in the table of species also.

3.5   Errors

The possible errors encountered are usually either error messages or unphysical solutions. In the parallel solver, Fluent process can also crash instead of the error messages.

3.5.1   Error messages

Divergence detected in AMG solver: uds-x

If this error message appears immediately after selecting a new aerosol process to be calculated, the previous solution has propably not yet converged sufficiently. Try initializing the CFD-TUTMAM variables by

```
tutmam_initialize_manually::tutmamudf
```

in Fluent's Define > User-Defined > Execute on Demand, or initializing the whole solution in Fluent, and start solving with only the diffusion process selected. If the error remains, try saving the file, closing Fluent, and starting it again. You can also try to patch UDS x manually to zero after initializing in Fluent's Solution Initialization > Patch...

If the error appears during solving, the under-relaxation factors may be too high. The error is also encountered when, for example, there is division by zero or a logarithm or negative number in the code; therefore, it may be a programming bug also.

FLUENT received fatal signal (ACCESS VIOLATION)

This error appears when unsufficient amount of UDS or UDM slots are selected in Fluent's Define > User-Defined > Scalars / Memory > Number of User-Defined Scalars/Memory Locations. Check the required amounts in CFD-TUTMAM > General settings > Show UDS and UDM mapping. If the error remains, it may be a programming bug.

chip-exec: function "x::tutmamudf" not found.

`tutmamudf` library is not loaded. Press CFD-TUTMAM > Compilation > 3. Load or load `tutmamudf` manually through Fluent's menus.

3.5.2   Unphysical solutions

If the solution is unphysical, for example, if there are negative UDSs, the mesh is propably too coarse. The negative UDSs can be caused by too rapid coagulation or deposition compared to mesh density. If mesh refinement is not an option, try decreasing the under-relaxation factors or use robustness model for coagulation.

BIBLIOGRAPHY

Allen, M. and Raabe, O.: Slip correction measurements of spherical solid aerosol particles in an improved Millikan apparatus, Aerosol Sci. Tech., 4, 269–286, doi:10.1080/02786828508959055, 1985.

Barrett, J. and Clement, C.: Growth rates for liquid drops, J. Aerosol Sci., 19, 223–242, 1988.

Dahneke, B.: Simple kinetic theory of brownian diffusion in vapors and aerosols, 49, pp. 97–133, 1983.

Fuchs, N. and Sutugin, A.: Highly dispersed aerosols, Ann Arbor Science Publ., Ann Arbor, Michigan, USA, 1970.

Hinds, W. C.: Aerosol technology: properties, behavior, and measurement of airborne particles, John Wiley & Sons, Inc., Hoboken, USA, 2nd edn., 1999.

Lehtinen, K. and Kulmala, M.: A model for particle formation and growth in the atmosphere with molecular resolution in size, Atmos. Chem. Phys., 3, 251–257, doi:10.5194/acp-3-251-2003, 2003.

Olin, M.: Dieselpakokaasun hiukkaspäästöjen muodostumisprosessin simulointi, Master's thesis, Tampere University of Technology, 2013.

Olin, M., Rönkkö, T., and Dal Maso, M.: CFD modeling of a vehicle exhaust laboratory sampling system: sulfur-driven nucleation and growth in diluting diesel exhaust, Atmos. Chem. Phys., 15, 5305–5323, doi:10.5194/acp-15-5305-2015, 2015.

Olin, M., Anttila, T., and Dal Maso, M.: Using a combined power law and log-normal distribution model to simulate particle formation and growth in a mobile aerosol chamber, Atmos. Chem. Phys., 16, 7067–7090, doi:10.5194/acp-16-7067-2016, 2016.

Vehkamäki, H., Kulmala, M., Napari, I., Lehtinen, K., Timmreck, C., Noppel, M., and Laaksonen, A.: An improved parameterization for sulfuric acid-water nucleation rates for tropospheric and stratospheric conditions, J. Geophys. Res.-Atmos., 107, AAC 3–1–AAC 3–10, doi:10.1029/2002JD002184, 2002.

Vehkamäki, H., Kulmala, M., Lehtinen, K., and Noppel, M.: Modelling binary homogeneous nucleation of water-sulfuric acid vapours: Parameterisation for high temperature emissions, Environ. Sci. Technol., 37, 3392–3398, doi:10.1021/es0263442, 2003.

Whitby, E. and McMurry, P.: Modal aerosol dynamics modeling, Aerosol Sci. Tech., 27, 673–688, doi:10.1080/02786829708965504, 1997.

Willeke, K.: Temperature dependence of particle slip in a gaseous medium, J. Aerosol Sci.,
7, 381–387, 1976.