# Parameterised Processes in FLEXPART (“Physics”)

**Petra Seibert**

Institute of Meteorology, University of Natural Resources and Life Sciences (BOKU), Vienna, Austria

and

Department of Meteorology & Geophysics, University of Vienna, Vienna, Austria

FLEXPART Training Course 2019

# Overview



Preliminary automated on-line code documentation:
`https://homepage.univie.ac.at/petra.seibert/Doc_fp_release-10_23Apr2019/`
Note: this will disappear when it is replaced by the next version.
Eventually, it will be linked from or moved to `https://flexpart.eu`

## Turbulence regimes

1. Boundary layer (standard)
   - Unstable (convective) boundary layer
   - Stable boundary layer
2. Convective boundary layer with skewed turbulence (new in Fp10)
3. Free atmosphere (background turbulence)
4. Mesoscale meandering

## Variables controlling the turbulent motion

- Boundary-layer height $h$
- Statistical moments of $u, v, w$; in the standard case, just the standard deviations $\sigma_{u,v,w}$
- Lagrangian time scales $\tau_{L_{u,v,w}}$
- A suitably distributed set of random numbers

# Langevin equation

Determination of the (vertical) turbulent velocity component

$$d\left(\frac{w}{\sigma_w}\right) = \underbrace{-\frac{w}{\sigma_w}\frac{dt}{\tau_{L_w}}}_{\text{autocorrelation}} + \underbrace{\frac{\partial \sigma_w}{\partial z}dt}_{\text{drift correction}} + \underbrace{\frac{\sigma_w}{\rho}\frac{\partial \rho}{\partial z}dt}_{\text{density gradient corr.}} + \underbrace{\left(\frac{2}{\tau_{L_w}}\right)^{1/2} dW}_{\text{stochastic part}} \qquad (1)$$

It is recommended to always use the *slow mode* corresponding to *this* equation (set `CTL`$>0$), it is more accurate!

# Turbulence scheme for unstable conditions

`hanna.f`, `hanna1.f`, `hanna_short.f`, according to **??**

Subscripts $u$ and $v$ refer to the along-wind and the cross-wind components (transformed to grid coordinates in subroutine `windalign.f`)

## Unstable

$$\frac{\sigma_u}{u_*} = \frac{\sigma_v}{u_*} = \left(12 + \frac{h}{2|L|}\right)^{1/3} \tag{2}$$

$$\tau_{L_u} = \tau_{L_v} = 0.15\frac{h}{\sigma_u} \tag{3}$$

$$\sigma_w = \sqrt{1.2 w_*^2 \left(1 - 0.9\frac{z}{h}\right)\left(\frac{z}{h}\right)^{2/3} + \left(1.8 - 1.4\frac{z}{h}\right) u_*^2} \tag{4}$$

$$\tau_{L_w} = \begin{cases} \dfrac{0.1z}{\sigma_w\left[0.55 - 0.38\left(z - z_0\right)/L\right]} & z/h < 0.1 \text{ and } z - z_0 > -L \\ 0.59z/\sigma_w & z/h < 0.1 \text{ and } z - z_0 < -L \\ \frac{0.15h}{\sigma_w}\left[1 - \exp -\left(\frac{5z}{h}\right)\right] & z/h > 0.1 \end{cases} \tag{5}$$

# Turbulence scheme for neutral & stable conditions

## Neutral

$$\frac{\sigma_u}{u_*} = 2.0 \exp(-3\, fz/u_*) \tag{6}$$

$$\frac{\sigma_v}{u_*} = \frac{\sigma_w}{u_*} = 1.3\ \exp(-2\, fz/u_*) \tag{7}$$

$$\tau_{L_u} = \tau_{L_v} = \tau_{L_w} = \frac{0.5\, z/\sigma_w}{1 + 15\, fz/u_*} \tag{8}$$

## Stable

$$\frac{\sigma_u}{u_*} = 2.0 \left(1 - \frac{z}{h}\right) \tag{9}$$

$$\frac{\sigma_v}{u_*} = \frac{\sigma_w}{u_*} = 1.3 \left(1 - \frac{z}{h}\right) \tag{10}$$

$$\tau_{L_u} = 0.15 \frac{h}{\sigma_u} \left(\frac{z}{h}\right)^{0.5} \tag{11}$$

$$\tau_{L_v} = 0.07 \frac{h}{\sigma_v} \left(\frac{z}{h}\right)^{0.5} \tag{12}$$

$$\tau_{L_w} = 0.1 \frac{h}{\sigma_w} \left(\frac{z}{h}\right)^{0.5} \tag{13}$$

# ABL parameters required for the turbulence scheme

- Friction velocity $u_*$: from met input (turbulent stress vector)
- Sensible heat flux $Q_H$: from met input
- Obukhov length `obukhov.f90`

$$L = \frac{\theta u_*^2}{kg\theta_*} \qquad (14)$$

$$\theta_* = \rho c_p \frac{Q_h}{u_*} \qquad (15)$$

- Coriolis parameter $f = 2\Omega \sin \varphi$

- Convective vertical velocity scale `richardson.f90`

$$w_* = \sqrt[3]{\frac{-Q_h\, g\, h}{c_p(\theta_r)}} \qquad (16)$$

$$\theta_r = \theta_0 + -8.5 \frac{Q_H}{c_p w_*} \qquad (17)$$

- Boundary-layer height $h$ is determined as the height where Ri>Ri$_c$ $= 0.25$

$$\mathrm{Ri} = \frac{g}{\theta_r} \frac{[\theta(z) - (\theta_r)](z)}{\underbrace{(\Delta u)^2 + (\Delta v)^2 + 100\, u_*^2}_{\text{or at least } 0.1\,\mathrm{m^2\,s^{-2}}}} \qquad (18)$$

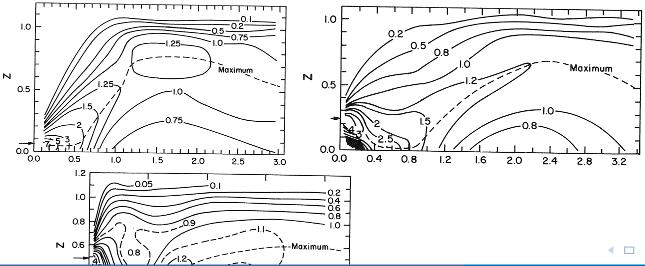
$$\Delta u = u(z) - u_{10} \quad \Delta v = v(z) - v_{10} \qquad (19)$$

where $u_{10}, v_{10}$ stands for the second model level

# Convective boundary layer

## Problem

- FLEXPART uses normally distributed turbulent velocities
- Vertical velocity fluctuations $w'$ in the CBL are skewed (few strong updrafts, large areas with slow subsidence)
- This causes plumes from near-ground sources to rise quickly (thus ground-level concentrations decrease more rapidly) and plumes from elevated sources to be pushed down (concentration maximum closer)



# Convective boundary layer

## CBL scheme in FLEXPART v10

- **?** developed an alternative Langevin equation model including both skewed turbulence and a vertical density gradient
- To activate, set `CBL=1` in the file `COMMAND`
- Requires short time steps, e.g. `CTL=10 and IFINE=10`
- Thus, CPU time will be approx. 2.5 $\times$ without CBL scheme
- As any emissions mix vertically within a few km in a strongly unstable CBL, this option makes sense only for local-scale calculations under highly convective conditions (but then it is essential for accurate results)

# Turbulence scheme in the free atmosphere

There is turbulence in the free atmosphere due to gravity wave breaking and wind shear (so-called CAT – clear-air turbulence). There is no detailed scheme for this, but their overall effects are parameterised by a constant background turbulence – horizontal in the free troposphere, vertical in the stratosphere.

## Above the ABL

Free troposphere:

$$D_h = 50\,\mathrm{m}^2\,\mathrm{s}^{-1} \tag{20}$$

Stratosphere according to LEGRAS2003 ($P > 2\,\mathrm{PVU}$):

$$D_z = 0.1\,\mathrm{m}^2\,\mathrm{s}^{-1} \tag{21}$$

$$\sigma_{v,w} = \sqrt{\frac{D_{h,z}}{\Delta t}} \tag{22}$$

$D_{h,z}$ ... horizontal / vertical diffusivity
$P = (\zeta_\theta + f)\frac{\partial\theta}{\partial p}$ ... potential vorticity `calcpar.f90`, `calcpv.f90`

# Mesoscale fluctuations

There are fluctuations of wind by mesoscale processes which are subgrid-scale but not turbulence. They are parameterised through a simple Langevin equation where the stochastic component is proportional to the turbulent velocity standard deviations:

## Mesoscale meandering

Once per `lsynctime` ($\Delta t$):

$$\sigma_{u,\text{meso}} = r\sigma_{u,\text{meso,old}} + r^2 W \sigma_u f_{\text{meso}} \tag{23}$$

$$\sigma_{v,\text{meso}} = r\sigma_{v,\text{meso,old}} + r^2 W \sigma_v f_{\text{meso}} \tag{24}$$

$$\sigma_{w,\text{meso}} = r\sigma_{w,\text{meso,old}} + r^2 W \sigma_w f_{\text{meso}} \tag{25}$$

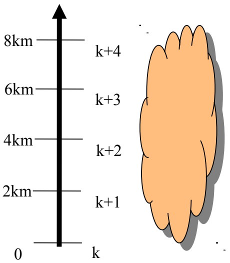
$$r = e^{-2\Delta t/\Delta T} \tag{26}$$

$f_{\text{meso}} = 0.16$ ... set in `par_mod.f90`
$W$ ...normally distributed random number
$\Delta T$ ... time interval of meteorological input

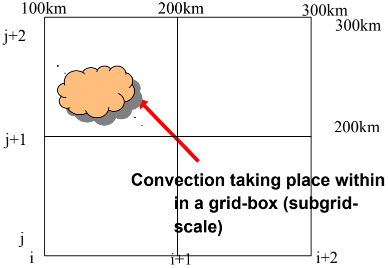
# (Moist) Convection (in Cb, Cu) – introduction

Figure from Delia Arnold

**convection is grid-scale in the vertical**




**but subgrid-scale in the horizontal**




## Problems with convection in dispersion models

- Needs to be parameterised (subgrid-scale process)
- Convection to be recreated from limited information in met. input
- Interaction with wet deposition scheme – does not know convective clouds

# Convection – solution in FLEXPART

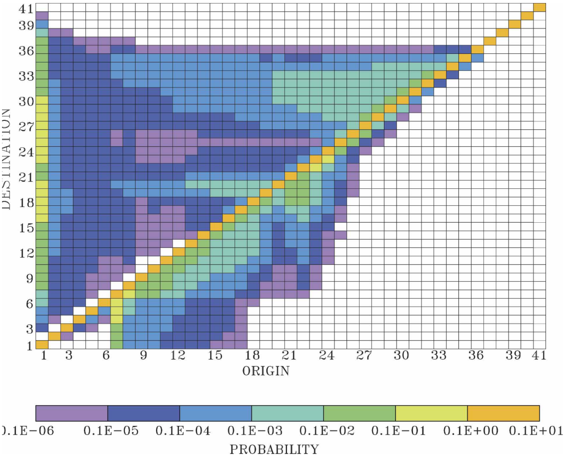



FIG. 1. Example of a mean convective redistribution matrix along 10° latitude for October 1983 calculated from the EZ99 scheme. The colors indicate the probability for a particle to be displaced from its origin to its destination level. White colors indicate probabilities below $10^{-6}$. The sum of each column is 1. Origin and destination are given

- Construct probability matrix for movement from level *k* to *l* from available large-scale fields using the convection scheme of Emanuel & Zivković-Rothmann
- Note that this is different from the scheme used in ECMWF
- Random displacement of particles based on matrix
- Significant increase of computation time
- Important in the moist tropics, useful in midlatitudes

Responsible source files:

`convect43c.f90, calcmatrix.f90, convmix.f90, redist.f90`

Details see Forster et al. (2007)

# Decay

Radioactive or chemical decay can be included:

$$m(t+\Delta t) = m(t)\exp\left(-\frac{\Delta t}{T_{1/2}}\ln 2\right) \qquad (27)$$

- Half life $T_{1/2}$ is specified in the SPECIES file
- Deposited pollutant mass decays at the same rate.

**Note: Radioactive decay can also be introduced in postprocessing.**

- For multiple nuclides with similar physical behaviour, this saves CPU time, memory, and disc space.
- Not accurate for nuclides with half-lives short compared to the output averaging time interval
- As FLEXPART is currently programmed, doses from deposited activity can only be correctly calculated with postprocessing, as the mean activity during each interval is needed, not the activity at the end of interval.

# What is dry deposition?

Mass of trace substances can be transferred from the atmosphere to the surface

- if in contact with the surface
- through chemical reactions (gas only)
- physical adsorption / absorption
- sticking (particles only)
- entering leaves through stomata (and subsequent retention)

It depends on the properties of both the substance and the surface.

Processes include

- aerodynamic resistance (turbulence)
  between $h_{ref}$ and surface $r_a$
- diffusion through the laminar sublayer (gases) $r_b$
- impaction (particles) $r_p$
- surface resistance (gases) $r_c$

# Dry deposition

Mass lost by deposition is calculated as

$$\Delta m(t) = m(t)\left[1 - \underbrace{\exp\left(-\frac{v_d \Delta t}{2h_{ref}}\right)}_{\text{deposition probability}}\right] \quad \text{if } z < 2h_{ref} \tag{28}$$

This mass is added to the dry deposition 2D field.

Reference height for dry deposition $h_{ref}$ is set in `par_mod.f90` (default: 15 m)

Except for giant particles and highly reactive gases, dry deposition is at least an order of magnitude smaller than wet deposition (if it rains, of course)

# Deposition velocity – gases & particles

- $v_d$ can be prescribed in SPECIES file
  deposition flux $= v_d \, c(h_{ref})$
- can be calculated from physical parameters, with different algorithms for gases and particles
  $v_d = (\sum_i r_i)^{-1}$ where $r_i$ are *resistances*

## Aerodynamic resistance (same for gases and particles)

calculated in function `raerod.f`
using the flux-profile relationship based on Monin-Obukhov similarity theory (Stull, 1988)

$$r_a = \frac{1}{ku_*}\left[\ln(h_{ref}/z_0) - \Psi_h(h_{ref}/L) + \Psi_h(z_0/L)\right] \tag{29}$$

$\Psi$ ... integrated profile functions $L$ ... Obukhov length
$u_*$ ... friction velocity $k$ ... Kármán constant
$z_0$ ... surface roughness length

# Deposition velocity – gases & particles /2

## Resistance of laminar sublayer

Source: ERISMAN1994

$$r_b = \frac{2}{ku_*}\left(\frac{\mathrm{Sc}}{\mathrm{Pr}}\right)^{\frac{2}{3}} \tag{30}$$

Pr ... Prandtl number = 0.72

Sc= $\nu(T)/D(T)$ Schmidt number $\nu$ ... kinematic viscosity of air

$D(T) = \frac{D_{H_2O}(T)}{D_r}$ ... molecular diffusivity of substance in air.

The diffusion relative to water vapour $D_r$ is read from the SPECIES file.

Note that $D_r$ is defined in a counter-intuitive way–might get changed in the future.

Note that $D_r$ *must* be specified as $> 0$ even if $v_d$ is explicity given! $r_b$ is calculated in function `getrb.f`.

# Deposition velocity – gases /3

## Surface resistance (gas)

Calculated in function `getrc.f`. Source WESELY1989

$$\frac{1}{r_c} = \frac{1}{r_s + r_m} + \frac{1}{r_{lu}} + \frac{1}{r_{dc} + r_{cl}} + \frac{1}{r_{ac} + r_{gs}} \tag{31}$$

$$r_s = r_i \left[ 1 + \left( \frac{200\,\mathrm{W\,m^{-2}}}{G + 0.1\,\mathrm{W\,m^{-2}}} \right)^2 \right] \frac{400}{T_C(40\,^\circ\mathrm{C} - T_C)}$$

File `surfdepo.t` (in the `Options` dir): tabulated values of all $r_x$ for 13 land-use classes and 5 seasons.

File `surfdepo.t`: $z_0$ for the 13 land-use classes (no seasonal dependence)

Binary file `IGBP_int1.dat`: IGBP-DIS 1-km Land-Cover Data Set DISCover at a resolution of 0.33°, read by `readlanduse.f90`

If input met data have snow cover, *snow and ice* LU class is used

# Deposition velocity – particles

## Overall formula

Calculated in `partdep.f`

$$v_d = \sum_{i=1}^{n} f_i \left[ \left( r_a + r_{p_i} + r_a r_{p_i} w_{s\,i} \right)^{-1} + w_{s\,i} \right] \tag{32}$$

$r_a$, $r_b$ . . . as for gases
$f_i$ . . . fraction of mass falling into the $i$-th of $n$ classes into which the aerosol particle size distribution is divided
$w_{s\,i}$ . . . gravitational settling velocity calculated according to SLINN1982.

# Deposition velocity – particles

## Settling velocity after Stokes

SLINN1982

$$v_g = \frac{g \rho_p d_p^2 C_{\mathrm{cun}}}{18\mu} \tag{33}$$

$\rho_p$ ... particle density $d_p$ ... particle diameter
$\mu$ ... dynamic viscosity of air $= \rho\nu$ ($1.8 \cdot 10^{-5}\,\mathrm{kg\,m^{-1}\,s^{-1}}$)
$C_{cun}$ ... Cunningham slip-flow correction

$v_g$ is calculated assuming a log-normal size distribution which is divided into $n = 20$ bins:

$$v_g = \sum_{i=1}^{n} f_i \, v_{g\,i}$$

$f_i$ ... mass fraction in bin $i$

$d_p$ (log mean diametre) and $\sigma_d$ (log standard deviation of diametre) have to be specified in the SPECIES file.
For particles, $\rho_g > 0$ must be specificed in SPECIES file.

# Deposition velocity – particles

## Particle-specific laminar sublayer resistance

Stokes number:

$$\mathrm{St} = \frac{w_{s\,i} u_*^2}{g\nu}$$

$$r_{p\,i} = \begin{cases} (u_*\,\mathrm{Sc})^{-1} & \mathrm{St} < 0.6 \\ \left[u_*\left(\mathrm{Sc} + 10^{\mathrm{St}_i/3}\right)\right]^{-1} & \mathrm{St} \geq 0.6 \end{cases} \tag{34}$$

St . . . Stokes number
Sc . . . Schmidt number (see above)

# Gravitational settling

## Mean displacement by settling

In addition to the role of settling for dry deposition, settling also leads to a slow sinking of particles which is considered as

$$w = w + w_s \tag{35}$$

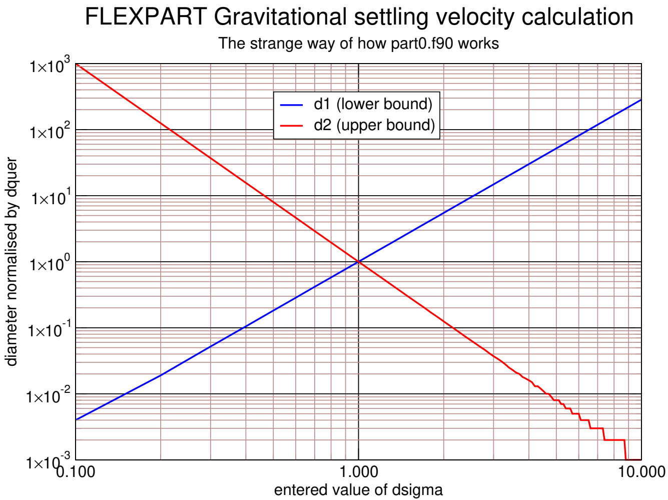
$w$ . . . resolved-scale vertical velocity

## Multiple species and settling

- Because the settling affects the motion of computational particles, each particle can carry only species having the same $d, \sigma_d$.
- Current implementation in FLEXPART allows only one set of $d, \sigma_d$ per run.
- The values from the first species are silently used for all subsequent species!!

# The logarithmic standard deviation

The range of normalised diametres used to calculate $w_s$ and other size-dependent quantities depends on $\sigma_d$(dsigma). Note: $> 1$ recommended, $= 1$ undefined :



# Details of settling velocity

## Iterative determination of $w_s$ and Reynolds number Re

$w_s$ is initialised with the Stokes value ($v_g$)

$$\mathrm{Re} = \frac{d_p\, w_s\, \rho(T)}{\mu(T)}. \tag{36}$$

$$w_s = -\sqrt{\frac{4 g \rho_p d_p C_{\mathrm{Cun}}}{3 c_d \rho}} \tag{37}$$

where the aerodynamic resistance coefficient is

$$c_d = \begin{cases} 24\,\mathrm{Re}^{-1} & \mathrm{Re} < 1.917 & 12.52 \\ 18.5\,\mathrm{Re}^{-0.6} & 1.917 \leq \mathrm{Re} < 500 & 12.52 - 0.444 \\ 0.44 & \mathrm{Re} > 500 & 0.44 \end{cases} \tag{38}$$

$$C_{cun} = 1 + 1.306\cdot 10^{-7}(1.257 + 0.4 e^{-8.42\cdot 10^6 d}) d^{-1} \tag{39}$$

$C_{cun} \approx 1.16$ for $d = 1\cdot 10^{-6}$ m

# Introduction to wet deposition

- What is wet deposition? Transfer of trace substance mass from the atmosphere to the ground through precipitation.
- Mechanisms:
  - Particles serving as cloud condensation nuclei or ice nuclei.
  - Impaction between particles and hydrometeors
  - Solution of gases into hydrometeors
- Regimes / regions:
  - In-cloud scavenging (high efficiency)
  - Below-cloud scavenging (lower efficiency)
  - Not considered in FLEXPART: Occult deposition due to settling / interception by vegetation of fog droplets (was important after Fukushima and for acid rain)

# Schemes in FLEXPART

1. Simple scheme in FLEXPART v6 and before
2. In-cloud / below-cloud scheme with parameterised cloud water content in FLEXPART v8 and v9
3. In-cloud / below-cloud scheme with cloud water content from ECMWF meteorological fields in FLEXPART v10

# Wet deposition – simple scheme

Change of particle mass due to wet deposition:

$$m(t+\Delta t) = m(t)\exp(-\Lambda \Delta t) \tag{40}$$

$$\Lambda = A\, I^B \tag{41}$$

$\Lambda$ ... scavenging coefficient ($\mathrm{s}^{-1}$)
$I$ ... precipitation intensity ($\mathrm{mm\,h}^{-1}$)
$A, B$ ... coefficients read from SPECIES file. $A \approx 1{\cdot}10^{-4} \ldots 1{\cdot}10^{-6}$, $B$ either 0.6 or 0.8.
The mass removed $(m(1-e^{\Lambda\Delta t})$ is added to the 2D wet deposition field.

Used in v6 and earlier, and as a fallback method in later versions.

Also used for below-cloud scavenging in later versions: in v8 and v9 with hardcoded A, B; in v10 read from SPECIES as `PWETA_GAS`, `PWETB_GAS`

# Subgrid-scale variability of precipitation rates

As the relationship between wet deposition and $I$ is nonlinear, and as precipitation is usually inhomogenous within one grid cell, this should be considered following HERTEL1995.

$$F = \max\left[0.05, \frac{(If)_{\mathrm{LSP}} + (If)_{\mathrm{CP}}}{I_{LSP+CP}}\, c\right] \qquad (42)$$
$$I_s = \frac{I_{LSP+CP}}{F}$$

$F$ ... fraction of grid cell with precipitation
$f$ ... fraction of grid cell covered by C/LS precipition
LSP ... large-scape precipitation (ECMWF variable)
CP ... convective precipitation (ECMWF variable)
$c$ ... cloud cover fraction (ECMWF variable)

# Subgrid-scale variability / 2

Table: Factors used for the calculation of the area fraction of a grid cell that experiences precipitation.

| | $I \leq 1$ | $1 < I \leq 3$ | $3 < I \leq 8$ | $8 < I \leq 20$ | $20 < I$ |
|---|---|---|---|---|---|
| $f_{LSP}$ | 0.50 | 0.65 | 0.80 | 0.90 | 0.95 |
| $f_{CP}$ | 0.40 | 0.55 | 0.70 | 0.80 | 0.90 |

Note: This parameterisation comes from a time when model data had typically 1° resolution. For today's high resolution models, this may be invalid, but is still used in FLEXPART.

These factors are also used in later FLEXPART versions!

# In-cloud scheme for FLEXPART v8, v9

Cloud: part of the model column where relative humidity $U > 80\%$ (single layer only)

## In-cloud scheme v8,v9 according to HERTEL1995

Particles:

$$S = \frac{0.9}{c_w} \tag{43}$$

$$\Lambda = \frac{SI}{\Delta H} = \frac{4.5 \cdot 10^6 \text{ m } I^{0.64}}{\Delta H} \tag{44}$$

Gases:

$$S = \frac{1}{(1 - c_w)(HRT)^{-1} + c_w} \tag{45}$$

$$\Lambda = \frac{SI}{\Delta H} \tag{46}$$

$\Delta H$ ... cloud height $c_w = 2 \cdot 10^{-7} I^{0.36}$ ... parameterised column cloud water (liquid+solid),

in $\text{kg m}^{-2}$ for $I$ in $\text{mm h}^{-1}$

$H$ ... Henry's constant (solubility, read from SPECIES) in odd units
$R$ ... universal gas constant
$T$ ... temperature at particle's location

# In-cloud scheme for FLEXPART v10 acc. to GRYTHE2017

## In-cloud scheme, gases (similar to v8/9)

$$\Lambda = \frac{c_r I_s}{(1 - c_w)(HRT)^{-1} + c_w} \tag{47}$$

$c_r = 6.2$ ... empirical cloud water replenishment factor

## In-cloud scheme, particles

$$\Lambda = c_r I_s \left[ \left( \alpha f_{\text{ccn}} + (1 - \alpha) \right) f_{\text{in}} \right] \tag{48}$$

$$\alpha = \max \left( \min \left[ \left( \frac{T - 253\,\text{K}}{20\,\text{K}} \right)^2, 1 \right], 0 \right) \tag{49}$$

$f_{\text{ccn}}(d)$ ... fraction of particles acting as CCN
$f_{\text{in}}(d)$ ... fraction of particles acting as ice nuclei
$\alpha$ ... fraction of cloud particles assumed to be in liquid form

# Below-cloud scheme

Later versions still use the simple scheme of Eq. 41, except for particles in v10:

## Size-dependency of below-cloud scavenging in v10

$$\lg\left(\frac{\Lambda}{1\,\mathrm{s}^{-1}}\right) = C_*\left(\sum_{k=0}^{4} a_k D_p^{-k} + b\sqrt{\frac{I_s}{1\,\mathrm{mm\,h}^{-1}}}\right) \tag{50}$$

$D_p = \lg \frac{d_p}{d_{p0}}$

$a_k$, $b$ empirical coefficients, different for rain and snow, see Table below

$C_*$ ... collection efficiencies for rain and snow, from SPECIES files

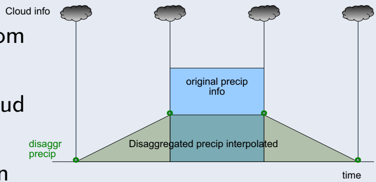
| | $C_*$ | $a_0$ | $a_4$ | $a_3$ | $a_2$ | $a_1$ | $b$ | Reference |
|---|---|---|---|---|---|---|---|---|
| Rain | $C_{\mathrm{rain}}$ | 274.36 | 332839.6 | 226656 | 58005.9 | 6588.38 | 0.24498 | LAAKSO2003 |
| Snow | $C_{\mathrm{snow}}$ | 22.7 | 0 | 0 | 1321 | 381 | 0 | KYRO2009 |

Note: There are problems with the way the curves have been fitted to the data in the original references. These relationships may be changed in the future.

# Problems in the wet deposition schemes

## Inconsistencies in v8/v9 in-cloud/below-cloud scheme

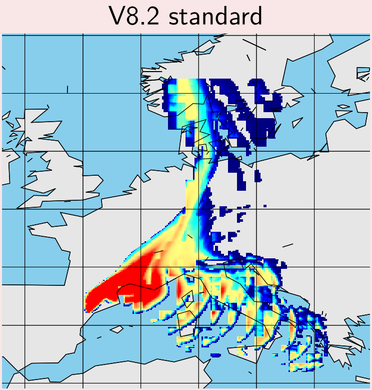
- Scheme needs cloud information (base and top heights), derived from relative humidity fields
- Precip info is for time interval, cloud info is only for a point in time




- Disaggregation spreads precip even more
  - Convective precip falls from clouds which have nothing to do with resolved-scale humidity
  - No cloud $\Rightarrow$ no wet depo – maybe severely underestimated, esp. for CP
  - Depending on combination of plume shape & transport velocity with precip/cloud fields & phase speed, + grid cell sizes, unrealistic depo patterns with met. grid structure can result

# Problems in the wet deposition schemes / 2

## Illustration, deposition in one time step



# Problems in the wet deposition schemes / 3

## Solution implemented in “quick fix” (no official version)

- Make diagnosed clouds more realistic
  - Remove the present `cloud` variables
  - Introduce new variables: cloud base height $z_{cldb}$ and cloud depth $h_{cld}$
  - If we don't get a cloud with $h_{cld} > 50$ m based on $U_{min} = 90\%$, reduce $U_{min}$ in 5% steps down to 25% or until cloud is found.
  - If diagnosed cloud top is $<6000$ m and precip is dominated by CP, set cloud to 500–8000 m for $<0.1$ mm/h and 0–10 km else (prelim. values based on statics of limited number of cases with CP and diagnosed clouds)
- Interpolate $z_{cldb}$ and cloud depth $h_{cld}$ in time and in space, disregarding grid points without clouds diagnosed
- If no cloud diagnosed but precip present, use old wetdepo scheme (possibly with modified `weta, wetb`)

# Problems in the wet deposition schemes / 3

## Situation in v10

- Cloud diagnostics with relative humidity replaced by cloud-water fields – more reliable
- All temporal interpolation removed in wet deposition – internally consistent, but "nearest neighbour" is not good enough. Checkerboard patterns still visible, especially with high-resolution output grid.
- Implementation of quick fix planned
- Artificial smoothing of precipitation fields in time due to the "disaggregation" in `flex_extract` – a new interpolation scheme which conserves the integral amounts of precipitation in each time interval has been prepared (Hittmeir et al., 2018).

  Integration into FLEXPART is planned.

# Bibliography

**FLEXPART Manuals:**

https://www.flexpart.eu/wiki/FpDocumentation Currently links to v6 ACP paper and v8.2 documentation

**Important papers:**

https://www.flexpart.eu/wiki/FpReferences

https://www.geosci-model-dev.net/special_issue878.html

virtual special issue, should collect all FLEXPART-related papers in GMD

https://www.geosci-model-dev-discuss.net/gmd-2018-333/ v10.3 paper

**References**

Forster, C., A. Stohl, and P. Seibert (2007), Parameterization of convective transport in a Lagrangian particle dispersion model and its evaluation. *J. Climate Appl. Meteorol.* **46**, 403–422. URL: http://journals.ametsoc.org/doi/abs/10.1175/JAM2470.1, doi:10.1175/JAM2470.1.

Hittmeir, S., A. Philipp, and P. Seibert (2018), A conservative reconstruction scheme for the interpolation of extensive quantities in the Lagrangian particle dispersion model FLEXPART. *Geosci. Model Dev.* **11** (6), 2503–2523. URL: https://www.geosci-model-dev.net/11/2503/2018.

Stull, R. B. (1988), *An Introduction to Boundary Layer Meteorology*. Kluwer Academic Publishers, Dordrecht, Netherlands, 666 pp.

For references not listed, please see the FLEXPART v8.2 and v10 papers.