FLEXPART in a nutshell

Questions? Write delia.arnold-arias@zamg.ac.at

What is FLEXPART?

The FLEXPART is…

flexpart.eu

… a (**Flex**ible) Lagrangian **Part**icle Dispersion Model, originally developed at the University of Natural Resources and Life Sciences in Vienna, further developed by its main developer Andreas Stohl at the Norwegian Institute for Air Research in the Department of Atmospheric and Climate Research and with by group of developers in different institutions

It is released under the GNU General Public License V3.0

- Variety of applications and application environments:
  - Operational / Quasi-operational → CTBTO, ZAMG, Meteoswiss, ...
  - Research → NILU, BOKU-Met, INTE, ZAMG, Reunion University, ...
  - Others → including private companies/consultancies

What is FLEXPART?

First:

- Track the particles in a given velocity field.

Second:

- Model the Sub-grid scale (SGS) unresolved physical processes that affect the particles dispersion:
  - Boundary Layer Turbulence
  - Mesoscale Turbulence
  - Cumulus turbulent convection

Third:

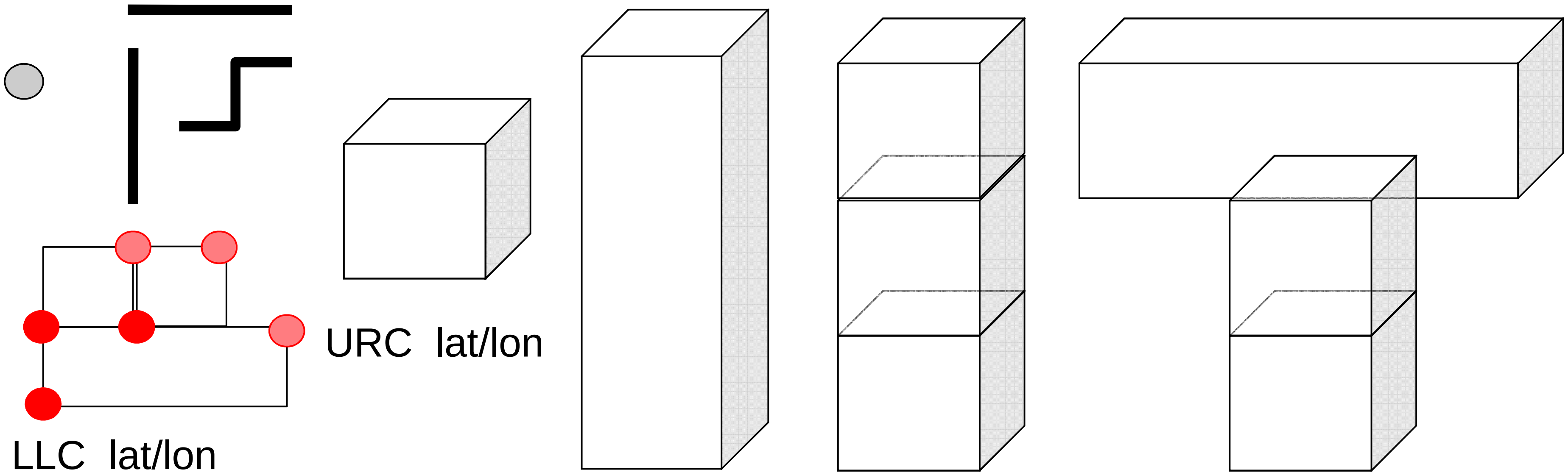
- Modify particles properties based on locally acting processes, e.g. radioactive decay , deposition

Fourth:

- Count particles in a volume and extract concentration value

EMISSIONS:

- Computational particles can carry more than one species (MIND that then there is automatic switching off of gravitational settling)
- Emissions can include radionuclides (radioactive decay specified in the SPECIES_XXX files)
- Emissions can be tracked independently (even if the same species is emitted)
- Releases have to be defined as one or multiple sets of rectangular prisms , allowing for point, line, area and volume sources. Coordinates given in lat lon m or meters in the projection space only for the WRF version



EMISSIONS in the real world:

Linear emission
in the vertical

Eyjafjallajökull frá Hvolsvelli

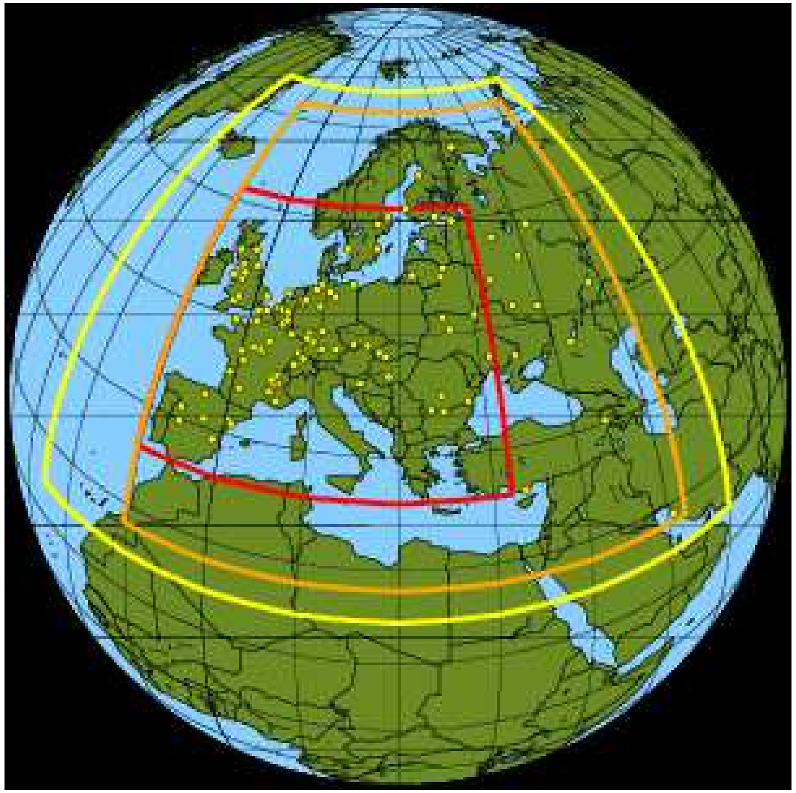
Area sources

More complex
sources

DOMAINS:

- Computational domain defined by the meteorological outermost domain (yellow)
- Output grid must be smaller than computational grid and can contain one single higher resolution (in the horizontal) domain (orange and red)

Nesting is allowed, both in the input (user-defined, usually 2) and in the output (max 1 nest)



… some very useful aspects

- Parallelised
- NetCDF output for easy visualization
- Testing environment
- Easy to automatise
- Improved python-based extraction routines
- Active and engaged developers
- Flexpart.eu system (issue tracking, version control, other information)
- Mailing list: at https://lists.univie.ac.at/mailman/listinfo/flexpart. As the this website has been blocked for certain countries because of high spam levels, you have the alternative to subscribe by sending email to flexpart-request /at/ lists.univie.ac.at (pls correct the address manually - spam protection), no subject and *subscribe flexpart* in the mail body.

Robust, with an active scientific community improving it constantly, used in many research centers BUT also in operational centers! Computationally very efficient, versatile.

User friendly

- No GUI interface
- No detailed manual with examples BUT:
    - Special issue: https://www.geosci-model-dev.net/special_issue878.html
    - Detailed manuscript: https://www.geosci-model-dev-discuss.net/gmd-2018-333/ Pisso, I., Sollum, E., Grythe, H., Kristiansen, N., Cassiani, M., Eckhardt, S., Arnold, D., Morton, D., Thompson, R. L., Groot Zwaaftink, C. D., Evangeliou, N., Sodemann, H., Haimberger, L., Henne, S., Brunner, D., Burkhart, J. F., Fouilloux, A., Brioude, J., Philipp, A., Seibert, P., and Stohl, A.: The Lagrangian particle dispersion model FLEXPART version 10.3, Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-333, in review, 2019.
- No easy to use web interface, not even for past events.
- Compressed difficult to use (but very efficient) output format now alleviated by the NetCDF option.
- ECMWF Meteorological data can not be directly used and processing and expertise is required
- ECMWF data must be extracted with specialized (provided) software.
- Libraries usually are one of the most problematic aspects for new users

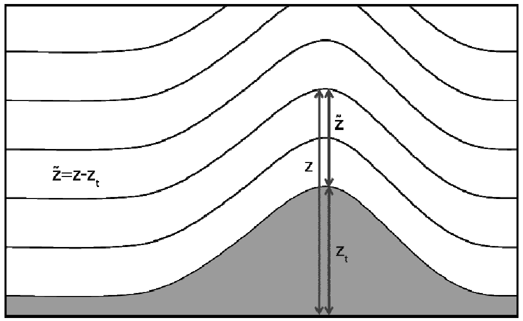
FLEXPART more detail

FLEXPART in a nutshell

COORDINATES:

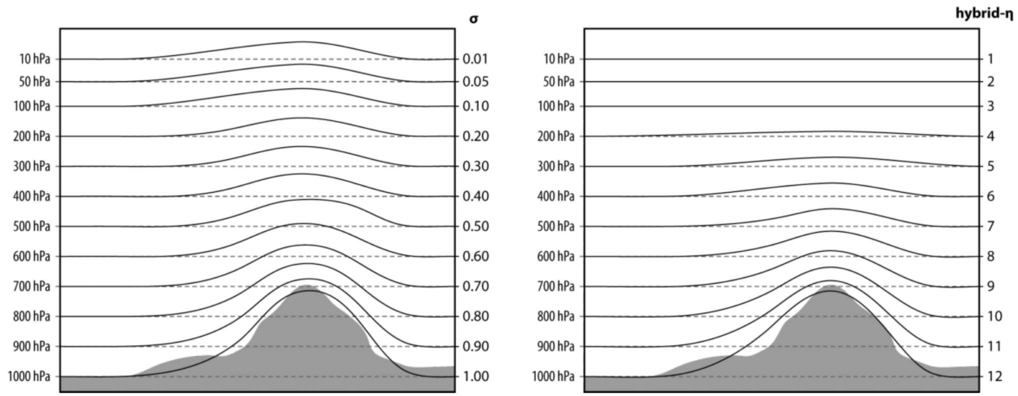
- FLEXPART internal coordinates are regular i,j positions with respect to the lower left coordinate of the computational domain
- FLEXPART needs the vertical velocity in m s^{-1} because of the parametrized random velocities

$$\tilde{w} = \dot{\tilde{z}} = \dot{\tilde{\eta}} \left(\frac{\partial p}{\partial z}\right)^{-1} + \left.\frac{\partial \tilde{z}}{\partial t}\right|_{\eta} + \cancel{\boldsymbol{v}_h \cdot \nabla_\eta \tilde{z}}$$




```
verttransform.f
```

```
verttransform_gfs.f
```

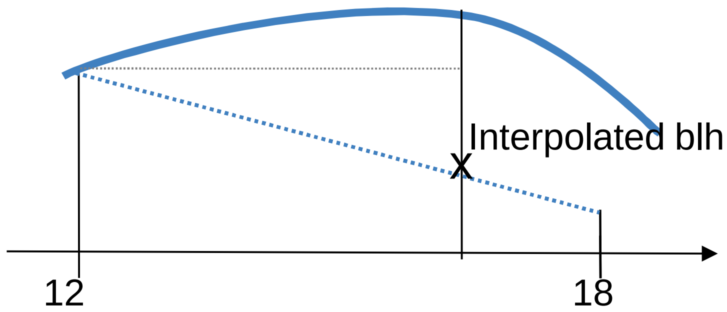


PBL:

- Boundary layer height calculated using critical Richardson number (Vogelezang and Holtslag, 1996). All parameters needed for parametrization coming/derived from the meteorological data (stresses, ustar, L)
- If convective (unstable) situations then one iteration is made (max number iterations 3) to account for excess temperature due to thermal rising

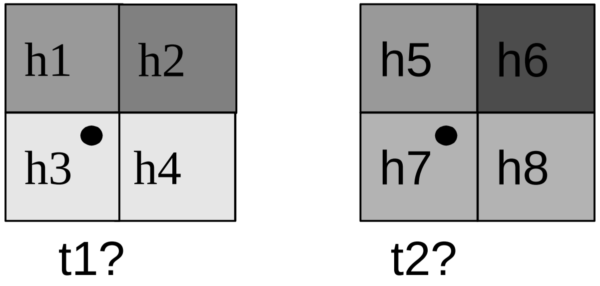
To notice:

- Related to the temporal interpolation of the BL height:



Nothing can be done here → importance of having frequent input data (3 hours)

- Spatial heterogeneities[12] – not interpolated to particle[18] position, but the max in space and time



→ Particle sees blh = h6

`advance.f`

- FLEXPART workaround – envelope mixing height

$$H_{env} = h_{mix} + \min\left[\sigma_Z, c\frac{V}{N}\right]$$

If unstable hmix + sigz

If stable then depending on Froude number

Remember to set an appropriate hmix_min in includepar or similar module!

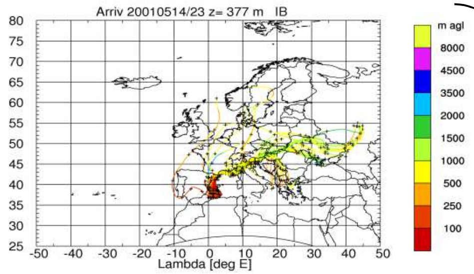
TRANSPORT AND DIFFUSION:

- FLEXPART calculates **trajectories** of *computational* particles (each particle carries a certain amount of mass or mixing ratio of species – *computational* -, as defined in the releases) *(change of mass described later)*

Integration (1st order, zero acceleration scheme)

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{v}[\boldsymbol{X}(t)] \longrightarrow \boldsymbol{X}(t+\Delta t) = \boldsymbol{X}(t) + \boldsymbol{v}(\boldsymbol{X},t)\Delta t$$

$\overline{\boldsymbol{v}}$ Grid scale wind $\rightarrow$ what simple trajectory models use (e.g. FLEXTRA)



$\boldsymbol{v}_t$ Turbulent wind fluctuations

$\boldsymbol{v}_m$ Mesoscale wind fluctuations (meandering)

$$\boldsymbol{v} = \overline{\boldsymbol{v}} + \boldsymbol{v}_t + \boldsymbol{v}_m$$

Turbulent wind fluctuations

- FLEXPART calculates the turbulent motions assuming a Markov process (for the velocity) based on the Langevin equation (Thomson 1987)

$$dv_{t_i} = a_i(\boldsymbol{x}, \boldsymbol{v}_t, t)dt + b_{ij}(\boldsymbol{x}, \boldsymbol{v}_t, t)dW_j$$

drift diffusion Wiener process - stochastic

We need to assume certain statistical proprieties of the atmosphere to sample dW and solve the equation – gaussian turbulence (not valid for skewed turbulence in convective)

$w(0) = w_r$
$\sigma^2_u = avg(u^2)$

$$dw = -w\frac{dt}{\tau_{L_w}} + \left(\frac{2}{\tau_{L_w}}\right)^{1/2} \sigma_w \, dW$$

Average speed non-zero

Turbulent wind fluctuations

HOWEVER two corrections are needed:

1) Drift correction (McNider at al. 1988) – to prevent accumulation of particles in areas of
low-turbulence

2) Density correction (Stohl and Thomson 1999) – to account for the decrease of air density with height

$$dw = - w\frac{dt}{\tau_{L_w}} + \frac{\partial \sigma_w^2}{\partial z} dt + \frac{\sigma_w^2}{\rho}\frac{\partial \rho}{\partial z} dt + \left(\frac{2}{\tau_{L_w}}\right)^{1/2} \sigma_w \, dW$$

Sample only for the vertical velocity, horizontal no drift/density corrections

Which time step?

COMMAND

Two possibilities:

1. Fixed time step (=synchronisation time) without adaptation to the lagrangian timescales → FASTER but LESS ACCURATE (`COMMAND ctl <0`) may be useful for long range applications if computational resources are limited

2. Time step adapting to the vertical lagrangian time scales → turbulence is described in a more accurate/realistic way (`COMMAND ctl >0, ifine`) and thus needed for any BL studies – the time step for the horizontal will be defined as:

$$\Delta t_i = \frac{1}{c_{tl}} \min\left(\tau_{L_w}, \frac{h}{2w}, \frac{0.5}{\partial\sigma_w/\partial z}\right)$$

The time step in the vertical is split: $\Delta t_w = \Delta t_i / \mathtt{ifine}$

$\sigma_{v_i} \quad \tau_{L_i}$

Vertical profiles of the turbulent quantities inside the ABL

Depend on the state of the turbulent atmosphere.
Following Hanna 1982.

1. Unstable meteorological conditions

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

$z/h < 0.1$ and $z - z_0 > -L$

$$\tau_{L_w} = 0.1 \frac{z}{\sigma_w \left[0.55 - 0.38\left(z - z_0\right)/L\right]}$$

$z/h < 0.1$ and $z - z_0 < -L$

$$\tau_{L_w} = 0.59 \frac{z}{\sigma_w}$$

$z/h > 0.1$

$$\tau_{L_w} = 0.15 \frac{h}{\sigma_w}\left[1 - \exp\left(\frac{-5z}{h}\right)\right]$$

2. Neutral

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

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

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

3. Stable

$$\frac{\sigma_u}{u_*} = 2.0\left(1-\frac{z}{h}\right) \quad \frac{\sigma_v}{u_*} = \frac{\sigma_w}{u_*} = 1.3\left(1-\frac{z}{h}\right)$$

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

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

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

What about above the ABL?

In the free atmosphere turbulence is in small places coming from gravity waves, around jet streams... it is not yet parameterized in detail.

FLEXPART treats the stratosphere with a constant vertical diffusivity (Legras et al. 2003)

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

And a constant horizontal diffusivity in the free troposphere

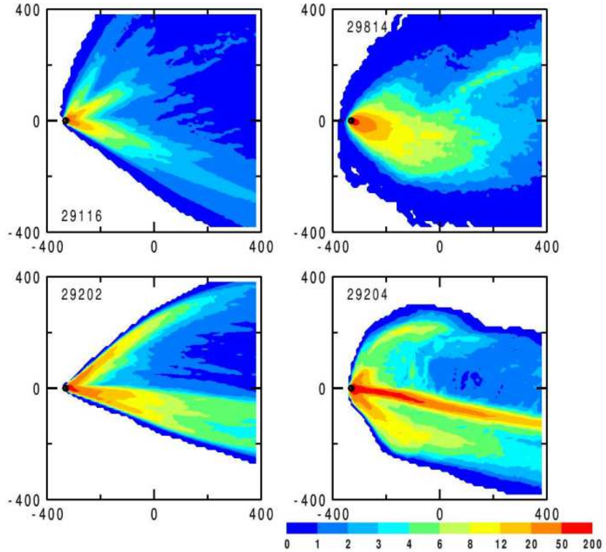
$$D_h = 50\ \mathrm{m}^2\mathrm{s}^{-1}$$

with an intermediate zone from free-t[illegible]o stratosphere. Turbulent velocity scales are then calculated by

$$\sigma_{v_i} = \sqrt{D_i/dt}$$

Mesoscale wind fluctuations (important for very low wind conditions)

- Fluctuations neither resolved by ECMWF nor by the turbulence parameterization, created by mountain waves, pulsating drainage flows, wake vortices, ...
- FLEXPART solves another Langevin equation and assumes that the variance of the wind a the grid scale provides sub-grid scale information (Maryon 1998) → this acts as a source of dispersion (*without really physically representing the phenomena*)



Vickers, D., Mahrt, L. and Belusic, D. 2008. Particle simulations of dispersion using observed meandering and turbulence. Acta Geophys. 56 (with permission of authors)

Gravitational settling

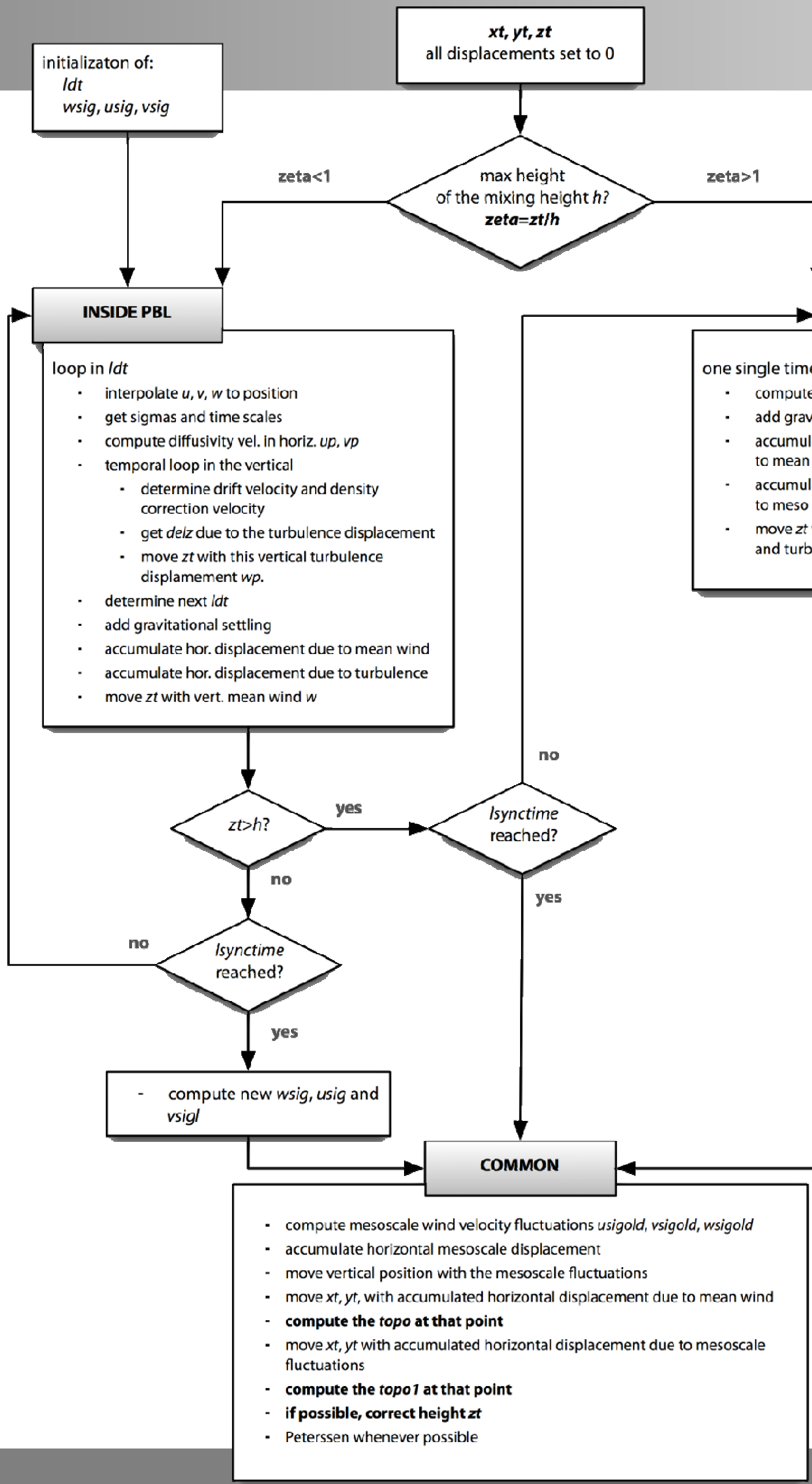
- Trajectories for aerosols are modified due to gravitational settling – a vertical velocity due to gravitational settling is added to the vertical velocity at each time step (get_settling.f90)
- Based on obtaining the Reynolds number and settling velocity with dependence on temperature of the dynamic viscosity(Naeslung and Thaning 1991 and Sutherland 1893)
- Particles are considered spheres

FLEXPART in a nutshell: more?

See Seibert’s and Thompson lectures:

- Deep convection
- Loss processes:
  - Dry deposition
  - Wet deposition
  - OH reaction
  - Radioactive decay

The life of a particle



Additional features … many

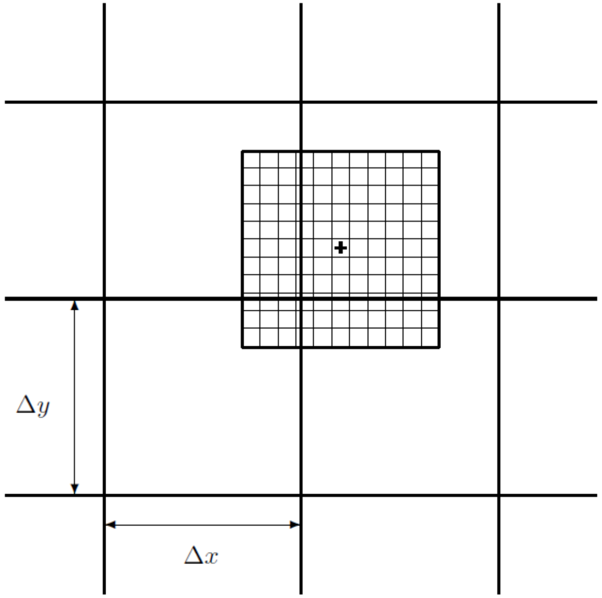
Plume trajectories

Particles are clustered into a number of trajectories (ncluster in includepar) via k-means clustering using the **horizontal** distance as the distance function. The centroids are given by x,y,z but z is not used in the clustering algorithm

Warm start

Particle positions may be dumped (with additional information) and used for a warm start.

Concentration /residence time uniform kernel



The mass of a particle is distributed into the adjacent grid cells.

$$C_{T_s} = \frac{1}{V}\sum_{i=1}^{N}(m_i f_i)$$

It is not used the 3 hours after the particle release to avoid smoothing → CAREFUL! The default flexpart version does use the kernel for dry and wet deposition regardless the time.

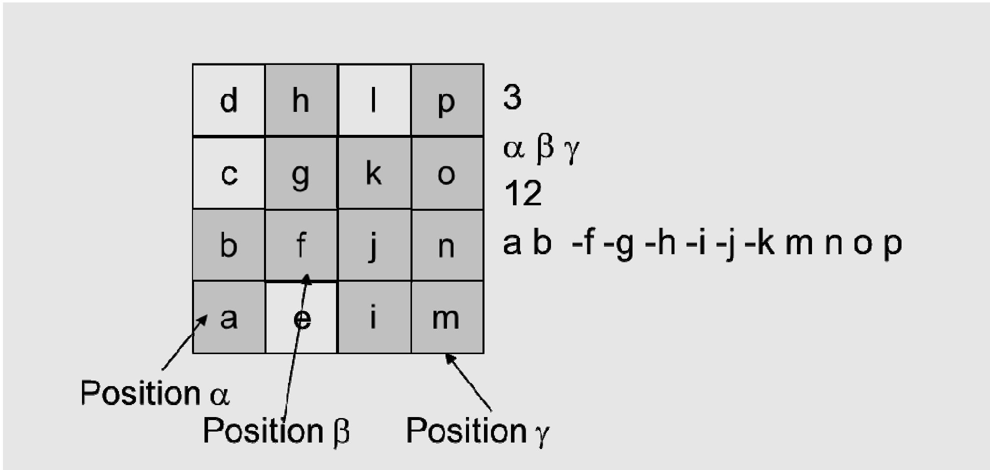
To know:

1. Concentrations are given as time averages whereas deposition is accumulated within the interval time and along the run

2. Radioactive decay IS applied to the deposited substance

Condensed sparse matrix output

pseudocode:




```
For species
 For maxpoint
    For ageclasses
        (wetdepo)
        sp_count_i=0
        sp_count_r=0
        sp_fact = -1
        sp_zer = .true.
             For y direction
                For  x direction
                   If wetgrid > 0 then
                         If sp_zer = .true. Then → first non 0 value
                              sp_count_i = spcount _i + 1  (total number of non-zero segments)
                              sparse_dump_i  →  index of non-zero segment
                              sp_zer= . False.
                              sp_fact = sp_fact*-1
                          sp_count_r = sp_count +1
                          sparse_dump = value of wet deposition (FACTORS! 1E12)
                     Else no wetgrid and sp_zer = .true


                             write(unitoutgrid) sp_count_i
                   write(unitoutgrid) (sparse_dump_i(i),i=1,sp_count_i)
                   write(unitoutgrid) sp_count_r
                   write(unitoutgrid) (sparse_dump_r(i),i=1,sp_count_r)
```

FLEXPART MODES

Forward and backward runs

Forward: the particles are released from the emission location(s) and a 4-D (space + time) concentration output is the result. Deposition fields are not output

Direction	file name	ind_source	ind_receptor	input unit	output unit
Forward	grid_conc*	1	1	kg	ng m^{-3}
	grid_conc*	1	2	kg	ppt by mass
	grid_conc*	2	1	1	ng m^{-3}
	grid_conc*	2	2	1	ppt by mass
	grid_conc*	1	1 or 2 (dep.)	kg	ng m^{-2}
	grid_conc*	2	1 or 2 (dep.)	1	ng m^{-2}
	grid_pptv* (IOUT=2, 3)	1	1	1	ppt by volume

COMMAND

Forward and backward runs

Forward: the particles are released from the emission location(s) and a 4-D (space + time) concentration output is the result. Deposition fields are included in the concentration files

10**12 factor

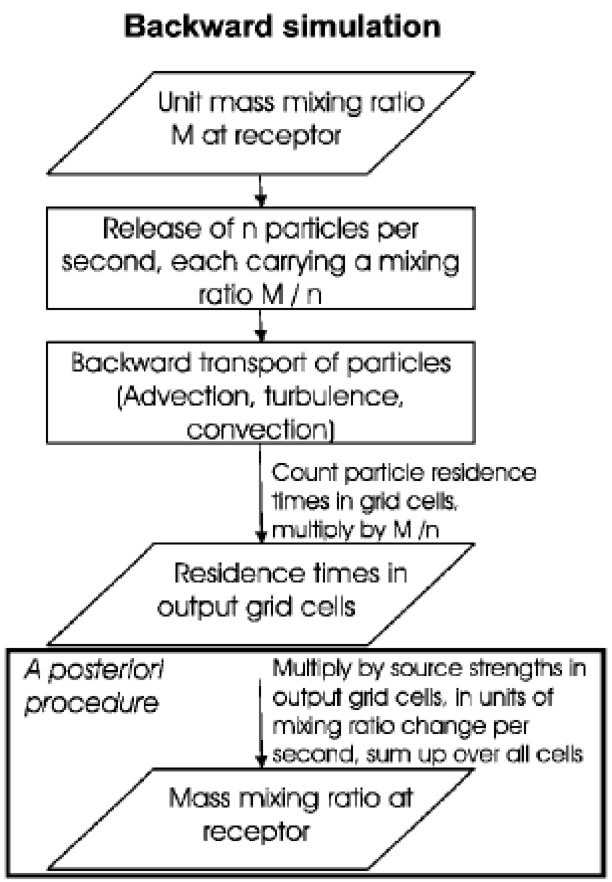
Direction	file name	ind_source	ind_receptor	input unit	output unit
Forward	grid_conc*	1	1	kg	ng m^{-3}
	grid_conc*	1	2	kg	ppt by mass
	grid_conc*	2	1	1	ng m^{-3}
	grid_conc*	2	2	1	ppt by mass
	grid_conc*	1	1 or 2 (dep.)	kg	ng m^{-2}
	grid_conc*	2	1 or 2 (dep.)	1	ng m^{-2}
	grid_pptv* (IOUT=2, 3)	1	1	1	ppt by volume

COMMAND

Forward and backward runs

Backward: the particles are released from a receptor (measurement site) and a 4-D (space + time) sensitivity function to the emision is given. This is especially useful to calculate source-receptor sensitivities when the number of receptors < number of potential sources. Everything is <u>normalized</u> with the mass released so any value other than 0 is correct. Dry and wet deposition correct the sensitivities. (Seibert 2001, Stohl et al. 2003, Seibert and Frank 2004).Deposition fields are not output

Direction	file name	ind_source	ind_receptor	input unit	output unit
Backward	grid_time*	1	1	1	s
	grid_time*	1	2	1	$s\ m^3\ kg^{-1}$
	grid_time*	2	1	1	$s\ kg\ m^{-3}$
	grid_time*	2	2	1	s
	grid_wetdep*	1	3 (wet dep.)	1	m
	grid_drydep*	1	4 (dry dep.)	1	m
	grid_wetdep*	2	3 (wet dep.)	1	$kg\ m^{-2}$
	grid_drydep*	2	4 (dry dep.)	1	$kg\ m^{-2}$
	grid_initial*	1	1	1	1
	grid_initial*	1	2	1	$m^3\ kg^{-1}$
	grid_initial*	2	1	1	$kg\ m^{-3}$
	grid_initial*	2	2	1	1

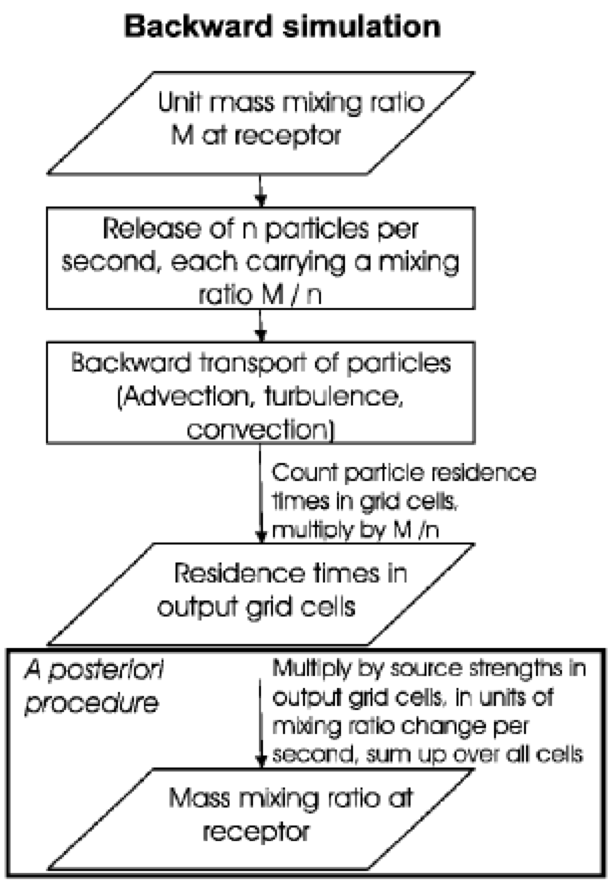


COMMAND

Forward and backward runs

Backward: the particles are released from a receptor (measurement site) and a 4-D (space + time) sensitivity function to the emission is given. This is especially useful to calculate source-receptor sensitivities when the number of receptors < number of potential sources. Everything is normalized with the mass released so any value other than 0 is correct. Dry and wet deposition correct the sensitivities (Seibert 2001, Stohl et al. 2003, Seibert and Frank 2004).

Direction	file name	ind_source	ind_receptor	input unit	output unit
Backward	grid_time*	1	1	1	s
	grid_time*	1	2	1	$\mathrm{s\,m^3\,kg^{-1}}$
	grid_time*	2	1	1	$\mathrm{s\,kg\,m^{-3}}$
	grid_time*	2	2	1	s
	grid_wetdep*	1	3 (wet dep.)	1	m
	grid_drydep*	1	4 (dry dep.)	1	m
	grid_wetdep*	2	3 (wet dep.)	1	$\mathrm{kg\,m^{-2}}$
	grid_drydep*	2	4 (dry dep.)	1	$\mathrm{kg\,m^{-2}}$
	grid_initial*	1	1	1	1
	grid_initial*	1	2	1	$\mathrm{m^3\,kg^{-1}}$
	grid_initial*	2	1	1	$\mathrm{kg\,m^{-3}}$
	grid_initial*	2	2	1	1



COMMAND

NO 10**12 factor

Other interesting aspects

Source receptor matrix calculation of deposited material

- Question: where/when this deposited material may have originated from? Example: ice cores, precipitation samples
- Eckhardt et al. 2017 introduced the option to calculate SRS for dry and wet deposition. See reference paper for details on set-up.
- A couple of details:
  - For dry deposition “backtracking”, particles are released in a shallow layer
  - For wet deposition “backtracking”, particles are released in the whole column, and those with no scavenging are immediately terminated

Other interesting aspects

Sensitivity to initial conditions

- Question: how do we get the SRS for long-lived species and therefore those with “background” non-zero conditions? How do we treat the baseline?
- Since version 8, FLEXPART provides the sensitivity functions to the initial conditions on the receptor.

FLEXPART PRACTICAL ASPECTS

Requirements

Libraries and installation:

- ECCODES / GRIB_API – to decode meteorological data
- For compressed GRIB2 data – Jasper libraries
- If output in NetCDF is wanted, then NetCDF Fortran libraries are needed

Structure and important files

Meteorological data

AVAILABLE

Options files

RELEASES
OUTGRID
COMMAND
SPECIES
…

FLEXPART src

par_mod.f90
makefile

Flexpart structure and input files

	Subdirectory or file	Contents	Description/comments
$flexhome/	pathnames (file)	$options/	path to options directory
		$output/	path to output directory
		$flex_winds/	path to meteorological input files
		$AVAILABLE	path to AVAILABLE file
	src/	*.f90	Fortran source files
		makefile	see Section 4 and Appendix A
		FLEXPART	executable file (see Section 4)
	options/	COMMAND, RELEASES, OUTGRID, SPECIES,	User input files
		AGECLASSES, OUTGRID_NEST, RECEPTORS,	see Section 5 and Table 5
		IGBP_int1.dat , surfdata.t, surfdepo.t, OH_variables.bin	
	AVAILABLE	list of meteorological input data files	file containing list, see Section 5
	output/	FLEXPART output files	see Section 6 and Table 9
	preprocess/	flex_extract/	see Section 5.2
	postprocess/	read_flex_fortran/, read_flex_matlab/,	see Section 6.4
	tests/	development tests for FLEXPART and ancillary software	see Section 7
	tests/examples/	example runs illustrating various FLEXPART functionalities	and Appendix C

FLEXPART

Flexpart structure and input files

	Subdirectory or file	Contents	Description/comments
$flexhome/	pathnames (file)	$options/	path to options directory
		$output/	path to output directory
		$flex_winds/	path to meteorological input files
		$AVAILABLE	path to AVAILABLE file
	src/	*.f90	Fortran source files
		makefile	see Section 4 and Appendix A
		FLEXPART	executable file (see Section 4)
	options/	COMMAND, RELEASES, OUTGRID, SPECIES,	User input files
		AGECLASSES, OUTGRID_NEST, RECEPTORS,	see Section 5 and Table 5
		IGBP_int1.dat , surfdata.t, surfdepo.t, OH_variables.bin	
	AVAILABLE	list of meteorological input data files	file containing list, see Section 5
	output/	FLEXPART output files	see Section 6 and Table 9
	preprocess/	flex_extract/	see Section 5.2
	postprocess/	read_flex_fortran/, read_flex_matlab/,	see Section 6.4
	tests/	development tests for FLEXPART and ancillary software	see Section 7
	tests/examples/	example runs illustrating various FLEXPART functionalities	and Appendix C

Flexpart input files

File name	Content
AGECLASSES†	Age-class definitions
COMMAND	Main control parameters
OUTGRID	Output grid definition
OUTGRID_NEST†	Nested output grid definition
RECEPTORS†	Receptor locations for receptor kernel output
RELEASES	Specification of the sources (forward run) or receptors (backward run)
SPECIES/	Directory containing files with definitions of physical and chemical parameters of species referenced in RELEASES
IGBP_int1.dat	Land cover input data
surfdata.t	Roughness length, leaf area index for different land cover types
surfdepo.t	Seasonal surface resistances for different land cover types
OH_variables.bin†	OH field

Flexpart output files

name	format	switches	description of contents
header	binary	default output	run metadata + ancillary data
header_txt	text	default output	human readable run metadata (from COMMAND)
header_txt_releases	text	default output	human readable run metadata (from RELEASES)
dates	text	default output	Time series: dates of output files
grid_conc_date_nnn	binary (sparse array)	LDIRECT=1 IOUT=1, 3, 5	3D tracer mass density + 2D deposition
grid_pptv_date_nnn	binary (sparse array)	LDIRECT=1 IOUT=2, 3	3D tracer mass density + 2D deposition
grid_time_date_nnn	binary (sparse array)	LDIRECT=-1 IOUT=1	3D sensitivity of atmospheric receptor to emissions
grid_drydep_date_nnn	binary (sparse array)	LDIRECT=-1 IOUT=1, IND_RECEPTOR=3	3D sensitivity of dry deposition receptor to emissions
grid_wetdep_date_nnn	binary (sparse array)	LDIRECT=-1 IOUT=1, IND_RECEPTOR=4	3D sensitivity of wet deposition receptor to emissions
grid_conc_date_nnn.nc	binary (NetCDF)	LDIRECT=1 IOUT=9,11,13	3D tracer + 2D wet and dry deposition
grid_time_date_nnn.nc	binary (NetCDF)	LDIRECT=-1 IOUT=9	3D sensitivity of atmospheric receptor to emissions
grid_drydep_date_nnn.nc	binary (NetCDF)	LDIRECT=-1 IOUT=9, IND_RECEPTOR=3	3D sensitivity of dry deposition receptor to emissions
grid_wetdep_date_nnn.nc	binary (NetCDF)	LDIRECT=-1 IOUT=9 , IND_RECEPTOR=3	3D sensitivity of wet deposition receptor to emissions
grid_initial_nnn	binary (sparse array)	LDIRECT=-1, LINIT_COND>0	3D sensitivity of receptor concentrations/deposition to initial conditions
partposit_date	binary	IPOUT=1,2	particle positions
		IPOUT=1,2 MQUASILAG=1	particle positions numbered consecutively
trajectories.txt	text	IOUT=4, 5	clustered trajectories

Flexpart output files

receptor_conc	binary	LDIRECT=1 IOUT=1, 3, 5, 9, 11, 13	mass density at receptors
receptor_pptv	binary	LDIRECT=1 IOUT=2, 3, 9, 11	volume mixing ratio at receptors
header_nest	binary	NESTED_OUTPUT=1	nest metadata + ancillary data
grid_conc_nest_date_nnn	binary (sparse array)		
grid_pptv_nest_date_nnn	binary (sparse array)	as for mother grid	as for mother grid in a
grid_time_nest_date_nnn	binary (sparse array)	+ NESTED_OUTPUT=1	higher resolution latitude-longitude grid
grid_drydep_nest_date_nnn	binary (sparse array)		
grid_wetdep_nest_date_nnn	binary (sparse array)		
grid_conc_nest_date_nnn.nc	binary (NetCDF)		
grid_time_nest_date_nnn.nc	binary (NetCDF)		
grid_drydep_nest_date_nnn.nc	binary (NetCDF)		
grid_wetdep_nest_date_nnn.nc	binary (NetCDF)		