The FLEXPART Particle Dispersion Model
Version 3.0

User Guide

ANDREAS STOHL

Lehrstuhl für Bioklimatologie und Immissionsforschung,

University of Munich

Am Hochanger 13, 85354 Freising, Germany

February 13, 1998

Contents

1 Introduction **1**

2 Input data and vertical model structure **3**

3 Physical parameterization **5**

4 Particle transport and diffusion **9**

4.1 Trajectory calculations 9

4.2 The Langevin equation 9

4.3 Determination of the time steps 11

4.4 The parameterization of the wind fluctuations 12

4.5 Mesoscale velocity fluctuations 14

4.6 Particle splitting 14

5 Radioactive decay **16**

6 Wet deposition **17**

7 Dry deposition **19**

7.1 Dry deposition of gases 19

7.2 Dry deposition of particulate matter 21

7.3 Loss of particle mass due to dry deposition 22

8 Concentration and deposition calculation **23**

9 References **26**

A Input files of the FLEXPART model **30**

A.1 Files in directory windfields 30

A.2 Files in directory options 31

A.3 Output files 43

B Source code description 44

Chapter 1

Introduction

Particle modeling is the Lagrangian approach towards the simulation of atmospheric dispersion. Lagrangian particle models compute trajectories of a large number of individual so-called particles (not necessarily representing real particles, but small air parcels) to describe the transport and diffusion of substances in the atmosphere. Usually they are used to simulate the dispersion of air pollutants, released for instance due to an accident in a nuclear power plant.

A main advantage of Lagrangian models is that there is no artificial numerical diffusion as occurring in Eulerian models. This is of special importance in the vicinity of the source of an air pollutant, where in Eulerian models the pollutant is instantaneously mixed over at least one grid box, which can cause large subsequent transport errors. The description of turbulence with particle models is of similar quality as large eddy simulations, and it is more accurate than with Eulerian models. Particle models are independent of a computational grid and, in principle, provide infinitesimally small resolution. Practically, however, their effective resolution is determined by the number of particles released and by the resolution of meteorological input data.

Tens or hundreds of thousand particles may be required to accurately describe a multiday transport episode. This posts high demands on computer resources and explains why particle models have only become popular in recent years since powerful computers became available. The basis for most of nowadays particle models was developed in a paper by Thomson (1987). A very thorough monograph on the theoretical background of stochastic Lagrangian models has been published by Rodean (1996). Another interesting paper is the one by Wilson and Sawford (1996). Reviews of particle modeling, which also describe practical aspects, were provided by Zannetti (1992) and Uliasz (1994).

FLEXPART is a newly developed Lagrangian particle dispersion model designed for emergency response and research applications. It simulates the long-range transport, diffusion, dry and wet deposition, and radioactive decay of air pollutants released from point, line, area or volume sources. The management of input data was largely taken from FLEXTRA, a kinematic trajectory model (Stohl *et al.,* 1995) developed at the Institute of Meteorology and Physics of the University of Agricultural Sciences in Vienna. The basic transport and diffusion code of the model was developed during the authors military service at the NBC school of the Austrian Forces (version 1.x). The deposition code was developed later on as a contract work for the Central Institute for Meteorology and Geodynamics in Vienna (version 2.x). Recently, the code underwent a major revision (current version 3.0), which optimized the runtime performance, especially when using short time steps. Some minor bugs were found and removed, the numerics of the model has been improved, and a density correction was introduced.

FLEXPART is based on model level data of the T213 L31 numerical weather prediction model of the European Centre for Medium-Range Weather Forecasts (ECMWF). Gridded data of this model are available at a maximum resolution of 0.5° longitude and latitude, 31 vertical levels, and with a frequency of 3 hours. Usually, however, data with lower horizontal and temporal resolution are used.

FLEXPART may be used in forward mode to simulate the dispersion of pollutants released at a location, or in backward mode to determine the "influence area" of a given receptor point. Forward trajectories are calculated in the forward mode, backward trajectories in the backward mode. Deposition processes are only simulated in forward mode.

FLEXPART was recently evaluated using data from practically all large-scale tracer experiments available, namely from the Cross-Appalachian Tracer Experiment (CAPTEX), the Across North America Tracer Experiment (ANATEX) and the European Tracer Experiment (ETEX), comprising a total of 40 usable releases (Stohl *et al.,* 1998). The model performed rather well compared to other models.

Chapter 2

Input data and vertical model structure

FLEXPART is based on gridded meteorological fields (analyses or forecasts) from the numerical weather prediction model of the ECMWF (ECMWF, 1995) given in a latitude/longitude coordinate system. The data are assumed to be in GRIB format. It needs five three-dimensional fields: horizontal and vertical wind components, temperature and specific humidity. If the vertical wind is missing, no three-dimensional trajectories can be computed. If the specific humidity is not available, computed heights above ground are inaccurate. The other fields are essential.

FLEXPART expects the input data on ECMWF model (i.e. η) levels which are defined by a hybrid coordinate system. The conversion from η to pressure coordinates is given by $p_k = A_k + B_k p_s$ and the heights of the η surfaces are defined by $\eta_k = A_k/p_0 + B_k$, where η_k is the value of η at the k^{th} model level, p_s is the surface pressure and p_0 is a pressure constant (101325 Pa). A_k and B_k are coefficients, chosen in such a way that the levels closest to the ground follow the topography, while the highest levels coincide with pressure surfaces. Intermediate levels show a gradual transition between topography-following and pressure levels. The vertical wind in hybrid coordinates is not directly available from the ECMWF archives, but is calculated by the pre-processor that extracts the data. A surface level is defined in addition to the regular ECMWF model η levels. 2 m temperature, 10 m winds and specific humidity from the first regular model level are attributed to this level. The inaccuracy introduced by the non-vanishing wind speed at the surface is marginal in most situations.

As mixing heights and parameterized random velocities (see sections 3 and 4) are given in meters and meters per second, respectively, the ECMWF model levels are transformed to terrain-following Cartesian z coordinates to avoid multiple coordinate transformations during the trajectory computations. The heights of the z levels equal the heights of the η levels at the lower left grid point of the computational domain and at the initial time. Thus, the vertical resolution is identical to that in the η coordinate system and the levels are close to the original levels in order to minimize interpolation errors. Interpolation from the η levels to the z levels is done with linear interpolation.

Surface parameters are needed as two-dimensional fields: surface pressure, total cloud cover, 10 m horizontal wind components, 2 m temperature and dew point temperature, large scale and convective precipitation, sensible heat flux, solar radiation, and east/west and north/south surface stress. If surface stress and heat flux are not available, friction velocity and surface sensible heat flux are computed from 10 m wind, 2 m temperature and their respective values at the first model level using the profile method (Berkowicz and Prahm, 1982). All other parameters must be available.

Topography, land-sea-mask and subgrid standard deviation of topography must be also available to the model. The former two may be included in the meteorological GRIB files, the latter, at the moment, must be provided in an additional file. If dry deposition of gases is to be calculated, a landuse inventory should be provided. For Europe, the inventory of Velde *et al.* (1994) is used.

Chapter 3

Physical parameterization of boundary layer parameters

Accumulated heat fluxes and surface stresses are available for forecasts at the ECMWF. The pre-processor used to extract the input data from the ECMWF archives (not provided with the FLEXPART source code) selects the appropriate short-term (3 h and 6 h) forecasts and deaccumulates the accumulated flux data. Given that deaccumulated flux data are available, friction velocities u_* and surface sensible heat fluxes $(\overline{w'\Theta_v'})_0$ are calculated from the deaccumulated surface stresses and heat fluxes (Wotawa *et al.,* 1996). The total surface stress is computed from

$$\tau = \sqrt{\tau_1^2 + \tau_2^2} \, , \tag{3.1}$$

where τ_1 and τ_2 are the surface stresses in east/west and north/south direction, respectively. Friction velocity is then

$$u_* = \sqrt{\tau/\rho} \, , \tag{3.2}$$

where ρ is the air density.

If deaccumulated surface stresses and surface sensible heat fluxes are not available, the profile method after Berkowicz and Prahm (1982) is applied to wind and temperature data provided at the first model level and at 10 m (for wind) and 2 m (for temperature). This is an iterative procedure. For its initialization, Obukhov length L is set to a large value. The following three equations are solved in five iterations which is sufficient to achieve convergence:

$$u_* = \frac{\kappa \Delta u}{\ln \frac{z_l}{10} - \Psi_m(\frac{z_l}{L}) + \Psi_m(\frac{10}{L})} \, , \tag{3.3}$$

$$\Theta_* = \frac{\kappa \Delta\Theta}{R\left[\ln\frac{z_l}{2} - \Psi_h(\frac{z_l}{L}) + \Psi_h(\frac{2}{L})\right]} , \tag{3.4}$$

$$L = \frac{\overline{T} u_*^2}{g\kappa\Theta_*} , \tag{3.5}$$

where κ is the von Kármán constant (0.4), z_l is the height of the first model level, Δu is the difference between wind speed at the first model level and at 10 m, $\Delta\Theta$ is the difference between potential temperature at the first model level and at 2 m, Ψ_m and Ψ_h are the stability correction functions for momentum and heat (e.g., Businger *et al.,* 1971; Beljaars and Holtslag, 1991), g is the acceleration due to gravity, Θ_* is the temperature scale and $\overline{T}$ is the average surface layer temperature (taken as the temperature at the first model level).

The heat flux is then computed by

$$(\overline{w'\Theta_v'})_0 = -\rho c_p u_* \Theta_* , \tag{3.6}$$

where ρc_p is the specific heat capacity of air at constant pressure.

Whenever possible, deaccumulated flux data from ECMWF should be provided since they are consistent with the ECMWF model calculations. Wotawa and Stohl (1997) have shown that friction velocities and heat fluxes calculated using the profile method are much less accurate.

Planetary boundary layer (PBL) heights are calculated according to Vogelezang and Holtslag (1996) using the critical Richardson number concept. The PBL height h_{mix} is set to the height of the first model level l for which the Richardson number

$$Ri_l = \frac{(g/\Theta_{v1})(\Theta_{vl} - \Theta_{v1})(z_l - z_1)}{(u_l - u_1)^2 + (v_l - v_1)^2 + 100u_*^2} , \tag{3.7}$$

exceeds the critical value of 0.25. Θ_{v1} and Θ_{vl} are the virtual potential temperatures, z_1 and z_l are the heights, and (u_1, v_1), and (u_l, v_l) are the wind speed components at the 1^{st} and l^{th} model level, respectively. Equation 3.7 holds only for stable and neutral conditions, but it is extended to convective situations by replacing Θ_{v1} with

$$\Theta_{v1}' = \Theta_{v1} + 8.5\frac{(\overline{w'\Theta_v'})_0}{w_*} , \tag{3.8}$$

where

$$w_* = \left[\frac{(\overline{w'\Theta_v'})_0 g h_{mix}}{\Theta_{v1}}\right]^{1/3} \tag{3.9}$$

is the convective velocity scale. The second term on the right hand side of equation 3.8 represents a temperature excess of rising thermals. As w_* is unknown beforehand, h_{mix} and w_* are calculated iteratively for unstable conditions.

Spatial and temporal variations of PBL heights on scales not resolved by the ECMWF model play an important role in determining the thickness of the layer over which the tracer material is distributed. To elucidate this, we first consider unresolved temporal variations. Given a typical evolution of a convective mixed layer, the PBL height reaches its maximum value (say 1500 m) in the afternoon (for instance at 1700 LST), before a much shallower stable PBL forms. Now, if the meteorological data are available only at 1200 LST and at 1800 LST and the PBL heights at those times are, say, 1200 m and 200 m, and some standard interpolation procedure is used to determine the PBL height at 1700 LST, the PBL height is significantly underestimated. In the above example, using linear interpolation, a value of 370 m would have been obtained instead of 1500 m. If a tracer release takes place shortly before the breakdown of the convective mixed layer, this leads to a serious overestimation of the surface concentrations (a factor of four in the above example). Even if the release occurs already in the morning hours, the thickness of the tracer cloud in the evening would be restricted to 1200 m in the above example, whereas the correct thickness would be 1500 m.

Similar arguments are valid for spatial variations of PBL heights. A major reason for spatial variability of PBL heights is complex topography. It increases mechanical mixing due to enhanced shear stress, induces gravity waves during stable conditions, and it provides elevated sources of sensible heat during daytime. All these complex effects can produce turbulence and can thereby locally increase the PBL heights above the elevated areas. In addition to topographical effects, significant variations of the PBL heights can also be caused by differences in landuse or soil wetness (Hubbe *et al.,* 1997). If a tracer cloud travels over such a patchy surface, its thickness would be determined by the maximum PBL height experienced along its path rather than by the average PBL height, which is obtained at the grid points of a coarse resolution meteorological model.

The best solution to this problem obviously is to use meteorological data with higher space and time resolution than the data from ECMWF. Unfortunately, such data are not as readily available, and if they are, it is not a priori clear whether the quality of these data (especially of the wind fields) is as high as the quality of the ECMWF data. Another solution would be to use a simple prognostic boundary layer model as an interpolation tool.

However, for the current FLEXPART version, we did not follow one of these solutions. Instead, we used a somewhat arbitrary parameterization to avoid a significant bias in the tracer cloud thickness which would result in strong overestimations of the surface tracer concentrations. To account for spatial variations induced by topography, we added one standard deviation of the subgrid topography to calculate an "envelope" PBL height. To account for temporal and other spatial PBL height variations, we took the maximum PBL height of the eight grid points surrounding a particle's position in space and time, instead of interpolating the PBL heights. Stohl *et al.* (1998) have shown through a comparison with data from 40 releases of perfluorocarbon tracers that this parameterization improves the model results significantly. It removes the overestimation of the surface concentrations which is otherwise found, and also slightly improves the simulation of the horizontal transport patterns.

Chapter 4

Particle transport and diffusion

4.1 Trajectory calculations

FLEXPART is based on FLEXTRA (Stohl *et al.*, 1995), a kinematic trajectory model that can easily be applied to different types of wind fields. Depending on the available data, it is possible to compute several different trajectories with FLEXTRA, of which three-dimensional and terrain-following trajectories can be used within FLEXPART .

In order to save CPU time, FLEXPART , contrary to FLEXTRA, uses the simple scheme

$$\mathbf{X}(t+\Delta t) = \mathbf{X}(t) + \mathbf{v}(\mathbf{X}, t)\Delta t\,, \tag{4.1}$$

which is accurate to the first order, to integrate the trajectory equation

$$\frac{d\mathbf{X}}{dt} = \mathbf{v}[\mathbf{X}(t)]\,, \tag{4.2}$$

with t being time, Δt the time increment used by the model, $\mathbf{X}$ the position vector, and $\mathbf{v} = \overline{\mathbf{v}} + \mathbf{v}_t + \mathbf{v}_m$ the wind vector that is composed of the grid scale wind $\overline{\mathbf{v}}$, the turbulent wind fluctuations $\mathbf{v}_t$ (Zannetti, 1992) and the mesoscale wind fluctuations $\mathbf{v}_m$. Δt is flexible and determined from criteria discussed in section 4.3. With sufficiently small time steps, the approximation 4.1 yields accurate trajectories.

4.2 The Langevin equation

Wind speeds are given on a grid with a resolution that is much too coarse to represent the turbulent eddies occurring in the planetary boundary layer (PBL). These turbulent motions $\mathbf{v}_t$ therefore have to be parameterized. This is done assuming a Markov process based on the Langevin equation (Thomson, 1987) for the individual wind components i

$$dv_{ti} = a_i(\mathbf{x}, \mathbf{v}_t, t)dt + b_{ij}(\mathbf{x}, \mathbf{v}_t, t)dW_j\,, \tag{4.3}$$

where the drift term a and the diffusion term b are functions of the position, the turbulent velocity and time. The dW_j are incremental components of a Wiener process with mean zero and variance dt, which are uncorrelated with the other components and are uncorrelated in time. This form of the Langevin equation was described by Legg and Raupach (1982).

Gaussian turbulence is assumed in FLEXPART, which is strictly valid only during stable and neutral conditions. During convective condtions, when turbulence is skewed (i.e. larger areas are occupied by downdrafts than by updrafts), this assumption is violated, but for long-range transport the error due to this is not so large. Cross-correlations between the different wind components are not taken into account, since Uliasz (1994) has shown that they have little effect for long-range dispersion.

Although density decreases with height, most particle models assume constant density flows. Especially for deep boundary layers, this is not justified. Stohl and Thomson (1998) have developed a density correction for Gaussian turbulence that takes the density effect into account.

With the above assumptions, the Langevin equation for the vertical wind component can be written as

$$dw = -w\frac{dt}{\tau_{L_w}} + \frac{\partial \sigma_w^2}{\partial z}dt + \frac{\sigma_w^2}{\rho}\frac{\partial \rho}{\partial z}dt + \left(\frac{2}{\tau_{L_w}}\right)^{1/2} \sigma_w dW\,, \tag{4.4}$$

where w is the turbulent vertical wind component, σ_w is the standard deviation of the turbulent vertical wind component, τ_{L_w} is the Lagrangian timescale for the vertical velocity autocorrelation and ρ is density. The second and the third term on the right hand side are the drift correction (e.g. McNider *et al.,* 1988) and the density correction (Stohl and Thomson, 1998), respectively. This Langevin equation is identical (except for the density correction term) to the one described by Legg and Raupach (1982).

Alternatively, the Langevin equation can be re-expressed in terms of w/σ_w instead of w (Wilson *et al.,* 1983):

$$d\left(\frac{w}{\sigma_w}\right) = -\frac{w}{\sigma_w}\frac{dt}{\tau_{L_w}} + \frac{\partial \sigma_w}{\partial z}dt + \frac{\sigma_w}{\rho}\frac{\partial \rho}{\partial z}dt + \left(\frac{2}{\tau_{L_w}}\right)^{1/2} dW\,, \tag{4.5}$$

This form was shown by Thomson (1987) to fulfill the well-mixed criterion which states that "if a species of passive marked particles is initially mixed uniformly in position and velocity space in a turbulent flow, it will stay that way" (Rodean, 1996). Although the method proposed by Legg and Raupach (1982) violates this criterion in strongly inhomogeneous turbulence, their formulation was found to be practical, as numerical experiments have shown that it is robust against an increase in the integration time step used by the model. Therefore, equation 4.4 is used when long time steps are allowed to save computation time (see section 4.3); otherwise, equation 4.5 is used. For the horizontal wind components, the Langevin equation is identical to equation 4.4, with vanishing drift and density correction terms.

For the discrete time step implementation of the above Langevin equations (at the example of equation 4.5), two different methods are used. When $(\Delta t/\tau_{L_w}) < 0.5$,

$$\left(\frac{w}{\sigma_w}\right)_{k+1} = \left(1 - \frac{\Delta t}{\tau_{L_w}}\right)\left(\frac{w}{\sigma_w}\right)_k + \frac{\partial \sigma_w}{\partial z}\Delta t + \frac{\sigma_w}{\rho}\frac{\partial \rho}{\partial z}\Delta t + \left(\frac{2\Delta t}{\tau_{L_w}}\right)^{1/2}\zeta\,, \tag{4.6}$$

where ζ is a normally distributed random number with mean zero and unit standard deviation. The subscripts k and $k+1$ refer to subsequent times separated by Δt. When $(\Delta t/\tau_{L_w}) \geq 0.5$,

$$\left(\frac{w}{\sigma_w}\right)_{k+1} = r_w\left(\frac{w}{\sigma_w}\right)_k + \frac{\partial \sigma_w}{\partial z}\tau_{L_w}(1 - r_w) + \frac{\sigma_w}{\rho}\frac{\partial \rho}{\partial z}\tau_{L_w}(1 - r_w) + \left(1 - r_w^2\right)^{1/2}\zeta\,, \tag{4.7}$$

where $r_w = \exp(-\Delta t/\tau_{L_w})$ is the autocorrelation of the vertical wind.

If a particle reaches the surface or the top of the PBL layer, it is reflected according to Wilson and Flesch (1993), and the sign of the turbulent velocity is changed. If a particle reaches the top or the lateral boundary of the model domain, the computation of its trajectory is terminated.

4.3 Determination of the time steps

FLEXPART can be used with one of two different options. The faster one does not adapt the computation time step to the Lagrangian timescale for the autocorrelations of the turbulent velocity fluctuations. Using this method, constant time steps according to the synchronisation time intervals of FLEXPART as specified by the user (see below) are used. Usually, if not very small synchronisation intervals are used, autocorrelations will be very low in this mode. Hence, turbulence is not described well and the density correction does also not work properly. Multiple reflections of the particle at the surface and the PBL top may occur. Nevertheless, for large scale applications, FLEXPART works very well with this option (Stohl *et al.,* 1998), although surface concentrations are usually somewhat underestimated.

If the transport in the PBL shall be described more accurately, the time steps must be limited by the Lagrangian time scales. Since the vertical wind component is the most important one, only τ_{L_w} is used to limit the time step. The user must specify two constants, c_{tl} and $ifine$. The first one determines the time step Δt_i according to

$$\Delta t_i = \frac{1}{c_{tl}} \min\left(\tau_{L_w}\,,\ \frac{h}{2w}\,,\ \frac{0.5\sigma_w}{\partial\sigma_w/\partial z}\right)\ . \tag{4.8}$$

The minimum value of Δt_i is 1 second. Δt_i is used for solving the Langevin equations for the horizontal turbulent wind components and for the interaction between the vertical turbulent motion and the horizontal wind,

For solving the Langevin equation for the vertical wind component, a shorter time step $\Delta t_w = \Delta t_i/ifine$ is used. This strategy (given sufficiently large values for c_{tl} and $ifine$) ensures that the particles stay vertically well-mixed also in very inhomogeneous turbulence, while keeping the computational cost at a minimum (i.e. only solution of the Langevin equation for w at the shortest time steps Δt_w).

4.4 The parameterization of the wind fluctuations

The solution of the Langevin equations requires the knowledge of $\sigma_{v_t i}$ and τ_{L_i} at any time and at any position of a particle trajectory. For their determination, Hanna (1982) proposed a parameterization scheme that is based on the boundary layer parameters h, L, w_*, z_0 and u_*, i.e. PBL height, Monin-Obukhov length, convective velocity scale, roughness length and friction velocity, respectively. It is used here with a modification taken from Ryall and Maryon (1997) for σ_w for convective conditions, as Hanna's scheme does not always yield smooth profiles of σ_w throughout the whole PBL, leading to an unmixing of well-mixed particles. In the following, the subscript u refers to the along-wind components, v to the cross-wind components and w to the vertical components of the turbulent velocities; f is the Coriolis parameter.

Unstable conditions:

$$\frac{\sigma_u}{u_*} = \frac{\sigma_v}{u_*} = \left(12 + \frac{h}{2|L|}\right)^{1/3} \tag{4.9}$$

$$T_{L_u} = T_{L_v} = 0.15\frac{h}{\sigma_u} \tag{4.10}$$

$$\frac{\sigma_w}{w_*} = \left[1.2\left(1 - 0.9\frac{z}{h}\right)\left(\frac{z}{h}\right)^{2/3} + \left(1.8 - 1.4\frac{z}{h}\right)u_*^2\right]^{1/2} \tag{4.11}$$

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

$$T_{L_w} = 0.1 \frac{z}{\sigma_w \left[0.55 - 0.38 \left(z - z_0\right) / L\right]} \tag{4.12}$$

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

$$T_{L_w} = 0.59 \frac{z}{\sigma_w} \tag{4.13}$$

For $z/h > 0.1$:

$$T_{L_w} = 0.15 \frac{h}{\sigma_w} \left[1 - \exp\left(\frac{-5z}{h}\right)\right] \tag{4.14}$$

Neutral conditions:

$$\frac{\sigma_u}{u_*} = 2.0 \exp(-3fz/u_*) \tag{4.15}$$

$$\frac{\sigma_v}{u_*} = \frac{\sigma_w}{u_*} = 1.3 \exp(-2fz/u_*) \tag{4.16}$$

$$T_{L_u} = T_{L_v} = T_{L_w} = \frac{0.5z/\sigma_w}{1 + 15fz/u_*} \tag{4.17}$$

Stable conditions:

$$\frac{\sigma_u}{u_*} = 2.0 \left(1 - \frac{z}{h}\right) \tag{4.18}$$

$$\frac{\sigma_v}{u_*} = \frac{\sigma_w}{u_*} = 1.3 \left(1 - \frac{z}{h}\right) \tag{4.19}$$

$$T_{L_u} = 0.15 \frac{h}{\sigma_u} \left(\frac{z}{h}\right)^{0.5} \tag{4.20}$$

$$T_{L_v} = 0.07 \frac{h}{\sigma_v} \left(\frac{z}{h}\right)^{0.5} \tag{4.21}$$

$$T_{L_w} = 0.1 \frac{h}{\sigma_w} \left(\frac{z}{h}\right)^{0.5} \tag{4.22}$$

Along-wind and cross-wind components are transformed to Cartesian wind components. In the free troposphere ($z > h$), $\sigma_u = \sigma_v = 0.3$ m/s and $\sigma_w = 0$ m/s are used and Lagrangian time scales are set to 1 hour. The minimum τ_{L_u}, τ_{L_v} and τ_{L_w} used are 10 seconds, 10 seconds and 30 seconds, respectively, to avoid excessive computation times for particles close to the surface.

4.5 Mesoscale velocity fluctuations

Mesoscale motions are neither resolved by the ECMWF data nor are they covered by the turbulence parameterization which is based on measurements that are representative for a temporal scale of less than an hour and correspondingly short length scales. This spectral gap must be filled, since mesoscale motions can significantly accelerate the growth of a dispersing plume (Gupta *et al.,* 1997). For this, we use a method that is similar to the one recently described by Maryon (1998), namely to solve an independent Langevin equation for the mesoscale wind velocity fluctuations ("meandering" in Maryon's terms), assuming that they are largely independent from the turbulent fluctuations covered by Hanna's (1982) parameterization. In contrast to Maryon (1998), who used time scales and velocity variances derived from a spectral analysis of a time series of wind measurements at a single station, we assume that the variance of the wind observed at the grid scale provides some information on its subgrid variance. The wind velocity standard deviation used for the mesoscale Langevin equation is set to half the standard deviation of the 16 grid points surrounding the particles' position in space and time. The corresponding time scale is taken as half the interval at which wind fields are available, assuming that the linear interpolation between the grid points can recover half the subgrid variability. According to Stohl *et al.* (1995), who compared wind field data at different resolution, this is not an unlikely assumption. This empirical approach does not describe actual mesoscale phenomena, but it is similar to the ensemble methods which are useful to assess trajectory accuracy (Kahl, 1996; Baumann and Stohl, 1997; Stohl, 1998). It provides an estimate of how the grid scale flow reacts to perturbations on smaller scales. It is not so important in the PBL, where the interaction of the vertical shear of the mean horizontal wind and the turbulent vertical motions dominates plume dispersion, but it is significant for long-range transport in the free troposphere.

4.6 Particle splitting

During the initial phase of dispersion in the atmosphere, the particles stay relatively close together and form a compact cloud. Relatively few particles suffice to simulate this initial phase correctly. After some time, however, the particle cloud gets distorted and particles spread over a much larger area. More particles are now needed to simulate the dispersion adequately. To save computing time for short travel times, but nevertheless give reliable results for long travel times, FLEXPART allows the user to specify a time constant Δt_s. Particles are split into two (each of which receives half of the mass of the original particle) after travel times of Δt_s, $2\Delta t_s$, $4\Delta t_s$, $8\Delta t_s$, and so on.

Chapter 5

Radioactive decay

Radioactive decay is accounted for by reducing the particle mass at each time step according to

$$M(t + \Delta t) = M(t) \exp(-\Delta t/\beta) \; , \tag{5.1}$$

where M is particle mass, and the time constant $\beta = T_{1/2}/\ln(1/2)$ is determined from the pollutants half life $T_{1/2}$. Deposited pollutant mass decays at the same rate.

Chapter 6

Wet deposition

Wet deposition removes particles and gases from the atmosphere. In principle, in-cloud and below-cloud scavenging must be separated (Asman, 1995). However, as data on cloud base height and cloud extension are not available, in-cloud and below-cloud scavenging are treated jointly by means of scavenging coefficients. Using scavenging coefficients, wet deposition takes the form of an exponential decay process (McMahon and Denison, 1979)

$$\frac{dM_i}{dt} = -\Lambda_i M_i \ , \tag{6.1}$$

and

$$M_i(t + \Delta t) = M_i(t) \exp(-\Lambda_i \Delta t) \ , \tag{6.2}$$

where M_i and Λ_i are the particle mass and the scavenging coefficient of species i, respectively.

The scavenging coefficients Λ_i are species-dependent and increase with precipitation rate according to

$$\Lambda_i = A_i I^{B_i} \ , \tag{6.3}$$

where I is the precipitation rate in mm/hour, A_i [s^{-1}] is the scavenging coefficient at I =1 mm/hour and B_i gives the dependency on precipitation rate. Both A_i and B_i must be specified by the user for each species. Scavenging coefficients for snow and rain may be different. For the sake of simplicity and for lack of data, however, the same scavenging coefficients are used for both snow and rain.

Precipitation can vary considerably over the area covered by a single grid cell. As wet deposition depends nonlinearly on precipitation rate, this subgrid variability must be accounted for (Hertel *et al.,* 1995). The area fraction which experiences precipitation

Table 6.1: Correction factors used for the calculation of the area fraction that experiences precipitation. Precipitation rates are in mm/hour.

	I_l and I_c				
Factor	$I \leq 1$	$1 < I \leq 3$	$3 < I \leq 8$	$8 < I \leq 20$	$20 < I$
fr_l	0.50	0.65	0.80	0.90	0.95
fr_c	0.40	0.55	0.70	0.80	0.90

given a certain grid-scale precipitation rate is calculated by

$$F = \max\left[0.05, CC\frac{I_l fr_l(I_l) + I_c fr_c(I_c)}{I_l + I_c}\right] , \tag{6.4}$$

where CC is the total cloud cover, I_l and I_c are the large scale and convective precipitation rates, respectively, and fr_l and fr_c are correction factors that depend on I_l and I_c (see table 6.1). The subgrid scale precipitation rate is then

$$I_s = (I_l + I_c)/F . \tag{6.5}$$

To save computing time, wet deposition is not calculated at each integration time step, but only once during each concentration sampling interval. With sampling intervals being shorter than the intervals between available meteorological fields, the inaccuracy introduced by this simplification is marginal.

Chapter 7

Dry deposition

The removal of particles and gases from the atmosphere by turbulent transfer and subsequent uptake or deposition at the surface constitutes a primary means of cleansing the atmosphere. This process, called *dry deposition*, is usually described by a deposition velocity

$$v_d(z) = -F_C/C(z)\ , \tag{7.1}$$

where v_d, F_C and C are the deposition velocity, and the flux and the concentration of a species at height z within the constant flux layer. A constant deposition velocity v_d may be specified to be used by FLEXPART if detailed information is not available for a substance. More comprehensive parameterizations for gases and particles are also available, but these require detailed specifications of the physical and chemical properties of the substance and may introduce additional uncertainties if these are not well known.

7.1 Dry deposition of gases

The deposition velocities are calculated with the *resistance method* (Wesely and Hicks, 1977). The deposition velocity of gases is given by

$$|v_d(z)| = [r_a(z) + r_b + r_c]^{-1}\ , \tag{7.2}$$

where r_a is the aerodynamic resistance between z and the surface (identical for all gases, but dependent on stability), r_b is the quasilaminar sublayer resistance (dependent on the molecular diffusivity of a species in air and on the thickness of the quasilaminar sublayer), and r_c is the bulk surface resistance (dependent on the type of surface and on the depositing species).

The aerodynamic resistance r_a can be expressed with the flux-profile relationship which is based on Monin-Obukhov similarity theory (see, e.g., Stull, 1988)

$$r_a(z) = \frac{1}{\kappa u_*}[\ln(z/z_0) - \Psi_h(z/L) + \Psi_h(z_0/L)] \ , \tag{7.3}$$

where κ is the von Kármán constant (0.4), L the Monin-Obukhov length and Ψ_h the similarity function for heat (Businger *et al.,* 1971) which corrects for the influence of stability.

Following Erisman *et al.* (1994), the quasilaminar sublayer resistance is

$$r_b = \frac{2}{\kappa u_*}\left(\frac{Sc}{Pr}\right)^{2/3} \ , \tag{7.4}$$

where Sc and Pr are the Schmidt and Prandtl numbers, respectively. Pr is 0.72 and $Sc = \upsilon/D_i$, with υ being the kinematic viscosity of air (typical value 0.15 cm^2s^{-1}) and D_i being the molecular diffusivity of species i in air. The slight dependency of υ on air temperature is formulated in accordance with Pruppacher and Klett (1978).

The surface resistance r_c is given by (Wesely, 1989)

$$r_c = [1/(r_s + r_m) + 1/r_{lu} + 1/(r_{dc} + r_{cl}) + 1/(r_{ac} + r_{gs})]^{-1} \ , \tag{7.5}$$

where r_s, r_m and r_{lu} represent the bulk values for leaf stomatal, leaf mesophyll and leaf cuticle surface resistances (alltogether the upper canopy resistance) , r_{dc} represents the gas-phase transfer affected by buoyant convection in canopies, r_{cl} the resistance of leaves, twig, bark and other exposed surfaces in the lower canopy, r_{ac} the resistance for transfer that depends only on canopy height and density, and r_{gs} the resistance for the soil, leaf litter, etc., at the ground surface. Each of these resistances is parameterized according to chemical reactivity and solubility of the chemical species, and the meteorological conditions. A detailed description of the applied procedure is given by Wesely (1989).

The required data on landuse is taken from a European database developed by van de Velde *et al.* (1994). It provides the area fractions of eight landuse classes on a grid with 10' resolution. The roughness lengths z_0 needed for parameterization of r_a are also estimated from this landuse inventory. Table 7.1 shows the landuse classes and associated values of z_0. Charnock's relationship (see Stull, 1988) is used to calculate z_0 for the classes "Ocean" and "Inland water", because of its dependence on wave height

$$z_0 = 0.016u_*^2/g \ . \tag{7.6}$$

Deposition velocities in a grid cell are calculated for all landuse classes which cover a non-vanishing area of the cell. The average deposition velocity in the cell is then computed

Table 7.1: List of the landuse classes and roughness lengths used by FLEXPART.

Grassland	0.10
Arable land	0.15
Permanent crops	0.30
Forest	0.60
Inland water	Equ. 7.6
Urban areas	0.70
Other	0.10
Ocean	Equ. 7.6

by weighting the individual deposition velocities with the respective area fractions of the landuse classes.

Outside the area of the European landuse database, the landuse classes are determined only from the ECMWF land-sea-mask, attributing the landuse classes "Ocean" to the sea surfaces and "Grasslands" to the land surfaces.

7.2 Dry deposition of particulate matter

The deposition of particulates is calculated according to

$$v_d(z) = [r_a(z) + r_b + r_a(z) r_b v_g]^{-1} + v_g \ , \tag{7.7}$$

where v_g is the gravitational settling velocity (Slinn, 1982).

The settling velocity is calculated from

$$v_g = \frac{g \rho_p d_p^2 C_{cun}}{18 \mu} \ , \tag{7.8}$$

where ρ_p and d_p are the particle density and diameter, μ the dynamic viscosity of air (0.000018 $\text{kgm}^{-1}\text{s}^{-1}$) and C_{cun} the Cunningham slip-flow correction. The quasilaminar sublayer resistance is calculated from the same relationship as for gases, with an additional impaction term. For further details see Slinn (1982).

Settling and dry deposition velocities are strongly dependent on the particulates sizes. Usually, their sizes cover some range (polydisperse aerosol) and it is not justified to describe particulate matter by a single average diameter. FLEXPART assumes a logarithmic normal distribution of the particulate mass over the range of particulate diameters. The user must specify the mean particulate diameter $\overline{d_p}$ and a measure of the variation around

$\overline{d_p}$, σ_p. $\sigma_p = 0.1$ or $\sigma_p = 10$ both indicate that 68% of the mass are between $0.1\overline{d_p}$ and $10\overline{d_p}$. Then, the settling and deposition velocities are calculated for several particle diameters within the range of ± 3 standard deviations of the logarithmic normal distribution. The particulate mass fractions represented by the diameter intervals are used to weight the respective velocities to yield average settling and deposition velocities. A value of $\sigma_p = 1$ indicates monodisperse aerosol, i.e. aerosol with a uniform diameter.

Gravitational settling is important not only for the computation of the dry deposition velocity, but also affects the particles trajectory. For practical reasons, however, gravitational settling velocities are only taken into account for trajectory calculations when the particles represent a single species. If FLEXPART simulates the transport of multiple species, gravitational settling is neglected, because otherwise each species would follow its own trajectory. This would violate the model concept with several species being described by a single particle. Thus, if the dispersion of large particles with significant settling velocities is to be calculated, a single species calculation is suggested.

7.3 Loss of particle mass due to dry deposition

The depositon velocity is calculated for a reference height z_{ref} of 15 m. All particles that are below $2z_{ref}$ are subjected to dry deposition. For all particles below $2z_{ref}$ deposition is described by

$$\frac{dM_i}{dt} = \frac{-v_d M_i}{2z_{ref}} \,, \tag{7.9}$$

The mass lost by deposition is then calculated by

$$\Delta M_i(t) = M_i(t) \left[1 - \exp\left(\frac{-v_d(z_{ref})\Delta t}{2z_{ref}} \right) \right] \,. \tag{7.10}$$

This simple representation allows the deposition velocities to be precalculated on the meteorological grid. A uniform deposition velocity for each species is then applied to all particles between the ground and $2z_{ref}$. This is appropriate especially for the fast option of the model using long time steps. It is less justified when time steps below the Lagrangian time scale are used, since the deposition process could be described more accurately under these conditions.

Chapter 8

Concentration and deposition calculation

Concentrations are calculated on a three-dimensional longitude-latitude grid. The output domain, grid resolution and vertical levels may differfrom the grid on which meteorological input data are given. Two-dimensional deposition fields are calculated over the same spatial domain. Wet and dry deposition as well as total deposition are given separately in the output files of FLEXPART .

The concentration of a pollutant at time t in a grid cell is calculated by sampling the mass fractions of all particles within the grid cell and dividing by the grid cell volume

$$c = \frac{1}{V} \sum_{i=1}^{N} (m_i f_i) \ , \tag{8.1}$$

with V being the grid cell volume, m particle mass, N the total number of particles, and f_i the fraction of the mass of particle i that is attributed to the respective grid cell. This mass fraction is calculated for each particle and each grid cell by a simple uniform kernel with bandwidths $(\Delta x, \Delta y)$, where Δx and Δy are the grid distances on the longitude-latitude output grid. Figure 8.1 illustrates how the mass fractions are calculated. The particle is located at the center of the shaded rectangle with side lengths $(\Delta x, \Delta y)$. Generally, the shaded area stretches over four grid cells. Then, the mass fraction of the particle attributed to each of these four cells equals the fraction of the shaded area that falls within this cell. The same method is used to calculate gridded deposition fields.

In addition to the simple uniform kernel method, a computationally more demanding parabolic kernel method as described in Uliasz (1994) can be used to calculate surface concentrations for a limited number of user-specified receptor points. The concentration

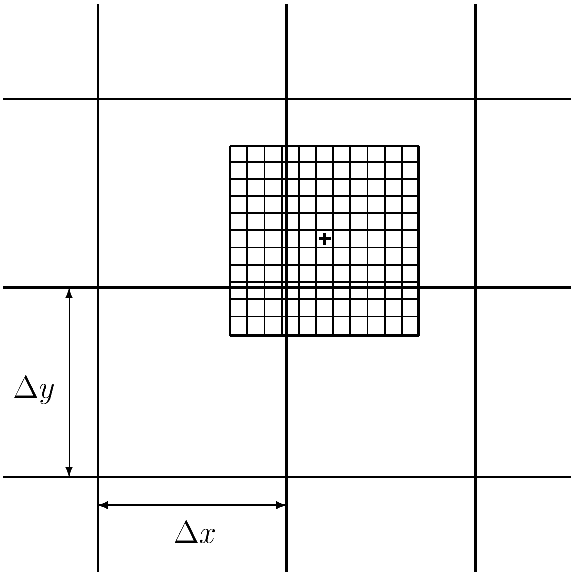

Figure 8.1: Illustration of the uniform kernel used to calculate gridded concentration and deposition fields. The particle position is marked by "+"

is estimated from

$$c(x, y, z = 0) = \sum_{i=1}^{N} \left[\frac{2m_i K(r_x, r_y, r_z)}{h_{x_i} h_{y_i} h_{z_i}} \right] , \tag{8.2}$$

where h_{x_i}, h_{y_i} and h_{z_i} are the kernel bandwidths which determine the degree of smoothing in each coordinate direction, $r_x = (X_i - x)/h_{x_i}$, $r_y = (Y_i - y)/h_{y_i}$, $r_z = Z_i/h_{z_i}$ with X_i, Y_i and Z_i being the position of particle i.

The horizontal kernel bandwidths for each particle are defined according to

$$h_{x_i} = h_{y_i} = 20000 \text{ m } + 0.8 \text{ ms}^{-1}\, t_t + 16500 \text{ m} \sqrt{t_t / 10800 \text{ s}} \,, \tag{8.3}$$

where t_t is the time (in seconds) since the release of particle i. This gives bandwidths of 20, 140 and 380 km after 0, 24 h and 96 h travel time, respectively. This function assumes a linear increase in horizontal trajectory position errors with travel time for long travel times and a faster increase at short travel times (Stohl and Seibert, 1998). Because this definition is based on rather small estimates of trajectory errors (Stohl, 1998), and because the estimated concentrations are most strongly affected by the particles closest to the receptor locations, results of a hypothetically perfect simulation are not spoiled by this definition of the bandwidths. But due to its smoothing effect, it can minimize the impact of small inaccuracies in the transport simulations.

The vertical kernel bandwidth for each particle is given by

$$h_{z_i} = 50 \text{ m } + 31.2 \text{ m} \sqrt{t_t / 10800 \text{ s}} \,. \tag{8.4}$$

The maximum value for h_{z_i} is 150 m, coinciding with the minimum value of the PBL height.

The kernel is defined as

$$K(r_x, r_y, r_z) = \frac{15}{8\pi}(1 - r^2)I \ , \tag{8.5}$$

where $I = 1$ for $r^2 = r_x^2 + r_y^2 + r_z^2 < 1$ and $I = 0$ otherwise.

With both the uniform and the parabolic kernel, concentrations C_{T_c} at time T_c are calculated as time-averages over the period $[T_c - \Delta T_c/2, T_C + \Delta T_c/2]$. ΔT_c must be specified by the user. To calculate the time-averages, concentrations C_{T_s} at times T_s within $[T_c - \Delta T_c/2, T_C + \Delta T_c/2]$ are sampled at shorter intervals ΔT_s and are then divided by the number of samples taken:

$$C_{T_c} = \frac{1}{N} \sum_{i=1}^{N} C_{T_s} \ , \tag{8.6}$$

where $N = \frac{\Delta T_c}{\Delta T_s}$ is the number of samples that fits into time interval ΔT_c. ΔT_s must also be specified by the user. Both ΔT_c and ΔT_s must be multiples of the specified synchronisation interval. The shorter the sampling intervals ΔT_s, the more samples are taken to compute a single average concentration and the more accurate are thus the time-averaged concentrations.

Chapter 9

References

Asman, W.A.H. (1995): Parameterization of below-cloud scavenging of highly soluble gases under convective conditions. *Atmos. Environ.*, 29, 1359-1368.

Baumann, K., and A. Stohl (1997): Validation of a long-range trajectory model using gas balloon tracks from the Gordon Bennett Cup 95. *J. Appl. Meteor.*, **36**, 711-720.

Businger, J.A., J.C. Wyngaard, Y. Izumi, and E.F. Bradley (1971): Flux-profile relationships in the atmospheric surface layer. *J. Atmos. Sci.*, **28**, 181-189.

De Baas, A.F., H. Van Dop, F.T.M. Nieuwstadt (1986): An application of the Langevin equation for inhomogeneous conditions to dispersion in a convective boundary layer. *Q. J. R. Meteorol. Soc.*, **112**, 165-180.

Beljaars, A.C.M., and A.A.M. Holtslag (1991): Flux parameterization over land surfaces for atmospheric models. *J. Appl. Meteor.*, **30**, 327-341.

Berkowicz, R., and L.P. Prahm (1982): Evaluation of the profile method for estimation of surface fluxes of momentum and heat. *Atmos. Environ.*, **16**, 2809-2819.
Businger, J.A., J.C. Wyngaard, Y. Izumi, and E.F. Bradley (1971): Flux-profile relationships in the atmospheric surface layer. *J. Atmos. Sci.*, **28**, 181-189.
ECMWF (1995): *User Guide to ECMWF Products 2.1.* Meteorological Bulletin M3.2. Reading, UK.

Erisman, J.W., A. Van Pul, and P. Wyers (1994): Parametrization of surface resistance for the quantification of atmospheric deposition of acidifying pollutants and ozone. *Atmos. Environ.*, **28**, 2595-2607.

Gupta, S., McNider, R.T., Trainer, M., Zamora, R.J., Knupp, K. and Singh, M.P. (1997): Nocturnal wind structure and plume growth rates due to inertial oscillations. *J. Appl. Meteor.*, **36**, 1050-1063.

Hanna, S.R. (1982): Applications in air pollution modeling. In: Nieuwstadt F.T.M. and H. van Dop (ed.): *Atmospheric Turbulence and Air Pollution Modelling*. D. Reidel Publishing Company, Dordrecht, Holland.

Hertel, O., J. Christensen, E.H. Runge, W.A.H. Asman, R. Berkowicz, M.F. Hovmand, and O. Hov (1995): Development and testing of a new variable scale air pollution model - ACDEP. *Atmos. Environ.*, **29**, 1267-1290.

Hubbe, J.M., Doran, J.C., Liljegren, J.C. and Shaw, W.J. (1997) Observations of spatial variations of boundary layer structure over the southern Great Plains cloud and radiation testbed. *J. Appl. Meteor.*, **36**, 1221-1231.

Kahl, J.D.W. (1996): On the prediction of trajectory model error. *Atmos. Environ.*, **30**, 2945-2957.

Legg, B.J., and M.R. Raupach (1982): Markov-chain simulation of particle dispersion in inhomogeneous flows: the mean drift velocity induced by a gradient in Eulerian velocity variance. *Boundary-Layer Meteorol.*, **24**, 3-13.

Maryon, R.H. (1998) Determining cross-wind variance for low frequency wind meander. *Atmos. Environ.*, **32**, 115-121.

McMahon, T.A., and P.J. Denison (1979): Empirical atmospheric deposition parameters - a survey. *Atmos. Environ.*, **13**, 571-585.

McNider, R.T., M.D. Moran, and R.A. Pielke (1988): Influence of diurnal and inertial boundary layer oscillations on long-range dispersion. *Atmos. Environ.*, **22**, 2445-2462.

Pruppacher, H.R., and J.D. Klett (1978): *Microphysics of clouds and precipitation.* D. Reidel Publishing Company, Dordrecht.

Rodean, H. (1996): Stochastic Lagrangian models of turbulent diffusion. *Meteorological Monographs*, **26** (48). American Meteorological Society, Boston, USA.

Ryall, D.B. and Maryon, R.H. (1997): Validation of the UK Met Office's NAME model against the ETEX dataset. In: Nodop, K. (editor): *ETEX Symposium on Long-Range Atmospheric Transport, Model Verification and Emergency Response*, European Commission EUR 17346, 151-154.

Slinn (1982): Predictions for particle deposition to vegetative canopies. *Atmos. Environ.*, **16**, 1785-1794.

Stohl, A. (1998) Computation, accuracy and applications of trajectories – a review and bibliography. *Atmos. Environ.*In press.

Stohl, A., Hittenberger, M., Wotawa, G. (1998): Validation of the Lagrangian particle dispersion model FLEXPART against large scale tracer experiment data. *Atmos. Environ.*, in press.

Stohl, A., and P. Seibert (1998): Accuracy of trajectories as determined from the conservation of meteorological tracers. *Q. J. R. Meteorol. Soc.*, in press.

Stohl, A. and Thomson, D.J. (1998): A density correction for Lagrangian particle dispersion models. Submitted to *Boundary-Layer Meteorol.*

Stohl, A., G. Wotawa, P. Seibert, and H. Kromp-Kolb (1995): Interpolation errors in wind fields as a function of spatial and temporal resolution and their impact on different types of kinematic trajectories. *J. Appl. Meteor.*, **34**, 2149-2165.

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

Thomson D.J. (1984): Random walk modelling of diffusion in inhomogeneous turbulence. *Q. J. R. Meteorol. Soc.*, **110**, 1107-1120.

Thomson D.J. (1987): Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *J. Fluid Mech.*, **180**, 529-556.

Uliasz, M. (1994): Lagrangian particle dispersion modeling in mesoscale applications. In: Zannetti, P. (ed.): *Environmental Modeling, Vol. II.* Computational Mechanics Publications, Southampton, UK.

Velde van de, R.J., W.S. Faber, V.F. Katwijk van,, J.C.I. Kuylenstierna, H.J. Scholten., T.J.M. Thewessen, M. Verspuij, and M. Zevenbergen (1994): *The Preparation of a European Land Use Database.* National Institute of Public Health and Environmental Protection, Report nr 712401001, Bilthoven, The Netherlands.

Vogelezang, D.H.P., and A.A.M. Holtslag (1996): Evaluation and model impacts of alternative boundary-layer height formulations. *Boundary-Layer Meteorol.*, in press.

Wesely, M.L., and B.B. Hicks (1977): Some factors that affect the deposition rates of sulfur dioxide and similar gases on vegetation. *J. Air Poll. Contr. Assoc.*, **27**, 1110-1116.

Wesely, M.L. (1989): Parameterization of surface resistances to gaseous dry deposition in regional-scale numerical models. *Atmos. Environ.*, **23**, 1293-1304.

Wilson, D.J., and Flesch, T.K. (1993): Flow boundaries in random-flight dispersion models: enforcing the well-mixed condition. *J. Appl. Meteor.*, **32**, 1695-1707.

Wilson, J.D., B.J. Legg, and D.J. Thomson (1983): Calculation of particle trajectories in the presence of a gradient in turbulent-velocity scale. *Boundary-Layer Meteorol.*, **27**, 163-169.

Wilson, J.D., and Sawford, B.L. (1996): Review of Lagrangian stochastic models for trajectories in the turbulent atmosphere. *Boundary-Layer Meteorol.*, **78**, 191-210.

Wilson, J.D., G.W. Thurtell, and G.E. Kidd (1981): Numerical simulation of particle trajectories in inhomogeneous turbulence, II: Systems with variable turbulent velocity scale. *Boundary-Layer Meteorol.*, **21**, 423-441.

Wotawa, G., A. Stohl, and H. Kromp-Kolb (1996): Parameterization of the planetary boundary layer over Europe - a data comparison between the observation based OML preprocessor and ECMWF model data. *Contr. Atmos. Phys.*, **69**, 273-284.

Wotawa, G. and Stohl, A. (1997) Boundary layer heights and surface fluxes of momentum and heat derived from ECMWF data for use in pollutant dispersion models – problems with data accuracy. In: Gryning, S.-E., Beyrich, F. and E. Batchvarova (editors): The Determination of the Mixing Height – Current Progress and Problems. EURASAP Workshop Proceedings, 1-3 October 1997, Risø National Laboratory, Denmark.

Zannetti, P. (1992): Particle Modeling and Its Application for Simulating Air Pollution Phenomena. In: Melli P. and P. Zannetti (ed.): *Environmental Modelling.* Computational Mechanics Publications, Southampton, UK.

Appendix A

Input files of the FLEXPART model

The file `pathnames` must be available in the actual working directory. It gives the names of the directories, where the user can specify his options, where the windfields are stored and where the output is put into. An example of this file is

```
/home/atmos1/as/flexpart3.0/options/
/daten2/windfields/
/home/atmos1/as/flexpart3.0/output/
/daten2/windfields/AVAILABLE
```

```
Line 1: path where control files (e.g. "COMMAND", "RECEPTORS") are available
Line 2: path where meteorological fields are available
Line 3: path where output is written to
Line 4: full filename of "AVAILABLE"-file
```

A.1 Files in directory windfields

The directory `windfields` contains grib code-files of the meteorological fields and other files related to the meteorological input. All meteorological fields must have the same structure, i.e. the same computational domain, the same resolution and must contain the same type of input data. An example listing of this directory is given below.

```
AVAILABLE       EN94102300      EN94102303      EN94102306
EN94102309      EN94102312      EN94102315      LANDSEAMASK
OROGRAPHY       SUBGRIDORO
```

The names of the grib code-files and the times (in UTC) for which their data are valid must be indicated in the file `AVAILABLE`:

```
DATE     TIME         FILENAME     SPECIFICATIONS
YYYYMMDD HHMISS
-------- ------      ----------      ----------
19941023 000000      EN94102300      ON DISC
19941023 030000      EN94102303      ON DISC
19941023 060000      EN94102306      ON DISC
19941023 090000      EN94102309      ON DISC
19941023 120000      EN94102312      ON DISC
19941023 150000      EN94102315      ON DISC
```

`LANDSEAMASK`, `OROGRAPHY` and `SUBGRIDORO` contain the ECMWF land-sea-mask, the topography and the subgrid topography standard deviation of the model domain. All three files are in ASCII. Only the file `SUBGRIDORO` is necessary when the topography and land-sea-mask information is contained in the GRIB files. The subgrid topography is, at the moment, read from the ASCII file in any case.

A.2 Files in directory options

The files in directory `options` are used to specify the model run. An example listing of `options` is given below.

```
COMMAND        RELEASES       surfdata.t
OUTGRID        SPECIES        surfdepo.t
RECEPTORS      landuse.t
```

`COMMAND` specifies (1) the simulation mode (either forward or backward), (2) the start and (3) the end time of the simulation, (4) the frequency T_c, (5) the averaging time ΔT_c and (6) the sampling intervals ΔT_s of the output fields, (7) the time constant for particle splitting Δt_s, (8) the synchronisation interval of FLEXPART , (9) the factor by which the time steps must be below the Lagrangian time scale c_{tl}, and (10) the refinement factor for the Langevin equation for the vertical turbulent wind component $ifine$. If (9) c_{tl} is negative, the Langevin equations are solved with constant time steps according to the synchronisation interval. The synchronisation interval is the minimum time interval used by the model for all activities (such as concentration calculations, wet deposition calculations, interpolation of data, mesoscale wind fluctuations or output of data) other than the simulation of turbulent transport and dry deposition.

```
********************************************************************************
*                                                                              *
*      Input file for the Lagrangian particle dispersion model FLEXPART        *
*                           Please select your options                         *
*                                                                              *
********************************************************************************

1. __                3X, I2
    1
   DIRECTION         1 FOR FORWARD SIMULATION, -1 FOR BACKWARD SIMULATION

2. ________ ______   3X, I8, 1X, I6
   19941023 150000
   YYYYMMDD HHMISS   BEGINNING DATE OF SIMULATION

3. ________ ______   3X, I8, 1X, I6
   19941027 090000
   YYYYMMDD HHMISS   ENDING DATE OF SIMULATION

4. _____             3X, I5
   10800
   SSSSS             OUTPUT EVERY SSSSS SECONDS

5. _____             3X, I5
   10800
   SSSSS             TIME AVERAGE OF OUTPUT (IN SSSSS SECONDS)

6. _____             3X, I5
     900
   SSSSS             SAMPLING RATE OF OUTPUT (IN SSSSS SECONDS)

7. ______            3X, I6
```

```
   999999
   SSSSSS            TIME CONSTANT FOR PARTICLE SPLITTING (IN SECONDS)

8. _____             3X, I5
     450
   SSSSS             SYNCHRONISATION INTERVAL OF FLEXPART (IN SECONDS)

9.  ---.--           4X, F6.4
      5.0
    CTL              FACTOR, BY WHICH TIME STEP MUST BE SMALLER THAN TL

10. ---              4X, I3
     12
    IFINE            DECREASE OF TIME STEP FOR VERTICAL MOTION BY FACTOR IFINE
```

```
1. Simulation direction, 1 for forward, -1 for backward in time

2. Beginning date and time of simulation. Must be given in format
   YYYYMMDD HHMISS, where YYYY is YEAR, MM is MONTH, DD is DAY, HH is HOUR,
   MI is MINUTE and SS is SECOND. Current version utilizes UTC.

3. Ending date and time of simulation. Same format as 3.

4. Average concentrations are calculated every SSSSS seconds.

5. The average concentrations are time averages of SSSSS seconds
   duration. If SSSSS is 0, instantaneous concentrations are oututted.

6. The concentrations are sampled every SSSSS seconds to calculate the time
   average concentration. This period must be shorter than the averaging time.

7. Time constant for particle splitting. Particles are split into two
   after SSSSS seconds, 2SSSSS seconds, 4SSSSS seconds, and so on.

8. All processes are synchronized with this time interval. Therefore,
   all other time constants must be multiples of this value.

9. CTL=factor by which time step must be shorter than the Lagrangian
   time scale
   CTL should be greater 1 if account of Lagrangian time scale is to
   be taken.
   If CTL<0, a purely random walk simulation is done
```

OUTGRID specifies the output grid. Up to 29 vertical levels may be chosen:

```
********************************************************************************
*                                                                              *
*      Input file for the Lagrangian particle dispersion model FLEXPART       *
*                       Please specify your output grid                        *
*                                                                              *
********************************************************************************

1.  ------.----       4X,F11.4
       -10.0000       GEOGRAFICAL LONGITUDE OF LOWER LEFT CORNER OF OUTPUT GRID
    OUTLONLEFT        (left boundary of the first grid cell - not its centre)


2.  ------.----       4X,F11.4
        41.0000       GEOGRAFICAL LATITUDE OF LOWER LEFT CORNER OF OUTPUT GRID
    OUTLATLOWER       (lower boundary of the first grid cell - not its centre)

3.  -----             4X,I5
      101             NUMBER OF GRID POINTS IN X DIRECTION (= No. of cells + 1)
    NUMXGRID

4.  -----             4X,I5
       47             NUMBER OF GRID POINTS IN Y DIRECTION (= No. of cells + 1)
    NUMYGRID

5.  ------.---        4X,F10.3
         0.500        GRID DISTANCE IN X DIRECTION
    DXOUTLON

6.  ------.---        4X,F10.3
         0.500        GRID DISTANCE IN Y DIRECTION
    DYOUTLAT

7.  -----.-           4X, F7.1
      100.0
    LEVEL 1           HEIGHT OF LEVEL (UPPER BOUNDARY)

8.  -----.-           4X, F7.1
      300.0
    LEVEL 2           HEIGHT OF LEVEL (UPPER BOUNDARY)

9.  -----.-           4X, F7.1
      600.0
```

```
   LEVEL 3           HEIGHT OF LEVEL (UPPER BOUNDARY)

10. -----.-          4X, F7.1
     1000.0
   LEVEL 4           HEIGHT OF LEVEL (UPPER BOUNDARY)

11. -----.-          4X, F7.1
     2000.0
   LEVEL 5           HEIGHT OF LEVEL (UPPER BOUNDARY)

12. -----.-          4X, F7.1
     3000.0
   LEVEL 6           HEIGHT OF LEVEL (UPPER BOUNDARY)
```

RECEPTORS specifies the receptor locations for which the kernel method shall be applied to calculate air concentrations. Up to 200 receptors may be specified:

```
*******************************************************************************
*                                                                             *
*      Input file for the Lagrangian particle dispersion model FLEXPART       *
*                        Please specify your receptor points                  *
*     For the receptor points, ground level concentrations are calculated     *
*                                                                             *
*******************************************************************************
1.  ----------------  4X,A16
    F15               NAME OF RECEPTOR POINT
    RECEPTORNAME

2.  ------.----       4X,F11.4
         6.1333       GEOGRAFICAL LONGITUDE
    XRECEPTOR

3.  ------.----       4X,F11.4
        49.0833       GEOGRAFICAL LATITUDE
    YRECEPTOR
================================================================================
1.  ----------------  4X,A16
    NL01              NAME OF RECEPTOR POINT
    RECEPTORNAME

2.  ------.----       4X,F11.4
         5.7833       GEOGRAFICAL LONGITUDE
    XRECEPTOR

3.  ------.----       4X,F11.4
        50.9167       GEOGRAFICAL LATITUDE
    YRECEPTOR
================================================================================
```

`RELEASES` defines the release specifications. In the first input line, the number N of emitted species is defined (two in the example below). At all locations, the same species must be released. The next N input lines give a cross-reference to file `SPECIES`, where the features of the species that are released are specified. Then follow up to 10 releases specified by the trajectory type to be used, the beginning and the ending time of the release, geographical coordinates of the lower left and upper right corners of the release location, lower level and upper level of source height, the number of particles to be used, and the total mass emitted. Please note that the mass specification must be made for as many species (N) as have been specified above. Finally, the name of the release point must be given.

The particles are released evenly distributed within a volume stretching from the lower to the upper level (both given in m above ground) above a rectangular area (in grid coordinates) defined by the geographical coordinates of the lower left and upper right corners. In the following example, two release points are specified, the first a volume source, the second a ground level point source.

```
*********************************************************************
*                                                                   *
*   Input file for the Lagrangian particle dispersion model FLEXPART   *
*                        Please select your options                 *
*                                                                   *
*   Kind of trajectory: 1 = 3-dimensional                           *
*                       2 = 2-dimensional (terrain-following)       *
*                                                                   *
*                                                                   *
*********************************************************************
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  2
---                        i3    Total number of species emitted

  3
---                        i3    Index of species in file SPECIES

  5
---                        i3    Index of species in file SPECIES

=====================================================================
 1
 -                         1X,I1 Kind of trajectory (see file header)

19941023 160000
-------- ------            i8,1x,i6 Beginning date and time of release

19941024 035000
-------- ------            i8,1x,i6 Ending date and time of release

  -2.50
----.----                  f9.4  Longitude [DEG] of lower left corner

  48.00
----.----                  f9.4  Latitude [DEG] of lower left corner

  -2.20
----.----                  f9.4  Longitude [DEG] of upper right corner

  48.40
----.----                  f9.4  Latitude [DEG] of upper right corner

   50.0
-----.---                  f10.3 Lower z-level (in m above ground)

  150.0
-----.---                  f10.3 Upper z-level (in m above ground)

    30000
---------                  i9    Total number of particles to be released

3.0000E00
-.----E--                  e9.4  Total mass emitted
```

```
5.0000E00
_.____E__                e9.4  Total mass emitted

TEST1
________________________________________   character*40 comment
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1
 _                       1X,I1 Kind of trajectory (see file header)

19941023 180000
________ ______          i8,1x,i6 Beginning date and time of release

19941024 030000
________ ______          i8,1x,i6 Ending date and time of release

  -2.50
____.____                f9.4  Longitude [DEG] of lower left corner

  48.00
____.____                f9.4  Latitude [DEG] of lower left corner

  -2.50
____.____                f9.4  Longitude [DEG] of upper right corner

  48.00
____.____                f9.4  Latitude [DEG] of upper right corner

    0.0
_____.___                f10.3 Lower z-level (in m above ground)

    0.0
_____.___                f10.3 Upper z-level (in m above ground)

    10000
_________                i9    Total number of particles to be released

1.0000E00
_.____E__                e9.4  Total mass emitted

1.0000E00
_.____E__                e9.4  Total mass emitted

TEST2
________________________________________   character*40 comment
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

`SPECIES` gives a list of species from which the user can select the ones to be simulated. The file contains the physical properties of these species used to simulate radioactive decay, and wet and dry deposition. For the wet deposition, A and B specify the factors defined by equation 6.3.

For dry deposition of gases, $D = D_{H_2O}/D_i$, where D_{H_2O} is the diffusivity of water vapor and D_i is the diffusivity of the species, H is the effective Henry's constant, and $f0$ varies between 0 and 1 and gives the reactivity of a species relative to that of ozone. For nonreactive species $f0$ is 0, for slightly reactive it is 0.1 and for highly reactive it is 1.

For the dry deposition of particulates, rho specifies the density of the substance, $dquer$ its mean diameter $\overline{d_p}$, and $dsig$ the measure of variation σ_p.

Radioactive decay is switched off by specifying a negative half life, wet deposition is switched off by specifying negative A, dry deposition of gases is switched off by negative D, dry deposition of particles is switched off by negative rho.

```
***************************************************************************
*                                                                         *
*     Input file for the Lagrangian particle dispersion model FLEXPART    *
*               Definition file of chemical species/radionuclides         *
*                                                                         *
***************************************************************************
           Radioactivity    Wet depo        Dry depo (gases)      Dry depo (particles)  Dry depo
     SPECIES HALF LIFE [s]     A       B        D       H      f0       rho     dquer    dsig       vd
   1 TRACER          -999.9   -9.9E-09        -9.9                   -9.9E09                      -9.99
   2 O3              -999.9   -9.9E-09         1.6  1.0e-02  1.0    -9.9E09                      -9.99
   3 NO              -999.9    8.0E-06  0.62   1.3  2.0e-03  0.0    -9.9E09                      -9.99
   4 NO2             -999.9    1.0E-05  0.62   1.6  1.0e-02  0.1    -9.9E09                      -9.99
   5 HNO3            -999.9    8.0E-04  0.62   1.9  1.0e+14  0.0    -9.9E09                      -9.99
   6 HNO2            -999.9   -9.9E-09         1.6  1.0e+05  0.1    -9.9E09                      -9.99
   7 H2O2            -999.9    1.0E-04  0.62   1.4  1.0e+05  1.0    -9.9E09                      -9.99
   8 SO2             -999.9   -9.9E-09  0.62   1.9  1.0e+05  0.0    -9.9E09                      -9.99
   9 HCHO            -999.9   -9.9E-09         1.3  6.0e+03  0.0    -9.9E09                      -9.99
  10 PAN             -999.9   -9.9E-09         2.6  3.6e+00  0.1    -9.9E09                      -9.99
  11 NH3             -999.9   -9.9E-09         1.0  2.0e+14  0.0    -9.9E09                      -9.99
  12 SO4-aero        -999.9    1.0E-04  0.80  -9.9                   2.0E03  4.0E-7  3.0E-1    -9.99
  13 NO3-aero        -999.9    1.0E-04  0.80  -9.9                   2.0E03  4.0E-7  3.0E-1    -9.99
  14 I2-131        691200.0    8.0E-05  0.62   2.7  1.0e+05  0.1    -9.9E09                      -9.99
  15 I-131         691200.0    1.0E-04  0.80  -9.9                   2.5E03  6.0E-7  3.0E-1    -9.99
  16 Cs-137          -999.9    1.0E-04  0.80  -9.9                   2.5E03  6.0E-7  3.0E-1    -9.99
  17 Y-91         5037120.0    1.0E-04  0.80  -9.9                   2.5E03  6.0E-7  3.0E-1    -9.99
  18 Ru-106      31536000.0    1.0E-04  0.80  -9.9                   2.5E03  6.0E-7  3.0E-1    -9.99
  19 Kr-85           -999.9   -9.9E-09        -9.9                   -9.9E09                      -9.99
  20 Sr-90           -999.9    1.0E-04  0.80  -9.9                   2.5E03  6.0E-7
```

`landuse.t` is a binary file that contains the landuse inventory. `surfdata.t` gives the roughness lengths for each landuse class:

```
8 landuse categories are related with Leaf Area Index and roughness length
---------------------------------------------------------
landuse    LAI         z0(m)       Albedo     comment
---------------------------------------------------------
 1         0.50       0.10          0.18      Grassland for agricultural use
 2         2.00       0.15          0.10      Arable land
 3         3.00       0.30          0.18      Permanent crops
 4         7.00       0.60          0.15      Forest
 5         0.00       0.10          0.12      Inland water
 6         0.20       0.70          0.20      Urban areas
 7         1.00       0.10          0.15      Other
 8         0.00       0.10          0.12      Ocean
```

`surfdepo.t` gives the resistances needed for the parameterization of dry deposition of gases for the eight landuse classes and five seasonal categories. This file must not be changed by the user.

```
===============================================================================
INPUT RESISTANCES (s/m) FOR THE COMPUTATION OF SURFACE RESISTANCES TO
DRY DEPOSITION
===============================================================================
AFTER WESELY, 1989
===============================================================================
1 to 8: Landuse types
===============================================================================
Values are tabulated for 5 seasonal categories:
1      Midsummer with lush vegetation
2      Autumn with unharvested cropland
3      Late autumn after frost, no snow
4      Winter, snow on ground and subfreezing
5      Transitional spring with partially green short annuals
===============================================================================
               1       2       3       4       5       6       7       8

-------------------------------------------------------------------------------
ri          120.     60.     65.     90.   9999.   9999.    150.   9999.     1
rlu        2000.   2000.   2000.   2000.   9999.   9999.   4000.   9999.
rac         100.    200.    365.   2000.      0.    100.    200.      0.
```

```
rgss        350.   150.   230.   500.      0.    400.   400.      0.
rgso        200.   150.   170.   200.   2000.    300.   200.   2000.
rcls       2000.  2000.  2000.  2000.   9999.   9999.  4000.   9999.
rclo       1000.  1000.  1000.  1000.   9999.   9999.  1000.   9999.
--------------------------------------------------------------------------------
ri         9999.  9999.  9999.   500.   9999.   9999.  9999.   9999.      2
rlu        9000.  9000.  9000.  5500.   9999.   9999.  9000.   9999.
rac         100.   150.   270.  1710.      0.    100.   140.      0.
rgss        350.   200.   285.   500.      0.    400.   400.      0.
rgso        200.   150.   170.   200.   2000.    300.   200.   2000.
rcls       9000.  9000.  9000.  3270.   9999.   9999.  9000.   9999.
rclo        400.   400.   400.   570.   9999.   9999.   400.   9999.
--------------------------------------------------------------------------------
ri         9999.  9999.  9999.   500.   9999.   9999.  9999.   9999.      3
rlu        9000.  9999.  9470.  5500.   9999.   9999.  9000.   9999.
rac         100.    10.    20.  1330.      0.    100.   120.      0.
rgss        350.   150.   230.   500.      0.    400.   400.      0.
rgso        200.   150.   170.   200.   2000.    300.   200.   2000.
rcls       9000.  9999.  9470.  4500.   9999.   9999.  9000.   9999.
rclo        400.  1000.   570.   570.   9999.   9999.   600.   9999.
--------------------------------------------------------------------------------
ri         9999.  9999.  9999.   800.   9999.   9999.  9999.   9999.      4
rlu        9999.  9999.  9999.  1200.   9999.   9999.  9000.   9999.
rac          10.    10.    20.  1330.      0.    100.    50.      0.
rgss        100.   100.   100.   100.      0.    100.    50.      0.
rgso       3500.  3500.  3500.  3500.   2000.    600.  3500.   2000.
rcls       9999.  9999.  9470.   390.   9999.   9999.  9000.   9999.
rclo       1000.  1000.   570.   632.   9999.   9999.   800.   9999.
--------------------------------------------------------------------------------
ri          240.   120.   130.   180.   9999.   9999.   300.   9999.      5
rlu        4000.  4000.  4000.  2670.   9999.   9999.  8000.   9999.
rac          80.    50.   100.  1500.      0.    100.   120.      0.
rgss        350.   150.   230.   500.      0.    500.   400.      0.
rgso        200.   150.   170.   200.   2000.    300.   200.   2000.
rcls       4000.  4000.  4000.  2670.   9999.   9999.  8000.   9999.
rclo        500.  1000.   670.   750.   9999.   9999.   800.   9999.
--------------------------------------------------------------------------------
```

A.3 Output files

The FLEXPART model produces two output files: `grid` which contains all gridded information (concentrations, dry and wet deposition) and `receptor` which contains the concentrations for the specified receptor points. Both files are binary. The file structure of both files is clear from the two subroutines `openoutput.f` and `gridout.f`. `openoutput.f` writes information such as grid specifications, species names, etc., at the start of the model run. `gridout.f` is called, when a field output is due. It writes the simulation time in seconds, the wet deposition field, the dry deposition field, the total deposition field (all 2-d) and the concentration field (3-d).

The concentrations and depositions are multiplied by 10^{12} for output. This means that if the released mass is given in kg, concentrations are given in ngm^{-3} and deposition in ngm^{-2}. If the release is given in TBq, then the output is in Bqm^{-3} and Bqm^{-2}, respectively.

Appendix B

Source code description

Each subroutine and function used by the model is described in detail in the source code of FLEXPART . Here, only a short listing and description of the subroutines and a flow chart are given.

```
SHORT DESCRIPTION OF THE SOURCE CODE FILES USED BY THE FLEXPART MODEL
----------------------------------------------------------------------
advance.f         advection of the trajectory using a first order scheme for one
assignland.f      assigns the fractions of 8 landuse classes to each ECMWF grid
                  point
calcpar.f         calculates boundary layer parameters: friction velocity,
                  convective velocity scale, boundary layer height, Obukhov length
caldate.f         compute calendar date from julian date
coordtrafo.f      transformation of release point coordinates from geografical
                  to grid coordinates
distanz.f         calculation of distance (km) between two points given in
                  geografical coordinates
drydepokernel.f   calculates gridded dry deposition field using a uniform kernel
                  of bandwidths dx, dy
erf.f             error function
ew.f              calculation of saturation vapor pressure at given temperature
                  synchronisation interval; uses mean wind and Langevin equations
                  for parameterizing turbulence and mesoscale wind fluctuations
fixparticles.f    determine heights, release times and masses of all particles
FLEXPART.f        main program, manages reading of run specifications, calls
                  timemanager to do the actual calculations
getfields.f       management of wind fields in memory
getrb.f           computation of quasilaminar sublayer resistance to dry
                  deposition of gases
getrc.f           calculation of surface resistance to dry deposition of gases
```

getvdep.f	calculation of dry deposition velocities
gridcheck.f	determines grid specifications (grid area, etc.) from GRIB file
gridconc.f	calculation of concentrations on a grid using volume sampling and at receptor points using kernel method
gridout.f	output of concentrations on grid and for receptor points
hanna.f	parameterization of the turbulent velocity variance and the respective timescales; Langevin equation for w'/sigw
hanna1.f	alternative model to hanna.f: Langevin equation for w' drift correction velocity after Wilson
hanna_short.f	same as hanna.f, but only for the vertical wind
includecom	include file containing all global variables
includeinterpol	include file containing variables used for interpolation
includehanna	include file used for turbulence parameterization
includepar	include file containing all globally used parameter statements
interpol_all.f	interpolation of all data needed by advance for two levels within the boundary layer and for the surface
interpol_uvrho.f	interpolation of u, v and rho for an additional level
interpol_wrhograd.f	interpolation of w and rhograd for an additional level
interpol_vdep.f	interpolation of the deposition velocity
interpol_wind.f	interpolation of wind data above the PBL
juldate.f	compute julian date from calendar date
levlin3interpol.f	linear interpolation of three fields to a point on a model layer and in time
numerical.f	modified interpolation procedures taken from Press et al.
obukhov.f	calculation of Obukhov length from surface heat flux
openoutput.f	opens the output files for gridded and receptor point output
part0.f	calculation of time independent factors for the dry deposition of particles
partdep.f	calculation of dry deposition velocities for particles
pbl_profile.f	calculation of surface stress and sensible heat flux using the profile method
psih.f	stability correction term for heat
psim.f	stability correction term for momentum
raerod.f	calculation of the aerodynamical resistance to dry deposition
random.f	subroutines for generation of normally distriubted random numbers
readavailable.f	read, which wind fields are available within the modelling period
readcommand.f	read user commands, e.g. start of release, duration of release
readdepo.f	read parameters needed for dry deposition of gases
readlanduse.f	read landuse inventory and the respective roughness lenghts
readoro.f	read ecmwf model orography and land sea mask
readoutgrid.f	read output grid specifications
readpaths.f	read pathnames, where input/output files are stored

readreceptors.f	read names and coordinates of receptor points; for receptor points, concentrations are calculated with the kernel method
readreleases.f	read release point specifications (e.g., coordinates, type of trajectories to be used, number of particles, mass, etc.)
readspecies.f	read physical and chemical properties of species
readwind.f	reads the ECMWF meteorological data fields; GRIB decoding
richardson.f	calculation of convective velocity scale and boundary layer height using critical Richardson number concept
scalev.f	computation of friction velocity from surface stress
sigma.f	calculation of standard deviation of a 1-D field
timemanager.f	time management of FLEXPART; determines when a integration time step of a trajectory is due; determines when a concentration calculation is due, etc, and calls the respective subprograms
verttransform.f	transformation of three-dimensional fields from eta coordinates to meter above ground coordinates
wetdepo.f	calculation of wet deposition
wetdepokernel.f	calculates gridded wet deposition field using a uniform kernel of bandwidths dx, dy
windalign.f	transformation of random velocities from along- and cross-wind components to Cartesian u and v components

```
*****************************************************************************
*            FLEXPART model structure for ECMWF wind fields.                *
*****************************************************************************

FLEXPART --> readpaths

         --> readcommand --> juldate

         --> readavailable --> juldate

         --> gridcheck

         --> readoutgrid

         --> readreceptors

         --> readspecies

         --> readoro

         --> readlanduse

         --> assignland

         --> readreleases --> part0

         --> readdepo

         --> coordtrafo

         --> fixparticles

         --> openoutput --> caldate

                        --> distanz

         --> gasdev1 --> ran3

         --> timemanager --> getfields --> %1

                         --> wetdepo --> levlin3interpol

                                     --> wetdepokernel
```

```
                        --> gridconc

                        --> gridout

                        --> advance --> ran3

                                    --> interpol_all --> sigma

                                    --> interpol_uvrho --> sigma

                                    --> interpol_wrhograd --> sigma

                                    --> hanna or hanna1

                                    --> hanna_short

                                    --> interpol_wind --> sigma

                                    --> windalign

                        --> drydepokernel

-----------------------------------------------------------------------------
%1

 --> readwind (--> pbl_profile --> psim)

 --> calcpar --> scalev --> ew

             --> obukhov

             --> richardson

             --> getvdep --> caldate

                         --> getrb

                         --> raerod --> psih

                         --> getrc

                         --> partdep
```

```
 --> verttransform
--------------------------------------------------------------------------------
```