The FLEXTRA Trajectory Model
Version 3.0

User Guide

ANDREAS STOHL

Lehrstuhl für Bioklimatologie und Immissionsforschung,

University of Munich

Am Hochanger 13, 85354 Freising, Germany

This manual is available from
http://www.forst.uni-muenchen.de/LST/METEOR/stohl/flextra.html

February 1, 1999

Contents

Chapter 1

Introduction

Trajectory models are important tools for studying transport phenomena in the atmosphere (Stohl, 1998). In the environmental sciences, they are often used to establish source-receptor relationships of air pollutants. There are numerous trajectory models described in the literature, but few offer the flexibility and broadness the user would like to have. We thus aimed at creating such a versatile tool, FLEXTRA. FLEXTRA is written strictly in FORTRAN 77 code, the only exception being the use of ”INCLUDE” statements.

FLEXTRA can be used to calculate different types of forward or backward trajectories, and has facilities to estimate the uncertainty of trajectories. It is specifically designed to compute long time sequences of trajectories for many receptor locations. FLEXTRA version 2.0 remained in use almost unchanged for several years now, and since the model got quite popular over the years, we decided to implement some improvements. This is also the first FLEXTRA manual in English language.

The new FLEXTRA version 3.0 has several additional attractive new features as compared to version 2.0:

- the possibility to use global wind fields,
- nesting capabilities,
- an option to compute many trajectories starting in a certain volume, such as needed for the coherent ensemble of trajectories (CET) method described by Wernli and Davies (1997),
- an option to define the starting time and starting location individually for each trajectory (as needed, for instance, to start trajectories along an aircraft flight leg)

- an improved method of adding random errors to the trajectory at each time step for assessing trajectory uncertainty.

FLEXTRA is currently based on model level data of the numerical weather prediction model of the European Centre for Medium-Range Weather Forecasts (ECMWF). Gridded data of this model are available at various horizontal resolutions, 31 vertical levels, and with a frequency of 3 hours. At the time of this writing, the ECMWF model is operational at T319 L31 resolution, but an extension to L50 is planned. Different FLEXTRA versions, not described here, are available for the German/Swiss numerical weather prediction model (Wotawa, personal communication), and for the CALMET diagnostic wind field model (Pechinger, personal communication). With some changes, the model may be based on any other type of suitable model output. Since the model internally uses grid coordinates, such changes can be easily accomplished.

In this manual, no validation of FLEXTRA is presented, but there exist some papers on this issue. FLEXTRA was validated through comparison of model results to independent data using wind fields of varying spatial and temporal resolution (Stohl *et al.,* 1995), the tracks of constant level balloons (Stohl *et al.,* 1997, Stohl and Koffi, 1998), gas balloons (Baumann and Stohl, 1997) and hot-air balloons (Baumann *et al.,* 1996), and using potential vorticity and specific humidity (Stohl and Seibert, 1998) as tracers. Several other papers were presented at conferences. See the FLEXTRA homepage for a more complete listing of publications.

There also exists a sister model of FLEXTRA, FLEXPART, which is driven by the same (somewhat extended) input data. FLEXPART is a Lagrangian particle dispersion model, which, in contrast to FLEXTRA, fully accounts for turbulent dispersion. See http://www.forst.uni-muenchen.de/LST/METEOR/stohl/flexpart.html for a description of this model.

Chapter 2

Input data and coordinate systems

FLEXTRA, as described here, is based on gridded meteorological fields (analyses or forecasts) from the numerical weather prediction model of the European Centre for Medium-Range Weather Forecasts (ECMWF, 1995) given in a latitude/longitude coordinate system. The data are assumed to be in GRIB format. GRIB decoding software is available, e.g., from ECMWF, NOAA or NCAR. FLEXTRA needs four three-dimensional fields: the two horizontal wind components u and v and the vertical one w, and temperature T. If the vertical wind is missing, three-dimensional trajectories are not available. Other trajectory types, such as isentropic trajectories may still be computed, though. Two additional two-dimensional fields are needed: topography z_{oro} and surface pressure p_s. An important restriction is that all data fields used within a model run must have the same domain size, resolution, number of levels, etc.

FLEXTRA expects the three-dimensional 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. These data extraction routines to be used at ECMWF have been prepared by G. Wotawa. The source code and an extensive documentation of these routines are available from:
`http://www.boku.ac.at/imp/envmet/ecmwf_extr.html`.

The data extracted with these routines, more than what is actually needed by FLEXTRA, can also serve as input to the FLEXPART particle dispersion model.

FLEXTRA automatically determines the wind field domain, resolution, etc., from the input wind fields, the only restriction being that the maximum field dimensions must not be exceeded. The fields can, but don't have to be, global. In FLEXTRA version 3.0, however, they have to be defined strictly on a latitude/longitude grid.

FLEXTRA may use several nested wind fields at the same time, in addition to one coarse-resolution mother grid. These wind fields can be nested within each other (i.e, form a hierarchy of nests), or can cover different regions of interest. Typically, the resolution of these wind fields will be quite different. For instance, we routinely use global fields with $2^\circ \times 2^\circ$ resolution, with a $1^\circ \times 1^\circ$ nest covering the whole of Europe and large parts of the Northern Atlantic Ocean, and a $0.4^\circ \times 0.4^\circ$ nest covering Central Europe. FLEXTRA treats nested grids differently from the mother grid. Thus, their field dimensions may be different from the mother grid.

Also the nested wind fields may differ from each other in their actual dimensions, but in contrast to the mother grid, they all occupy the same amount of storage space (i.e., they all have the same maximum field dimensions). If their actual dimensions are very different, computer memory is wasted because their FORTRAN field dimensions must be set to the maximum value.

Trajectories may freely move into or out of nests with higher resolution. For a given trajectory position, FLEXTRA automatically determines the highest nesting level (with the highest resolution) available for the trajectory position to interpolate the wind velocities. The internal grid coordinates accord to the coarse-resolution mother domain, whereas positions with respect to the nested domains are calculated only when needed.

At the start of a model run, FLEXTRA automatically checks whether the mother domain is global or not. If it is not global, trajectories are terminated when leaving it. In contrast, if it is global, cyclic boundary conditions are used, i.e., if a trajectory leaves the domain at the eastern boundary, it re-enters the domain at the western boundary (and vice versa). Special considerations have been given to the poles. Since, in a latitude/longitude grid, the wind direction at the poles is not defined, additional fields of the horizontal wind are created. They are defined on the same latitude/longitude grid, but contain the wind components transformed to a polar stereographic projection. For the trajectory calculations, north and south of some parallel of latitude (definable by the user via parameters `switchnorth` and `switchsouth`, e.g., $\pm 75^\circ$), FLEXTRA switches back and forth between the latitude/longitude grid and the polar stereographic one. That means, the latitude/longitude grid remains the reference grid, but for advecting the trajectory, the polar stereographic projection grid is used intermediately. For this, the wind is transformed to the polar stereographic coordinates, but is still given at their original positions on the latitude/longitude grid. * For the coordinate transformations, the FORTRAN routines described by Taylor (1997) are used.

The polar stereographic grid has priority over nested grids. Thus, whenever a trajectory is poleward of the user-defined parallel of latitude, the polar stereographic grid is used, no matter whether nested grids are available or not.

Important parameters, variables and fields related to the wind fields:

```
maxnests            maximum number of nested wind fields
nxmax, nymax        maximum field dimensions in x and y of the mother grid
nxmaxn, nymaxn      same as above, but for the nested grids
nuvzmax,nwzmax      maximum number of vertical levels for the u and v wind
                    components, and for the w component, respectively

xglobal             switch, that tells whether global fields (in longitude)
                    are available
sglobal, nglobal    switchs, that tell whether south (north) pole is within domain
nx, ny              actual field dimensions of the mother grid
nxn, nyn            actual field dimensions of the nested grids
dx, dy              grid distances (in degrees) of the mother grid
dxn, dyn            same as above, but for the nested grids
nuvz,nwz            actual number of vertical levels for the u and v wind
northpolemap, southpolemap     variables that define the p.s. projection
xlon0, ylat0        lower left long/lat of the mother grid
xlon0n, ylat0n      same as above, but for the nested grids
uu, vv              horizontal wind components (mother grid)
uupol, vvpol        horizontal wind components (mother grid) projected on p.s. grid
uun, vvn            horizontal wind components (nested grids)
ww                  vertical wind (mother grid)
wwn                 vertical wind (nested grids)
tt                  temperature (mother grid)
ttn                 temperature (nested grids)
ps                  surface pressure (mother grid)
psn                 surface pressure (nested grids)
oro                 orography (mother grid)
oron                orography (nested grids)
akm, bkm, akz, bkz  define the vertical discretization of the grid (the eta levels)
wftime              times of wind fields (in seconds relative to starting time)
wfname              filenames of the wind fields
```

*Actually, the wind is transformed for (almost) the whole grid immediately after having been read in.

`memtime`	`same as above, but only for the three wind fields kept in memory`
`memind`	`pointer shuffled in memory instead of the wind fields`

Chapter 3

Integration method, time step criteria and interpolation methods

FLEXTRA internally uses seconds as the relevant unit to advance time. Thus, the minimum time step is 1 second. For date/time transformation, extensive use is made of the Julian date, which is a (double precision) real variable. The time difference between two dates can easily be calculated by transforming both dates to Julian dates and subtracting these from each other. Julian date is given in days.

FLEXTRA advances time in steps of 1 second (or -1 second in backward mode) and checks at each time step for each trajectory whether an integration time step is necessary. Furthermore, it checks whether the calculation of a trajectory is completed, or whether a new trajectory is to be started.

3.1 Integration method

A trajectory is defined by the differential equation

$$\frac{d\mathbf{X}}{dt} = \dot{\mathbf{X}}[\mathbf{X}(t)] \ , \tag{3.1}$$

with t being time, $\mathbf{X}$ the position vector, $\dot{\mathbf{X}}$ the wind velocity vector and $\frac{d}{dt}$ the differential operator with respect to time. It is solved using the constant acceleration scheme

$$\mathbf{X}(t_1) \approx \mathbf{X}(t_0) + \frac{1}{2}(\Delta t)[\dot{\mathbf{X}}(t_0) + \dot{\mathbf{X}}(t_1)] \ , \tag{3.2}$$

where Δt is the time step, which is also known as Petterssen's (1940) scheme. Equation (3.2) is accurate to the second order. It has to be solved by iteration:

$$\mathbf{X}^1(t_1) \approx \mathbf{X}(t_0) + (\Delta t)\dot{\mathbf{X}}(t_0) \ ,$$

$$
\begin{aligned}
\mathbf{X}^2(t_1) &\approx \mathbf{X}(t_0) + \frac{1}{2}(\Delta t)[\dot{\mathbf{X}}(t_0) + \dot{\mathbf{X}}^1(t_1)] \ , \\
&\vdots \\
\mathbf{X}^i(t_1) &\approx \mathbf{X}(t_0) + \frac{1}{2}(\Delta t)[\dot{\mathbf{X}}(t_0) + \dot{\mathbf{X}}^{i-1}(t_1)] \ . \qquad (3.3)
\end{aligned}
$$

The superscripts indicate the number of iteration, and $\dot{\mathbf{X}}^i(t_1)$ is taken at position $\mathbf{X}^i(t_1)$. Since $\dot{\mathbf{X}}(t_1)$ is not *a priori* known, the scheme is initialized with the zero acceleration scheme

$$\mathbf{X}^1(t_1) \approx \mathbf{X}(t_0) + (\Delta t)\dot{\mathbf{X}}(t_0) \ . \qquad (3.4)$$

Given sufficiently short time steps (see below), the Petterssen scheme always converges (Seibert, 1993). Typically, only three iterations are needed for convergence. Convergence is assumed to be achieved when the trajectory positions between two subsequent iterations differ by less than 0.0001 grid units (parameters `deltahormax` and `deltavermax`).

The scheme after Pudykiewicz et al. (1985), available in previous FLEXTRA versions, is no longer supported in version 3.0.

3.2 Time step criteria

FLEXTRA uses flexible time steps which are determined according to the actual flow situation. To resolve also the smallest scales in the flow fields, the Courant-Friedrichs-Lewy (CFL) criterion $\Delta t < \frac{\Delta x_i}{CFL|v_i|}$, where Δx_i are the grid distances in the three space directions and v_i are the corresponding wind components, is applied.* CFL can be chosen by the user, but must be larger than 1. A reasonable value is, e.g., 5.

In addition to the CFL criterion, an equivalent criterion is applied according to the time resolution of the wind fields, $\Delta t < \frac{\Delta T}{CFLT}$, where ΔT is the time interval between two subsequent wind fields, and $CFLT$, selectable by the user, must be larger than 1. A reasonable value is, again, 5.

Because of the flexible time steps, the number of time steps needed to complete one trajectory calculation is not known *a priori*. Since FLEXTRA always keeps a trajectory in its memory until its calculation is completed, the user must take care that enough storage space is assigned to the appropriate variables. The relevant parameter is `maxtime`, defined in the include file `includepar`. Under exceptional circumstances, in order not to exceed these field dimensions, FLEXTRA allows longer time steps than determined by the above

*In contrast to Eulerian models, the CFL criterion is not needed for stability, but it ensures optimum use of the available wind information.

criteria. In this case, a warning message "Warning: necessary time step exceeds CFL criterion" is issued. If this does not happen too often, the message may be ignored. However, it indicates that `maxtime` is too small and should be increased.

3.3 Interpolation methods

Stohl *et al.* (1995) tested the accuracy of several interpolation methods. Two methods were found to yield the best results:

1. Bilinear horizontal interpolation, linear vertical and linear temporal interpolation, which is rather accurate and computationally efficient.

2. Bicubic horizontal, polynomial vertical and linear temporal interpolation is more accurate than the first method, but requires more computing time.

The user may select one of the above two methods. The second one is termed "ideal" in the FLEXTRA input files.

Chapter 4

Trajectory types and trajectory uncertainty

4.1 Trajectory types

FLEXTRA can calculate either forward or backward trajectories. It is not possible to calculate both forward *and* backward trajectories within a single model run. However, several different trajectory types* may be calculated within a single model run. If a trajectory (of any type) touches the lowest model level, it is continued at that level. Similarly, trajectories reaching the highest model level are continued there.

4.1.1 Three-dimensional trajectories

For three-dimensional trajectories all three wind components are used.

4.1.2 Iso-eta trajectories

For iso-eta trajectories, the vertical wind is set to zero. Thus, they follow the η (model) surfaces.

4.1.3 Mixing layer trajectories

Mixing layer trajectories are two-dimensional trajectories which are advected with the average (horizontal) wind from the ground to the "mixing height". The mixing height

*Trajectory types differ in their treatment of the vertical wind component and coordinate system.

is not computed by FLEXTRA, but must be set by the user and is constant along all trajectories.

4.1.4 Isobaric trajectories

Isobaric trajectories follow pressure surfaces. For this, the height in η coordinates of the required pressure level is determined at each time step (by interpolation), and the horizontal wind at that level is used to advect the trajectory.

4.1.5 Isentropic trajectories

Isentropic trajectories are advected on surfaces of constant potential temperature. Similarly to the isobaric trajectories, the wind is taken at the corresponding η level. Potential temperature is a sensible vertical coordinate only for stable conditions. Thus, if isentropic trajectories encounter unstable conditions, their height is ill-defined. FLEXTRA uses the mean wind of the unstable layer to advect the trajectory. For the next time step (when the atmosphere may have stabilized), the pressure-weighted average potential temperature of the unstable layer is used as the reference level. Thus, under these conditions, the potential temperature may change along a trajectory.

4.2 Trajectory uncertainty

4.2.1 Potential vorticity tracer

The accuracy of a trajectory is hardly known without comparing with tracers (Stohl, 1998). However, suitable tracers are usually not available. Isentropic potential vorticity is conserved under inviscid, adiabatic conditions. It is defined by

$$PV = -g\eta_\Theta \frac{\partial \Theta}{\partial p} \,, \tag{4.1}$$

with $\eta_\Theta = \zeta_\Theta + f$, ζ_Θ being the relative vorticity on an isentropic surface, and g the gravitational acceleration.

PV is computed along each trajectory and written to the output files along with the position information and potential temperature.[†] For upper-level trajectories, it may serve as an indicator for trajectory accuracy. However, since PV is not conserved perfectly, not

[†]Potential temperature and potential vorticity are always taken from the coarse-resolution mother grid.

even under ideal conditions, it is a very rough indicator for trajectory accuracy. Often, non-conservation of PV is simply due to diabatic effects or mixing processes.

4.2.2 Ensemble trajectories

Another method to estimate trajectory accuracy is the use of ensemble methods. They assess how initially small position errors (or small trajectory errors caused by, e.g., wind interpolation) amplify in a given flow situation (i.e. in regions of divergence for forward trajectories, or in regions of convergence for backward trajectories).

The computation of ensemble trajectories is based on two basic ideas. First, if a trajectory is started with some distance to another one, the growth of their separation in space gives some information about their uncertainty (Merrill, 1985). Thus, if one starts an ensemble of uncertainty trajectories around a reference trajectory, the spread of the ensemble is an indicator for the accuracy of the reference trajectory. Within FLEXTRA, it is possible to specify a circle (in grid coordinates) around the starting location of a reference trajectory along which uncertainty trajectories are automatically started (a reasonable radius would be 0.5 to 2 grid units).

Second, random errors may be added along each uncertainty trajectory. These random errors should be on an order typical for the errors of the wind used to advect the trajectory (for instance, caused by interpolation). It is assumed that the errors increase with wind speed; thus, they are given relative to the wind speed. Furthermore, the relative errors follow a Gaussian distribution with the standard deviation specified by the user (reasonable values are 0.02 to 0.2, with higher values for the vertical wind than for the horizontal one). From this distribution, errors are drawn with a random number generator. The errors in the individual wind components are assumed to be independent from each other.

It can also be expected that there is some "memory" of these errors, i.e., an error in the wind at one time step is not completely independent from the error occurring during the previous time step. The user may thus specify a time period for this "memory" (i.e, a decorrelation time scale) (reasonable values 0.5 to 2). This period is scaled by the interval of the wind fields, since it may be assumed that errors in the individual wind fields are (almost) independent. Furthermore, it may be assumed that the errors in neighboring grid cells are also (almost) independent from each other. Thus, the decorrelation timescale is scaled by the time it takes a trajectory to cross one grid cell (x_i/v_i), if this time is shorter than the interval between the wind fields.

FLEXTRA facilitates the computation of an ensemble of "uncertainty" trajectories in an easy way. The user must select

1. the number of uncertainty trajectories,
2. the radius of a circle around the reference trajectory along which the uncertainty trajectories are started,
3. the time constant at which random interpolation errors are relaxed,
4. the standard deviation of wind errors (relative to the interpolated wind).

Then FLEXTRA calculates the appropriate uncertainty trajectories. By setting either the radius or the standard deviations of the wind errors to zero, one can calculate ensemble trajectories based exclusively on one of the two methods described above. By setting the number of the uncertainty trajectories to zero, the ensemble is switched off.

4.3 The normal mode, the CET mode and the FLIGHT mode

Normally, FLEXTRA calculates trajectories (and respective uncertainty trajectories) starting at locations and at time intervals as specified by the user. In this way, long time sequences of trajectories, e.g. trajectories starting every 3 hours over a period of one month, may be calculated within a single model run. All trajectories of one series start from the same starting point (more than one starting point can be defined, though). Thus, since the trajectories overlap (in a temporal sense), several trajectories starting from the same location have to be kept in the computer memory at the same time. For instance, if trajectories are started every 3 hours and each individual trajectory is 96 hours long, 32 trajectories per starting point (and also per uncertainty trajectory starting point) occupy storage space at the same time. Only after 96 h, when a trajectory is finished and written to the output files, the storage space is readdressed to a new trajectory. [‡]

In the CET mode, trajectories are initialized on a uniform grid in a three-dimensional domain as specified by the user. Because many trajectories starting at the same time are required for this, the whole FLEXTRA trajectory storage space is assigned to a single CET. It is thus not possible to calculate time sequences of CETs. Also no uncertainty trajectories can be calculated in the CET mode.

[‡]If a trajectory is terminated before its nominal end, for instance because it leaves the computational domain, it is kept in memory until its nominal completion (in the above example, 96 hours). This is done in order to have the trajectories sorted (in the output files) according to their starting time.

In the FLIGHT mode, the starting location *and* starting time can be specified for each trajectory individually. For this, a complete list of all trajectories to be calculated is needed. This mode is useful to calculate, for instance, trajectories along the flight leg of an aircraft. It may also be helpful to start trajectories at the ending points of trajectories calculated with a different (e.g., finer-scale) model.

Important parameters, variables and fields related to the trajectories:

```
PARAMETERS:
maxpoint           maximum number of trajectory starting points
maxtra             maximum number of trajectories (>maxpoint)
maxtime            maximum number of (flexible) time steps along one trajectory
itermax            maximum number of iterations used for integration

GENERAL VARIABLES:
numpoint           actual number of starting points
numtra             actual number of trajectories (currently in memory)
ibdate, ibtime     beginning date and time of calculation
iedate, ietime     ending date and time of calculation
lentra             desired trajectory length in seconds
interv             interval in seconds at which trajectories are started
ideltas            duration of the whole model run (seconds)
itime              "clock" of FLEXTRA (i.e. time in seconds)
cfl, cflt          cfl criteria in space and time
ldirect            direction of trajectories (1 forward, -1 backward)
inpolkind          switch between two interpolation methods
modecet            switch between normal and CET mode
runcomment         comment identifying a model run
epsu, epsv, epsw   relative wind errors that are assumed to occur at each time step
relaxtime          time constant (in units of the wind field interval)
                   at which the above errors are relaxed

VARIABLES DIRECTLY CONNECTED TO TRAJECTORIES:
nttra              number of time steps that are already computed
npoint             identification of startpoint of trajectory
kind               index for kind of trajectories (e.g. isobaric, 3-d,..)
randerroru,randerrorv,  Random errors added to the trajectory during
randerrorw              the last time step
kindz              indicates the unit of the z coordinate (1: masl, 2: magl, 3: hPa)
xtra,ytra,ztra     spatial positions
htra,ptra          height in meters and in hPa
ittra              temporal position
pottem             potential temperature
```

`vorttra`	`potential vorticity`

Chapter 5

Input files

5.1 The pathnames file

A file `pathnames` is expected in the starting directory of FLEXTRA. It contains the information, where input files are stored, and to which directory the output is directed:

```
/home/atmos1/as/flextra3.0/options/
/home/atmos1/as/flextra3.0/output/
/daten2/ecfields/2.0/
/daten2/ecfields/2.0/AVAILABLE
/daten2/ecfields/1.0/
/daten2/ecfields/1.0/AVAILABLE
/daten2/ecfields/0.5/
/daten2/ecfields/0.5/AVAILABLE
============================================

Line 1: path where control files "COMMAND" and "STARTPOINTS" are available
Line 2: name of directory where output files are generated
Line 3: path where meteorological fields are available (mother grid)
Line 4: full filename of "AVAILABLE"-file (mother grid)
Subsequent lines:
Line 2n+3: path where meteorological fields are available (nested grid n)
Line 2n+4: full filename of "AVAILABLE"-file (nested grid n)

Line below last pathname must be:
============================================

The grids must be arranged such as that the coarse-scale nests
come before the fine-scale nests. Multiple nests of the same
nesting level are allowed. In that case, the order is arbitrary.
```

5.2 The AVAILABLE files

The `AVAILABLE` files contain lists of wind fields that are available for trajectory calculations. They indicate the times for which the wind fields are valid and the respective filenames. The additional information (i.e., SPECIFICATIONS) is ignored. An example:

```
DATE     TIME         FILENAME     SPECIFICATIONS
YYYYMMDD HHMISS
-------- ------      ----------      ----------
19980401 000000      EN98040100      ON DISC
19980401 060000      EN98040106      ON DISC
19980401 120000      EN98040112      ON DISC
19980401 180000      EN98040118      ON DISC
19980402 000000      EN98040200      ON DISC
19980402 060000      EN98040206      ON DISC
19980402 120000      EN98040212      ON DISC
19980402 180000      EN98040218      ON DISC
19980403 000000      EN98040300      ON DISC
19980403 060000      EN98040306      ON DISC
19980403 120000      EN98040312      ON DISC
19980403 180000      EN98040318      ON DISC
19980404 000000      EN98040400      ON DISC
19980404 060000      EN98040406      ON DISC
19980404 120000      EN98040412      ON DISC
19980404 180000      EN98040418      ON DISC
19980405 000000      EN98040500      ON DISC
19980405 060000      EN98040506      ON DISC
19980405 120000      EN98040512      ON DISC
19980405 180000      EN98040518      ON DISC
19980406 000000      EN98040600      ON DISC
19980406 060000      EN98040606      ON DISC
19980406 120000      EN98040612      ON DISC
19980406 180000      EN98040618      ON DISC
19980407 000000      EN98040700      ON DISC
19980407 060000      EN98040706      ON DISC
19980407 120000      EN98040712      ON DISC
19980407 180000      EN98040718      ON DISC
19980408 000000      EN98040800      ON DISC
19980408 060000      EN98040806      ON DISC
19980408 120000      EN98040812      ON DISC
19980408 180000      EN98040818      ON DISC
19980409 000000      EN98040900      ON DISC
19980409 060000      EN98040906      ON DISC
```

5.3 Files in directory windfields

The directory `windfields` (in the above listing of file `pathnames` this is the directory `/daten2/ecfields/2.0/`) contains the GRIB code files of the meteorological input fields as indicated in the `AVAILABLE` file. 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    EN98040218   EN98040500   EN98040706
a.out*       EN98040300   EN98040506   EN98040712
EN98040100   EN98040306   EN98040512   EN98040718
EN98040106   EN98040312   EN98040518   EN98040800
EN98040112   EN98040318   EN98040600   EN98040806
EN98040118   EN98040400   EN98040606   EN98040812
EN98040200   EN98040406   EN98040612   EN98040818
EN98040206   EN98040412   EN98040618   EN98040900
EN98040212   EN98040418   EN98040700   EN98040906
```

Only the EN-Files (i.e, the wind fields) are really necessary. However, in the above example, the `AVAILABLE` file is stored in the same directory. The directories containing the nested wind fields have the same structure. They must contain the same number of files (valid at the same time) as the mother grid, but the filenames can be different, as indicated in the respective `AVAILABLE` files.

Wind fields may be available at irregular intervals. However, if the interval between two wind fields is large (in the standard configuration more than 3 hours (10800 seconds), parameter `idiffnorm` in file `includepar`), a warning message is issued. If the interval is even larger (in the standard configuration more than 6 hours, parameter `idiffmax` in file `includepar`), all trajectories that are followed in the respective time interval are stopped.

5.4 Files in directory options

In the above pathnames file, the directory where the FLEXTRA command files are stored is `/home/atmos1/as/flextra.global/options/`. There must be at least two files: a file `COMMAND`, and a file `STARTPOINTS` (or a file `STARTCET` or a file `STARTFLIGHT`, depending on what is selected in `COMMAND`). For all these files, except for `STARTFLIGHT`, two format options are available. One is based on a form that the user has to fill in. This is the most convenient way for the inexperienced user. For the advanced user, a more compact format is possible. FLEXTRA automatically recognizes which file type is used. Examples for both file types are given below.

This is the form version of the `COMMAND` file:

```
********************************************************************************
*                                                                              *
*    Input file for the trajectory model FLEXTRA: Please select your options   *
*                                                                              *
********************************************************************************

1. __________________________________________________ 3X, A50
   Test run #1
   LABEL FOR THE MODEL RUN

2. __                 3X, I2
   -1
   DIRECTION          1 FORWARD, -1 BACKWARD TRAJECTORIES

3. _______            3X, I7
    960000
   HHHMISS            LENGTH OF AN INDIVIDUAL TRAJECTORY

4. ________ ______    3X, I8, 1X, I6
   19980402 180000
   YYYYMMDD HHMISS    BEGINNING DATE

5. ________ ______    3X, I8, 1X, I6
   19980405 120000
   YYYYMMDD HHMISS    ENDING DATE

6. _______            3X, I7
   0060000
   HHHMISS            TIME INTERVAL BETWEEN STARTING TIMES OF TRAJECTORIES
```

```
7. _  _____          3X, I1, 2X, I5
   2  03600
   i  SSSSS          i>0: INTERPOLATED OUTPUT OF TRAJECTORY EVERY SSSSS SECONDS

8. _____  ___.___  _.___  _.___  _.___  _.___ 3X, I5, 2X, F7.3 4(2X,F5.3)
   00000  000.500  2.000  0.080  0.080  0.200
   NUMBER            NUMBER, DISTANCE (GRID UNITS), TIME CONSTANT (WIND FIELD INTERVAL UNITS) AND INTERPOLATION ERRORS (IN U, V AND W) OF UNCERTAINTY TRAJECT.

9. _                 3X, I1
   1
   INTERPOLATION     1 = IDEAL INTERPOLATION   >1 = LINEAR INTERPOLATION

10. ---.--           4X, F6.4
      5.0
    CFL              TIMESTEP CRITERION HORIZONTAL AND VERTICAL

11. ---.--           4X, F6.4
      5.0
    CFLT             TIMESTEP CRITERION TIME GAP BETWEEN INPUT WIND FIELDS

12. -                4X, I1
    1
    MODE             1 NORMAL MODE, 2 CET MODE, 3 FLIGHT MODE
=================================================================
1. Comment to identify the current model run

2. Direction of trajectories (1 means forward trajectories, -1 backward)

3. Temporal lengths of the trajectories in the format HHHMISS, where HHH is
   HOUR, MI is MINUTE and SS is SECOND

4. Beginning date and time of trajectory calculation. 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. All times are in UTC.

5. Ending date and time of trajectory calculation. Same format as 4.

6. Time interval between two trajectory calculations. Same format as 3.

7. Options for the trajectory output:
   0 = original data in irregular time intervals
   1 = constant time intervals, interpolated output every SSSSS seconds
   2 = 0 plus 1
```

```
8. Six parameters have to be inputted. The first is the number of
   uncertainty trajectories. They are starting in a distance from
   the starting point of the reference trajectory as given by the
   second parameter (in grid units).
   Additionally, random errors may be added at each time step of the
   trajectory calculation. Using a Langevin equation, they are relaxed
   with a time constant (in units of the wind field interval, third
   parameter) specified by the user. These random errors are
   thought to reflect typical wind errors caused, for instance, by
   interpolation. The magnitude of these errors (in relative units,
   relative to the wind velocity) must be specified by the user for
   the three wind components u, v and w (last three parameters).

9. Kind of interpolation
   1  - horizontal interpolation bicubic
        vertical interpolation polynomial
        temporal interpolation linear
   >1 - horizontal interpolation bilinear
        vertical interpolation linear
        temporal interpolation linear

10.cfl criterion horizontal/vertical
   factor by which time step must be shorter than that determined
   from the CFL criterion, i.e.

               delta_t1=delta x/u/cfl
               delta_t2=delta y/v/cfl
               delta_t3=delta z/w/cfl

               delta_t(space) = min(delta_t1,delta_t2,delta_t3)

11. cfl criterion time
    factor by wich time is shorter than time interval of the wind
    fields

               delta_t(time) = delta_T(input wind)/cflt

     The time step used for trajectory calculation is the minimum of
     delta_t(space) and delta_t(time)

     cfl and cflt must not be less than 1!
```

```
12.  1 NORMAL mode -> read file STARTPOINTS and calculate a time
     series of trajectories starting all from the same starting
     points
     2 CET mode -> read file STARTCET and calculate trajectories
     starting uniformly spaced from a user-defined domain
     (for a single starting time)
     3 FLIGHT mode -> read file STARTFLIGHT and calculate
     trajectories starting neither uniformly spaced nor with
     constant time intervals (as needed, for instance, to start
     trajectories along an aircraft leg)
```

Alternatively, the compact version of the COMMAND file looks like this:

```
********************************************************************************
*                                                                              *
*    Input file for the trajectory model FLEXTRA: Please select your options   *
*                                                                              *
********************************************************************************

'Test run #1'                      LABEL FOR THE MODEL RUN
-1                                 DIRECTION (1 FORWARD, -1 BACKWARD TRAJECT.)
 960000           HHHMISS          LENGTH OF AN INDIVIDUAL TRAJECTORY
19980402 180000   YYYYMMDD HHMISS  BEGINNING DATE
19980405 120000   YYYYMMDD HHMISS  ENDING DATE
 060000           HHHMISS          TIME INTERVAL BETWEEN STARTING TIMES OF TRAJECTORIES
2 3600            i  SSSSS         i>0: INTERPOLATED OUTPUT OF TRAJECTORY EVERY SSSSS SECONDS
0  0.5 2.0 0.08 0.08 0.20          NUMBER, DISTANCE (GRID UNITS), TIME CONSTANT (WIND FIELD INTERVAL UNITS) AND INTERPOLATION ERRORS (IN U, V AND W) OF UNCERTAINTY TRAJECT.
1                 INTERPOLATION    1 = IDEAL INTERPOLATION   >1 = LINEAR INTERPOLATION
5.0               CFL              TIMESTEP CRITERION HORIZONTAL AND VERTICAL
5.0               CFLT             TIMESTEP CRITERION TIME GAP BETWEEN INPUT WIND FIELDS
1                 MODE             1 NORMAL MODE, 2 CET MODE, 3 FLIGHT MODE
=================================================================================
```

With the above options, the user makes a FLEXTRA run labeled "Test run #1", computing 96 hours backward trajectories starting between April 2, 18 UTC and April 5, 12 UTC, at intervals of 6 hours. * Furthermore, output files are generated for both the flexible trajectory time steps and for the constant time steps with position invormation given every 1 hour (3600 seconds). No uncertainty trajectories are calculated, ideal interpolation is used, and both CFL and CFLT are set to 5. In the last line, it is specified

*Actually, for backward trajectories, the whole computation is reversed in time, i.e. it starts on April 5, 12 UTC and proceeds toward April 2, 18 UTC.

that normal trajectories are to be calculated. Their starting points must be given in the file `STARTPOINTS`. If the CET mode is selected, a file `STARTCET` is expected instead, specifying the starting domain of the CET. In CET mode, trajectories are started only at a single time, specified by "BEGINNING TIME". Thus, for CETs, "ENDING DATE" and "TIME INTERVAL" are ignored.

The STARTPOINTS file contains the information on starting locations and trajectory types used. For specifying the vertical starting position of a trajectory, the user has three options: height in metres above sea level (m asl), height in metres above (model) ground level (m agl), and the pressure level (hPa). Each trajectory may be of a different type. The file may contain an arbitrarily long list of starting points, the number of starting points being only limited by the parameter `maxpoint`, as specified in the file `includepar`. If too many starting points are specified, FLEXTRA issues a message and stops. The STARTPOINTS file, in the form version, contains the following information:

```
**********************************************************************
*                                                                    *
*                 TRAJECTORY MODEL                                   *
*                 DEFINITION OF STARTING/ENDING POINTS               *
*                                                                    *
*  The first 7 characters of the comment are also used as filenames. *
*  Therefore, they cannot be blank and they must be different for    *
*  each starting point.                                              *
*                                                                    *
*  Kind of trajectory: 1 = 3 dimensional                             *
*                      2 = on model layers                           *
*                      3 = mixing layer                              *
*                      4 = isobaric                                  *
*                      5 = isentropic                                *
*                                                                    *
**********************************************************************
*                                                                    *
*  Unit of z coordinate: 1 = Meters above sea level                  *
*                        2 = Meters above ground                     *
*                        3 = Hectopascal                             *
*                                                                    *
*  For mixing layer trajectories (kind 3), the z coordinate must be  *
*  given in m.a.g.l. (option 2)                                      *
*                                                                    *
**********************************************************************
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  10.0
____.____                  f9.4  Longitude [DEG]

  48.0
____.____                  f9.4  Latitude [DEG]

 1
 _                         1X,I1 Kind of trajectory (see file header)
```

```
 1
 -                         1X,I1, Unit of z coordinate

 3000.0
-----.---                  f10.3 z-coordinate (see file header)

TEST1
----------------------------------------  character*40 comment
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 -15.0
----.----                  f9.4  Longitude [DEG]

  48.0
----.----                  f9.4  Latitude [DEG]

 5
 -                         1X,I1 Kind of trajectory (see file header)

 2
 -                         1X,I1, Unit of z coordinate

 2000.0
-----.---                  f10.3 z-coordinate (see file header)

TEST2
----------------------------------------  character*40 comment
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

This is the alternative compact version of the `STARTPOINTS` file:

```
**********************************************************************
*                                                                    *
*                 TRAJECTORY MODEL                                   *
*                 DEFINITION OF STARTING/ENDING POINTS               *
*                                                                    *
*  The first 7 characters of the comment are also used as filenames. *
*  Therefore, they cannot be blank and they must be different for    *
*  each starting point.                                              *
*                                                                    *
*  Kind of trajectory: 1 = 3 dimensional                             *
*                      2 = on model layers                           *
*                      3 = mixing layer                              *
*                      4 = isobaric                                  *
```

```
*                      5 = isentropic                              *
*                                                                  *
********************************************************************
*                                                                  *
*  Unit of z coordinate: 1 = Meters above sea level                *
*                        2 = Meters above ground                   *
*                        3 = Hectopascal                           *
*                                                                  *
*  For mixing layer trajectories (kind 3), the z coordinate must be  *
*  given in m.a.g.l. (option 2)                                    *
*                                                                  *
********************************************************************
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  10.0                    F Longitude [DEG]
  48.0                    F Latitude [DEG]
1                         I Kind of trajectory (see file header)
1                         I Unit of z coordinate
3000.0                    F z-coordinate (see file header)
'TEST1'                   A Name of starting point
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 -15.0                    F Longitude [DEG]
  48.0                    F Latitude [DEG]
5                         I Kind of trajectory (see file header)
2                         I Unit of z coordinate
2000.0                    F z-coordinate (see file header)
'TEST2'                   A Name of starting point
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

In both of the above files, two starting locations are specified, at (10 °E, 48 °N) and at (15 °W, 48 °N). From the first location, three-dimensional trajectories are starting at a height of 3000 m asl. From the second location, isentropic trajectories are starting at 2000 m agl. The names of the starting locations are TEST1 and TEST2. These names are also used for the filenames of the output files.

If the CET mode is selected in the `COMMAND` file, a file `STARTCET` must be in the `options` directory instead of the `STARTPOINTS` file. This file defines the starting domain by its lower left and upper right coordinates, and the resolution of the starting grid superimposed on this domain. All trajectories of a CET are of the same type. This is the form version of `STARTCET`:

```
**********************************************************************
*                                                                    *
*                 TRAJECTORY MODEL                                   *
*                 DEFINITION OF THE CET DOMAIN                       *
*  A CET STARTING DOMAIN IS DEFINED BY THE LOWER LEFT AND UPPER RIGHT*
*  CORNER IN A LATITUDE/LONGITUDE COORDINATE SYSTEM, AND BY A LOWER  *
*  AND UPPER LEVEL. TRAJECTORIES ARE STARTED AT DISTANCES DX, DY AND *
*  DZ WITHIN THIS DOMAIN.                                            *
*                                                                    *
*  Kind of trajectory: 1 = 3 dimensional                             *
*                      2 = on model layers                           *
*                      3 = not allowed in CET mode                   *
*                      4 = isobaric                                  *
*                      5 = isentropic                                *
*                                                                    *
**********************************************************************
*                                                                    *
*  Unit of z coordinate: 1 = Meters above sea level                  *
*                        2 = Meters above ground                     *
*                        3 = Hectopascal                             *
*                                                                    *
*  The vertical distance DZ between the trajectories must be         *
*  given in the same units.                                          *
*                                                                    *
**********************************************************************
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  10.0
____.____                f9.4  Lower left longitude [DEG]

  48.0
____.____                f9.4  Lower left latitude [DEG]

  15.0
____.____                f9.4  Upper right longitude [DEG]

  50.0
____.____                f9.4  Upper right latitude [DEG]
```

```
   0.5
----.----                f9.4  DX: Longitudinal distance between trajectories [DEG]

   0.5
----.----                f9.4  DY: Latitudinal distance between trajectories [DEG]

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

 1
 -                       1X,I1, Unit of z coordinate

 2000.0
-----.---                f10.3 Lower z-coordinate (see file header)

 3000.0
-----.---                f10.3 Upper z-coordinate (see file header)

  500.0
-----.---                f10.3 DZ: Vertical interval of trajectories (see file header)

TEST
----------------------------------------   character*40 comment
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

And this is the compact version of `STARTCET`:

```
**********************************************************************
*                                                                    *
*                 TRAJECTORY MODEL                                   *
*                 DEFINITION OF THE CET DOMAIN                       *
*  A CET STARTING DOMAIN IS DEFINED BY THE LOWER LEFT AND UPPER RIGHT*
*  CORNER IN A LATITUDE/LONGITUDE COORDINATE SYSTEM, AND BY A LOWER  *
*  AND UPPER LEVEL. TRAJECTORIES ARE STARTED AT DISTANCES DX, DY AND *
*  DZ WITHIN THIS DOMAIN.                                            *
*                                                                    *
*  Kind of trajectory: 1 = 3 dimensional                             *
*                      2 = on model layers                           *
*                      3 = not allowed in CET mode                   *
*                      4 = isobaric                                  *
*                      5 = isentropic                                *
*                                                                    *
**********************************************************************
```

```
*                                                                     *
*  Unit of z coordinate: 1 = Meters above sea level                  *
*                        2 = Meters above ground                      *
*                        3 = Hectopascal                              *
*                                                                     *
*  The vertical distance DZ between the trajectories must be         *
*  given in the same units.                                           *
*                                                                     *
***********************************************************************
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  10.0                     F Lower left longitude [DEG]
  48.0                     F Lower left latitude [DEG]
  15.0                     F Upper right longitude [DEG]
  50.0                     F Upper right latitude [DEG]
   0.5                     F DX: Longitudinal distance between trajectories [DEG]
   0.5                     F DY: Latitudinal distance between trajectories [DEG]
 1                         I Kind of trajectory (see file header)
 1                         I Unit of z coordinate
 2000.0                    F Lower z-coordinate (see file header)
 3000.0                    F Upper z-coordinate (see file header)
  500.0                    F DZ: Vertical interval of trajectories (see file header)
'TEST'
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

If the FLIGHT mode is selected in the `COMMAND` file, a file `STARTFLIGHT` must be in the `options` directory instead of the `STARTPOINTS` file. This file provides a list of trajectory starting points, to be strictly in temporal order (reverse temporal order for backward trajectories). The trajectory type and the unit of the vertical coordinate is the same for all trajectories, but both starting times *and* coordinates may be different for all trajectories. Her is an example of this file:

```
**********************************************************************
*              TRAJECTORY MODEL                                      *
*              DEFINITION OF STARTING/ENDING POINTS IN FLIGHT MODE   *
*  This file defines starting points separated non-uniformly in      *
*  space as well as in time. Thus, both starting times AND starting  *
*  coordinates must be given.                                        *
*  The starting times must be strictly in temporal order.            *
*  For backward trajectories, the temporal order must be reversed.   *
*  In line #28 of this file, the name of the output file must be     *
*  indicated. Lines #29 and #30 must contain kind of trajectory and  *
*  the unit of the z coordinate to be used. Line #31 is arbitrary,   *
*  then follows a sequence of points.                                *
*  Kind of trajectory: 1 = 3 dimensional                             *
*                      2 = on model layers                           *
*                      3 = mixing layer                              *
*                      4 = isobaric                                  *
*                      5 = isentropic                                *
**********************************************************************
*                                                                    *
*  Unit of z coordinate: 1 = Meters above sea level                  *
*                        2 = Meters above ground                     *
*                        3 = Hectopascal                             *
*                                                                    *
*  For mixing layer trajectories (kind 3), the z coordinate must be  *
*  given in m.a.g.l. (option 2)                                      *
*                                                                    *
**********************************************************************
'Flight#7'
1                          I Kind of trajectory (see file header)
1                          I Unit of z coordinate
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
19980404 000000           2I Starting date and time
  34.0                     F Longitude [DEG]
  72.0                     F Latitude [DEG]
1000.0                     F z-coordinate (see file header)
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
19980404 140000           2I Starting date and time
```

```
 -15.0                    F Longitude [DEG]
  48.0                    F Latitude [DEG]
2000.0                    F z-coordinate (see file header)
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
19980405 130000          2I Starting date and time
  10.0                    F Longitude [DEG]
  48.0                    F Latitude [DEG]
3500.0                    F z-coordinate (see file header)
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
19980406 130000          2I Starting date and time
  10.0                    F Longitude [DEG]
  48.0                    F Latitude [DEG]
3000.0                    F z-coordinate (see file header)
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

5.5 FLEXTRA output files

The whole FLEXTRA output is directed to a single output directory. In the above `pathnames` specificiations, this is `/home/atmos1/as/flextra3.0/output/`. FLEXTRA may produce six different types of output files:

1. In the normal mode, time sequences of trajectories are generated for each starting point. The results for each starting point (i.e. a time sequence of trajectories) are stored in an individual file. The filenames are `T_name` (for flexible time steps) and `TI_name` (for constant time steps), where `name` are the first 12 characters of the starting point comment.

2. CET files which contain all trajectories of a given CET (with flexible or with constant time step), all starting at the same time. The names of these files are `CET_name` (for flexible time steps) and `CETI_name` (for constant time steps), where `name` are the first 12 characters of the CET comment (in the above example this would be "TEST", and the filename for constant time step output `CETI_TEST`).

3. FLIGHT files which contain all trajectories along a given flight leg (either with flexible or with constant time steps). The names of these files are `FLIGHT_name` (for flexible time steps) and `FLIGHTI_name` (for constant time steps), where `name` are the first 12 characters of the FLIGHT comment (in the above example this would be "Flight#7", and the filename for constant time step output `FLIGHTI_Flight#7`).

An example `TI_name` file may look like this:

```
*********************************************************************
*                                                                   *
*                     FLEXTRA MODEL OUTPUT                          *
*                     FOR ECMWF WINDFIELDS                          *
*                                                                   *
*********************************************************************
*                                                                   *
*            TIME OF COMPUTATION:   19990114 14:22:23.2             *
*                                                                   *
*********************************************************************
*                                                                   *
*           TYPE OF TRAJECTORIES:  ISENTROPIC                       *
*                                                                   *
*********************************************************************
*                                                                   *
*             INTEGRATION SCHEME:  PETTERSSEN                       *
*            INTEPOLATION METHOD:  IDEAL                            *
*                                                                   *
*          SPATIAL CFL CRITERION:  5.00                             *
*         TEMPORAL CFL CRITERION:  5.00                             *
*                                                                   *
*********************************************************************
*                                                                   *
* START POINT COMMENT: TEST                                         *
* MODEL RUN COMMENT: Test run #1                                    *
*                                                                   *
*********************************************************************

DATE: 19980402    TIME:  180000    STOP INDEX: 1    # OF POINTS:    9
    SECS   LONGIT    LATIT    ETA  PRESS     Z Z-ORO     PV THETA
       0 -10.0000  48.0000  .7809  772.4 2001. 2000.   .556 288.4
  -10800 -13.8437  46.8852  .7797  770.1 2010. 2010.   .576 288.7
  -21600 -18.0137  46.1880  .7539  746.1 2271. 2271.   .514 288.5
  -32400 -22.2734  46.0866  .7454  738.7 2352. 2352.   .586 288.6
  -43200 -26.1761  46.3039  .7060  701.1 2760. 2760.   .668 288.8
  -54000 -29.4075  46.6924  .6882  685.6 2950. 2950.   .761 289.3
  -64800 -32.0994  47.0156  .6683  666.8 3166. 3166.   .745 289.4
  -75600 -34.7240  47.2701  .6693  668.1 3150. 3150.   .864 289.9
  -86400 -37.5549  47.5151  .6511  650.5 3338. 3337.  1.160 291.0
DATE: 19980403    TIME:  180000    STOP INDEX: 1    # OF POINTS:    9
    SECS   LONGIT    LATIT    ETA  PRESS     Z Z-ORO     PV THETA
       0 -10.0000  48.0000  .7804  765.3 2001. 2000.   .849 289.1
  -10800 -13.0518  48.4605  .7796  760.8 2004. 2003.   .837 288.9
```

```
 -21600 -15.2082  49.7889  .7599  741.2 2198. 2198.   .830 289.1
 -32400 -15.6836  51.3854  .7606  739.2 2186. 2187.   .856 288.6
 -43200 -14.3157  52.5602  .7614  738.5 2175. 2175.   .953 288.9
 -54000 -12.2525  52.9209  .7630  742.0 2160. 2159.   .991 289.1
 -64800 -10.8281  52.5667  .7489  729.8 2310. 2302.   .940 288.7
 -75600 -10.2955  51.5967  .7604  739.8 2221. 2190.   .847 288.3
 -86400 -11.2324  50.3553  .7322  716.9 2485. 2485.   .668 289.0
```

The file header gives the relevant information concerning trajectory type, interpolation method, comments, etc. The time of computation is only available if the system call in routine `openoutput.f` is enabled. Following the header is a time sequence of trajectories. In the first line, general information about the trajectory is given: starting date and time, an index which gives information why a trajectory was stopped (normally 1, 2 if the trajectory left the computation domain, 3 if time difference between two wind fields was too large, 4 if no wind fields were available) and the number of positions available. The second line contains sub-header information. This is followed by the required number of lines, each containing: the travel time of the trajectory in seconds (negative for backward trajectories), the longitude and latitude, and the vertical position given in η, pressure and Cartesian (in m asl and m agl) coordinates, PV and potential temperature.

The filenames containing uncertainty trajectories follow this syntax: `TI_name_Unumber`, where `number` is a four-digit number. Uncertainty trajectories are numbered consecutively. An example file name may be `TI_TEST1_U0001`. The same syntax is used for uncertainty trajectories with flexible time steps (e.g. `T_TEST1_U0001`). They contain the same information except that the positions are given at irregular intervals.

The structure of the `CET_name`, `CETI_name`, `FLIGHT_name` and `FLIGHTI_name` files is the same, except that the files contain the information of all starting points, but only for one starting time (for the CET files) or at varying intervals (for the FLIGHT files).

Chapter 6

Short source code description

Following is a complete listing of source code files with a short description of their content.

```
--------------------------------------------------------------------
caldate.f         compute calendar date from julian date
checklimits.f     Check, if the user specifications exceed dimensional settings
cmapf1.0.f        Program package for coordinate transformations between
                  latitude/longitude grid and polar stereographic projection
coordtrafo.f      transformation of trajectory starting points to grid coordinates
eta.ecmwf.f       computes eta coordinate for a given air pressure
etatrafo.f        transformation from z-coordinate in metres to eta coordinate
FLEXTRA.f         main program
geteta.f          calculates eta for a given potential temperature
getfields.f       management of windfields, calls readwind when needed
getheight.f       calculates height of trajectories in metres and hPa
getpv.f           calculates potential vorticity at trajectory position
getwind.f         according to trajectory type, calls the appropriate
                  interpolation routines and wind retrieval procedure
gridcheck.f       checks the wind field domain available and establishes
                  the computation grid
gridcheck_nests.f same as above, but for nested wind fields
includecom        includefile containing globally used variables
includepar        includefile containing globally used parameters: this
                  is were you can change field dimensions and other important parameters
inter3d.f         manages interpolation of wind data for 3d trajectories
interisentrop.f   manages interpolation of wind data for isentropic trajectories
interisobar.f     manages interpolation of wind data for isobaric trajectories
intermix.f        manages interpolation of wind data for mixing layer trajectories
intermod.f        manages interpolation of wind data for model layer trajectories
interpol.f        ideal interpolation of fields to a trajectory point in space and time
interpol_nests.f  same as above, but for nested wind fields
```

juldate.f	compute julian date from calendar date
lamphi.ecmwf.f	transformation from ECMWF coordinates to geographical coordinates
lastprocessor.f	reverses the trajectory output files for backward trajectories
levinterpol.f	ideal interpolation of fields to a point on a layer and in time
levinterpol_nests.f	same as above, but for nested wind fields
levlininterpol.f	linear interpolation of fields to a point on a layer and in time
levlininterpol_nests.f	same as above, but for nested wind fields
lininterpol.f	linear interpolation of fields to a trajectory point in space and time
lininterpol_nests.f	same as above, but for nested wind fields
numerical.f	modified interpolation procedures taken from Press et al.
opencetoutput.f	opens the output files for the CET output
openflightoutput.f	opens the output files for the CET output
openoutput.f	opens the output files for the trajectory output
orolininterpol.f	linear interpolation of the topography to a given point
orolininterpol_nests.f	same as above, but for nested wind fields
petters.f	petterssen scheme for trajectory integration
potlininterpol.f	linear interpolation of potential temperature to a point in space and time
pp.ecmwf.f	calculates the pressure as a function of eta
pvinterpol.f	Interpolation of potential vorticity on a small subgrid
readavailable.f	read, which wind fields are available within the modelling period
readcet.f	read CET definitions (domain, etc.) from file STARTCET
readcommand.f	read user commands, e.g. start times of trajectories
readflight.f	read FLIGHT definitions (i.e. a list of starting times AND starting locations) from file STARTFLIGHT
readoro.f	read ECMWF model orography from file OROGRAPHY
readpaths.f	read pathnames, where input/output files are stored
readpoints.f	read trajectory starting/ending points from file STARTPOINTS
readwind.f	reads the wind fields, temperature fields and surface pressure from data files
readwind_nests.f	same as above, but for nested wind fields
skplin.f	skips a number of lines in a given file
subtractoro.f	subtraction of ECMWF orography from trajectory starting/ending point height
timemanager.f	manages the temporal scheme of the computation and checks wich trajectories are to be held in memory
trajinterpol.f	interpolation of flexible trajectory time step to constant one
trajout.f	writes trajectories to output file
uncertcoor.f	determines the starting points for the uncertainty trajectories
utransform.f	transformation of u [m/s] to dx/dt [grid units/s]
vtransform.f	transformation of v [m/s] to dy/dt [grid units/s]
wtransform.f	transformation of w [m/s] to dz/dt [units/s]
zztrafo.f	calculation of geometric height above model orography

```
as a function of eta, temperature and pressure
```

This is a flux diagram of the source code:

```
****************************************************************************
*                    FLEXTRA model structure                               *
****************************************************************************

FLEXTRA  --> readpaths

         --> readcommand --> skplin
                         --> juldate

         --> readavailable --> juldate

         --> gridcheck --> GRIB DECODING CALLS
                       --> POLAR STEREOGRAPHIC PROJECTION CALLS

         --> gridcheck_nests --> GRIB DECODING CALLS

         (--> readoro)

         --> readpoints --> skplin
         --> readcet --> skplin
         --> readflight --> skplin

         --> checklimits

         --> coordtrafo

         --> uncertcoor

         --> subtractoro --> orolininterpol
                         --> orolininterpoln

         --> openoutput --> (date and time system call)
         --> opencetoutput --> (date and time system call)
         --> openflightoutput --> (date and time system call)

         --> timemanager --> trajout --> orolininterpol

                                     --> orolininterpoln

                                     --> lamphi_ecmwf

                                     --> trajinterpol
```

```
                              --> caldate

                  --> coordtrafo

                  --> subtractoro

                  --> petters --> getwind --> getfields --> readwind

                                                        --> readwind_nests

                                          --> inter3d      |
                                                           |
                                          --> intermod     |
                                                           |
                                          --> intermix     |
                                                           |
                                          --> interisobar  |
                                                           |
                                          --> interisentrop|

                              --> utransform

                              --> vtransform

                              --> POLAR STEREOGRAPHIC PROJECTION CALLS

                              --> gasdev --> ran1

                              --> getheight --> levlininterpol
                                            --> levlininterpoln

                                            --> pp

                                            --> zztrafo -->levlininterpoln
                                                        -->levlininterpol
                              --> getpv  --> pvinterpol

        --> lastprocessor

------------------------------------------------------------------------------
```

```
| --> levllininterpoln
| --> levllininterpol
|
| --> etatrafo --> levllininterpol
|
| --> pp
|
| --> eta
|
| --> potllininterpoln
| --> potllininterpol
|
| --> geteta --> levllininterpoln
|           --> levllininterpol
|
| --> interpol --> bicubic --> bcucof
|              --> polynom
|
|     lininterpol
|     levinterpol --> bicubic --> bcucof
|                 --> polynom
|     levllininterpol
|     interpoln --> bicubic --> bcucof
|               --> polynom
|     lininterpoln
|     levinterpoln --> bicubic --> bcucof
|                  --> polynom
|     levllininterpoln
|
| --> wtransform (only inter3d)

------------------------------------------------------------------------------
```

Chapter 7

References

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

Baumann, K., Langer, M., and Stohl, A. (1996) Hot-air balloon tracks used to analyze air flow in alpine valleys. Proc. 24th Conf. on Alpine Meteorology, Bled, Slovenia, 60-66.

Merrill, J.T., Bleck, R. and Avila, L. (1985) Modeling atm ospheric transport to the Marshall islands. *J. Geophys. Res.***90**, 12927 -12936.

Petterssen S. (1940) *Weather Analysis and Forecasting.* McGraw-Hill Book Company, New York. pp. 221-223.

Pudykiewicz, J., R. Benoit, A. Staniforth (1985) Preliminary results from a partial LR-TAP model based on an existing meteorological forecast model. *Atmos.-Ocean* **23**, 257-303.

Seibert, P. (1993) Convergence and accuracy of numerical methods for trajectory calculations. *J. Appl. Meteor.***32**, 558-566.

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

Stohl, A., Baumann, K., Wotawa, G., Langer, M., Neininger, B., Piringer, M., and Formayer, H. (1997) Diagnostic downscaling of large scale wind fields to compute local scale trajectories. *J. Appl. Meteor.***36**, 931-942.

Stohl, A., and Koffi, N.E. (1998) Evaluation of trajectories calculated from ECMWF data against constant volume balloon flights during ETEX. *Atmos. Environ.***24**, 4151-4156.

Stohl, A., and P. Seibert (1998) Accuracy of trajectories as determined from the conservation of meteorological tracers. *Q. J. R. Meteorol. Soc.***125**, 1465-1484.

Stohl, A., Wotawa, G., Seibert, P., and Kromp-Kolb, H. (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.

Taylor, A.D. (1997) Conformal map transformations for meteorological modelers. *Computers & Geosciences* **23**, 63-75.

Wernli, H. and Davies, H.C. (1997) A Lagrangian-based anal ysis of extratropical cyclones. I: The method and some applicati ons. *Q. J. R. Meteorol. Soc.***123**, 467-489.