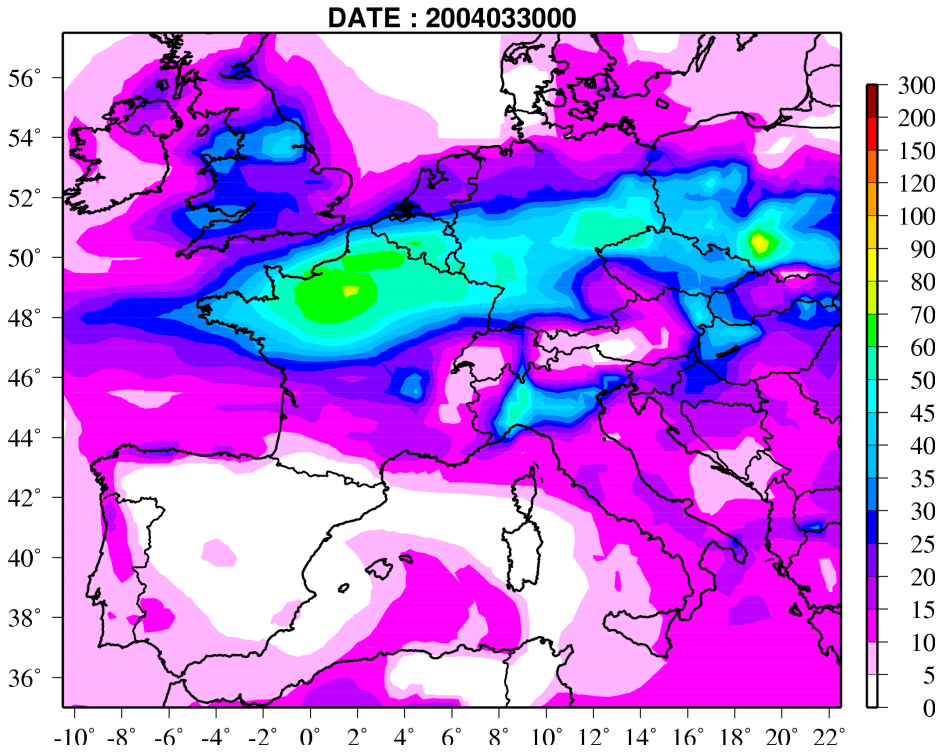
Documentation of the chemistry-transport model

Chimere

[version V200511B]



Surface PM10 concentration for the March 30, 2004

- This documentation and the model are freely available at the following internet adresse:

http://euler.lmd.polytechnique.fr/chimere/

- CHIMERE is distributed under the GNU General Public License
- Copyright (C) 2005 Institut Pierre-Simon Laplace (C.N.R.S.), INERIS, LISA (C.N.R.S.)
- For questions, send an e-mail to chimere@lmd.polytechnique.fr
- *Last update of this documentation: November 24, 2005*

Contents

1 Short Description of the model **9**
- 1.1 Purpose . . . 9
- 1.2 Input data required . . . 9
- 1.3 OS, performance and software . . . 10
- 1.4 The physics and numerics in short . . . 10
- 1.5 What can be done with . . . 11

2 Last changes: the version V200511B **13**

3 Model scientific documentation **15**
- 3.1 Main options . . . 15
- 3.2 Model geometry . . . 15
  - 3.2.1 Horizontal grid . . . 15
  - 3.2.2 Vertical grid . . . 16
  - 3.2.3 Grid nesting . . . 17
- 3.3 Transport and mixing . . . 17
  - 3.3.1 Horizontal transport . . . 17
  - 3.3.2 Vertical transport . . . 18
  - 3.3.3 Turbulent mixing . . . 18
- 3.4 Aerosol processes . . . 19
- 3.5 Land use and related parameters . . . 19
- 3.6 Meteorological input data . . . 19
- 3.7 Boundary concentrations . . . 20
- 3.8 Initial concentrations . . . 21
- 3.9 Chemistry . . . 22
  - 3.9.1 Chemical mechanisms (gas phase) . . . 22
  - 3.9.2 Species list of the reduced MELCHIOR2 gas-phase chemical mechanism . . . 22
  - 3.9.3 Aqueous-phase chemistry . . . 27
  - 3.9.4 Heterogeneous chemistry . . . 28
  - 3.9.5 Secondary organic aerosol chemistry . . . 28
- 3.10 Clouds and impact on photolysis . . . 29
- 3.11 Emissions . . . 30
  - 3.11.1 Anthropic emissions . . . 30
  - 3.11.2 Biogenic emissions . . . 31
- 3.12 Dry deposition . . . 31
  - 3.12.1 For gases : . . . 31
  - 3.12.2 For aerosols: . . . 32
- 3.13 Wet scavenging . . . 32
- 3.14 Time numerical solver . . . 33

Technical documentation **34**

4 First CHIMERE simulation **35**
- 4.1 Main sources installation . . . 35

4.1.1 Software prerequisites . . . 35
4.2 Start CHIMERE with the 2003 HEAT-WAVE test case . . . 36
4.2.1 Download required data and programs . . . 36
4.2.2 Simulation configuration . . . 37
4.3 Outputs . . . 38
4.3.1 The output on screen . . . 39
4.3.2 The output files . . . 40
4.4 Post-processing the results . . . 41
4.4.1 Dump data in ASCII form . . . 42
4.4.2 Use data for further numerical processing . . . 42
4.4.3 Plot data . . . 42
4.4.4 Interactively display data . . . 43
4.5 Download the complete available CHIMERE database . . . 45
4.5.1 download emissions data . . . 45
4.5.2 download prepdomains interface . . . 45
4.5.3 download EMEP emissions interface . . . 46

5 The CHIMERE model 47
5.1 Directories . . . 48
5.2 Model Files . . . 49
5.2.1 Scripts . . . 49
5.2.2 Meteo interface - 1st stage . . . 49
5.2.3 Main programs . . . 51
5.2.4 Subroutines . . . 51
5.2.5 Header modules . . . 51
5.2.6 Makefiles . . . 52
5.2.7 Files used for land processes (deposition+emissions) . . . 52
5.2.8 Static files for chemistry and model information used for running the model . . . 52
5.2.9 Dynamic files generated at runtime . . . 52
5.3 Routine calls sequence . . . 54
5.4 Core model input files . . . 55
5.5 Core model output files . . . 56
5.6 Running real cases . . . 57

6 Make your own CHIMERE configuration 59
6.1 Meteorological interface . . . 59
6.1.1 First stage: the "interf" programs . . . 59
6.1.2 Second stage: the diagmet diagnostic model . . . 64
6.2 Model domains and landuse . . . 65
6.2.1 Generalities about the domain files . . . 66
6.2.2 Use the predefined domain files . . . 67
6.2.3 Built new domain files inside Europe: prepdomain . . . 67
6.2.4 Built new domain files outside of Europe . . . 69
6.3 Anthropic emission interface . . . 69
6.4 Biogenic emission interface . . . 71
6.5 Boundary conditions interface . . . 71
6.6 Initial Conditions . . . 72

Appendix 72

A References 73
A.1 Publications using CHIMERE . . . 73
A.2 Useful links . . . 76

B How To install netCDF under GNU/Linux 81
B.1 Background . . . 81
B.2 Download . . . 81
B.3 Configure netCDF . . . 81
B.3.1 ifort . . . 82
B.3.2 g95 . . . 82
B.3.3 pgf90 . . . 83
B.3.4 ifort 64 bit on Intel EMT64 or AMD Opteron . . . 83
B.3.5 g95 64 bit on Intel EMT64 or AMD Opteron . . . 83
B.4 Configure CHIMERE . . . 84

C How To install Udunits under GNU/Linux 85
C.1 Background . . . 85
C.2 Download . . . 85
C.3 Configure Udunits . . . 85
C.3.1 gcc on a 32 bit machine . . . 86
C.3.2 gcc on a 64 bit machine . . . 86
C.4 configure CHIMERE . . . 86

D The color palettes 87

E NetCDF Conventions 89

F Structure of the CHIMERE netCDF files 91
F.1 exdomout.nc . . . 91
F.2 METEO.nc . . . 93
F.3 AEMISSIONS.nc . . . 94
F.4 BEMISSIONS.nc . . . 95
F.5 LAT_CONCS.nc . . . 95
F.6 TOP_CONCS.nc . . . 96
F.7 end.nc . . . 96

G The chemical preprocessor: CHEMPREP 103
G.1 Conditions on the mechanism . . . 103
G.2 Syntax of the REACTIONS file . . . 105
G.3 Syntax of the PHOTORATES file . . . 105
G.4 Files produced . . . 106

H The top-calling script chimere-mm5 107

I Definition of new CHIMERE domains 111
I.1 Horizontal domain: the COORD_[dom] file . . . 111
I.2 Vertical domain: the VCOORD file . . . 112

Introduction

This documentation describes the last version of the CHIMERE chemistry-transport model named **V200511B**. The model and associated source codes are freely available on a web site:

http://euler.lmd.polytechnique.fr/chimere/

- The chapter 1 presents a very short model description, including an overview of the model capabilities, the required input data, operating system and softwares.
- The chapter 2 (p.13) presents an overview of the last changes in this model version compared to the previous ones.
- The chapter 3 (p.15) presents the scientific model documentation, with the parameterizations developed and used, both for chemistry and physics.
- The chapter 4 explains how install the model from scratch and perform a first run using the prepared test case 'Heat Wave summer 2003 over Europe'.
- The chapters 5 and 6 (p.47 and p.59) present the model sources documentation including the pre- and post-processing interfaces.
- The Appendix present "how-to" files explaining the way to install informatic librairies such as netCDF and Udunits, and the format of all files used by the model.
- A list of published papers about studies using CHIMERE is displayed in the Chapter A, p.73.

For further questions, please send an e-mail to the developers at chimere@lmd.polytechnique.fr.

Chapter 1

Short Description of the model

1.1 Purpose

The CHIMERE multi-scale model is primarily designed to produce daily forecasts of ozone, aerosols and other pollutants and make long-term simulations for emission control scenarios. CHIMERE runs over a range of spatial scales from the regional scale (several thousand kilometers) to the urban scale (100-200 Km) with resolutions from 1-2 Km to 100 Km. On CHIMERE server, documentation and source codes are proposed for the complete multi-scale model. However most data are valid only for Europe and should be revisited for applications on other continents.
CHIMERE proposes many different options for simulations which make it also a powerful research tool for testing parameterizations, hypotheses. Its use is relatively simple so long as input data is correctly provided. It can run with several vertical resolutions, and with a wide range of complexity. It can run with several chemical mechanisms, simplified or more complete, with or without aerosols.
The **Figure 1.1** shows the example of the ozone simulation for August 8th, 2003 at 14h UT, during the exceptional heat wave that occured over Western Europe. The gray dots show monitoring stations where the 180 μg.m^{-3} exceedance is reached at this exact time.

1.2 Input data required

CHIMERE is a portable model which has been adapted to various types of input data. It requires meteorological data, boundary conditions, land-use information, emissions. In all cases but emissions, a default solution is proposed for the user. Refinements on input data have to be made by the user.

- **Meteorology:** Meteo data is not provided on the server, but a CHIMERE interface for the MM5 mesoscale is proposed. MM5 (free software [http://box.mmm.ucar.edu/mm5]) can be used combined with AVN/NCEP (as forcing) public weather forecasts.
- **Boundary conditions:** A set of boundary conditions from the MOZART II model is proposed as a default solution. Data are kindly provided by Max-Planck Institut, Hamburg, thanks to M. Schultz, G. Brasseur, C. Granier and D. Niehl. This allows tropospheric simulations below 200 hPa.
- **Land-use:** CHIMERE needs input landuse information, as well as biogenic emission potentials, based on land cover. The default proposed land use is the GLCF data base [http://glcf.umiacs.umd.edu]). The data for deriving biogenic emission potentials only cover Europe. It needs to be completed for other continents.

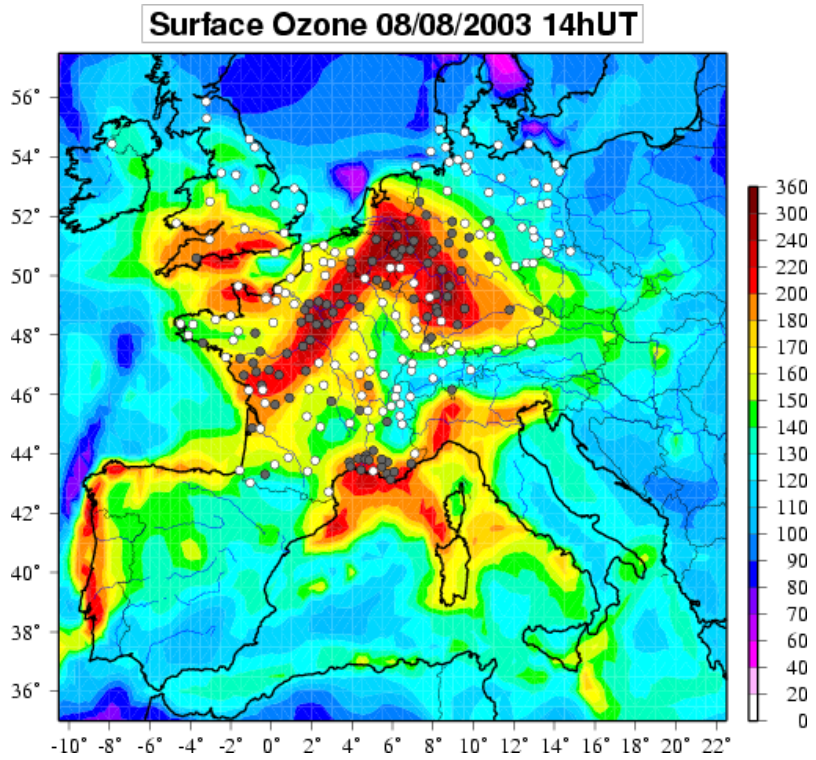


Figure 1.1: *The simulation of the 2003 heat wave: surface ozone concentrations ($\mu g.m^{-3}$)*

- **Emissions:** Emissions are usually coming from local, nonpublic databases. Here, for model testing, emissions are proposed for Western Europe at a resolution of 1/2 degree, and for the region (150Kmx150Km) around Paris.

1.3 OS, performance and software

The model has only been tested on PCs under GNU/Linux. It should also easily work on SUN and other UNIX systems with minor changes. Most changes should be due to the unformatted binary data interfaces. Please report to us all problems. A typical time for simulation on a PC, using the European version with a 65x33x8 grid, is 5 minutes for one simulated day with the gas-phase version, and 30 minutes with the aerosol version. It can then easily simulate entire seasons or years. A minimal setting of 1 Gb RAM is necessary. Software required is a Fortran 95 compiler (g95 is really great and free!). Simple graphical interfaces are available using either the GMT [`http://gmt.soest.hawaii.edu/`] or the GrADS [`http://grads.iges.org/grads/grads.html`] free softwares (to be installed).

1.4 The physics and numerics in short

- The chemical mechanism (MELCHIOR) is adapted from the original EMEP mechanism.
- Photolytic rates are attenuated using liquid water or relative humidity
- Boundary layer turbulence is represented as a diffusion (Troen and Mahrt, 1986, BLM)
- Vertical wind is diagnosed through a bottom-up mass balance scheme.
- Dry deposition is as in Wesely (1989). Wet deposition is included.

- Six aerosol sizes represented as "bins" in the model.
- Aerosol thermodynamic equilibrium is achieved using the ISORROPIA model.
- Several aqueous-phase reactions considered
- Secondary organic aerosols formation considered
- Advection is performed by the PPM (Piecewise Parabolic Method) 3d order scheme for slow species.
- The numerical time solver is the TWOSTEP method.

1.5 What can be done with

- Use the model for simulating episodes or long-term periods.
- Use the model for operational forecasting.
- Use the model for studying emission scenarios.
- sensitivity tests: CHIMERE is an easy-to-modify code which allows testing of various processes and making sensitivity tests. For instance the influence of new chemical reactions can be tested.
- Model development: It is the wish of the authors that CHIMERE be an open model, hence all model modifications in the direction of larger realism are welcome. There are a number of such improvements still to be done.

Chapter 2

Last changes: the version V200511B

The user will find here a description of the model evolution for the last version. The following informations are also reported into the CHANGELOG file (in the model source directory).

There are no changes in the scientific core of the model.
But this version represents a major change in the format of internal and output files, since most of the files are now written in the netCDF format. The rationale was :

- to allow for a better portability among different computer architectures,
- to allow for the use of widespread post-processing and analysis tools,
- to increase I/O performance through direct record addressing.

Another change is the generalization of Fortran 95 as a programming language.
However, the format of the external interface input files has not changed. There are no modification in the database format and structure.
We will not walk in detail through all the changes, since most subroutines where file I/O are involved have been completely rewritten.
To summarize, from the user point of view, the main changes are :

- modification in the chimere scripts. The previous monolithic script has been splitted into:
  - a main script "chimere-mm5" which deal with model parameters and file locations,
  - an execution script "chimere-mm5.exec which does all the low-level job setup and execution.
- more error management included in the scripts.
- error management in the netCDF I/O, which reports the line number in the source code where the error occured.
- use of "Makefiles" to drive the program compilation.
- rewriting of the meteo interface script.
- use of configuration files or namelists (*.nml) passed as arguments to scripts or executables. The previous way was to pass arguments through Unix pipes. The new way makes debugging easier since configuration files are static and can be edited to run separately some parts of the model.
- Model internal emission and boundary concentrations files are now in the netCDF format, but the files in the data base remain in their original binary format. The preparation routines have been modified accordingly. For those who use only the emission and boundary concentrations data provided with CHIMERE distribution, there is nothing to change. Those who use their own database may have

to check the preparation programs : prepemis_std.f90, preplat_lmdzinca2.f90, preptop_gocart.f90, preplat_mozart2.f90, preptop_lmdzinca2.f90, preplat_gocart.f90, preptop_mozart2.f90 for compatibility with their data.

- Model internal meteo files are now in the netCDF format. But the external format is still the well known MM5 binary format. Users who connect CHIMERE to MM5 will not have to modify the meteo interface. In other cases, users will have to rewrite their own meteo interface, taking prepexmm5.f90 as a template.
- In the MM5 meteo interface, prepmm5.f and the interpolation routine exdom-mm5.f have been merged into a single program prepexmm5.f90.
- The meteo diagnostic program has been split into two programs :
  - ☐ diagmet for pure meteo diagnostics
  - ☐ diagbio for bioemissions calculations
- diagmet has been deeply rewritten, but care has been taken to separate the netCDF overhead mainly contained in diagmet.f90, from the scientific core of the parametrizations contained in diagmet_science.f90.
- The processing of meteo information is now :
  - ☐ input : MM5 (raw binary)
  - ☐ processed by prepexmm5.e to produce :
  - ☐ intermediate : exdomout.nc (netCDF)
  - ☐ processed by diagmet.e to produce :
  - ☐ final : METEO.nc (netCDF)
  - ☐ used by chimere.e
- The output concentration files created by chimere.e are now in the netCDF format. Users will have to modify their post-processing programs. A template program to read CHIMERE netCDF files is provided in toolbox/readcdf directory.
- the visualization tool "mapit" has been adapted to netCDF and renamed "ncmapit".
- another visualization tool, "exachim", has been added.
- Most fundamental physical constants are now defined in a single file. This may lead to subtle numerical differences in the results, compared to V200501H.

Chapter 3

Model scientific documentation

3.1 Main options

CHIMERE is an air-quality simulation model with several options making it a flexible tool for detailed episode analysis as well as forecasting or long-term simulations. It can be adapted to many meteorological input data. CHIMERE is invoked by a **top calling script** named chimere-mm5 where the options for the simulations/forecasts are defined. This section lists these main options:

- The user must choose between the **gas-phase** only or the **aerosol** option. The gas-phase version is much lighter in terms of computational cost (about 4-5 times faster) than the aerosol version.
- The chemical mechanism is optional: it contains mechanisms auch as MELCHIOR1 and MELCHIOR2.
- The horizontal transport may be done using a simple first-order scheme, the Van leer scheme or the Parabolic Piecewise Method (PPM)
- The meteorological data to be used is a main option. In the present version we propose an interface for the MM5 model. The top script parameter is "meteo". If the user's data come from an other model or are given in another format, user should build her/his own interface and name it meteo=[new_interface_name].
- The boundary conditions are issued from global models such as GOCART for aerosols, LMDz-INCA or MOZART2 for gas and aerosol.

3.2 Model geometry

3.2.1 Horizontal grid

The model uses any regular rectangular horizontal grid system as presented in the **Figure 3.1**. The grid is entirely defined by a user-provided coordinate file named COORD_[domain] containing the geographical coordinates of the grid cell centres. From that file the cell edge lengthes are calculated and used in the calculation of transport fluxes. The cell surface is also calculated. All these calculations take into account sphericity to first order. One should not use grid cells with too large dimensions (less than, say, 2 degrees or 200 Km). All geometrical definitions and calculations are set up at once in the initialization routine **inigeom.f90** in the **src/** directory.
The horizontal grid is generally derived from the grid on which the emission inventory is given, in order not to interpolate (and then smooth) emissions. As input to CHIMERE, emissions must be given on the

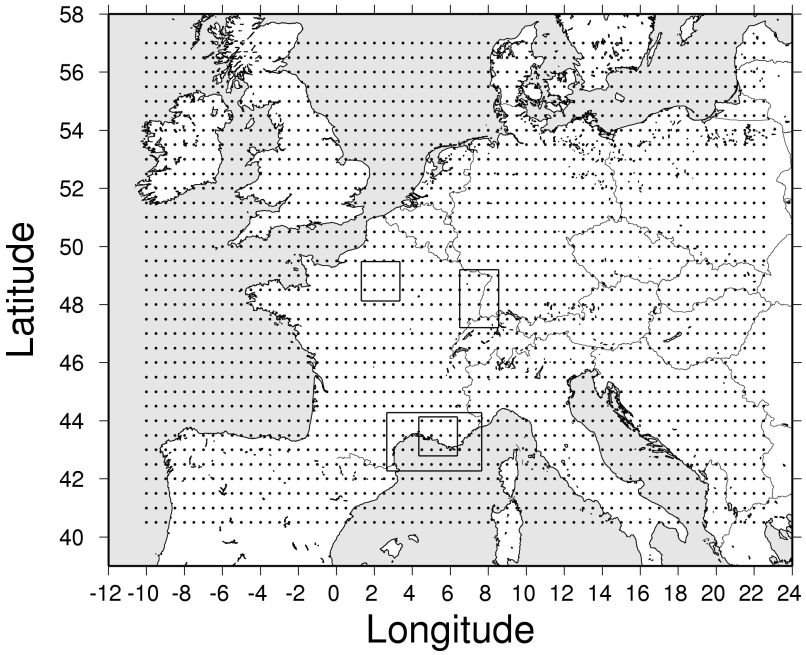


Figure 3.1: *CHIMERE domains*

model grid. There is no particular need for the grid to coincide with the meteorological or land-use grid. Interpolation programs are provided for MM5 outputs.

3.2.2 Vertical grid

The model uses any number of vertical layers, described in hybrid sigma-p coordinates. The pressure in hPa at the top of each layer k is given by the following formula:

$$P_k = a_k 10^5 + b_k P_{surf}$$

where P_{surf} is the surface pressure and the coefficients a_k and b_k are given.
The hybrid coefficients a_k and b_k are provided by the user as an input file, **VCOORD.[vgname]**, in the **data/vgrid/** directory. [vgname] is a user-defined name. As an option, the vertical grid can be defined automatically given the top pressure and sigma at the top of the surface layer. Then the resolution varies upward as a geometric progression. These options are defined in the top calling script and are explained in the **Appendix I.2, p.112**.

Recommandations

- When defining manually the hybrid coefficients the user must verify that under all possible meteorological conditions these layers do not cross one another or get mixed up.
- It is highly recommended that top level be a pressure level (b_k = 0), since it has been found to minimize mass loss through top boundary.
- Using the MOZART II boundary conditions, one should not use a model top above 200 hPa
- In the present formulation for cloud/radiation/photolysis, it is assumed that the model domain is "below the clouds", i.e. no cloud albedo is taken into account. This puts constraints on model top to be not too high. This will be changed in future releases.

- Deep convection mass fluxes are not taken into account in the present formulation. Again, this will be changed in future releases, but puts also constraints on model top.
- For these reasons model ozone and other gases and particles should not be considered too seriously far above the boundary layer. In the present version V200511B the model top should not exceed 500 hPa.

3.2.3 Grid nesting

Model simulations can be nested in a one-way mode. That is, a coarse CHIMERE simulation can give boundary conditions to a high-resolution simulation. Nested simulations must be performed one after the other. There is no need for any correspondence or relationship between the coarse and the fine horizontal grids. The vertical grids are not required to match : A nested grid may have a higher vertical resolution. This allows for performing a coarse simulation wiith moderate vertical resolution and a high-resolution simulation with accuracy in all three directions.

3.3 Transport and mixing

3.3.1 Horizontal transport

The horizontal transport is taken into account by calculating pollutant fluxes at the boundaries of each model cell. For a given pollutant, let us denote by C_{ij} the concentration at the center of each model cell, and U_{ij} and V_{ij} the associated wind components (zonal+meridian). First, wind is calculated at each of the 4 cell boundaries, taking into account only the normal component, by averaging the values of this wind component between the cell and its adjacent one. Let us assume that we are dealing with the interface between a cell i,j and its western one i-1,j. The normal component of the wind is thus **U**=1/2(U_{ij}+U_{i-1j}), and the thickness of the interface is **H**=1/2(H_{ij}+H_{i-1j}). The flux **F** used for changing concentration in cell i,j is then calculated as:

$$F = U \frac{H}{H_{ij}} \frac{C}{L_{ij}}$$

which has the dimension of a concentration time derivative, and where **C** is the pollutant concentration at the interface, and L_{ij} the zonal extent of the cell.
The concentration **C** is calculated either by the **UPWIND** method or by the **PPM** (Parabolic Piecewise Method, [Colella and Woodward, 1984]), depending on the pollutant. In order to reduce computer time, PPM is used only for a reduced set of "slow" species. These species are listed in the **PPM_SPECIES** input file. All the other ones are transported using the **UPWIND** method (that is, using **C**=C_{i-1j} when **U**$>$0 and **C**=C_{ij} when **U**$<$0).
Since most large-scale weather models do not contain any "urban parametrization", the possibility of correcting the wind speed in the surface layer (due to increased roughness) in cities is offered. This will automatically be balanced by a vertical wind component calculated in the mass balance (see vertical transport below). This correction has however no effect at continental scale where anyway the fraction of urban area in model grid cells remains very small. It has effects on urban versions of the model, and mostly on primary pollutants. The urban parameterization option switch lies in the **chimere-[metname]** top calling script, and is relayed by the **METINFOS-[metname].nml** file to be given as input to **src/diagmet.f90**. However it is not advised to use the switch for MM5 data.

Horizontal mass fluxes (independent of species, first order) are calculated first at each coarse time step in routine **htransport.f90**, then the complete calculation using the species concentration and PPM is performed at each fine time step in routine **transmix.f90**.

3.3.2 Vertical transport

Vertical transport is assumed to balance horizontal mass divergence/convergence. The scheme used is the first-order UPWIND scheme. More precisely, the algorithm is as follows: In the lowest layer, the mass imbalance is calculated by summing up all incoming mass fluxes and substracting all outgoing mass fluxes, and a vertical wind through the interface between the lowest and the second layer is calculated to compensate this imbalance. Then the same operation is performed on the second layer, including now in the mass imbalance the fluxes at the cell bottom interface. Then all layers are scanned upward to the top of the model. Vertical transport mass fluxes (independent of species, first order) are calculated first at each coarse time step in routine **src/vtransport.f90**, then the complete calculation using the species concentration is performed at each fine time step in routine **src/transmix.f90**.

3.3.3 Turbulent mixing

Horizontal turbulent fluxes are not considered. Vertical turbulent mixing takes place only in the boundary-layer. The formulation uses K-diffusion following the parameterization of [Troen and Mahrt, 1986], without counter-gradient term. In each model column, diffusivity K(z) is calculated as:

$$K_z = k w_s z \left(1 - \frac{z}{h}\right)^{1/3} \tag{3.1}$$

where w_s is a vertical scale given by similarity formulae.

- In the stable case (surface sensible heat flux < 0): $w_s = u_*/(1 + 4.7z/L)$
- in the unstable case: $w_s = (u_*^3 + 2.8ew_*^3)^{1/3}$

where e = max(0.1,z/h), **L** is the Obukhov Length, w_* is the convective velocity scale, u_* the friction velocity and h the boundary layer height. A minimal K(z) is assumed, with a value of 0.01 m^2/s.
The boundary layer height h can be either directly provided by the meteorological model or recalculated in the **src/diagmet.f90** meteo diagnostic model. In the latter case the boundary layer height is considered as the maximum of the [Troen and Mahrt, 1986] boundary layer height calculated from the Richardson number profile, as the lowest altitude where R_i = 0.5, and a more convectively-based boundary layer height calculation. The latter is based on a simplified and diagnostic version of the approach of [Cheinet, 2002]. It consists in the resolution of the (dry) thermal plume equation with diffusion. The in-plume vertical velocity and buoyancy equations are solved and the boundary layer is taken as the height where vertical velocity stops. Thermals are initiated with a non-vanishing vertical velocity and potential temperature departure, depending on the turbulence similarity parameters in the surface layer.
At each interface between layers k and k+1, one calculates an equivalent turbulent vertical velocity V_k and vertical mixing is expressed in layer k by the mass flux:
Net Flux = $\mathbf{V}_k(\mathbf{C}_{k+1}.\mathbf{D}_k/\mathbf{D}_{k+1} - \mathbf{C}_k)/\mathbf{H}_k$,
where **C** denotes concentrations, **D** densities and **H** layer thicknesses. Density ratios are applied since turbulent mixing must conserve mass in each model cell. An equivalent formula is used for the upper layer. The turbulent velocity V_k is deduced from K(z) using:

$$V_k = \frac{K_z}{(\frac{1}{2}(H_k + H_{k+1}))} \tag{3.2}$$

When the top of the boundary layer is above the model top, H_{k+1} is replaced by h - <model-top> in the previous equation.
The K(z) profile is calculated in routine **src/mixing.f90**, as well as the turbulent velocity profile. Vertical mixing mass fluxes (independent of species, first order) are calculated first at each coarse time step in routine **src/vtransport.f90**, then the complete calculation using the species concentration is performed at each fine time step in routine **src/transmix.f90**.

3.4 Aerosol processes

CHIMERE consists now in the baseline gas-phase version with MELCHIOR2 chemistry, together with a sectional aerosol module. This module accounts for 7 species (primary particle material, nitrate, sulfate, ammonium, biogenic secondary organic aerosol (SOA), anthropogenic SOA and water). Potentially, Chloride et Sodium can be included (high computing time). In its initial version the module uses 6 bins from 10 nm to 40 μm. Physical processes into account are described below. See also sections on chemistry (aqueous-phase chemistry, heterogenous chemistry and SOA formation) below. Dry and wet scavenging/deposition are also described in separate sections.
The other physical processes accounted for are:

- **Coagulation** : Classical theory already used by [Gelbard and Seinfeld, 1980].
- **Absorption** : Absorption is taken into account for both inorganic and organic species but their processing is different. For inorganic species, equilibrium concentrations are computed with the thermodynamic module ISORROPIA (adress p.76) (or a tabulated version) presented in [Nenes et al., 1998]. For secondary organic species, equilibrium concentrations are calculated through a temperature dependent partitioning coefficient ([Pankow, 1994]. Please note that the ISORROPIA code is included in CHIMERE, please check the ISORROPIA web site for downloading the latest version.
- **Nucleation** : The parameterization of [Kulmala et al., 1998] for sulfuric acid nucleation is used. This process, favoured by cold humid atmospheric conditions, affects the number of ultrafine particles. The nucleated flux is added to the smallest bin in the sectional distribution. Although nucleation of condensable organic species has been clearly identified in many experimental studies ([Kavouras et al., 1998], there is no available parameterization. Since the sulfuric acid nucleation process competes with absorption processes, it is expected to occur in weakly particle polluted conditions.

3.5 Land use and related parameters

3.6 Meteorological input data

The CHIMERE model requires meteorological input variables ; these are :

- **3D data:**
  - ☐ Horizontal wind (transport)
  - ☐ Temperature (chemistry,thermodynamics)

- ☐ Density (chemistry and transport)
- ☐ Specific humidity (chemistry/thermodynamics)
- ☐ Height of model layers (model geometry)
- ☐ Cloud liquid water content (for the aerosol option, aqueous-phase chemistry, scavenging)

- **2D data:**
  - ☐ Temperature at 2m (deposition, biogenic emissions)
  - ☐ Photolysis attenuation in the column due to clouds (chemistry/photolysis)
  - ☐ Boundary layer height for mixing
  - ☐ Friction velocity u^* for mixing and deposition
  - ☐ Obukov length for mixing
  - ☐ Convective velocity scale w^* for mixing
  - ☐ Aerodynamic resistance for deposition
  - ☐ Total Precipitation rate for aerosol scavenging

These data are obtained either directly or through the proposed 2-step chain of preprocessing (see the technical documentation) from meteorological analyses or forecasts from standard variables issued in meteorological simulations. An interface is proposed for the MM5 model. This interface contains horizontal and time interpolation, and a diagnostic meteo model (the **src/diagmet.f90** program), allowing diagnostic calculation and vertical interpolation of meteo variables. The model has now been tested using meteorological data from AVN/NCEP, MM5, ECMWF, ARPEGE and ALADIN models.
Among the variables that may be calculated from the interface are:

- Radiation attenuation
- Boundary layer height
- Friction velocity U* (From the [Louis, 1979],[Louis et al., 1982] parameterization)
- Aerodynamic resistance ([Seinfeld and Pandis, 1997])
- Sensible heat flux (read or recalculated using the [Priestley and Taylor, 1967] formula.
- Obukov length
- Convective velocity W*

The other variables (3D) are vertically averaged into the CHIMERE layers.

3.7 Boundary concentrations

Boundary conditions can be either "external" or given by a coarse-resolution CHIMERE simulation. In the first case and in the present release, a choice of monthly climatologies is offered between the MOZART second-generation global chemistry-transport model (MOZART2) [Horowitz et al., 2003], or the LMDZ-INCA2 chemistry-transport model. The concentration of a few species are considered as constant (but depending on month) throughout the simulation. The MOZART climatologies have been calculated and are kindly provided here by the Max-Planck Institut, Hamburg, by M. Schultz, G. Brasseur, C. Granier and D. Niehl. In the second case, a first coarse run must have been carried out.
The user can also provide a different, independent set of boundary conditions to the model, which can be time-varying or constant during the simulation. A new interface to build data for CHIMERE has to be constructed. The current interfaces in **src/** directory are **preplat_mozart2.f90**, **preplat_lmdzinca2.f90**, and **preplat_chimere.f90** for lateral boundaries, and **preptop_mozart2.f90**, **preptop_lmdzinca2.f90** and **preptop_chimere.f90** for top boundaries. See technical documentation for the format of boundary conditions files.

Several other files are needed, for describing the species considered at the boundaries. For MOZART2 lateral or top boundaries, only the following species are present in the available files:

- O3 ..... Ozone
- NO2 ..... Nitrogen Dioxide
- CO ..... Carbon Monoxide
- PAN ..... Peroxy-Acetyl-Nitrate
- CH4 ..... Methane
- C2H6 ..... Ethane
- HCHO ..... Formaldehyde
- HNO3 ..... Nitric Acid

The file containing the MOZART2 "lateral" species names is **LAT_SPEC.MOZART2**. The file containing the "top" species names is **TOP_SPEC.MOZART2**. If other interfaces had to be built, two other files like these ones would have to be present in the data_[model]_[meca] directories.

There is no yet coupling to any "global" climatology for aerosols. Therefore the boundary aerosol concentrations are set to 0.

When performing a nest run it is advised to output at least certain number of species in the coarse simulation. Here is the list of compounds passed from the coarse run to the high-resolution run at the boundaries. The list is only valid for the gas-phase model. Check the boundary species for the aerosol version equivalent file in the data_aero_[meca] directory.

- O3 ..... Ozone
- NO2 ..... Nitrogen Dioxide
- NO ..... Nitrogen Monoxide
- PAN .. Peroxy-Acetyl-Nitrate
- HNO3 ..... Nitric Acid
- SO2 ..... Sulphur Dioxide
- CO ..... Carbon Monoxide
- CH4 ..... Methane
- C2H6 ..... Ethane
- NC4H10 ..... Butane
- C2H4 ..... Ethene
- C3H6 ..... Propene
- OXYL ..... o-Xylene
- C5H8 ..... Isoprene
- APINEN ..... Alpha-Pinene
- HCHO ..... Formaldehyde
- CH3CHO ..... Acetaldehyde
- GLYOX ..... Glyoxal
- MGLYOX ... Methyl-Glyoxal
- CH3COE ..... Methyl-Ethyl Ketone

The boundary species names and species are read in routine **src/iniboun.f90**. Then in practice the boundary concentrations values are written to the edges outside the **conc** array containing all concentrations in routine **src/locvalues.f90**.

3.8 Initial concentrations

There are two possible initialization modes:

- Initialization by reading initial concentrations in a restart file. This file must contain the "end" concentrations of a previous simulation, as provided in the **end.[label].nc**. The species list is contained in the **end.[label].nc** file itself and is read by CHIMERE during initialization.
- Initialisation by interpolating boundary conditions. This is the case when no initial file is available.

In practice, the top calling script checks the existence and nonemptiness of a file called **ini.[label].nc** in the simulation directory, which can be a copy of an "end" file. If it exists and is nonempty, then the initial conditions are used. In the other case, initial conditions are interpolated from boundary conditions. Initial conditions are read and put into arrays **conc** and **conco** in routine **src/iniconc.f90**.

3.9 Chemistry

3.9.1 Chemical mechanisms (gas phase)

CHIMERE offers the option to include different gas phase chemical mechanisms. The original, complete scheme ([Lattuati, 1997]), hereafter called MELCHIOR1, describes more than 300 reactions of 80 gaseous species.
The hydrocarbon degradation is fairly similar to the EMEP gas phase mechanism ([Simpson, 1992]). Adaptations are made in particular for low NO_x conditions and NO_x-nitrate chemistry. All rate constants are updated according to [Atkinson et al., 1997] and [De Moore et al., 1994].
Heterogeneous formation of HONO from deposition of NO2 on wet surfaces is now considered, using the formulation of [Aumont et al., 2003]. In order to reduce the computing time a reduced mechanism with 44 species and about 120 reactions is derived from MELCHIOR ([Derognat et al., 2003]), following the concept of "chemical operators" ([Carter, 1990]). This reduced mechanism is called MELCHIOR2 hereafter.
The chemical mechanisms files are placed in the data_[model]_[meca]. New mechanisms can be introduced. In this case, copy for instance the directory data_gas_MELCHIOR2 onto data_gas_XXXX, and change the **REACTIONS** file for a new set of reactions. Not all forms of reactions are allowed. Arrhenius, Troe, photolysis and few other classical reaction rate types are only allowed. Introducing other types implies some recoding. See the technical documentation (the chemprep preprocessor). A file with altitude-dependent photolysis rates must also be provided, named **PHOTO_PARAMETERS.[meca]**. Then run the **chemprep** chemical preprocessor to make FORTRAN-readable files.
Photolysis rates are calculated under clear sky conditions as a function of height using the TUV model ([Madronich et al., 1998]). Then clouds are taken into account in a highly parameterized fashion (see next Section).
Gas-phase chemistry constants is read in initialization routine **src/inichem.f90**. Then rates are calculated at the coarse time step in routine **src/rates.f90**, but chemical fluxes are calculated for all species at the fine time step in routine **src/chemistry.f90**.

3.9.2 Species list of the reduced MELCHIOR2 gas-phase chemical mechanism

Species	Definition
Inorganic compounds	
O3	ozone
H2O2	hydrogen peroxide
OH	hydroxy radical
HO2	hydroperoxy radical
NO	nitric oxide
NO2	nitrogen dioxide
NO3	nitrogen trioxide
N2O5	dinitrogen pentoxide
HONO	nitrous acide
HNO3	nitric acide
CO	carbon monoxide
SO2	sulfur dioxide
Hydrocarbon species[1]	
CH4	methane
C2H6	ethane
NC4H10	n-butane
C2H4	ethene
C3H6	propene
OXYL	o-xylene
C5H8	isoprene
APINEN	α-pinene
Carbonyls	
HCHO	formaldehyde
CH3CHO	acetaldehyde
CH3COE	methyl ethyl ketone
GLYOX	glyoxal
MGLYOX	methyl glyoxal
CH3COY	dimethyl glyoxal
MEMALD	unsaturated dicarbonyls, reacting like 4-oxo-2-pentenal
MVK	methyl vinyl ketone
MAC	methacroleine

Organic nitrates

PAN ..... peroxyacetyl nitrate
CARNIT ..... nitrate carbonyl taken as α-nitrooxy acetone
ISNI ..... unsaturated nitrate from isoprene degradation

Organic peroxides

CH3O2H ..... methyl hydroperoxide
PPA ..... peroxy acetyl acid

(Per)oxy radicals

CH3O2 ..... methyl peroxy radical
CH3COO ..... peroxy acetyl radical

Operators [2]

oRO2 representing peroxy radicals from OH attack to C2H5, NCHH10, C2H4, C3H6, OXYL, CH3COE, MEMALD, and MVK
oROOH .. representing organic peroxides from oRO2+HO2 reactions
obio ... representing peroxy radicals produced by C5H8 and APINEN + OH reaction
obioH ..... representing biogenic organic peroxides from obio+HO2 and obio+obio reactions
oPAN .... representing PAN homologue compounds (except PAN)
PANH ..... representing results from oPAN+HO2 reaction
toPAN representing results from oPAN+NO2 reaction
oRN1 representing organic nitrate peroxy radicals from NO3 attack to C2H4, C3H6, C5H8, APINEN and OH attack to ISNI

Reaction list of the reduced MELCHIOR2 chemical mechanism

Reactions	Kinetic constants	Ref.
Inorganic chemistry		
O_3+NO → NO_2	$Ae^{-B/T}$,A=1.8e-12,B=1370	[1]
O_3+NO_2 → NO_3	$Ae^{-B/T}$,A=1.2e-13,B=2450	[1]
O_3+OH → HO_2	$Ae^{-B/T}$,A=1.9e-12,B=1000	[1]
O_3+HO_2 → OH	$Ae^{-B/T}$,A=1.4e-14,B=600	[1]
NO+HO_2 → OH+NO_2	$Ae^{-B/T}$,A=3.7e-12,B=-240	[1]
NO_2+OH+M→ HNO_3	troe(3.4e-30,0,3.2,4.77e-11,0,1.4,0.30)	[2], n9
HO_2+OH→ H_2O	$Ae^{-B/T}$,A=4.8e-11,B=-250	[1]
H_2O_2+OH→ HO_2	$Ae^{-B/T}$,A=2.9e-12,B=160	[1]
HNO_3+OH→ NO_3	$Ae^{-B/T}$,A=5.5e-15,B=-985	[3], n1
CO+OH→ HO_2+CO_2	$Ae^{-B/T}(300/T)^N$, A=2e-13, B=0, N=1	[3], n1
HO_2+HO_2 → H_2O_2	$Ae^{-B/T}$,A=2.2e-13,B=-740	[1], n1
HO_2+HO_2+H_2O→ H_2O_2	$Ae^{-B/T}$,A=4.52e-34,B=-2827	[1], n1
NO_3+HO_2→ NO_2+OH	4e-12	[1]
NO_3+H_2O_2 → HNO_3+HO_2	2e-15	[3]
NO_3+NO→ 2*NO_2	$Ae^{-B/T}$,A=1.8e-11,B=-110	[1]
NO_2+NO_3→ NO+NO_2	$Ae^{-B/T}$,A=4.5e-14,B=1260	[3]
NO_2+NO_3+M→ N_2O_5	troe(2.7e-30,0,3.4,2e-12,0,-0.2,0.33)	[1], n9
N_2O_5+M→ NO_3+NO_2	troe(1e-3,11000,3.5,9.7e14,11080,-0.1,0.33)	[1], n9
N_2O_5+H_2O→ 2*HNO_3	2.6e-22	[4]
N_2O_5+H_2O+H_2O→ 2*HNO_3	2e-39	[4]
NO+OH+M→ HONO	troe(7.e-31,0,2.6,1.5e-11,0,0.5,0.6)	[1], n9
HONO+OH→ NO_2	$Ae^{-B/T}$,A=1.8e-11,B=390	[1]
NO+NO+O_2→ 2*NO_2	$Ae^{-B/T}$,A=3.30E-39,B=-530.0	[1]

[2]For the concept of “chemical operators” see Carter (1990) and Aumont et al. (1997)

Reactions	Kinetic constants	Ref.
NO_2→ HONO	0.5*depo(NO_2)	[18]
SOx chemistry		
SO_2+CH_3O_2→ H_2SO_4+HCHO+HO_2	4e-17	[3]
SO_2+OH+M→ H_2SO_4+HO_2	troe(4e-31,0,3.3,2e-12,0,0,0.45)	[3], n9
OH attack to hydrocarbons		
CH_4+OH→ CH_3O_2	$Ae^{-B/T}$,A=2.3e-12,B=1765	[1]
C_2H_6+OH→ CH_3CHO+oRO_2	$Ae^{-B/T}$,A=7.9e-12,B=1030	[1]
NC_4H_{10}+OH→ 0.9*CH_3COE +0.1*CH_3CHO +0.1*CH_3COO +0.9*oRO_2	$Ae^{-B/T}(300/T)^N$,A=1.36e-12, B=-190, N=-2	[5]
C_2H_4+OH+M→ 2*HCHO+oRO_2	troe(7e-29,0,3.1,9e-12,0,0,0.7)	[1], n9
C_3H_6+OH+M→ HCHO+CH_3CHO+oRO_2	troe(8e-27,0,3.5,3e-11,0,0,0.5)	[1], n9
OXYL+OH→ MEMALD+MGLYOX+oRO_2	1.37e-11	[5]
C5H_8+OH→ 0.32*MAC +0.42*MVK +0.74*HCHO +obio	$Ae^{-B/T}$, A=2.55e-11, B=-410	[1]
APINEN+OH→ 0.8*CH_3CHO+0.8*CH_3COE+obio	$Ae^{-B/T}$, A=1.21e-11, B=-444	[17]
OH attack to carbonyls and peroxides		
HCHO+OH → CO+HO_2	$Ae^{-B/T}$,A=8.6e-12,B=-20	[1]
CH_3CHO+OH →CH_3COO	$Ae^{-B/T}$,A=5.6e-12,B=-310	[1]
MEMALD+OH →GLYOX+MGLYOX+oRO_2	5.6e-11	[1]
CH_3COE+OH →CH_3COY+oRO_2	$Ae^{-B/T}(300/T)^N$, A=2.92e-13, B=-414, N=-2	[1]
GLYOX+OH → 2*CO+HO_2	1.1e-11	[1]
MGLYOX+OH → CH_3COO+CO	1.5e-11	[1]
MVK+OH → 0.266*MGLYOX +0.266*HCHO +0.684*CH_3CHO +0.684*CH_3COO +0.05*ISNI +0.95*oRO_2	$Ae^{-B/T}$,A=4.1e-12,B=-453	[6]
MAC+OH → 0.5*CH_3COE+0.5*CO_2+0.5*oPAN	$Ae^{-B/T}$,A=1.86e-11,B=-175	[6]
CH_3O_2H+OH → CH_3O_2	$Ae^{-B/T}$,A=1.9e-12,B=-190	[1]
PPA+OH → CH_3COO	$Ae^{-B/T}$,A=1.9e-12,B=-190	n2
CH_3O_2H+OH → HCHO+OH	$Ae^{-B/T}$,A=1.e-12,B=-190	[1]
oROOH+OH → 0.8*OH+0.2*oRO_2	$Ae^{-B/T}$,A=4.35e-12,B=-455	n2
obioH+OH → OH	8e-11	n2
PANH+OH → 0.2*oPAN	1.64e-11	n2
CARNIT+OH → CH_3CHO+CO+NO_2	k(T)=$Ae^{-B/T}$,A=5.6e-12,B=-310	
NO_3 attack to hydrocarbons and aldehydes		
C_2H_4+NO_3 → 0.5*CARNIT+HCHO+oRN1	2e-16 [1]	
C_3H_6+NO_3 → 0.5*CARNIT +1.5*HCHO +0.5*CH_3CHO +0.5*HO_2 +oRN1	9.45e-15	[1]
APINEN+NO_3 → CH_3CHO+CH_3COE+oRN1	$Ae^{-B/T}$,A=1.19e-12,B=490	[8]
C5H_8+NO_3 → 0.85*ISNI +0.1*MAC +0.05*MVK +0.15*HCHO +0.8*HO_2 +oRN1	7.8e-13	[6]
HCHO+NO_3 → CO+HNO_3+HO_2	5.8e-16	[1]
CH_3CHO+NO_3 → CH_3COO+HNO_3	2.8e-15	[9]
CH_3O_2+NO_3 → HCHO+HO_2+NO_2	1.2e-12	[10]
CH_3COO+NO_3 → CH_3O_2+NO_2+CO_2	4e-12	[10]
oRO_2+NO_3 → NO_2+HO_2	1.2e-12	[10]
obio+NO_3 → NO_2+HO_2	1.2e-12	[10]
oPAN+NO_3 → NO_2+HO_2	4e-12	[10]

Reactions	Kinetic constants	Ref.
oRN1+NO_3→ 1.5*NO_2	1.2e-12	[10]
O_3 attack to unsaturated hydrocarbons		
C_2H_4+O_3 → HCHO+0.12*HO_2+0.13*H_2+0.44*CO	$Ae^{-B/T}$,A=9.1e-15,B=2580	[1]
C_3H_6+O_3 → 0.53*HCHO +0.5*CH_3CHO +0.31*CH_3O_2 +0.28*HO_2 +0.15*OH +0.065*H_2 +0.4*CO +0.7*CH_4	$Ae^{-B/T}$,A=5.5e-15,B=1880	[1]
C5H_8+O_3 → 0.67*MAC +0.26*MVK +0.55*OH +0.07*C_3H_6 +0.8*HCHO +0.06*HO_2 +0.05*CO +0.3*O_3	$Ae^{-B/T}$,A=1.2e-14,B=2013	[6]
MAC+O_3 → 0.8*MGLYOX +0.7*HCHO +0.215*OH +0.275*HO_2 +0.2*CO +0.2*O_3	$Ae^{-B/T}$,A=5.3e-15,B=2520	[6]
MVK+O_3 → 0.82*MGLYOX +0.8*HCHO +0.04*CH_3CHO +0.08*OH +0.06*HO_2 +0.05*CO +0.2*O_3	$Ae^{-B/T}$,A=4.3e-15,B=2016	[6]
APINEN+O_3 → 1.27*CH_3CHO +0.53*CH_3COE +0.14*CO +0.62*oRO_2 +0.42*HCHO +0.85*OH +0.1*HO_2	$Ae^{-B/T}$,A=1.0e-15,B=736	[8]
organic radical conversion		
CH_3O_2+NO → HCHO+NO_2+HO_2	$Ae^{-B/T}$,A=4.2e-12,B=-180	[1]
CH_3COO+NO → CH_3O_2+NO_2+CO_2	2e-11	[1]
oRO_2+NO → NO_2+HO_2	4e-12	n3
obio+NO → 0.86*NO_2+0.78*HO_2+0.14*ISNI	$Ae^{-B/T}$,A=1.4e-11,B=180	[6]
oPAN+NO → NO_2+HO_2	1.4e-11	n3
oRN1+NO → 1.5*NO_2	4e-11	n3
organic radical recombination		
CH_3O_2+HO_2→ CH_3O_2H	$Ae^{-B/T}$,A=4.1e-13,B=-790	[1]
CH_3COO+HO_2→ 0.67*PPA+0.33*O_3	$Ae^{-B/T}$,A=4.3e-13,B=-1040	[1]
oRO_2+HO_2→ oROOH	$Ae^{-B/T}$,A=2.7e-13,B=-1000	n4
obio+HO_2→ obioH	$Ae^{-B/T}$,A=2.7e-13,B=-1000	n4
oPAN+HO_2→ PANH	$Ae^{-B/T}$,A=2.7e-13,B=-1000	n4
oRN1+HO_2→ X	$Ae^{-B/T}$,A=2.7e-13,B=-1000	n4
CH_3O_2+CH_3O_2→ 1.35*HCHO+0.7*HO_2	$Ae^{-B/T}$,A=1.13e-13,B=-356	[1], n5
CH_3COO+CH_3O_2→ 0.5*CH_3O_2+0.5*CO_2+HCHO+0.5*HO_2	$Ae^{-B/T}$,A=3.34e-12,B=-400	[1], n5
oRO_2+CH_3O_2→ 0.65*HCHO+0.8*HO_2+0.35*CH_3OH	$Ae^{-B/T}$,A=1.5e-13,B=-220	n5
obio+CH_3O_2→ 0.8*HO_2+0.5*HCHO	$Ae^{-B/T}$,A=2.44e-11,B=223	n5
oPAN+CH_3O_2→ HCHO+0.5*HO_2	$Ae^{-B/T}$,A=7.9e-12,B=-140	n5
CH_3COO+CH_3COO→ 2.*CH_3O_2+2.*CO_2	$Ae^{-B/T}$,A=2.8e-12,B=-530	[1], n5
oRO_2+CH_3COO→ 0.8*CH_3O_2+0.8*CO_2+0.8*HO_2	$Ae^{-B/T}$,A=8.6e-13,B=-260	n5
obio+CH_3COO→ 0.5*HCHO+1.5*HO_2+0.7*CO_2	$Ae^{-B/T}$,A=1.18e-11,B=127	n5
oPAN+CH_3COO→ CH_3O_2+CO_2+HO_2	$Ae^{-B/T}$,A=3.34e-12,B=-400	n5
oRO_2+oRO_2→ 1.3*HO_2	6.4e-14	n5
organic nitrates and pernitrates		
CH_3COO+NO_2+M → PAN	troe(2.7e-28,0,7.1,1.2e-11,0,0.9,0.3)	[1], n9

Reactions	Kinetic constants	Ref.
oPAN+NO_2+M → toPAN	troe(2.7e-28,0,7.1,1.2e-11,0,0.9,0.3)	n6 , n9
PAN+M → CH_3COO+NO_2	troe(4.9e-3,12100,0,5.4e16,13830,0,0.3)	[1], n9
toPAN+M → oPAN+NO_2	troe(4.9e-3,12100,0,5.4e16,13830,0,0.3)	n6 , n9
PAN+OH → HCHO+NO_3+CO_2	$Ae^{-B/T}$,A=9.5e-13,B=650	[1]
toPAN+OH → NO_3+CO_2	$Ae^{-B/T}$,A=3.25e-13,B=-500	n6
ISNI+OH → 0.95*CH_3CHO +0.475*CH_3COE +0.475*MGLYOX +0.05*ISNI +0.05*HO_2 + oRN1	3.4e-11	[6]
photolytic reactions		
O_3→ 2*OH	photorate	[3], n7
NO_2→ NO+O_3	photorate	[3]
NO_3→ NO_2+O_3	photorate	[3], [11]
NO_3→ NO	photorate	[3], [11]
N_2O_5→ NO_2+NO_3	photorate	[3]
H_2O_2→ 2*OH	photorate	[3]
HNO_3→ NO_2+OH	photorate	[3]
HONO→ NO+OH	photorate	[3]
HCHO→ CO+2*HO_2	photorate	[3]
HCHO→ CO+H_2	photorate	[3]
CH_3CHO→ CH_3O_2+HO_2+CO	photorate	[1]
CH_3COE→ CH_3COO+CH_3CHO+oRO_2	photorate	[11], [12]
CH_3COY→ 2*CH_3COO	photorate	[13], [14]
MGLYOX→ CH_3COO+HO_2+CO	photorate	[1]
GLYOX→ 0.6*HO_2+2*CO+0.7*H_2	photorate	[13], [14]
MEMALD→ 0.5*MVK +0.5*MALEIC +0.5*oPAN +0.5*HCHO +0.5*HO_2	photorate	[15]
CH_3O_2H→ HCHO+OH+HO_2	photorate	[3]
PPA→CH_3O_2+CO_2+OH	photorate	n8
oROOH→OH+HO_2	photorate	n8
obioH→ OH+HO_2	photorate	n8
PANH→OH+HO_2	photorate	n8
PAN→CH_3COO+NO_2	photorate	[3]

- [1] Atkinson et al., 1997
- [2] Donahue et al., 1997
- [3] DeMore et al., 1997
- [4] Mentel et al., 1996
- [5] Atkinson, 1990
- [6] Paulson and Seinfeld, 1992
- [7] Le Bras, 1997
- [8] Atkinson et al., 1994
- [9] DeMore et al., 1990
- [10] Canosa-Mas et al., 1996
- [11] Johnston et al., 1996
- [12] Martinez et al., 1992
- [13] Jenkins et al., 1997
- [14] Plum et al., 1983
- [15] Bierbach et al., 1994
- [16] Lightfoot et al., 1996
- [17] Kirchner and Stockwell, 1996
- [18] Aumont, 2002

- n1 : Rate constants containing a density dependent term in a minor reaction channel have been simplified by putting an average boundary layer molecular density of 2.5*10^{19} molecules/cm^3.
- n2 : For operator reactions with OH, oROOH is interpreted as 2-butyl hydrogen peroxide, obioH as a peroxide from isoprene degradation, PANH represents unsaturated organic peroxides; due to lack of data, the rate constants of these compounds and of peroxy acetyl acid have been estimated using structure reactivity relationships in Kwok and Atkinson (1995).

- n3 : The rate constant of the operator reaction oRO$_2$+NO has been set to that of the i-C$_3$H7O$_2$+NO reaction given in [16], that of oPAN+NO to that of CH$_3$COO$_2$+NO [1], the rate constant for oRO$_2$NO+NO is taken from that of ISNIR (nitrate peroxy radicals from isoprene degradation) + NO [6].
- n4 : The rate constants for the operator reactions oRO$_2$, obio, RO$_2$NI+HO$_2$ have been set to that of the reaction C$_2$H5O$_2$+HO$_2$ [1], that of oPAN+HO$_2$ has been taken from CH$_3$COO$_2$+HO$_2$ [1].
- n5 : Radical conserving and terminating pathways are combined to single reactions; rate constants in operator reactions are affected by interpreting oRO$_2$ as 2-butyl peroxy radical [16], obio as a peroxy radical from isoprene degradation [6] and oPAN as CH$_3$COO$_2$.
- n6 : Reaction rates of the operators oPAN and OPAN have been set to those of the peroxy acetyl radical and PAN, respectively.
- n7 : The reactions O$_3$+hν $\rightarrow$ O(1D), O(1D)+H$_2$O $\rightarrow$ 2OH, O(1D)+M $\rightarrow$ O(3P)+M and O(3P)+O$_2$+M $\rightarrow$ O$_3$+M have been combined to reaction O$_3$+h$\nu$$\rightarrow$ O(1D) by taking into account H$_2$O and M concentrations in order to adjust the photolysis frequency.
- n8 : Due to lack of data, photolysis frequencies of higher organic peroxides are taken as that of CH$_3$OOH.
- n9 : Three body Troe reactions [1], given in the form:

$$k = \frac{k_0[M]}{1 + \frac{k_0[M]}{k_\infty}} f^p$$

with

$$p = (1 + (log_{10}(k_0[M]/k_\infty))^2)^{-1},$$

$$k_0 = A_0 e^{\left(-\frac{B_0}{T}\right)} \left(\frac{T}{300}\right)^{-n}, \text{ and } k_\infty = A_\infty e^{\left(-\frac{B_\infty}{T}\right)} \left(\frac{T}{300}\right)^{-n},$$

The parameters are given in the order $A_0, B_0, n, A_\infty, B_\infty, m, f$

3.9.3 Aqueous-phase chemistry

If the aerosol option of the model is selected, Aqueous sulphate chemistry is considered; Sulfate is produced in the following aqueous reactions ([Berge, 1993]; [Hoffman and Calvert, 1985]; [Lee and Schwartz, 1983]):

- $SO_2^{aq} + O_3^{aq} \rightarrow SO_4^{2-}$
- $HSO_3^- + O_3^{aq} \rightarrow SO_4^{2-}$
- $SO_3^{2-} + O_3^{aq} \rightarrow SO_4^{2-}$
- $SO_2^{aq} + H_2O_2^{aq} \rightarrow SO_4^{2-}$
- $SO_2^{aq} + NO_2^{aq} \rightarrow SO_4^{2-}$
- $SO_3^{2-}(Fe^{3+}) \rightarrow SO_4^{2-}$
- $HSO_3^-(Mn^{2+}) \rightarrow SO_4^{2-}$

SO$_2$, H$_2$O$_2$ and O$_3$ in the aqueous phase are in equilibrium with the concentrations in the gas phase. Moreover, aqueous SO2 is dissociated into HSO_3^- and SO_3^{2-}. Catalyzed oxidation reactions of sulfur dioxide in aqueous droplets with iron and manganese are considered, following

[Hoffman and Calvert, 1985] among others. Henry's law coefficient and other aqueous equilibrium constants are used ([Seinfeld and Pandis, 1997]). Sulfur chemistry is very pH sensitive. pH is kept bound between 4.5 and 6.0.

3.9.4 Heterogeneous chemistry

Again with the aerosol option of the model, a few heterogeneous reactions are also considered. Nitric acid is produced onto existing particles and fog droplets. Although aerosol particles and cloud droplets occupy a very small fraction of the atmosphere, it is now well established that reactions involving gas species onto their surfaces may significantly contribute to atmospheric chemistry cycles. [Jacob, 2000] recommends for ozone model to include a minimal set of reactions with associated uptake coefficients given by [Harrison and Kito, 1990], and other references in [Jacob, 2000]:

- $HO_2 \rightarrow 0.5H_2O_2$ ..... γ = 0.2
- $NO_3 \rightarrow HNO_3$ ..... γ = 0.001
- $NO_2 \rightarrow 0.5HNO_3$ + 0.5HONO ..... γ = 0.0001
- $N_2O_5 \rightarrow 2HNO_3$ ..... γ = 0.01 - 1

The first-order rate constant k for gas heterogeneous loss onto particles is given by:

$$k = \sum_l \left(\frac{d_l}{2D_g} + \frac{4}{V\gamma} \right)^{-1} A_l \tag{3.3}$$

with d_l, the particle diameter (m), D_g, the reacting gas molecular diffusivity (m2 s^{-1}), n, the mean molecular velocity (m s-1), A_l, the total surface area in the particle bin l and gamma the uptake coefficient of reactive species. The uptake coefficient for N_2O_5 is assumed to be temperature-dependent in the range 0.01 - 1 ([De Moore et al., 1994]) with increasing values for decreasing temperatures. A recent study ([Aumont et al., 2003]) suggests that NO2 reactions on wet surfaces could be an important source for HONO production during wintertime smog episodes, so, a new reaction is added (also present in the gas-phase mechanism, see above).

3.9.5 Secondary organic aerosol chemistry

Secondary organic aerosols (SOA), produced through photochemical reactions in the atmosphere contain only oxidized species, carbonyls, carboxylic acid, etc.. [Pun et al., 2002], [Griffin et al., 2002a], and [Griffin et al., 2002b] propose a very complete modeling system for secondary organic formation and partitioning between gas and particle phases.
Up to now, a very simplified scheme for SOA formation has been implemented in the chemical module MELCHIOR. Anthropogenic aerosol yields (ASOA) come from [Grosjean and Seinfeld, 1989], Moucheron and Milford (1996), [Odum et al., 1996], [Odum et al., 1997] and [Schell et al., 2001]. For biogenic secondary organic aerosols (BSOA), [Pankow et al., 2001] have proposed aerosol yields for terpene oxidation. In the model, precursor volatile organic compounds able to form secondary aerosol species are high chain alkanes, aromatics and monoterpenes. ASOA and BSOA are partitioned between gas and aerosol phases. Mass transfer is not only driven by the gas phase diffusion but also by the thermodynamic equilibrium through a temperature dependent partition coefficient ([Pankow, 1994]).

3.10 Clouds and impact on photolysis

In this version clouds are assumed to lie above the model top, so that there is no cloud albedo effect within the model domain depth. For all photolysed species, clear sky photolysis rates J_c(z) are multiplied throughout model columns by an **attenuation coefficient** A(d) depending on the **total cloud optical depth (COD)** d. Using the TUV model, and a large set of CODs for clouds at various altitudes, the attenuation relative to the clear-sky case has been fitted as a function of COD with the formula:

$$A(d) = e^{-0.11 d^{2/3}} \tag{3.4}$$

Several options are offered in order to calculate the COD. Total COD, d, is the sum of partial CODs from 3 cloud layers, low clouds d_l, medium clouds d_m and high clouds d_h. The limits between these cloud is user-chosen, but depend on the meteorological model. For MM5, limits of 2000m and 6000m are proposed. For each cloud layer three options are possible for the calculation of the partial cloud optical depth:

- Calculation as a function of Liquid/Ice Water Content in the column; using an assumption of sphericity and equivalent droplet size of 6 microns, an assumption of hexagonal shape for ice particles, the formula for cloud optical depth is 180.C_w + 67.C_i, where C_w and C_i are respectively the liquid water column (in Kg/m^2) and ice column for this cloud layer. This option is chosen by setting the cloud option to "1" in the **METINFOS.nml** file.
- Liquid/Ice water are generally unverified and unstable parameters in atmospheric models. Basing a cloud parameterization on such parameters can be risky. A more robust parameterization consists in using relative humidity only. The proposed parameterization (option "2" in **METINFOS.nml** file) consists in parameterizing the COD as a function of the integral **R**, over the cloud depth, of the relative humidity above 75%. It is assumed that small cloud formation (in particular cumulus clouds) starts at 75% relative humidity. Normalization of **R** leads to the formulation (for instance for low clouds): d_l = aR/dz, where a=0.02 is chosen such that a 1000m-thick layer has an optical cloud depth of 20.
- An even simpler parameterization (Option "0" in METINFOS.nml) can be achieved by making the COD simply proportional to the cloud fraction (if available) for each cloud layer. Coefficient tuning led to proportionality coefficients of: 50 for low-clouds, 10 for medium clouds and 2 for high clouds. This means, for instance, that a sky covered with 100% of high clouds has an optical depth of 2. Tuning was performed with ECMWF cloud fraction data and should change with the meteorological model.

Recommandations The first option using liquid/ice water (whenever available),is based on a variable which can be very model-dependent and which is rarely verified against observations. Moreover it does not take into account fractionary cloudiness. However it may be suited for high/medium clouds which are often stratiform clouds in the midlatitudes. The second option uses cloud fractions (whenever available), which is less meaningful to radiation but could be used when cloud/ice water is not available. The third option uses relative (to liquid water) humidity, and has been tuned using J(NO2) measurements taken over a long period near Paris (manuscript in preparation). The formulation, modifying radiation for RH$>$0.75 should only be used for low clouds. It has the advantage of being a more robust parameterization since it depends on a highly verified variable. In summary, our preference is: RH formulation (option 2 in METINFOS.nml) for low clouds, and for liquid/ice water formulation (whenever available) for medium and high clouds, and cloud fractions otherwise.
The attenuation coefficient is calculated in the **src/diagmet.f90** diagnostic interface model. It is read in **src/iniphys.f90**, photolysis rates are calculated in routine **src/photorates.f90** every coarse time-step.

3.11 Emissions

3.11.1 Anthropic emissions

CHIMERE requires input **Anthropic** emission data for 14 model species. These model species are:

- NO
- NO2
- SO2
- CO
- CH4[3]
- C2H6 Ethane
- NC4H10 n-Butane
- C2H4 Ethene
- C3H6 Propene
- C5H8 Isoprene
- OXYL o-Xylene
- HCHO Formaldehyde
- CH3CHO Acetaldehyde
- CH3COE Methyl ethyl Ketone

In the present version, emissions of Methanol and Ethanol are also included but are not used. They are only used in the **complete** version of the chemical mechanism (soon available). Annual data of anthropogenic emissions for the four classes NOx, SO2, CO and NMVOC are taken from the EMEP data base for 1998 ([Mylona, 1999], see also the EMEP web site (adress p.76). The data are spatially interpolated from the EMEP grid onto the CHIMERE grid.

The NMVOC emissions have to be split into 9 classes represented within the model chemical mechanism. To this end they are first distributed for each country into different activity sectors (traffic, solvents, industrial and residential combustion, others), according to data prepared by IER (Institute for Energy Economics and Rational Use of Energy, University of Stuttgart; GENEMIS, 1994). For each sector, NMVOC emissions are then split into 32 classes with similar structure and reactivity, following a classi2004033000.epsfication of [Middleton et al., 1990], and using standard NMVOC profiles again from IER [Society, 1994]. Third, NMVOCs from these 32 classes are aggregated into the 9 classes represented within CHIMERE, by applying mass and reactivity weighting as proposed by Middleton et al., 1990. It should be noted that the uncertainty in the spatial distribution of individual VOC emissions is large, given that the VOC profiles are assumed to be the same all over Europe, with the exception of traffic, where national differences (e.g. the ratio gasoline / diesel) are taken into account. Monthly, daily and hourly variations of the emissions are modeled by imposing respective variations from the [Society, 1994] data base.

In rural areas NO emissions from microbial processes may be an important source. Since these emissions strongly depend on temperature, they are processed in the model as "biogenic" emissions and often called in this manual "biogenic NO emissions". In soils NO is produced in a reaction chain of oxidation and reduction from ammonium which is used in fertilizers. CHIMERE uses a European inventory of soil NO emissions from [Stohl et al., 1996]. This inventory estimates the soil emissions to be during the summer months at the order of about 20% of the emissions from combustion on a European average but with large differences between the countries. In the model, these NO emissions are only considered during the months of May to August.

Anthropic emissions (except soil NO) are given in 12 files corresponding to each month (**EMISSIONS.**MM), MM being the calendar month. They are given for three typical days: Week day, Saturday, Sunday and for each UT hour. For some days weekday emissions are considered as a sunday (for holidays), but this assumption is often valid only for France (e.g. July 14 is considered as a sunday), although it is extended to Europe!

[3] Actually CH4 emissions are zero in this model version

3.11.2 Biogenic emissions

Biogenic emissions of isoprene and terpenes (affected to alpha-pinene in the chemical mechanism) are parameterized as fluxes $F_i = \epsilon_i D \gamma_i$ [Günther, 1997], with ϵ_i being the species dependent "emission potential" and D the "foliar density", both values taken from [Simpson, 1999].
The "environmental correction factor" γ_i accounts for the dependence on temperature and insolation (e.g. [Günther, 1997]). The spatial distribution of tree species is established following the methodology outlined in [Simpson, 1999]. Therefore the SEI (Stockholm Environment Institute) land cover data base, which details the fraction of different tree species over Europe, is interpolated on the CHIMERE grid. The 136 SEI land use classes are aggregated into 11 "emitter" classes. Oaks, for example are differentiated in the classes highly isoprene emitting, highly terpene emitting and low biogenic VOC emitting oaks. As this attribution of classes is sometimes ambiguous, national tree species inventories from [Simpson, 1999] are used to verify the fraction of each emitter class on a country-wise level, and to adjust it, if necessary.

3.12 Dry deposition

3.12.1 For gases :

Dry deposition is considered for model gas species i and is parameterized as a downward flux $F_{d,i} = -v_{d,i} c_i$ out of the lowest model layer with c_i being the concentration of species i. The deposition velocity is, as commonly, described through a resistance analogy ([Wesely, 1989]):

$$v_{d,i} = \frac{1}{R_a + R_{b,i} + R_{c,i}} \tag{3.5}$$

The aerodynamic resistance R_a is calculated as the integral of the inverse of the diffusivity coefficient K(z) up to the middle of the model surface layer, which can be estimated using the analytical formulae of the surface-layer similarity profiles for K ([Seinfeld and Pandis, 1997]). The formulae read:
Stable case

$$R_a = \frac{1}{k u_*} \left[ln\left(\frac{z}{z_0}\right) + 4.7(\zeta - \zeta_0) \right] \tag{3.6}$$

Neutral case

$$R_a = \frac{1}{k u_*} \left[ln\left(\frac{z}{z_0}\right) \right] \tag{3.7}$$

Unstable case

$$R_a = \frac{1}{k u_*} \left[ln\left(\frac{z}{z_0}\right) + ln\left(\frac{(\eta_0^2+1)(\eta_0+1)^2}{(\eta_r^2+1)(\eta_r+1)^2}\right) + 2\left(tan^{-1}\eta_r - tan^{-1}\eta_0\right) \right] \tag{3.8}$$

where $\eta_0 = (1 - 15\zeta_0)^{1/4}$, $\eta_r = (1 - 15\zeta_r)^{1/4}$ and $\zeta_0 = z_0/L$, **L** is the Obukov length, k the von Karman constant and z_0 is the dynamical roughness length. z_0 depends on the fraction of land-use for each category and on the season. See Section on land-use and on meteorology for more details.
According to [Hicks et al., 1987] the quasi laminar resistance R_b, i is calculated as:

$$R_{b,i} = \frac{2}{k u_*} \left(\frac{Sc_i}{Pr}\right)^{2/3} \tag{3.9}$$

where Sc_i and P_r are the Schmidt and Prandtl number, respectively, k the von Karman's constant and u_* the friction velocity.
The formulation of the surface resistance R_c follows [Erisman et al., 1994]. It uses a number of different other resistances accounting mainly for stomatal and surface processes which are again dependent on the land use type and season. Necessary chemical parameters for the calculation of R_c are also taken from [Erisman et al., 1994] except for carbonyls ([Sander et al., 1999]; [Baer and Nester, 1992]) and peroxide species ([Hall et al., 1999]). All species-related parameters are gathered in the DEPO_SPEC file located in all data_[model]_[meca] directories.
The main land/seasonal parameters can be found in the file DEPO_LAND file which can be found in all data_[model]_[meca] directories. Seasonal variations of resistances are taken from 3 Wesely Seasons:

- #1 (summer with full vegetation), from may to october. In August, September and October, however, LAI, stomatal resistance and deposition height are divided by two for the CROPLAND category in order to take into account harvested fields.
- #3 (winter without frost) from november through march.
- #5 (transitional season with short annuals) for April.

Most land parameters are taken from [Wesely, 1989], but LAI are drawn from the NASA/EOSDIS Oak Ridge National Laboratory (adress p.76) using average LAI field measurements, for Summer.
Dry deposition (for gases) parameters are initialized in routine **src/inidepo.f90**. At the coarse time step frequency, deposition velocities are calculated in routine **src/depvel.f90**. Deposition fluxes are calculated at the fine time step frequency in routine **src/deposition.f90**.

3.12.2 For aerosols:

Dry deposition for aerosols also makes use of a resistance scheme. The dry deposition velocity is expressed as :

$$v_d = v_s + \frac{1}{r_a + r_b + r_a r_b v_s} \tag{3.10}$$

where v_s is the sedimentation velocity expressed as in [Seinfeld and Pandis, 1997], R_a the aerodynamic resistance and R_b the quasi-laminar resistance for particles.

3.13 Wet scavenging

When the aerosol option is selected, scavenging for gas/aerosols in clouds or rain droplets is taken into account. Wet scavenging is formulated as follows:

- For **gases** in **clouds**: Nitric acid, ammonia in the gas phase are scavenged by cloud droplets. This process is assumed to be revertible. Moreover, for in-cloud scavenging, dissolved gases in a non precipitating cloud can reappear in the gas phase due to cloud dissipation. Equilibrium between dissolved gases concentration and gas-phase concentrations follow Seinfeld and Pandis (1998).
- For **gases** in **rain droplets below the clouds**: Dissolution of gases in precipitating drops is assumed to be irrevertible, both for HNO_3 and NH_3. The scavenging coefficient is expressed as:

$$\Gamma = \frac{pD_g}{6.10^5 u_g D^2}(2 + 0.6R_e^{1/2}S_c^{1/3}) \tag{3.11}$$

p being the precipitation rate (mm h^{-1}), D_g the molecular diffusion coefficient ($m^2 s^{-1}$), u_g the raindrop velocity (m s^{-1}), Re and Sc respectively the Reynolds and Schmidt numbers of drops. [Mircea and Stefan, 1998] and references therein give relationships between u_g and hydrometeor diameter for various types of precipitation. In the model, sulfur dioxide and hydrogen peroxide are also scavenged by precipitation.

- For **particles** in **clouds** : Particles can be scavenged either by coagulation with cloud droplets or by precipitating drops. Particles also act as cloud condensation nuclei to form new droplets. This latter process of nucleation is the most efficient one in clouds. According to [Tsyro, 2002] and [Guelle et al., 1998], the deposition flux is written as:

$$\left[\frac{dQ_l^k}{dt}\right] = -\frac{\epsilon_l P_r}{w_l h} Q_l^k \tag{3.12}$$

with p_r being the precipitation rate released in the grid cell (g cm^{-2} s^{-1}), w_l the liquid water content (g cm^{-3}), h the cell thickness (cm) and eipsilon an empirical uptake coefficient (in the range 0 - 1) depending on particle composition. l and k are respectively the bin and composition subscripts.

- For **particles** in **rain droplets below the clouds** : Particles are scavenged by raining drops, the deposition flux of particles being:

$$\left[\frac{dQ_l^k}{dt}\right] = -\frac{\alpha p E_l}{u_g} Q_l^k \tag{3.13}$$

with a being an empirical coefficient, p the precipitation rate in the grid cell (g cm^{-2} s^{-1}), E a collision efficiency coefficient between particles and raining drops ([Loosmore and Cederwall, 2004]) and u_g the falling drop velocity (cm s^{-1}).

3.14 Time numerical solver

The numerical method for the temporal solution of the stiff system of partial differential equations is adapted from the second-order TWOSTEP algorithm originally proposed by [Verwer, 1994] for gas phase chemistry only. It is based on the application of a Gauss-Seidel iteration scheme to the 2-step implicit backward differentiation (BDF2) formula:

$$c^{n+1} = \frac{4}{3}c^n - \frac{1}{3}c^{n-1} + \frac{2}{3}\Delta t R(c^{n+1}), \tag{3.14}$$

with c^n being the vector of chemical concentrations at time t_n, Δt the time step leading from time t_n to $t_n + 1$ and $R(c) = \dot{c} = P(c) - L(c)c$ and $R(c) = \dot{c} = P(c) - L(c)c$"> the temporal evolution of the concentrations due to chemical production and emissions (P) and chemical loss and deposition (L". Note that L is a diagonal matrix here. After rearranging and introducing the production and loss terms this equation reads

$$c^{n+1} = \left(I + \frac{2}{3}\Delta t L(c^{n+1})\right)^{-1} \left(\frac{4}{3}c^n - \frac{1}{3}c^{n-1} + \frac{2}{3}\Delta t P(c^{n+1})\right). \tag{3.15}$$

The implicit nonlinear system obtained in this scheme can be solved pertinently with a Gauss-Seidel method (Verwer, 1994).

In CHIMERE the production and loss terms P and L in equation 3.15 are replaced by the modified terms $\tilde{P} = P + P_h + P_v$ and $\tilde{L} = L + L_h + L_v$, respectively. P_h and P_v denote the temporal evolution of the

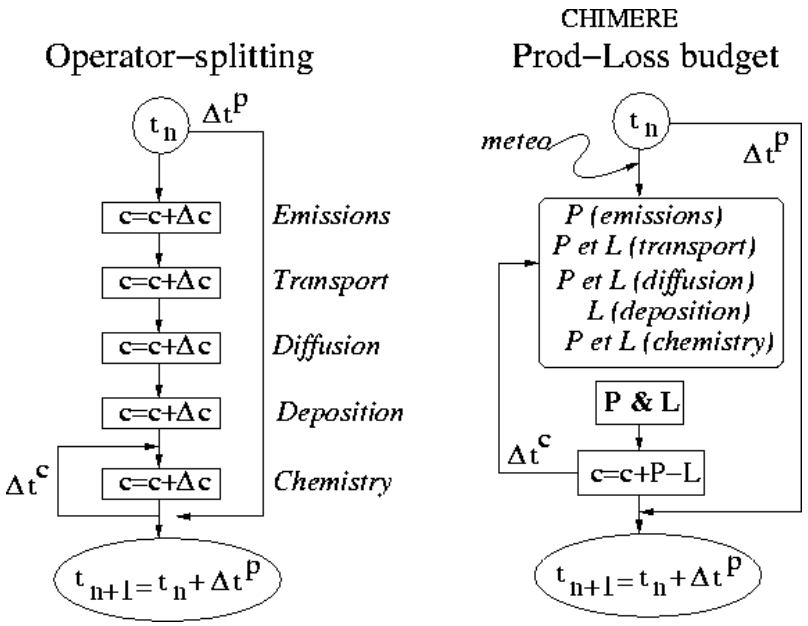


Figure 3.2: *Principle of 'operator-splitting' versus chimere integration*

concentrations due to horizontal (only advection) and vertical (advection and diffusion) inflow into the concerned grid box,
L_h and L_v the temporal evolution due to the respective outflow divided by the concentration itself. With this integration scheme, at each time step all physical and chemical processes are updated simultaneously. The "operator splitting" technique (see e.g. [McRae et al., 1982], which is still a standard way of solving the 3D transport-chemistry-problem, and the associated "splitting error" are therefore avoided. Further advantages of the scheme are its stability even for quite long time steps due to the implicitness of the formulation and the simplicity of the code which facilitates the development of secondary models (adjoint, tangent linear) enormously.
In practice, time stepping has two time steps: A **coarse time step**, where all physical variables are calculated, and a **fine step** (refined step in order to gain accuracy on chemistry. During loops over fine steps, physical variables are maintained constant.
In the provided version for the continental models, we chose a "very quick formulation", with a 10-minute physical step, no sub-chemical steps, i.e. all processes are stepped to 10 minutes, with only 1 Gauss-Seidel iteration. **It is STRONGLY recommanded**, in order to have more accurate results, to use 2 Gauss-Seidel iterations, which increases by 2 the computer time. This change affects mostly concentrations during morning. However, for values of ozone concentration, the proposed version does not appear to degrade simulations, especially during afternoon hours.
The TWOSTEP formulae and loops are performed in routine **src/twostep.f90**.

Chapter 4

First CHIMERE simulation

In the case of a user who never used the CHIMERE model before, this chapter presents how to:

- Install all softwares prerequisites, §4.1
- Perform a simulation with the pre-defined test case available on the CHIMERE web site, §4.2.2
- See the results, §4.3

4.1 Main sources installation

4.1.1 Software prerequisites

The CHIMERE model has only been tested on GNU/Linux systems. However it should be working on most UNIX systems provided the following software is installed. The foreseenable changes should be related to **shell**, **awk** and **make** syntax, and also to unformatted binary files which may have to be converted.
The model requires :

- a **Fortran 95** compiler
- GNU **bash** Bourne shell
- GNU **awk**
- GNU **make**
- Unidata **netCDF** library (free)
- Unidata **Udunits** library (free)
- The Generic Mapping Tools **GMT** (free)
- **GrADS** graphical software (free)

NB: The corresponding adresses of the related web sites are reported p.76.
The installation process of netCDF and Udunits is fully described in **Appendix B, p.81** and **Appendix C, p.85**, and in files **HOWTO-INSTALL-NETCDF** and **HOWTO-INSTALL-UDUNITS** in the top level CHIMERE directory. It is mandatory that the netCDF library be compiled with the same compiler that will be used to build the CHIMERE model.
If no **bash** is installed, the model may work with a baseline Bourne-shell, but the user may have to edit the scripts to take in account some syntactic features specific to **bash** . The same remark apply to the **awk** and **make** utilities.
The model has been tested with the following compilers :

- g95
- ifort
- pgf90

Udunits, GMT and GraDS are only required if the user wants to use the post-processing stages proposed in the **toolbox/** directory, i.e. **ncmapit** and **exachim** or **exadiff**.
In the following examples, the **wget** utility is used for downloading, because it is a robust and powerful downloading tool. However, if **wget** is not installed on your system, you can obviously use your favorite browser to download files.

4.2 Start CHIMERE with the 2003 HEAT-WAVE test case

The real test case with a simulation over Europe of a 5-day period starts on July 30 2003 and ends on August 3 2003, i.e. the beginning of the 2003 heat wave, with MM5 forcing. The MM5 run is forced by the 6-hourly NCEP/AVN analyses, and the "nudging" option (grid FDDA) is chosen, with coefficients of 1e-4 for temperature, wind, and 2e-5 for humidity. The model domain covers Western Europe with an approximate 36km horizontal resolution and 32 vertical levels.

4.2.1 Download required data and programs

First, it is highly recommended to run the test case proposed on the web site. To perform this, you need:

- The CHIMERE model itself in its version V200511B
- An already prepared MM5 output file, provided for this time period.
- The CONT3-2002, INCA-200501 and GOCART-200410 emissions and boundary conditions data

4.2.1.1 The CHIMERE model

Choose a directory to store the model code and data. You need about 160 Mo of free disk space. Let us assume your model directory is **/mycodedir**

```
cd /mycodedir
wget http://euler.lmd.polytechnique.fr/chimere/downloads/V200511B.tar.gz
```

4.2.1.2 The test case MM5 meteo file

Choose a directory to store the example MM5 meteo data. You need about 900 Mo of free disk space.

```
cd /mymeteodir
wget http://euler.lmd.polytechnique.fr/\
    chimere/downloads/MMOUT_EUR1_20030730_20030803_S.gz
```

4.2.1.3 The emissions data

Choose a directory to store emissions data. You need about 124 Mo of free disk space for the zipped archive.
Let us assume this directory is **/myemisdir**

```
cd /myemisdir
wget http://euler.lmd.polytechnique.fr/chimere/downloads/CONT3-2002.tar.gz
```

For the use of these emissions data, please acknowledge EMEP (Yearly totals), IER (Time variations), TNO (Aerosol emissions), UK Dept of Environment (VOC speciation, Passant, 2002).

4.2.1.4 The boundary conditions data

Choose a directory to store boundary conditions data. You need about 550 Mo of free disk space. Let us assume this directory is **/mybounddir**

```
cd /mybounddir
wget http://euler.lmd.polytechnique.fr/chimere/downloads/INCA-200501.tar.gz
wget http://euler.lmd.polytechnique.fr/chimere/downloads/MOZART2-200303.tar.gz
wget http://euler.lmd.polytechnique.fr/chimere/downloads/GOCART-200410.tar.gz
```

These data were kindly provided by colleagues, thus please acknowledge D. Hauglustaine and S. Szopa (Institut Pierre-Simon Laplace) for INCA data, M. Schultz (MPI Hamburg), C. Granier, G.Brasseur and D. Niehl for MOZART2 data and M. Chin, P. Ginoux (NASA) for GOCART data. Please contact sophie.szopa@cea.fr and mian.chin@nasa.gov for any question about INCA and GOCART, respectively.

4.2.2 Simulation configuration

Once the code is downloaded, the next step is to set up some configuration files to match your system. To achive this, edit the top calling script **chimere-mm5** and adapt the paths. A copy of this file is given in the **Appendix H, p.107**. Then you will be ready to run the test case by launching the command line: `./chimere-mm5`. CHIMERE uses configuration files that have to be tailored to user's requirements and environment. These files reside in **V200511B** top directory:

makefile.hdr is a symbolic link to a preformated file, specific to a given compiler. It contains all the informations pertinent to your execution environment: compilers options, libraries, and so on. makefile.hdr is included by all subsequent makefiles in CHIMERE source tree, to ensure a consistent compilation.

chimere-mm5 is the top calling script of CHIMERE. It contains all the information necessary to run CHIMERE: data base files locations, domain definition, physical and numerical options. **chimere-mm5** sources an execution script, **script_functions/chimere-mm5.exec.sh**, which in turn launches all CHIMERE processes.

Only experienced users should try to modify **script_functions/chimere-mm5.exec.sh**.

To configure CHIMERE:

- Link Makefile.hdr to your compiler specific makefile header. For instance:

```
ln -sf makefiles.hdr/Makefile.hdr.ifort Makefile.hdr
```

- Edit Makefile.hdr for :
  - ☐ the location of your compiler
  - ☐ the location of your netCDF package
  - ☐ the location of your Udunits package, if you plan to use **exachim** to display your results
- Edit chimere-mm5 to :
  - ☐ set up the time span of a run
  - ☐ tell CHIMERE the location of pertinent files and directories :
    - ☐ domain geometry

- ☐ species list
- ☐ emission files
- ☐ boundary conditions files
- ☐ meteo file
- ☐ working directory
- ☐ output directory

☐ choose model options

If you are just running the test case, you only need to customize directories and files locations depending on where you stored them after download.
Now just execute the top-calling script using the command:

```
chimere-mm5
```

and see the messages on the screen to check that all installations were correctly done.

4.3 Outputs

Outputs on screen are described below and may be easily change by the user. If you are running the proposed test case called 'heat wave', you must have on your screen the outputs as below:

```
Temporary Directory : /tmp
Compilation of all programs...
diagmet.f90:
chimere.f90:
Preparing anthropic emissions...
prepemis_std.f90:
+ Preparing lateral boundary concentrations...
+ This is a COARSE run, lateral B. C. are taken from: LMDZINCA2
preplat_LMDZINCA2.f90:
 + Definition of CHIMERE lateral grid boundaries
 + Horizontal interpolation for the lateral boundaries
+ Preparing top boundary concentrations...
+ This is NOT a nest run, top B. C. are taken from: LMDZINCA2
preptop_LMDZINCA2.f90:
 + Horizontal interpolation for the top boundaries
* Initialization file empty, interpolation of B. C. used instead
Building the INFOS / chimere.h file...
mm5 interface running...
 Date= 2003072919 ; Central 2m Temperature=  15.2764893
 Date= 2003072920 ; Central 2m Temperature=  14.0648499
 Date= 2003072921 ; Central 2m Temperature=  13.3481445

 .... etc.

 Date= 2003080321 ; Central 2m Temperature=  19.6983337
 Date= 2003080322 ; Central 2m Temperature=  19.1453552
 Date= 2003080323 ; Central 2m Temperature=  18.7027893
 Date= 2003080400 ; Central 2m Temperature=  18.4093323
+ Constructing weights for regridding MM5 domain MM5EUR1 to CHIMERE domain CONT3
Meteo Horizontal Interpolation...
Meteo diagnostic model...
2003073000  0.119  0.124 -0.009  0.000   320.  0.725   4.94   0.00   0.00    12.
2003073001  0.143  0.125 -0.010  0.000  1000.  0.457  18.68   0.00   0.00  1000.
2003073002  0.149  0.124 -0.009  0.000  1000.  0.266  40.42   0.56   0.00  1000.
2003073003  0.221  0.125 -0.003  0.000   977.  0.193  53.00   3.80   0.00  1000.

 .... etc.

2003080320  0.134  0.143 -0.013  0.000   254.  0.964   0.00   0.00   0.20    -1.
2003080321  0.137  0.142 -0.013  0.000   222.  0.976   0.00   0.00   0.10    -1.
2003080322  0.138  0.141 -0.013  0.000   206.  0.989   0.00   0.00   0.03     1.
2003080323  0.137  0.139 -0.012  0.000   195.  0.993   0.00   0.00   0.02     2.
2003080400  0.132  0.137 -0.012  0.000   187.  0.984   0.02   0.00   0.03     4.
Model running...
 SUMMARY...
 Number of components                                7
 Number of sections                                  1
 Number of primary species                           2
 Number of Semi volatile organic species             1
 Number of Semi volatile inorganic species           3
 * PPM_SPECIES: species NH3            ignored
```

```
 * PPM_SPECIES: species HCL             ignored
 * PPM_SPECIES: species NAAQ            ignored

 .... etc.

 * PPM_SPECIES: species p6HNO3          ignored
 * PPM_SPECIES: species p6NH3           ignored
 * PPM_SPECIES: species p6HCL           ignored
 * PPM_SPECIES: species p6WATER         ignored
 * DEPO_SPEC species NH3             ignored
 * DEPO_SPEC species HCL             ignored
 * WETD_SPEC: species NH3             ignored
 * WETD_SPEC: species HCL             ignored
 * WETD_SPEC: species H2SO4AQ         ignored
 * WETD_SPEC: species HNO3AQ          ignored
 * WETD_SPEC: species SOAAQ           ignored
 * WETD_SPEC: species NH3AQ           ignored
 * WETD_SPEC: species PPMAQ           ignored
 * WETD_SPEC: species DUSTAQ          ignored
 * WETD_SPEC: species SALTAQ          ignored
 * WETD_SPEC: species NAAQ            ignored
 * WETD_SPEC: species HCLAQ           ignored
 * INIEMIS: Emitted Anth. species PPM_big         ignored
 * INIEMIS: Emitted Anth. species PPM_coa         ignored
 * INIEMIS: Emitted Anth. species PPM_fin         ignored
 * INIEMIS: Emitted Anth. species NH3             ignored
 * INIEMIS: Emitted Biog. species DUST_big        ignored
 * INIEMIS: Emitted Biog. species DUST_coa        ignored
 * INIEMIS: Emitted Biog. species DUST_fin        ignored
 * INIEMIS: Emitted Biog. species SALT_coa        ignored
 * INIEMIS: Emitted Biog. species NA_coa          ignored
 * INIEMIS: Emitted Biog. species HCL_coa         ignored
 * INIEMIS: Emitted Biog. species H2SO4_coa       ignored
 * INIEMIS: Emitted Biog. species WATER_coa       ignored
DATE            O3     NO2 Temp (2m) Surf Wind  PBL Hght  Rad Atte     Kz(1)
2003073000    33.4     1.3      15.6       3.7      377.      0.70      5.00      5.00
2003073001    33.2     3.5      15.5       3.7      943.      0.48      5.00      5.00
2003073002    30.9     5.7      15.3       3.7     1000.      0.28      5.00      5.00
2003073003    28.8     7.6      16.5       3.6      979.      0.20      4.82      4.82

 .... etc.

2003080321    48.1     9.6      21.3       4.3      225.      0.98      0.10      0.10
2003080322    48.5     8.0      20.6       4.3      207.      0.99      0.55      0.39
2003080323    54.1     5.4      20.0       4.2      196.      0.99      1.45      0.87
2003080400    55.9     4.1      19.4       4.2      188.      0.98      2.40      1.28
```

4.3.1 The output on screen

Outputs on screen are predefined in the routine **outprint.f90** to give some informations about the run. The choices of outputs are:

- The current date in format `yyyymmddhh`
- The first two species defined in the **OUTPUT_SPECIES** file
- The 2m temperature (in Kelvin)
- The surface wind speed (in m/s) (in fact, this value correspond to the first vertical levels wind speed, typically 30-40m AGL).
- The boundary layer height (in meters) diagnosed using the **diagmet.f** pre-processor
- The radiative attenuation: from 0 to 1 with 1 for the complete clear sky.
- The K_z diffusion coefficient at levels 1 and 2 (m^2/s^{-2})

All these values are given for the prescribed **nsho** value. This value may be changed in the **top calling script** or directly forced in the **src/chimere_params.f90.sed** file. An easy way to change these outputs is to modify the **OUTPUT_SPECIES** file. The two first species (in ppb) will appear on screen during the simulation. A more complicated way is, of course, to change directly the fortran source, at the end of the subroutine.

4.3.2 The output files

Four different output files are delivered. They are encoded in the netCDF format, which presents the following advantages :

- portability among different architectures of computers
- self-documentation since a lot of meta-data are included in the file itself
- direct access
- compatibility with many free or commercial post-processing tools

Moreover, CHIMERE uses a netCDF convention (see **Appendix E, p.89**) derived from the one used in NCAR'S WRF mesocscale modelling system, simplifying the toolset for post-processing.

4.3.2.1 The chemical fields outputs on file "out.[label].nc"

Some three-dimensional chemical fields are written in an output *binary* file called **out.[label].nc**. The selected species correspond to those listed in the **OUTPUT_SPECIES** parameter file. Default values are expressed in ppb. Output values may be transformed into μg.m^{-3} if the molar mass of each species is added as a second column in the parameter files **/data/output_species/OUTPUT_SPECIES...**. The default value is zero. These potential transformations are managed in the routines **src/inichem.f90** and **src/outprint.f90**. The user is encouraged to have a look to the routine **toolbox/readcdf/readcdf.f90** to learn how to read the model output files.
example: In the file **/data/output_species/OUTPUT_SPECIES...**: If the line is "`O3 0`", ozone results will be in ppb. If the line is "`O3 48`", the results will be in μg.m^{-3}.

4.3.2.2 The deposition fields outputs on file "dep.[label].nc"

This file contains fluxes integrated over the chemical time-step duration. The calculations are done in **src/twostep.f90** and written in files in **src/write_depo.f90**. For the dry deposition flux, ony the first vertical cell is taken into account. On the other hand, for the wet deposition, fluxes are sumed up the whole column since, in the model, each vertical level may contribute to a net sink. Units of these two integrated fluxes are g/cm^2.

4.3.2.3 The restart fields outputs on file "end.[label].nc"

The structure of this file is described in **Appendix F.7, p.96**. The main goal of this file is to save all concentrations fields every 24h. For example, for a second run following in time the first one, this "end..." file will be link to a "ini..." file and thus used to give realistic initial conditions during the restart.

4.3.2.4 The additional fields outputs on file "par.[label].nc"

An additional output file is named **par.[label].nc**. Its goal is to provide meteorological outputs, corresponding to the values employed during the run to calculate the concentration fields written in **out.[label].nc**. The extracted parameters are hardcoded in the routine **src/iniparams.f90**. The possible values are displayed in the **Table 4.1**.

Model Parameter	Variable
wind	
iwinhs	Wind speed $\|U\|$ (m/s 1st vertical level)1
iwinh2	Wind speed $\|U\|$ (m/s 2nd vertical level)1
iwinvs	Vertical wind speed (m/s 1st vertical level)
iwinvt	Vertical wind speed (m/s top domain vertical level)
specific humidity, air density and temperature	
isphu	Specific humidity
iairm	Air density
itemp	Temperature (K)
iwinxs	Velocity-equivalent of mixing exchange rate (surface)
iwinxt	Velocity-equivalent of mixing exchange rate (top domain)
ithlay	Thickness of the model layer (in cm)
mixing height, radiation	
ihght	Mixing height (in meters)
iatte	Radiation attenuation [0-1]
izeni	zenithal angle (degrees)
item2	2m temperature (K)
diagnostic turbulent parameters	
iusta	Friction velocity u_* (m/s)
iaerr	Aerodynamical resistance
iobuk	Monin-Oblukov lengtth
iwsta	Convective velocity w_*
idepo	Dry deposition velocity

Table 4.1: *The parameters model outputs on file.*

4.4 Post-processing the results

We provide here some indications to help the user for displaying and processing CHIMERE raw results. The complexity of post-processing depends obviously on the user intents, and the tools proposed here should only be considered as helpers or templates for developping more sophisticated programs.

Apart some configuration files and static data files written in ASCII code, all the input and output files of the model are written in the netCDF format, using conventions similar to the WRF conventions. Thus the tools described below may be used to process almost any CHIMERE netCDF file, including meteo files. CHIMERE netCDF files are suffixed with **.nc** .

Once you have succesfully performed a CHIMERE run, you may want to :

- Dump data in ASCII form
- Use data for further numerical processing
- Plot data
- Interactively display data

Guidelines are proposed in the following paragraphs.

4.4.1 Dump data in ASCII form

If you have netCDF installed, and you do have it since you were able to run this version of CHIMERE, just use the **ncdump** utility to dump the content of any of CHIMERE netCDF file :

- `ncdump -h <ncfile>` prints the file header
- `ncdump -c <ncfile>` prints the header, plus the values of so-called "coordinate variables"
- `ncdump -b f <ncfile>` prints the whole content of the file, with fortran indexes.
- `ncdump -v <myvar> -b f <ncfile>` prints the header, followed by the selected variable

For more details, read the `ncdump` man page :

```
man ncdump
```

4.4.2 Use data for further numerical processing

Brute force method :
Redirect the output of ncdump to a file, and open this file with your Fortran code. This method is NOT recommended, since :

- it produces large ASCII files, with inefficient sequential access
- you loose numerical accuracy

Transform CHIMERE netCDF file into a COARDS compliant netCDF file, and open it with GrADS, Ferret, IDL, or any netCDF-COARDS aware post-processing tool you are familiar with. See §4.4.4 (p.44) to get a COARDS compliant file.

Download the netCDF Fortran 90 manual from
http://my.unidata.ucar.edu/content/software/netcdf/index.html. Read, read again and understand this manual. Code by yourself some simple examples. Have a look at the netCDF code in **toolbox/** directory, for instance **toolbox/readcdf/readcdf.f90**. Write your own interface to get CHIMERE data into an array in your Fortran 90 code. This is the right method, with a guaranteed Return On (intellectual) Investment.

4.4.3 Plot data

The **toolbox/ncmapit/ncmapit.sh** script takes as input a CHIMERE output file (end.*, out.*, dep.*, par.*) or the meteo file METEO.nc, and :

- launches **ghostview** to display the selected field
- produces a "publication-ready" PostScript file with Chimere-standard color palette.

ncmapit.sh is the netCDF counterpart of the former **mapit** shell script proposed in earlier versions of CHIMERE.
ncmapit.sh provides a way to display many CHIMERE netCDF files, including OUTPUT and METEO. **ncmapit.sh** tries to use the standard palettes contained in the directory **toolbox/ncmapit/palettes**. If a palette is not available for a species, then **ncmapit.sh** generates it, with rounded thresholds, tailored to the actual scale of the field to be displayed.
In details, **ncmapit.sh** :

- takes a CHIMERE file as input,

- launches a Fortran program, chimere2gmt, which :
  - ☐ scans the CHIMERE file to compute range information and palette data
  - ☐ generates a header file containing the required metadata
  - ☐ generates an ASCII file containing a list of X, Y, VAL triplets passed to the Generic Mapping Tools (GMT, http://gmt.soest.hawaii.edu/)
- spawns GMT to produce a PostScript file suitable for display or printed publication.
- spawns a GhostScript (www.cs.wisc.edu/ ghost/) window to display this PostScript.

You can tailor the script to :

- modify the style of the figure. See GMT documentation.
- keep the PostScript file (erased by default)

ncmapit.sh cannot read AEMISSONS, BEMISSIONS, and Boundary conditions files due to the very specific format of these files.

Usage :

- 1st : build chimere2gmt.e
  ```
  make
  ```
- 2nd : run ncmapit.sh
  ```
  ./ncmapit.sh <chimere_file> <variable> <date> <level> <map_file.ps>
  ```

Where

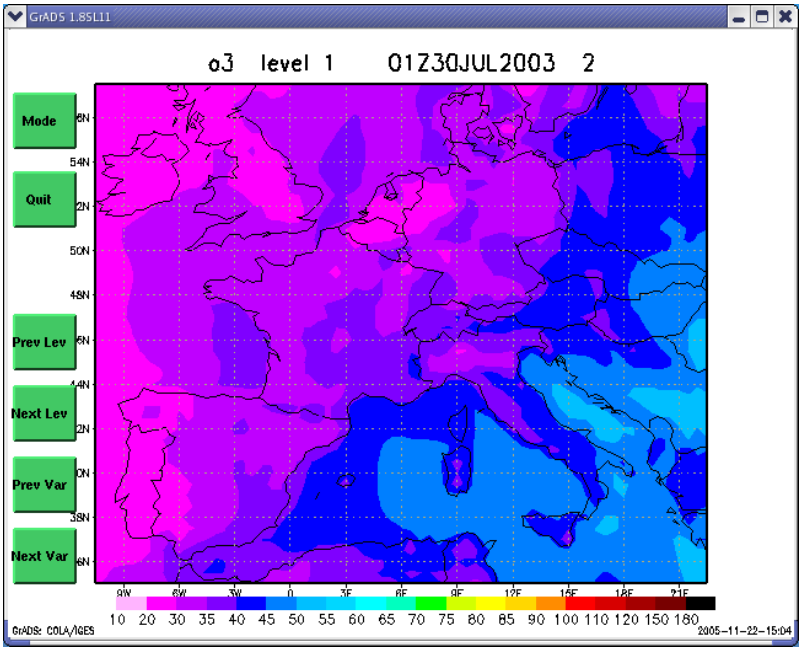
- <chimere_file> : your CHIMERE output file : dep, end, out or par
- <variable> : the name of the variable you want to display, like O3. Must be among the **OUTPUT_SPECIES** list.
- <date> : the date/time in chimere format, like 2004122612
- <level> : the index of the layer you want to display. For 2D fields, set level=1
- <map_file.ps> : the name you want to give to the postscript output file this script will generate

In the case of a 2D field, set level=1 in <args>.

Predefined palette files for some pollutants are given in the **palettes** subdirectory. For some other pollutants the user must define a new palette file by using the GMT command **makecpt**. The predefined palettes are described in **Appendix D, p.87**.

4.4.4 Interactively display data

Exachim and examout are (maybe) what you are looking for.
exachim.sh takes any CHIMERE netCDF file (at the exception of boundary condition and emission files), transforms it into a COARDS compliant file, and launches a GrADS script to display the fields. Once in the display window, you can interactively, by simple mouse clicks, change the time slot, variable, level, or print the value of the variable for a specific location. The user can also focus on a particular field, and by point clicking on the image, get the values of the triplet X, Y, VAL
Moreover, **exachim.sh** can keep its intermediate netCDF COARDS file. This is useful if you plan to post-process the data with COARDS-aware tools (such as the free and well-known Ferret and GrADS).



examout.sh has the same functions as **exachim.sh**, but it uses the standard CHIMERE palettes. Thus its use is restricted to "out." files.
While **exachim.sh** may work on almost any netCDF CHIMERE file, **examout.sh** is designed for use with "out" files. It includes prebuilt palettes which are now a de facto standard in CHIMERE, thus making the display familiar to those users who were using the former **mapit** display tool.

In details, **exachim.sh** and **examout.sh** :

- take as input a CHIMERE file, containing 2D and/or 3D fields.
- launch a Fortran program, **chm2coards**, to transform the CHIMERE file into a COARDS-compliant file.
- launch a **GrADS** (http://grads.iges.org) script, **exachim.gs** (examout.gs), to generate the interactive display.

exachim.sh and **examout.sh** cannot read AEMISSONS, BEMISSIONS, and Boundary conditions files due to the very specific format of these files.

Usage (same usage for exachim.sh)

- 1st : build chm2coards
  ```
  make
  ```
- 2nd : run ./exachim.sh
  ```
  exachim.sh [-k] <netcdf_file>
  ```

If **-k** is set, then the intermediate netCDF-COARDS file will be kept. It allows for further processing with other COARDS compliant, more sophisticated analysis tools. The output file is automatically named : **tmp.NNNN.nc**, with NNNN=some_unique_integer_value.
It is up to you to rename it to a more meaningful name.

The display window has two modes :

Browse . You are in this mode when you start.

- a left-click on the picture increases the time slot index
- a right-click decreases the time slot. If you click fast enough, you can simulate a movie loop. The palette remains constant when you loop along time axis. It may vary when you loop along level axis.
- you can change of variable or level by clicking on one of the four bottom-left buttons.
- if you click on the "Mode" button, then you enter the "Data" mode

Data . The “Mode” button appears to be pushed.

- a click on the picture prints on the xterm window :
  - ☐ the name of the variable
  - ☐ its value at the selected grid point
  - ☐ the coordinates of the grid point
- clicks on the Level or Var buttons are ineffective.
- a click on "Mode" resets the display in "Browse" mode.

A click on **Quit** exits the application.

4.5 Download the complete available CHIMERE database

If you want to use the model for some other studies, you would need some additional data files. They are listed below and at this address:
http://euler.lmd.polytechnique.fr/chimere/download.html

4.5.1 download emissions data

Choose a directory to store emissions data. You need about 850 Mo of free disk space. Let us assume this directory is **/myemisdir**

```
cd /myemisdir
wget http://euler.lmd.polytechnique.fr/chimere/downloads/CONT3-2001.tar.gz
wget http://euler.lmd.polytechnique.fr/chimere/downloads/CONT3-1999.tar.gz
wget http://euler.lmd.polytechnique.fr/chimere/downloads/FRA10-2002.tar.gz
wget http://euler.lmd.polytechnique.fr/chimere/downloads/AemiIDF-200302.tar.gz
```

For these data, please acknowledge EMEP (Yearly totals), IER (Time variations), TNO (Aerosol emissions), UK Dept of Environment (VOC speciation, Passant, 2002). for the CONT3 emissions; AIRPARIF, the monitoring association in the Ile-de-France area to which data belong for the Paris area emissions

4.5.2 download prepdomains interface

You may want to define new model domains. In this case, download the prepdomains preprocessor, which allows for the construction of landuse files and biogenic emission potentials. Please acknowledge the University of Maryland and the Global Land Cover Facility (GLCF) in any publication using the data. The directories below allow construction of files for model domains in Europe only. See the technical documentation on landuse preprocessing. The URL is :

```
http://euler.lmd.polytechnique.fr/\
    chimere/downloads/prepdomains-200310.tar.gz
```

4.5.3 download EMEP emissions interface

Our interface for the calculation of CHIMERE (MELCHIOR) emissions from EMEP annual emission data is now available. It can be used in principle for any new domain in Europe, for the construction of emissions corresponding to the MELCHIOR primary species. The archive contains a short README file giving instructions on how to proceed to build standard emissions for CHIMERE from EMEP data. Beware that the initial emissions are of resolution of 50kmx50km, and please refer to Vestreng (2003: Review and revision of Emission data reported to CLRTAP, EMEP status report). For other chemical mechanisms please contact us for more details about what to change in the emission building code. The URL is:

```
http://euler.lmd.polytechnique.fr/\
    chimere/downloads/emiEMEP-200505.tar.gz
```

Chapter 5

The CHIMERE model

The CHIMERE model consists of two separate modules. The first one is the meteorological interface (in the `meteo/` directory). It prepares the meteorological data in the CHIMERE format, but also diagnosed turbulent parameters, calculate biogenic emissions (meteorologically dependent). The second one is the CHIMERE code itself (in the `src/` directory).
The general structure contains a central chemistry-transport calculation unit with interfaces to several databases (emissions, meteorology, boundary conditions, land use, as shown in **Figure 5.1**. Each interface transforms original or processed data to CHIMERE-input formatted data adapted to the time period and region to simulate.

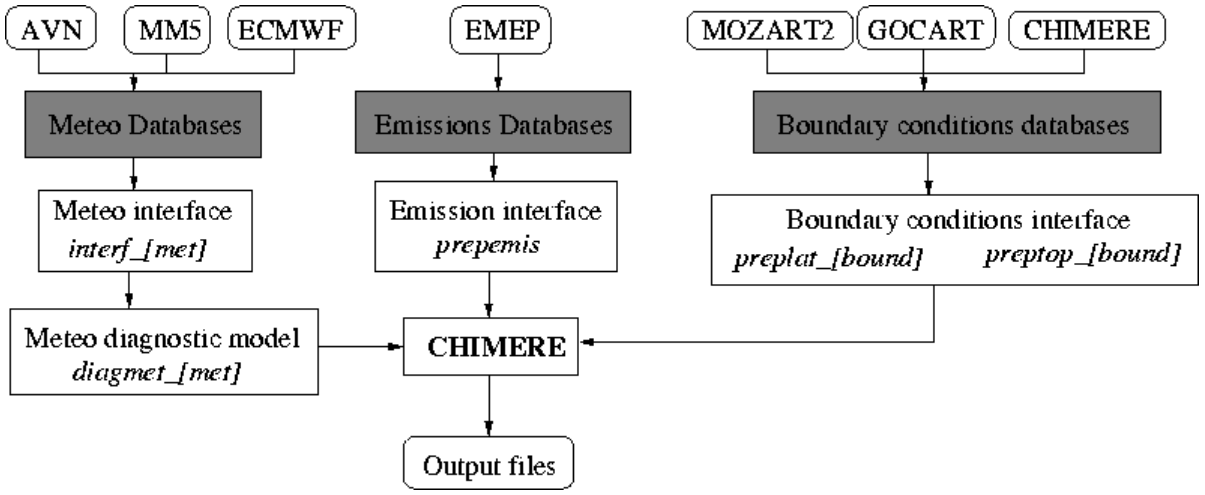


Figure 5.1: *Schematic structure of the CHIMERE input data*

The most complex interface is the meteorological interface because it uses two steps. The first one (**meteo**) transforms original meteorological data (standard variables) given on the meteo model grid, at a frequency that is not necessarily one hour, to the same variables but on the CHIMERE grid (space interpolation) at the hourly sampling rate (time interpolation) in a specific file format. An example (the **interf-mm5** interface) is provided (in the V200511B/meteo directory), for data coming from MM5. The mm5 interface can be directly applied to a standard output MM5 file (MMOUT files).
The second step for the meteorological interface is the diagnostic model **diagmet**, used to take standard meteorological variables on the CHIMERE grid and transform them into variables necessary for input to CHIMERE. For instance, the boundary layer height can be diagnosed from the vertical profile of

temperature, humidity and wind.
The boundary conditions interface takes either a "coarse run" produced by chimere and interpolates hourly pollutant concentrations at the domain boundaries, or provide climatological (monthly mean) values at the domain boundaries from a global database consisting of simulation outputs.
The land-use interface (**prepdomains**) is used to construct necessary terrain files and is used only once per domain construction.
The model core organization is described below. Briefly, it consists of the call of a main FORTRAN program (**chimere.f90**) which initializes and reads all data, and runs an integration convering the time period specified. The output files are then available for postprocessing.
The whole system is piloted by a **top calling script** which :

- reads parameters of the simulation
- links all necessary files into a **temporary directory** where all programs are executed
- compiles all necessary code
- runs the 3 "dynamic" interfaces (meteo, emission and boundary conditions)
- runs the model itself

The top calling script is executed to carry out a simulation. It must be edited and the options/directories must be chosen and defined by changing the values of parameters inside the script. In the distribution proposed on the CHIMERE web site, the example top calling script is **chimere-mm5** (see p.107).

5.1 Directories

The **V200511B** top directory contains individual files, the test case top calling script and subdirectories. These subdirectories are :

- **src/** : Contains the source routines of CHIMERE.
- **data/** : Contains data files for gas, aerosols and MELCHIOR 1 and 2 mechanisms. Only the MELCHIOR2 gas-phase mechanism can be used when aerosol option is choosen.
- **util/** : Contains some utility programs.
- **domains/** : A subdirectory where coordinate files COORD_[domain], biogenic emission potential files BIOFACS_[domain] and land-use files LANDUSE_[domain] must be placed for each CHIMERE model domain. This directory contains also the file **DOMAINS** to which one line should be added per new model domain.
- **meteo/** : This directory contains all the elements for the first step of meteo interfaces, i. e. the horizontal/time interpolation and reformatting of data. When the user starts from a new meteorological data set, a new shell script named **interf-[metname]** along with a read/interpolate fortran routine and the associated **Makefile-[model]** should be created by the user and placed in this directory. It is recommended to start from **interf-mm5**, **prepexmm5.f90** and **Makefile-mm5** as templates to help creating these interface programs.
- **toolbox/** : This directory contains two simple graphical interfaces for post-processing. See the documentation below.
- **script_functions/** : Contains scripts called by the top level script. While the top level script mainly defines variables, the scripts in this directory perform the low-level execution.
- **makefiles.hdr/** : Contains examples of Makefile headers suited for several architectures and compilers.

There are other directories created when running the model:

- A *temporary directory*, where all executable and temporary data files go. This allows in particular several simulations to be performed simultaneously by calls from a unique directory, provided the name of the temporary directory contains the unique process number (**$$** in Bourne Shell). Optionally this directory can be left after a model simulation (especially when debugging). By default it is removed at the end of top calling script execution. The address of this directory is given in the top calling script (environment variable **tmpdir**). It does not need to be created before the run.
  Hint for developpers :
  You can `cd` to **tmpdir** to modify the model code and test your modifications. The model environment and parameters are defined in files **chimere.nml** (also known as INFOS in previous verions) and **chimere_params.f90** which remain until you erase **tmpdir**. To perform a model run, without the overhead of replaying all the meteo and boundary conditions stuff, just type :

  ```
  make && time ./chimere.e
  ```

- An *output* directory where simulation output files are placed during the run. The address of this directory is given in the top calling script (variable **simuldir**. It does not need to be created before the run.

5.2 Model Files

The **Figure 5.2** shows the input and output files together with the executable programs or script that are used at all stages of the calling script. See the color legend in the right bottom corner. The yellow files are constructed before the run. File paths below are given relative to **V200511B**.

5.2.1 Scripts

chimere-mm5 is the top calling script to run the test case. It should be used as a template for user top calling scripts. When all data sets are ready for a simulation, the top calling script must be executed for running the simulation.

data/chemprep.awk to rebuilt a chemical mechanism (see explanations p.103)

data/chemprep.[meca] to rebuilt a chemical mechanism (see explanations p.103)

util/define_geom utility script used to define the geometric progressing layer vertical grid. It outputs a file with hybrid sigma-p coefficients defining the grid.

5.2.2 Meteo interface - 1st stage

The interface discribed below is for the MM5 mesoscale modelling system.

tmpdir/meteo/metargs-mm5 : ASCII file generated by the top calling script. Passed to **meteo/interf-mm5**. Contains parameters to drive **prepexmm5.e**.

meteo/interf-mm5 : script to run the first-step meteorological interface (interpolation + formatting). It is called by the top calling script.

tmpdir/meteo/prepexmm5.nml : namelist generated by **meteo/interf-mm5** to drive **prepexmm5.e**.

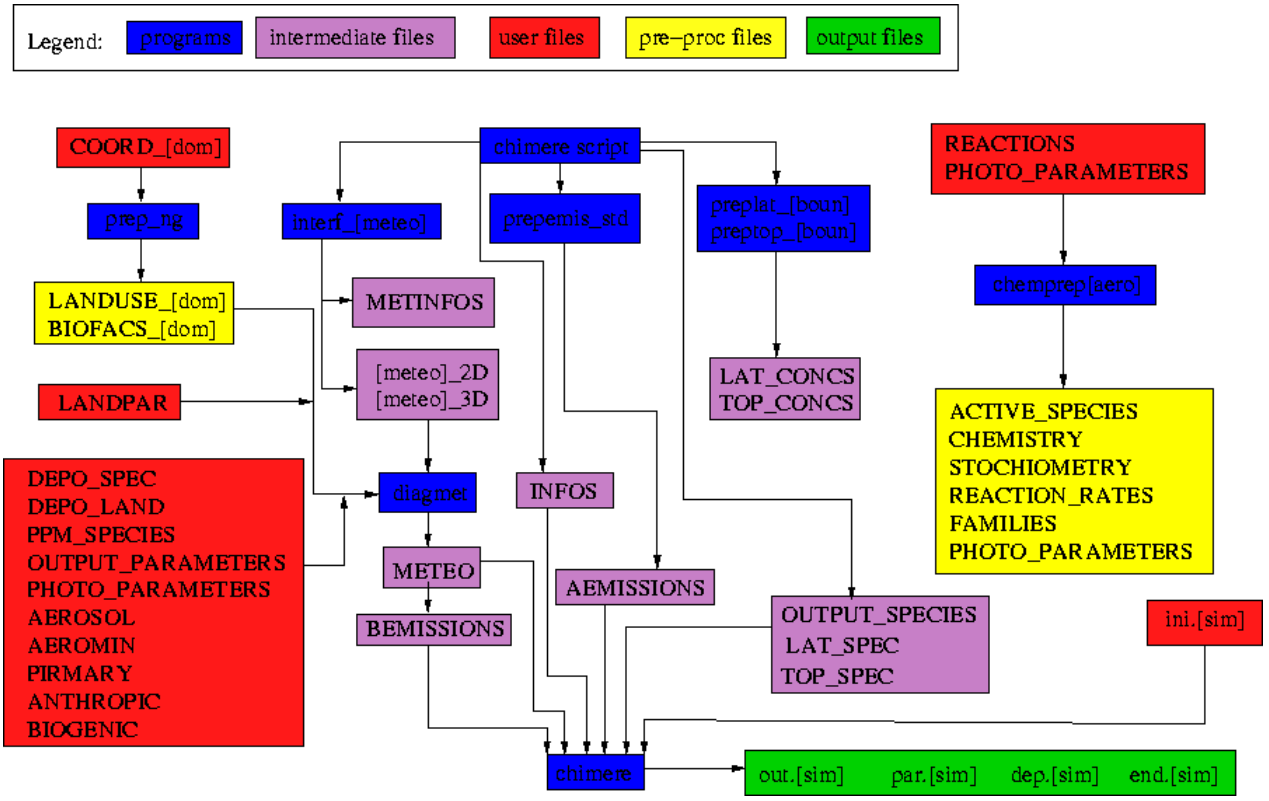


Figure 5.2: *Schematic general structure of the scripts, data files of CHIMERE*

meteo/COORD_MM5[domain] : contains the “cross” coordinates of the MM5 file.

meteo/COORD_MM5[domain _DOT] : contains the “dot” coordinates of the MM5 file.

util/make_mm5_coord : utility script allowing to extract coordinates and terrain altitude, in longitude/latitude format, of a MM5 **TERRAIN** file. Edit the script itself for giving information. The output files are **meteo/COORD_MM5[domain]** and **meteo/COORD_MM5[domain]_DOT** used by **interf-mm5**.

meteo/prepexmm5.f90 : reads the MM5 standard output files and performs the interpolation to the model grid. Compiled to **prepexmm5.e** and called by the script **interf-mm5**.

meteo/mm5_consts.f90 : defines uniquely the physical constants used by **prepexmm5.f90**.

meteo/subs.f90 : contains various helper subroutines.

meteo/wcal1.f90 : subroutine calculating the weighting coefficients for horizontal interpolation.

meteo/Makefile-mm5 : drives the compilation of the first stage of the meteo interface.

tmpdir/METINFOS.nml : namelist generated by **meteo/interf-mm5** to drive **diagmet.e** and **diag-bio.e**.

tmpdir/meteo/exdomout.nc : output netCDF file generated by **meteo/interf-mm5**. Contains meteo variables interpolated to the model grid.

5.2.3 Main programs

src/diagmet.f90 is the second step (diagnostic meteorological model) of the meteorological interface. Compiled to **diagmet.e** and called by the top calling script. It is not origin-dependent for meteo data. It ingests the output files from the first meteo interface.

src/diagbio.f90 generates the biogenic emissions file. Compiled to **diagbio.e** and called by the top calling script.

src/prepemis_std.f90 is the standard emission interface. Compiled to **prepemis_std.e** and called by the top calling script.

src/preplat_[bounds].f90 is the lateral boundary conditions interface.
Compiled to **preplat_[BOUNDS].e** and called by the top calling script.

src/preptop_[bounds].f90 is the top boundary conditions interface. Compiled to **preptop_[BOUNDS].e** and called by the top calling script.

src/chimere.f90 is the main program for chimere runs. Compiled to **chimere.e** and called by the top calling script.

5.2.4 Subroutines

Subroutines are located in the **src/** directory and are named *.f or *.f90 . Most of the subroutine files contain one subroutine, which has the same name as the file, e.g. **iniemis.f90** contains the subroutine **iniemis**. Comments can be found within the subroutines.
In some cases like **diagmet**, the code has been splitted into 3 files, for clarity. The main program **diagmet.f90** deals with all the I/O overhead, and calls subroutines in **diagmet_science.f90** which contains only the subroutines needed for meteorological diagnostic. The shared variables are held in **diagmet_common.f90**. In principle, a scientist wanting to develop a new diagnostic routine has mostly to deal with the **diagmet_science.f90** file, and has little to worry about the tedious netCDF I/O interface.

5.2.5 Header modules

Since the rewriting of CHIMERE in Fortran 95, modules are used instead of include files, at the exception of the ISORROPIA code.

src/chimere_consts.f90 defines uniquely the physical constants used throughout the model.

src/chimere_common.f90 contains the variables shared by all subroutines called by **chimere.f90**.

src/chimere_params.f90.sed contains all the parameters required to dimension the arrays at compile time. Each parameter is defined by a macro, which is subsituted before compile time by the top level script.

src/diagmet_common.f90 contains the variables shared by all subroutines called by **diagmet.f90**.

src/diagbio_common.f90 contains the variables shared by all subroutines called by **diagbio.f90**.

Despite the capability of Fortran 95 to perform dynamic memory allocation, CHIMERE still uses static allocation, and “common” declarations, because it proved to be significantly faster at run time.

5.2.6 Makefiles

The compilation stage is driven by one Makefile :

src/Makefile

The Makefile calls a unique top level header, **Makefile.hdr**, which deals with all the user installation details. **Makefile.hdr** is in fact a symbolic link to one of the predefined headers located in **makefiles.hdr** subdirectory.

5.2.7 Files used for land processes (deposition+emissions)

Their name is in capital letters and they lie in the **data/** directory and its subdirectories.

- domains/COORD_[domain]
- domains/BIOFACS_[domain]
- domains/LANDUSE_[domain]
- domains/DOMAINS
- data/DEPO_SPEC.[meca]
- data/DEPO_PARS
- data/BIOGENIC._[meca]
- data/ANTHROPIC._[meca]
- data/LANDPAR

5.2.8 Static files for chemistry and model information used for running the model

Their name is in capital letters and they lie in the **data/** directory and its subdirectories.

- data/vgrid/VCOORD.[vvgrid]
- data/boundaries_spec/[bounds]_SPEC.[meca]
- data/PPM_SPECIES.[meca]
- data/output_species/OUTPUT_SPECIES.[outspec]
- data/OUTPUT_PARAMETERS
- data/AEROMIN
- data/AEROSOL.[model]

5.2.9 Dynamic files generated at runtime

Dynamic parameters

tmpdir/METINFOS.nml : namelist generated by **meteo/interf-[metname]** to drive **diagmet.e** and **diagbio.e**.

tmpdir/diagmet.nml : namelist generated by the top calling script to drive **diagmet.e**.

tmpdir/diagbio.nml : namelist generated by the top calling script to drive **diagbio.e**.

tmpdir/preplat_[latboundgas].nml : namelist generated by the top calling script to drive **preplat_[latboundgas].e**. [latboundgas] is one of chimere, lmdzinca2 or mozart.

tmpdir/preptop_[topboundgas].nml : namelist generated by the top calling script to drive **preptop_[topboundgas].e**. [topboundgas] is one of chimere, lmdzinca2 or mozart.

tmpdir/concat.nml : namelist generated by the top calling script to drive **concat.e** which concatenates the gas and aerosols boundary concentrations.

tmpdir/prepemis_std.nml : namelist generated by the top calling script to drive **prepemis_std.e**.

tmpdir/chimere.nml : namelist containing most of the runtime parameters for the model. This file was the file named INFOS in previous model versions

Dynamic data files

tmpdir/meteo/exdomout.nc : netCDF file generated by the first stage of the meteo interface. Contains meteo parameters interpolated to the model grid.

tmpdir/METEO.nc : netCDF file generated by the meteo diagnostic stage. Contains meteo and diagnosed parameters.

tmpdir/AEMISSIONS.nc : netCDF file generated by the anthropogenic emissions interface. Contains timely anthropogenic emissions.

tmpdir/BEMISSIONS.nc : netCDF file generated by **diagbio.e**. Contains timely biogenic emissions.

tmpdir/LAT_CONCS.nc : netCDF file generated by the lateral boundary interface. Contains timely lateral concentrations.

tmpdir/TOP_CONCS.nc : netCDF file generated by the top boundary interface. Contains timely top concentrations.

Output files from chemprep preprocessor:

- data/chemistry/ACTIVE_REACTIONS.[meca].[model]
- data/chemistry/ACTIVE_SPECIES.[meca].[model]
- data/chemistry/CHEMISTRY.[meca].[model]
- data/chemistry/STOICHIOMETRY.[meca].[model]
- data/chemistry/FAMILIES.[meca].[model]
- data/chemistry/REACTION_RATES.[meca].[model]

Input files to chemprep preprocessor:

- data/chemistry/REACTIONS.[meca].[model]
- data/PHOTO_PARAMETERS.[meca]
- data/E-DEPENDENCIES.[meca]

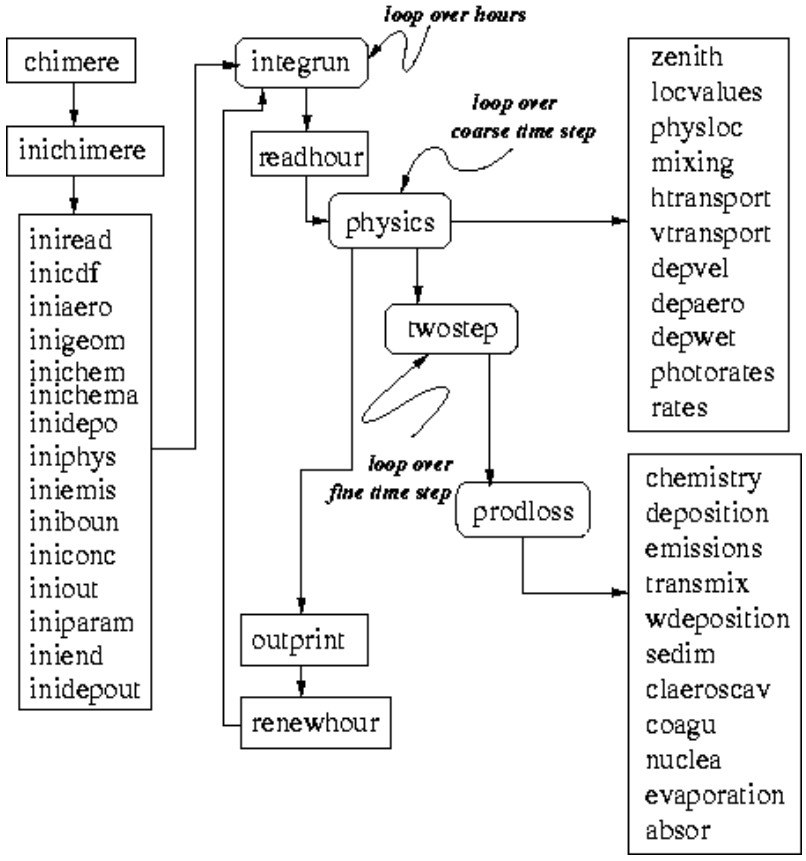


Figure 5.3: *Detailed list of CHIMERE routines and time loops*

5.3 Routine calls sequence

The general schematic diagram of calls of the CHIMERE model **chimere.f90** is given in figure 5.3. Only major routines are cited, not the utility ones.

1. The main program **chimere.f90** first calls all initialization routines (in **inichimere.f90** and further calls) where all "static data" data are read and many variables defined.
2. Then it calls the integration routine **integrun.f90** which then
   (a) reads hourly data every hour (routine **readhour.f90**
   (b) loops on the coarse time step,
   (c) outputs results on files (routine **outprint.f90**) and
   (d) shifts hourly data (routine **renewhour**).
3. At each model step, one needs hourly data for the start and the end of the hour in order to linearly interpolate, for meteo data, boundary conditions and biogenic emissions. The code keeps only these two instants in memory as explained in the **Figure 5.4**. Within the coarse time step, a call to routine **physics.f90** is made, in order to define all variables that will be taken as constants during the time step (linear interpolation of meteo data, calculation of reaction rates etc...), and a loop on the fine time step is carried out, with a call to the numerical solver **twostep.f90**.

4. The **twostep.f90** routine calls the core routine **prodloss.f90** which calculates, for all processes, the loss and production fluxes and sums them. Thus **prodloss.f90** calls **emissions.f90** (emission fluxes), **transmix.f90** (transport and mixing fluxes), **chemistry.f90** (chemical fluxes) and **deposition.f90** (deposition fluxes). The routine **transmix.f90** calculates the PPM concentrations at the boundaries with routines contained in the **Bppm.f90** package.

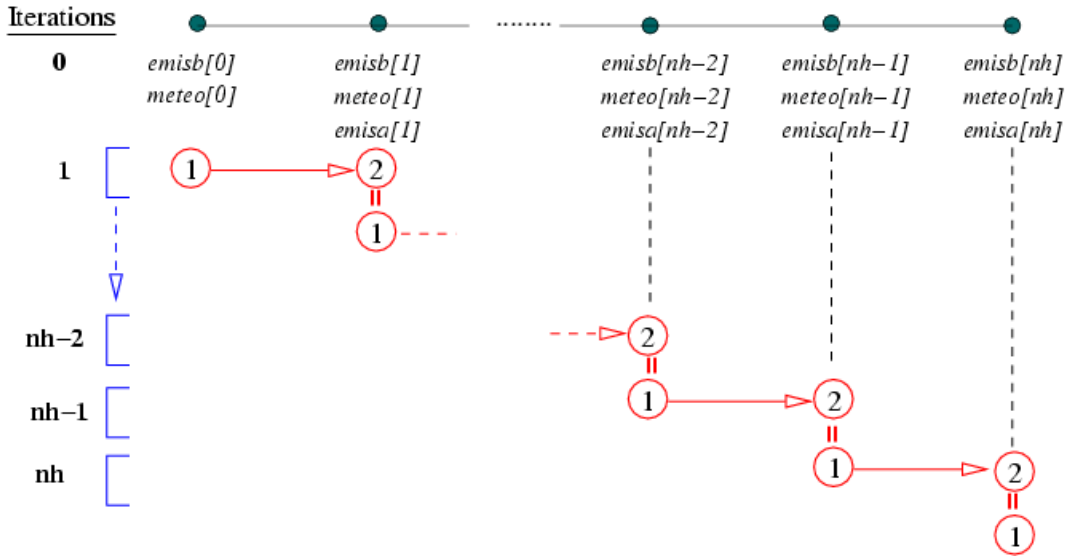


Figure 5.4: *Time processing in CHIMERE*

5.4 Core model input files

As discussed above, a CHIMERE run consists in several actions which are performed in the top calling script:

1. running the interfaces,

2. building the namelists and

3. executing **./chimere.e**.

This section describes the necessary input files to the third action. All these files must be present in the temporary directory before running the executable **chimere.e**. The top calling script copies and constructs these files in the temporary directory automatically. These required input files are listed in file **chimere.nml**.

Output files		
fnout	SIMULDIR/out.SIM	Timely concentrations of selected output species and families
fnothers	SIMULDIR/par.SIM	Timely meteorological parameters
fnconcs	SIMULDIR/end.SIM	Concentrations of active species at the end of the model run. Generally used as a restart file for subsequent runs.
fndepos	SIMULDIR/dep.SIM	Timely depositions
Model geometry		
fncoord	COORD	Coordinates (centers) file
chemistry		
fnoutspec	OUTPUT_SPECIES	File containing the name of output species
fnspec	ACTIVE_SPECIES	File of names of all active species
fnchem	CHEMISTRY	File containing preprocessed reactions
fnstoi	STOICHIOMETRY	File containing stoichiometric coefficients
fnrates	REACTION_RATES	File containing tables of reaction rates
fnppmsp	PPM_SPECIES	List of species transported with the PPM scheme
fnfamilies	FAMILIES	Definition of families
fnphot	PHOTO_PARAMETERS	Zenith angles tabulated
fnoutparam	OUTPUT_PARAMETERS	List of output parameters
fnaerosol	AEROSOL	aerosols: number of bins, molar weights
fnaeromin	AEROMIN	Parameters for ISORROPIA scheme for aerosols
fnprim	PRIMARY	Size distribution of aerosols
Emissions		
fnanthro	ANTHROPIC	List of anthropic emitted species
fnbiogen	BIOGENIC	List of biogenic emitted species
fnemisa	AEMISSIONS.nc	Anthropic Emission file (molec/cm2/s)
fnemisb	BEMISSIONS.nc	Biogenic emissions
Boundary conditions		
fnlatspec	LAT_SPEC	List of species for the lateral boundary conditions
fntopspec	TOP_SPEC	List of species for the top boundary conditions
fnlatconc	LAT_CONCS.nc	Species concentrations for the lateral boundary conditions
fntopconc	TOP_CONCS.nc	Species concentrations for the top boundary conditions
Meteorology		
fnmeteo	METEO.nc	All meteo input fields in a single file
Dry and wet deposition		
fndepoespe	DEPO_SPEC	coefficients for calculation of resistances / species
fndeposoil	DEPO_LAND	coefficients for calculation of resistances / landuse
fndepopars	DEPO_PARS	landuse for the grid
fnwetd	WETD_SPEC	wet deposition species and parameters
fnlanduse	LANDUSE	Land-use fractions into categories

5.5 Core model output files

These output files are listed in file **chimere.nml**

fnout Timely concentrations of selected output species and families

fnothers Timely meteorological parameters

fnconcs Concentrations of active species at the end of the model run. Generally used as a restart file for subsequent runs.

fndepos Timely depositions

The format of these files is more precisely described in **§4.3.2,p.40**.

5.6 Running real cases

Here are the following steps to achieve for running a simulation with the model:

1. First step is to install all the code and data, and configure **Makefile.hdr**. We assume this step has been done since you were able to run the test case.

2. Second step to achieve is to find meteorological data. There is no meteorological model provided with CHIMERE. CHIMERE can use many meteorological models but users should provide their own interface, except for MM5, for which an interface is proposed in the code, which takes as input standard MM5 output files (MMOUT files). If MM5 (which is a free software, available on the NCAR[1] web site, is chosen as the meteorological driver, it has to be initialized with other global meteorological data (analyses), which often are not public. However the National Centres for Environmental Prediction (NCEP) and the National Centre for Atmospheric Research provide a wide variety of meteorological data, analyses etc... online. Our experience deals with real-time AVN/NCEP analyses and forecasts that can be obtained on a web server. When setting up a meteorological simulation using MM5, choose a meteorological domain entirely containing the CHIMERE domain, and a time period entirely containing the CHIMERE period to simulate. The PBL option "MRF" (Option #5), based on the Troen and Mahrt (1986) parameterization should be prefered because it is most consistent with the model mixing formulation. The Schultz (Option #8) microphysics parameterization has also been tested with CHIMERE and is recommended. Also in case of the use of MM5, the MM5 coordinate files (**COORD_MM5[dom]** and **COORD_MM5[dom]_DOT** must be constructed from the MM5 TERRAIN file.

3. Prepare the land-parameters for the CHIMERE domain. There are 3 necessary land files: The coordinate file, the file containing the fractions in all of 9 land-use classes for all grid cells, and the file containing biogenic emission potentials. The directory provided online (**prepdomains**) only works **for Europe**. It works with the Global Land Cover Facility (GLCF) land-use data base (see documentation below). Other methodologies should be found for other parts of the world, in particular for constructing the emission potentials. Perhaps as a very first experiment for a simulation outside Europe it could be advised to remove them by setting a BIOFACS_[domain] file with 5 columns with zeros in them. LANDUSE_[domains] files are easier to construct from the GLCF data base LANDUSE_[domains] files are easier to construct from the GLCF data base. See **6.2** for more details.

4. Emission files have to be constructed and specified for the chosen mechanism. This is often a difficult task. In paragraphs **§6.3, p.69** for anthropogenic and **§6.4, p.71** for biogenic, indications are provided on how to produce emission files to be input to CHIMERE. If the standard emission procedure is followed (one emission file per calendar month containing hourly emissions for all MELCHIOR emitted species and for 3 typical days (week day, saturday and sunday), the interface program **prepemis_std.f** proposed here (in the **src_[model]** directories can be used. Otherwise the interface program must be recoded.

5. The **top calling script** (**chimere-mm5** for instance) must be modified according to the conditions of the simulation. Edit this Bash-shell script and make changes in the values of the variables as required.

[1] http://www.ncar.ucar.edu

6. Run the model by invoking the top calling script.

```
nice -10 chimere-mm5 | tee chimere.log
```

Chapter 6

Make your own CHIMERE configuration

In this part, all data files required by the model to work are described. The adaptations to be done to define a new simulation domain are discussed.

6.1 Meteorological interface

CHIMERE does not contain any meteorological driver. It is only a chemical-transport model, meaning that meteorological data have to be provided in order to perform a simulation. The provided code only contains a meteorological interface which allows to start from standard meteorological variables and transforms it to CHIMERE inputs. In practice the meteorological interface consists in the construction of a single file **METEO.nc** to be made available as input to the executable **chimere.e** in the temporary directory **tempdir**. It contains all hourly meteorological variables in a single netCDF file.
The construction of this files is made in two stages :

- The first stage is an **interpolation stage** : it takes meteorological variables from a meteorological database, simulation or forecast file and transforms it into standard variables given on the horizontal CHIMERE grid as hourly values. **This stage has to be constructed by the user**. Specific files with specific formats and filenames must be built. One example, with interface to MM5 output files are given.
- The second stage is a diagnostic meteorological interface (the **src/diagmet.f90**. It is a generic program allowing to transform standard meteo variables into CHIMERE inputs. It also allows construction of possibly non-available variables from standard variables and some parameterizations.

6.1.1 First stage: the "interf" programs

The top calling script calls a program named **interf-[metname]** in the **meteo** directory. This program (usually a bourne shell script) has to be written by the user. One example is given : **interf-mm5** which is a generic interface to mm5. If other meteorological data is to be used, the user should give it a name, [metname], and write the interf-[metname] bash program. It should also change the **"meteo="** line in the top calling script.
In turn, **interf-[metname]** calls a fortran routine, which was named in the case of MM5 **prepexmm5.f90**. This **prepex[metname].f90** routine has several roles to play :

- extract hourly meteorological fields and time information from the Meteorological Model files, starting on the first CHIMERE hour simulation and ending on the last simulation hour.

- Horizontally interpolate variables to CHIMERE grid. However the original vertical levels are kept.
- Write these variables and metadata to the **exdomout.nc** netCDF file with a given **mandatory** structure.

interf-[metname] generates also two parameter files :

- **METINFOS.nml** contains parameters for the 2nd meteo stage **diagmet.f90**
- **prepex[metname].nml** contains parameters for **prepex[metname].f90**

The **Table 6.1** lists the mandatory and optional variables which must or can be output by the interpolation stage.

File Name	Variable	Dimension	Units
Mandatory Variables			
lon	longitude of gridpoints	2D	degrees_east
lat	latitude of gridpoints	2D	degrees_north
winz	Zonal component of the wind	3D	m/s
winm	Meridional component of the wind	3D	m/s
temp	Temperature	3D	K
sphu	Specific humidity	3D	kg/kg
alti	Altitude of half layers on which all 3D meteo variables should be given	3D	m
pres	Pressure at half layers on which all 3D meteo variables should be given	3D	Pa
cliq	Cloud liquid water content (not including rain water)[1]	3D	K
tem2	2m Temperature	2D	K
copc	Convective Precipitation[1]	2D	kg/m^2/hour
lspc	Large-scale Precipitation[1]	2D	kg/m^2/hour
Optional Variables			
cliq	3D cloud water content for radiation, optional for gas-phase, mandatory for aerosols	3D	Kg/Kg
rain	3D rain water (to be added to cloud water for radiation)	3D	Kg/Kg
cice	3D ice (for radiation)	3D	Kg/Kg
sshf	Surface sensible heat flux, optionally read or recalculated	2D	W/m2
slhf	Surface latent heat flux, optionally read	2D	W/m2
usta	Friction velocity, optionally read or recalculated from Louis [1982]	2D	m/s
hght	Boundary layer height, optionally read or recalculated	2D	m
lowc	Low cloud fraction, optional for radiation	2D	m
medc	Medium cloud fraction, optional for radiation	2D	m
higc	High cloud fraction, optional for radiation	2D	m

Table 6.1: *Mandatory and optional variables in* **exdomout.nc** *meteo file. The parameters quoted with* [1] *are mandatory only for the aerosol option*

6.1.1.1 Recommendations

There are a number of recommendations that can be given about the choices of the optional variables. Some are listed below.

- If **boundary layer height** is not available from meteo data, it will be calculated using the maximum between a formula for stable air (using a critical Richardson number=0.5), and a parameterization for unstable boundary layer based on a simplification of the Cheinet [2002, PhD thesis] work. These calculations assume a dry boundary layer. The user should use meteorological model's boundary layer height if more elaborate and whenever possible, especially if it accounts for clouds.
- The **friction velocity** u_* is used for deposition and calculation of diffusivities. it is a particularly sensitive parameter for ozone in Summer through the calculation of aerodynamic resistance Ra. It should strongly depend on land use type, which are critical to deposition. In meteorological models, especially large-scale, roughness lengths are sometimes estimated too roughly for the application to high-resolution deposition. The recalculation of u_* which is proposed in the **diagmet.f90** interface uses the Louis [1982] formulation which is particularly robust and gave satisfactory results. We advise to use it anyway, in particular to have deposition consistent with a high-resolution land use.
- **Heat fluxes** are also used for the calculation of W* and therefore mixing, and the height of the boundary layer. In fact only the virtual heat flux is required, which can be recalculated from an empirical formula [Priestley, 1948] using temperatures in the first meteorological model layers. This formula is not very accurate. It is strongly advised to use heat fluxes from the meteorological model.
- For **radiation** several calculation options are offered. All are based on a statistical model of the clouds optical thickness.
  - ☐ The first option uses liquid/ice water (whenever available), a variable which can be very model-dependent and which is rarely verified against observations. Moreover it does not take into account fractionary cloudiness. However it may be suited for high/medium clouds which are often stratiform clouds in the midlatitudes.
  - ☐ The second option uses cloud fractions (whenever available), which is less meaningful to radiation but could be used when cloud/ice water is not available.
  - ☐ The third option uses relative (to liquid water) humidity, and has been tuned using J(NO2) measurements taken over a long period near Paris (manuscript in preparation). The formulation, modifying radiation for RH>0.75 should only be used for low clouds. It has the advantage of being a more robust parameterization since it depends on a highly verified variable.

  In summary, our preference is: RH formulation (option 2 in **METINFOS.nml**) for low clouds, and for liquid/ice water formulation (whenever available) for medium and high clouds, and cloud fractions otherwise.

6.1.1.2 The METINFOS.nml file

METINFOS.nml is a parameter file that is passed to **diagmet.f90** to provide information about the chosen options. In the MM5 example it is constructed from a template file **meteo/METINFOS-mm5.nml.sed** where macros are replaced by values when running **interf-[metname]**. **METINFOS.nml** should be placed in **tmpdir** temporary directory. It has the following structure (example drawn from MM5) :

```
&metoptions
  opt%idstart = 2003073000 ! Start date of the run
  opt%nsho    = 1904       ! grid cell for screen display
```

```
  !
  opt%w10m    = 0    ! 10m wind option (0=read from file, 1=recalculate)
  opt%usta    = 0    ! Ustar option (0=read from file, 1=recalculate from Louis 1982)
  opt%flux    = 0    ! Flux  option (0=read, 1=recalculate from Priestley 1948)
  opt%pblh    = 0    ! Boundary layer height (0= read from file,1=recalculate)
  opt%cice    = 0    ! Cloud Ice Wat (0=Use in addition to cloud water, 1=don't)
  opt%rain    = 0    ! Rain Wat      (0=Use in addition to cloud water, 1=don't)
  !
  opt%upm     = 0    ! Urban corrections for BLH
  opt%ufx     = 0    ! Urban corrections for flux
  opt%uwc     = 1    ! Urban corrections for wind
  !
  opt%clol    = 2    ! Low cloud option (0=read cloudiness, 1=Use Liq. Wat., 2=Use RH)
  opt%crhl    = 0.85
  !
  opt%clom    = 1    ! Medium cloud option for attenuation ...
  opt%crhm    = 0.95
  !
  opt%cloh    = 1    ! High cloud option for attenuation ...
  opt%crhh    = 0.95
  !
  opt%soim    = 0    ! Erosion/Resusp.: (0=read soil moist. [m3/m3],1=don't=>no erosion)
  opt%eros    = 1    ! Erosion option
  opt%resu    = 0    ! Resuspension option
/
```

6.1.1.3 The prepex[metname].nml file

prepex[metname].nml is a parameter file that is passed to **prepex[metname].f90** to provide information about options and file paths. It is constructed by **interf-[metname]**. **prepex[metname].nml** file should be placed in the **tmpdir/meteo** directory. It has the following structure (example drawn from MM5) :

```
&args
ids         = 2003073000
              ! start date of the run
nhours      = 12
              ! length of the run in hours
domain      = 'CONT3'
              ! CHIMERE domain
fn_mdom     = 'COORD_MM5EUR1'
              ! MM5 ''crs'' coordinates file
fn_mdom_dot = 'COORD_MM5EUR1_DOT'
              ! MM5 ''dot'' coordinates file
fn_coo      = '/ws3/monge/CHIMERE-AERO-NEW/V200501Hjlm/domains/COORD_CONT3'
              ! CHIMERE domain coordinates file
nzonal      = 67
              ! number of model grid cells in the zonal direction
nmerid      = 46
              ! number of model grid cells in the meridian direction
fn_mm5      = '/COMPERES/pub/MM5/AVN/2003/EUR1/MMOUT_EUR1_20030730_20030803_S'
              ! MM5 meteo file
fn_out      = '/ws3/monge/CHIMERE-AERO-NEW/V200501Hjlm/28089/exdomout.nc'
              ! prepexmm5.e output file. Do not modify.
/
```

6.1.1.4 Example : the MM5 interface

The **interf-mm5** interface starts from a unique MM5 file, **MMOUT_[mm5-domain]**, specified **exactly as in the V3.6 MM5 format**, see the NCAR web site for a full documentation of MM5 and output format (http://ncar.ucar.edu). The MMOUT file is assumed to be written with word byte order reversed relative to standard PC words. In this interface it is assumed that the output frequency for MM5 simulations is 1 hour, so there is no time interpolation for input to CHIMERE.

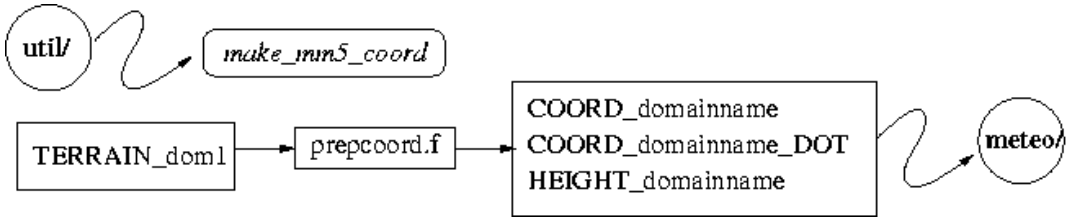


Figure 6.1: *How to prepare MM5 coordinates files for CHIMERE, from the original MM5 terrain file*

Before using the MM5 interface, three coordinate files for the MM5 grid must be constructed manually **Figure 6.1**. There is a file for the CRS coordinates (used for all variables but wind) and a file for the DOT coordinates (shifted grid for the wind fields). A utility for making these file coordinates is provided in the CHIMERE **util** directory. Edit the **make_mm5_coord** utility script for more help. These files (**COORD_MM5[domain]**, **COORD_MM5[domain]_DOT**) and **HEIGHT_MM5[domain]**) must be placed in the **meteo** directory once they are constructed.
There are no prior assumptions on the choice of physical options in MM5. In our applications, however, the following options have been used and can therefore be recommended:

- Schultz microphysics,
- MRF (Troen-Mahrt) PBL
- Grell cumulus scheme
- RRTF radiation scheme
- In all applications over Europe where AVN/NCEP is used as forcing data, we found that nudging was necessary (grid FDDA option), with nudging coefficients varying from 0.00001 to 0.0001, depending on the spatial scale, but no meteorological observations were used.

6.1.1.5 Building your own "interf" interface

There are two steps to perform this task :

- create a new **interf-[metname]** script
- create a new **prepex[metname].f90** routine

The first step is trivial, since it consists in reading arguments from a file **metargs-[metname]** generated by the top calling script, copying/translating them to two namelists **prepex[metname].nml** and **METINFOS.nml**, and finally compiling and launching the executable **prepex[metname].e**. It is recommended to use **interf-mm5** as a template.

The second step might be a painful job.
The user may start from scratch. In that case, the only prerequisites are :

- The format of input files
- The list of output variables, together with their **mandatory** name
- The structure of the netCDF output file named **exdomout.nc**. This structure is entirely defined by the header listing in **Appendix F.1, p.91**.

However, the recommended procedure is to start from **prepexmm5.f90** as a template. The interpolation stage and the output to the netCDF file are already written and should not be so much modified. The real work for the developper of a new interface would be :

- to understand the data structures of the program. In short, a table contains all metadata of input and output files. This metadata stucture is linked to a meteo data array by the index of the current variable. The fields in the metadata structure are used to instruct a rather general program what to do with the current variable. The macro `_LINE_` is used to generate unique and monotonic indexes to variables.
- to create the read interface for their meteorological data, paying attention to synchronisation : read and write the right data at the right time slot.

6.1.2 Second stage: the diagmet diagnostic model

The second stage of the meteorological interface consists in the production of the meteo input file for CHIMERE, **METEO.nc**, from the output meteo file of the first stage **exdomout.nc**. This is achieved by the program **src/diagmet.f90**. It is a generic program, designed to be independent on the origin of the meteorological data.

diagmet.f90 requires the following set of files :

- **exdomout.nc** : Contains the hourly 2D and 3D meteorological fields constructed by the first stage of the meteo interface, interpolated to the CHIMERE grid. It is encoded in the netCDF format. For 3D fields, the vertical grid is still that of the meteorological forcing model. There are mandatory and optional fields (see first-stage section above).
- **METINFOS.nml** : Namelist containing the meteorological option flags and parameters.
- **diagmet.nml** : Namelist containing file paths.
- **LANDUSE_domain** : File containing the fraction of land use per grid cell. There are 9 land use categories in this model version, see the Land-use interface below for the description of these classes. The file therefore contains 9 columns with numbers between 0 and 1. This file must be constructed once for each model domain using the Land-use (or an equivalent) interface. This file is generally located in the **domains/** directory and is pointed to by an address in **diagmet.nml**.
- **VCOORD.[vgname]** : This file contains the vertical grid hybrid sigma-p coefficients for the top of all CHIMERE layers. This file is generally located in the **data/vgrid/** directory and is pointed to by an address in **diagmet.nml**.
- **LANDPAR** : This file gives some land-use/season dependent surface parameters: roughness height Zo, extra fluxes, wind reduction or extra urban heat-island temperature increment. It is generally located in **data/** directory, and is pointed to by an address in **diagmet.nml**.

Model coordinates are read from the netCDF file **exdomout.nc**.
The output file **METEO.nc** contains one record per hour and its structure is documented in **Appendix F.2, p.93**.

6.2 Model domains and landuse

CHIMERE needs three files to describe the surface:

1. A coordinate file named **COORD_[domain]**
2. A landuse file named **LANDUSE_[domain]**
3. A biogenic emission factor file named **BIOFACS_[domain]**

where in the following, [domain] is a name given by the user to the model horizontal grid/domain. Last, a fourth file named **DOMAINS** has to be filled. It is available in the **domains/** directory and must then have a line containing the name of the domain, the number of cells in the x-direction, the number of cells in the y-direction, and a grid point number for outputing values on screen. When the model is run, either **diagmet.f90** or **chimere.f90** use this grid cell for displaying values on the screen.
For each of these three files, it is possible:

1. Just to use the already prepared files (contained in the model source distribution)
2. To built new files inside the Europe domain and thus use the database proposed in the model source distribution
3. To built new files outside of Europe and, thus, use your own database.

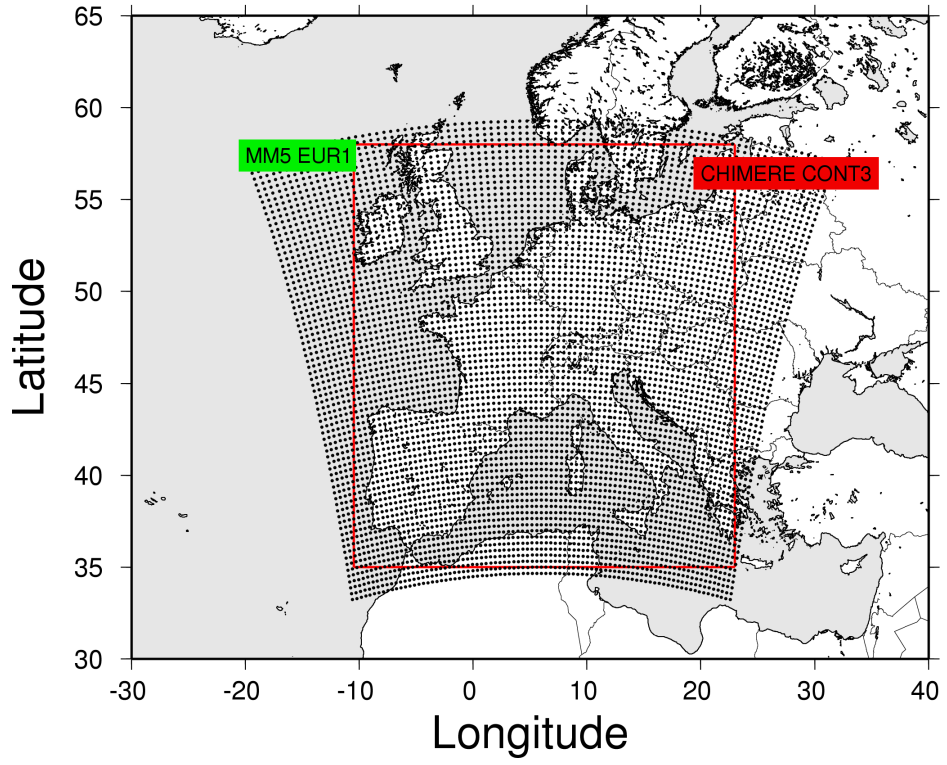


Figure 6.2: *Examples of horizontal domains. The dots represent the MM5 EUR1 grid cells centers and the rectangle represents the boundaries of the CHIMERE CONT3 domain.*

6.2.1 Generalities about the domain files

Model domains are entirely defined by their grid cell centres description. It is given in the COORD_[domain] file which must be placed in the **domains/** directory. The **COORD_[domain]** file contains lines with a longitude (in decimal degree) and a latitude (in decimal degree). Each pair of coordinates stands for a grid cell centre, described (from top to bottom of file) from West to East then from South to North.
Model grids can be any quasi-rectangular grid, with weakly varying spatial step. It is assumed, in particular for the Parabolic Piecewise Method for tranpsort that locally (over 5 consecutive cells) the grid size is constant in each direction, with sizes equal to that of the locally central cell. Sphericity effects are taken into account, but therefore linearized. It is important to note that the COORD file for CHIMERE is absolutly not the same that the one used for MM5 (or others meteorological inputs). An example is diaplyed in the **Figure 6.2**.
Finally, the user may also change the number and size of the vertical levels. To produce a new **VCOORD** file, follow the steps as explained in the **Appendix I.2, p.112**. It is necessary to report the correct number of vertical levels into the top calling script.

6.2.1.1 Horizontal domain COORD

The file **DOMAINS** contained in the **domains/** directory must then have a line containing the name of the domain, the number of cells in the x-direction, the number of cells in the y-direction, and a grid point number for outputing values on screen. When the model is run, either **diagmet.f90** or **chimere.f90** use this grid cell for displaying values on the screen.
example:

```
domain  NX  NY  NSHO
CONT3   67  46  1904
```

6.2.1.2 Biogenic emissions potentials BIOFACS

The **BIOFACS_[domain]** file contains, for each grid cell, 5 values describing the biogenic emission potentials to be combined with the meteorological factors (temperature, PAR) in order to construct the hourly biogenic emission file **BEMISSIONS.nc** in the biogenic interface **diagbio.f90**.

- The first column is for isoprene emission. It is the isoprene landuse-dependent coefficient Eo (in molec/cm2/s), in the cell, which has to be multiplied by the meteorological factors
- The second and third columns are the coefficients E_1 and E_2 coefficients for terpene emissions.
- The fourth and fifth columns are the emission potential Eo for biogenic emission for two seasons: growing season (4th column), assumed to be from may to July, and for the other months (5th column).

6.2.1.3 Land-use types

The **LANDUSE_[domain]** file contains the fraction of land use per grid cell (numbers between 0 and 1). In the V200511B model version, 9 land-use types are recognized. They are described in the **Table 6.2**, with the correspondance to the GLCF (University of maryland) data base used for the propoposed **prep-domains** interface.

Category Number	Description	Corresp. GLCF (0-13)
1	Agricultural land / crops	[11]:Cropland
2	Grassland	[10]+1/2[7]:Grassland+1/2Wooded grassland
3	Barren land/bare ground	[12]:Bare ground
4	Inland Water	
4	Urban	[13]:Urban and built-up
6	Shrubs	[8]+[9]: Shrublands
7	Needleaf forest	[1]+[3]+1/2([5]+[6])+1/4[7]
8	Broadleaf forest	[2]+[4]+1/2([5]+[6])+1/4[7]
9	Ocean	[0]: Water

Table 6.2: *Landuse categories used in CHIMERE*

6.2.2 Use the predefined domain files

There are predefined domains :

- **CONT1** (1/2x1/2 degree resolution over limited Western Europe domain)
- **CONT2** (1/2x1/4 degree resolution over the same domain)
- **CONT3** (extended 1/2x1/2 resolution, domain now including Spain, Portugal and Italy.
- **IDF** A domain of 150kmx150km around Paris with a 6-km resolution
- .... and some others used in the CHIMERE community...

For each of these domains, the correspondings COORD, LANDUSE and BIOFACS are present in the **domains** directory.

6.2.3 Built new domain files inside Europe: prepdomain

The land-use interface job is to construct files that depends on the CHIMERE model domain only. Therefore these files need only to be constructed once per model domain. The interface is not called in the top calling script, and it is assumed when running the CHIMERE script chimere-mm5 that the domain files are already constructed and placed in the **domains/** directory. The user can either make his own interface to produce the files or use the (**prepdomains**) proposed interface which, at present, *works only for model domains over Europe*.In the definition of a new CHIMERE domain, a certain number of steps have to be followed:

- Check whether domain is quasi-rectangular. Most projection systems work, even a regular grid in geographic coordinates (longitude-latitude) provided the resolution is not too coarse (say > 2 degree).
- Build a **COORD_[domain]** file containing longitude and latitudes of the grid cell centres. An easy to use script is proposed in the **Appendix I.1, p.111**.
- use the **prepdomains** interface or your own interface to build the **LANDUSE_[domain]** and the **BIOFACS_[domain]** files. Using **prepdomains** is achieved by editing the **prep_ng** script and changing the first 3 program lines.
- The **LANDUSE_[domain]** file can be built using the **prepdomains** interface
- Once the files are constructed, move them into the **domains** CHIMERE directory.
- In the **domains** CHIMERE directory, edit the **DOMAINS** file and add relevant information for the new domain.

We can turn to more detailed explanations about the **prepdomain** user interface. Of course, there is not one way to construct the previous domain files. Different methodologies can be used for calculating biogenic emissions and landuse.
The propose **prepdomain** interface is based on the Global Land Cover Facility (GLCF[1]) 1km×1km resolution database from the University of Maryland, following the methodology of Hansen et al. (2000, J. Remote Sensing).
The **prepdomains** interface downloadable for CHIMERE applies only to Europe because:

- Only a subset of the GLCF land cover is used (over Europe),
- The tree distribution used here for biogenic emissions contains only european countries informations. Therefore biogenic emissions procedure should be completely changed for other continents.

By "Europe" it is meant the domain extending from 15W to 30E and 34N to 61N.
Once the tar file has been downloaded and extracted (about 1 gbytes !!!, see the **section 4.5, p.45** for details), one should find several files:

- **LAND_EUR1**: An ASCII piece of the GLCF landuse classes: Format: Longitude (decimal degree), Latitude (dec. degree) and land cover category [0-13] (see the GLCF documentation).
- **COUNTRY_EUR1**: An ASCII file giving the country code (see the order in first line of **FACT_FOREST** file).
- **FACT_FOREST**: ASCII file giving the tree distribution by country, and the total amount of Fertilizer used (second line, in Kg/ha/Year of Nitrogen), from Stohl (1996, Atmospheric Environment).After Line 3 follow the tree species distribution.
- **FACT_RATE**: These are emission factors for individual trees and other characteristics (deciduous or permanent leaves).
- **FACT_LAND**: Emission factors from soil (unit ???)
- **LAND_AGGREGATION**: This is the matrix of correspondence between the 14 GLCF categories and the 9 CHIMERE categories.
- **README**: Read it
- **biofacs.f**: Program constructing the biogenic emission potentials, given the land-use and the other distribution (trees, countries, ...).
- **landuse_ng.f**: Calculates the 9-category fractions on the CHIMERE grid from GLCF.
- **countries_ng.f**: Calculates country area fraction on the CHIMERE grid
- **prep_ng**: This is the program to edit, modify and execute for each new domain.

The interface use is very simple:

- Provide first, in the **prepdomains** directory, the **COORD_[domain]** file containing the centres of the grid cells,
- edit **prep_ng**. Modifiy the first 3 program lines:
  domain name, nxx= and nyy=.
- Then simply execute the **prep_ng** program.

This can take 5-30 minutes (or more...). At the end, a directory with the domain name is created containing the necessary files for CHIMERE. Move these files to the CHIMERE **domains/** directory. Then edit the **domains/DOMAINS** files and add a line describing the new domain. Interface work is done (if things worked correctly...).

[1] http://glcf.umiacs.umd.edu/data/landcover

6.2.4 Built new domain files outside of Europe

In this case, you also have to provide the three files. COORD, LANDUSE and BIOFACS.

6.2.4.1 COORD outside of Europe

The **COORD_[domain] file** is easy to generate as described in the **section I.1,p.111**.

6.2.4.2 LANDUSE outside of Europe

The **LANDUSE_[domain] file** may be generate by modifying the **prepdomain** program. The methodology available here for Europe is fairly general. The solution is to download the original 1Km×1Km GLCF land cover data base, and use an extraction program as the extract_eur.f program (provided separately). In this program it is assumed that the name of the GLCF global file is **latlon**. Take therefore the latitude-longitude version of this file from the GLCF server.

6.2.4.3 BIOFACS outside of Europe

The extension of biogenic emissions to other regions than Europe is much more difficult. While the general methodology remains correct, the tree species list must be changed for other continents, especially for tropical areas. The tree distribution per country has to be redefined with new countries, and the tree emission factors must be redefined. Also the biogenic emission of NO methodology from Stohl (1996) may be difficult to extend to other areas in the world, at least to tropical areas where soil properties may be very different.
Another possibility is of rebuilding a new methodology for constructing the **BIOFACS_[domain]** file, which would be more adequate to the new area. Also biogenic emissions can be calculated separately on an hourly basis instead of on a "emission potential X meteo factor" methodology. In this case it is advised to set the numbers in the **BIOFACS_[domain]** file to 0 and to read biogenic emissions and build directly a **BEMISSION.nc** file from an interface program that would be called just after the anthropic emission interface in the top calling script. Sometimes also biogenic emissions are simply included in the anthropic emissions, in which case the **BIOFACS_[domain]** file should be set to 0.

6.3 Anthropic emission interface

This Section explains how to provide data to the anthropic emissions interface that is in the CHIMERE package, assuming that the user has his own emissions data bases.
CHIMERE requires a file containing hourly anthropic emissions for the time period to simulate. This file must be placed in the temporary directory and its name is **AEMISSIONS.nc**. This file is encoded in the netCDF format, and its structure is defined in **Appendix F.3, p.94**.
The construction of the **AEMISSIONS.nc** file can be done in two ways: either by using the standard interface provided here, which however requires initial data in a specific format, or by a specific, user-provided interface. The standard procedure proposed here consists in two steps:

- The construction of an emission data base consisting of 12 files containing each hourly emissions for 3 typical days for a given calendar month,
- The construction of the **AEMISSIONS.nc** file from this data base.

The first stage is the most difficult task, and there is no software on CHIMERE server to perform this stage. We give here only indications on how to proceed in practice in most cases. The output of this stage are 12 ASCII files, named **EMISSIONS-[domain].[MM]** ([MM] is the calendar month) like those provided for European or Ile-de-France in CHIMERE server in the **dowload** Section. The format is such that it can be read by fortran as:

```
do ne=1,nemisa
do nt=1,ntypedays
do nh=1,nhoriz
   read(iunit,*)((emisprof(ne,nt,nh,ih,nl),ih=1,24),nl=1,nlevemis
enddo
enddo
enddo
```

ntypedays is 3, **nt=1** standing for weekday emissions, **nt=2** for saturday emissions and **nt=3** for sunday emissions. This does not allow, in particular, to have emissions depending on meteorology or other factors than the day of week, but this keeps databases relatively compact.

In general, in order to construct model-species emission files like these from emission inventories, the starting point can be an annual inventory of VOC, NOx, CO, SO2, NH3 and PM per activity sectors. Then the annual totals undergo time disaggregation by applying typical monthly and hourly profiles depending on week day. Such profiles depend very much on the application and no such profiles are given on CHIMERE server.

Then the VOC and NOx are disaggregated into real compounds using a speciation profile (depending on activity sector). In the data provided on CHIMERE server, the NOx speciation is 90% of the NO2-equivalent mass of NOx is NO, 9.2% is NO2 and 0.8% is HONO. The VOC speciations come from either the british PORG speciation for the continental model or from the University of Stuttgart (IER) for the Ile-de-France case. The british speciation is public.

The construction of model-species emissions then requires a Middleton (1990) type of aggregation procedure. Each real VOC emissions are aggregated into one or several model VOC(s), with a branching coefficient and a reactivity-dependent coefficient (see the Middleton procedure), for which reactivities and molar masses for both the real and the model VOCs have to be known. In the british speciation, 227 real VOCs are considered, and the following **AGGREGATION** file gives these informations for the MELCHIOR mechanism. Reactivities have been calculated on the basis of the Leeds University MCM Master Mechanism and the Kwok structure/reactivity approach. The **SPECIATION** files gives the VOCs list and the 9-activity speciation, and the way VOCs are preprocessed in CHIMERE for continental scale.

The second stage consists in making the hourly, simulation adapted, **AEMISSIONS.nc** file from the database. If we forget the netCDF overhead, this is a fairly simple procedure, achieved by the **src/prepemis_std.f90** program. Take a look at this program. It calls a calendar routine that identifies day types and possibly extra fixed holidays taken as Sundays (Christmas, 1st January, ...). Beware that these days are for France. One may want to edit this subprogram (idaytype in src_[model]/Bcalendar.f) and changes a few specific dates.

In this second stage there is a possibility to apply a flat reduction (or increase) on all primary pollutants, in order to account for time shift between the simulation year and the inventory year. Unfortunately the possibility is not given to distinguish factors for various pollutants, but this is not a difficult change in **src/prepemis_std.f**.

6.4 Biogenic emission interface

The **BEMISSIONS.nc** required by the model file should contain 2D hourly biogenic emissions with Isoprene, Terpene and Nitrogen Monoxide.
It is created by the **src/diagbio.f90** program using as input the **BIOFACS_[domain]** biogenic emission potentials file.
The format of the **BEMISSIONS.nc** netCDF file is given in **Appendix F.4, p.95**, with 2D records containing the biogenic emission rates in molec/cm2/s.
Biogenic emissions are calculated in the **calculate_bioemissions** subroutine from **src/diagbio_science.f90** module, which is called by **src/diagbio.f90**.

6.5 Boundary conditions interface

Boundary conditions must be provided to CHIMERE in 2 files constructed dynamically from the top calling script :

- **LAT_CONCS.nc** : A netCDF file containing hourly lateral mixing ratios in ppbv. Its structure is given in **Appendix F.5, p.95**. The list of lateral boundary species is read by CHIMERE from the file itself.
- **TOP_CONCS.nc** : A netCDF file containing hourly top mixing ratios in ppbv. Its structure is given in **Appendix F.6, p.96**. The list of top boundary species is read by CHIMERE from the file itself.

There are basically three options for building these files :

- Either boundary conditions are provided by the user from external data, in which case she/he should only follow the guidelines on the format of files.
- This is a "nest run" and a coarse-resolution run is used for boundary conditions (the CHIMERE option).
- This is a "coarse run" and climatological or monthly gas boundary conditions from MOZART II or LMDZINCA2 are used.

In the case of a nest run, the **preplat_chimere.f90** and **preptop_chimere.f90** interfaces are used.
In the case of a coarse run when using MOZART II or LMDZINCA2, then the process involves three steps. The first consists in building the gas boundary files **LAT_CONCS_gas.nc** and **TOP_CONCS_gas.nc** with the interfaces :

- preplat_mozart2.f90
- preplat_lmdzinca2.f90
- preptop_mozart2.f90
- preptop_lmdzinca2.f90

The second step consists in building the aerosols boundary conditions **LAT_CONCS_aer.nc** and **TOP_CONCS_aer .nc** with the interfaces :

- preplat_gocart.f90
- preptop_gocart.f90

The last step, uses a routine, **src/concat.f90**, to "concatenate" in the netCDF format the corresponding gas and aerosols files. All these steps are driven from the top calling script.
When nesting (1-way) a CHIMERE high-resolution simulation into a coarse-resolution one, the coarse simulation must be complete (**out.[label].nc** file). The address of the coarse-run simulation file is given in the top calling script, as well as the option for nesting and other necessary parameters of the coarse run.

6.6 Initial Conditions

There are two types of initial conditions that can be used for CHIMERE : Initialized concentrations and non initialized concentrations. In the first case a file named **ini.[label].nc** and placed in the simulation directory (the same as where the CHIMERE output files **out.[label].nc**, **par.[label]** and **end.[label]** are). In the second case there is no need to provide a file ; initial conditions will be taken as interpolated from the boundaries, layer by layer. There is a flag (**iopinit**), to be passed to **chimere.f** by the top calling script saying whether simulation is initialized or not.
It is clear that initialization is important for large-scale simulations (domains larger than about 500 Km $\times$ 500 km). Under extremely stagnant conditions, spin-up of one or two days may also be necessary at local (100 Km) scale. At regional scale, the spin-up time is in the order of 4-5 days (for ozone and precursors.
The format of the **ini.[label].nc** file is exactly the same as that of the **end.[label].nc** file. It contains the concentrations encoded in the netCDF format, described in **Appendix F.7, p.96**.
Concatenating short simulations in order to perform a long-term simulation is really easy : At the end of each simulation the **end.[previous_label].nc** can be copied or linked to the **ini.[new_label].nc** file.

Appendix A

References

A.1 Publications using CHIMERE

2005

- Hodzic A., Vautard, R., Bessagnet, B., Lattuati, M., Moreto, F., Long-term urban aerosol simulation versus routine particulate matter observations. Atmospheric Environment, 39, 5851-5864, 2005
- Vautard R., B.Bessagnet, M.Chin and L.Menut, On the contribution of natural Aeolian sources to particulate matter concentrations in Europe: testing hypotheses with a modelling approach, Atm. Env., 39, Issue 18, 3291-3303, 2005
- Menut L., I. Coll and S. Cautenet, Impact of meteorological data resolution on the forecasted ozone concentrations during the ESCOMPTE IOP 2a and 2b, Atmospheric Research - ESCOMPTE Special Issue, 74, 139-159, 2005.

2004

- Bessagnet B., A.Hodzic, R.Vautard, M.Beekmann, S.Cheinet, C.Honoré, C.Liousse and L.Rouil, Aerosol modeling with CHIMERE - preliminary evaluation at the continental scale, Atmospheric Environment, 38, 2803-2817, 2004
- Blond, N. and R.Vautard, Three-dimensional ozone analyses and their use for short-term ozone forecasts, J. Geophys. Res., 109, D17303, doi:10.1029/2004JD004515, 2004.
- Wackernagel, H., C. Lajaunie, N. Blond, C. Roth and R. Vautard, Geostatistical risk mapping with chemical transport model output and ozone station data, Ecological Modelling, 179, 177-185, 2004.
- Hodzic A., Chepfer H., Vautard R., Chazette P., Beekmann M., Bessagnet B., Chatenet B., Cuesta J., Drobinski P., Goloub P., Haeffelin M., Morille Y., 2004: Comparisons of aerosol chemistry-transport model simulations with lidar and sun photometer observations at a site near Paris., J. Geophys. Res., 109, D23201,doi:10.1029/2004JD004735, 2004
- Konovalov, I., Beekmann, M., Vautard, R., Burrows, J., Richter, A., Nuess, H., and N. Nelansky, 2004: The comparison and evaluation of modelled and GOME measurement derived tropospheric NO2 columns over western and eastern Europe Atmospheric Chemistry and Physics, in press.
- Labouze, E., C. Honoré, L. Moulay, B. Couffignal, and M. Beekmann, 2004. Photochemical ozone creation potentials: a new set of characterization factors for different gas species at the scale of the Western Europe, Int. Journal of Life Cycle Assessment, Vol.9, Nr.3, 187-195.
- Monteiro, A., Borrego, C., Vautard, R., and A. I. Miranda, 2004: long-term simulations of photooxidant pollution over Portugal using the CHIMERE model. Atmospheric Environment, submitted.

- Pison I. and L.Menut, Quantification of the impact of aircraft traffic emissions on tropospheric ozone over Paris area, Atmospheric Env., 38-7, 971-983, 2004
- Vautard, R., C. Honoré, M. Beekmann, L. Rouil, 2004: Simulation of ozone during the August 2003 heat wave and emission control scenarios, Atmospheric Environment, in press.
- Zhang, B.-N., R. Vautard and N.-T. Kim Oanh, 2004: Comparison of the UAM-V/SAIMM and CHIMERE/ECMWF Photochemical Modeling System Performance in a Tropical Urban Area, J. Air and Waste Management, in press

2003

- Beekmann M. and C.Derognat, Monte carlo uncertainty analysis of a regional scale transport chemistry model constrained by measurements from the ESQUIF campaign, Journal of Geophysical Research, ESQUIF special section, 2003
- Blond, N., Bel, L. and Vautard, R., Three-dimensional ozone data analysis with an air quality model over the Paris area, J. Geophys. Res., 108(D23), 4744, doi:10.1029/2003JD003679, 2003.
- Derognat C., M.Beekmann, H.Schmidt, R.Vautard, D.Martin, B.Neininger, Effect of biogenic VOC emissions on the tropospheric chemistry during elevated ozone periods in Ile-de-france, Journal of Geophysical Research, ESQUIF special section, 2003
- Menut L., Adjoint modelling for atmospheric pollution processes sensitivity at regional scale during the ESQUIF IOP2, Journal of Geophysical Research, ESQUIF special section, 2003
- Sillman S., R. Vautard, L. Menut and D. Kley, O3-NOx-VOC sensitivity and NOx-VOC indicators in Paris: results from models and ESQUIF measurements, Journal of Geophysical Research, ESQUIF special section, 2003
- Schmidt H. and D.Martin, Continental scale adjoint sensitivity of ozone concentrations in the paris region during the ESQUIF episodes, Journal of Geophysical Research, ESQUIF special section, 2003
- Vautard R., D. Martin, M. Beekmann, R. Friedrich, A. Jaubertie, D. Kley, M. Lattuati, P. Moral, B. Neininger and J. Theloke, Paris emission inventory diagnostic from the ESQUIF airborne measurements and a chemistry-transport model, Journal of Geophysical Research, ESQUIF special section, 2003
- Vautard R. et al., A synthesis of the ESQUIF field campaign, Journal of Geophysical Research, ESQUIF special section, 2003

2001

- Schmidt H., C. Derognat, R.Vautard and M.Beekmann, 'A comparison of simulated and observed ozone mixing ratios for the summer of 1998 in western Europe', Atmospheric Environment, 2001
- Vautard R., M. Beekmann, J. Roux, D. Gombert, 'Validation of a deterministic forecasting system for the ozone concentrations over the Paris area', Atmospheric Environment, 35 2449-2461, 2001.

2000

- Honoré, C., Vautard, R., Beekmann, M., Low and High Nox chemical regimes in an urban environment, Environmental Modelling and Software, 15, 559-564, 2000
- Honoré, C., Vautard, R. and Beekmann, M., Photochemical regimes in urban atmospheres: the influence of dispersion. Geophys. Res. Letters, 27, 1895-1898, 2000.
- Menut L., R.Vautard, M.Beekmann and C.Honoré, 'Sensitivity of Photochemical Pollution using the Adjoint of a Simplified Chemistry-Transport Model', Journal of Geophysical Research - Atmospheres 105, D12, 15,379-15,402, 2000

- Menut L., R. Vautard, C. Flamant, C. Abonnel, M. Beekmann, P. Chazette, P. H. Flamant, D. Gombert, D. Guedalia, D. Kley, M.P. Lefebvre, B. Lossec, D. Martin, G. Megie, M. Sicard, P. Perros, G. Toupance, 'Atmospheric pollution over the Paris area: The ESQUIF project', Annales Geophysicae, 18, 1467-1481, 2000
- Vautard R., M.Beekmann and L. Menut, 'Applications of adjoint modelling in atmospheric chemistry: sensitivity and inverse modelling', Environmental Modeling and Sofware, 15, pp 703-709, 2000

A.2 Useful links

Here are reported useful URL adresses for the development and the use of the CHIMERE model and its modules.

Models	
MM5 model (free)	http://box.mmm.ucar.edu/mm5
NCAR	http://ncar.ucar.edu
ISORROPIA	http://nenes.eas.gatech.edu/ISORROPIA
Librairies	
Unidata netCDF library (free)	http://www.unidata.ucar.edu/
Unidata Udunits library (free)	http://www.unidata.ucar.edu/
Graphics softwares	
GrADS graphical software (free)	http://grads.iges.org/grads/grads.html
GMT software (free)	http://gmt.soest.hawaii.edu/
Fortran compilers	
g95 compiler (free)	http://g95.sourceforge.net/
ifort compiler	http://www.intel.com/cd/software/products/asmo-na/eng/compilers/index.htm
pgf90 compiler	www.pgroup.com
Databases	
EMEP	http://www.emep.int
Global Land Cover Facility (GLCF)	http://glcf.umiacs.umd.edu/data/landcover
LAI from NASA/EOSDIS	http://www.daac.ornl.gov
Operational air quality forecasts with CHIMERE	
Western Europe	http://prevair.ineris.fr
The Paris Region	http://www.airparif.asso.fr
The Alsace Region	http://www.atmo-alsace.net
The Lyon Region	http://www.atmo-rhonealpes.org
The Toulouse Region	http://www.oramip.org

Bibliography

[Atkinson et al., 1997] Atkinson, R., Baulsch, D. L., Cox, R. A., Hampton, R. F., Kerr, J. A., Rossi, M. J., and Troe, J. (1997). Evaluated kinetics, photochemical and heterogeneous data. *J. Phys. Chem.*, 26(3):521–1012.

[Aumont et al., 2003] Aumont, B., Chervier, F., and Laval, S. (2003). Contribution of HONO to the NOx/HOx/O_3 chemistry in the polluted boundary layer. *Atmos. Environ.*, 37:487–498.

[Baer and Nester, 1992] Baer, M. and Nester, K. (1992). Parameterization of trace gas dry deposition velocities for a regional mesoscale diffusion model. *Ann. Geophys.*, 10:912–923.

[Berge, 1993] Berge, E. (1993). Coupling of wet scavenging of sulphur to clouds in a numerical weather prediction model. *Tellus*, 45B:1–22.

[Carter, 1990] Carter, W. P. L. (1990). A detail mechanism for the gas-phase atmospheric reactions of organic compounds. *Atmos. Environ.*, 24:481–518.

[Cheinet, 2002] Cheinet, S. (2002). *The parameterization of clear and cloudy convective boundary layers*. Ph.D. Thesis, Ecole Polytechnique, Paris, France.

[Colella and Woodward, 1984] Colella, P. and Woodward, P. R. (1984). The piecewise parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics*, 11:38–39.

[De Moore et al., 1994] De Moore, W. B., Sandetr, S. P., Golden, D. M., Hampton, R. F., Kurylo, M. J., Howard, C. J., Ravishankara, A. R., Kolb, C. E., and Molina, M. J. (1994). Chemical kinetics and photochimical data for use in stratospheric modelling evaluation. *JPL publication*, 94, 26, JPL, Pasadena, US.

[Derognat et al., 2003] Derognat, C., Beekmann, M., Baeumle, M., Martin, D., and Schmidt, H. (2003). Effect of biogenic volatile organic compound emissions on tropospheric chemistry during the Atmospheric Pollution Over the Paris Area(ESQUIF) campaign in the Ile-de-France region. *J. of Geophys. Res.*, 108(D17):8560.

[Erisman et al., 1994] Erisman, J. W., van Pul, A., and Wyers, P. (1994). Parameterization of surface resistance for the quantification of atmospheric deposition of acidifying pollutants and ozone. *Atmos. Env.*, 28:2595–2607.

[Gelbard and Seinfeld, 1980] Gelbard, F. and Seinfeld, J. H. (1980). Simulation of multicomponent aerosol dynamics. *Journal of colloid and Interface Science*, 78:485–501.

[Griffin et al., 2002a] Griffin, R. J., Dabdub, D., and H., S. J. (2002a). Secondary organic aerosol: 1. Atmospheric chemical mechanism for production of molecular constituents. *J. Geophys. Res.*, 107(D17):4332. doi:10.1029/2001JD000541.

[Griffin et al., 2002b] Griffin, R. J., Dabdub, D., Kleeman, M. J., Fraser, M. P., Cass, G. R., and H., S. J. (2002b). Secondary organic aerosol: 3. Urban/regional scale model of size- and composition-resolved aerosols. *J. Geophys. Res.*, 107(D17):4334. doi:10.1029/2001JD000544.

[Grosjean and Seinfeld, 1989] Grosjean, D. and Seinfeld, J. (1989). Parameterization of the formation potential of secondary organic aerosols. *Atmos. Environ.*, 23:1733–1747.

[Guelle et al., 1998] Guelle, W., Balkanski, Y. J., Dibb, J. E., Schulz, M., and Dulac, F. (1998). Wet deposition in a global size-dependent aerosol transport model. 2. Influence of the scavenging scheme on Pb vertical profiles, and deposition. *J. Geophys. Res.*, 103(D22):28875–28891.

[Günther, 1997] Günther, A. (1997). Seasonal and spatial variations in the natural volatile organic compound emissions. *Ecol. Appl.*, 7(1):34–45.

[Hall et al., 1999] Hall, S., Khudaish, E., and Hart, A. (1999). Electrochemical oxidation of hydrogen peroxide at platinum electrodes. part iii: Effect of temperature. electrochim. *Electrochim. Acta*, 44:2455–2462.

[Harrison and Kito, 1990] Harrison, R. and Kito, A. (1990). Field intercomparison of filter pack and denuder sampling methods for reactive gaseous and particulate pollutants. *Atmos. Environ.*, 24:2633–2640.

[Hicks et al., 1987] Hicks, B. B., Baldocchi, D. D., Meyers, T. P., Hosker Jr., R. P., and Matt, D. R. (1987). A preliminary multiple resistance routine for deriving dry deposition velocities from measured quantities. *Water, Air, Soil Pollut.*, 36.

[Hoffman and Calvert, 1985] Hoffman, M. R. and Calvert, J. G. (1985). Chemical transformation modules for eulerian acid deposition models,. EPA/600/3-85/017.

[Horowitz et al., 2003] Horowitz, L. W., Walters, S., Mauzeralles, D. Z., Emmonds, L. K., Rash, P. J., Granier, C., Tie, X., Lamarque, J. F., Schultz, M. G., and P., B. J. (2003). A global simulation of tropospheric ozone and related tracers: Description and evaluation of MOZART, version 2. *J. Geophys. Res.*, 108(D24):4784.

[Jacob, 2000] Jacob, D. J. (2000). Heterogeneous chemistry and tropospheric ozone. *Atmos. Environ.*, 34(34):2131–2159.

[Kavouras et al., 1998] Kavouras, I., Mihalopoulos, N., and Stephanou, E. (1998). Formation of atmospheric particles from organic acids produced by forests. *Nature*, 395:683–686.

[Kulmala et al., 1998] Kulmala, M., A., L., and Pirjola, L. (1998). Parameterization for sulfuric acid / water nucleation rates. *J. Geophys. Res.*, 103(No D7):8301–8307.

[Lattuati, 1997] Lattuati, M. (1997). *Contribution à l'étude du bilan de l'ozone troposphérique à l'interface de l'Europe et de l'Atlantique Nord: modélisation lagrangienne et mesures en altitude*. Thèse de sciences, Université Paris 6, France.

[Lee and Schwartz, 1983] Lee, Y. N. and Schwartz, S. E. (1983). *Precipitation scavenging, dry deposition and resuspension, vol. 1.*, chapter Kinetics of oxidation of aqueous sulfur (IV) by nitrogen dioxide. Elsevier, New York.

[Loosmore and Cederwall, 2004] Loosmore, G. and Cederwall, R. (2004). Precipitation scavenging of atmospheric aerosols for emergency response applications: testing an updated model with new real-time data. *Atmos. Env.*, 38:993–1003.

[Louis, 1979] Louis, J. (1979). A parametric model of vertical eddy fluxes in the atmosphere. *Boundary Layer Meteorology*, pages 187–202.

[Louis et al., 1982] Louis, J., Tiedke, M., and Geleyn, J. (1982). A short history of the pbl parametrization at ecmwf. *ECMWF Workshop on Planetary Boundary Layer parametrization*, pages 59–80.

[Madronich et al., 1998] Madronich, S., McKenzie, R. E., Bjorn, L. O., and Caldwell, M. M. (1998). Changes in biologically active ultraviolet radiation reaching the earth's surface. *J. Photochem. Photobiol. B:Biology*, 46:5–19.

[McRae et al., 1982] McRae, G., Goodin, W., and Seinfeld, J. (1982). Development of a second generation mathematical model for urban air pollution: I. model formulation. *Atmospheric Environment*, 16:679–696.

[Middleton et al., 1990] Middleton, P., Stockwell, W. R., and Carter, W. P. (1990). Agregation and analysis of volatile organic compound emissions for regional modelling. *Atmos. Environ.*, 24:1107–1133.

[Mircea and Stefan, 1998] Mircea, M. and Stefan, S. (1998). A theoretical study of the microphysical parameterization of the scavenging coefficient as a function of precipitation type and rate. *Atmos. Env.*, 32:2931–2938.

[Mylona, 1999] Mylona, S. (1999). EMEP emission data, status report 1999. Technical Report EMEP/MSC-W Note 1/99, EMEP, Oslo, Norway.

[Nenes et al., 1998] Nenes, A., Pilinis, C., and Pandis, S. (1998). ISORROPIA: A new thermodynamic model for inorganic multicomponent atmospheric aerosols. *Aquatic Geochem.*, 4:123–152.

[Odum et al., 1996] Odum, J. R., Hoffman, T., Bowman, F., Collins, D., Flagan, R., and Seinfeld, J. (1996). Gas/Particle partitioning and secondary organic aerosol yields. *Environ. Sci. Technol.*, 30:2580–2585.

[Odum et al., 1997] Odum, J. R., Jungkamp, T. P. W., Griffin, R. J., Forster, H. J. L., Flagan, R. C., and Seinfeld, J. H. (1997). Aromatics, reformulated gasoline and atmospheric organic aerosol formation. *Environ. Sci. Technol.*, 31:1890–1897.

[Pankow, 1994] Pankow, J. F. (1994). An absorption model of gas/aerosol partition involved in the formation of secondary organic aerosol. *Atmos. Environ.*, 28:189–193.

[Pankow et al., 2001] Pankow, J. F., Seinfeld, J. H., Asher, W. E., and Erdakos, G. B. (2001). Modeling the formation of secondary organic aerosol. 1. Application of theoretical principles to measurements obtained in the α-pinene, β-pinene, sabinene, δ^3-carene and cyclohexene/ozone systems. *Environmental Sci. and Tech.*, 35:1164–1172.

[Priestley and Taylor, 1967] Priestley, C. H. B. and Taylor, R. J. (1967). On the assessment of surface heat flux and evaporation using large-scale parameters. *Monthly Weather Review*, 2:81–93.

[Pun et al., 2002] Pun, B. K., Griffin, R. J., Seigneur, C., and Seinfeld, J. H. (2002). Secondary organic aerosol 2. Thermodynamic model for gas/particle partitioning of molecular constituents. *J. Geophys. Res.*, 107(D17):4333.

[Sander et al., 1999] Sander, G.C. and. Parlange, J.-Y., Smith, R.E.and Haverkamp, R., and Hogarth, W. (1999). Estimation of ponding time for constant surface flux. *Hydrology days Ed. AGU Pub. 19*, pages 402–410.

[Schell et al., 2001] Schell, B., Ackermann, I. J., Hass, H., Binkowski, F. S., and Ebel, A. (2001). Modeling the formation of secondary organic aerosol within a comprehensive air quality modeling system. *J. Geophys. Res.*, 106:28275–28293.

[Seinfeld and Pandis, 1997] Seinfeld, J. H. and Pandis, S. N. (1997). *Atmospheric chemistry and physics: From air pollution to climate change*. Wiley-Interscience.

[Simpson, 1992] Simpson, D. (1992). Long period modeling of photochemical oxidants in Europe, Calculations for July 1985. *Atmos. Environ.*, 26:1609–1634.

[Simpson, 1999] Simpson, D. (1999). Inventorying emissions from nature in Europe. *J. Geophys. Res.*, 104:8113–8152.

[Society, 1994] Society, E. I. (1994). Generation of European Emission Data for Episodes (GENEMIS) project, EUROTRAC annual report 1993, part 5. Technical report, EUROTRAC, Garmish-Partenkirchen, Germany.

[Stohl et al., 1996] Stohl, A., Williams, E., Wotawa, G., and Kromp-Kolb, H. (1996). A European inventory of soil nitric oxide emissions and the effect of these emissions on the photochemical formation of ozone in Europe. *Atmos. Environ.*, 30:741–3,755. 3.

[Troen and Mahrt, 1986] Troen, I. and Mahrt, L. (1986). A simple model of the atmospheric boundary layer: Sensitivity to surface evaporation. *Bound.-Layer Meteorol.*, 37:129–148.

[Tsyro, 2002] Tsyro, S. (2002). First estimates of the effect of aerosol dynamics in the calculation of PM_{10} and $PM_{2.5}$. *EMEP report*.

[Verwer, 1994] Verwer, J. (1994). Gauss-seidel iterations for stiff odes from chemical kinetics. *SIAM Journal of Scientific Computing*, 15:1243–1250.

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

Appendix B

How To install netCDF under GNU/Linux

B.1 Background

Since version V200510 CHIMERE requires the netCDF package to work. This note describes how to install netCDF for the supported compilers.
NetCDF installation has been tested with :

- ifort 32 bits (version 8.1.028)
- ifort 64 bits (version 8.1.027)
- g95 32 bits (version 4.0.0 20050129)
- g95 64 bits (version 4.0.0 20050124)
- pgf90 32 bits (version 5.2.4)
- netcdf (version 3.6.0-p1)

B.2 Download

- **Download netCDF** (source package) from the URL :

```
cd /tmp
wget ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf.tar.gz
```

- **Decompress and detar**

```
tar xvfz netcdf.tar.gz
```

B.3 Configure netCDF

At the time of writing of this How-To, the current version of netCDF is 3.6.0-p1. If you downloaded a newer version, replace 3.6.0-p1 by the new version number in the following text.

```
cd netcdf-3.6.0-p1/src
```

The configuration process depends on several environment variables that you shall set according to your compiler. Although it may be possible to link CHIMERE executables compiled with compiler A with the netCDF library compiled with compiler B, we strongly recommend to use the same compiler. The time you will spend to install a new netCDF library is negligible compared to the time you could loose trying to match inconsistent libraries.
We suppose you want to install netcdf somewhere under `/opt` . If you choose to install under `/usr/local`, or under you home directory, replace `/opt` your installation path in the following text.
We also suppose your shell is like `bash` or `ksh` . If you use `csh`, modify the `export` statements according to your shell syntax.

B.3.1 ifort

As root :

```
mkdir /opt/netcdf-3.6.0-p1-ifort
```

As an ordinary user :

```
export FC=ifort
export F90=ifort
export FFLAGS="-mp"
export CPPFLAGS="-DNDEBUG -DpgiFortran"
./configure --prefix=/opt/netcdf-3.6.0-p1-ifort
make
make test
```

As root :

```
make install
```

B.3.2 g95

As root :

```
mkdir /opt/netcdf-3.6.0-p1-g95
```

As an ordinary user :

```
export CC='gcc'
export CXX='g++'
export CPPFLAGS='-DNDEBUG -DpgiFortran'
export CFLAGS='-O -g'
export FC='g95'
export F90='g95'
export FFLAGS='-O -g -fno-second-underscore'
export F90FLAGS='-O -g -fno-second-underscore'
./configure --prefix=/opt/netcdf-3.6.0-p1-g95
make
make test
```

As root :

```
make install
```

B.3.3 pgf90

Some versions of pgf90 prior to 5.2 have experienced problems with the F90 interface of netCDF. Do not try to install a library for which "make test" has failed.
As root :

```
mkdir /opt/netcdf-3.6.0-p1-pgf90
```

As an ordinary user :

```
export CC=/usr/bin/gcc
export CPPFLAGS='-DNDEBUG -DpgiFortran'
export CFLAGS=-O
export FC=pgf90
export FFLAGS="-O -w"
export CXX=g++
./configure --prefix=/opt/netcdf-3.6.0-p1-pgf90
make
make test
```

As root :

```
make install
```

B.3.4 ifort 64 bit on Intel EMT64 or AMD Opteron

As root :

```
mkdir /opt/netcdf-3.6.0-p1-ifort
```

As an ordinary user :

```
export FC=ifort
export F90=ifort
export CFLAGS="-O -m64"
export FFLAGS="-mp"
export CPPFLAGS="-DNDEBUG -DpgiFortran"
./configure --prefix=/opt/netcdf-3.6.0-p1-ifort
make
make test
```

As root :

```
make install
```

B.3.5 g95 64 bit on Intel EMT64 or AMD Opteron

As root :

```
mkdir /opt/netcdf-3.6.0-p1-g95-64
```

As an ordinary user :

```
export CC='gcc'
export CXX='g++'
export CPPFLAGS='-DNDEBUG -DpgiFortran'
export CFLAGS='-O -m64'
export FC='/usr/local/g95-64/bin/x86\_64-unknown-linux-gnu-g95'
export F90='/usr/local/g95-64/bin/x86\_64-unknown-linux-gnu-g95'
export FFLAGS='-O -g -fno-second-underscore'
export F90FLAGS='-O -g -fno-second-underscore'
./configure --prefix=/opt/netcdf-3.6.0-p1-g95-64
make
make test
```

As root :

```
make install
```

B.4 Configure CHIMERE

In the directory `makefiles.hdr/`, you will find 6 files :

```
Makefile.hdr.ifort
Makefile.hdr.ifort-64
Makefile.hdr.g95
Makefile.hdr.g95-64
Makefile.hdr.pgf90
Makefile.hdr.pgf90-64
```

In the file corresponding to your compiler, modify the line `NETCDF =` to point to your netCDF directory. For instance :

```
NETCDF = /opt/netcdf-3.6.0-p1-g95
```

Follow the comments in `Makefile.hdr.xxxx` for the other options. Do not forget to link your compiler-specific `Makefile.hdr.xxxx` to `Makefile.hdr` . For instance :

```
ln -sf makefiles.hdr/Makefile.hdr.ifort Makefile.hdr
```

Appendix C

How To install Udunits under GNU/Linux

C.1 Background

Since version V200510 of CHIMERE a new analysis tool, `exachim`, has been included in the package. `exachim` is optional, but requires the "Udunits" package to work.

C.2 Download

- **Download Udunits** (source package) from the URL :

```
cd /tmp
wget ftp://ftp.unidata.ucar.edu/pub/udunits/udunits.tar.Z
```

- **Decompress and detar**

```
tar xvfz udunits.tar.Z
```

C.3 Configure Udunits

At the time of writing of this How-To, the current version of udunits is 1.12.4. If you downloaded a newer version, replace 1.12.4 by the new version number in the following text.

```
cd udunits-1.12.4/src
```

The configuration process depends on several environment variables that you shall set according to your compiler.
We suppose you want to install udunits somewhere under `/opt`. If you choose to install under `/usr/local`, or under you home directory, replace `/opt` by your installation path in the following text.
We also suppose your shell is like `bash` or `ksh` . If you use `csh`, modify the `export` statements according to your shell syntax.

C.3.1 gcc on a 32 bit machine

As root :

```
mkdir /opt/udunits-1.12.4
```

As an ordinary user :

```
export CFLAGS=-O
export LDFLAGS="-lm"
export CPPFLAGS="-DNDEBUG -DpgiFortran  -D_GNU_SOURCE"
./configure --prefix=/opt/udunits-1.12.4
make
make test
```

As root :

```
make install
```

C.3.2 gcc on a 64 bit machine

As root :

```
mkdir /opt/udunits-1.12.4-64
```

As an ordinary user :

```
export CFLAGS="-O -m64 -fPIC"
export LDFLAGS="-lm"
export CPPFLAGS="-DNDEBUG -DpgiFortran  -D_GNU_SOURCE"
./configure --prefix=/opt/udunits-1.12.4-64
make
make test
```

As root :

```
make install
```

C.4 configure CHIMERE

At the top level of CHIMERE tree, you will find 6 files :

```
Makefile.hdr.ifort
Makefile.hdr.ifort-64
Makefile.hdr.g95
Makefile.hdr.g95-64
Makefile.hdr.pgf90
Makefile.hdr.pgf90-64
```

In the file corresponding to your compiler, modify the line `UDUNITS =` to point to your Udunits directory. For instance :

```
UDUNITS =       /opt/udunits-1.12.4
```

Appendix D

The color palettes

The predefined palettes are described in the **Table D.1**.

Filename	Min	Max	Step
CH4.cpt	1700	1900	20
CO.cpt	0	1000	50
HNO3.cpt	0	20	1
NO.cpt	0	50	2.5
NO2.cpt	0	150	10
NOY.cpt	0	150	10
O3.cpt	0	150	10/50
PAN.cpt	0	6	0.3
pHNO3-reg.cpt	0	50	5
pASOA.cpt	0	200	5/10/50
pH2SO4.cpt	0	200	5/10/50
pHNO3.cpt	0	200	5/10/50
PM10.cpt	0	200	5/10/50
PM25.cpt	0	200	5/10/50
pNH3.cpt	0	200	5/10/50
pNH3-reg.cpt	0	20	2
pPPM.cpt	0	200	5/10/50
SO2.cpt	0	100	10

Table D.1: *The predefined color tables to be used with the GMT software for concentrations maps*

These palettes were defined to be the most adaptated to 'usual' regional surface concentrations. You can create a new palette by using the following command line:

makecpt -C TABLE -T vmin/vmax/step > OUT_PALETTE_FILE

with:

- TABLE: The GMT predefined color tables: type 'makecpt' in a x-terminal to have the complete list. You can use, for example:
  - gray: Grayramp from black to white [0/1]

- rainbow: Rainbow colors: magenta-blue-cyan-green-yellow-red
- cool: Linear change from blue to magenta

- vmin/vmax/step: the minimum/maximum and step values (examples are provided in the **Table D.1**)
- OUT_PALETTE_FILE the output file for this new color palette. Please note that if the name is not composed of the species name and the ".cpt" extension, you have to change the corresponding line in the `mapit` script.

Appendix E

NetCDF Conventions

What are conventions in netCDF ?
It is in general not sufficient to specify "netCDF" to describe the structure of a netCDF file. NetCDF defines a programmation interface, i.e. a set of Fortran calls to write and read data to/from a file. NetCDF considers data as arrays, and a wide freedom exists to define the dimensions of these arrays, to name these dimensions, to define their units. NetCDF allows also for including "Attributes" in the header of the file.
What attributes ? what meaning ?
CONVENTIONS restrict the degrees of freedom by answering these questions and defining rules. A netCDF convention is generally specific to a community. If a file follows the requirements of a convention, and if an analysis tool is aware of this convention, then all the implied knowledge about dimensions, attributes, variables and so on are automatically shared between the file and the analysis tool, greately simplifying the usage of the file. COARDS is one of these conventions, and is of widespread use in meteorology/climatology.
Why does not CHIMERE natively produces COARDS compliant files ?
COARDS imposes rectilinear coordinate systems in longitude/latitude. CHIMERE is more general, and may work in other coordinate systems.
More precisely, the mesoscale models many people use to provide CHIMERE with high resolution forecasts (MM5, WRF) use curvilinear projections like Lambert conformal, Polar stereographic or Mercator. This is the main reason why we have choosen to stick to another convention, which is the one used in NCAR's WRF Mesoscale Meteorological Model.

Appendix F

Structure of the CHIMERE netCDF files

F.1 exdomout.nc

```
netcdf exdomout {
dimensions:
Time = UNLIMITED ; // (13 currently)
DateStrLen = 19 ;
west_east = 67 ;
south_north = 46 ;
bottom_top = 32 ;
variables:
char Times(Time, DateStrLen) ;
float lon(south_north, west_east) ;
lon:units = "degrees_east" ;
lon:long_name = "Longitude" ;
float lat(south_north, west_east) ;
lat:units = "degrees_north" ;
lat:long_name = "Latitude" ;
float sshf(Time, south_north, west_east) ;
sshf:units = "W/m^2" ;
sshf:long_name = "Surface sensible heat flux" ;
float slhf(Time, south_north, west_east) ;
slhf:units = "W/m^2" ;
slhf:long_name = "Surface latent heat flux" ;
float usta(Time, south_north, west_east) ;
usta:units = "m/s" ;
usta:long_name = "Frictional velocity" ;
float tem2(Time, south_north, west_east) ;
tem2:units = "K" ;
tem2:long_name = "2m air temperature" ;
float soim(Time, south_north, west_east) ;
soim:units = "m^3/m^3" ;
soim:long_name = "Soil Moisture level 1" ;
float hght(Time, south_north, west_east) ;
hght:units = "m" ;
hght:long_name = "PBL height from MM5" ;
float rh2m(Time, south_north, west_east) ;
rh2m:units = "fraction" ;
rh2m:long_name = "Relative Humidity at 2m" ;
float lspc(Time, south_north, west_east) ;
lspc:units = "kg/m^2" ;
```

```
lspc:long_name = "Large scale precipitation" ;
float copc(Time, south_north, west_east) ;
copc:units = "kg/m^2" ;
copc:long_name = "convective precipitation" ;
float u10m(Time, south_north, west_east) ;
u10m:units = "m/s" ;
u10m:long_name = "10 m U wind" ;
float v10m(Time, south_north, west_east) ;
v10m:units = "m/s" ;
v10m:long_name = "10 m V wind" ;
float temp(Time, bottom_top, south_north, west_east) ;
temp:units = "K" ;
temp:long_name = "Temperature" ;
float cliq(Time, bottom_top, south_north, west_east) ;
cliq:units = "kg/kg" ;
cliq:long_name = "Cloud liquid water mixing ratio" ;
float rain(Time, bottom_top, south_north, west_east) ;
rain:units = "kg/kg" ;
rain:long_name = "Rain water mixing ratio" ;
float sphu(Time, bottom_top, south_north, west_east) ;
sphu:units = "kg/kg" ;
sphu:long_name = "Specific humidity" ;
float cice(Time, bottom_top, south_north, west_east) ;
cice:units = "kg/kg" ;
cice:long_name = "Ice mixing ratio" ;
float pres(Time, bottom_top, south_north, west_east) ;
pres:units = "Pa" ;
pres:long_name = "Pressure" ;
float alti(Time, bottom_top, south_north, west_east) ;
alti:units = "m" ;
alti:long_name = "Altitude of half-sigma level" ;
float winz(Time, bottom_top, south_north, west_east) ;
winz:units = "m/s" ;
winz:long_name = "Zonal wind" ;
float winm(Time, bottom_top, south_north, west_east) ;
winm:units = "m/s" ;
winm:long_name = "Meridional wind" ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Interpolated Meteo file" ;
:Generating_process = "Generated by prepexmm5" ;
:Conventions = "" ;
:Domain = "CONT3" ;
:history = "File exdomout.nc was generated on",
                   "Tue Nov 15 14:25:27 2005 by <user> on <host>",
                    "from input file EUR1/MMOUT_EUR1_20030730_20030803_S\n" ;
:mm5_nxx = 85 ;
:mm5_nxx_DOT = 86 ;
:mm5_nyy = 75 ;
:mm5_nyy_DOT = 76 ;
:mm5_nlev = 32 ;
}
```

F.2 METEO.nc

```
netcdf METEO {
dimensions:
Time = UNLIMITED ; // (13 currently)
DateStrLen = 19 ;
west_east = 67 ;
south_north = 46 ;
bottom_top = 8 ;
variables:
char Times(Time, DateStrLen) ;
float lon(south_north, west_east) ;
lon:units = "degrees_east" ;
lon:long_name = "Longitude" ;
float lat(south_north, west_east) ;
lat:units = "degrees_north" ;
lat:long_name = "Latitude" ;
float tem2(Time, south_north, west_east) ;
tem2:units = "K" ;
tem2:long_name = "2m air temperature" ;
float atte(Time, south_north, west_east) ;
atte:units = "0-1" ;
atte:long_name = "Cloud attenuation" ;
float hght(Time, south_north, west_east) ;
hght:units = "m" ;
hght:long_name = "PBL height" ;
float usta(Time, south_north, west_east) ;
usta:units = "m/s" ;
usta:long_name = "Frictional velocity" ;
float aerr(Time, south_north, west_east) ;
aerr:units = "*" ;
aerr:long_name = "*" ;
float obuk(Time, south_north, west_east) ;
obuk:units = "m" ;
obuk:long_name = "Obukov length" ;
float wsta(Time, south_north, west_east) ;
wsta:units = "m/s" ;
wsta:long_name = "Wstar" ;
float sreh(Time, south_north, west_east) ;
sreh:units = "0-1" ;
sreh:long_name = "Sfc rel. humidity" ;
float topc(Time, south_north, west_east) ;
topc:units = "kg/m^2" ;
topc:long_name = "Total precip" ;
float w10m(Time, south_north, west_east) ;
w10m:units = "m/s" ;
w10m:long_name = "10m wind" ;
float w10s(Time, south_north, west_east) ;
w10s:units = "m/s" ;
w10s:long_name = "Saltation wind" ;
float soim(Time, south_north, west_east) ;
soim:units = "m^3/m^3" ;
soim:long_name = "Soil moisture" ;
float alti(Time, bottom_top, south_north, west_east) ;
alti:units = "m" ;
alti:long_name = "Altitude of layer" ;
```

```
float winz(Time, bottom_top, south_north, west_east) ;
winz:units = "m/s" ;
winz:long_name = "Zonal wind" ;
float winm(Time, bottom_top, south_north, west_east) ;
winm:units = "m/s" ;
winm:long_name = "Meridional wind" ;
float temp(Time, bottom_top, south_north, west_east) ;
temp:units = "K" ;
temp:long_name = "Temperature" ;
float sphu(Time, bottom_top, south_north, west_east) ;
sphu:units = "kg/kg" ;
sphu:long_name = "Specific humidity" ;
float airm(Time, bottom_top, south_north, west_east) ;
airm:units = "molec/cm3" ;
airm:long_name = "Air density" ;
float kzzz(Time, bottom_top, south_north, west_east) ;
kzzz:units = "*" ;
kzzz:long_name = "Kz" ;
float clwc(Time, bottom_top, south_north, west_east) ;
clwc:units = "Kg/Kg" ;
clwc:long_name = "Cloud wat. content" ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Meteo file" ;
:Generating_process = "Generated by diagmet" ;
:Conventions = "" ;
:Domain = "CONT3" ;
:history = "File METEO.nc was generated on",
                    "Tue Nov 15 14:25:39 2005 by <user> on <host>",
                    "from input file exdomout.nc\n",
                    "File exdomout.nc was generated on",
                    "Tue Nov 15 14:25:27 2005 by <user> on <host>",
                    "from input file MMOUT_EUR1_20030730_20030803_S\n";
}
```

F.3 AEMISSIONS.nc

```
netcdf AEMISSIONS {
dimensions:
Time = UNLIMITED ; // (12 currently)
DateStrLen = 19 ;
west_east = 67 ;
south_north = 46 ;
bottom_top = 2 ;
a_species = 20 ;
variables:
char Times(Time, DateStrLen) ;
float aemission(Time, bottom_top, south_north, west_east, a_species) ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Anthropic emissions file" ;
:Generating_process = "Generated by prepemis" ;
:Conventions = "" ;
```

```
:Domain = "CONT3" ;
:history = "File AEMISSIONS.nc was generated",
                 "on Tue Nov 15 14:25:15 2005 by <user> on <host>",
                 "from input file EMI_aero_2002/2LEVELS/EMISSIONS-CONT3.07  \n" ;
}
```

F.4 BEMISSIONS.nc

```
netcdf BEMISSIONS {
dimensions:
Time = UNLIMITED ; // (13 currently)
DateStrLen = 19 ;
west_east = 67 ;
south_north = 46 ;
biospecies = 11 ;
variables:
char Times(Time, DateStrLen) ;
float lon(south_north, west_east) ;
float lat(south_north, west_east) ;
float emisb(Time, south_north, west_east, biospecies) ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Bioemissions file" ;
:Generating_process = "Generated by calbio" ;
:Conventions = "" ;
:Domain = "CONT3" ;
:history = "File BEMISSIONS.nc",
                  "was generated on Tue Nov 15 14:25:40 2005 ",
                  "by <user> on <host> from input file",
                  "METEO.nc\n",
}
```

F.5 LAT_CONCS.nc

```
netcdf LAT_CONCS {
dimensions:
Time = UNLIMITED ; // (13 currently)
DateStrLen = 19 ;
SpStrLen = 23 ;
bottom_top = 8 ;
h_boundary = 226 ;
Species = 14 ;
variables:
char Times(Time, DateStrLen) ;
char species(Species, SpStrLen) ;
float lon(h_boundary) ;
float lat(h_boundary) ;
float conc(Time, bottom_top, h_boundary, Species) ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Lateral boundary concentrations file" ;
```

```
:Generating_process = "Generated by preplat_LMDZINCA2" ;
:Conventions = "" ;
:Domain = "CONT3" ;
:history = "File LAT_CONCS_gas.nc was generated",
                  "on Tue Nov 15 14:25:21 2005 by <user> on <host>",
                  "from input file LMDZINCA/INCA.07  \n" ;
}
```

F.6 TOP_CONCS.nc

```
netcdf TOP_CONCS {
dimensions:
Time = UNLIMITED ; // (13 currently)
DateStrLen = 19 ;
SpStrLen = 23 ;
horiz1D = 3082 ;
Species = 14 ;
variables:
char Times(Time, DateStrLen) ;
char species(Species, SpStrLen) ;
float lon(horiz1D) ;
float lat(horiz1D) ;
float conc(Time, horiz1D, Species) ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Top concentrations file" ;
:Generating_process = "Generated by preptop_LMDZINCA2" ;
:Conventions = "" ;
:Domain = "CONT3" ;
:history = "File TOP_CONCS.nc was generated on",
                  "Tue Nov 15 14:25:24 2005 by <user> on <host>",
                  "from input file LMDZINCA/INCA.07  \n" ;
}
```

F.7 end.nc

```
netcdf end.20030730_20030803_heat-wave {
dimensions:
Time = UNLIMITED ; // (1 currently)
DateStrLen = 19 ;
SpecStrLen = 16 ;
Species = 119 ;
west_east = 67 ;
south_north = 46 ;
bottom_top = 8 ;
variables:
float lon(south_north, west_east) ;
lon:units = "degrees_east" ;
lon:long_name = "Longitude" ;
float lat(south_north, west_east) ;
lat:units = "degrees_north" ;
lat:long_name = "Latitude" ;
```

```
char species_name(Species, SpecStrLen) ;
char Times(Time, DateStrLen) ;
float APINEN(Time, bottom_top, south_north, west_east) ;
APINEN:units = "molecules/cm3" ;
float C2H4(Time, bottom_top, south_north, west_east) ;
C2H4:units = "molecules/cm3" ;
float C2H6(Time, bottom_top, south_north, west_east) ;
C2H6:units = "molecules/cm3" ;
float C3H6(Time, bottom_top, south_north, west_east) ;
C3H6:units = "molecules/cm3" ;
float C5H8(Time, bottom_top, south_north, west_east) ;
C5H8:units = "molecules/cm3" ;
float CARNIT(Time, bottom_top, south_north, west_east) ;
CARNIT:units = "molecules/cm3" ;
float CH3CHO(Time, bottom_top, south_north, west_east) ;
CH3CHO:units = "molecules/cm3" ;
float CH3COE(Time, bottom_top, south_north, west_east) ;
CH3COE:units = "molecules/cm3" ;
float CH3COO(Time, bottom_top, south_north, west_east) ;
CH3COO:units = "molecules/cm3" ;
float CH3COY(Time, bottom_top, south_north, west_east) ;
CH3COY:units = "molecules/cm3" ;
float CH3O2(Time, bottom_top, south_north, west_east) ;
CH3O2:units = "molecules/cm3" ;
float CH3O2H(Time, bottom_top, south_north, west_east) ;
CH3O2H:units = "molecules/cm3" ;
float CH4(Time, bottom_top, south_north, west_east) ;
CH4:units = "molecules/cm3" ;
float CO(Time, bottom_top, south_north, west_east) ;
CO:units = "molecules/cm3" ;
float DUSTAQ(Time, bottom_top, south_north, west_east) ;
DUSTAQ:units = "molecules/cm3" ;
float GLYOX(Time, bottom_top, south_north, west_east) ;
GLYOX:units = "molecules/cm3" ;
float H2O2(Time, bottom_top, south_north, west_east) ;
H2O2:units = "molecules/cm3" ;
float H2SO4(Time, bottom_top, south_north, west_east) ;
H2SO4:units = "molecules/cm3" ;
float H2SO4AQ(Time, bottom_top, south_north, west_east) ;
H2SO4AQ:units = "molecules/cm3" ;
float HCHO(Time, bottom_top, south_north, west_east) ;
HCHO:units = "molecules/cm3" ;
float HNO3(Time, bottom_top, south_north, west_east) ;
HNO3:units = "molecules/cm3" ;
float HNO3AQ(Time, bottom_top, south_north, west_east) ;
HNO3AQ:units = "molecules/cm3" ;
float HO2(Time, bottom_top, south_north, west_east) ;
HO2:units = "molecules/cm3" ;
float HONO(Time, bottom_top, south_north, west_east) ;
HONO:units = "molecules/cm3" ;
float ISNI(Time, bottom_top, south_north, west_east) ;
ISNI:units = "molecules/cm3" ;
float MAC(Time, bottom_top, south_north, west_east) ;
MAC:units = "molecules/cm3" ;
float MEMALD(Time, bottom_top, south_north, west_east) ;
MEMALD:units = "molecules/cm3" ;
```

```
float MGLYOX(Time, bottom_top, south_north, west_east) ;
MGLYOX:units = "molecules/cm3" ;
float MVK(Time, bottom_top, south_north, west_east) ;
MVK:units = "molecules/cm3" ;
float N2O5(Time, bottom_top, south_north, west_east) ;
N2O5:units = "molecules/cm3" ;
float NC4H10(Time, bottom_top, south_north, west_east) ;
NC4H10:units = "molecules/cm3" ;
float NH3(Time, bottom_top, south_north, west_east) ;
NH3:units = "molecules/cm3" ;
float NH3AQ(Time, bottom_top, south_north, west_east) ;
NH3AQ:units = "molecules/cm3" ;
float NO(Time, bottom_top, south_north, west_east) ;
NO:units = "molecules/cm3" ;
float NO2(Time, bottom_top, south_north, west_east) ;
NO2:units = "molecules/cm3" ;
float NO3(Time, bottom_top, south_north, west_east) ;
NO3:units = "molecules/cm3" ;
float O3(Time, bottom_top, south_north, west_east) ;
O3:units = "molecules/cm3" ;
float obio(Time, bottom_top, south_north, west_east) ;
obio:units = "molecules/cm3" ;
float obioH(Time, bottom_top, south_north, west_east) ;
obioH:units = "molecules/cm3" ;
float OH(Time, bottom_top, south_north, west_east) ;
OH:units = "molecules/cm3" ;
float oPAN(Time, bottom_top, south_north, west_east) ;
oPAN:units = "molecules/cm3" ;
float oRN1(Time, bottom_top, south_north, west_east) ;
oRN1:units = "molecules/cm3" ;
float oRO2(Time, bottom_top, south_north, west_east) ;
oRO2:units = "molecules/cm3" ;
float oROOH(Time, bottom_top, south_north, west_east) ;
oROOH:units = "molecules/cm3" ;
float OXYL(Time, bottom_top, south_north, west_east) ;
OXYL:units = "molecules/cm3" ;
float PAN(Time, bottom_top, south_north, west_east) ;
PAN:units = "molecules/cm3" ;
float PANH(Time, bottom_top, south_north, west_east) ;
PANH:units = "molecules/cm3" ;
float PPA(Time, bottom_top, south_north, west_east) ;
PPA:units = "molecules/cm3" ;
float PPMAQ(Time, bottom_top, south_north, west_east) ;
PPMAQ:units = "molecules/cm3" ;
float SO2(Time, bottom_top, south_north, west_east) ;
SO2:units = "molecules/cm3" ;
float SOA(Time, bottom_top, south_north, west_east) ;
SOA:units = "molecules/cm3" ;
float SOAAQ(Time, bottom_top, south_north, west_east) ;
SOAAQ:units = "molecules/cm3" ;
float toPAN(Time, bottom_top, south_north, west_east) ;
toPAN:units = "molecules/cm3" ;
float p1PPM(Time, bottom_top, south_north, west_east) ;
p1PPM:units = "molecules/cm3" ;
float p1DUST(Time, bottom_top, south_north, west_east) ;
p1DUST:units = "molecules/cm3" ;
```

```
float p1SOA(Time, bottom_top, south_north, west_east) ;
p1SOA:units = "molecules/cm3" ;
float p1H2SO4(Time, bottom_top, south_north, west_east) ;
p1H2SO4:units = "molecules/cm3" ;
float p1HNO3(Time, bottom_top, south_north, west_east) ;
p1HNO3:units = "molecules/cm3" ;
float p1NH3(Time, bottom_top, south_north, west_east) ;
p1NH3:units = "molecules/cm3" ;
float p1WATER(Time, bottom_top, south_north, west_east) ;
p1WATER:units = "molecules/cm3" ;
float p2PPM(Time, bottom_top, south_north, west_east) ;
p2PPM:units = "molecules/cm3" ;
float p2DUST(Time, bottom_top, south_north, west_east) ;
p2DUST:units = "molecules/cm3" ;
float p2SOA(Time, bottom_top, south_north, west_east) ;
p2SOA:units = "molecules/cm3" ;
float p2H2SO4(Time, bottom_top, south_north, west_east) ;
p2H2SO4:units = "molecules/cm3" ;
float p2HNO3(Time, bottom_top, south_north, west_east) ;
p2HNO3:units = "molecules/cm3" ;
float p2NH3(Time, bottom_top, south_north, west_east) ;
p2NH3:units = "molecules/cm3" ;
float p2WATER(Time, bottom_top, south_north, west_east) ;
p2WATER:units = "molecules/cm3" ;
float p3PPM(Time, bottom_top, south_north, west_east) ;
p3PPM:units = "molecules/cm3" ;
float p3DUST(Time, bottom_top, south_north, west_east) ;
p3DUST:units = "molecules/cm3" ;
float p3SOA(Time, bottom_top, south_north, west_east) ;
p3SOA:units = "molecules/cm3" ;
float p3H2SO4(Time, bottom_top, south_north, west_east) ;
p3H2SO4:units = "molecules/cm3" ;
float p3HNO3(Time, bottom_top, south_north, west_east) ;
p3HNO3:units = "molecules/cm3" ;
float p3NH3(Time, bottom_top, south_north, west_east) ;
p3NH3:units = "molecules/cm3" ;
float p3WATER(Time, bottom_top, south_north, west_east) ;
p3WATER:units = "molecules/cm3" ;
float p4PPM(Time, bottom_top, south_north, west_east) ;
p4PPM:units = "molecules/cm3" ;
float p4DUST(Time, bottom_top, south_north, west_east) ;
p4DUST:units = "molecules/cm3" ;
float p4SOA(Time, bottom_top, south_north, west_east) ;
p4SOA:units = "molecules/cm3" ;
float p4H2SO4(Time, bottom_top, south_north, west_east) ;
p4H2SO4:units = "molecules/cm3" ;
float p4HNO3(Time, bottom_top, south_north, west_east) ;
p4HNO3:units = "molecules/cm3" ;
float p4NH3(Time, bottom_top, south_north, west_east) ;
p4NH3:units = "molecules/cm3" ;
float p4WATER(Time, bottom_top, south_north, west_east) ;
p4WATER:units = "molecules/cm3" ;
float p5PPM(Time, bottom_top, south_north, west_east) ;
p5PPM:units = "molecules/cm3" ;
float p5DUST(Time, bottom_top, south_north, west_east) ;
p5DUST:units = "molecules/cm3" ;
```

```
float p5SOA(Time, bottom_top, south_north, west_east) ;
p5SOA:units = "molecules/cm3" ;
float p5H2SO4(Time, bottom_top, south_north, west_east) ;
p5H2SO4:units = "molecules/cm3" ;
float p5HNO3(Time, bottom_top, south_north, west_east) ;
p5HNO3:units = "molecules/cm3" ;
float p5NH3(Time, bottom_top, south_north, west_east) ;
p5NH3:units = "molecules/cm3" ;
float p5WATER(Time, bottom_top, south_north, west_east) ;
p5WATER:units = "molecules/cm3" ;
float p6PPM(Time, bottom_top, south_north, west_east) ;
p6PPM:units = "molecules/cm3" ;
float p6DUST(Time, bottom_top, south_north, west_east) ;
p6DUST:units = "molecules/cm3" ;
float p6SOA(Time, bottom_top, south_north, west_east) ;
p6SOA:units = "molecules/cm3" ;
float p6H2SO4(Time, bottom_top, south_north, west_east) ;
p6H2SO4:units = "molecules/cm3" ;
float p6HNO3(Time, bottom_top, south_north, west_east) ;
p6HNO3:units = "molecules/cm3" ;
float p6NH3(Time, bottom_top, south_north, west_east) ;
p6NH3:units = "molecules/cm3" ;
float p6WATER(Time, bottom_top, south_north, west_east) ;
p6WATER:units = "molecules/cm3" ;
float M(Time, bottom_top, south_north, west_east) ;
M:units = "molecules/cm3" ;
float O2(Time, bottom_top, south_north, west_east) ;
O2:units = "molecules/cm3" ;
float N2(Time, bottom_top, south_north, west_east) ;
N2:units = "molecules/cm3" ;
float H2O(Time, bottom_top, south_north, west_east) ;
H2O:units = "molecules/cm3" ;
float TOTPAN(Time, bottom_top, south_north, west_east) ;
TOTPAN:units = "molecules/cm3" ;
float NOX(Time, bottom_top, south_north, west_east) ;
NOX:units = "molecules/cm3" ;
float OX(Time, bottom_top, south_north, west_east) ;
OX:units = "molecules/cm3" ;
float HOX(Time, bottom_top, south_north, west_east) ;
HOX:units = "molecules/cm3" ;
float NOY(Time, bottom_top, south_north, west_east) ;
NOY:units = "molecules/cm3" ;
float ROOH(Time, bottom_top, south_north, west_east) ;
ROOH:units = "molecules/cm3" ;
float HCNM(Time, bottom_top, south_north, west_east) ;
HCNM:units = "molecules/cm3" ;
float PM10(Time, bottom_top, south_north, west_east) ;
PM10:units = "molecules/cm3" ;
float PM25(Time, bottom_top, south_north, west_east) ;
PM25:units = "molecules/cm3" ;
float PM10_anth(Time, bottom_top, south_north, west_east) ;
PM10_anth:units = "molecules/cm3" ;
float PM25_anth(Time, bottom_top, south_north, west_east) ;
PM25_anth:units = "molecules/cm3" ;
float PPM10_anth(Time, bottom_top, south_north, west_east) ;
PPM10_anth:units = "molecules/cm3" ;
```

```
float PPM25_anth(Time, bottom_top, south_north, west_east) ;
PPM25_anth:units = "molecules/cm3" ;
float pPPM(Time, bottom_top, south_north, west_east) ;
pPPM:units = "molecules/cm3" ;
float pDUST(Time, bottom_top, south_north, west_east) ;
pDUST:units = "molecules/cm3" ;
float pSOA(Time, bottom_top, south_north, west_east) ;
pSOA:units = "molecules/cm3" ;
float pH2SO4(Time, bottom_top, south_north, west_east) ;
pH2SO4:units = "molecules/cm3" ;
float pHNO3(Time, bottom_top, south_north, west_east) ;
pHNO3:units = "molecules/cm3" ;
float pNH3(Time, bottom_top, south_north, west_east) ;
pNH3:units = "molecules/cm3" ;
float pWATER(Time, bottom_top, south_north, west_east) ;
pWATER:units = "molecules/cm3" ;

// global attributes:
:Title = "CHIMERE SUITE" ;
:Sub-title = "Final concentrations file" ;
:Chimere_type = "end" ;
:Generating_process = "Generated by chimere" ;
:Conventions = "None" ;
:Domain = "CONT3" ;
:history = "File end.20030730_20030803_heat-wave.nc",
                  "was generated on Tue Nov  8 12:27:58 2005",
                  "by <user> on <host>\n" ;
}
```

Appendix G

The chemical preprocessor: CHEMPREP

chemprep allows to modify reactions, add reactions or even completely change the chemical mechanism. It allows an easy writing of reactions and from two initial input files **REACTIONS** and **PHOTORATES** produces other files which are read by **chimere**.
There are however certain conditions for the chemical set to be fulfilled for **chemprep** to work as it is proposed, otherwise some rewriting has to be done in the pre-processor. Also the two input files must be follow a certain syntax. Once the two input files are prepared by the user, the preprocessor is invoked by running the **chemprep** script.

G.1 Conditions on the mechanism

The conditions for the set of reactions to be used are listed below:

- Only reactions of the type A1(+B1(+C1....)->(a*)A2+(b*)B2+... are accepted. Letters with index 1 denote the reactants and those with index 2 the products. a, b, (if any) are stoichiometric coefficients. If omitted they are considered to equal 1. Negative stoichiometric coefficients are accepted, but not recommended since it can lead to negative concentrations. Stoichiometry in the reactants is not allowed.
- Only species which are *consumed* at least once in the mechanism will be considered as **active species** in the chimere model. If the user intends to use in chimere a species which is only results of chemical reactions but which is not reactant in any reaction, she (he) should add an artificial reaction to the mechanism, for instance A->X (A is the concerned product), with a negligible reaction rate, i. e. k=1e-10 (s-1).
- There are special species whose concentration is only prescribed in the model. These are H2O (water vapor), N2 (molecular nitrogen), O2 (molecular oxygen) and M (80% N2 + 20% O2). Their values is given by the meteo data. Reaction with these species is possible with no constraint. Changes of these "prescribed species" imply several changes in the chimere code and is not an easy change.
- Reactions have a maximum of 4 reactant species and as many products as desired (less than 1000!). To increase the number of reactants is easy: change the parameter **nreactamax** in header file **chimere.h**.
- It may be desirable that stoichiometric coefficients vary with some parameters like temperature. This is however not possible here in this version.
- All reactions will be processed by the solver in an equivalent manner. There is no family grouping (like often used for the QSSA solver) possible.
- Clear sky photolysis rates must be precalculated by the user, and depend only on zenith angle Z. No dependence on height is allowed, even in the presence of clouds. The proposed tabulated values corre-

spond to an average value between 0 and 1000m. This should be a good compromise for the boundary layer. Introducing a dependence with height will be considered in a future model version. The 12 Tabulated J values must correspond to zenith angles of (in degree) 90,88,86,83,80,75,70,60,45,30,15,0

- There is a possibility of defining *families*. These are groups of species whose concentrations sum can be output in the model. However these families are only used in outputs, not in the chemistry itself. See syntax in the next section.
- The mathematical expressions for reaction rates (the K's and J's) can take 20 forms, but here only a limited number is available. The table below show the available possibilities, together with associated syntax.

Type number	Name	Math	syntax in REACTIONS file
1	Constant K	K	k=*value*
2	Simplified Arrhenius	K=Aexp(-B/T)	k(T)=Aexp(-B/T), A=*value*, B=*value*
3	Arrhenius	K=Aexp(-B/T)(300/T)**N	k(T)=Aexp(-B/T)(300/T)**N, A=*value*, B=*value*, N=*value*
4	Troe or Falloff	K=K0*f**E/(1+M*(K0/K_∞)) K0=A0exp(-B0/T)(300/T)**N0 K_∞=A_∞exp(-B_∞/T)(300/T)**N_∞ E=1/(1+(Log10(M*K0/K_∞))**2) M=air-density	K(T,M)=troe(A0,B0,N0,A_∞,B_∞,N_∞,f)
5	Photolysis	J=J(Z)	J(Z)=photorate(*label*)[1].
6	Special type	K=K1*K2/(1+K2) K1=A1exp(-B1/T) K2=A2exp(-B2/T)	k(T)=SPECIAL_1(A1,B1,A2,B2)[2]
7	Special type	K=K1/(1+K2) K1=A1exp(-B1/T) K2=A2exp(-B2/T)	k(T)=SPECIAL_2(A1,B1,A2,B2)
8	Special type	K=2*sqrt(f1*f2*f3*f4/((1+f3)*(1+f4))) f1=A1exp(-B1/T) f2=A2exp(-B2/T) f3=A3exp(-B3/T) f4=A4exp(-B4/T)	k=SPECIAL_3(A1,B1,A2,B2,A3,B3,A4,B4)
9	Special type	K=2*sqrt(f1*f2)*(1-sqrt(f3*f4))*(1-f4)/(2-f3-f4) f1=A1exp(-B1/T) f2=A2exp(-B2/T) f3=A3exp(-B3/T) f4=A4exp(-B4/T)	k=SPECIAL_3(A1,B1,A2,B2,A3,B3,A4,B4)
10 to 12	Not used [3]		

[1] *label* is a label standing for recognition of **chemprep** in the **PHOTORATES** file. All photolysis rates must appear tabulated in this latter file, the tabulated values corresponding to specific zenith angles (see above)

[2] * Special form for radical recombination (not used in this mechanism)

[3] User can define a new type. Changes must be done in **chemprep.awk** (recognition of a new syntax), **Binichem.awk** (define the number of constant **ltabrate**, and in **Brates.f** (new math expression)

13	Aggregated ozone photolysis	J=J0*H2O/(H2O+M*(0.02909*exp(70/T) +0.06545*exp(110d0/T))) H2O=water vapor (molec/cm3) M=air density (molec/cm3)	J(T,Z,H2O)=photorate(O3) [4]
14	Modified Troe [5]	Like Type #4 (Troe) but: E=1/(1+((Log10(M*K0/K_∞)-0.12)/1.2)**2)	K(T,M)=troe(A0,B0,N0,A_∞,B_∞,N_∞,f)
15 to 20	Not used [6]		

G.2 Syntax of the REACTIONS file

Following are the rules for writing the REACTIONS file

- Each line is considered as a reaction except blank lines or lines starting with a "#" sign (left for comments).
- Species names must contain at most 15 characters.
- A reaction is written as: **A1(+A2(+A3(+A4)))->(a1*)B1+(a2*)B2+(a3*)B3...** *Rate expression* where **Ai** are the reactants (max=4), **Bi** the products and **ai** are possible stoichiometric (written if different from 1).
- *Rate expression* follows the syntax rules defined in the above table.
- Reactants can also be one (or several) of the prescribed species (H2O, N2, O2 or M).
- A1+A1(+A1)...->... are allowed (repetition of the same reactant, like HO2+HO2...
- Syntax like A1+A2-A3 is NOT allowed
- Negative stoichiometric coefficients are allowed, but not recommended (correctness of results not garanteed because never tested). In that case, however, use the syntax ±a1*B1 in the products.
- Families are defined using the syntax:
  FAM=A1+A2+A3...
  Ai must be active species, and can be repeated as many times as desired. There is no limitation to the number of family elements.

G.3 Syntax of the PHOTORATES file

The syntax is very simple: Each line contains information for one photolysis reaction. There can be more photolysis reactions than in the **REACTIONS** file. The unused ones are ignored. There must be an exact correpsondence between the **labels** of the photolysis reactions defined in the **REACTIONS** file. This label is the first column. The other columns contain the 12 tabulated photolysis rate values for the zenith angles from 90 to 0 degrees (see above table).

[4] Aggregation of the ozone photolysis with reactions with atomic oxygen. The photolysis rates must be those of ozone only, hence the budget reaction is O3->2*OH.

[5] Fall-off specific to NO2+OH+M->HNO3

[6] User can define a new type. Changes must be done in **chemprep.awk** (recognition of a new syntax), **Binichem.awk** (define the number of constant **ltabrate**, and in **Brates.f** (new math expression).

G.4 Files produced

Once the files **REACTIONS** and **PHOTORATES** are prepared, run the **chemprep** script. The produced files are (see description in the INPUT FILES to chimere Section above):

- ACTIVE_SPECIES (the list of "active species")
- CHEMISTRY (the list of reactions as read by subroutine **inichem**
- REACTION_RATES (the list of constants used for calculating rates)
- STOICHIOMETRY (the list of stoichiometric coefficients)
- FAMILIES (the list and definitions of families)

Appendix H

The top-calling script chimere-mm5

This version is the one done for the heat wave test case.

```
#!/bin/bash

#############################################################
#  General script running the CHIMERE model                 #
#                                                           #
#  SOFTWARE REQUIRED:                                       #
#     gawk, gmake, netcdf, lam-mpi,                         #
#     a fortran 90 compiler.                                #
#     CHIMERE works without modifications with compilers :  #
#                         - ifort                           #
#                         - pgf90                           #
#                         - g95                             #
#                                                           #
#       *** ONLY TESTED ON i386 architecture ***            #
#                    under GNU/Linux                        #
#                                                           #
#  http://euler.lmd.polytechnique.fr/chimere                #
#                                                           #
#  Questions: chimere@lmd.polytechnique                     #
#                                                           #
#  LAST REVISION  : NOVEMBER 2005                           #
#############################################################

# EXAMPLE OF AUTOMATIC RUN.
#
# This script assumes an already existing data base of MM5 runs.


################################
#      Timing of the run       #
################################
#
hour=00                      # Starting hour
nhours=120                   # Number of simulation hours


di=20030730                  # Initial date in YYYYMMDD format


# If you prefer to have the initial date "di" passed as an argument
# to this script, uncomment the two following lines :

#[ $# -ne 1 ] && { echo "usage : $0 YYYYMMDD" ; exit 1; }
#di=$1

# This is an example for a 5 days run
# Edit script_functions/calc_dates.sh if you want to change the length
# of the runs.
# Time boundaries should match the MM5 runs data base.
# Here we derive some useful dates from "di"

. script_functions/calc_dates.sh

################################
#          Files               #
################################

##### Simulation label
# Choose a label to uniquely identify this run
lab=heat-wave

##### CHIMERE data base files

# Vertical hybrid coordinates file (in data/vgrid)
vcoord=VCOORD.8GEOM
```

```
# Coarse-run Vertical hybrid coordinates file (if nested run)
cvcoord=VCOORD.8GEOM

# Ouptut species file (in data/output_species)
output_species=OUTPUT_SPECIES.MELCHIOR.aero.tonest

# Anthropic emissions files
# standard
emissdir_std=/COMPERES/pub/chimere/emiEMEP-200411/EMI_aero_2002/2LEVELS
# if you want to use Airparif emission files (restricted access)
emissdir_airparif=/COMPERES/priv/chimere/emiAIRPARIF-200412/EMI_aero_2004

# Directories for boundary conditions
mozart2dir=/COMPERES/pub/chimere/MOZART2
lmdzinca2dir=/pollution1/pollux/CHIMERE/LMDZINCA
gocartdir=/COMPERES/pub/chimere/GOCART



##### Meteorological data (MM5 raw binary outputs)

mm5fil=/COMPERES/pub/MM5/AVN/${year}/EUR1/MMOUT_EUR1_${di}_${de}_S



##### Outputs directories and files

# Directory where the output files of this run will be written
simuldir=‘pwd‘/OUTPUTS

# If this run is a nested one, boundary conditions will be obtained
# from this coarse run output :
coarsefile=‘pwd‘/OUTCOARSE/out.20041226_20041230_coarse.nc

# If this run is a continuation one, initial conditions will be
# obtained from the "end" file in this directory
psimuldir=‘pwd‘/OUTPUTS



########################
# Temporary directory #
########################

chimere_tmp=‘pwd‘/$$
                            # All operations are done in a temporary directory.
                            # CHIMERE_TMP is the temporary directory. Give a name where
                            # there is a lot of temporary disk space
                            # The size of the directory is typically 100 mo but
                            # can be much more in high-resolution simulations
                            # CHIMERE is exectuted in this directory.
                            # !!!! This name must be an absolute path (not ./mydir ...) !!!!
clean=yes
                            # clean=no:    The entire directory is kept intact.
                            # clean=light: Only small files are kept.
                            # clean=full:  The directory is removed at the end.
                            # By setting a CHIMERE_TMP directory name that depends
                            # on process number ($$), it is possible
                            # to perform several simulations simultaneously
                            # without undesired interactions



################################
# Parameters of the simulation #
################################

aerosols=1                  # 0 for gas version
                            # 1 for aerosol version without sea salts
                            # 2 for aerosol version with sea salts
                            # 3 for aerosol version with inert sea salts

meca=MELCHIOR2              # Chemical mechanism (MELCHIOR2 or MELCHIOR1)
                            # MELCHIOR2 is the fast, reduced version of MELCHIOR1

dom=CONT3                   # Domain name, as specified in file names
                            # of the "domains" directory

levout=1                    # Number of layers in output files (starting from bottom)

ivs=1                       # Emission sublayer (2) or not (1).
                            # *** USE IVS=2 ONLY FOR THE SMALLEST DOMAIN, DO NOT NEST ANY
                            # DOMAIN WITHIN A DOMAIN WITH IVS=2

#############
# Numerics #
#############


phys=6                      # Frequency of physics calculation (number per hour)

step=1                      # Refined time step relative to physics calculation
                            # 1/(phys*step) is the time step in hours
                            # Down to 1/4 degree resolution, phys=6 and step=1 is fine
```

```
                            # For finer grids (5-10 kms), take at least step=4
                            # For grids of 2-5 kms, try step=6 or more

ngs=1                       # Number of Gauss-Seidel iterations in the TWOSTEP solver.
                            # Take 1 for model testing, 2 for higher accuracy.

nsu=5                       # Number of Gauss-Seidel iterations in the TWOSTEP solver during
                            # Spin up

irs=1                       # Number of hours for Spin up

nequil=0                    # 1 : online coupling with ISORROPIA
                            # 0 : ISORROPIA tabulation

npeq=1                      # Compute equilibrium each "npeq" physical time step : < 5

nsconcs=${nhours}           # Save all concentrations every ... hours (file end....)

nsdepos=12                  # Save all cumulated deposition every ... houres (file dep...)

##########################################
# Meteorological information  and options #
##########################################

meteo=mm5                   # Name of meteo interface. A BASH script named
                            # interf-<meteo> must be present in the meteo
                            # directory for the preparation of meteo data
                            # This script has a certain number of arguments
                            # which are passed below in this script.
                            # From these arguments, the script interf-<meteo>
                            # Must construct input files to the diagmet.f program
                            # "metargs" is the list of these arguments

mm5dom=MM5EUR1

upblmin=0     # Urban correction : Minimal urban PBL height.    Set 0 if not used
uflxadd=0     # Urban correction : Additional urban heat flux.  Set 0 if not used
uwinfac=1     # Urban correction : Urban wind factor in the SL. Set 1 if not used


###################################
# Anthropic emissions information #
###################################

emis=std                    # Anthropic emission interface: std only available at present
                            # User can define its own interface. This assumes the
                            # definition of command lines added below below the
                            # comment "Hourly anthropic emissions"
                            # std emissions are given for each model species in an ascii
                            # files containing 3 types of day per month and 12 months.

nlevemis=2                  # Number of emission levels

flatreduc=1.00              # Flat reduction of emissions (multiplication factor), which
                            # can be used for extrapolating emissions in time for another year


#################################################
# Options for boundary conditions; available :  #
# MOZART2:      For gas-phase only (monthly av) #
# LMDZINCA2:    For gas-phase only (monthly av) #
# GOCART:       For aerosols  only (monthly av) #
# set nothing if initialized to zero            #
# set latnest/topnest to "yes" if nest run      #
#################################################

latnest=no
latboungas=LMDZINCA2
latbounaer=GOCART
topnest=no
topboungas=LMDZINCA2
topbounaer=GOCART


#############################################
# Informations about the coarse simulation  #
# only required if simulation is a nest run #
#############################################

# Coarse domain
cdom=CONT3                  # Make sure the COORD_<cdom> file is present in domains dir.


######################
# Initial conditions #
######################

init=0                      # Initialization with previous run (1) or interpolation
                            # of boundary conditions (0)



###################################
# Biogenic emissions information #
###################################
```

```
iopt_eros=1                 # Optional wind erosion dust emissions (0/1)
iopt_resu=0                 # Optional resuspension parameterization (0/1)
                            # The resuspension parameterization is
                            # to be considered as "in test"


####################################
# End of user configurable section  #
####################################

sim=${di}_${de}_${lab}       # Simulation label:
                             # output files will be named out.<sim>

psim=${dib}_${deb}_${lab}    # Simulation label of previous run (if continuation run)

case ${emis} in
  std) emissdir=$emissdir_std ;;
  airparif) emissdir=$emissdir_airparif ;;
  *)   echo "*** INTERFACE ${emis} UNDEFINED"; exit 1 ;;
esac

. script_functions/chimere-mm5.exec.sh
```

Appendix I

Definition of new CHIMERE domains

I.1 Horizontal domain: the COORD_[dom] file

You can use the following commands and scripts to create a new COORD file. To achieve this, you need to add a new line in the **DOMAINS** file describing your new domain as for example:

```
domain     NX  NY  NSHO  DX    DY   XMIN    XMAX
CONT3      67  46  1904  0.5   0.5  -10.5    35
```

Run the **makeCOORDdomains** script (in the **domains/** directory) with the required domain name
The makeCOORDdomains script:

```
#!/bin/sh
dom=CONT3
nzo=‘gawk ’$1=="’${dom}’" {print $2}’ DOMAINS‘
nme=‘gawk ’$1=="’${dom}’" {print $3}’ DOMAINS‘
show=‘gawk ’$1=="’${dom}’" {print $4}’ DOMAINS‘
dx=‘gawk ’$1=="’${dom}’" {print $5}’ DOMAINS‘
dy=‘gawk ’$1=="’${dom}’" {print $6}’ DOMAINS‘
xbotleft=‘gawk ’$1=="’${dom}’" {print $7}’ DOMAINS‘
ybotleft=‘gawk ’$1=="’${dom}’" {print $8}’ DOMAINS‘
touch COORD
\rm COORD
gawk -v xbotleft=${xbotleft} -v ybotleft=${ybotleft}
-v dx=${dx} -v dy=${dy} -v nzo=${nzo}
-v nme=${nme} -v syst=${syst} -- ’BEGIN{
    for(j=1;j<=nme;j++){
    for(i=1;i<=nzo;i++){
       x=(i-1)*dx
       y=(j-1)*dy
       xlon=xbotleft+x
       xlat=ybotleft+y
       print  xlon "  " xlat >> "COORD"
    }
    }
}’
mv COORD COORD_${dom}
```

I.2 Vertical domain: the VCOORD file

In the directory **util/**, edit the **define_geom** script and adapt the top pressure level to the one you need. **NB** It is recommended to have a maximum ptop of 200 hPa since the global model used for boundary conditions delivered data only up to this value.
The define_geom script:

```
#!/bin/sh

#  Defines a geometrical-progression vertical layering
#  For the CHIMERE model. It actually produces sets of
#  Hybrid sigma-p coefficients Ak and Bk such that the
#  pressure at top of layer k is Pk = Ak*1e5 + Bk*Ps in Pa
#  where Ps is Surface pressure

#  Define the parameters below and run the script by just typing "define_geom"

ptop=700    # Pressure at model top
nlay=8      # Number of layers
pre1=995    # Pressure (hPa) of the first layer top when Ps = 1000 hPa
fout=../data/vgrid/VCOORD.8GEOM700

echo ${nlay} ${pre1} ${ptop} | gawk -f geom.awk > ${fout}
```