MECCA User Manual

Rolf Sander & Astrid Kerkweg

sander@mpch-mainz.mpg.de, akerkweg@mpch-mainz.mpg.de
Air Chemistry Department
Max-Planck Institute of Chemistry
PO Box 3060, 55020 Mainz, Germany
www.mpch-mainz.mpg.de/~sander/messy/mecca/

Version from 2005/01/31

Abstract

MECCA (Module Efficiently Calculating the Chemistry of the Atmosphere) is a module that calculates tropospheric as well as stratospheric chemistry. MECCA is compatible with the MESSy structure (Jöckel et al., 2005, see also http://www.messy-interface.org) and can therefore easily be coupled to a variety of base models, e.g. a simple box model or a global general circulation model. The KPP software (Damian et al., 2002; Sandu and Sander, 2004) is used for the integration of the set of stiff differential equations. The code is written in Fortran95. Here we describe how to install, execute and modify MECCA.

Contents

1 Installation and configuration of MECCA

This section can be skipped if MECCA is already installed on your computer.

MECCA has been tested successfully on several UNIX-like operating systems. The easiest installation is probably on a Linux PC since several auxiliary programs are already included in a typical Linux distribution. Installation under "Windows" is neither recommended nor supported. MECCA consists of the files listed in Table 1. The prerequisites are:

A Fortran95 compiler (mandatory): Several compilers have been tested successfully: Lahey (for Linux), GNU g95 (for Linux), Intel (for Linux), Compaq (alpha unix). Other compilers can be used as well if they accept standard Fortran95 code. It should be noted that the GNU compiler for Linux is free and can be downloaded from `http://www.g95.org/`.

The kinetic preprocessor KPP (mandatory): This flexible numerical integration package (`http://www.cs.vt.edu/~asandu/Software/KPP`) transforms the set of chemical equations into Fortran95 code. The current version of MECCA needs the KPP version v1.1-f90-alpha12, which is included (source code and Linux binary) in this electronic supplement. To install KPP, first unzip the file `v1.1-f90-alpha12.zip`. Set the environment variable `KPP_HOME` to the KPP root directory which has just been created. The path of the binary (`$KPP_HOME/bin`) must be added to the `$PATH` variable. Using the tcsh, the necessary commands are:

```
setenv KPP_HOME /MYPATH/v1.1-f90-alpha12
setenv PATH $KPP_HOME/bin:$PATH
```

To install KPP on a system other than Linux, you need to compile the binary yourself, which requires a C compiler, and the tools flex and yacc. More information can be found in the `README` file inside the KPP directory.

Unix tools tcsh, gawk, sed, make (mandatory): These tools are standard on Linux systems. Please check that recent versions of them are installed. Especially gawk versions prior to 3.1.3 may lead to strange error messages.

LaTeX (optional): If you have LaTeX installed on your computer, you can print a table (including rate coefficients and references) of the currently selected mechanism (see Section 4.2.3 for details).

netcdf library (optional): The netcdf library is needed to create model output in netcdf format. It can be obtained from `http://my.unidata.ucar.edu/content/software/netcdf/`. If you don't have it, you can still run the model but produce only ascii output.

Once all prerequisites are fulfilled, you can install mecca by simply unpacking `mecca_src_0.9.1.zip`. Enter the name of your Fortran95 compiler (`F90`) and compiler options (`F90FLAGS`) into the `Makefile` in the `boxmodel` directory. Values for `LINCLUDES` and `LLIBS` are not needed for the default model run with ascii output. If you have the netcdf library installed on your system, you can also choose netcdf output. To activate netcdf output, you have to edit the `Makefile`:

- Enter the netcdf library information in `LINCLUDES` and `LLIBS`.
- Change the variable `OUTPUT` from `ASCII` to `NETCDF`.

Should there be any problems with the MECCA installation, please check the following:

- Confirm that all prerequisites (see above) are fulfilled!
- Confirm that the perl path in the first line of `boxmodel/sfmakedepend` is correct.
- Confirm that the tcsh paths in the first lines of `boxmodel/xbox` and `xmecca` are correct.

2 Compiling and running the MECCA box model with the shell script `xbox`

The tcsh script `xbox` guides you through the process of running the box model:

```
cd mecca/boxmodel
./xbox
```

First, you are asked if you want to create a chemical mechanism with `xmecca`. If you answer "y", you can select a new chemical mechanism with `xmecca` as described in detail in Section 3. However, for the first tests with MECCA we recommend that you answer "n" and use the default mechanism, i.e. marine gas-phase and aerosol chemistry including halogens. Next, when asked for an option, choose "c" to compile the Fortran95 code. After a successful compilation you can run the model. The model time is printed on the screen during the model run. The default is to integrate 8 days. Mixing ratios of all chemical species are output to the ascii file `mecca_tracer.dat`. Alternatively, if you have chosen netcdf output, the file `mecca_tracer.nc` is created. It can be viewed with any graphics program that can read netcdf files (e.g. ferret (`http://ferret.wrc.noaa.gov/Ferret/`) or idl).

Table 1: List of MECCA fortran files

static core files	
`messy_mecca.f90`	core file
`messy_mecca_mbl.f90`	mbl core file
`messy_mecca_func.f90`	functions to calculate rate coefficients
`mecca.nml`	namelist for run control
KPP-produced files	
`messy_mecca_kpp.f90`	main numerical integrator
`messy_mecca_kpp_g_mem.f90`	declarations of kpp variables
`messy_mecca_kpp_s_mem.f90`	info about sparse matrix
box model related files	
`mecca.f90`	master file of the box model
`messy_mecca_box.f90`	interface between box model and MECCA subroutines
`mo_netcdf.f90`	netcdf subroutines
`species.inc`	include file
ECHAM5 related files	
`messy_mecca_e5.f90`	interface between ECHAM5 and MECCA subroutines
`messy_mecca_mbl_e5.f90`	interface between ECHAM5 and MECCA-MBL subroutines
`messy_mecca_mem_e5.f90`	declaration and memory for variables
`messy_mecca_idt_e5.inc`	include file
`messy_mecca_c2mr_e5.inc`	include file
`messy_mecca_mr2c_e5.inc`	include file
`messy_mecca_trac_e5.inc`	include file
`mecca_t.nml`	namelist for tracer initialization

3 Selecting a chemical mechanism with the shell script `xmecca`

MECCA contains a very comprehensive set of chemical reactions in both the gas phase and in liquid particles. For most applications, using the complete mechanism will consume too much CPU time. Therefore, the shell script `xmecca` has been written which allows the user to create a custom-made subset of the chemical mechanism interactively. Normally, `xmecca` is called via `xbox`. However, you can also start it manually:

```
cd mecca
./xmecca
```

First, you have to select MECCA as the submodel (the alternative SCAV is used for a cloud scavenging model and not described here). Then, you must choose a numerical integrator. Next, choose a subset of chemical reactions. Finally, by running KPP, `xmecca` creates several Fortran95 files. These steps are explained individually in the following sections.

3.1 Selecting the submodel

The first user input is requested when `xmecca` prompts to "Choose a submodel". Currently it is possible to choose between `mecca` and `scav`.

- `mecca` is the default option. It uses the input files `gas.spc`, `aqueous.spc`, `gas.eqn`, `aqueous.eqn`. The contents of these files is explained in Section 4.
- `scav` is a cloud chemistry mechanism for the scavenging submodel developed by H. Tost. `scav` is not described in this manual. More information about the scavenging submodel is available at `http://www.mpch-mainz.mpg.de/~tost/messy/scav/`.

3.2 Selecting a numerical integrator

Several numerical integrators are defined in the subdirectory `mecca/integr/` and can be used with KPP. The default is the third-order Rosenbrock solver with automatic time-step control (ros3). It is very robust and capable of integrating very stiff sets of equations (e.g. chemical mechanisms including both gas- and aqueous-phase chemistry). A Rosenbrock solver of second order (ros2) with manual time-step control is also available. It is very fast but only suitable for less stiff sets of equations (e.g. gas-phase only chemistry). If you choose manual time-step control (e.g. ros2-log5), do so at your own risk!

3.3 Selecting a set of chemical reactions

All chemical reactions are marked. Each marker consists of several labels which contain information about the altitude

(troposphere/stratosphere), the phase where the reaction occurs (gas/aqueous), its relevant chemical elements, and more. See Section 4.2 for a complete list of labels. To define a set of chemical reactions, you can either choose a pre-defined selection by number or enter a boolean expression based on the labels. Boolean expressions are typed in gawk syntax. The most important operators and expressions are:

`&&` = AND
`||` = OR
`!` = NOT
`()` = parentheses
`1` = TRUE
`0` = FALSE

For example, to select all gas-phase reactions (G) except for those including halogens (Cl, Br, I), type:

```
G && !Cl && !Br && !I.
```

After the chemical mechanism has been selected, it is possible to create a table of it. Only the selected reactions are listed. The table also contains the rate coefficients and their references, as described in Section 4.2.3.

3.4 Producing the chemistry Fortran95 code

Once all necessary selections have been made, the question "run kpp?" can be answered with "y" and KPP starts. KPP produces three temporary Fortran95 files. These files are post-processed and then saved under new names:

- `messy_mecca_kpp.f90`: The main KPP-generated file containing the numerical integration code.
- `messy_mecca_kpp_g_mem.f90`: Definitions of the indices `KPP_*` and some arrays
- `messy_mecca_kpp_s_mem.f90`: Definitions of arrays for the sparse matrix technique.

The script `xmecca` then produces include files (`*.inc`) for coupling the MECCA chemistry to other models, as explained in Section 5.

3.5 The `CTRL` namelist

Next, `xmecca` lists the contents of the `CTRL` namelist in the file `mecca.nml`. The purpose of this namelist is to control the behaviour of MECCA during run-time. Currently it contains only the switches `lmbl` and `lskipkpp`. `lmbl` must be set to `.true.` to switch on marine aerosol chemistry calculations. Setting `lskipkpp` to `.true.` skips the KPP integration completely and is only useful for debugging purposes.

4 Modifying the chemical mechanism

The chemical mechanism can be updated or extended by modifying the species files (`*.spc`), the equation files (`*.eqn`) and the Fortran95 files (`*.f90`).

4.1 The species files `gas.spc` and `aqueous.spc`

The files `*.spc` declare chemical species for KPP. All species that may occur in an equation must be declared here. Additional dummy species may also be declared here.

Gas-phase species are declared in `gas.spc` and aqueous species in `aqueous.spc`. Examples for gas-phase species are `O2`, `O1D`, and `NO2`. Examples for aerosol species are `NO3m`, `H2O2`, `Hp`, and `SO4mm`. Anions end with "`m`" for minus, and cations with "`p`" for plus. All aqueous-phase species have the suffix "`_##`", which is a placeholder for the aerosol type. `xmecca` replaces it by either "`_as`" (accumulation soluble) or "`_cs`" (coarse soluble). This allows separate chemistry calculations for aerosol particles of different sizes.

All species are defined here with `#DEFVAR`, i.e. KPP considers them as prognostic variables. To treat a species as a constant (e.g. CO_2), it can be added to the `#SETFIX` command in the file `mecca.k`.

4.2 The equation files `gas.eqn` and `aqueous.eqn`

The equation files `*.eqn` define the chemical reaction mechanism for KPP. Each reaction occupies one line in a `*.eqn` file. An example is:

```
O2 + O1D = O3P : 3.2E-11*EXP(70./temp);
```

The first part (up to the colon) defines the reaction, and the second part (between the colon and the semicolon) defines the rate coefficient. The lines may also contain comments. Comments in MECCA `*.eqn` files are either enclosed in curly braces, or the comment line starts with `//`. When using `xmecca`, some comments have a special meaning. Comments starting with the percent symbol "`{%...}`" are markers (see Section 4.2.1). Comments starting with the hash mark "`{#...}`" contain the reaction numbers (see Section 4.2.2). Comments starting with the ampersand "`{&...}`", the at symbol "`{@...}`", or the dollar "`{$...}`" are used to store information for the listing of reactions, as explained in Section 4.2.3.

If the definition of a rate coefficient is very complex, it can be stored in a Fortran95 variable and the variable is put into the `gas.eqn` file. For example, the rate of the self reaction of HO_2 is quite complex since it depends on humidity. It is predefined and the reaction line can be simplified to:

```
HO2 + HO2 = H2O2 : k_HO2_HO2;
```

The declaration and definition of `k_HO2_HO2` are also in the `gas.eqn` file. They can be found in the so-called KPP "program fragments" `F95_DECL` and `F95_RCONST`, e.g.:

```
#INLINE F95_DECL
  REAL :: k_HO2_HO2
#ENDINLINE

#INLINE F95_RCONST
  k_HO2_HO2 = (1.5E-12*EXP(19./temp)+ &
               1.7E-33*EXP(1000./temp)*cair)* &
               (1.+1.4E-21* &
               EXP(2200./temp)*C(KPP_H2O))
#ENDINLINE
```

A third method to add reaction rates with complex dependencies are Fortran95 functions. This is done for example for the third order reaction rates (`k_3rd`). A function call is given as the rate in the `*.eqn` file. These functions are defined outside of the KPP code in the file `messy_mecca_func.f90`. They are made available to the KPP code via Fortran95 USE statements.

4.2.1 Markers and labels

Each reaction must contain a marker. A marker contains one or more labels. The syntax is "`{%...}`" where the dots represent the labels. Each label must start with an upper case letter and can be followed be one or more lower case letters or numbers. The labels are placed in the marker without separators. The following labels are available and should appear in this order:

1. altitudes at which the reaction occurs (mandatory, include at least one)
   - St = Reactions relevant in the stratosphere
   - Tr = Reactions relevant in the troposphere
2. phase (mandatory, include exactly one)
   - Aas = Aqueous, accumulation soluble mode aerosol
   - Acs = Aqueous, coarse soluble mode aerosol
   - G = Gas phase reactions
   - Psc = Reactions on polar stratospheric clouds
3. elements (include all elements that occur in the reaction)
   - N = Nitrogen
   - C = Carbon with > 1 C atom (only used for C,N,O species but not for halogenated or sulfur-containing organics)
   - F = Fluorine
   - Cl = Chlorine
   - Br = Bromine
   - I = Iodine
   - S = Sulfur
4. other
   - J = Photolysis reactions
   - Mbl = Minimum reaction mechanism for MBL chemistry
   - Sc = Scavenging chemistry mechanism
   - Scm = Scavenging chemistry mechanism, minimum selection

It is possible to add new labels. Element symbols must not be used because they are reserved for reactions of that element. For example, since S is sulfur, the symbol S cannot be used for "stratosphere".

4.2.2 Reaction numbers

Each reaction in the equation files `*.eqn` has a unique reaction number ("number" is not quite correct, since letters are included as well). The syntax is "`{#...}`". The reaction number starts with one or more upper case letters denoting the type of reaction. The following types exist:

A	aqueous-phase reactions
H	heterogeneous reactions and Henry's law
EQ	equilibria in the aqueous phase (acid/base and others)
G	gas-phase reactions
J	J-values of photolysis reactions
PSC	polar stratospheric clouds

The type is followed by a sequence of 3 or 4 digits. The first digit is the number of the main element of the reaction. The following numbers are used:

1)	O	oxygen
2)	H	hydrogen
3)	N	nitrogen
4)	C	carbon
5)	F	fluorine
6)	Cl	chlorine
7)	Br	bromine
8)	I	iodine
9)	S	sulfur

Out of those elements that occur in a reaction, the one with the highest number is called the main element. Accordingly, the second digit is determined by the element with the second highest number (or set to zero if there is no second element in the reaction). There is one exception in this numbering scheme: For the carbon group, the second digit is the number of C atoms in the largest organic molecule.

The following digits have no special meaning. If a reaction branches into several pathways, a suffix "a", "b", "c", ... is added.

4.2.3 Creating a table of the chemical mechanism

To ensure that the documentation of the chemical mechanism is always up to date, the necessary information is contained together with the KPP code in the `*.eqn` files. The awk script `eqn2tex.awk` converts information from the selected reactions into a LaTeX table. BibTeX citations are included in comments starting with an ampersand "`{&...}`". If there is a second ampersand "`{&&...}`", additional information about reactions can be found in `meccanism.tex` as a footnote to the tables. Comments starting with the at symbol "`{@...}`" or the dollar "`{$...}`" can be used to put LaTeX commands directly into the `*.eqn` files. `eqn2tex.awk` produces several `*.tex` files which are included into `meccanism.tex`.

4.3 Fortran95 files

Functions to calculate rate coefficients are contained in `messy_mecca_func.f90`. The accommodation coefficients (`alpha`) and the mean velocity (`vmean`) are used for the calculation of the mass transfer coefficients (`ykmt`). Together with the inverse dimensionless Henry's law coefficients (`yhenry`), they are needed to calculate equilibria between the gas and the aqueous phase. Heterogeneous reactions are described with the forward (`k_exf`) and backward (`k_exb`) rate coefficients.

5 Coupling MECCA to other models via MESSy

Since MECCA is a pure chemistry model, it needs to be coupled to a meteorological base model. In the simplest case, this can be a box model. More complex 1-, 2- or 3-dimensional models are also possible. The currently implemented base models are explained in the following sections. As a coupling interface, we use the Mainz Earth Submodel System MESSy (Jöckel et al., 2005, see also `http://www.messy-interface.org`). To facilitate coupling of the selected chemistry mechanism to the base models described below, `xmecca` starts the awk script `tracdef.awk` which produces the five files `messy_mecca_trac_e5.inc`, `messy_mecca_c2mr_e5.inc`, `messy_mecca_mr2c_e5.inc`, `messy_mecca_idt_e5.inc`, and `species.inc`. The coupling can be changed at run time by the namelist `CPL` in `mecca.nml` (see Section 5.2.3).

5.1 Coupling MECCA to a box model

MECCA has been coupled to a box model of the marine boundary layer, similar to the MOCCA model by Sander and Crutzen (1996). The file `species.inc` provides the names of the chemical species as Fortran95 strings. The box model files are the main program `mecca.f90` and the submodel interface layer (SMIL) file `messy_mecca_box.f90`. Compilation is controlled by a `Makefile`, and the shell script `xbox` guides the user through the selection of the chemical mechanism, compilation and execution of the code as described above.

5.2 Coupling MECCA to ECHAM5

We have coupled the MECCA chemistry to the general circulation model (GCM) ECHAM5 (`http://www.mpimet.mpg.de/en/extra/models/echam/echam5.php`) using the MESSy interface. A detailed description how to run ECHAM5/MESSy together with MECCA and other submodels can be found at: `http://www.messy-interface.org`. The file structure of the coupling is shown in Figure 1.

After running `kpp`, `xmecca` calls the `awk` script `tracdef.awk` which extracts all `KPP_*` from `messy_mecca_kpp_g_mem.f90` and declares the corresponding tracer indices `idt_*` in `messy_mecca_idt_e5.inc`. For all `KPP_*` that are non-zero, it also inserts the species into `messy_mecca_c2mr_e5.inc`, `messy_mecca_mr2c_e5.inc`, and `messy_mecca_trac_e5.inc`, using the tracer info in `tracdef.tex`. The files `messy_mecca_c2mr_e5.inc` and `messy_mecca_mr2c_e5.inc` contain the conversions between mixing ratios (used for ECHAM5/MESSy tracers) and concentrations (used in the MECCA chemistry). The `tracdef.tex` file contains information for the tracer definition that is done in ECHAM5/MESSy via the `new_tracer` command. Here the user can select which processes other than chemistry (e.g. advection, convection, dry deposition, sedimentation, ...) affect the tracers. LaTeX syntax is used in `tracdef.tex`. The first column contains the name of the species enclosed in single quotes. The following columns contain values for the optional parameters of `new_tracer`. The order of the columns must be exactly the same as for the `addproperty` commands in `tracdef.awk`.

5.2.1 Photolysis rates

Photolysis rates ("J-values") are supplied by the JVAL submodel via the MESSy interface in the form of global 3D fields. MECCA, however, needs scalar values. When adding the photolysis of a new species (called "xyz" here), the following steps are necessary:

- Add the new species to the JVAL submodel.
- Enter the new photolysis reaction in `gas.eqn`.
- Define in the `F95_DECL` block of `gas.eqn`:
  ```
  REAL :: J_xyz
  ```
- Extract data for the current grid box into the scalar in `messy_mecca_e5.f90`:
  ```
  REAL, SAVE, DIMENSION(:,:,:), &
    POINTER :: j_xyz_3d
  CALL get_stream_element( &
  ```

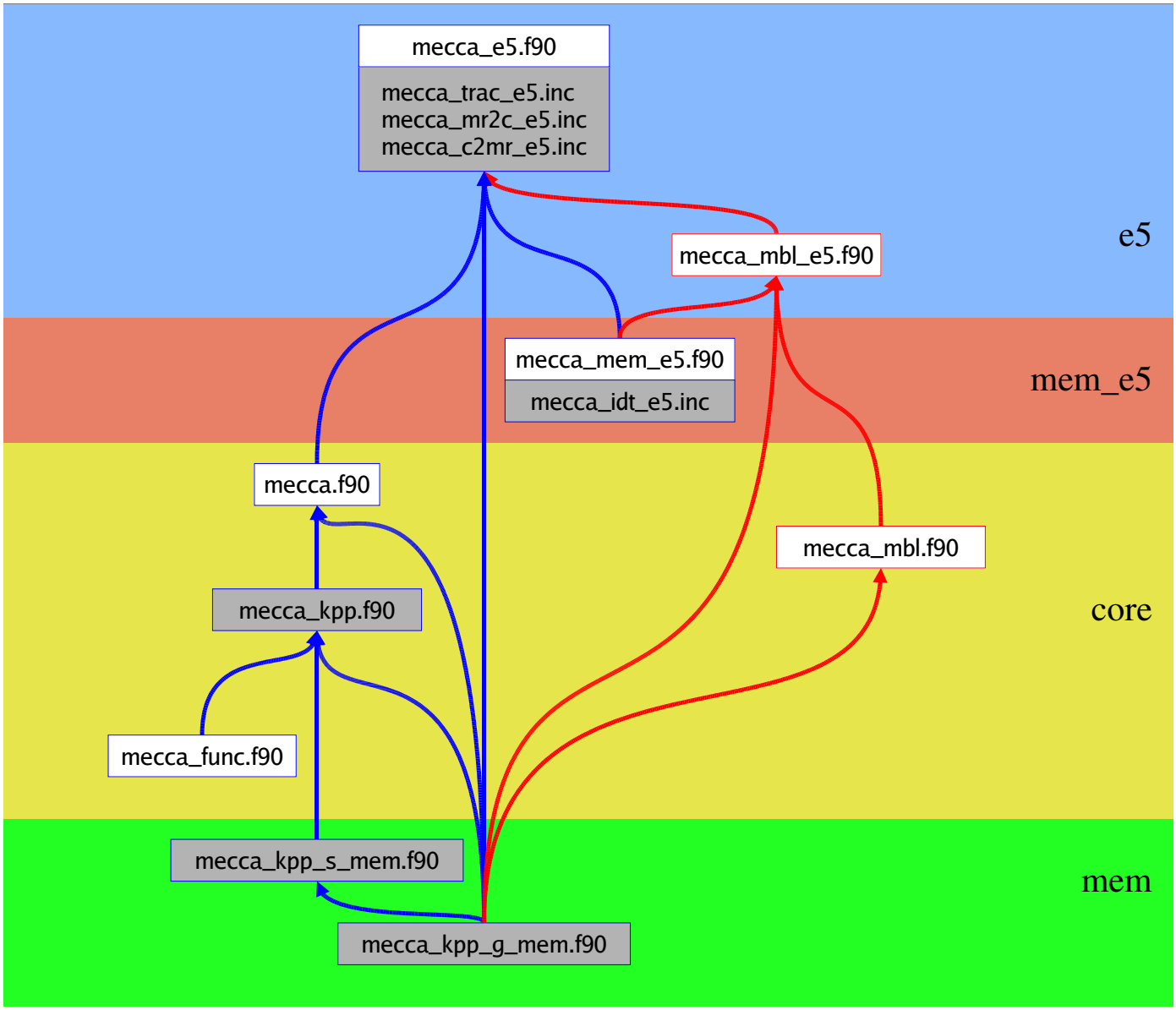

Figure 1: USE diagram showing interdependencies when MECCA is coupled to ECHAM5/MESSy. The MESSy submodel interface layer (SMIL) files are in the `e5` and `mem_e5` layers, and the MECCA submodel core layer (SMCL) files are in the `core` and `mem` layers. See Jöckel et al. (2005) for details about the MESSy layers. Each arrow starts at the module that exports a variable or subroutine and points to the module that imports it via the Fortran95 USE instruction. Blue arrows refer to the submodel MECCA, red arrows to the sub-submodel MECCA_MBL. The prefix `messy_` has been omitted from the file names for better readability. Files with the suffix `.inc` are include files that are incorporated into fortran files. Boxes with a grey background show files that are automatically produced by `KPP` and `xmecca`.

```
    streamptr, 'J_xyz', j_xyz_3d)
  IF (ASSOCIATED(j_xyz_3d)) &
    J_xyz = J_xyz_3d(lon,lev,lat)
```

To activate the import of J-values from JVAL into MECCA, the `CPL` namelist in `mecca.nml` must contain the entry:

```
jval_stream  = 'jval_gp'
```

5.2.2 Reactions on polar stratospheric clouds

Heterogeneous reactions on polar stratospheric clouds (PSCs) can be considered by MECCA if the corresponding rate coefficients are supplied. When MECCA is coupled to ECHAM5 via MESSy, these values are obtained from the submodel PSC. To activate the import of heterogeneous rate coefficients from PSC into MECCA, the `CPL` namelist in `mecca.nml` must contain the entry:

```
het_stream = 'psc'
```

5.2.3 Marine aerosol chemistry

Since most of the aqueous-phase rate coefficients are given in units of mol/l (i.e. refer to the volume of the liquid), and MECCA-MBL requires a unit of molecules/cm^3 (refering to the gas-phase volume), using MECCA-MBL depends on the knowledge of aerosol properties for the conversion. Coupling the aerosol chemistry to MECCA-MBL is controlled through the `CPL_MBL` namelist in `mecca.nml`. To activate the import of aerosol properties from the modal aerosol submodel M7 into MECCA, the namelist must contain the entries:

```
aerosol_module = 'modal'
aerosol_stream = 'm7'
```

The aerosol submodel provides information about soluble accumulation mode (`as`) and coarse mode (`cs`) particles. MECCA-MBL imports the number densities (`number_as`, `number_cs`), the dy and wet radius (`aerosol_dryradius`, `aerosol_wetradius`), and the standard deviation (`sigma`) for each mode as well as the numbers of the mode within the aerosol submodel (`modenumber_as`, `modenumber_cs`). If the aerosol submodel includes the process of H_2SO_4 condensation, this process must be excluded in MECCA-MBL to avoid calculating the same process twice. This is controled by the switches `lcpl_aerophase_as` and `lcpl_aerophase_cs`. If set to `.TRUE.`, the transition of H_2SO_4 from the gas into the aerosol phase is ignored by MECCA. The switch also controls the coupling of S(VI) between the submodels. M7 treats S(VI) as one lumped species, whereas MECCA-MBL distinguishes between H_2SO_4, HSO_4^-, and SO_4^{2-}. For this reason the name of the S(VI) tracers `sulfVI_as` and `sulfVI_cs` of M7 must be given in the CPL namelist, too.

A second coupling possibility in MECCA-MBL is the calculation of cations and anions from seasalt emissions. The switch `l_calc_emis` must be set to `.TRUE.`. The necessary aerosol information is imported from other submodels through the MESSy data transfer interface. The aerosol emissions must be specified in `emis_stream`, `emis_ss_as`, and `emis_ss_cs`.

For numerical reasons, the user can choose if the emissions are written directly into the tracer tendency (`l_tendency=.TRUE.`) or coupled to the vertical diffusion.

References

Damian, V., Sandu, A., Damian, M., Potra, F., and Carmichael, G. R.: The kinetic preprocessor KPP – a software environment for solving chemical kinetics, Comput. Chem. Eng., 26, 1567–1579, 2002.

Jöckel, P., Sander, R., Kerkweg, A., Tost, H., and Lelieveld, J.: Technical Note: The Modular Earth Submodel System (MESSy) – a new approach towards Earth System Modeling, Atmos. Chem. Phys., this issue, 2005.

Sander, R. and Crutzen, P. J.: Model study indicating halogen activation and ozone destruction in polluted air masses transported to the sea, J. Geophys. Res., 101D, 9121–9138, 1996.

Sandu, A. and Sander, R.: Technical Note: Simulating chemical systems in Fortran90 and Matlab with the kinetic preprocessor KPP-2.0, Atmos. Chem. Phys., (in preparation), 2004.