Tutorial #2

LMDZ team

December 6, 2015

Contents

This tutorial makes use of a zoomed configuration of LMDZ. Section 1 is an initialization phase. In section 2, you can either take a closer look at tracers, or the Orchidee model or switch some parameterizations of the physics part of the program. The three experiments in section 2 are independent. In section 3, you will nudge the model toward analyzed wind fields.

Start by installing the model with the script `install.sh`, if you have not already done so in tutorial #1.

1   Setting up a case with a zoomed grid

- Go to the directory `LMDZ20151130.trunk/modipsl/modeles/LMDZ5`, which contains files `makelmdz_fcm`, `libf`... In this directory, download the following tar file :

  ```
  wget http://www.lmd.jussieu.fr/~lmdz/Distrib/tutorial.tar
  tar -xf tutorial.tar
  cd TUTORIAL
  ```

- In the directory `TUTORIAL`, take a look at the extracted files. You should edit file `gcm.def` if you want to place the center of the zoom at your preferred location. For that you just have to change the longitude and latitude of the zoom center, `clon` and `clat`.

- If you are interested in coupling LMDZ with the surface scheme Orchidee, edit the `init.sh` file and set option `veget=1`. If not (`veget=0`), the model will be run with a simplified scheme for surface hydrology: the "bucket" scheme.

- Increase the stack memory you can use by typing the following command:

  ```
  ulimit -Ss unlimited
  ```

  or, even better, add this command in you shell start-up file (if you use Bash, your start-up file can be `.profile` or `.bash_profile`).

- Run the script :

  ```
  ./init.sh
  ```

  The script first compiles the model (`gcm.e`) again because it uses a different resolution, $48 \times 36 - L39$, than the one used in `install.sh`. `init.sh` also compiles the program `ce0l.e`, which creates initial state and boundary conditions. `init.sh` then downloads new NetCDF files which contain the surface orography, sea-surface temperature, and so on, as well as 3D meteorological files taken from ECMWF analyses at a particular date. `init.sh` then runs `ce0l.e` which creates files `start.nc`, `startphy.nc` and `limit.nc`. Check that these files have actually been created. If not, please ask for our help. If you set `veget=1`, the program `gcm.e` should also have run automatically for one day and created a start file for Orchidée, called `sechiba_rest_in.nc`.

- If you have the files `start.nc`, `startphy.nc` and `limit.nc` (and optionally `sechiba_rest_in.nc` for `veget=1`), you are almost done. In the file `config.def`, add the line:

  ```
  flag_pres__00003 =            4
  ```

  Thus, the NetCDF variable pres, which contains the pressure at model layers, will be included in the file #3, `histhf.nc`.

- You can now run the model by executing the command `./gcm.e`.

2   Experiments

The three subsections 2.1, 2.2 and 2.3 are independent.

2.1   Tracers

2.1.1   Radon and lead

In the file `traceur.def`, you can see that two trace species, named RN, for radon, and PB, for lead, are already defined in addition to vapor and condensed water. Visualize these two tracers in the file `histhf.nc`. The NetCDF variables RN and PB are numbers of atoms per unit mass, in kg^{-1}. Usually, activities are preferred for comparison with observations. The activities per unit mass, in $Bq\ kg^{-1}$, are:

$$A_{\mathrm{Rn}} = \frac{\mathrm{RN}}{4{,}765 \cdot 10^5}$$
$$A_{\mathrm{Pb}} = \frac{\mathrm{PB}}{1{,}028 \cdot 10^9}$$

(The numerical values that appear in these fractions are the lifetimes of radon and lead, in s.) You can take a look at LMDZ info number 7, figure 19, or the plot in the General introduction of the LMDZ model (§ 4 "Operating modes") to check that you have sensible results. A recent work on the modeling of tracers with LMDZ is Pilon et al. (2015, QJRMS). If you want to see activities per unit volume, you should divide by the mass density, using NetCDF variables pres, temp and ovap in the file `histhf.nc`. (The Ferret color palette in figure 19 of LMDZ info number 7 is saz2.)

2.1.2   Inserting new tracers

We will now explain how to modify the code in order to add idealized tracers. As an example, we will add two tracers, and call them `NewTr1` and `NewTr2`. We will define a domain of the horizontal grid in which the two tracers will initially have the same constant value. The first tracer will be transported by the boundary layer and convective sub-grid-scale motion, in addition to large-scale advection. The second tracer will only be transported by large-scale advection.

We will have to modify the Fortran program but let us first consider the run-time parameters that we have to modify. At run-time, we decide to include the tracers `NewTr1` and `NewTr2` in the simulation by changing the file `traceur.def`.

3

Change the number of tracers at the first line of `traceur.def` and append one line for each tracer:

```
6
14 14 H2Ov
10 10 H2Ol
10 10 RN
10 10 PB
10 10 NewTr1
10 10 NewTr2
```

Now let us turn to the Fortran program. The only file we need to modify is `phylmd/traclmdz_mod.F90`. Here are the changes you should make in that file.

- Declare two new module variables, `id_NewTr1` and `id_NewTr2`, with type integer. These are the identifying numbers of the tracers in the program. You can take a previous declaration (`id_pcsat`, `id_pcocsat` ...) as a template. (Do not forget the OpenMP directives.)

- All remaining changes will be made in the procedure `traclmdz_init`, which is inside the module `traclmdz_mod`. In order to define `id_NewTr1` and `id_NewTr2`, the program will scan the file `traceur.def`, looking for NewTr1 and NewTr2. So you should initialize `id_NewTr1` and `id_NewTr2` to 0 before the loop beginning at line 175. There is a comment just above, saying "Recherche des traceurs connus", which means "looking for known tracers".

- In the body of this loop, set `id_NewTr1` to the value of index `it` if tname(iiq) equals NewTr1. You can add the test near line 265, for instance, after the test for pcq0. You can take another tracer as a template. Do the same for `id_NewTr2`.

- For NewTr2, just after setting `id_NewTr2`, deactivate convective and boundary layer transport by setting `conv_flg(it)` and `pbl_flg(it)` to 0.

- Finally, we will initialize the tracers. There is a loop on tracers, beginning at line 294, which tests whether the intial tracer field, read from the file `start.nc`, is zero everywhere. At this point, the tracer field would also be zero if it was not found in `start.nc`. There is a comment just above the test which says "Initialize tracer that was not found in restart file". In the body of the test, for our two tracers, change the value at the surface in some horizontal domain. The value of tracers is in the variable `tr_seri`. The first dimension of `tr_seri` is for the horizontal position, the second dimension is for the vertical level and the third dimension identifies the tracer. The index of the surface in the vertical dimension is 1. Use variables `xlat` (latitudes) and `xlon` (longitudes) to choose the horizontal domain.

In summary, after making those changes, svn diff should give you something like this:

```
$ svn diff traclmdz_mod.F90
Index: traclmdz_mod.F90
```

4

```
====================================================================
--- traclmdz_mod.F90 (revision 2304)
+++ traclmdz_mod.F90 (working copy)
@@ -58,6 +58,8 @@
    LOGICAL,SAVE :: rnpb=.FALSE. ! Presence du couple Rn222, Pb210
 !$OMP THREADPRIVATE(rnpb)

+  INTEGER, SAVE:: id_newtr1, id_newtr2
+  !$OMP THREADPRIVATE(id_newtr1, id_newtr2)

 CONTAINS

@@ -172,6 +174,8 @@
 ! ---------------------------------------------
    id_rn=0; id_pb=0; id_aga=0; id_be=0; id_o3=0
    id_pcsat=0; id_pcocsat=0; id_pcq=0; id_pcs0=0; id_pcos0=0; id_pcq0=0
+   id_newtr1 = 0
+   id_newtr2 = 0
    DO it=1,nbtr
       iiq=niadv(it+2)
       IF ( tname(iiq) == "RN" ) THEN
@@ -262,6 +266,12 @@
       ELSE IF ( tname(iiq) == "pcq0" .OR. tname(iiq) == "Pcq0" ) THEN
          id_pcq0=it
          conv_flg(it)=0 ! No transport by convection for this tracer
+      else if (tname(iiq) == "NewTr1") then
+         id_newtr1 = it
+      else if (tname(iiq) == "NewTr2") then
+         id_newtr2 = it
+         conv_flg(it) = 0
+         pbl_flg(it) = 0
      ELSE
         WRITE(lunout,*) 'This is an unknown tracer in LMDZ : ', trim(tname(iiq))
      END IF
@@ -325,6 +335,9 @@
                   tr_seri(i,:,it) = 100.
               END IF
            END DO
+         else if (it == id_newtr1 .or. it == id_newtr2) then
+            where (xlat >= 40. .and. xlat <= 45. .and. xlon >=0. &
+                    .and. xlon <= 5.) tr_seri(:, 1, it) = 1.
          ELSE
            ! No specific initialization exist for this tracer
            tr_seri(:,:,it) = 0.
```

Re-compile the program:

```
cd some_path/LMDZ20151130.trunk/modipsl/modeles/LMDZ5/TUTORIAL
rm gcm.e
cd ..
```

If you ran `init.sh` with veget=0:

```
./makelmdz -d 48x36x39 gcm
```

If you set veget=1 in `init.sh`:

```
./makelmdz -d 48x36x39 -v true gcm
```

Rename restart files:

```
cd TUTORIAL
mv restart.nc start.nc
mv restartphy.nc startphy.nc
mv sechiba_rest_out.nc sechiba_rest_in.nc
```

Run the model:

```
../gcm.e
```

(could take about 15 mn). Visualize the two new tracers in `histhf.nc` and the difference between them.

2.2 Switching physics parameterizations

- Prepare for five more days of simulation: rename files `restart.nc` and `restartphy.nc` (created at the end of the first simulation) to `start.nc` and `startphy.nc`.

- Create a directory where you will run the model with modified `.def` files. In this new directory, you can create symbolic links to the files `start.nc`, `startphy.nc` and `limit.nc` which are in TUTORIAL, and you need to copy the `*.def` files from TUTORIAL, except those beginning with `used_`.

- In your new directory, in file `physiq.def`, change the boundary layer parameterization by deactivating the Mellor and Yamada scheme for the boundary layer, based on a prognostic equation for the Turbulent Kinetic Energy:

  ```
  iflag_pbl = 1
  ```

  instead of the nominal value 11 (which corresponds to the variant for very stable planetary boundary layer and exact dissipation). Thus, you will switch to the "Standard Physics" (SP, LMDZ5A).

- Run the model in the two directories and compare the results.

- Another test you can try is to deactivate the thermal plume model:

  ```
  iflag_thermals=0
  ```

  instead of the nominal value 15.

Remember that you can modify the content or time frequency of the output files if you want to focus on a particular question.

2.3 Sensitivity to the soil scheme

2.3.1 Running with ORCHIDEE 2-layers

Execute `init.sh` with `veget=1`. As explained in Section 1, the model is run automatically for one day and creates a start file named `sechiba_rest_in.nc` (`sechiba` is the hydrological part of the soil/vegetation model Orchidee).

Then you can run `gcm.e` to do a simulation with Orchidee activated. The number of days, set in `run.def`, is `nday=5`. To get results faster, decrease `nday` to 3, and change `5day` in `3day` in `config.def`, in the line 'phys_out_filetimesteps' (otherwise your output file `histmth.nc` will be empty).

You can play with the `sechiba` output frequency by changing in `orchidee.def` the variable `WRITE_STEP` (in seconds; default: `86400` for daily output); `0` means no `sechiba` output; `-1` means monthly output; `N*86400` means output written every N days). A second output file `sechiba_out_2.nc` is for high-frequency output, modulated by `WRITE_STEP2` (default: `10800`, for 3 hours)

You can change the complexity level of outputs by playing with the `SECHIBA_HISTLEVEL` variable: higher `SECHIBA_HISTLEVEL` means more variables in output. The variables corresponding to the various output levels are coded in

`modipsl/modeles/ORCHIDEE/src_sechiba/intersurf.f90`

2.3.2 Running with ORCHIDEE 11-layers

Set `HYDROL_CWRR` to `y` in `orchidee.def` (the default is `n`, meaning that ORCHIDEE 2-layers is used).

-> If it's the first time that you do a run with ORCHIDEE in your current working directory (which is the case if you don't have a `sechiba_rest_in.nc` file), the procedure is the same as described in 2.3.1. for ORCHIDEE 2-layers.

-> If you have already done a run with ORCHIDEE -2 layers, you need to re-create an initial state file adapted to the multi-layer hydrology: (i) remove the file `sechiba_rest_in.nc` in your working directory, (ii) set `SECHIBA_restart_in.nc` to `NONE` in orchidee.def, and (iii) relaunch the gcm.

2.3.3 Running with the simple bucket scheme

If `VEGET=n` (meaning that the vegetation is not activated) instead of `y` in file `config.def`, the soil scheme is a simple bucket (even if you compiled with `makegcm -v true` as done by `install.sh` when ran with `veget=1`).

2.3.4 Running with bucket scheme with imposed soil water content

Same as in 2.3.3., you should run with `VEGET=n` in `config.def`. Evaporation is computed as the potential evaporation multiplied by the aridity coefficient `vbeta`, which is a function of the soil water content `qsol0`:

`vbeta(i) = MIN(2.0*qsol/mx_eau_sol, 1.0)`

(here `mx_eau_sol=150mm`). So, if `qsol0` is constant, `vbeta` is constant as well. You can fix `qsol0` to a chosen value `qsol0_val` (in mm), by adding in `physiq.def` the line `qsol0=qsol0_val` ; try for example `qsol0_val=5` or 10, that result in `vbeta` values typical of summertime.

You can compare the turbulent fluxes for the austral summer (variables `flat` and `sens` in the LMZ output files) computed using the different options.

3 Nudging

- The program `ce0l.e` (which was run by `init.sh` in the first part of this tutorial) created the file `grilles_gcm.nc`. `grilles_gcm.nc` contains the longitudes and latitudes of the model grids (staggered grids for zonal wind, meridional wind and temperature). You can visualize the orography map as seen by the zoomed grid, by opening this file with `ferret` or `grads` and plotting the variable `phis` (which, despite its name, actually contains the surface altitude, in m). You can also easily plot the horizontal resolution of the model as the square root of the grid mesh area (variable `aire`, in m^2).

- We will nudge the model with reanalysis fields. The script `get_era.sh`, in the directory `TUTORIAL`, gets the reanalysis files. Run the script. It will also interpolate the winds on the model grid (by reading the model grid from file `grilles_gcm.nc`). You should end up with files `u.nc` and `v.nc` in your current directory. Note that for this tutorial we have given open access to a subset of the ERA-interim wind fields. ERA-interim files are stored at IDRIS, CCRT and Climserv, with restricted access. To access these files at IDRIS or on Climserv, you should contact Sophie Bouffies-Cloché (IPSL). For access at CCRT, contact Anne Cozic (LSCE). `get_era.sh` is a very simplified script for the tutorial, but more general scripts are available on the Subversion server of IGCMG.

- Take a look at file `guide.def`. Nudging is activated for variables u and v only (as is often the case). The relaxation time is set to 3 hours inside the zoomed area (`tau_max=0.125` days) and half an hour outside (`tau_min=0.0208333` days). The smaller the relaxation time, the stronger the nudging. You can change parameters in this file if you want.

- Add the line:

  `INCLUDEDEF=guide.def`

  in `run.def`.

- Rename your `hist*` files so that they are not overwritten by the next run, and delete the file `sechiba_rest_out.nc`.

- Run the model again with nudging:

  `../gcm.e`

- Compare the results of the simulations with and without nudging.