

# TD2-Traceurs:

Formation LMDZ

November 9, 2011

Ce TD utilise des configuration zoomée de LMDZ. Il contient la phase de création des états initiaux, l'utilisation du zoom et éventuellement du guidage. Si vous travaillez sur les machines du réseau LMD, commencez par vous créer un répertoire à votre nom puis à vous déplacer dessus.

## 1 Mise en place d'un cas avec zoom

Il faut commencer par se placer sur le répertoire LMDZ5 contenant les fichiers **makegcm**, **libf** ..., c'est à dire un répertoire du type

**LMDZ20110921.trunk/modipsl/modeles/LMDZ5**

Sur ce répertoire, vous récupérez le fichier tar suivant :

```
wget http://www.lmd.jussieu.fr/~lmdz/DistribG95/tutorial.tar
tar xvf tutorial.tar
cd TUTORIAL
```

Regarder les fichiers installés. Une fois sur le répertoire **TUTORIAL**, vous pouvez éditer le fichier **gcm.def** afin de changer par exemple la position du zoom décrite par les paramètres de longitude et latitude, **clon** et **clat**. Placez le zoom à votre endroit de prédilection.

Puis lancer le script **./init.sh**. Le script va d'abord compiler (en résolution 48×36-L39) le modèle et l'exécutable permettant de créer l'état initial (**ce0l.e**). Le script rapatrie ensuite de nouveaux fichiers, contenant le relief, des jeux de température de l'océan (AMIP), etc.

Il lance ensuite **ce0.e** qui va créer les fichiers **start.nc**, **startphy.nc** et **limit.nc**. Vérifiez que ces fichiers sont bien créés. Si ce n'est pas le cas, demandez de l'aide sans plus attendre.

Si les fichiers sont présents, vous pouvez alors lancer le modèle (**./gcm.e**).

## 2 Modification de paramètres

Dans **traceur.def**, vous voyez que les 2 traceurs RN(radon) et PB(plomb) y sont déjà. Commencer par visualiser ces 2 traceurs à partir du fichier **histhf.nc** par exemple.

Ensuite il s'agit de modifier le code pour ajouter des nouveaux traceurs idéalisés.

1. Tout d'abord, mettre à jour le modèle avec la dernière version avec **svn update**. La mise à jour va permettre d'accéder aux variables **xlat** et **xlon** dans **traclmdz\_init**.
2. Initialiser 2 nouveaux traceurs **NewTr1** et **NewTr2** dans la routine **traclmdz\_init**, dans **traclmdz\_mod.F90**. Utilisez les variables **xlat**(latitudes) et **xlon**(longitudes) pour définir un domaine dans lequel les traceurs auront par exemple une valeur constante imposée. En dehors de ce domaine, les traceurs sont uniquement advectés. Les 2 traceurs seront initialisés de façon équivalente. Le premier traceur sera transporté par la convection et la couche limite (en plus de la grande échelle). Pour cela mettre **conv\_flg(it)=1** et **pbl\_flg(it)=1** pour ce traceur. Désactiver le transport par la convection et le transport par la couche limite pour le deuxième traceur : **conv\_flg(it)=0** et **pbl\_flg(it)=0** pour ce traceur.

En pratique :

Dans la boucle "Recherche des traceurs connus", ajouter l'identification des traceurs "NewTr1" et "NewTr2" (autour de ligne 207). Initialiser les variables **id\_NewTr1** et **id\_NewTr2** avec l'index du traceur. Déclarer les variables des identifiants **id\_NewTr1** et **id\_NewTr2** en tête du module, et les initialiser les 0. Dans la boucle pour **NewTr1** ajoutez **conv\_flg(it)=1** et **pbl\_flg(it)=1** et pour **NewTr2** **conv\_flg(it)=0** et **pbl\_flg(it)=0**.

Dans la boucle "Initialise tracer that was not found in restart file", ajoutez l'initialisation du traceur à la surface, utilisant **xlat** et **xlon** (autour de la ligne 270).

Re-compiler.

3. Modifier `traceur.def` et lancer le modèle avec le nouveau exécutable.

```
> cat traceur.def
6
14 14 H20v
10 10 H20l
10 10 RN
10 10 PB
10 10 NewTr1
10 10 NewTr2

> ./gcm.e
```

4. Visualiser `histhf.nc` et regarder les différences entre les 2 nouveaux traceurs.

Extra aide pour modification de **traclmdz\_mod.F90** Si manque de temps.

```
[jgips1@dhcp-236 phylmd]$ svn diff
```

```
Index: traclmdz_mod.F90
```

```
=====
```

```
--- traclmdz_mod.F90 (r?vision 1579)
```

```
+++ traclmdz_mod.F90 (copie de travail)
```

```
@@ -51,6 +51,10 @@
```

```
    INTEGER, SAVE:: id_o3
```

```
    !$OMP THREADPRIVATE(id_o3)
```

```
+
```

```
+    INTEGER, SAVE:: id_NewTr1, id_NewTr2
```

```
+$OMP THREADPRIVATE(id_NewTr1, id_NewTr2)
```

```
+
```

```
! index of ozone tracer with Cariolle parameterization
```

```
! 0 means no ozone tracer
```

```
@@ -154,6 +158,7 @@
```

```
! -----
```

```
    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
```

```
@@ -204,6 +209,14 @@
```

```
        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
```

```
+            conv_flg(it)=1 ! Do transport by convection
```

```
+            pbl_flg(it)=1 ! Do transport by pbl
```

```
+        ELSE IF ( tname(iiq) == "NewTr2" ) THEN
```

```
+            id_NewTr2=it
```

```
+            conv_flg(it)=0 ! Do not transport by convection
```

```
+            pbl_flg(it)=0 ! Do not transport by pbl
```

```
        ELSE
```

```
            WRITE(lunout,*) 'This is an unknown tracer in LMDZ : ', trim(tname(iiq))
```

```
        END IF
```

```
@@ -267,6 +280,13 @@
```

```
            tr_seri(i,:,it) = 100.
```

```
        END IF
```

```
    END DO
```

```
+    ELSE IF (it==id_NewTr1 .OR. it==id_NewTr2) THEN
```

```
+        ! Initialize le traceur
```

```
+        DO i = 1, klon
```

```
+            IF (xlat(i) > 30 .AND. xlat(i) < 50 .AND. xlon(i) > 0 .AND. xlon(i) < 30) THEN
```

```
+                tr_seri(i,1,it) = 30
```

```
+            END IF
```

```
+        END DO
```

```
    ELSE
```

```
        ! No specific initialization exist for this tracer
```

### 3 Mise en place du guidage

1. Créer le fichier **grilles\_gcm.nc** . Pour cela, il faut relancer avec **ce01.e** après avoir ajouté la ligne **grilles\_gcm\_netcdf=y** dans **run.def** . Vous pouvez tracer le relief vu par la grille zoomée en ouvrant sous **ferret** ou **grads** le fichier **grilles\_gcm.nc** et en traçant la variable **phis** . Vous pouvez aussi tracer la résolution du modèle, calculée comme la racine carrée de l'aire de la maille (**aire** ).
2. Il faut maintenant récupérer des analyses pour le guidage. Pour ca, vous trouverez le script **get\_era.sh** dans le répertoire **TUTORIAL** . Exécuter ce script. Il va mettre un peu de temps pour récupérer les fichiers Era-Interim pour le vent.

Remarque : Pour cette TD, nous avons mis en disposition un mois de u et v sur le dods public de IDRIS (voir la variable indir). Normalement les fichiers ERAI sont stockés sur IDRIS, CCRT et Climserv mais en accès restreint. Pour avoir le droit à y accéder, il suffit de se prononcer auprès de Sophie Bouffies-Cloch  (IPSL) pour acc s   l'IDRIS ou Climserv et Anne Cozic(LSCE) pour un accs au CCRT. Les scripts qu'on utilise dans ce TD sont pr vu pour utilisation sur ulam(IDRIS) ou cesium(CCRT), une machine qui a acc s au fichiers ERAI.

**./get\_era.sh** extrait  galement des scripts ferret pour installer ces donn es r cup r es via svn sur

[http://forge.ipsl.jussieu.fr/igcmg/svn/CONFIG/LMDZOR/branches/LMDZOR\\_v4/CREATE/SCRIPT](http://forge.ipsl.jussieu.fr/igcmg/svn/CONFIG/LMDZOR/branches/LMDZOR_v4/CREATE/SCRIPT)

puis vous explique finalement comment modifier et lancer ce script, ce que vous  te invit s   faire maintenant.

3. Si vous avez bien travaill , vous allez vous retrouv  avec un r pertoire **ERAI/2011** contenant les champs de vent ERAinterim sur leur grille native (**u.201101.aphei.GLOBAL\_075.nc** ) et interpol s sur la grille du mod le **u.201101\_era.nc** . Il faut faire un lien entre les fichiers interpol s et des fichiers avec un nom g n rique sur le r pertoire **TUTORIAL** .

```
cd ../../TUTORIAL
ln -s ERAI/2011/u_201101_era.nc u.nc
ln -s ERAI/2011/v_201101_era.nc v.nc
```

4. Modifier  ventuellement le fichier **guide.def** .

```
ok_guide=y
guide_u= y
guide_v= y
guide_T= n
guide_P= n
guide_Q= n
tau_min_u=0.0208333
tau_max_u=0.125
tau_min_v=0.0208333
tau_max_v=0.125
```

Ici, dans ce **guide.def** , on active uniquement le guidage des variables u et v (c'est tr s souvent ce qu'on fait). Le temps de relaxation est de 3 heures   l'ext rieur du zoom (**tau\_max=0.125** jour) et 30 minutes   l'ext rieur (**tau\_min =0.0208333** jour). Plus le temps de relaxation(tau) est petit, plus le guidage est fort.

5. Ajouter la ligne **INCLUDEDEF=guide.def** dans **run.def** .
6. En essayant de ne pas  craser les sorties de la simulation pr c dente, lancer une simulation guid e avec la commande :

```
./gcm.e > out_guide 2>&1
```

Comparer les r sultats.