

Mini-projet contrôle de l'extension latitudinale de la Moussoon Africaine sur le Sahel

Cours M2 Modélisation numérique, Frédéric Hourdin

13 février 2026

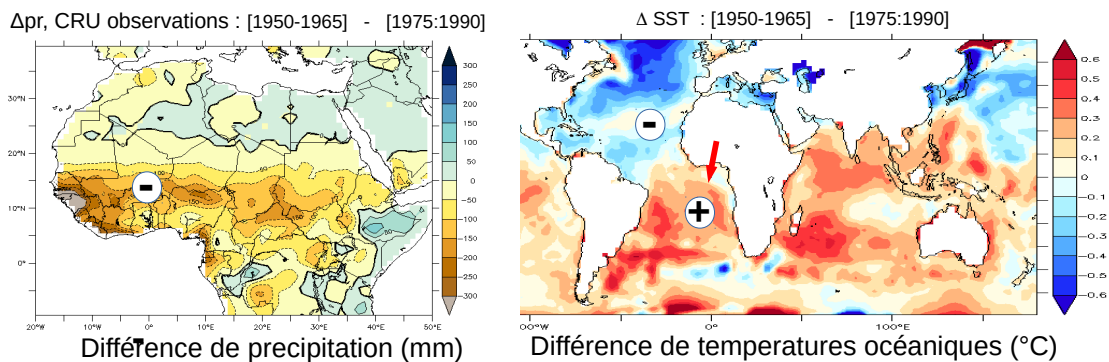


FIGURE 1 – Différence de températures de surface de l'océan et de précipitations entre les décennies 60 et 80, la première correspondant à une période particulièrement pluvieuse au Sahel et la seconde à la grande sécheresse Sahélienne. Les différence de pluies sont montrées en cumul annuel (mm).

1 Positionnement du sujet

Dans les modèles de circulations atmosphériques, des biais systématiques persistent.

Les modèles peinent notamment souvent à simuler correctement les pluies de mousson sur la bande sahélienne.

Les pluies ont par exemple une tendance systématique à ne pas progresser suffisamment vers le Nord dans le modèle de l'IPSL. L'erreur obtenue dans les simulations ressemble par beaucoup d'aspect au déficit de pluie enregistré dans les années 1970-1990.

Pour essayer de comprendre ce qui contrôle la position en latitude de la bande de précipitations de mousson sur la région, on mettra en place une simulation zoomée sur l'Atlantique tropical et l'Afrique de l'Ouest (disons sur [70W:30E] et [10S:30N]).

On pourra s'inspirer pour ce sujet de la référence suivante :

<https://web.lmd.jussieu.fr/~hourdin/PUBLIS/Ruti2011.pdf>

2 En pratique

Voici les principales instructions pour installer le modèle LMDZ pour le mini-projet.

Si le répertoire EnseignementLMDZ n'est pas installé :

```
git clone https://lmdz-forge.lmd.jussieu.fr/Git/fhourdin/EnseignementLMDZ.git
cd EnseignementLMDZ
```

S'il est déjà installé et qu'il faut faire des mises à jour :

```
cd EnseignementLMDZ
git pull origin master
```

Pour voir les options de la commande `exemples.sh` :

```
./exemples.sh --help
```

Puis installer la configuration particulière avec

```
./exemples.sh --config clim --grid 48x36x39 --parallel 1 -f 2. -x -15. -y 5.
```

Pour les options `--config` parmi `clim|slab|terra|aqua`, la commande `exemples.sh` permet de mettre en place typiquement une simulation de référence, qu'on pourra éventuellement accompagner de tests de sensibilité. En pratique, cette commande à (partir du répertoire TUTORIAL) crée un répertoire `config_IMxJMxLM` et :

1. crée un état initial (`start.nc` et `startphy.nc`) et des conditions initiales (`limit.nc`) sur le répertoire INITIAL. On peut visualiser la grille contenue dans `grilles_gcm.nc` and lançant le script `grille.jnl` sous `ferret` ou tout équivalent.
2. lance automatiquement une première simulation de quelques jours dans le sous-répertoire SIMU1. Cette simulation permet de contrôler que les fichiers de démarrage ont bien été créés et que la simulation peut tourner un certain nombre de jours sans planter.
3. crée enfin un répertoire PRODO qui va permettre d'enchaîner automatiquement des simulations mensuelles pendant quelques mois ou plusieurs années.

Pour exécuter de telles simulations, il est fortement conseillé de commencer par dupliquer ce répertoire

```
cp -r PRODO CTL
cd CTL
./enchaine.sh > out 2>&1 &
```

Le nom CTL n'est pas choisi au hasard. Il désigne une simulation de contrôle sur laquelle pourront être décliné des tests de sensibilité.

Avant de lancer la commande `enchaine.sh`, vous pouvez l'éditer pour changer par exemple la variable `stopsim=200101` par défaut. Avec cette valeur, la simulation qui commence par défaut le 1er janvier 2000, s'arrête avant le mois de janvier 2001.

Vous pouvez aussi changer les valeurs de certains paramètres dans les fichiers `*.def` contenus dans le répertoire CTL/DEF.

En pratique, chaque mois de simulation s'exécute sur des répertoires temporaires `WD*` (qui peuvent être détruit a posteriori). Les fichiers `start*` et `hist*` sont rapatriés sur EXP1. Le fichier `etat` contient l'information sur les mois faits ou à faire.

On peut lancer une nouvelle simulation à partir de la fin de la précédente. Par exemple si la simulation CTL a fait une année, on peut créer une expérience EXP1 :

```
cp -r PRODO EXP1
cd EXP1

# cree un fichier etat avec la ligne indiquant le mois a executer
cat > etat <<eod
200101 affaire
eod

# pointe sur les start et startphy de cette simulation
ln -s ../CTL/start.200101.nc .
ln -s ../CTL/startphy.200101.nc .
```

Bien sûr il faut prolonger la simulation CTL sur la même période pour s’y référer.

Quand faut-il relancer la commande `exemples.sh` ? Si vous voulez mettre en place une nouvelle configuration (choix différent de l’option `--config`).

Eventuellement aussi par souci de clarté si vous voulez lancer une nouvelle série de simulations avec un nouveau contrôle. Mais vous pouvez aussi gérer ça avec un CTL2 dans le répertoire `config_IMxJMxLM` existant.

Ici, on effectuera une première simulation de 3 ans avec afin de permettre aux réservoirs de surface de se se mettre à l’équilibre.

Le modèle de surface simplifié dit “bucket” utilisé ici est décrit en annexe.

On repartira de l’état final de cette simulation pour effectuer d’une part une simulation de contrôle et d’autre part des simulations de sensibilité.

2.1 Explication de la circulation de Mousson

Dans un premier temps, on essaiera de documenter et expliquer le phénomène de la mousson Africaine en contrastant par exemple un mois de février et un mois d’août en termes d’ensoleillement, de températures de surface océaniques, de circulation dans les basses couches, d’humidité près de la surface et de précipitations.

On pourra en particulier décrire la circulation dans un plan méridien en moyennant les variables sur une bande de longitude 10W:10E.

Sous ferret

```
reg/x=-10.:10./y=-10:30.
fill temp[i=@ave]
contour/o vitu[i=@ave]
```

2.2 Sensibilité de la mousson aux conditions de surface

On étudiera comment la mousson est sensible aux température de surface océaniques en ajoutant un “patch” de températures chaudes sur le golfe de Guinée. Une bonne partie

de la sécheresse Sahélienne était sans doute expliquée par la persistance pendant les années concernées d'une anomalie chaude dans les océans sud et froide dans l'Atlantique Nord. Retrouve-t-on ce type de réponse dans la modèle ?

On étudiera également comment un changement d'albédo sur le Sahara ou le Sahel peut perturber les gradients latitudinaux d'énergie et, en retour, a pénétration du flux de mousson sur le continent.

On pourra essayer de diminuer ou augmenter la rugosité sur la bande côtière et regarder comment ce changement de rugosité affecte la circulation et les flux d'énergie. Se reporter à l'annexe pour ce faire.

Dans tous les cas, on se basera sur des simulations d'un mois de Juillet, ou deux mois (Juillet-Août) et on s'intéressera plus particulièrement aux changement de circulations moyenne et de précipitation, en moyennant les champs sur une bande de longitude [10W:10E].

On pourra aussi regarder comment cette sensibilité aux conditions de surface est modifiée si on guide le vent et la température partout dans le domaine. Pour cela, on se référera à l'installation du guidage rappelé dans l'annexe C et on modifiera les constantes de guidages, pour prendre 1 jour à l'intérieur du zoom, de façon à contraindre la dynamique même dans la région zoomée.

A Modification du fichier `limit.nc`

En pratique, on peut utiliser pour modifier les conditions au limites comme la SST ou l'albedo et la rugosité quand le modèle de surface orchidee n'est pas utilisé, grace à la fonction python `shapemylimit.py` récupérable à l'adresse suivante

<https://lmdz.lmd.jussieu.fr/pub/Training/2022FormationMaroc/shapemylimit.py>

Le script modifie le fichier `limit.nc` et sauvegarde de l'original en tant que `limit_back.nc`. Pour afficher la figure qui se crée avec la variable originale, altérée, et la différence, il faut activer le paquet `basemap` ce qui peut se faire comme suit avec `conda`

Sur les ordinateurs de la formation :

```
conda create -n basemap python=3.7 basemap proj4=5.2.0
conda activate basemap
python -c "from mpl_toolkits.basemap import Basemap"
python shapemylimit.py
```

sur spirit :

```
conda activate
conda create -n basemap python=3.7 basemap proj4=5.2.0
conda activate basemap
python -c "from mpl_toolkits.basemap import Basemap"
ipython3 shapemylimit.py
```

B Le modèle bucket

Le modèle d'hydrologie est utilisé Les variables `qsol0` et `qsolinp` contrôlent l'évaporation dans un modèle d'hydrologie de surface dit « bucket » ou saut d'eau. Le réservoir en eau du

sol se remplit en fonction du bilan entre précipitation et évaporation :

$$dh/dt = PE \quad (1)$$

où q est en mm d'eau (kg/m^2). Si $h > 150\text{mm}$ on suppose que l'eau ruisselle ailleurs. L'évaporation elle même s'exprime en fonction de l'évaporation potentielle (celle d'une surface d'eau libre dans le mêmes conditions) et un coefficient d'aridité β :

$$E = rCd||V||\beta[q_{\text{sat}}(T_s) - q_{\text{air}}] \quad (2)$$

où le coefficient d'aridité est 1 pour $h > 75$ mm et décroît pour des valeurs plus faibles, limitant l'évaporation pour manque de disponibilité en eau :

$$\beta(h) = \min(h/75\text{mm}, 1) \quad (3)$$

`qsolinp` est la valeur initiale de h et `qsol0` une valeur imposée tout au long de la simulation. Si `qsol0` n'est pas donné, alors h évolue sous l'effet de la précipitation et de l'évaporation.

C Guidage en vent

Le modèle peut être guidé en vent (et éventuellement en température et humidité) pour forcer la situation synoptique à suivre la situation observées.

Le guidage consiste à rajouter un terme de relaxation vers les réanalyses X_{rea} aux tendances du modèles $M(X)$, disponibles toutes les 6 heures (on interpole linéairement ces champs à chaque pas de temps du modèle), avec une constante de temps τ :

$$\frac{\partial X}{\partial t} = M(X) + \frac{X_{rea} - X}{\tau} \quad (4)$$

En imposant ainsi la circulation à grande échelle, on se concentre sur l'impact des paramétrisations physiques et on limite le caractère chaotique de l'écoulement, et donc la nécessité de réaliser des grands ensembles de simulations pour avoir des résultats statistiquement significatifs.

Le guidage peut être activé pour les `config clim` et `slab`¹ distribuées avec `./exemples.sh`.

Avec le fichier `guide.def` proposé par défaut on guide les deux composantes horizontales du vent avec une constante de temps de 6 heures en dehors du zoom et 10 jours à l'intérieur. Avec ces guidage, c'est un peu comme si on forçait la circulation à l'extérieur du zoom et qu'on la laissait libre à l'intérieur.

En pratique, pour faire fonctionner le guidage il faut :

1. disposer des fichiers de guidage `u.nc` et `v.nc`
2. ajouter une ligne `INCLUDEDEF=guide.def` dans `run.def`
3. Utiliser un calendrier "vrai" pour que les mois aient la bonne longueur

La création des fichiers de guidage doit se faire préalablement avec le script `get_era.sh`. Pour lancer la récupération/interpolation

```
./get_era.sh 200001
```

1. Il fonctionnerait aussi pour `aqua` et `terra` mais pas évident de voir à quoi l'utiliser.

pour le mois de janvier 2000, ou, pour toute l'année 2000 :

```
year=2000
for ym in $( seq ${year}01 ${year}12 ) ; do ./get_era.sh $ym ; done
```

² Les fichiers sont stockés sur le répertoire `GUIDAGE/2000` dans ce cas.

Le reste est pris en charge par l'option `nudging=1`, à changer en dur dans `enchaine.sh`.
On utilise le guidage classiquement avec Deux objectifs assez différents :

climat régional : c'est la configuration par défaut. On contraint fortement le modèle à l'extérieur du zoom mais on le laisse quasiment libre à l'intérieur. Cette utilisation est proche d'un modèle à aire limitée.

études physiques à dynamique contrainte : , en mettant les mêmes constantes de guidage à l'intérieur et à l'extérieur du zoom. Typiquement 1 jour. Le modèle ainsi guidé suit les situations synoptiques même à l'intérieur du zoom, ce qui permet de travailler sur la physique de la "colonne atmosphérique" et son couplage avec la surface par exemple, à dynamique imposée, réduisant ainsi le caractère chaotique des simulations.

2. La commande `set 200001 200012` crée la séquence de nombres "200001 200002 ... 200012", et `$(commande)` renvoie le résultat de la commande.