

Module HAT918T  
Outils de modélisation hydro-morphodynamique littorale et portuaire  
**Maîtrise de SLURM et calcul parallèle sous  
python (pour l'analyse de données)**  
Session 06-07

Louis Saisset

Ce TD est la suite de la prise en main des outils de modélisation hydro-morphodynamique littorale et portuaire. Pour prendre en main les lancements sur cluster, nous utiliserons les simulations D3D réalisées durant le TP2. Nous effectuerons le traitement des données directement sur le cluster, en utilisant un programme python fourni.

## Introduction

L'objectif de ce TP est de vous permettre de prendre en main les lancements de simulation et de post-traitement directement sur le cluster. Pour cela, vos objectifs vont être :

1. Lancer une série de simulations D3D faisant varier les paramètres principaux de calibration du modèle. (cf Partie 1)
2. Lancer le post-traitement python lui aussi sur le cluster pour récupérer uniquement les informations essentielles à la calibration
3. Déterminer le set de paramètres optimal pour cette simulation en faisant les affichages du post-traitement sur votre ordinateur personnel.

Avant de commencer le TP, il vous faudra avoir accès à un environnement python possédant les bibliothèques nécessaires à la phase de post-traitement. Pour cela, il vous faut mettre en place un environnement conda puis y installer les différentes bibliothèques. Connectez vous sur le cluster puis suivez les instructions suivantes :

---

```
1  \$$ # Créez en environnement python (ici appelé "maupiti")
2  \$$ conda create --name maupiti python=3.7
3  \$$ # Acceptez de lancer l'installation des bibliothèques de base puis patientez
4  \$$
5  \$$ # Comme indiqué à la fin de l'installation, activez l'environnement pour
6  \$$ # avoir accès à son contenu
7  \$$ conda activate maupiti
8  \$$ # il est possible que conda vous indique un message : "IMPORTANT: ..."
9  \$$ # Suivez ses instructions.
10 \$$ conda init
11 \$$ exit
12 \$$ # Reconnectez vous au cluster puis retentez d'activer l'environnement
13 \$$ conda activate maupiti
14 \$$ # installez dans cet environnement l'ensemble des bibliothèques voulues
15 \$$ conda install pandas numpy scipy pyproj netCDF4
16 \$$ # Patientez.
```

---

Maintenant que vous possédez un environnement python correctement configuré, copiez le dossier TP\_Hydro\_SLURM dans votre répertoire personnel.

---

```
1 \ $ cp -r ~/work_f_gcl/TP_Hydro_SLURM/ ~/
```

---

Ce dossier contient :

- Un dossier `SimulTest` contenant les fichiers nécessaires au lancement d'une simulation type (voir TP Delft3D)
- Un dossier `Mesures` contenant un fichier de localisation des instruments et un fichier de mesures de Hs pour les 6 sondes du profil SW (voir TP Delft3D).
- Un fichier exécutable `LAUNCH_PY.cmd` contenant les instructions SLURM classiques pour lancer le post-traitement sur le cluster.
- Un fichier exécutable `post_treatment.py` contenant le programme python de post-traitement des données de simulation.

## 1 Lancements multiples

### 1.1 Rappels du TP précédent

Les lectures et écritures de fichiers sont des phases particulièrement coûteuses en temps. Le cluster MUSE possède par conséquent une zone dédiée à cela : `~/scratch`. Pour optimiser les temps de calculs dans le cas de très grosses simulations, il faut donc toujours travailler à partir de ce dossier.

Comme lors du TP D3D précédent, il va vous falloir copier le dossier de simulation `SimulTest` dans la zone de calculs `~/scratch`, vérifier que le fichier de commande `maupiti.mdw` est correctement complété, puis lancer la simulation depuis son dossier sur `/scratch` grâce à la commande `sbatch submitter_dwave_parallel.cmd`.

### 1.2 Introduction à la calibration

L'objectif du modèle numérique fourni est de représenter au mieux les valeurs de Hs lors de la propagation de la houle au dessus de la barrière récifale SW. Pour acter la validité d'un tel modèle numérique, on passe souvent par une démarche dite de **calibration-validation** qui consiste à prouver que le set d'informations choisies pour la modélisation numérique donne le meilleur résultat possible comparé à des mesures existantes. Dans notre cas, ces mesures correspondent aux mesures de Hs pour les 6 pressiomètres sur le profil SW.

Parmi les informations classiques à fournir à un modèle hydromorphodynamique, on pense rapidement au fichier de bathymétrie pour le système étudié et à une version numérique des forçages météo-marins ayant eu lieu lors des mesures sur le terrain. Dans notre cas, nous considérerons que la bathymétrie est réaliste et que les forçages météo-marins sont correctement représentés par les spectres de houle fournis. Peut-on pour autant considérer que le modèle numérique représente au mieux la propagation de la houle ?

## 1.3 Paramètres de calibration

Vous n'êtes pas sans savoir que les interactions avec le fond peuvent considérablement changer la propagation et la dissipation de la houle. Diverses caractéristiques entrent alors en jeu en plus de la géométrie globale des fonds et des caractéristiques de la houle de départ.

Les récifs coralliens sont des systèmes à géométrie très complexe. Il est difficile (voire impossible) de mesurer ou d'établir théoriquement la totalité de leurs caractéristiques physiques. Ainsi, la porosité, la rugosité et la position exacte du fond de la colonie corallienne ne nous sont pas connues. De plus, les différents paramètres adimensionnés contrôlant la propagation et la dissipation de la houle dans le modèle numérique ne sont pas adaptés au système étudié. Il nous est donc impossible d'établir *a priori* les valeurs des paramètres suivants :

- Le paramètre de déferlement géométrique (*BreakGamma* dans le fichier de paramètres)
- Le coefficient de friction au fond (*BedFricCoef* dans le fichier de paramètres)
- La section disponible pour le transfert de l'énergie au dessus du récif, qu'on représentera par un ajout d'eau dans tout le système (augmentation  $L_\alpha$  de la valeur de *WaterLevel* dans le fichier de paramètres).

Une fois la première simulation type lancée, choisissez un paramètre à faire varier ( $L_\alpha$ , Coefficient de friction ou paramètre de déferlement géométrique) et lancez autant simulations qu'il vous semble nécessaire pour déterminer la meilleure valeur de ce paramètre pour notre système. Vous prendrez comme valeur par défaut dans le fichier `maupiti.mdw` :

- `WaterLevel = 9.359706959706959317e-01`
- `BreakGamma = 5.000000000000000000e-01`
- `BedFricCoef = 5.000000000000000000e-02`

## 2 Post-treatment

Les bienfaits d'un traitement sur le cluster sont nombreux. En effet, pour traiter les simulations sur votre machine personnelle, il vous faudra toujours télécharger les résultats des simulations. Ce processus est souvent long et coûteux en mémoire pour votre ordinateur. A l'inverse, sur le cluster, la donnée est déjà présente et disponible pour le calcul.

Un programme python permettant l'extraction des résultats de simulation sur le profil SW vous a été fourni. Ce programme python `post_treatment.py` peut être lancé sur votre machine, ou sur le cluster en utilisant `LAUNCH_PY.cmd`. Copiez ces deux programmes dans votre dossier `TOOLS` pour pouvoir les exécuter depuis n'importe quelle position dans l'arborescence. Une fois ces fichiers dans `TOOLS`, vous devriez pouvoir trouver ces commandes avec l'auto-complétion. Pour que ces programmes fonctionnent, il vous faudra TOUJOURS les lancer depuis le dossier contenant les résultats d'une simulation D3D. Pour comprendre le rôle de chacun de ces programmes, lisez rapidement leur contenu.

Pour bien comprendre la différence entre un lancement sur le noeud de connexion et un véritable lancement sur la partition du cluster dédiée au calcul, ouvrez deux terminaux et connectez vous dans les deux cas sur le cluster.

### Premier cas :

1. Dans le premier terminal, tapez la commande `htop`. Vous verrez alors l'ensemble des tâches actuellement en exécution sur le noeud de connexion que vous êtes en train d'utiliser.
2. Sur l'autre terminal, déplacez vous vers un dossier contenant les résultats d'une simulation D3D (par exemple `SimulTest`) et lancez y le programme de post-traitement `post_treatment.py`<sup>1</sup>.
3. Vous devriez alors voir apparaître puis disparaître votre nom et votre tâche sur la liste des tâches en cours dans le premier terminal.

Puisque ce calcul est visible sur le noeud de connexion, c'est que vous êtes en train d'en utiliser les ressources. Honte à vous !

### Deuxième cas :

1. Gardez le premier terminal ouvert sur l'écran `htop`.
2. Sur l'autre terminal, déplacez vous vers un dossier contenant les résultats d'une simulation D3D et lancez y le programme `sbatch LAUNCH_PY.cmd`.
3. Le premier terminal ne devrait voir passer AUCUNE commande.
4. Préparez vous à être rapide. relancez la même commande que précédemment puis lancez la commande `squeue`. Vous devriez voir apparaître la liste des calculs en cours sur la partition de calcul. Avec un peu de chance, vous verrez apparaître et disparaître le programme de post-traitement.

Puisque le noeud de connexion ne voit pas votre calcul et que SLURM vous a transmis un numéro de "job" c'est que votre requête a été effectivement lancée sur la partition de calcul. Vous évitez donc d'encombrer le noeud de connexion. Toutes les informations à propos de votre calcul se trouvent maintenant dans le fichier `*.out` écrit par SLURM dans le dossier où vous avez lancé le programme.

### Troisième cas :

1. Ouvrez le programme `LAUNCH_PY.cmd`. Vous y trouverez les instruction SLURM de base. Dans l'ordre :
  - `#SBATCH -J` pour le nom de la simulation
  - `#SBATCH -p` pour le nom de votre groupe de calcul (ici toujours "gladys")
  - `#SBATCH -t` le temps maximal prévu pour la simulation
  - `#SBATCH -o` le nom du fichier de sortie
  - `#SBATCH -e` le nom du fichier d'erreur
  - `#SBATCH -N` le nombre de noeuds

---

1. Il est possible que vous deviez changer les chemins d'accès contenus au début du programme python

- `#SBATCH -n` le nombre de tâche total
- `#SBATCH --ntasks-per-node` le nombre de tâches par noeud
- `#SBATCH --ntasks-per-core` le nombre de tâche par coeur (souvent 1)

2. Modifiez les paramètres SLURM et relancez la commande pour en comprendre le sens.

Maintenant que vous savez effectuer des simulations et le post traitement sur le cluster, récupérez les fichiers `*.txt` générés par le post traitement et déterminez le set de paramètres optimal pour simuler l'évènement.

BON COURAGE!