17 PySPH dans un environnement virtuel Python sur RO-MEO (10/02/2022)

17.1 Compilation et installation de Zoltan

Pré-requis

- `module purge`
- `module load 2021/openmpi/4.0.5.1-gnu10.2.0`
- pour savoir où sont les includes et librairies de OpenMPI : `mpicc -showme`
- dans ce cas le répertoire racine d'OpenMPI est :
  `/apps/2021/openmpi/4.0.5.1-gnu/`

Compilation et installation de Zoltan

- dans `~/test_install/` : `tar xvfz Zoltan-3.901.tar.gz`
- création d'un répertoire : `mkdir Zoltan`
- `cd Zoltan`
- `../Zoltan-3.901/configure --with-cflags=-fPIC --enable-mpi --prefix=/home/dcordier/tes`
- on lance la compilation : `make everything`
- installation de Zoltan : `make install`

Il faut à présent renseigner les variables d'environnement :

```
export ZOLTAN_INCLUDE=/home/dcordier/test_install/Zoltan/include
export ZOLTAN_LIBRARY=/home/dcordier/test_install/Zoltan/lib
```

17.2 Installation de PySPH

Pré-requis

- `module purge`
- `module load 2021/openmpi/4.0.5.1-gnu10.2.0`
- `module load python/3.9.0_spack2021_gcc-10.2.0-ukyf`
- pour voir si cela marche : `module load cuda/11.4`
- variables pour Zoltan :
  ```
  export ZOLTAN_INCLUDE=/home/dcordier/test_install/Zoltan/include
  export ZOLTAN_LIBRARY=/home/dcordier/test_install/Zoltan/lib
  ```

Création de l'environnement virtuel pour PySPH

- dans `~/PYTHON_ENV/` : `python3 -m venv env_PySPH`
- `source env_PySPH/bin/activate`
- `pip install --upgrade pip`
- `pip install --upgrade numpy`
- `pip install --upgrade pyproject.toml`
- `pip install Cython mako pytest --upgrade`
- `pip install cyarray --upgrade`
- `pip install Beaker --upgrade`
- `pip install compyle --upgrade`
- `pip install meshio --upgrade`
- `pip install mpi4py --upgrade`

- `pip install Scipy --upgrade`
- `pip install matplotlib --upgrade`
- `pip install h5py --upgrade`
- `pip install pycuda --upgrade`
- `pip install cupy --upgrade` **TRES LONG**
- `pip install pyzoltan --upgrade`
- `pip install pyopencl --upgrade` **ATTENTION** il vaut mieux ne pas installer `pyopencl` (et sans doute les modules `cuda`) cela bloque la phase de test au premier item !

Installation de PySPH

- `git clone https://github.com/pypr/pysph.git`
- `cd pysph`
- `python setup.py install` **PAS MAL DE WARNING** : Finished processing dependencies for PySPH==1.0b1.dev0

Séquence de tests de PySPH

- `pysph test -v`

```
-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
================================================================================
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_aa - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_ab - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_ba - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_bb - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_cc - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_neighbors_dd - ModuleNotFoun
FAILED base/tests/test_nnps.py::ZOrderGPUNNPSTestCaseCUDA::test_repeated - ModuleNotFoundErr
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_aa - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_ab - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_ba - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_bb - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_cc - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_neighbors_dd - ModuleN
FAILED base/tests/test_nnps.py::ZOrderGPUDoubleNNPSTestCaseCUDA::test_repeated - ModuleNotFo
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_aa - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_ab - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_ba - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_bb - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_cc - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_neighbors_dd - Module
FAILED base/tests/test_nnps.py::TestZOrderGPUNNPSWithSortingCUDA::test_repeated - ModuleNotF
FAILED parallel/tests/test_openmp.py::TestOpenMPExamples::test_ldcavity_example - RuntimeErr
====================================================== 22 failed, 664 passed, 204 ski
```

REMARQUE IMPORTANTE : pour exécuter le script `db2d.py`, écrit par Bastien, il faut avant de lancer l'exécution renseigner la variable d'environnement `export HDF5_USE_FILE_LOCKING` :

```
export HDF5_USE_FILE_LOCKING=FALSE
```

sinon on obtient un message d'erreur. Ensuite l'exécution se lance facilement avec :

```
python db2d.py
```