High Performance Computing with Python

Execution Instructions

THEODOROS ATHANASIADIS

Matr. Number: 5365502

theodorathanasiadis@gmail.com
University of Freiburg

August 15, 2022

Execution Instructions

To execute locally:

1. Create an anaconda environment as follows:

   ```
   conda create --name hpc python=3.8 -y
   conda activate hpc
   pip install -U pip
   pip install -r requirements.txt
   ```

2. Navigate into `src` directory.

3. Execute the `sliding_lid_parallel.py` using mpirun and the following command line arguments:

   - Use the mpirun `-n` command line argument to specify the number of cores.
   - Use the `-g` command line argument of `sliding_lid_parallel.py` to specify the lattice's x- and y-dimensions.
   - Use the `-d` command line argument of `sliding_lid_parallel.py` to specify the desired domain decomposition in the x- and y-dimensions.
   - Use the `-o` command line argument of `sliding_lid_parallel.py` to specify the collision frequency.
   - Use the `-v` command line argument of `sliding_lid_parallel.py` to specify to specify the lid velocity.

Example Execution

Execute the sliding lid simulation in parallel on a 300x300 grid with collision frequency equal to 1.7, lid velocity equal to 0.1 for 10000 steps. Use 4 cores and 2x2 discretization.

```
mpirun -n 4 python sliding_lid_parallel.py -g 300 300 -d 2 2 -o 1.7 -v 0.1
```

To get further information on the command line arguments available for `sliding_lid_parallel.py` execute:

```
python sliding_lid_parallel.py -h
```

The execution will create two `npy` files for the components of velocity. Execute the following script to visualize the end result

```
python visualize_velocity_field_ilias.py
```

For additionally information on the parallel execution as well as the execution of the rest of the experiments please advise the report's Github repository `https://github.com/theodorju/fr-hpcpy-pub`.