Documentation for Running the Scaling Code

Code description

The code solves Boussinesq equations in 360^3 grid points using MPI parallelisation. The time step size $dt = 0.01$ and the code will be executed for total time unit of $T = 0.2$ such that the total number of time steps will be $T/dt = 20$. A detailed step by step procedure for running the code is given below. The total wall clock time taken by the code will be saved in a file named *Total_ Time_ taken.txt*.

Steps to run the code

- To use mpi4py package, install *intelpython* and *intel MPI*.
- Create 20 directories with names t_0, t_1 and so on upto t_20 such that data in each time will be saved to the respective directories. Note that the code saves data in each time step and hence number of directories should be same as the number of time steps.
- A sample shell script is also provided in the folder (*create_ directories.sh*) which can be used to create 20 directories by executing it from the terminal as *./create_ directories.sh*.
- We would like to know the scaling if we use 90 and 120 cores from one machine, 180 cores from 2 machines and 360 cores from three machines.
- It will be good to have a table as below.

No. of cores	**No. of machines**	**Time taken for 20 steps**	**Time taken for 1 step**
90	1		
120	1		
180	2		
360	3		

- In order to change the total number of time steps, change the value of T in the code such that the total time step will be decided by T/dt. Note that the number of directories should be same as the number of time steps.