Distributed Computing and Introduction to High Performance Computing

Ahmed Ratnani[1]

[1] Mohammed VI Polytechnic University, Benguerir, Morocco

Outline of this lecture

- Motivations and Goals of this course
- Prerequisites, Computer and software requirements
- Overview of the material used in this course
- Example: Matrix-Matrix multiplication using OpenMP

Motivations

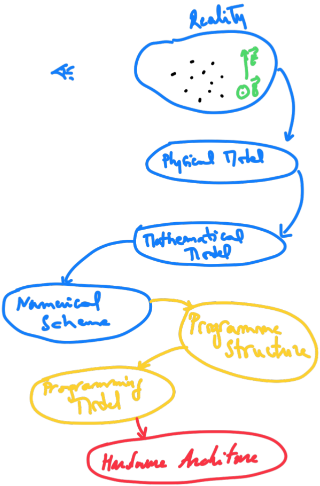
- In many applications, used softwares or developed codes tend to be slow :
  - Heavy computations
  - Huge data
- Two kind of applications
  - Numerical simulations
  - Data Analytics & ML/AI
- High Performance Computing (HPC) usually means heavy computations on clusters or supercomputers. But that's not all
  - HPC is different from Ditributed Computing
  - We can use HPC techniques on a laptop or a Rasberry-Pi!

Motivations

Numerical simulations

Relevant engineering problems require performance that is orders of magnitude higher than what is available

- CFD and Plasma-Physics: Simulation of turbulence at a reasonable level of resolution
- Combustion: Combination of turbulence simulation and realistic chemical models
- Climate simulation: Resolution required that is orders of magnitude higher than today



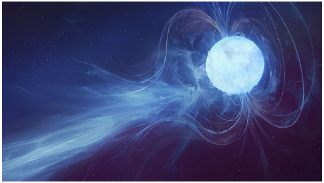
- Computation *vs* Data ?
  - ➠ Computational Intensity (see later)
- Can the numerical scheme be parallelized?
- How to manage Data? Distributed *vs* Shared ?
- Why do we need to know the Hardware Architecture?

➠ Why can't just say `cc -Parallel mycode.c` ?

Motivations

HPC Applications

- Astro-Physics
- Particle Physics
- Plasma Physics
- CFD
- Weather & Climate
- Artificial Intelligence
- Genomics / Bio-Informatics
- Molecular Dynamics
- Big Data Analytics
- Financial
- Cyber Security
- . . .



Requirements

Prerequisites

- Some programming experience in some language,
- You should be familiar with Python language,
- You should have knowledge of Fortran or C,
- You should have knowledge of linear algebra,

Requirements

Computer/Software

- We will not use Windows, only UNIX or LINUX will be used
- A Laptop is enough, but you will also get access to our supercomputer
- We will need the following tools:
    - Fortran or C compiler; instructions on installation will be given
    - MPI; instructions on installation will be given
    - Python; a requirements file will be provided to create a virtual environement
    - a github account

Material used in this course

Codes on **github**

Install compilers and libraries

```
sudo apt update
sudo apt install gfortran
sudo apt install libblas-dev liblapack-dev
sudo apt install libopenmpi-dev openmpi-bin
sudo apt install git
```

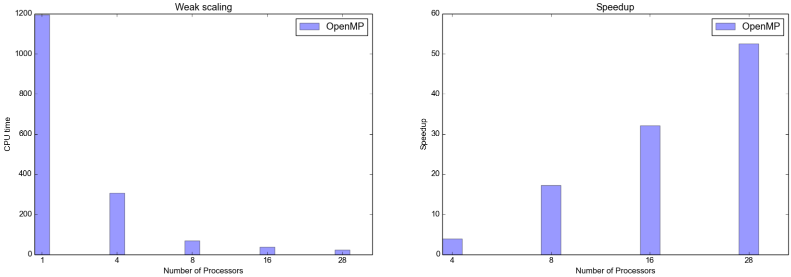
Cloning the Course directory

```
git clone git@github.com:UM6P/Distributed-Computing-HPC.git
```

Example

Matrix-Matrix multiplication using OpenMP

```
#$ omp parallel
#$ omp do schedule(runtime)
for i in range(m):
  for j in range(n):
    c[i, j] = 0.0
    for k in range(p):
      c[i, j] += a[i, k] * b[k, j]
#$ omp end do
#$ omp end parallel
```



Matrix-multiplication results for (n,m,p) = (5000,7000,5000). (left)CPU time in seconds

Outline of this Course

- Data Locality
- Accelerating Python codes
- Parallel Programming using MPI
- Parallel Programming using OpenMP
- Introduction to Parallel Programming for GPUs