Hands on iPic3D

This hands-on session is divided in three parts. First, we will learn how to run a magnetic reconnection simulation with iPic3D. You will then be able to run a small and short simulation of kinetic magnetic reconnection yourself. Then, we will learn how to visualise iPic3D results. Finally, we will examine the outcome of longer, bigger simulations already run.

1 Part 1: how to run an iPic3D simulation

iPic3D [9] is a 3D3V, pure MPI code for Particle-In-Cell (PIC) plasma simulations which relies on the Implicit Moment Method [8].
To run an iPic3D simulation of magnetic reconnection, follow the instructions here:

1. copy and untar the iPic3D code, `iPic3D-master.tar`, in your home folder
2. iPic3D uses the HDF library [4] for storing and managing output data. Load the HDF library by typing

   ```
   module load hdf5/1.8.9
   ```

   or add it to your ∼`/.bashrc` file.
3. the main file is `PICFOLDER/iPIC3D.cpp`. Inspect it to identify the main phases of a PIC code, depicted in Figure 1:

   - field initialisation: line 95;
   - particle initialisation: line 106;
   - moments deposition: line 184;
   - field solver: line 226;
   - particle mover: line 230.

   The field and particle initialisation already loaded, `initDoubleHarris` and `maxwellian`, are used to simulate a double Harris sheet [7], a standard initial configuration for magnetic reconnection. Two current layers are present, but only the bottom one is perturbed to initiate reconnection. You can find field and particle initial conditions for different problems in `fields/EMfields3D.h` and `particles/Particles3D.h`.
4. compile the application by typing

   ```
   module load hdf5/1.8.9
   make clean
   make
   ```

   in the main folder.
5. inspect the script we will use to launch a batch job on Judge, `scriptJudge`. Use [5] and the annotations in the script to understand the MSUB options.
   ** If you want to change the number of cores used in the simulation change `XLEN` and `YLEN` in `processtopology/VCtopology3D.h` so that `XLEN*YLEN= NSLOTS`.

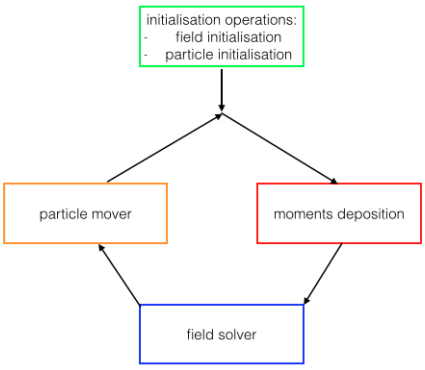

Figure 1: Main phases of a PIC simulation.

6. inspect the input file containing the simulation parameters, `DP.inp`. Use the annotations to understand the meaning of the input parameters.
   ** If you want to change the physical parameters of the simulation (magnetic field $B0X$, $B0Y$, $B0Z$, mass ratio qom, velocities, half thickness of the current sheet $delta$), be sure of respecting the relations between quantities from the Harris initial conditions. These quantities are marked as `!!! Harris!!!` in the input file.
   ** If you want to change the temporal dt and spatial dx resolution of the simulation, check the Implicit Moment Method stability constraint:

$$0.01 < v_{th,e}\frac{dt}{dx} < 1, \tag{1}$$

   where $v_{th,e}$ is the thermal velocity of electrons. You can obtain the spatial resolution dx, dt by dividing the domain length Lx, Ly by the number of cells nxc, nyc per direction. Notice that `nxc/ XLEN`, `nxc/ YLEN` must be an integer.

7. type

```
msub scriptJudge
```

   to launch the simulation. You can check the status of the simulation with

```
showq -u $USER
checkjob JOB NUMBER
```

8. inspect the error and output files, `PICFOLDER/testDP.e` and `PICFOLDER/testDP.o`, and check for the successful completion of the simulation. The last lines of `PICFOLDER/testDP.o` show the execution time of the application, note it down. You can use this metric for scaling tests

9. inspect the content of the `PICFOLDER/dataGEM` folder, where the `*.hdf` output files are stored

10. significantly increase the number of particles per cell in the input file. Also change the name of the folder where the output is stored, `SaveDirName` and `RestartDirName` in the input file, to prevent the previous output files from being rewritten. Then run another simulation. How does the execution time change? You may proceed to Part 2 while the new simulation runs

2 Part 2: how to visualise iPic3D data

We will visualise the results of the iPic3D simulations by converting the output `*.hdf` files into the `*.vtk` format [1] and then opening them with Paraview [2]. To do so, cd to the folder `PICFOLDER/visu` and then follow the instructions here:

1. inspect the `hdf2vtk.sh`. script file. It will generate an executable that we will launch as a batch job for the `*.hdf` to `*.vtk` file conversion. Check that the `Input_HDF_folder` corresponds to the folder where you stored your output files.

2. launch the `hdf2vtk.sh` script to produce the `hdf2vtk` executable by typing

   ```
   ./hdf2vtk.sh
   ```

   You may need to change the script permissions as in

   ```
   chmod a+x hdf2vtk.sh
   ```

3. inspect the batch file `PICFOLDER/visu/scriptVisu` and read the annotations to understand how to extract different quantities at different cycles.

4. type

   ```
   msub scriptVisu
   ```

   to launch the batch job. The `*.vtk` files will be saved in the `PICFOLDER/dataGEM_Visu` folder as specified in the `hdf2vtk.sh` script file.

5. download the `*.vtk` files in your local machine by typing

   ```
   rsync -urv user@host:source destination
   ```

   OR
   connect to Paraview in client/ server mode following 4.1. This option is preferred for the visualization of large files, as in Part 3.

6. use Paraview to open the `*.vtk` files.

You have just visualised your first iPic3D simulation data. However, the simulation you just run is too short, too small and not resolved enough for in depth analysis. Let's use now a larger, longer and better resolved simulation to understand something more about collisionless magnetic reconnection.

3 Part 3: Analysis of a magnetic reconnection simulation

We will analyse now a 2D3V (only the xy plane is simulated, differentials in z are suppressed) simulation of collisionless magnetic reconnection without guide field. You may find helpful to check the HOW TO... section for help with the different tasks.

- copy the `BG0_CSAM.tar` archive in your home folder and untar it by typing

  ```
  tar -cvf BG0_CSAM.tar
  ```

- inspect the input file of the simulation, `inputfile.BG0`. All lengths are normalised with respect to the ion skin depth d_i and all times/ frequencies with respect to the ion plasma frequency ω_{pi}

- open the `*.vtk` files in Paraview. Follow 4.1 to use Paraview in client/ server mode.
- use the "Annotate Time Filter" filter to time stamp the plot in terms of ion plasma frequency ω_{pi} or ion cyclotron frequency Ω_{pi} (the second option is preferred).
** Hints: 1 cycle corresponds to $dt\ \omega_{pi}$; in the system of normalisation used, $\Omega_{ci}/\omega_{pi} = B_{0x}$, with B_{0x} from inputfile.
- inspect the characteristic quadrupolar signature of antiparallel magnetic reconnection in the out of plane component of the magnetic field and draw field lines (check section 4.2 for help)
- draw flow lines for electrons and ions across the reconnection region following 4.3. How do electrons and ions move across the reconnection region? Which are the entry points? Do electron and ions interact with the separatrices in comparable ways? Can you guess why?
- now compare directly the fluid velocities of electrons and ions in the three directions. What do you notice?
- the $\mathbf{J} \cdot \mathbf{E}$ metric measures where the electric field does work on particles and help identifying the areas of relevance under the point of view of energy exchange [6]. Use the Python Calculator filter to compute this metric separately for electrons and ions. Which areas are highlighted here? What are the difference between the species? Are they consistent with the previous point?
** Hint: pay attention to the color scale
- determine the width of the electron and ion diffusion regions taking vertical cuts of $\mathbf{J}_e \cdot \mathbf{E}$ and $\mathbf{J}_i \cdot \mathbf{E}$ at $x/d_i = 100$ with the Plot Over Line filter. Which values do you obtain? Are they consistent with the expected scaling? Can you trust the width of the electron diffusion region you obtain?

4 HOW TO...

4.1 Connect to Paraview in client/ server mode

To visualise large files, it is convenient to run Paraview on the cluster and visualise results in client/ server mode [3]. Follow instructions here:

- connect to Judge
- start an interactive job:

```
qsub -I -l nodes=1:ppn=8 -l vmem=48G
```

- load paraview

```
module load paraview
```

- start the Paraview server on Judge

```
mpiexec -n 8 pvserver
```

Note down the number of the node accepting the connection

```
Waiting for client...
Connection URL: cs://judge033:11111
Accepting connection(s):  judge033:11111
```

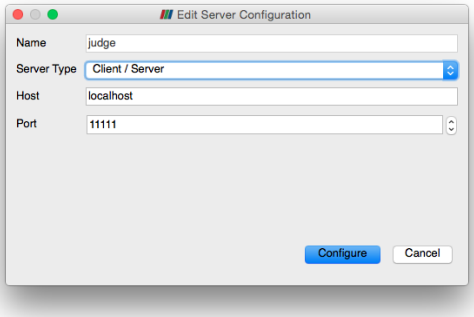

Figure 2: Configuration of the Paraview client

- on your local machine, start the ssh-tunnel to the compute nodes on judge

```
ssh -L 11111:judge033:11111 userName@judge
```

- open Paraview and configure the client: `File` → `Connect` → `AddServer`, see Figure 2
- connect and start working

4.2 Visualise magnetic field lines

- you want magnetic field lines in a 2D plane, so you need to feed the Stream Tracer filter with the magnetic field in the xy plane; use the Calculator filter to generate

```
Bplane= B_X*iHat + B_Y*jHat
```

- draw field lines with the Stream Tracer filter and Bplane as vector input; use a Line Source appropriately placed in the xy plane
- finalise your results using the Tube filter

4.3 Visualise flow lines for the particle species

- use the Append Attributes filter to be able to operate together on the currents J and on the densities rho; to select multiple variables at the same time in the Pipeline Browser keep ctrl pressed + double right click
- use the Calculator filter to calculate the fluid velocity for the species s as

```
Us= Js/rhos
```

  In the `*.vtk` files, electrons are labelled as e (in the vtk for the currents J) or 0 (in the vtk for the density rho), ions as i or 1;
- see 4.2 for the stream lines, using electron and ion fluid velocities as inputs

References

[1] https://en.wikipedia.org/wiki/vtk.

[2] http://www.paraview.org.

[3] `https://en.wikipedia.org/wiki/Client--server_model`.

[4] `https://www.hdfgroup.org/HDF5/`.

[5] `http://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/JUDGE/Userinfo/Quick_Introduction.html?nn=1041384`.

[6] M.V. Goldman, D.L. Newman, and G. Lapenta. What can we learn about magnetotail reconnection from 2d pic harris-sheet simulations? *Space Science Reviews*, pages 1–38, 2015.

[7] E. Harris. On a plasma sheath separating regions of oppositely directed magnetic field. *Il Nuovo Cimento (1955-1965)*, 23:115–121, 1962.

[8] Giovanni Lapenta, JU Brackbill, and Paolo Ricci. Kinetic approach to microscopic-macroscopic coupling in space and laboratory plasmas. *Physics of plasmas*, 13:055904, 2006.

[9] Stefano Markidis, Giovanni Lapenta, et al. Multi-scale simulations of plasma with ipic3d. *Mathematics and Computers in Simulation*, 80(7):1509–1519, 2010.