EURO

Plasma physics simulations with PIC codes

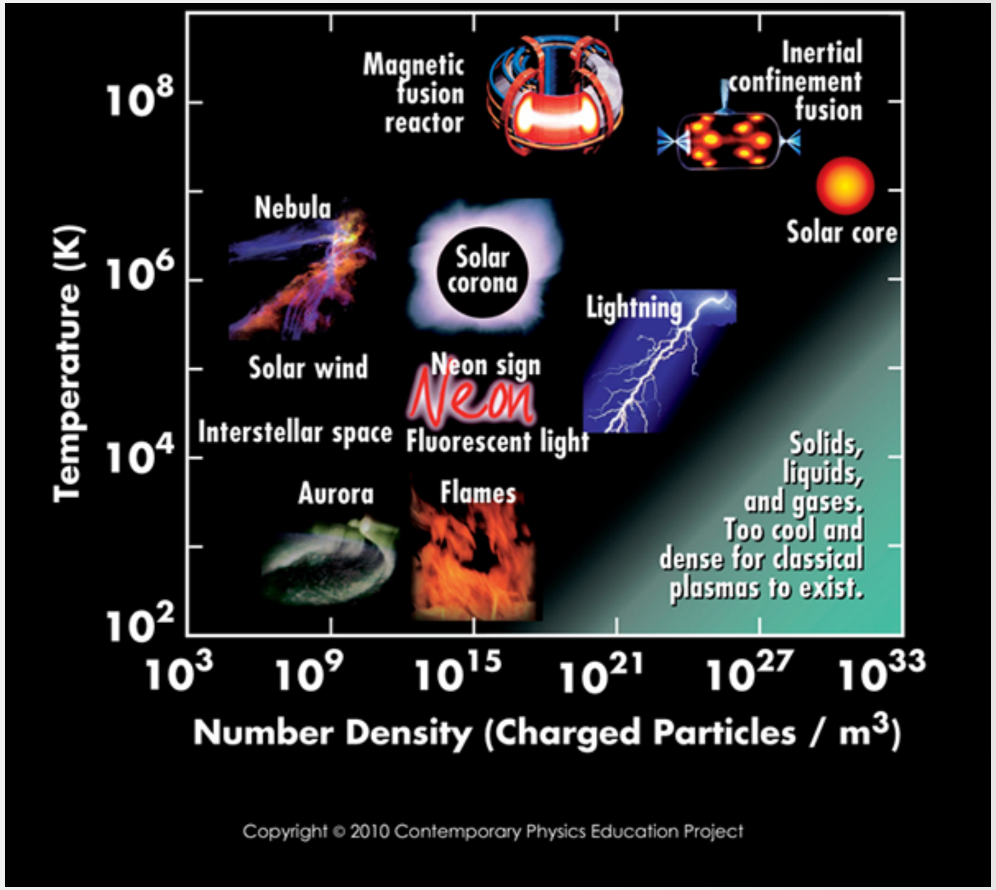
Ivona Vasileska

LECAD laboratory, Faculty of mechanical engineering, University of Ljubljana

What is plasma?

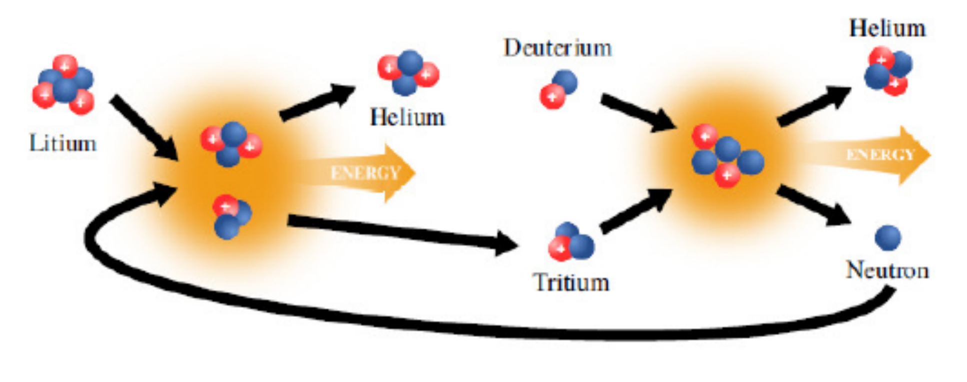
EURO

- A plasma is a quasineutral gas containing charged particles. In the most common case, the plasma consists of electrons and positively charged ions.
- The fundamental characteristics and properties of plasma are:
  - The quasineutrality (*Ne = Ni*).
  - The volume where the plasma quasineutrality is violated is characterized by the Debye length $\left(\lambda_D = \sqrt{\frac{\epsilon_0 k T_e}{n_0 e^2}}\right)$.
  - Plasma temperature $\left(T \gg \frac{e^2 n^{1/3}}{\epsilon_0 k}\right)$.
  - Plasma frequency $\left(\omega_{pe} = \sqrt{\frac{n e^2}{\epsilon_0 m_e}}\right)$.
  - Plasma density $\left(n \gg {}^1/_{\lambda_D}\right)$.



Nuclear fusion

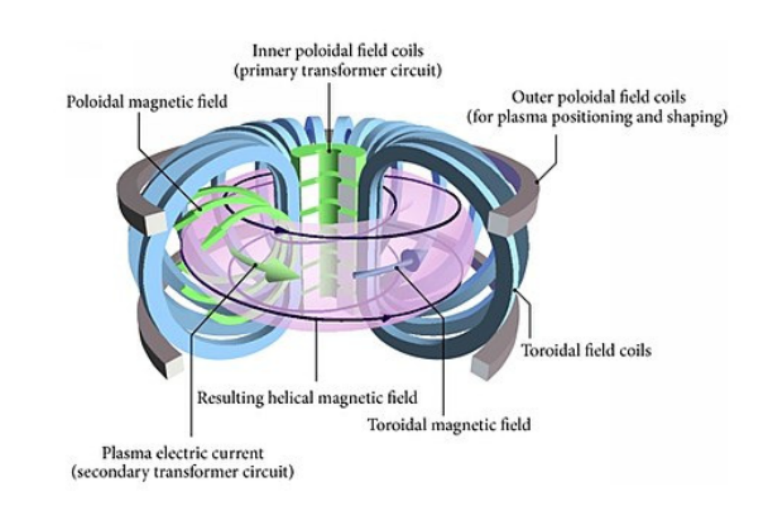
- Nuclear fusion - process of combining light atomic nucleus into heavier ones due to the kinetic energy of their thermal motion.
- The thermonuclear fusion energy occurs in the form of:
  - thermonuclear weapons (hydrogen bomb), where an uncontrolled nuclear reaction with a explosive nature takes place;
  - fusion reactors, where a controlled nuclear reaction takes place.

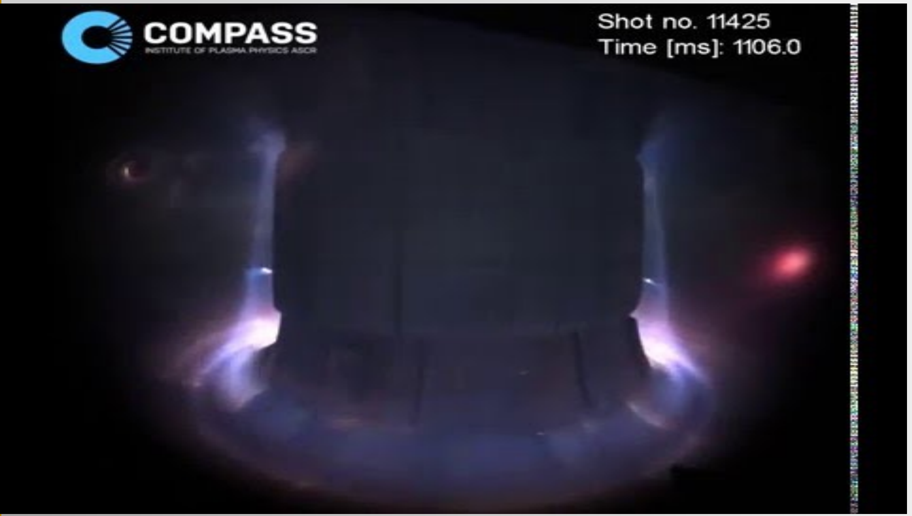


6Li + n →T + 4.8 MeV

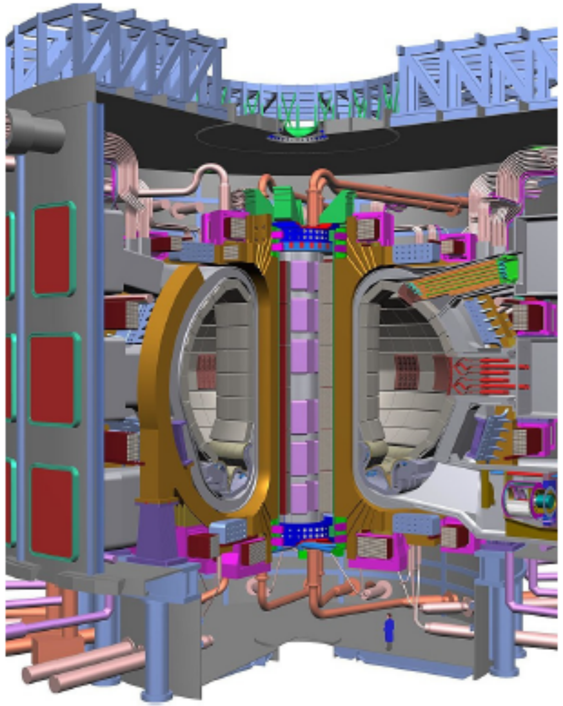
D+T→ 4He + n + 17.6 MeV

Fusion reactors - tokamak





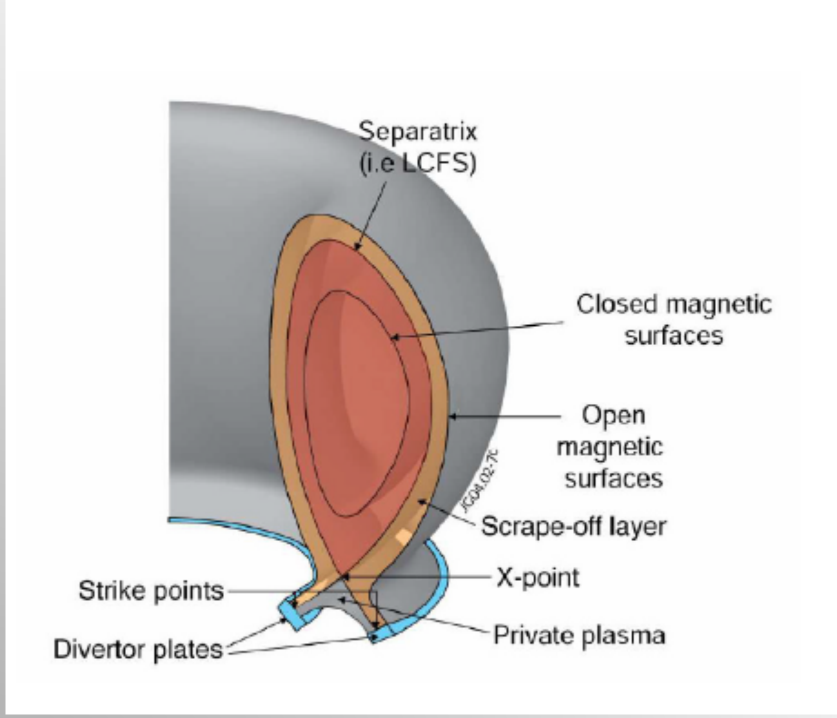
- ITER parameters:
  - power 500 MW,
  - ratio of the fusion energy synthesis to heating energy not less than 10,
  - burning plasma time 400 s,
  - large and small radii torus 6.2 m and 2 m, respectively,
  - plasma volume 840 m3,
  - plasma current 15 MA. As a fuel it uses a mixture of deuterium with tritium.



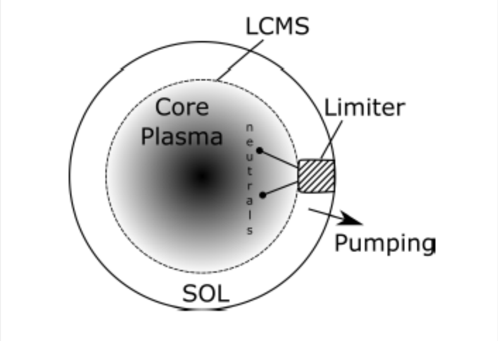
ITER

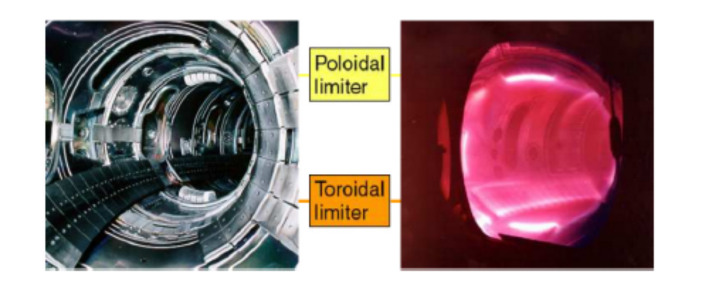
Plasma Scrape-Off Layer (SOL)

Field lines in tokamak

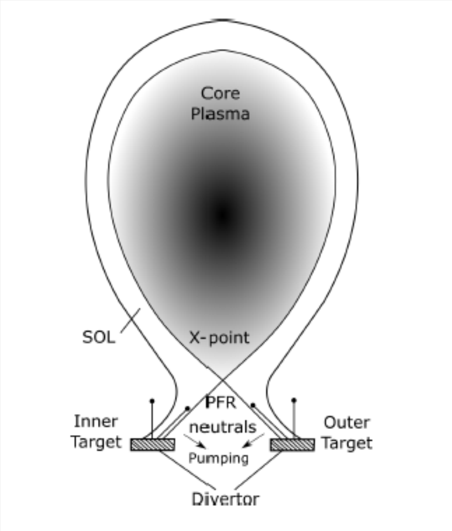


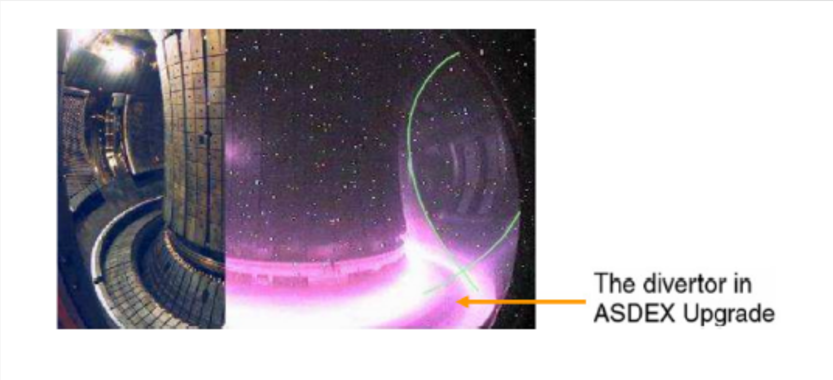
Limiter configuration



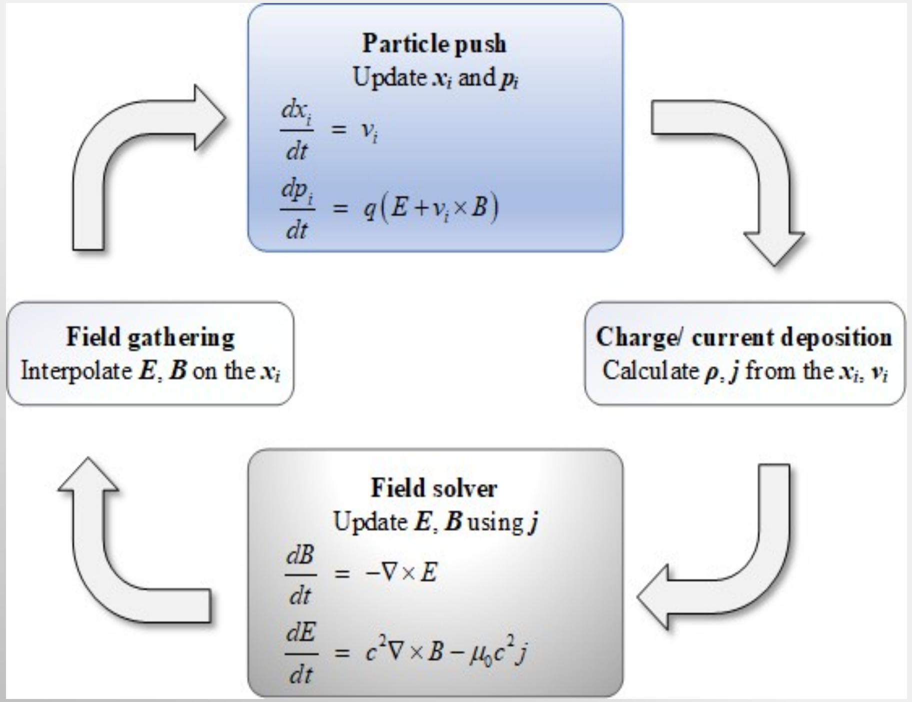


Divertor configuration





Kinetic modelling plasma tool - Particle-In-Cell (PIC) codes

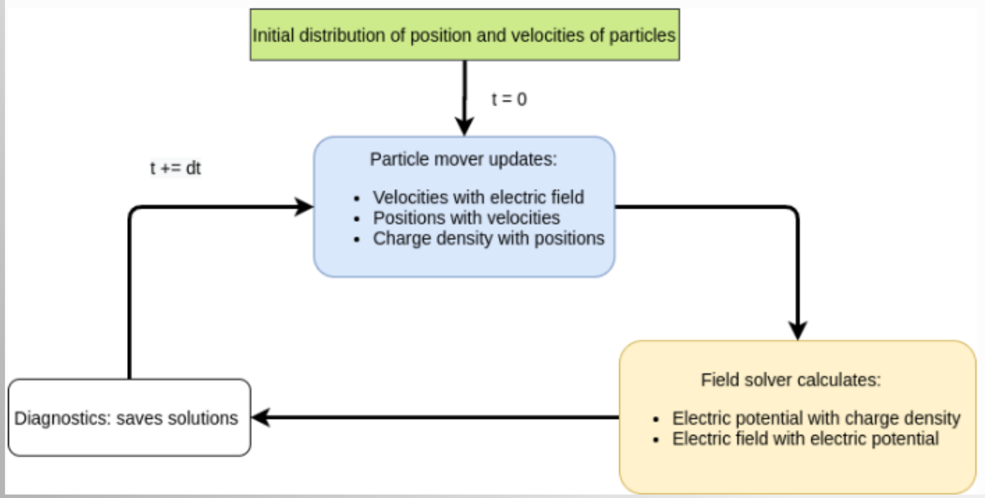


Main PIC code algorithm

- Features:
  - Calculates super-particles motion. Each super-particle represents many particles in the plasma (more particles → more accuracy). This feature is called particle mover.
  - Takes into account fields generated by the particles and external circuits, this feature is called field solver. Two types of description:
    - Electrostatic codes: Solves the Poisson's equation for the electric field.
    - Electromagnetic codes: Solves a set of Maxwell's equations.

Simple PIC code (SimPIC)

- Code documentation: https://lecad-peg.bitbucket.io/simpic/simpic.html?highligh
- SIMPIC workflow:



- Input variables in SIMPIC:

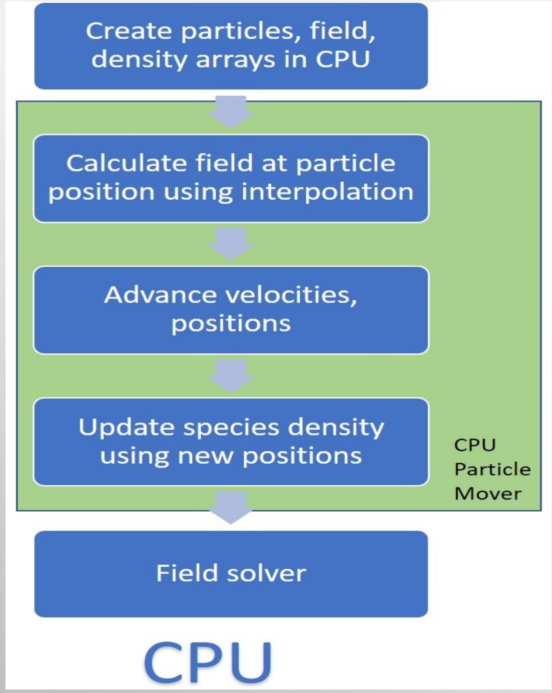
NT=200 *# of time steps*
PPC=100 *# of particles per cell*
CPP=64 *# of cell per process*
DTFACTOR=0.001 *#defines a fraction of dt vs. time steps from plasma frequency; #must be positive*
NPROC=4 *# of processors*
LHSV=25000 *#applied voltage on left-hand side; RHS is grounded;*

- Output results:

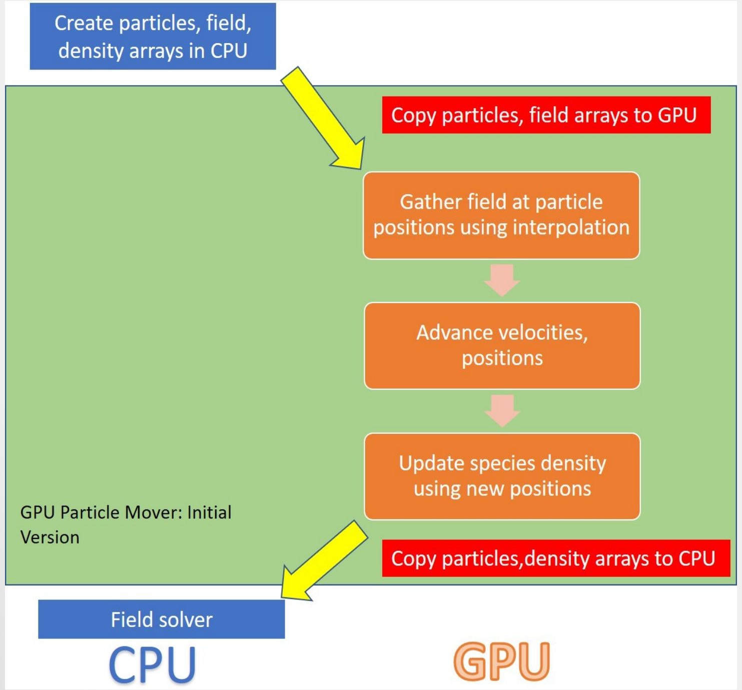
den *#density*
E *# electric field*
phi *# potential*
vxx *# phase space*
nt *# time evolution of number of particles per processor*

File format for each processor is:
time, relative_position, value

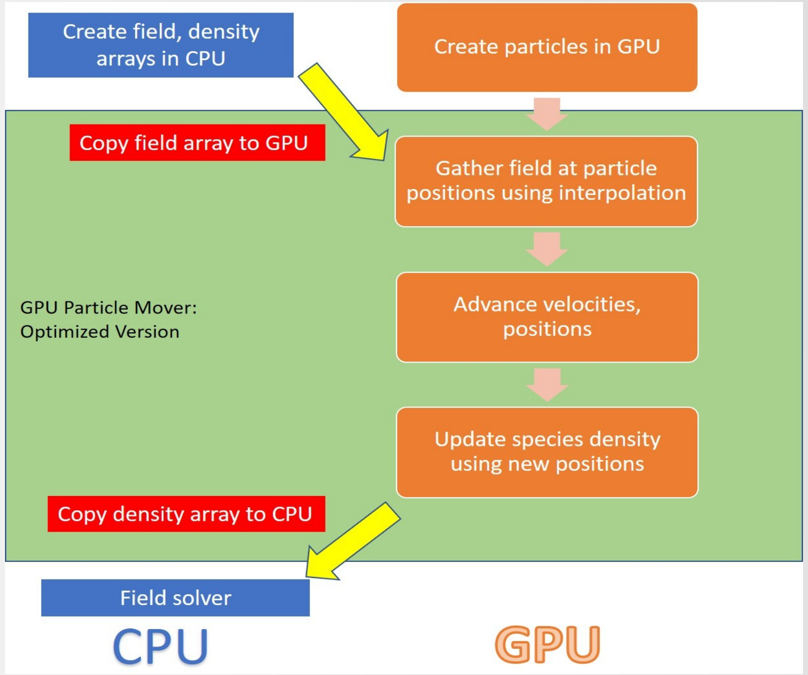
Comparing CPU vs GPU Particle Mover



CPU SIMPIC of the particle mover

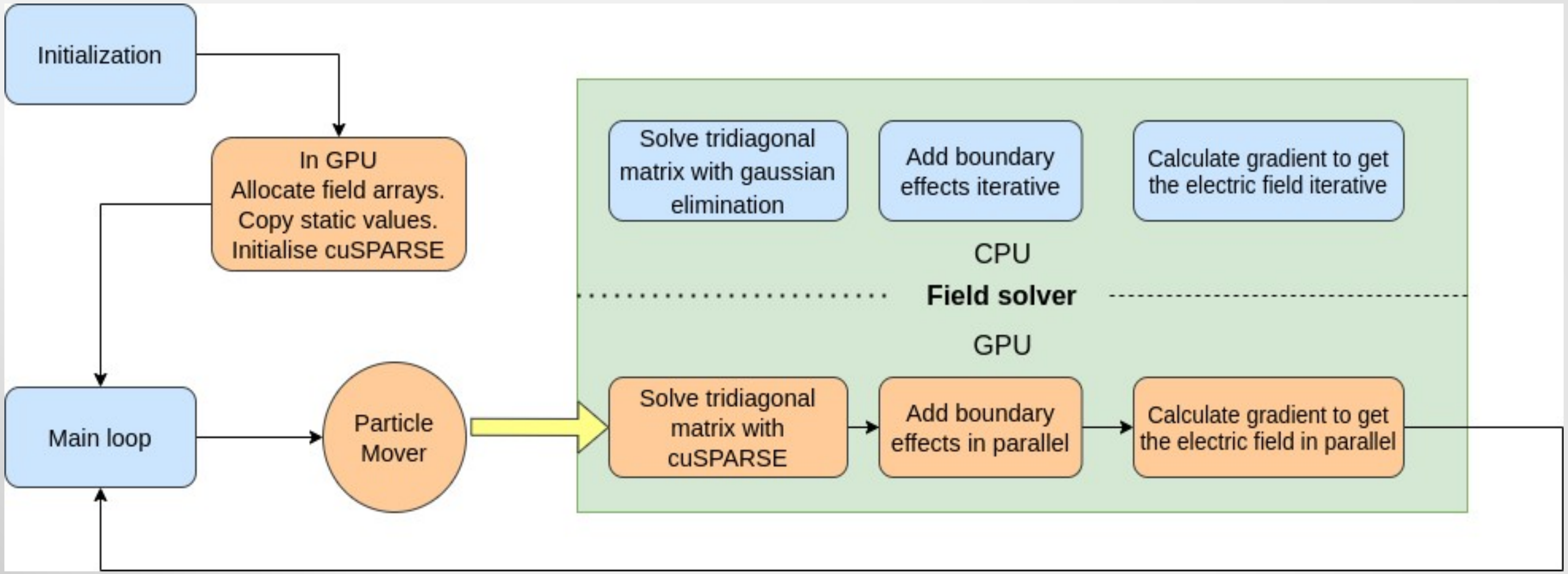


GPU SIMPIC of the particle mover: Initial Version



GPU SIMPIC of the particle mover: Optimized Version

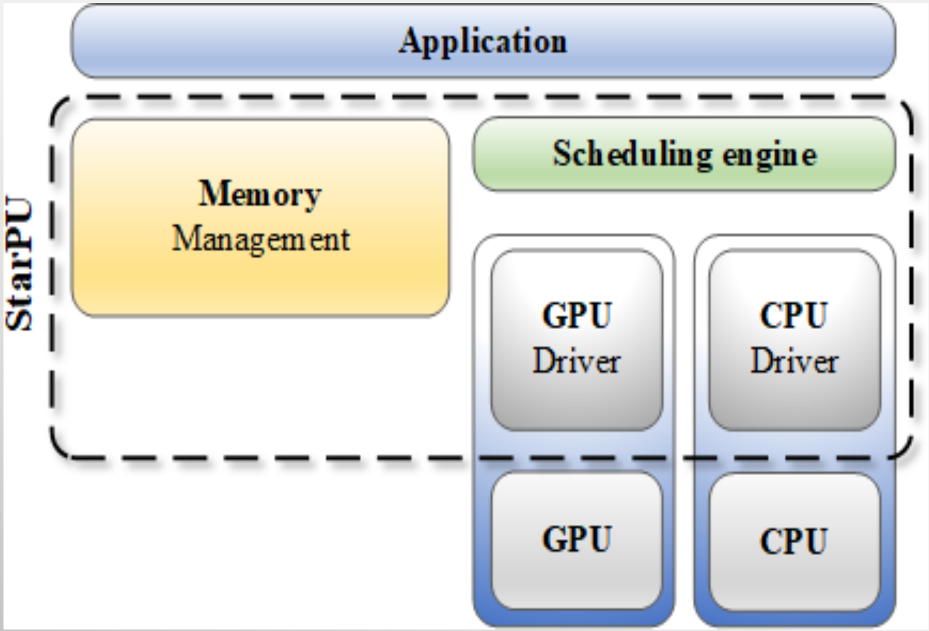
GPU field solver



GPU of the filed solver

Heterogeneous Computing. Why StarPU?

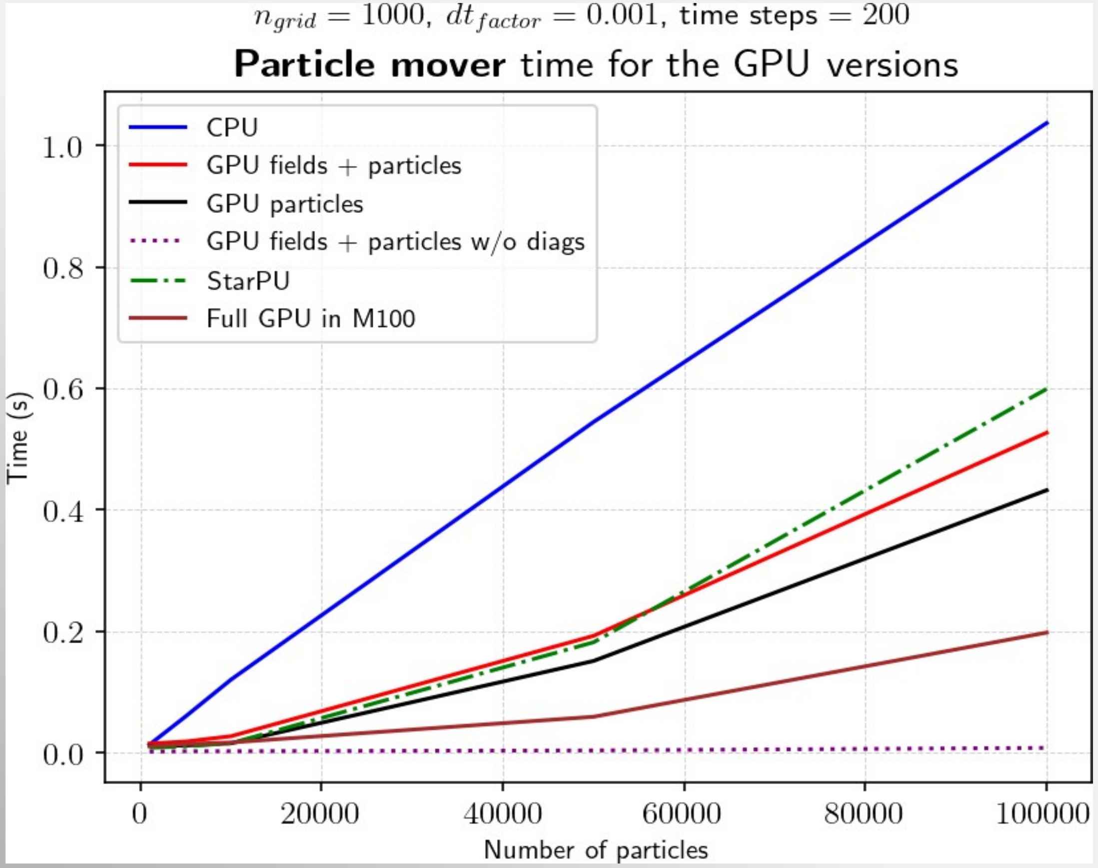
EURO

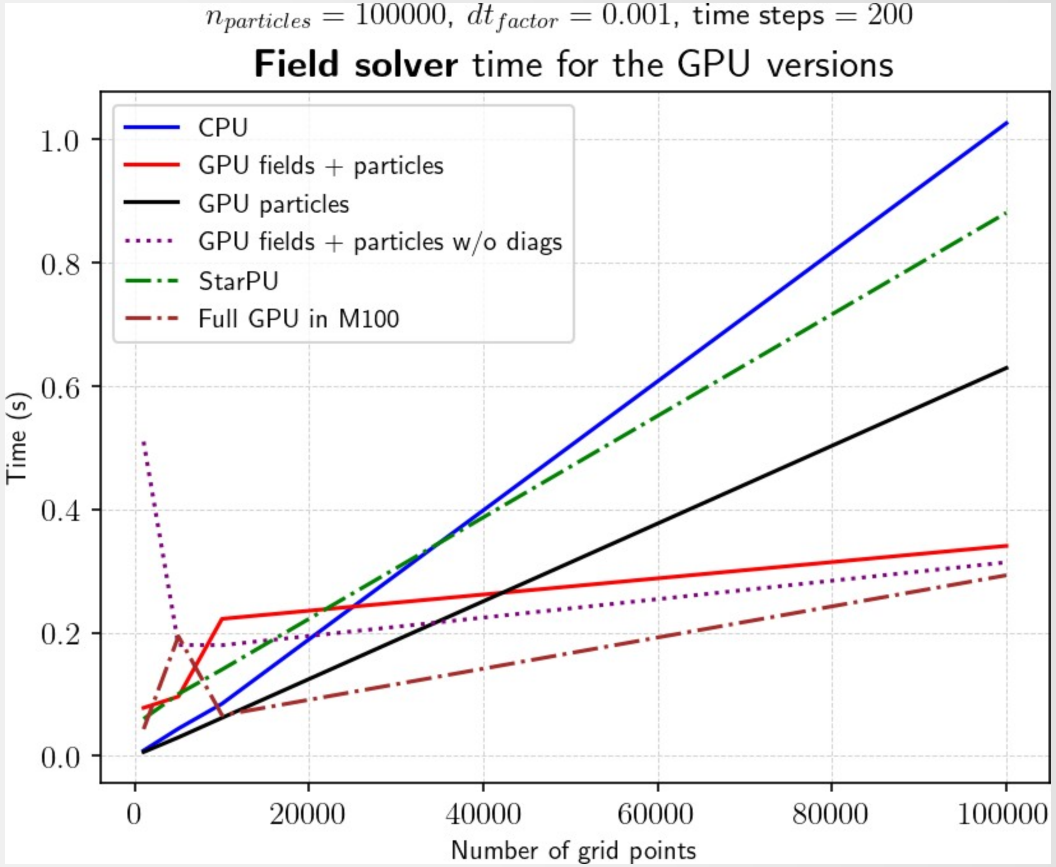


Function scheme of the StarPU run system

- Heterogeneous Computing uses all available processing units (CPUs,GPUs) for computation
- StarPU enables portability of code across various architectures and ensures interaction between different PUs
- StarPU implements task based parallelism. Each task could be a CPU function, CUDA kernel or OpenCL kernel.

Benchmarks for fully GPU version





Building SIMPIC on VIZ-HPC

EURO

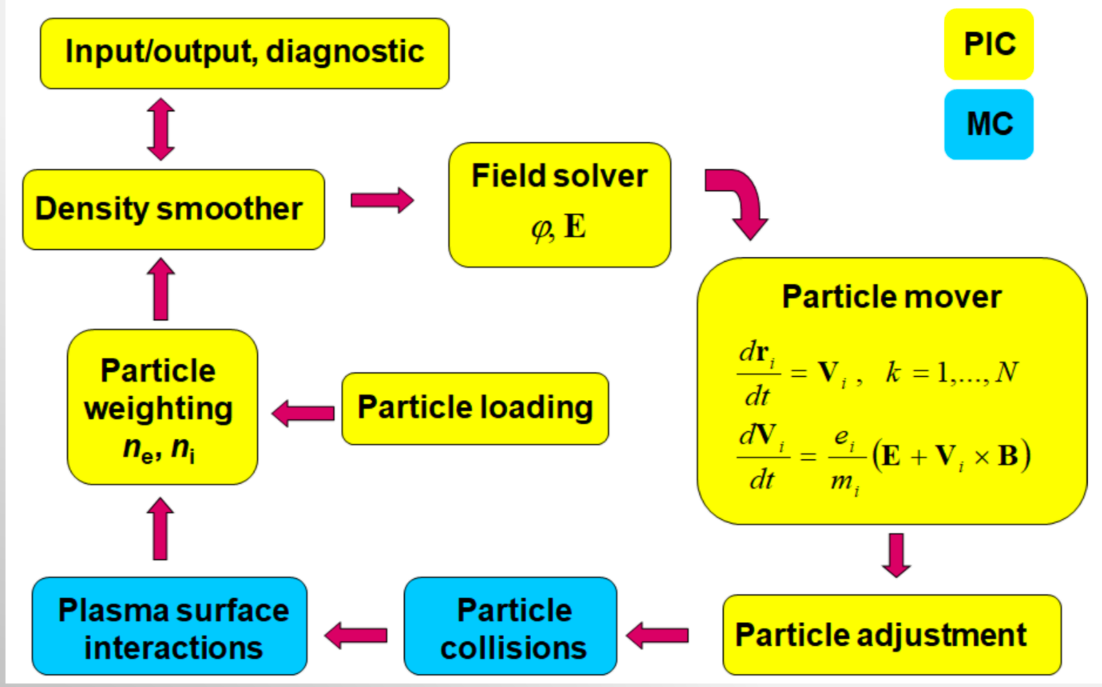
```
git clone https://bitbucket.org/lecad-peg/simpic.git
module load Python/3.7.4-GCCcore-8.3.0 #HOME DIRECTORY
module load OpenMPI/3.1.4-gcccuda-2019b #HOME DIRECTORY
python3 -m venv simpyenv #HOME DIRECTORY
source simpyenv/bin/activate #HOME DIRECTORY
python3 -m pip install --upgrade pip sphinx_rtd_theme #HOME DIRECTORY
pip install numpy scipy pandas matplotlib #HOME DIRECTORY
pip list #HOME DIRECTORY
cd simpic
cd GPU_mover_fields #GPU version
make
./runsimpic.sh 2> run.log

For CPU versions first is needed to install STARPU
wget https://files.inria.fr/starpu/starpu-1.3.4/starpu-1.3.4.tar.gz
tar xvf starpu-1.3.4.tar.gz
cd starpu-1.3.4
mkdir build
cd build
../configure --prefix=$HOME/starpu
make
make install
export PKG_CONFIG_PATH=$PKG_CONFIG_PATH:$HOME/starpu/lib/pkgconfig #IN SIMPIC
export LD_LIBRARY_PATH=$HOME/starpu/lib:$LD_LIBRARY_PATH #IN SIMPIC
export PATH=$PATH:$HOME/starpu/bin #IN SIMPIC
```

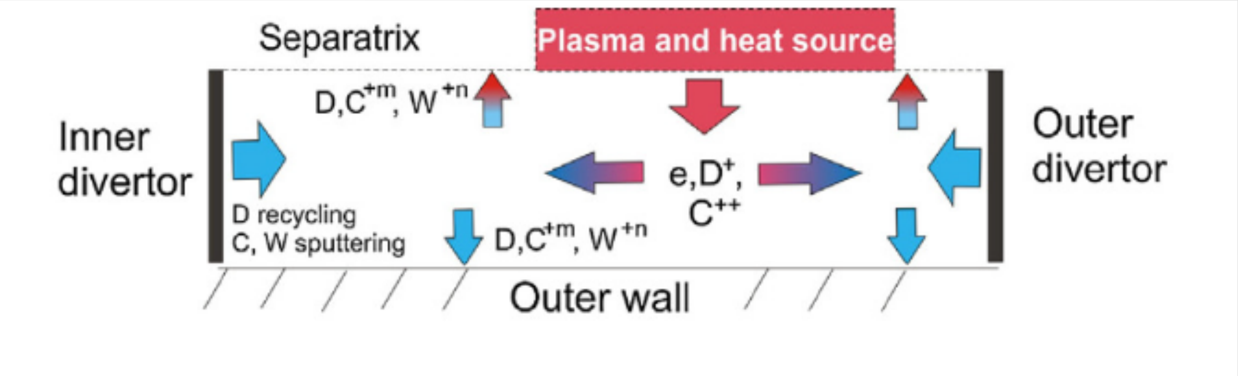
SIMPIC tasks:

- Run the code changing the input parameters (increasing and decreasing the number of cells, number of particles and applied voltage);
- Plot the plasma parameters with different input parameters;
- Do the same with GPU version of the code;
- With fixed input parameters run both codes CPU and GPU version and compare running time, plot graphs

OOPD1 and BIT1, PIC code



PIC/MC algorithm codes



Simulation geometry of tokamak in BIT1

Building OOPD1,PIC on VIZ-HPC

```
cp -r /home/ivasileska/test_oopd1 .
module load tk/8.6.8/intel-18.0.2-kzhkvu6
module load X11/20190717-GCCcore-8.3.0
module load OpenMPI/3.1.4-GCC-8.3.0
cd test_oopd1
cd xgrafix
nano run_conf.sh
IN run_conf.sh #CHANGE PLEASE PREFIX prefix=$HOME/opt/xgrafix
#SAVE
./run_conf.sh
make
make install
cd ..
cd oopd1
make
./pd1 -h

./pd1 -i inp/test.inp -s 1000 #SIMPLE CASE
./pd1 -i inp/argon_10mTorr_27MHz.inp -s 1000 #ARGON CASE
```

Building BIT1,PIC on VIZ-HPC

```
module load OpenMPI/3.1.3-GCC-8.2.0-2.31.1
cp /home/ivasileska/BIT1.tar.gz . #home directory
tar xzfv BIT1.tar.gz . #home directory
cd PTC
cd BIT1
tar xvzf bit1_n_18.tar.gz
cd BIT1_n_18
make
./BIT1
```

OOPD1 and BIT1 tasks:

- OOPD1 tasks:
  - Run first simple test.inp and than argon_10mTorr_27MHz.inp, compare both parameters.
  - Changing input parameters compare output plasma parameters;
- BIT1 tasks:
  - Input file and output results,
  - Tokamak simulations JET and ITER.
  - Compare all PIC results

THANK YOU FOR YOUR ATTENTION

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 951732. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, United Kingdom, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Switzerland, Turkey, Republic of North Macedonia, Iceland, Montenegro