FLARE & MOOSE Installation and Usage Guide

Shahinul Islam

25th September 2025

Note: FLARE requires MOOSE, so MOOSE must be installed first.

1 Repositories and Documentation

- Clone MOOSE:

```
git clone https://gitlab.com/hfrerichs/flare
cd moose
```

- MOOSE documentation: `https://hfrerichs.gitlab.io/moose/intro.html#installation`
- FLARE documentation: `https://hfrerichs.gitlab.io/flare/getting_started/installation.html`

2 Install Prerequisites

```
sudo apt update
sudo apt install pkgconf libopenmpi-dev liblapack-dev \
                 libnetcdff-dev python3-dev
```

3 Check pkg-config, also check readme

```
pkg-config --version
pkg-config --cflags hdf5
pkg-config --libs hdf5
pkg-config --cflags netcdf
pkg-config --libs netcdf

# If needed:
export PKG_CONFIG_PATH=/usr/lib/x86_64-linux-gnu/pkgconfig:
   $PKG_CONFIG_PATH
```

4 Build MOOSE

```
./configure
make
make install
```

5 Verify MOOSE Installation

```
ls /home/shahinul/.local/lib/pkgconfig/moose.pc

# If found, extend path
export PKG_CONFIG_PATH=$HOME/.local/lib/pkgconfig:$PKG_CONFIG_PATH

# Check linking
pkg-config --libs moose
```

Now build Flare

6 Clone FLARE Repository

```
git clone https://gitlab.com/hfrerichs/flare
cd FLARE
./configure
make
make install
```

7 Verify Python Modules

```
ls /home/shahinul/.local/lib/python
# Should show: flare  moose
# # Also, need to export paths every session after logging out and
    starting a new terminal session

export PYTHONPATH=/home/shahinul/.local/lib/python:$PYTHONPATH
export LD_LIBRARY_PATH=/home/shahinul/.local/lib:$LD_LIBRARY_PATH
```

8 Generating Mesh for EMC3-EIRENE

- Modify paths and set boundary files, and other '.bfield' files
- **Key:** All input files must be stored inside the `./boundary` folder, including the `.boundary` file itself, e.g., `mnt\C:\W7X\20171207_21-20250921T133000Z-1-001\20171207_21\.boundary`
- For details about the `./boundary` file and its structure, see the official FLARE repository: `https://gitlab.com/hfrerichs/flare/-/blob/master/examples/database/W7-X/.boundary/default/.boundary?ref_type=heads`.
- **Example workflow to generate a W7X grid:**

```
#preparing the model: if you change amplitude of the currents
#update the magnetic axis after changing currents:
#flare equi3d_autoconf -R0 5.2 -Z0 0 -Phi0 -36 -nsym 5

# Run the program
flare run -c poincare_map.ctr
flare poincare_plot poincare_maps_2.dat
flare mgen -t inner_boundary
```

```
flare poincare_plot poincare_maps_2.dat -mplot 1st_inncer_boundary0
   .dat,r 2nd_inner_boundary0.dat,b
flare poincare_plot poincare_maps_2.dat -fit 1 -nctrl 32 -o gc4.dat
    # 32 is the number of control points for fitting at the outer
   surface, increasing this number provides better quality
flare poincare_plot poincare_maps_2.dat -fit 3 -nctrl 32 -o gc3.dat
flare poincare_plot poincare_maps_2.dat -fit 6 -nctrl 32 -o gc2.dat
flare poincare_plot poincare_maps_2.dat -fit 7 -nctrl 32 -o gc1.dat
flare mgen -t base_mesh
mview base0.dat -mplot gc*
flare mgen -t flux_tubes
flare mgen -t n0_domain
flare mgen -t plates
mmesh check
mmesh rzslice 18 -P domain -B
```

- **Everything about** `mmesh.ctr`**:** This file defines the mesh and geometry parameters for both plasma and neutral simulations. It specifies symmetry, grid resolution, boundary definitions, and additional mesh regions such as the core and vacuum. Below is a structured breakdown of its main sections and parameters:

```
[mmesh_parameters]

# --- 1. Symmetry and domain definition ---
symmetry:           5      # Number of symmetric toroidal sectors (
   blocks).
blocks:             1      # Subdivision of each block. 1 = no
   subdivision.
                           # Larger values improve resolution by
                              subdividing each block.

updown_symmetry:    1      # 1 = up-down symmetry (only one half
   simulated, (e.g., lower half). Selection of which symmetry is
   used should follow what's defined in flare_slides_manual (slide
   9).
                           # Domain: 0 < phi < 360/symmetry/2.
                           # 0 =  It is -360/symmetry/2 < phi < 360/
                              symmetry/2 in the case of
                              updown_symmetry==0. This is a
                              limitation of FLARE, and EMC3 does not
                              have this limitation. You can start the
                               simulation domain from any toroidal
                              angle. Domain: -360/symmetry/2 < phi <
                              360/symmetry/2.
                           # (FLARE limitation; EMC3 allows arbitrary
                               phi domain start.)

# Inner flux surface defined by two reference points (r [m], z [m],
    phi [deg]):
p1 = 5.05, 0.0, 27.0    # Innermost plasma boundary (at phi = $27^\
   circ$).
p2 = 5.03, 0.0, 27.0    # Next radial surface (used to construct
   radial surfaces).

# --- 2. Base mesh ---
# Mesh divided into radial regions with user-defined cell counts.
# Example with four radial zones:
```

```
nr(0) =  36
nr(1) =  72
nr(2) =  36
nr(3) =  18

np    = 512   # Number of poloidal cells. Large np needed in EMC3
   to ensure
              # accurate mapping of the first and last toroidal
                 planes.
              # Insufficient np may cause spatial gaps, particle
                 loss, or errors.

guiding_contour = "gc1.dat,gc2.dat,gc3.dat,gc4.dat"
qmesh_generator = "INTERPOLATE"

# --- 3. Flux tubes ---
nt    = 36    # Number of toroidal surfaces (slices) around the
   torus.
              # Each slice uses the same poloidal/radial cell info
                 as base mesh.

# --- 4. Core and vacuum extensions (for neutrals only) ---
# Plasma ends at inner flux surface (p1), but neutrals may enter
   the core.
nr_core = 8
core_domain:   "FLUX_SURFACES 0.2"   # Radial cells inside core,
   neutrals only.

nr_vac = 1
vacuum_domain: "MODEL_BOUNDARY 1; RZBUFFER 5.0; AUTO_BLOCKS"
# MODEL_BOUNDARY = vessel wall.
# RZBUFFER = radial/vertical extension of the vacuum vessel.

nsside = -1   # Direction of recycled neutrals:
              # -1 = inward (toward plasma), +1 = outward (away
                 from plasma).
              # Note: Meaning may reverse if surface definition
                 order is inverted.

generate_add_sf_n0 = false

# --- 5. Divertor plates and plasma cells ---
plate_generator: "SCAN_RPATHS 4 nint"
# Detects non-plasma cells behind the plate. Normally unchanged,
   but may fail
# in complex cases. Consult manual if plasma region not defined
   correctly.

cell_def  = 2   # Format type of plate.dat (defines plate geometry)
   .
              # Different formats exist but do not affect physics
                 /numerics.
cell_param = 6

# Note: Outer plasma surface is based on a Poincar\'e plot.
# This surface may be irregular or broken always check and adjust
   the boundary
# to ensure mesh continuity and correct geometry.
```

- Poincare map: only for plotting

```
[poincare_map_Rlinspace]
R_start: 5.64
R_end: 5.9
NR: 27 # number of radial points
nsym: 5 # number of symmters
nsections: 4 # number of section, nr(0) to nr(3) in this example
    above
bounded: False
```

- input par: This file defines the initial simulation conditions. In this case, the setup is for a pure hydrogen (H) plasma, while impurity setting is given in the next section. Initial plasma conditions (e.g., density, temperature profiles) must be defined, often using the eelab.After the first run, you can re-use the generated output files (e.g., files 30, 31, 33, 40, 42, 43, 46, etc.) to restart or continue the simulation — now including both plasma and neutral models. Continue iterating until the solution converges — typically when residuals are on the order of 1% or lower.

```
*-------------------------------------------------------------------------

*** 4. ion species
                                                          ---
*-------------------------------------------------------------------------

1
H         1       1.0
*-------------------------------------------------------------------------

*** 5. transport parameters
                                                 ---
*-------------------------------------------------------------------------

* anomalous cross-field particle diffusion [cm**2 s**(-1)] for all
   ion species
0.500000e+04
* anomalous cross-field energy diffusion [cm**2 s**(-1)] for el.
   and ions
0.7000e+04    0.70000e+04
*-------------------------------------------------------------------------

*** 6. boundary conditions
                                                   ---
*-------------------------------------------------------------------------

*** 6.1 particle transport
* H ions
1            1.9e+13
2
   1   1   1  0.0
   1   2  -2  1.0
*** 6.2 energy transport (el. and ions)
2
   1   1   1  144500.0  144500.0
   1   2  -2  1.0  1.0
*** 6.3 momentum transport
2
   1   1   1  0.0
```

```
   1   2  -2  1.0
*--------------------------------------------------------------------------

*** 7. initial conditions
                                                    ---
*--------------------------------------------------------------------------

 30      40.0      40.0
 31    1.000000e+13
 33
*--------------------------------------------------------------------------

*** 8. volume sources
                                                        ---
*--------------------------------------------------------------------------

 40     41
 42     0
 43     0
 46     47
```

- input par: for both h and impurity spices

```
*--------------------------------------------------------------------------

*** 4. ion species
                                                          ---
*--------------------------------------------------------------------------

2
H         1      1.0
O         8     16.0
*--------------------------------------------------------------------------

*** 5. transport parameters
                                                  ---
*--------------------------------------------------------------------------

* anomalous cross-field particle diffusion [cm**2 s**(-1)] for all
   ion species
0.500000e+04
0.500000e+04
* anomalous cross-field energy diffusion [cm**2 s**(-1)] for el.
   and ions
0.7000e+04    0.70000e+04
*--------------------------------------------------------------------------

*** 6. boundary conditions
                                                    ---
*--------------------------------------------------------------------------

*** 6.1 particle transport
* H ions
1           1.9e+13
2
   1   1   1  0.0
   1   2  -2  1.0
* O ions
```

```
0             0.00
2
   1   1   1  0.0
   1   2  -2  1.0
*** 6.2 energy transport (el. and ions) # this is the input power
   for a simulation box, so, total power needs to be multiplited
   with the number of toroidal symmetry defined above (here is 5).
2
   1   1   1  144500.0  144500.0
   1   2  -2  1.0  1.0
*** 6.3 momentum transport
2
   1   1   1  0.0
   1   2  -2  1.0
*-------------------------------------------------------------------------

*** 7. initial conditions
                                                  ---
*-------------------------------------------------------------------------

 30       40.0       40.0
 31     1.000000e+13
 33
*-------------------------------------------------------------------------

*** 8. volume sources
                                                      ---
*-------------------------------------------------------------------------

 40      41
 42      0
 43     0
 46      47
```

- ADD SF NO : All the surface data and will be used in surface number 4 in section 3A (input.eir)

```
17  # Number of external surface files to be read.
    # These files define surfaces such as the target and baffle,
       typically referenced by path below.

0  -4   1
# 0  Use external files for source definition (not number of
   triangles)
# -4  Refers to Surface No. 4 from input.eir (e.g., target + baffle
   ) as given below

* SURFACE No. 4 Target + Baffle
     1     1     4
     1     2     0     0     0     3
     1
      1206.0     -0.0533
  1.0000E+00  1.0000E+00  0.0000E+00  1.0000E+00  0.5000E+00
     0.1000E+00

1206 = Material (e.g., carbon)
-0.0533 = Energy
```

```
* SURFACE No. 5 Back shielding plates

# Final switch (1)  Enables neutral particle generation.
# If set to 0, no neutrals will be emitted from this surface.


           17
   0  -4   1
/u/fene/run_flare/targets/bafhor1.b # b is the baffle and
   0  -4   1
/u/fene/run_flare/targets/bafhor1.t # t is the target plate
   0  -4   1
/u/fene/run_flare/targets/bafhor2.b
0  -4   1
/u/fene/run_flare/targets/bafhormid.b
0  -4   1
/u/fene/run_flare/targets/bafhormid.t
0  -4   1
/u/fene/run_flare/targets/baf_m02.t
0  -4   1
/u/fene/run_flare/targets/baf_m03.t
0  -4   1
/u/fene/run_flare/targets/bafmod1.t
0  -4   1
/u/fene/run_flare/targets/bafver1.b
0  -4   1
/u/fene/run_flare/targets/bafver1.t
0  -4   1
/u/fene/run_flare/targets/bafver2.b
0  -4   1
/u/fene/run_flare/targets/bafver3.b
0  -4   1
/u/fene/run_flare/targets/bafvern8.b
0  -4   1
/u/fene/run_flare/targets/bafvern8.t
0  -4   1
/u/fene/run_flare/targets/divhoran7.b
0  -4   1
/u/fene/run_flare/targets/divhorn9.t
0  -4   1
/u/fene/run_flare/targets/divvern8.t
```

- : pumping surface in input eir

```
* SURFACE No. 6 pumping plate (neutral shielding for div2)
     1     1     6
     1     2     0     0     0     3
     1
      1206.0 (carbon)    -0.0259 (energy and sign indicates
         Maxwellian distribution)
  1.0000E+00  0.9413E+00  0.0000E+00  1.0000E+00  0.5000E+00
     0.1000E+00

    RECYCT= 0.9413, see EIRENE manual for speed, S = A(1 - RECYCT)
       3.638 sqrt(T/m), https://www.eirene.de/old_eirene/eirene.pdf

You can use any surface type for pumping surface if you  change the
    reflection coefficient. You should use the surface type with
```

```
   coefficient<1 in ADD_SF_N0 file. The particle balance is
   somewhat special thing in EMC3. The particle is not  always
   balanced. If pumping flux is large, recycling neutral by surface
    recombination is artificially increase to keep the particle
   balance. Therefore, you can run EMC3 with only  pumping or with
   only puffing.
```

- : gas puffing surface in input eir

```
  * 2. Gas-puffing
FFFFF
10000 6001 1 1 10000
1.00000e+00 0.00000E+00 0 0 0 0 0 0 0 0
FTFFF
1
TFFFF
1
6 4 -6 25 25 1 99 0 0
1.00000E+00 1.04000E+02 6.00000E+00 0.00000E+00 1.00000E+03
0 0 0 0 0 0 0
1.61500E+02 0.00010E+00-1.25000E+02 0.61002E+00 0.00000E+00 0.79238
   E+00
0.02600E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.00000E+03 9.00000E+01 0.00000E+00 0.00000E+00 0.00000E+00
```

- Navigate to FLARE examples:

```
cd /mnt/c/FLARE/flare/examples/python
```

- Run the control file:

```
flare run -c flare.ctr
# Should see: Magnetic field from MGRID file: mgrid_w7x_mod1.nc
# Finished task execution: 35.088 s
```

- Field line tracing:

```
python fieldline_trace.py
```

- Poincaré maps:

```
python Poincare_maps.py
```

- Magnetic field visualization:

```
python magnetic_field.py
```

- First steps script:

```
python first_steps.py
```

- Radial incursion:

```
python radial_incursion.py
```

- Strike points:

```
python strike_point.py
```

- Field line connection:

```
python fieldline_connection.py
```

- Mesh generation:

```
flare run -c mmesh.ctr
```

Additional Instructions

- Follow the instructions in the FLARE documentation to create a W7-X grid.
- Input parameters are in files named `.bfield`, `.boundary`, etc. Modify paths as needed.
- Provide an `mgrid` file (magnetic field generated by each coil), generated using `stellopt xgrid` starting from a coil file (provided by K. Hammond). See `https://princetonuniversity.github.io/STELLOPT/MAKEGRID`.
- Targets (description of solid surfaces) are required, provided by V. Winters (IPP).
- Other input parameters reside in `mmesh.ctr` and `poincare_map.ctr`.
- To check grid validity, use the `mmesh` routine from the EELAB Python package (EMC3-EIRENE numerical laboratory).
- EMC3 free plotting librabry: `https://xemc3.readthedocs.io/en/stable/`
- EIRENE manual: `https://www.eirene.de/old_eirene/eirene.pdf`

Particle Balance and Boundary Conditions

- **Particle Balance:**

  EMC3-EIRENE uses a normalized ionization source of **1 Ampere** for all particle balance calculations. The ionization profile is normalized to this value, and upstream density is specified as a fixed boundary condition (BC). EMC3 scales all related quantities according to the ratio between the normalized and actual source to maintain numerical consistency and particle conservation. Due to gaps in grid spacing and numerical limitations, a *1 A normalized ionization profile* is used as a practical trick to maintain a consistent particle balance. After the simulation, results (e.g., temperature and density profiles) are rescaled. Scaling is performed only for particle related quantities like ionization, density, energy loss due to neutrals but not for temperature. Temperature is solved by independent step of density.

  Recycling flux is treated as the total ionization rate (in Amperes). Recycling is assumed to be unity (fully recycled), although it may appear slightly higher or lower in the results to mimic numerical effects or compensate for mesh inconsistencies.

  External pumping and puffing are more complex:

- *Puffing only:* The ionization profile is normalized to 1 A and later rescaled to match the fixed upstream density. Even without pumping, EMC3 ensures particle balance by artificially adjusting the divertor recycling rate.
- *Pumping only (no puffing):* EMC3 maintains normalized particles, but increases the divertor-sourced particle flux artificially to preserve balance.

Particles striking the outer wall boundary are treated as lost. The number of particles lost can be tuned using the **radial decay length** parameter: shorter decay lengths reduce wall losses. However, to restore global balance, particles are artificially injected from the divertor region.

Tip: Start with a baseline case without puffing or pumping, and track particle balance to understand the default behavior.

- **Boundary Conditions (BC):**

  Density: Fixed density BC is commonly used (switch 1 below). Flux BC (switch 0, below case used 1, so, density BC) is also supported.

```
* H ions in cm^-3
1           1.9e+13
```

  Particle balance switch (pumping vs puffing): If both puff and pump are present in 'input.eir', use the switch in 'input.par':

```
2   → balance pump and puff
1   → upstream density is fixed
0   → balance based on recycling flux
```

  Core BC setup:

```
1   1   1   0.0
```

  Where: - 1st: Zone number - 2nd: BC index (from input.ngo) - 3rd: Particle source flag from core (1 = particle source in the core) - 4th: particle source magnitude in A (0 A here, no core ion source)

  Edge BC setup (outer radial boundary):

```
1   2  -2   1.0
```

  Where: - 2nd: Radial boundary index (outermost radial surface) - 3rd: Decay length flag (negative = use decay length) - 4th: Decay length in cm (1.0 cm here)

- **Power Input:**

  Section 6.2: Energy transport (electron and ions):

```
1   1   1   144500.0   144500.0
1   2  -2   1.0        1.0
```

Where: - First line gives heating power (in Watts) per box for electrons and ions. - Second line sets radial decay length (1.0 cm) for both species.

Total power = box power × symmetry factor (e.g., 5). Total power = box power × periodic factor (e.g., 5) × symmetry factor (e.g., 2). The last factor corresponds to stellarator symmetry. `https://wiki.fusion.ciemat.es/wiki/Stellarator_symmetry`

Bohm BC is used near the target. Electron and ion sheath transmission coefficients are automatically adjusted based on plasma conditions and can be verified post-simulation.

- **Momentum Balance:**

  Section 6.3: Momentum transport:

```
1   1   1   0.0
1   2  -2   1.0
```

  Where: - **First line:** Defines the core momentum source. - The third term (‘1‘) indicates that a momentum source is present. - The fourth term (‘0.0‘) specifies the source magnitude — here, no core momentum source is applied. - **Second line:** Specifies the boundary condition at the outer radial edge. - The third term (‘-2‘) indicates a radial decay condition. - The fourth term (‘1.0‘) sets the decay length to 1.0 cm.

- **Input Control (input ctr)**

  – **Program Control Block**

```
1     12345
```

  - First value (‘1‘): Control flag (keep as is). It is a global iteration number. It is usually 1. - Second value (‘12345‘): Seed for the random number generator (can be modified but no need to change).

  – **Iteration Block**

```
10     4
```

  - ‘10‘: Number of main iteration loops. - ‘4‘: Number of transport physics blocks enabled below (ENERGY, STREAMING, NEUTRAL, IMPURITY). You can change the order or repeat some blocks as you like. —

  ENERGY Block

```
0     20000     0     0
0.0     0.5     2e-07     5e-08
```

  - First line: - ‘0‘: Unused parameter or flag (keep as is). - ‘20000‘: Number of Monte-Carlo particle (increase if simulation is noisy). - Next two values: Reserved or unused (set to ‘0‘).
  - Second line: - ‘0.0‘: no need to change

  - `0.5`: **Relaxation factor, R**
  Controls how the solution is updated between iterations: Calculation result are replaced with [new]*(1-R) + [previous]*R.

    * `0.0` → Use only the new solution (completely overwrite the previous).
    * `0.5` → Average of old and new results (commonly used for stable convergence).
    * `1.0` → Use only the old solution (no update from the new iteration).

* Higher values (e.g., `0.8`) place more weight on the previous solution, which is helpful in detached conditions or when plasma density is high. This improves stability and avoids oscillations in strongly nonlinear regimes.

- ‘2e-07‘: Time step for perpendicular transport (s). - ‘5e-08‘: Time step for parallel transport (s).

—

STREAMING Block

```
0    20000    0    0
0.0    0.4    0.0    5e-08
```

- Similar format as ENERGY block. - ‘20000‘: Number of particles for streaming. - ‘0.4‘: Relaxation factor. - ‘0.0‘ : no meaning? - Time steps: ‘5e-08‘ for parallel and perpendicular.

—

NEUTRAL Block

```
0    10000    0    0
0.0    0.4    0.0    0.0
```

- ‘10000‘: Number of neutral particles (can increase for better statistics). - Relaxation factor: ‘0.4‘ -

—

IMPURITY TRANSPORT Block

```
2    10000    0    0    1
0    0.5      0    2.0e-07
```

- First line: - ‘2‘: Impurity species index: Impurity species index or transport flag. This is ion species index starting from 2, e.g., 2, 3, 4 ... - ‘10000‘: Number of impurity particles. - ‘0 0‘: Reserved or unused. - ‘1‘: Enable impurity transport output data. It is a control flag of some output data. You do not need to change it. - Second line: - ‘0‘: It is unused - ‘0.5‘: Relaxation factor. - ‘0‘: Unused. - ‘2.0e-07‘: Time step (likely for perpendicular or general transport).

—

End of Control Block

```
END
FIN
```

- Signals the end of the input control block. - ‘FIN‘ terminates the configuration file.

Overall, the structure of `input.ctr` is as follows:

```
[global iteration] [random seed]
[block iteration] [item number]
list of items 1
END
[block iteration] [item number]
list of items 2
END
[block iteration] [item number]
list of items 3
END
(Arbitrary number of blocks can be put here.) FIN
```

- EMC3 requires plasma to touch some surface in the toroidal direction, otherwise there is no neutral injected by surface recombine. You need a sufficiently large mesh covering at least some part of vessel to ensure some contacting areas of the plasma on the vessel. That may be the most difficult point to make a mesh because such radially far region have magnetic island or stochastic field, and defining a guiding surface become difficult or even impossible withe the current version of FLARE. In such case, you have to ask Heinke to find a workaround.