Instructions for the Norah ASCOT workflow

S. Scott

CFS

July 21, 2021

*this file: norah\_workflow\_rev2.docx*

On 7/21/2021 did the following copies:

cp template\_step\_1b.py template\_step\_1d.py

cp template\_step\_2b.py template\_step\_2d.py

cp template\_step\_3b.py template\_step\_3d.py

cp template\_step\_1c.py template\_step\_1e.py

cp template\_step\_2c.py template\_step\_2e.py

cp template\_step\_3c.py template\_step\_3e.py

Sets ‘d’ and ‘e’ are new. They will be modified to use the reverse equilibrium.

template\_step\_2x 🡪 template\_step\_4x (x=a,b,c) step 4: \_reverse

template\_step\_3x 🡪 template\_step\_5x (x =a,b,c) step\_5: \_reverse

where \_reverse means we use the reversed equilibrium.

Step ‘d’ is higher resolution: Nmrk = 5,360,000/21,440 versus 2,680,200/10,720 for step ‘e’.

Change from rev1: for version

* cp template\_step\_2x.py 🡪 template\_step\_4x.py x=a,b,c. then reverse dir’n of equilibrium
* cp template\_step\_3x.py 🡪 template\_step\_5x.py x=a,b,c. then reverse dir’n of equilibrium

This document describes the ‘Norah’ workflow for ASCOT simulations. It comprises three different ASCOT simulations.

Sequencing: you can run steps #1 and #2 in parallel, i.e. the result of step #1 is not needed as input to step 2. Step #3 requires some output from step #1 and from step #2.

All simulations follow markers “to the wall”, i.e. not to the LCFS.

**Step 1: Generate a large (~multi-million) ensemble of markers**

Simulation time: 1 microsecond.

Number of markers = 5,360,000

This is an ‘ab-initio’ generation of markers: the lossmap algorithm is not applied.

Markers are typically followed from their birth at rho\_poloidal = [0.7,1.] to the wall (not LCFS).

template file: template\_step\_1a.py 🡪 assume that the ASCOT output file is ascot\_12345678.5

Changes to template file: you may need to change filenames for the magnetic field, kinetic profiles, plasma equilibrium, and wall shape. These are identified by the label “possible\_change” in the template file.

weights\_file: none (the lossmap algorithm is not applied)

lossmap\_file: none (the lossmap algorithm is not applied)

Nodes: 10.

Required run time < 3 minutes (can use debug queue)

Sample: group\_go\_1166.py, knl\_1166.py 🡪 ascot\_41298536.h5 (this one used only 5,000,000 markers). Took 2 minutes, 31 seconds on 10 knl nodes.

Postprocessing: None required.

**Step 2: Compute the prompt and nonprompt loss fraction and save data that will be used for the lossmap**

21,440 markers generated ab-initio over rho\_poloidal = [0.7,1.0]. This is an ‘ab-initio’ simulation: no lossmap is applied to the ensemble of markers.

Follow markers to the wall for a simulation time of 50 msec.

template file: template\_step\_2b.py 🡪 assume that the ASCOT output file is ascot\_87654321.h5

Changes to template file: you may need to change filenames for the magnetic field, kinetic profiles, plasma equilibrium, and wall shape. These are identified by the label “possible\_change” in the template file.

Required run time: 4.4 hours on 10 knl nodes

weights\_file: none (the lossmap algorithm is not applied)

lossmap\_file: none (the lossmap algorithm is not applied)

Sample: group\_go\_1235.py, knl\_1235.py 🡪 ascot\_42741085.h5 (but this simulation ran for a simulation time of only 50 ms). Took 2 hours, 46 minutes on 10 knl nodes.

Postprocessing: pnp\_losses.py 🡪 compute energy loss fractions for prompt- and non-prompt losses.

**Step 3: Compute the spatial pattern of the alpha losses**

Use the ensemble of markers created in step #1 and the lossmap file generated in step #2 to create a large ensemble of markers, but excluding those markers that have no chance of hitting the wall.

Follow orbits for 0.5 ms or until they hit the wall.

template file: template\_step\_3a.py

Changes to template file:

* fn\_lossmap = 'ascot\_87654321.h5’ (i.e. ASCOT output file from step #2)
* fn\_weights = 'ascot\_12345678.h5’ (i.e. ASCOT output file from step #1)
* You may need to change filenames for the magnetic field, kinetic profiles, plasma equilibrium, and wall shape. These are identified by the label “possible\_change” in the template file.

Nodes: 20 Run time: 90 minutes (might want to ask for 3 hours on first run to be sure)

Sample: group\_go\_1200.py, knl\_1200.sh 🡪 ascot\_42375387.h5

Postprocessing: remy.py