SD1D and Hermes

INTRODUCTION AND HANDS-ON EXERCISES

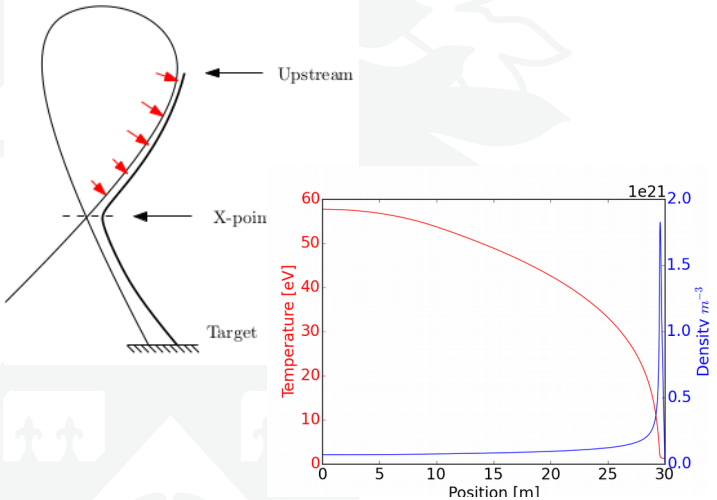
Ben Dudson, J.Leddy, M.Thomas, B.Lipshultz, D.Moulton, P.Hill, J.Parker and others

BOUT++ workshop, LLNL
16th August 2018

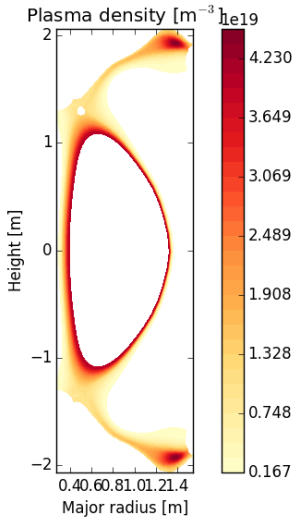
This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Edge modelling codes

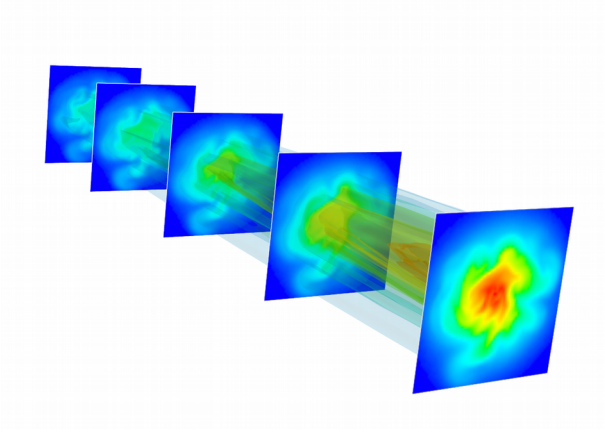
SD1D



Hermes (2D)



Hermes (3D)

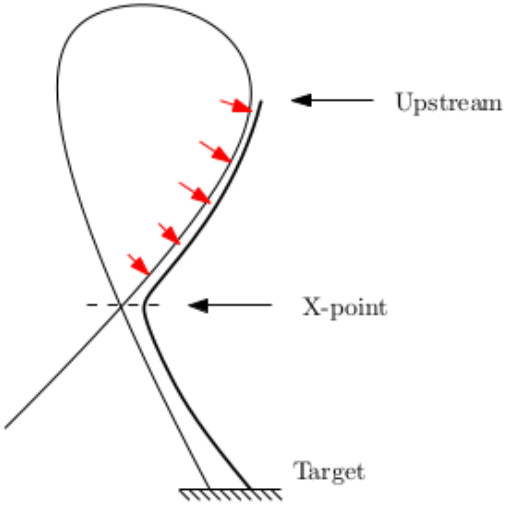


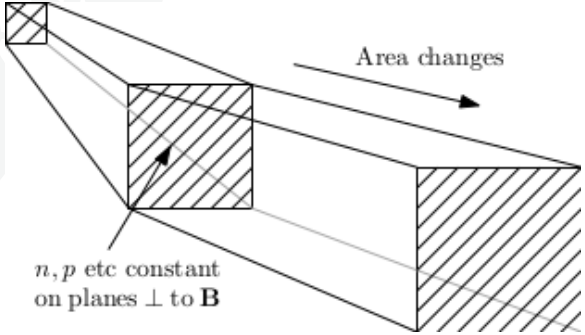
http://boutproject.github.io/
https://github.com/boutproject

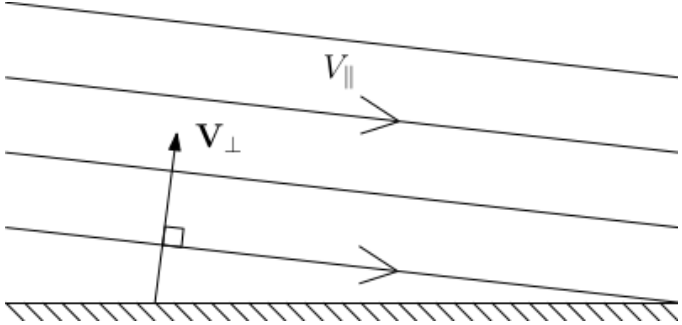
SD1D: Dynamics in 1D

https://github.com/boutproject/SD1D

- Only parallel variation is included
- Flux tube area changes due to changes in |B| → Changes heat flux
- Cross-field transport (of neutrals) is included as an effective parallel diffusion

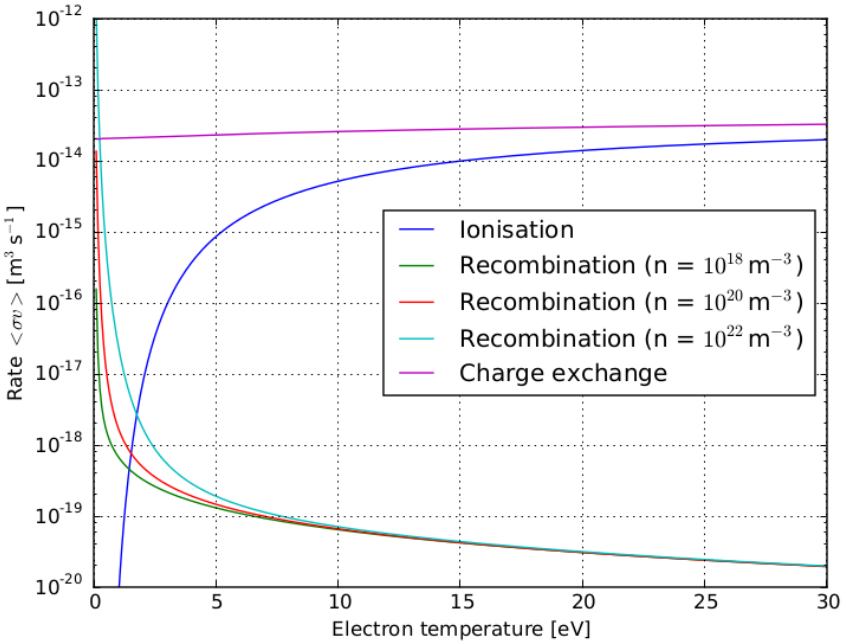






Atomic physics

- No molecules, only atoms
- Fixed fraction impurity model
- Semi-analytic lookup tables for charge exchange, ionisation and recombination.
- ADAS database used for cooling curves, or simple analytic fits.



E.Havlicova, H.Willett, K.Gibson

Plasma equations

Evolving the plasma density, pressure, and parallel momentum. This is coupled to external sources and neutrals:

$$
\begin{aligned}
\frac{\partial n}{\partial t} &= -\nabla \cdot \left[\mathbf{b} V_{||e} n\right] + S_n - S \\
\frac{\partial}{\partial t}\left(\frac{3}{2} p\right) &= -\nabla \cdot \mathbf{q}_e + V_{||i} \partial_{||} p + S_E - E - R \\
\frac{\partial}{\partial t}\left(m_i n V_{||i}\right) &= -\nabla \cdot \left[m_i n V_{||i} \mathbf{b} V_{||i}\right] - \partial_{||} p - F \\
V_{||e} &= V_{||i} \\
\mathbf{q}_e &= \frac{5}{2} p \mathbf{b} V_{||i} - \kappa_{||} \partial_{||} T_e
\end{aligned}
$$

Neutral equations

Similar set of equations for the neutral density, pressure, and parallel momentum:

$$\frac{\partial n_n}{\partial t} = -\nabla \cdot [\mathbf{b} V_n n_n] + S$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} p_n\right) = -\nabla \cdot \mathbf{q}_n + V_{||n} \partial_{||} p_n + E$$

$$\frac{\partial}{\partial t}\left(m_i n_n V_{||n}\right) = -\nabla \cdot \left[m_i n_n V_{||n} \mathbf{b} V_{||n}\right] - \partial_{||} p_n + F$$

$$V_n = V_{||n} - \boxed{\left(\frac{B_\phi}{B_\theta}\right)^2 \frac{\partial_{||} p_n}{\nu}}$$

Cross-field diffusion

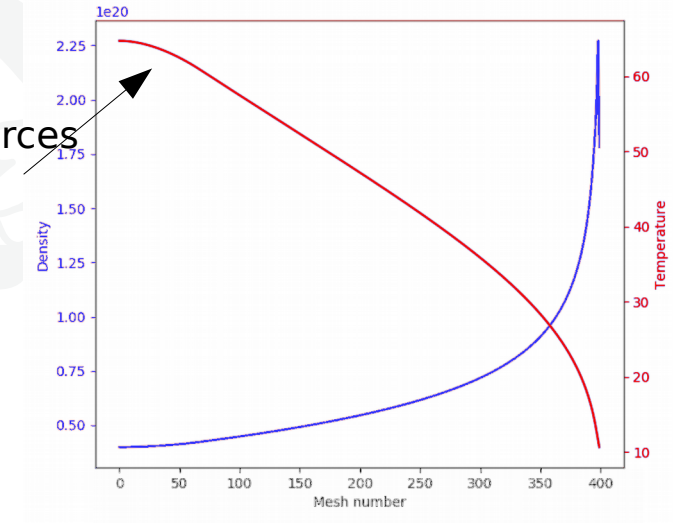
$$\mathbf{q}_n = \frac{5}{2} p_n \mathbf{b} V_{||n} - \kappa_{||n} \mathbf{b} \partial_{||} T_n$$

Comparison to SOLPS-ITER

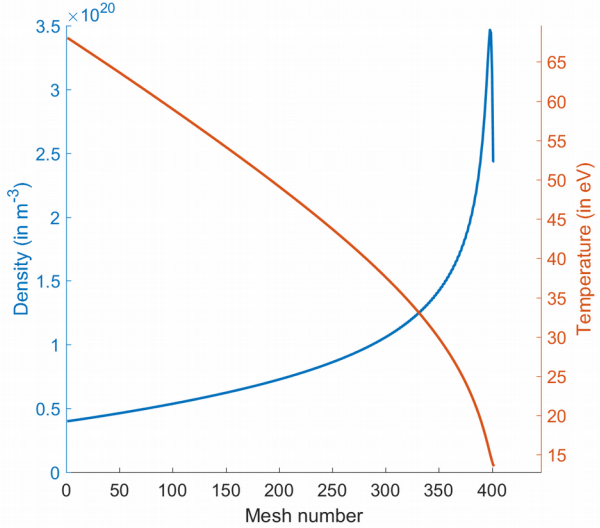
- $L_{||}$ = 30 m, upstream density 4e19 m^{-3}, power flux 50 MW/m^2
- SOLPS-ITER with fluid neutrals on the same grid.

  ~ 20% difference in target density, temperature.

Different treatment of sources (volume vs boundary)



SD1D



1D SOLPS-ITER (fluid neutrals)

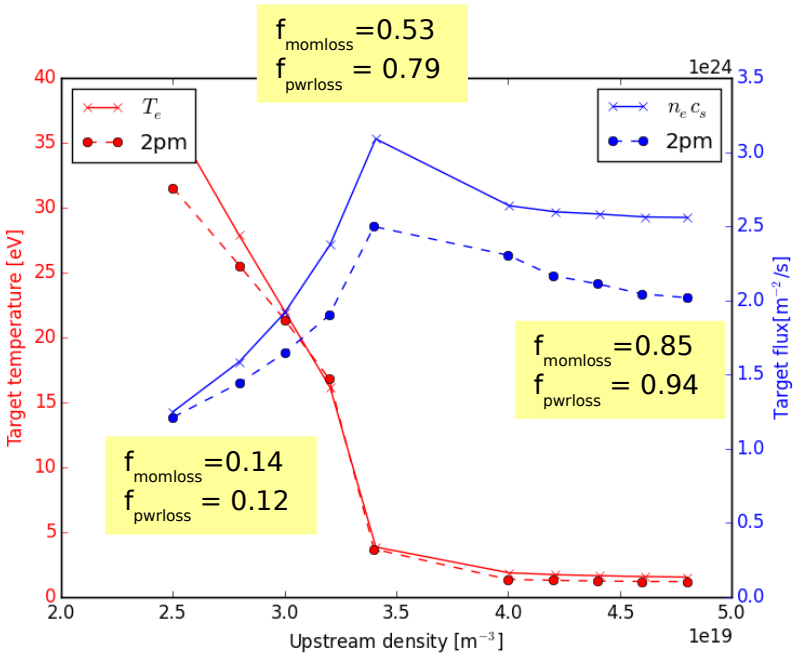
Comparison to M2PM

$$n_t = \left[\frac{\gamma^2}{32m}\right]\left[\frac{p_{tot,u}^3}{\tilde{q}_{\|u}^2}\right]\left[\frac{(1-f_{momloss})^3}{(1-f_{pwrloss})^2}\right]$$

$$\times\left[\frac{4}{(1+T_{it}/T_{et})^2}\right]\left[\frac{8M_t^2}{(1+M_t^2)^3}\right]\left[\left(\frac{B_u}{B_t}\right)^2\right]$$

$$T_{et} = \left[\frac{8m}{e\gamma^2}\right]\left[\frac{\tilde{q}_{\|u}^2}{p_{tot,u}^2}\right]\left[\frac{(1-f_{pwrloss})^2}{(1-f_{momloss})^2}\right]$$

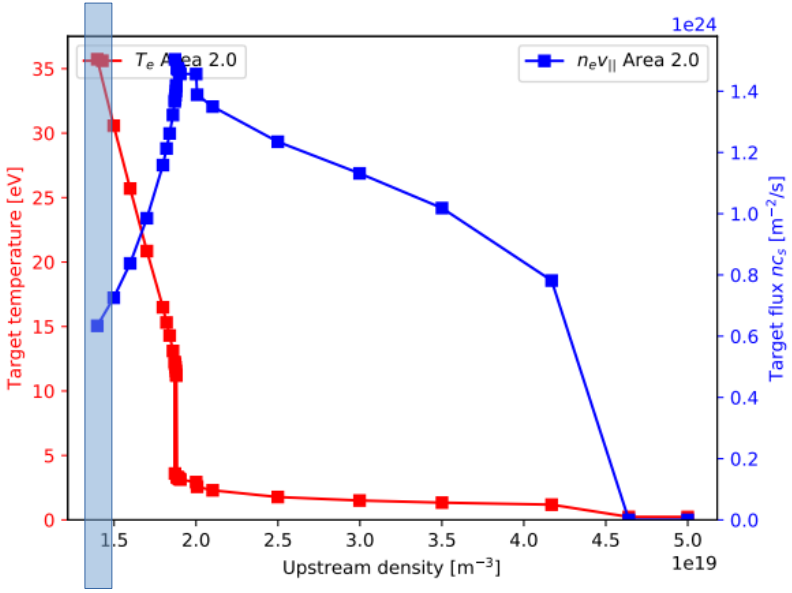
$$\times\left[\frac{(1+T_{it}/T_{et})}{2}\right]\left[\frac{(1+M_t^2)^2}{4M_t^2}\right]\left[\left(\frac{B_t}{B_u}\right)^2\right]$$

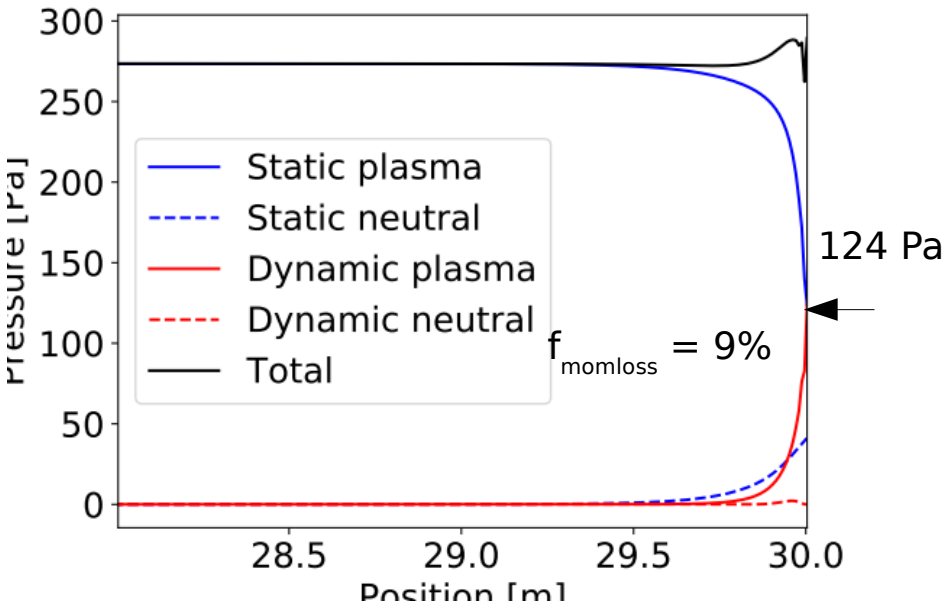


[1] V.Kotov, D.Reiser PPCF 51, 115002 (2009)
[2] P.C.Stangeby APS 2015
[3] D.Moulton, J.Harrison, B.Lipschultz, D.Coster PPCF 59, 065011 (2017)

Density scan

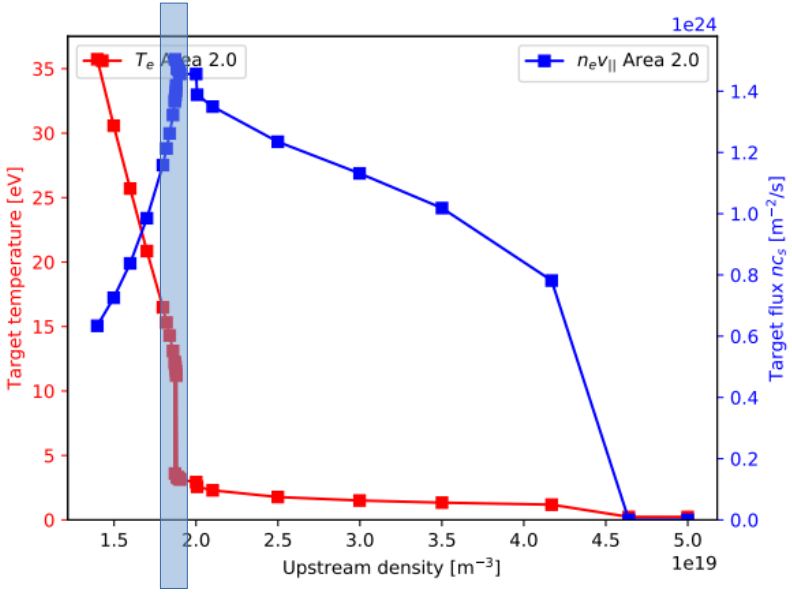
- Area expansion factor = 2
- 99% recycling at target
- Includes Hydrogen excitation and 1% Carbon (ADAS model)

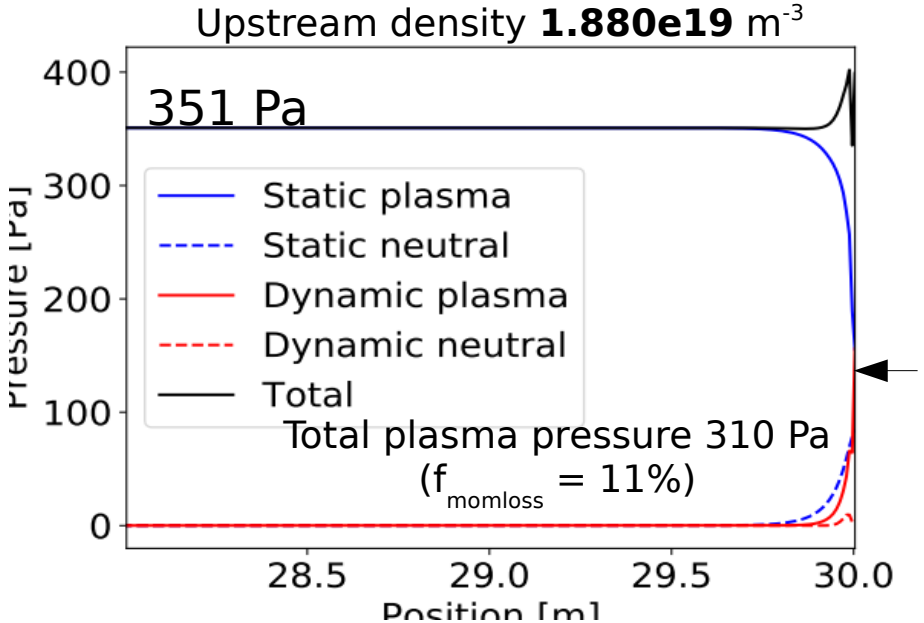




Density scan

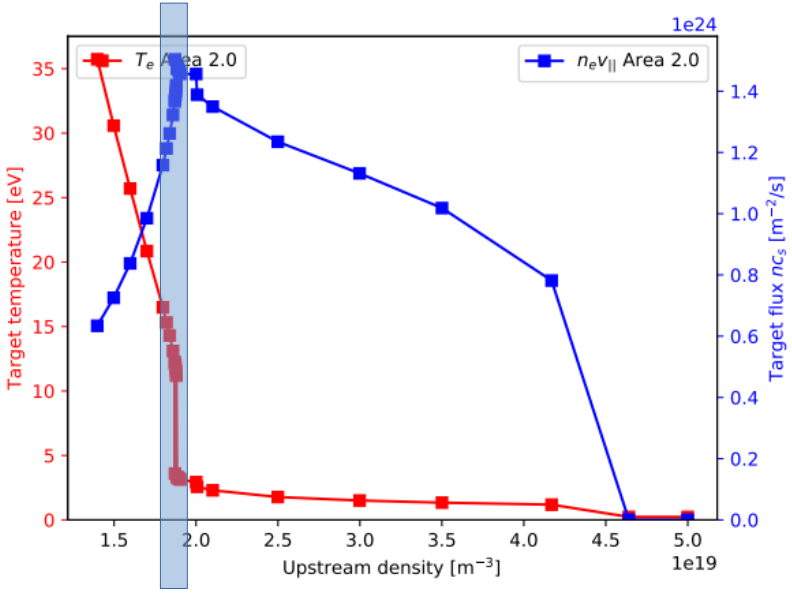
- Rapid change seen at flux rollover
- Some small hyseresis found

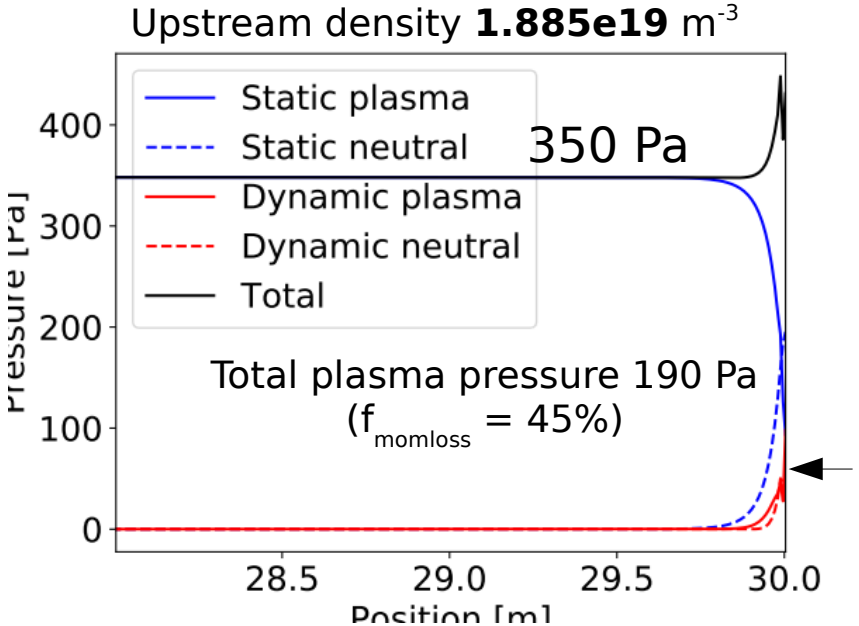




Density scan

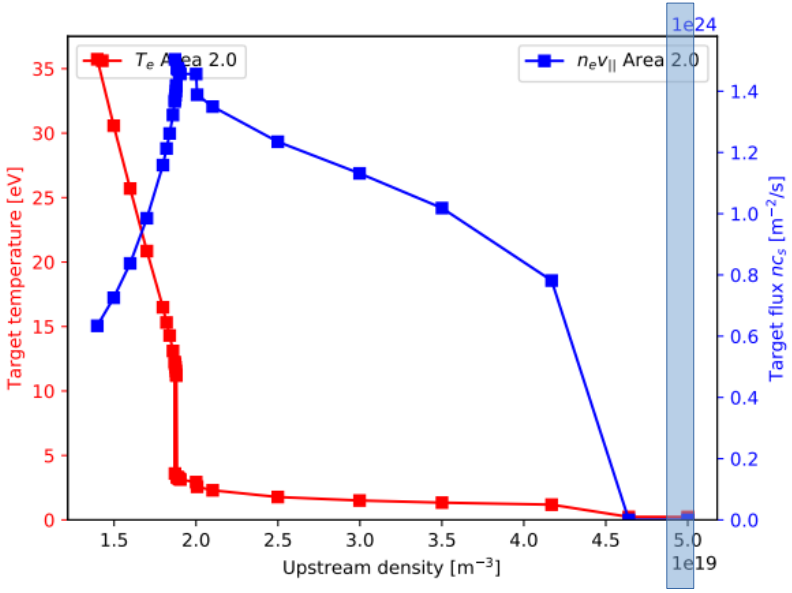
- Rapid change seen at flux rollover
- Some small hyseresis found

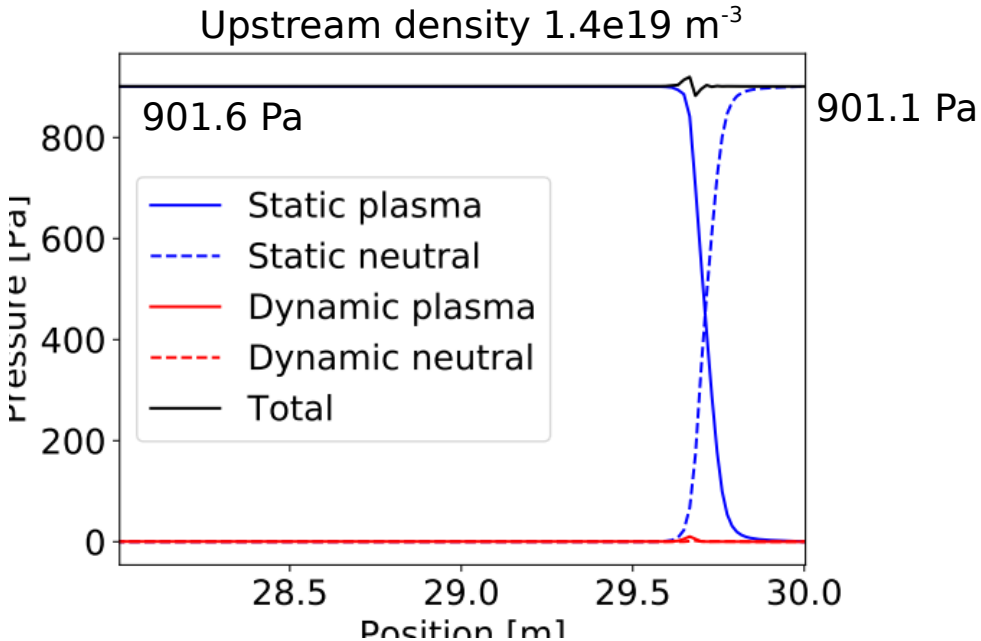




Density scan

- Loss of plasma pressure mostly balanced by neutral static pressure
- Front width ~cm in parallel direction

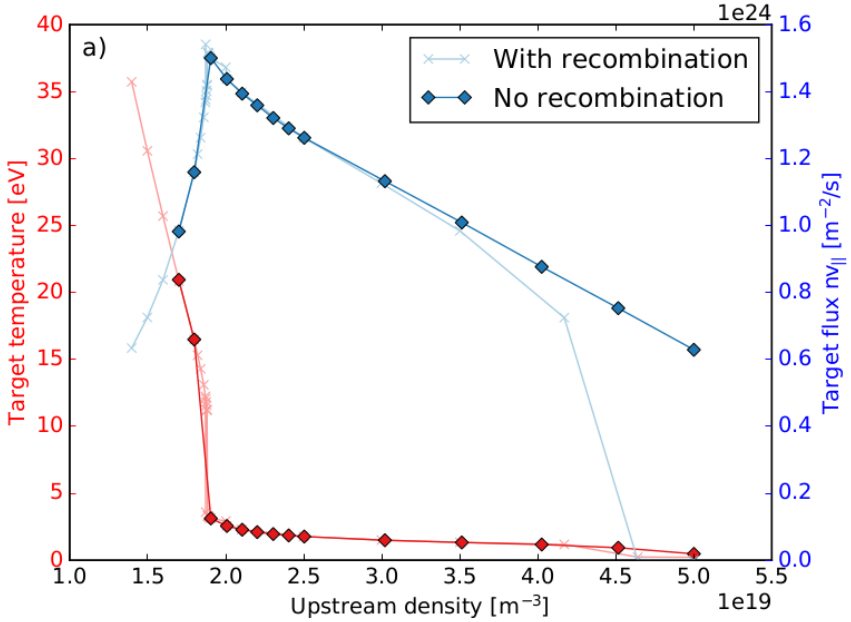




The role of recombination

Paper: "The role of particle, energy and momentum losses in 1D simulations of detachment" in preparation

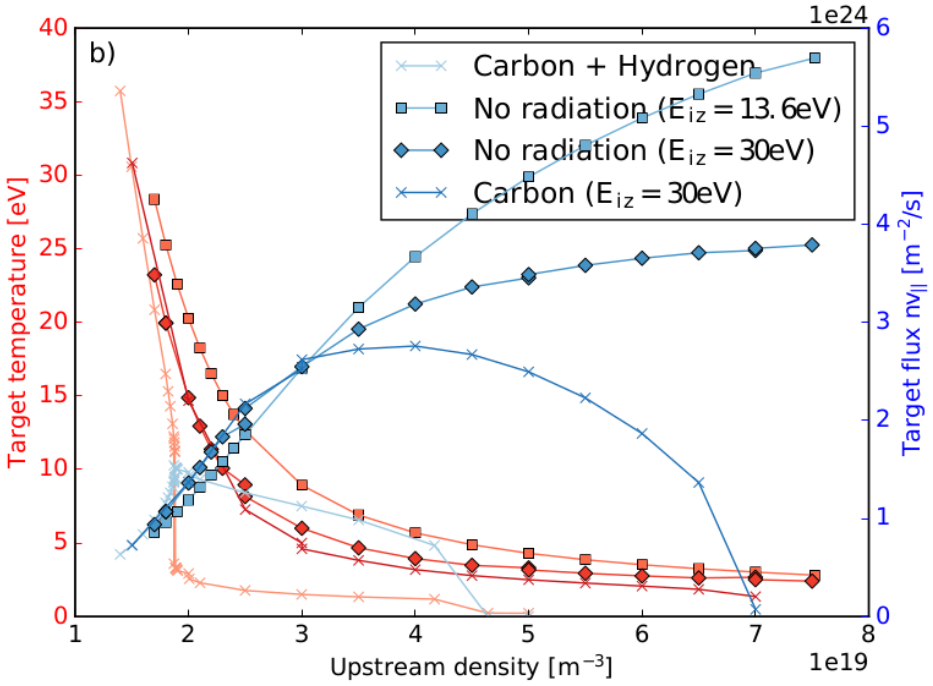
- Recombination not important in flux rollover in these cases (at $n_{up} \sim 1.9 \times 10^{19}\ m^{-3}$)
- Recombination becomes significant only at low temperatures after rollover (at $n_{up} \sim 4 \times 10^{19}\ m^{-3}$)



The role of radiation

Paper: "The role of particle, energy and momentum losses in 1D simulations of detachment" in preparation

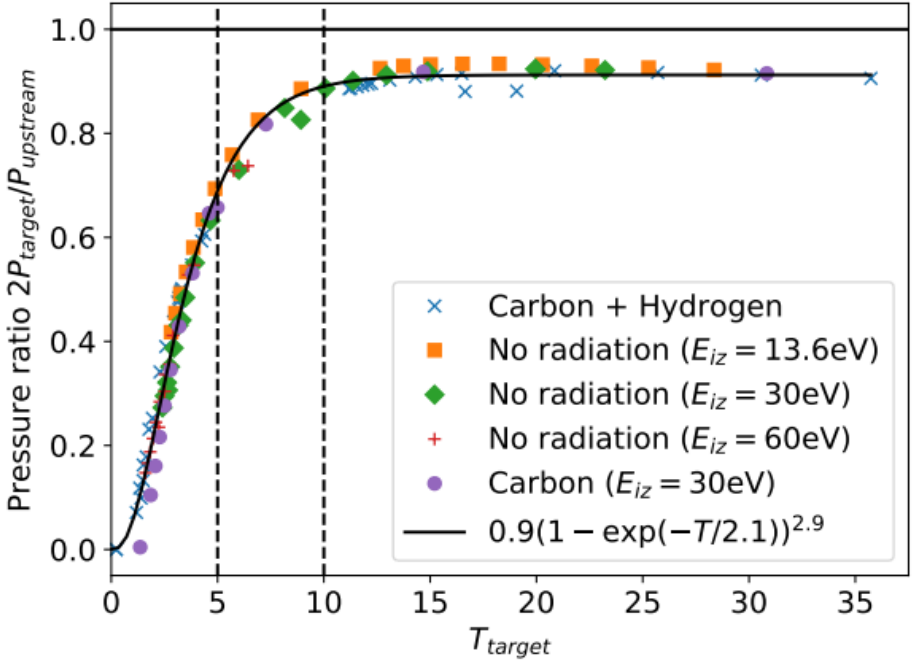
- When the energy cost per ionisation is fixed, no flux roll over occurs.
- Even fixed ionisation cost E_{iz} = 60eV does not have flux rollover
- Either hydrogen or impurity radiation which varies with T_e is required for flux rollover.



Pressure loss and target Te

Paper: "The role of particle, energy and momentum losses in 1D simulations of detachment" in preparation

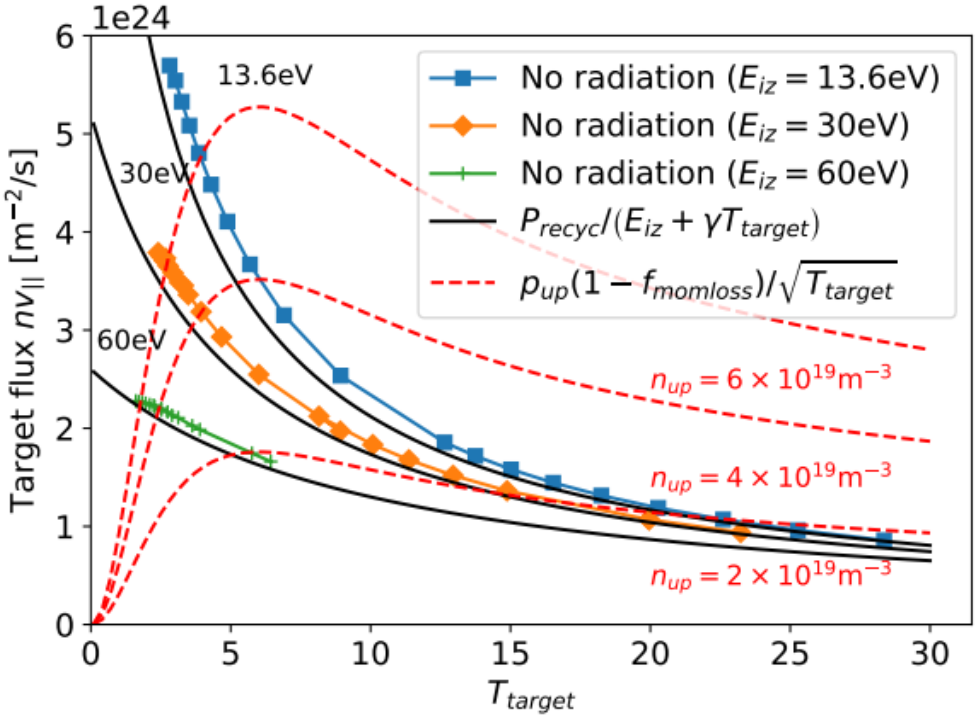
- Most simulations follow the same pressure loss curve
- Significant pressure loss occurs in cases with no flux rollover



Pressure loss and target Te

Paper: "The role of particle, energy and momentum losses in 1D simulations of detachment" in preparation

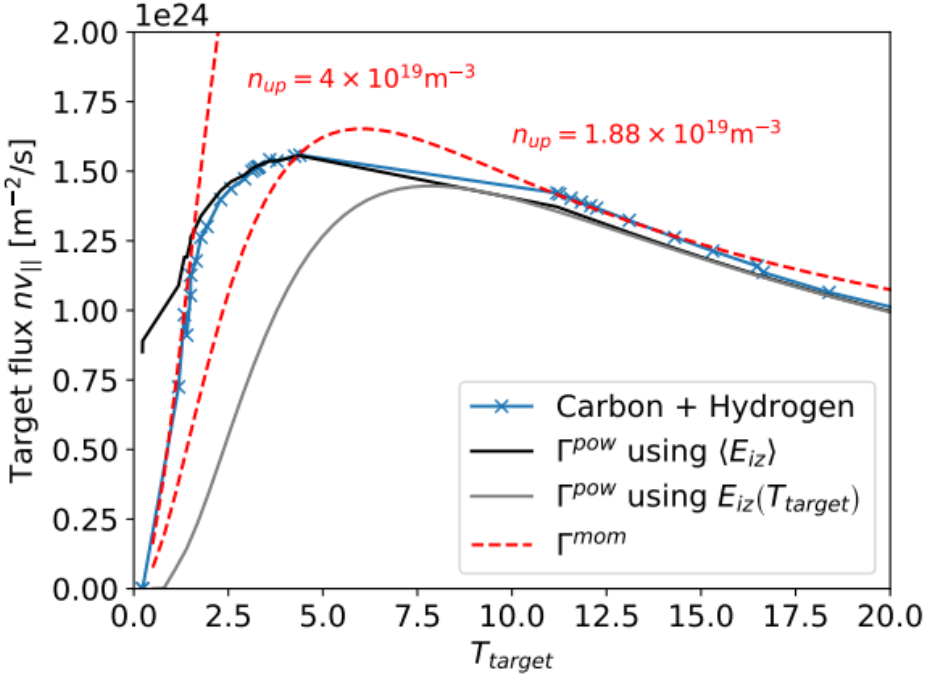
- In steady state both power balance (**black**) and momentum balance (red) must be obeyed.
- Shows why no rollover in cases with fixed E_{iz} and no impurities.
- Recombination could remove particle flux, but is not significant here.



Pressure loss and target Te

Paper: "The role of particle, energy and momentum losses in 1D simulations of detachment" in preparation

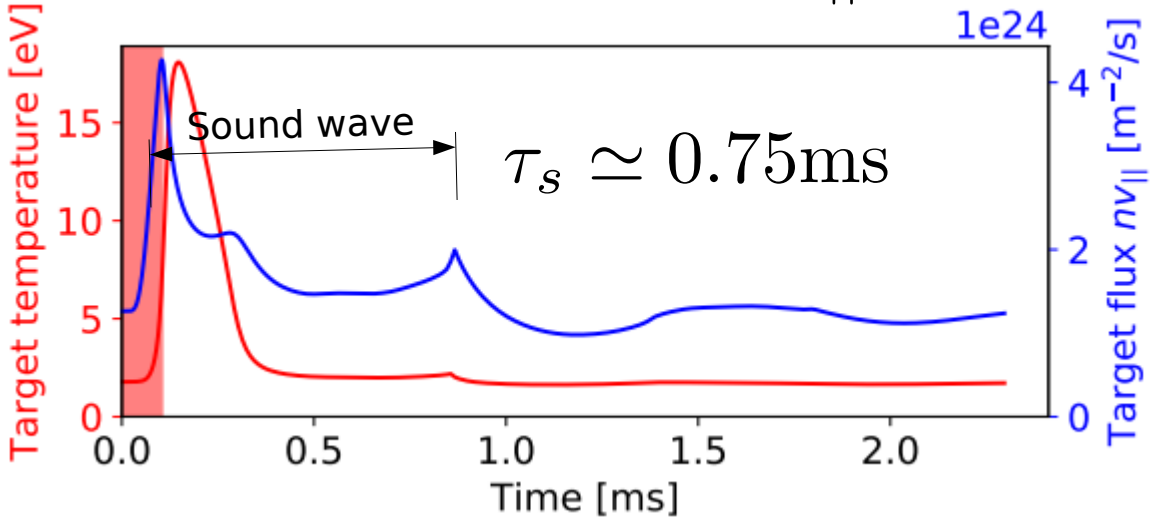
- In steady state both power balance (**black**) and momentum balance (red) must be obeyed.
- In the case where a rapid transition occur, the black and red lines diverge and the solution "jumps"



Time dependence example

- Power increased from 50 to 200 MW/m² for 100μs.
- Target flux pulse ahead of temperature increase.
- Detachment recovery ~ 200μs after 100μs power pulse.

$$c_s \simeq 4 \times 10^4 \mathrm{m/s} \qquad L_{||} = 30\mathrm{m}$$



SD1D summary

- Simple but flexible tool for understanding divertor physics
- Time-dependent, but can be used to study steady-state solutions
- Studies of dynamics, response to heat pulses, feedback control etc. just beginning
- Improvements planned e.g. adding molecules
- Collaborations, contributions welcome!

SD1D: Getting started

GitHub, Inc. (US) https://github.com/boutproject/SD1D 13

https://github.com/boutproject/SD1D

SD1D

SOL and Divertor in 1D. Simulates a plasma fluid in one dimension (along the magnetic field), inte
fluid.

- Evolves the density, momentum and pressure (internal energy) of both plasma and neutrals.
- Includes exchange of particles, momentum and energy through ionisation, recombination and
- Several different numerical methods are implemented, both upwind and central differencing

Author: Ben Dudson, University of York benjamin.dudson@york.ac.uk

Released under the GPL license

SD1D: Manual installation

- Needs BOUT++ version >= 4.0
- Configured with SUNDIALS 2.6 or 2.7 for preconditioning

BOUT++

```
git clone https://github.com/boutproject/BOUT-dev.git
cd BOUT-dev
./configure --with-cvode
make
```

SD1D

```
git clone https://github.com/boutproject/SD1D.git
cd SD1D
make BOUT_TOP=/path/to/BOUT-dev/
```

```
mpirun -np 4 ./sd1d -d case-01
```

Install docker

Jarrod Leddy (Tech-X)

- Provides a reproducible environment (like a virtual machine or BSD jail)

Ubuntu linux
https://docs.docker.com/install/linux/docker-ce/ubuntu/

Mac:
https://docs.docker.com/docker-for-mac/install/

Windows:
https://docs.docker.com/docker-for-windows/install/

Others:
https://docs.docker.com/v17.12/install/

SD1D docker image

Jarrod Leddy (Tech-X)

- Provides a reproducible environment (like a virtual machine or BSD jail)
- Preinstalled with SUNDIALS, BOUT++, SD1D, and python analysis tools

```
$ sudo docker pull boutproject/sd1d:049f085

$ sudo docker run --rm -it boutproject/sd1d:049f085
```

Delete changes after exit

Interactive shell

SD1D docker image

```
$ sudo docker run --rm -it boutproject/sd1d:049f085

boutuser@4f8d6160306c:~$ ls
BOUT-next  SD1D  bout-img-shared

boutuser@4f8d6160306c:~$ cd SD1D
boutuser@4f8d6160306c:~/SD1D$ ls
LICENSE      case-05                  json_database          profiles.py
README.md    case-05-impurity         loadmetric.cxx         radiation.cxx
atomicpp     div_ops.cxx              loadmetric.hxx         radiation.hxx
atomicpp.a   div_ops.hxx              loadmetric.o           radiation.o
case-01      div_ops.o                makefile               sd1d
case-02      doc                      modified-two-point.py  sd1d.cxx
case-03      energy-flux.py           pressure-balance.py    sd1d.o
case-04      impurity_user_input.json processes.py

boutuser@4f8d6160306c:~/SD1D$ ./sd1d -d case-01
```

SD1D docker options

```
$ mkdir shared
$ sudo docker run --rm -it \
    -e DISPLAY -v $HOME/.Xauthority:/home/boutuser/.Xauthority --net=host \
    -v $PWD/shared:/home/boutuser/bout-img-shared \
    --cpuset-cpus=0-3 \
    boutproject/sd1d:049f085
```

SD1D docker options

```
$ mkdir shared
$ sudo docker run --rm -it \
    -e DISPLAY -v $HOME/.Xauthority:/home/boutuser/.Xauthority --net=host \
    -v $PWD/shared:/home/boutuser/bout-img-shared \
    --cpuset-cpus=0-3 \
    boutproject/sd1d:049f085
```

- Enable X11 for graphics / plotting
- Link a shared folder to copy files between docker and host
- Restrict number of cores, here up to 4

Docker on NERSC (1/4)

- Docker images can be pulled onto NERSC and run using Shifter

```
shifterimg -v pull docker:boutproject/sd1d:049f085
```

```
> shifterimg images | grep sd1d
edison    docker    READY    b7f3ead172    2018-08-13T15:49:21 boutproject/sd1d:049f085
> salloc -N 1 -p debug -t 00:30:00 \
    --image=docker:boutproject/sd1d:049f085
> shifter bash
> cd $SCRATCH
> cp -r /home/boutuser/SD1D .
> cp -r /home/boutuser/BOUT-next/tools/pylib .
> cd SD1D
> ./sd1d -d case-01
> exit  [ Ctrl-D ]
```

NERSC: Run in parallel (2/4)

```
> salloc -N 1 -p debug -t 00:30:00 \
    --image=docker:boutproject/sd1d:049f085

> cd $SCRATCH/SD1D
> srun -n 20 shifter ./sd1d -d case-05/
```

NERSC: Analyse data (3/4)

```
> module load python/3.6-anaconda-5.2
> pip install netcdf4
> export PYTHONPATH=$SCRATCH/pylib:$PYTHONPATH
> cd $SCRATCH/SD1D/case-01
> ipython
In [1]: from boutdata import collect
In [2]: n = collect("Ne")
In [3]: from boututils.showdata import showdata
In [4]: showdata(n[:,0,:,0])
```

See manual (doc/ subdirectory) with table of outputs

NERSC: analysis scripts

- Several analysis scripts available (in Python)

```
> module load python/3.6-anaconda-5.2
> export PYTHONPATH=$SCRATCH/pylib:$PYTHONPATH

> cd $SCRATCH/SD1D
> python energy-flux.py case-05
> python modified-two-point.py case-05
```

- Run without path to see options
- Add “-h” to see help
- Add “-x” to avoid plotting to screen (e.g. no X server)

Hermes

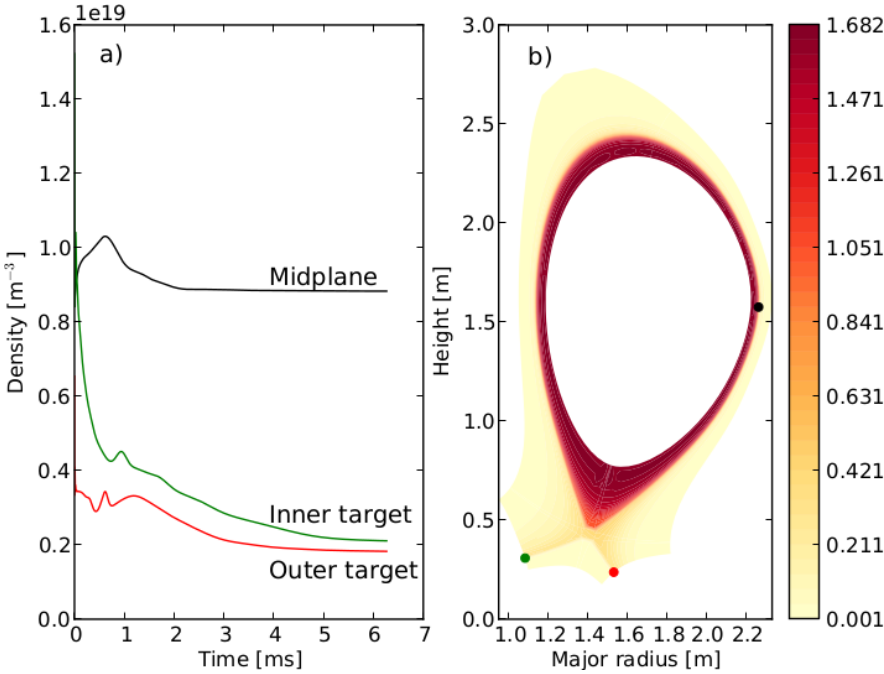
INTRODUCTION AND HANDS-ON EXERCISES

Ben Dudson, Jarrod Leddy, Matt Thomas

BOUT++ workshop, LLNL
16th August 2018

Hermes introduction

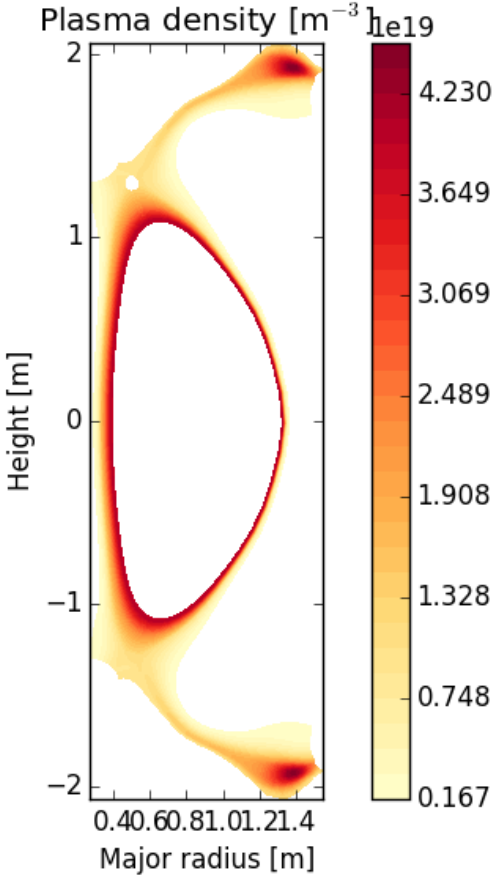
- A 2D or 3D electromagnetic model, based on drift-reduced fluid equations
- Reduces to SD1D if perpendicular dynamics, currents are turned off.
- In 2D it can be used as a transport code like UEDGE
- Fluid neutrals model, atoms only (like SD1D)



B.Dudson, J.Leddy PPCF 59 054010 (2017) https://doi.org/10.1088/1361-6587/aa63d2

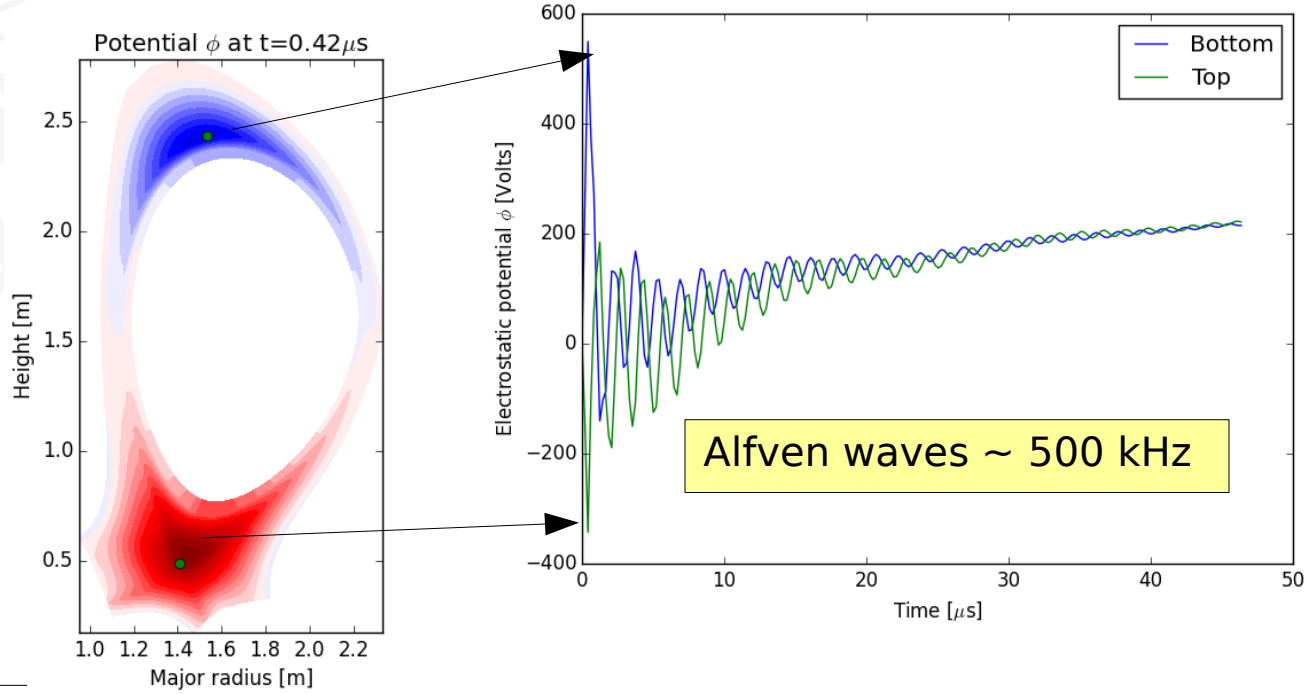
Hermes introduction

- A 2D or 3D electromagnetic model, based on drift-reduced fluid equations
- Reduces to SD1D if perpendicular dynamics, currents are turned off.
- In 2D it can be used as a transport code like UEDGE
- Fluid neutrals model, atoms only (like SD1D)
- Can be used to study detached plasmas (e.g. MAST-U, right)



Electromagnetic fields (2D)

- Time dependent code, evolves global electric field, with diamagnetic and ExB drifts.



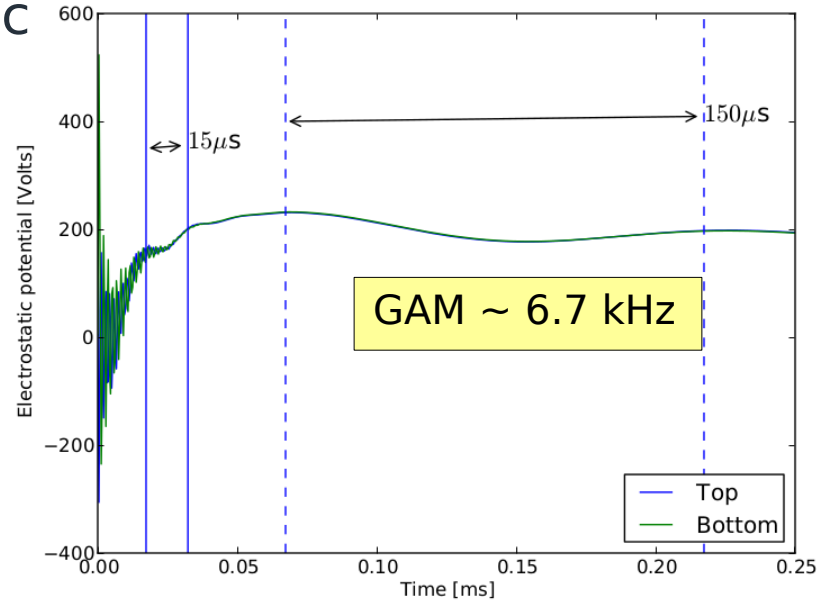
B.Dudson, J.Leddy PPCF 59 054010 (2017) https://doi.org/10.1088/1361-6587/aa63d2

Electromagnetic fields (2D)

- Time dependent code, evolves global electric field, with diamagnetic and ExB drifts.

$$f_{GAM} = \frac{c_s}{2\pi R}\sqrt{2 + 1/q^2}$$
$$\simeq 3 - 11\text{kHz}$$

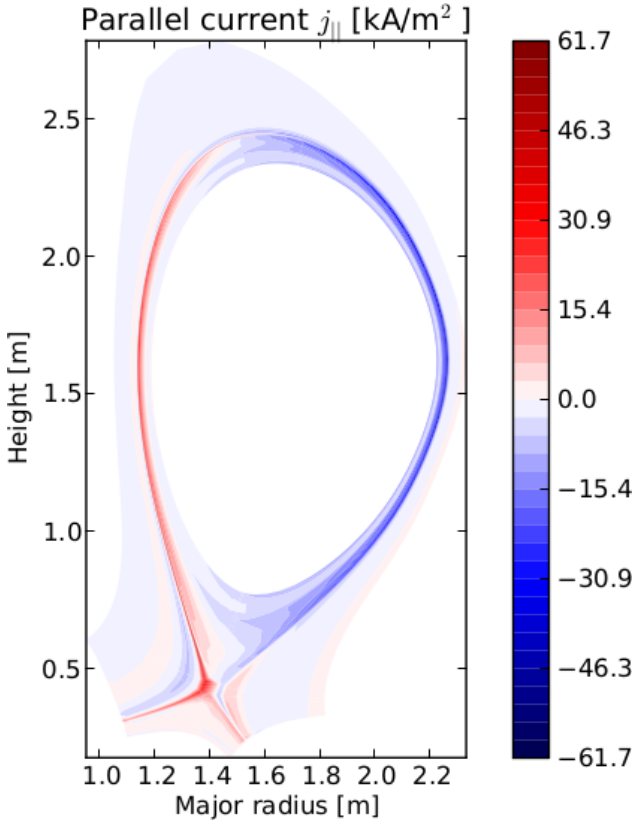
- Alfven waves and Geodesic Acoustic Mode (GAM) oscillations seen.



B.Dudson, J.Leddy PPCF 59 054010 (2017) https://doi.org/10.1088/1361-6587/aa63d2

Electromagnetic fields (2D)

- Time dependent code, evolves global electric field, with diamagnetic and ExB drifts.
- Alfven waves and Geodesic Acoustic Mode (GAM) oscillations seen.
- Evolves to a steady state with Pfirsh-Schluter currents.

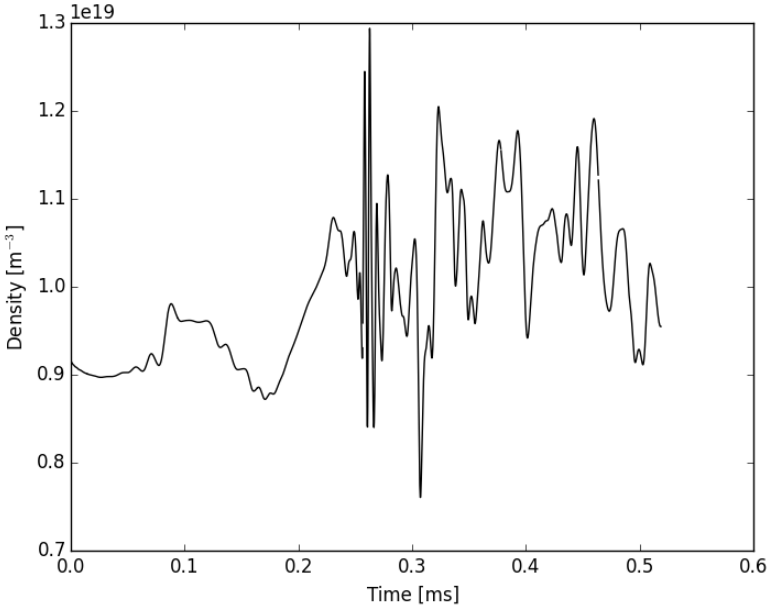


B.Dudson, J.Leddy PPCF 59 054010 (2017) https://doi.org/10.1088/1361-6587/aa63d2

3D turbulence

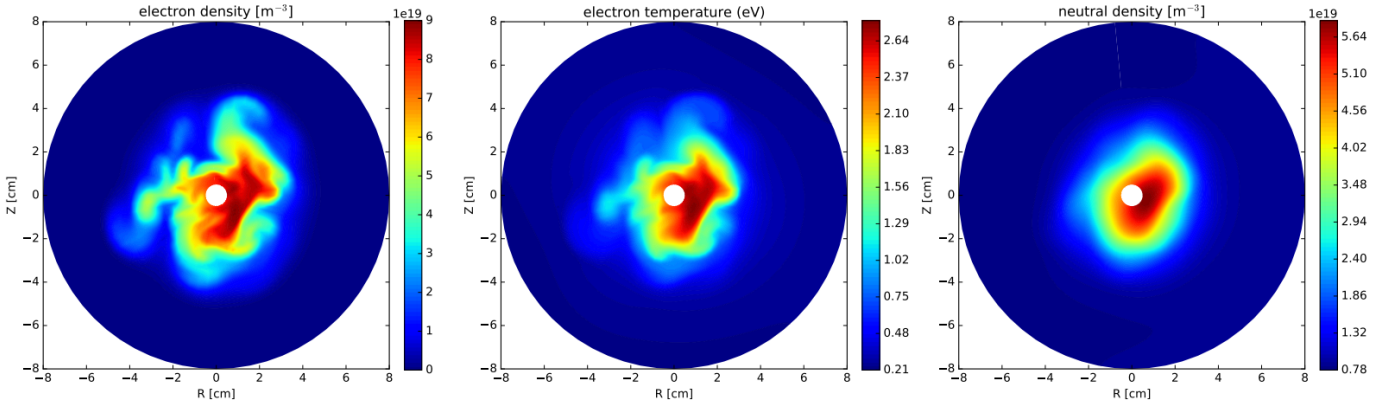
- Extending the grid in the toroidal direction includes n > 0 modes.
- Turbulence determines the transport self-consistently
- Includes resistive, ideal ballooning modes and drift waves
- Main version is cold ion
- Hot ion version being developed

Density, outboard midplane



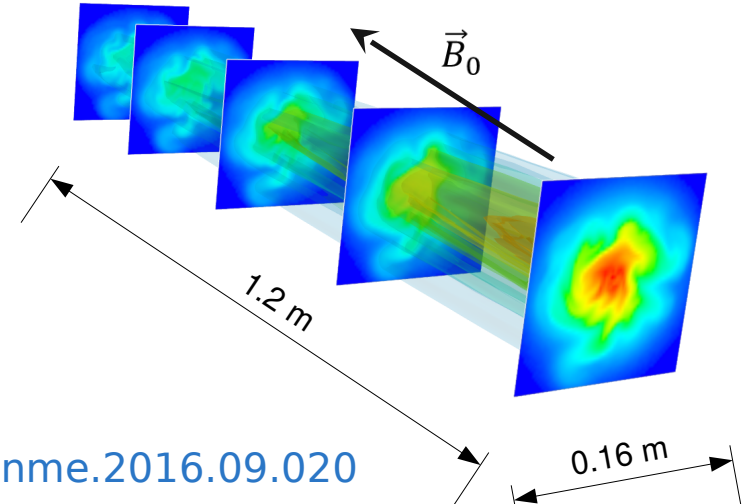
B.Dudson, J.Leddy PPCF 59 054010 (2017) https://doi.org/10.1088/1361-6587/aa63d2

Linear device simulations



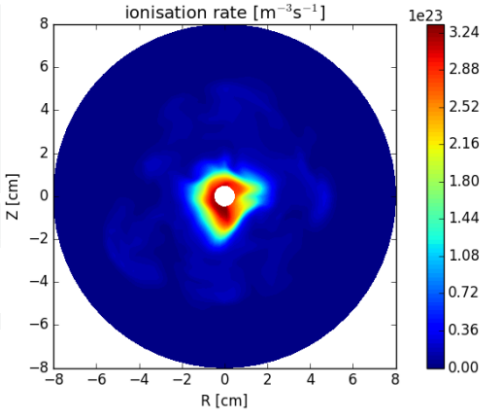
Magnum PSI – sized device

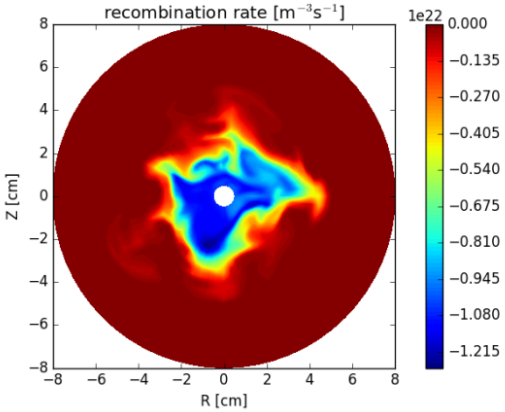
Magnetic field	0.15 T
Length	1.2 m
Radius	10 cm

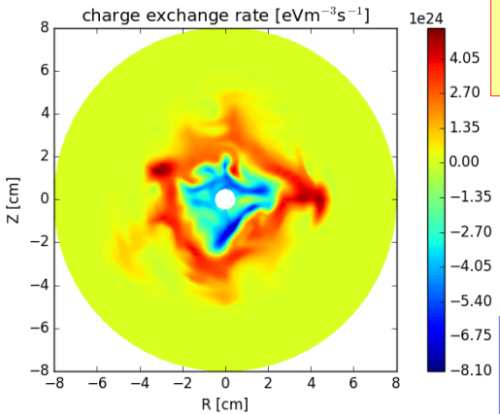


J.Leddy, B.Dudson JNM (2016) https://doi.org/10.1016/j.nme.2016.09.020

Linear device simulations





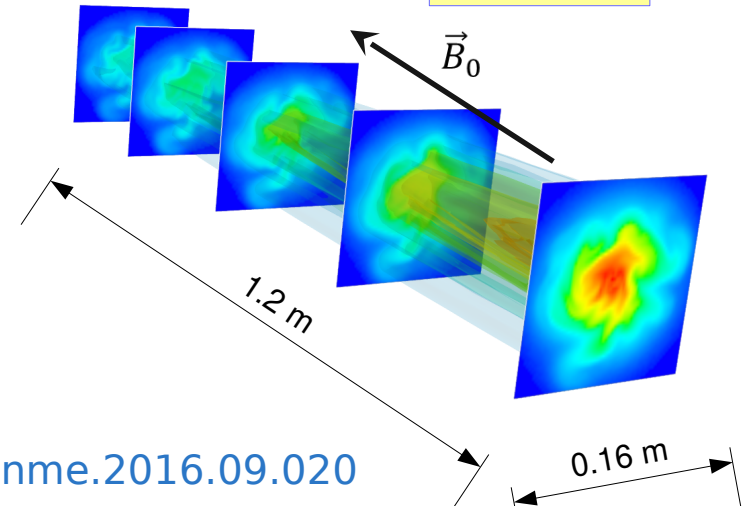


Plasma heating

Plasma cooling

Magnum PSI – sized device

Magnetic field	0.15 T
Length	1.2 m
Radius	10 cm



J.Leddy, B.Dudson JNM (2016) https://doi.org/10.1016/j.nme.2016.09.020

Hermes equations (1/2)

$$\frac{\partial n_e}{\partial t} = -\nabla \cdot \left[n_e \left(\mathbf{V}_{E\times B} + \mathbf{V}_{mag} + \mathbf{b} v_{||e} \right) \right] + \nabla \cdot \left(D_\perp \nabla_\perp n_e \right) + S_n$$

$$\frac{3}{2}\frac{\partial p_e}{\partial t} = -\nabla \cdot \left(\frac{3}{2} p_e \mathbf{V}_{E\times B} + \frac{5}{2} p_e \mathbf{b} v_{||e} + p_e \frac{5}{2} \mathbf{V}_{mag} \right) - p_e \nabla \cdot \mathbf{V}_{E\times B} + v_{||e} \partial_{||} p_e + \nabla_{||} \left(\kappa_{e||} \partial_{||} T_e \right) + 0.71 \nabla_{||} \left(T_e j_{||} \right) - 0.71 j_{||} \partial_{||} T_e + \frac{\nu}{n} j_{||}^2 + \nabla \cdot \left(D_\perp T_e \nabla_\perp n_e \right) + \nabla \cdot \left(\chi_\perp n_e \nabla_\perp T_e \right) + S_p$$

SD1D

With ExB and magnetic drifts given by:

$$\mathbf{V}_{E\times B} = \frac{\mathbf{b} \times \nabla \phi}{B} \qquad \mathbf{V}_{mag} = -T_e \nabla \times \frac{\mathbf{b}}{B}$$

Hermes equations (2/2)

Flows and currents are evolved through the vorticity, ion parallel momentum, and vector potential

$$\frac{\partial \omega}{\partial t} = -\nabla \cdot (\omega \mathbf{V}_{E\times B}) + \nabla_{||} j_{||} - \nabla \cdot (n \mathbf{V}_{mag}) + \nabla \cdot (\mu_\perp \nabla_\perp \omega)$$

$$\frac{\partial}{\partial t}(n_e v_{||i}) = -\nabla \cdot \left[n_e v_{||i} \left(\mathbf{V}_{E\times B} + \mathbf{b} v_{||i}\right)\right] - \partial_{||} p_e + \nabla \cdot \left(D_\perp v_{||i} \nabla_\perp n\right) - F$$

$$\frac{\partial}{\partial t}\left[\frac{1}{2}\beta_e \psi - \frac{m_e}{m_i}\frac{j_{||}}{n_e}\right] = \nu \frac{j_{||}}{n_e} + \partial_{||}\phi - \frac{1}{n_e}\partial_{||} p_e - 0.71 \partial_{||} T_e + \frac{m_e}{m_i}\left(\mathbf{V}_{E\times B} + \mathbf{b} v_{||i}\right) \cdot \nabla \frac{j_{||}}{n_e}$$

Boussinesq approximation

$$\omega = \nabla \cdot \left(\frac{n_0}{B^2} \nabla_\perp \phi\right)$$

SD1D

Finite electron mass, electromagnetic

Hermes: Getting started

https://github.com/boutproject/hermes

- Needs BOUT++ version 3.1
- SUNDIALS 2.6 or 2.7 for preconditioning
- PETSc for calculating n=0 electromagnetic fields

```
 _   _
| | | |
| |_| | ___ _ __ _ __ ___   ___  ___
|  _  |/ _ \ '__| '_ ` _ \ / _ \/ __|
| | | |  __/ |  | | | | | |  __/\__ \
\_| |_/\___|_|  |_| |_| |_|\___||___/
```

Hermes plasma edge simulation model. Uses BOUT++ framework, adds finite volume operators and neutral gas models.

This is Hermes-1, a cold ion drift-reduced model.

Hermes: Docker image

- Provides a reproducible environment (like a virtual machine or BSD jail)
- Preinstalled with PETSc, Sundials, Python, BOUT++ and Hermes

```
$ sudo docker pull boutproject/hermes:1849a42
```

```
$ mkdir shared
$ sudo docker run --rm -it  \
    -e DISPLAY -v $HOME/.Xauthority:/home/boutuser/.Xauthority --net=host \
    -v $PWD/shared:/home/boutuser/bout-img-shared \
    --cpuset-cpus=0-4 \
    boutproject/hermes:1849a42
```

Hermes examples

- Start with a transport simulation, no electric fields:

```
$ mpirun -np 16 ./hermes -d d3d
```

Note: Needs 16 cores due to branch cuts

- Turn on currents, drifts

```
j_diamag      = true
j_par         = true
```

- Reduce the output timestep (`TIMESTEP = 10`) and restart

```
$ mpirun -np 16 ./hermes -d d3d restart
```

Note: Recommend saving a copy of the restart files

See manual for output variables

Hermes on NERSC: Setup

```
shifterimg -v pull docker:boutproject/hermes:1849a42
```

```
> shifterimg images | grep hermes
edison    docker    READY    35a042dc67   2018-08-14T06:23:16 boutproject/hermes:1849a42
> salloc -N 1 -p debug -t 00:30:00 \
    --image=docker:boutproject/hermes:1849a42
> shifter bash
> cd $SCRATCH
> cp -r /home/boutuser/hermes .
> cp -r /home/boutuser/BOUT-dev-3.1/tools/pylib pylib-hermes
> exit
```

Hermes on NERSC

Run

```
> salloc -N 1 -p debug -t 00:30:00 \
    --image=docker:boutproject/hermes:1849a42

> cd $SCRATCH/hermes
> srun -n 16 shifter ./hermes-1 -d d3d
```

Analyse

```
> module load python/3.6-anaconda-5.2
> pip install netcdf4
> export PYTHONPATH=$SCRATCH/pylib-hermes:$PYTHONPATH
> cd $SCRATCH/hermes/d3d
> ipython
```