

M3D-C1 ZOOM Meeting

10/12/2020

Agenda

1. CS Issues
 1. GPU solve status... J. Chen
 2. Local systems
 3. Other systems
 4. NERSC Time
 5. Changes to github master since last meeting
 6. Restarting with a different number of planes...Lyons
 7. Request to replace $bf=f$ with $bfp=df/d(\phi)$
2. Physics Studies
 1. Status of first coupled M3D-C1/LP Simulation .. Lyons/Samulyak
 2. NSTX shot 134020
 3. Carbon Pellet Mitigation on NSTX
 4. RE Fluid Modeling of DIII-D shot 17704
 5. DIII-D shot 177053 with Argon – Chen Zhao
 6. Other?

GPU solve status -- Traverse

From Jin Chen: 10/12/30

1. Working with PETSc group and Sherry's group, we finally were able to build PETSc and Superlu_Dist for GPU nodes without OpemMP interference.
2. M3DC1 runs on TRAVERSE with its solvers running on GPU nodes
3. HDF5 writing crashes at the end of run with the following error
malloc(): unsorted double linked list corrupted

This error shows up even running the whole code on CPU nodes only.

But no error shows up when mumps was used. I have communicated this with Sherry and her team. I was told that there might be some problems on the superlu side. But they want me to debug it. Right now I have a problem running the X11 application from home, even though a VPN has been set up.

GPU solve status – Cori GPU

From Jin Chen: 10/12/20

1. Code works now with solvers running on GPU nodes
2. KSPSolve Timings for 2D real runs on CORI HASWELL, CORI KNL, and CORI GPU
HASWELL=9.7s
KNL=37.6s
GPU=35.3s
3. KPS Solve Timings for 3D real runs on CORI HASWELL, CORI KNL, and CORI GPU
HASWELL=117.15s
KNL=240.22s
GPU= 351.79

Local Systems

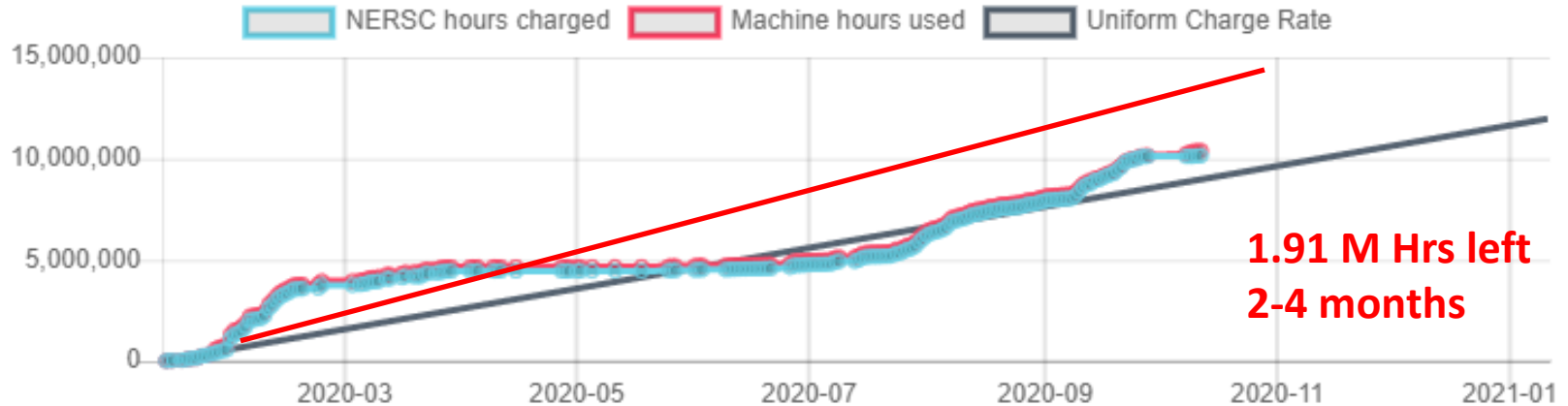
- PPPL centos7(10/10)
 - 5 regression tests PASSED on centos7:
 - “adapt” failed due to “corrupted double-linked list”
 - All tests pass using Sept30 centos7 (S. Seol)
- PPPL greene (10/10)
 - 4 regression tests PASSED
 - “adapt” failed due to “corrupted double linked list”
 - All 5 tests pass using Sept30 centos7 (S. Seol)
 - No batch file found for pellet
- EDDY (10/09)
 - All 6 regression tests PASSED on eddy
- TRAVERSE(10/05)
 - Code compiles
 - Regression test failed: split_smb not found in PATH

Other Systems

- Cori-KNL (10/9)
 - 6 regression tests passed on KNL
- Cori-Haswell j(10/9)
 - 5 regression tests passed
 - KPRAD_RESTART did not pass, but differences are very small in velocity variables. All magnetic and thermal good. Similar difference as Cori-KNL
- PERSEUS
 - All 6 regression tests PASSED on perseus (J. Chen, 9/04/20)
- MARCONI
 - All regression tests PASSED on MARCONI (J. Chen, 9/04/20)
- CORI GPU (10/9)
 - salloc: error: No architecture specified, cannot estimate job costs
 - salloc: Job submit/allocate failed: Unspecified error

NERSC Time

mp288



m3163

Closed for general use

- Should be enough mp288 time to last until new PU/PPPL computer arrives in fall – red line is linear usage until Nov 1
- John Mandrakes (DOE) has 45M hours to distribute, but got requests for 3x that. Has not yet decided. We can expect ~ 5 M hours.

Changes to github master since last meeting

- S. Seol
 - 10/05/20: changes in m3dc1_scorec – 2nd order adjacency and debugging on adaptation
 - 10/08/20: a routine to remove wedges btw planes added
- S. Jardin
 - 10/09/20: Phase 1 of changing f to fp
 - 10/10/20: a small tweak to fp calculation
- N. Ferraro
 - 10/06/20: Removed debug statement

Jump in KE when restarting with increased planes

Brendan showed some slides on 9/28/20 of how there is a glitch in the kinetic energy when he restarts with additional planes.

From Brendan: 10/12/20

With cori back up, I restarted a case that had 8 planes, once with the same 8 planes and again with 32, both from output slice 40 and then output every time step:

8-plane: /global/cscratch1/sd/blyons/C1_35007033/

32-plane: /global/cscratch1/sd/blyons/C1_35007045/

The deviation is present at output slice 40, so it's in the mapping of the 8 Hermite cubics to 32 Hermite cubics. At the original plane locations (every 45 degrees), the fields are exactly the same, but they start varying in between, peaking halfway in between. I've attached a movie showing the difference in 'phi' at every integer toroidal angle.

I've submitted another run with 16 planes, to see if doubling will work properly. I'd appreciate your thoughts.

Request to replace $\mathbf{bf} = \mathbf{f}$ with $\mathbf{bfp} = d\mathbf{f}/d\varphi$

$$\mathbf{A} = R^2 \nabla \varphi \times \nabla f + \psi \nabla \varphi - F_0 \ln R \hat{Z}$$

$$\mathbf{B} = \nabla \psi \times \nabla \varphi - \nabla_{\perp} f' + F \nabla \varphi$$

$$F \equiv F_0 + R^2 \nabla \bullet \nabla_{\perp} f \quad (\text{note: } f' \equiv \partial f / \partial \varphi)$$

Presently,

$$\dot{F} = \dots$$

$$R^2 \nabla \bullet \nabla_{\perp} f = F - F_0$$

Proposed (by Yao) to eliminate one derivative on f

$$\dot{F} = \dots$$

$$R^2 \nabla \bullet \nabla_{\perp} f' = F'$$

PHASE I: Just solve for fp to compare with f' but don't use it

1. In M3Dmodules.f90
 1. Add: `type(field_type) :: bfp_field(0:1), bfp_ext`
 2. Remove: `integer, parameter :: bf_mat_rhs_index = 39` (never used)
 3. Add: `integer, parameter :: bfp_mat_rhs_index = 39`
2. In module newvar_mod
 1. `integer, parameter :: NV_IP_MATRIX = 8`
 2. `type(newvar_matrix) :: mass_mat_rhs_bfp`
3. In subroutine set_newvar_indices
 1. `call set_matrix_index(mass_mat_rhs_bfp%mat, bfp_mat_rhs_index)`
4. In subroutine create_newvar_matrices
 1. `if(ifbound.eq.1) then`
 2. `call create_newvar_mat(mass_mat_rhs_bfp, NV_DC_BOUND, NV_IP_MATRIX,0)`
 3. `else if (ifbound.eq.2) then`
 4. `call create_newvar_mat(mass_mat_rhs_bfp, NV_NM_BOUND, NV_IP_MATRIX,0)`
 5. `endif`
5. In subroutine create_newvar_matrix
 1. `case(NV_IP_MATRIX)`
 2. `Temp(:, :, 1, 1) = intxx2(mu79(:, :, OP_1), nu79(:, :, OP_DP))`
6. In subroutine space
 1. `call create_field(bfp_field(0))`
 2. `call create_field(bfp_field(1))`
 3. `call create_field(bfp_ext)`

7. In subroutine derived_quantities

! Toroidal derivative of vector potential stream function

if((i3d.eq.1 .or. ifout.eq.1) .and. numvar.ge.2) then

if(myrank.eq.0 .and. iprint.ge.2) print *, " fp", ilin

call solve_newvar1(bf_mat_lhs,bfp_field(ilin),mass_mat_rhs_bfp, &

bz_field(ilin), bf_field(ilin))

endif

8. . In output.f90:

call output_field: fp_plasma, fp_plasma_i, fp, fp_i

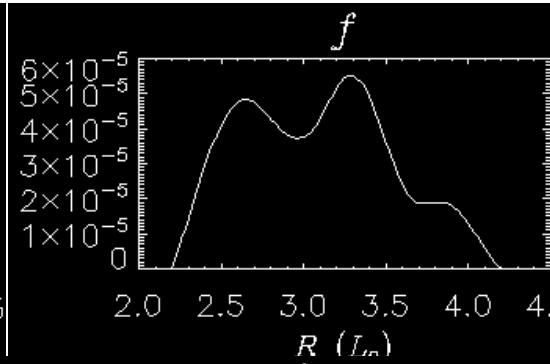
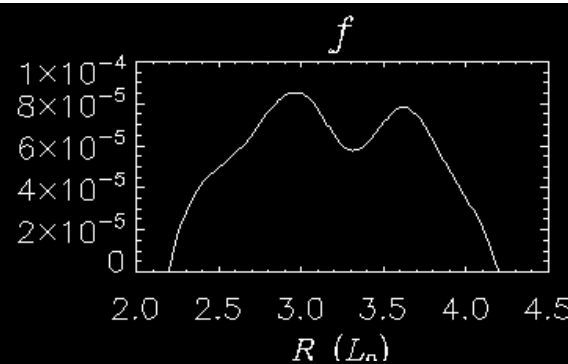
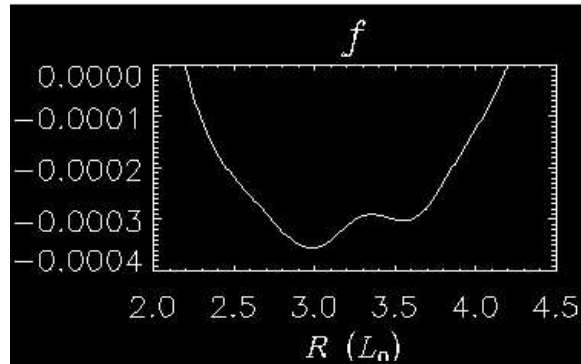
Status: Phase 1 implemented and committed

$\Phi = 0^\circ$

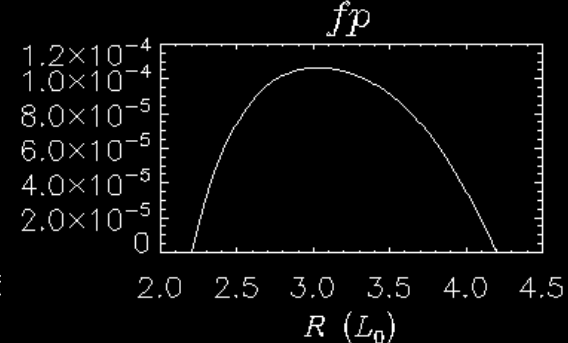
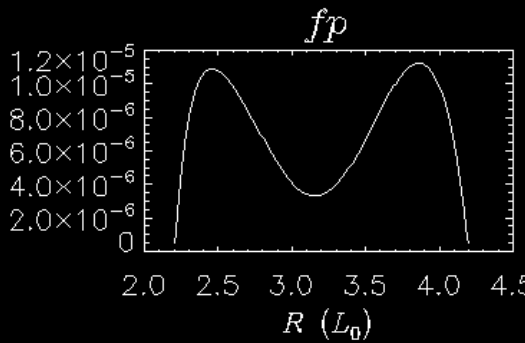
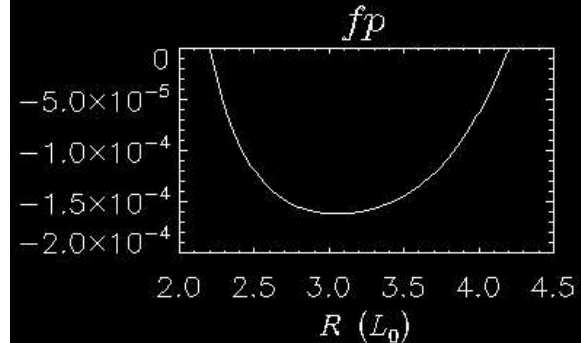
$\Phi = 10^\circ$

$\Phi = 20^\circ$

f
opt=11



Fp
opt=1



I restarted a 3D NL calculation and ran 1 time step with the new coding

The top and bottom curves should be identical but they are not. There is an error I would like someone else to look at the coding and do tests

Phase II: replace f with fp in calculations

1. In subroutine derived_quantities

789 ! vector potential stream function

790 **if(imp_bf.eq.0 .or. ilin.eq.0 .or. ntime.eq.0) then**

← **Remove**

791 if((i3d.eq.1 .or. ifout.eq.1) .and. numvar.ge.2) then

792 if(myrank.eq.0 .and. iprint.ge.2) print *, " f", ilin

793 if((ilin.eq.0 .and. eqsubtract.eq.1) .or. eqsubtract.eq.0) then

794 if(itor.eq.0) then

795 temp = bzero

796 else

797 temp = bzero*rzero

798 end if

799 call add(bz_field(ilin),-temp)

800 endif

801 call solve_newvar1(bf_mat_lhs,bf_field(ilin),mass_mat_rhs_bf, &

802 bz_field(ilin), bf_field(ilin))

803 if((ilin.eq.0 .and. eqsubtract.eq.1) .or. eqsubtract.eq.0) call add(bz_field(ilin), temp)

804 endif

805 **end if**

2. In m3dc1_nint.f90: Change

call eval_ops(itri,bf_ext,bfx79,rfac) → call eval_ops(itri,bfp_ext,bfpx79,rfac)

call eval_ops(itri,bf_field(1),bf179,rfac) → call eval_ops(itri,bfp_field(1),bfp179,rfac)

call eval_ops(itri,bf_field(0),bf079) → call eval_ops(itri,bfp_field(0),bfp079)

3. In time_step.f90, time_step_split.f90, time_step_unsplit.f90: Change

bf_field → bfp_field everywhere

4. In restart_hdf5.f90

add h5r_read_field for bfp_field, bbfp_ext **backward compatibility?**

5. In particle.f90

Need Help. Can we replace bf_field with bfp_field? What else needs to be changed?

6. rmp.f90

Need Help:

7. In lundef_t.f90

In bf_equation_lin, change:

```
ssterm(:,bz_g) = intx2(trial(:, :, OP_1), lin(:, OP_1))
```

With

```
ssterm(:,bz_g) = intx2(trial(:, :, OP_1), lin(:, OP_DP))
```

In bf_equation_nolin, just replace coding with

```
r4term = 0
```

```
return
```

bft79	bf079	bf179	bftx79
auxiliary_fields		auxiliary_fields.f90	
bootstrap.f90	bootstrap.f90		
diagnostics.f90			diagnostics.f90
electric_field.f90	electric_field.f90	electric_field.f90	electric_field.f90
lundef_t.f90	lundef_t.f90	lundef_t.f90	lundef_t.f90
m3dc1_nint.f90	m3dc1_nint.f90	m3dc1_nint.f90	m3dc1_nint.f90
metricterms_new.f90		metricterms_new.f90	metricterms_new.f90
temperature_plots.f90		temperature_plots.f90	temperature_plots.f90
transport.f90			transport.f90

init_basicj.f90

parallel_heat_flux.f90

bft79(:,OP_DZP) → bfpt79(:,OP_DZ)
 bft79(:,OP_DRP) → bfpt79(:,OP_DR)
 bft79(:,OP_DZPP) → bfpt79(:,OP_DZP)
 bft79(:,OP_DRPP) → bfpt79(:,OP_DRP)
 bft79(:,OP_DRRP) → bfpt79(:,OP_DRR)
 bft79(:,OP_DZZP) → bfpt79(:,OP_DZZ)
 bft79(:,OP_DRZP) → bfpt79(:,OP_DRZ)
 bft79(:,OP_LPP) → bft79(:,OP_LP)
 bft79(:,OP_GSP) → bft79(:,OP_GS)

Etc for bf079, bf179, bftx79.

Make these substitutions in
all of the above routines

metricterms_new.f90

v1psif(x, x,f)	b1psifn1(x,x,f,x)
v1bf(x,x,f)	b1psifn2(x,x,f,x)
v1huf(x,x,f)	b1bfn1(x,x,f,x)
v1hvf(x,x,f)	b1bfn2(x,x,f,x)
v1hchif(x,x,f)	b1psifd1(x,x,f,x)
v2psif1(x,x,f)	b1psifd2(x,x,f,x)
v2psif2(x,x,f)	b1bfd1(x,x,f,x)
v2bf(x,x,f)	b1bfd2(x,x,f,x)
v2ff(x,f,f)	b2feta(x,f,x)
v2huf(x,x,f)	b2jrefeta(x,x,f,x,x)
v2hvf(x,x,f)	b2fv(x,f,x)
v2hchif(x,x,f)	b2psifn(x,x,f,x)
v3psif(x,x,f)	b2bfn(x,x,f,x)
v3bf(x,x,f)	b2psifd(x,x,f,x)
v3huf(x,x,f)	b2bfd(x,x,f,x)
v3hvf(x,x,f)	b3psifeta(x,x,f,x,x)
v3hchif(x,x,f)	b3bfeta(x,x,f,x,x)
b1jrefeta(x,x,f,x,x)	b3ffeta(x,f,f,x,x)
B1feta(x,f,x)	b3pefd(x,x,f,x)
b1fu(x,f,x)	tepsifkappar(x,x,f,x,x,x)
b1fv(x,f,x)	tebfkappar(x,x,f,x,x,x)
b1fchi(x,f,x)	teffkappar(x,f,f,x,x,x)

bootstrap.f90

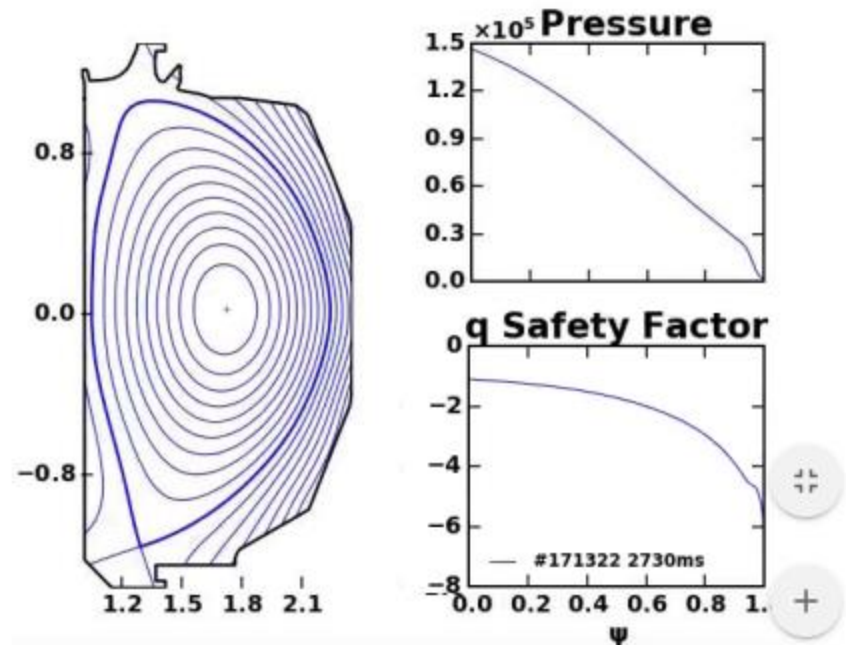
bs_b1psibf(x,x,x,f)
bs_b2psibf(x,x,x,f)

In each of these routines,
remove the last P in OP_XP
for only the argument
marked f

Status of First Coupled M3D-C1 / LP Simulation

- **Iterate independent simulations of MHD and LP codes**
 - Run pellet injection in MHD code with analytic, Parks ablation formula
 - Send plasma states along pellet path to LP code to compute ablation rate at each point
 - Rerun MHD codes with LP ablation rates
 - Iterate between codes until convergence
- **Test case for DIII-D modeling**
 - 1 mm Ne pellet using extruder parameters
 - 160606, standard case for SPI modeling
 - 171322, super-H target for upcoming small-pellet ablation experiment
 - Latter will be used for predict-first of experiment

DIII-D 171322 @ 2730 ms



8/10/20 – proposed

10/5/20 – Brendan sent data from a 2D run

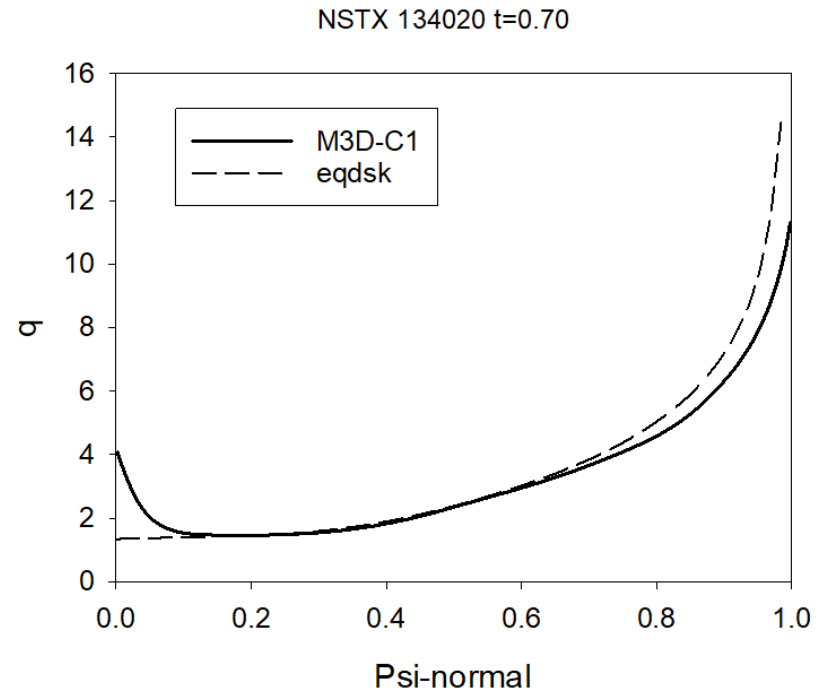
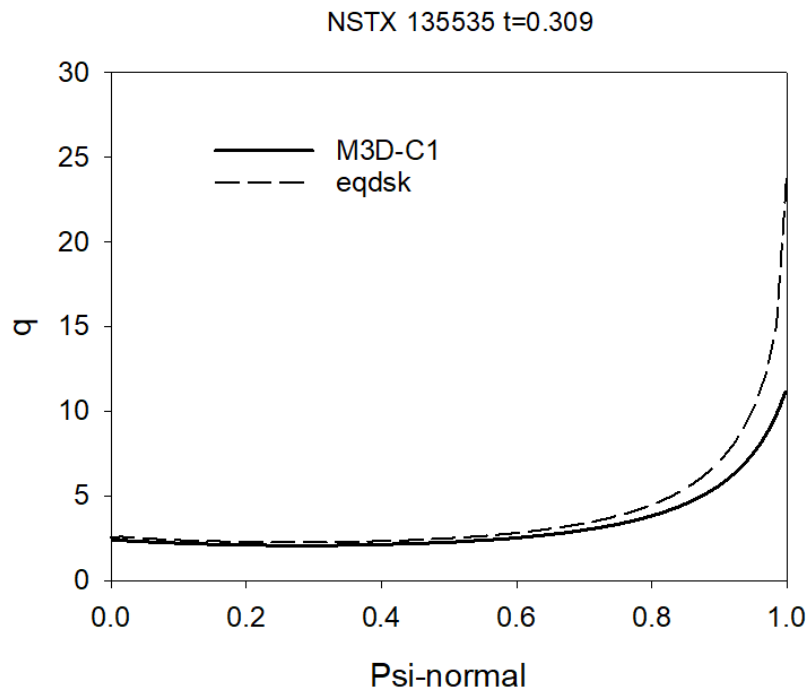
10/7/20 – Roman requested more concise data from around pellet vs time

10/7/20 – need to decide where to evaluate the electron density, temperature, and magnetic field. In the m3dc1 pellet cloud, or upstream.

NSTX shot 134020

Fast ion transport with coupled kink and tearing modes

J. Yang, C. Liu



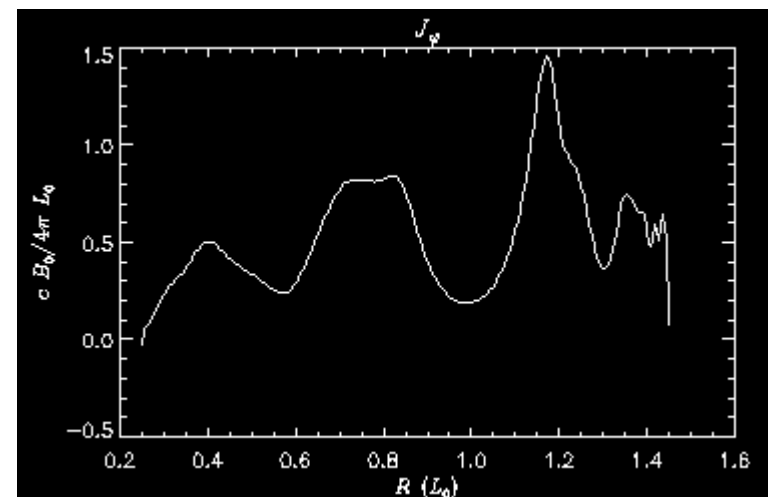
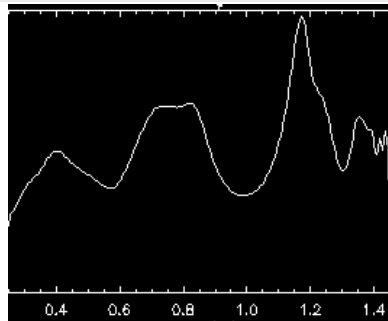
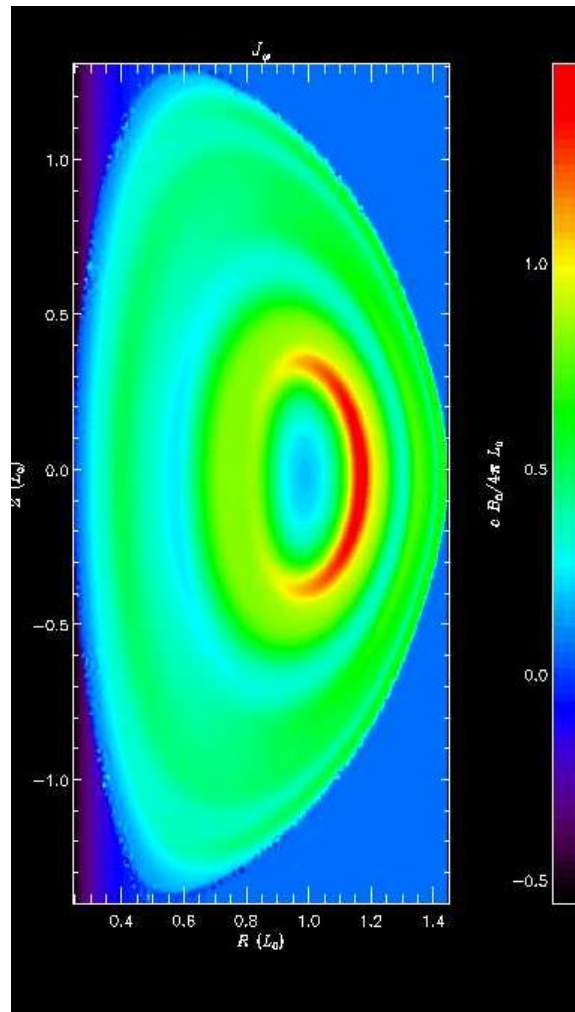
q-profile as calculated from M3D-C1 does not agree with that from geqdsk at the origin!

NSTX shot 134020 (cont)

Fast ion transport with
coupled kink and tearing
modes

J. Yang, C. Liu

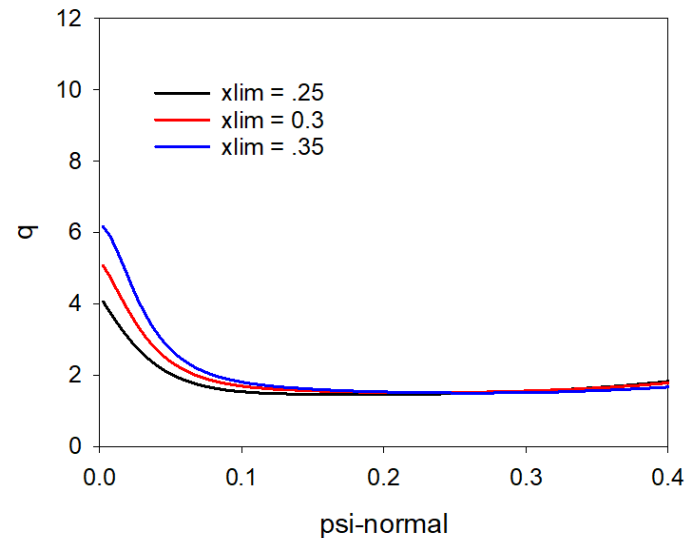
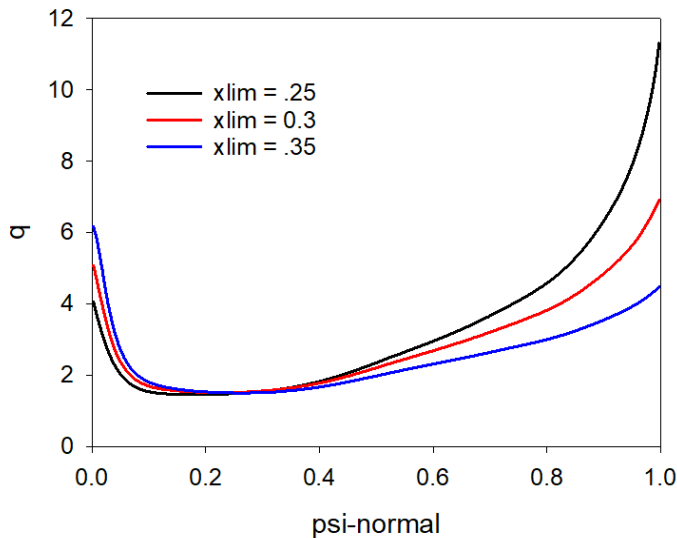
- Plasma limited by mesh boundary!
- Mid-plane current density very jagged



NSTX shot 134020 (cont)

Fast ion transport with coupled kink and tearing modes

J. Yang, C. Liu



Central value of q depends on limiter position. This geqdsk had a left boundary of 0.185 instead of 0.01 as in earlier geqdsk files. Also, mesh needs to be extended.

Meeting held 10/6 to discuss. Devon will provide coil and vessel currents and we will re-computer the free boundary equilibrium

Carbon Pellet Mitigation on NSTX

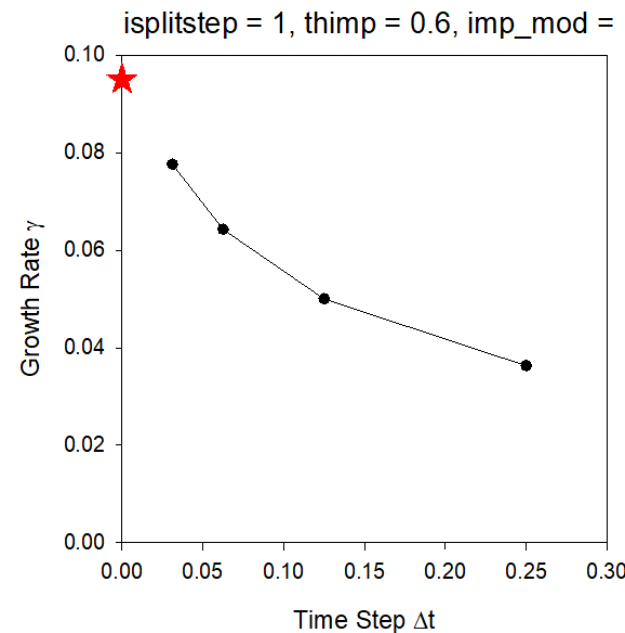
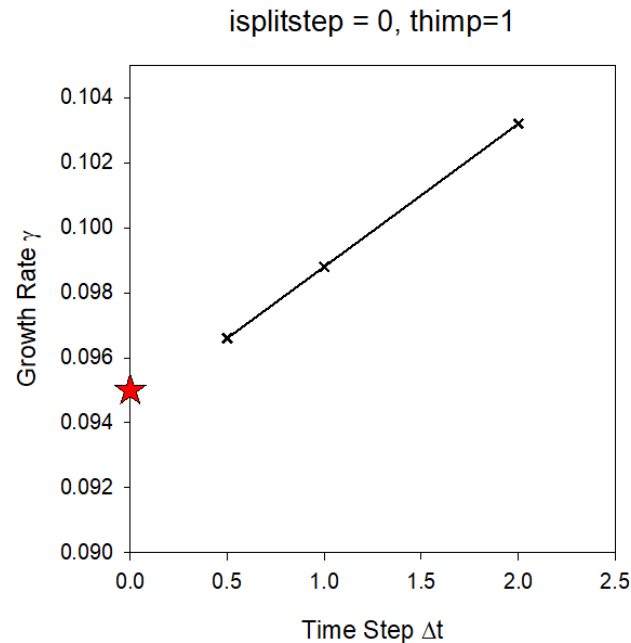
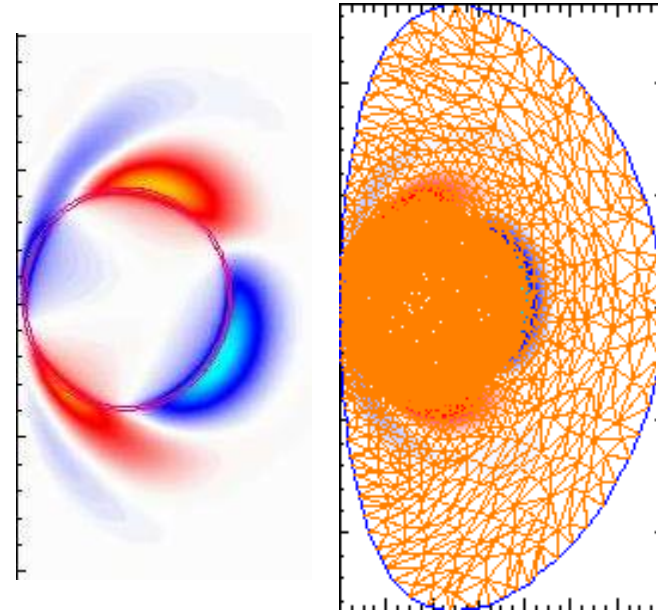
C. Clauser and S. Jardin met with Roger Raman on 10/7 to discuss Cesar's Carbon mitigation runs

- Roger wants to model a carbon powder filled shell pellet, that deposits the carbon uniformly when the pellet reaches the plasma center
- To do long time simulations of the current quench, Cesar wants to reduce the number of toroidal planes once the impurity is mixed. Is this possible?
- How best to model the carbon deposition from the shell pellet?

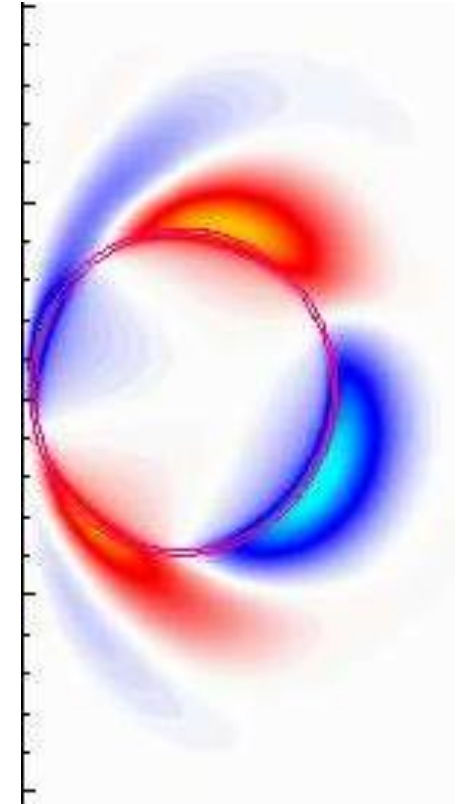
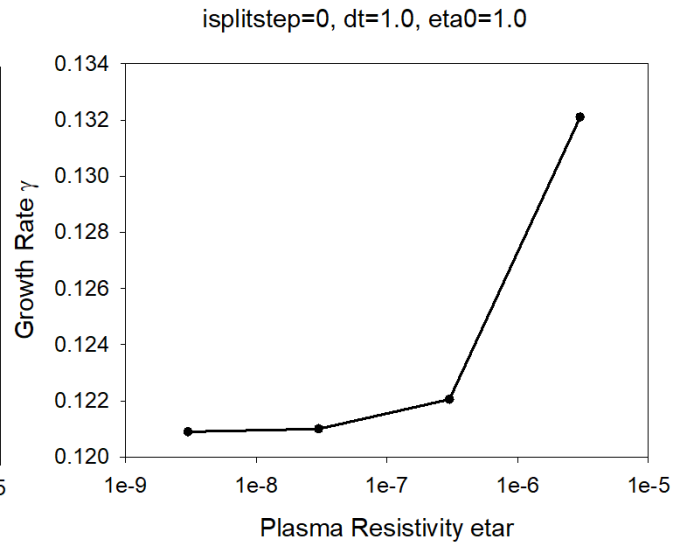
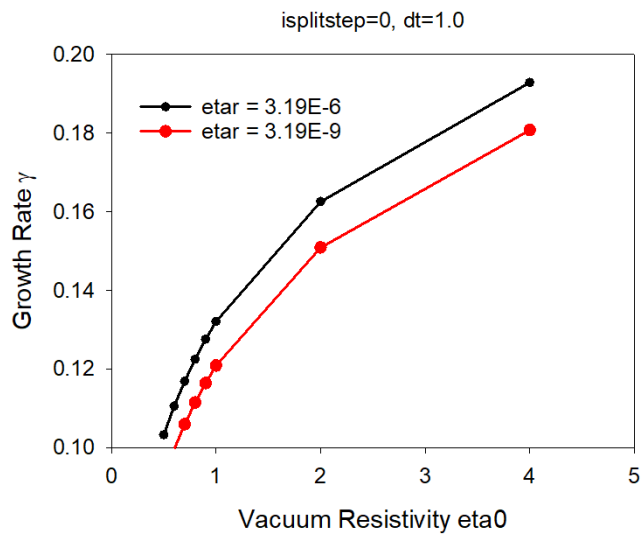
RE Fluid Modeling of DIII-D shot 177040

Carlos Paz-Soldan email 9/14/20 (not he had this in his talk last week!!)

- 1) Dependence of RE current carriers (or not) on saturated mode amplitude (dB/B)?
- 2) Dependence of saturated dB/B on absolute B&I (fixed a , all $q=2$). —> ITER extrapolation
- 3) Dependence of saturated dB/B on absolute I (fixed B , smaller a , all $q=2$). —> DIII-D comparison

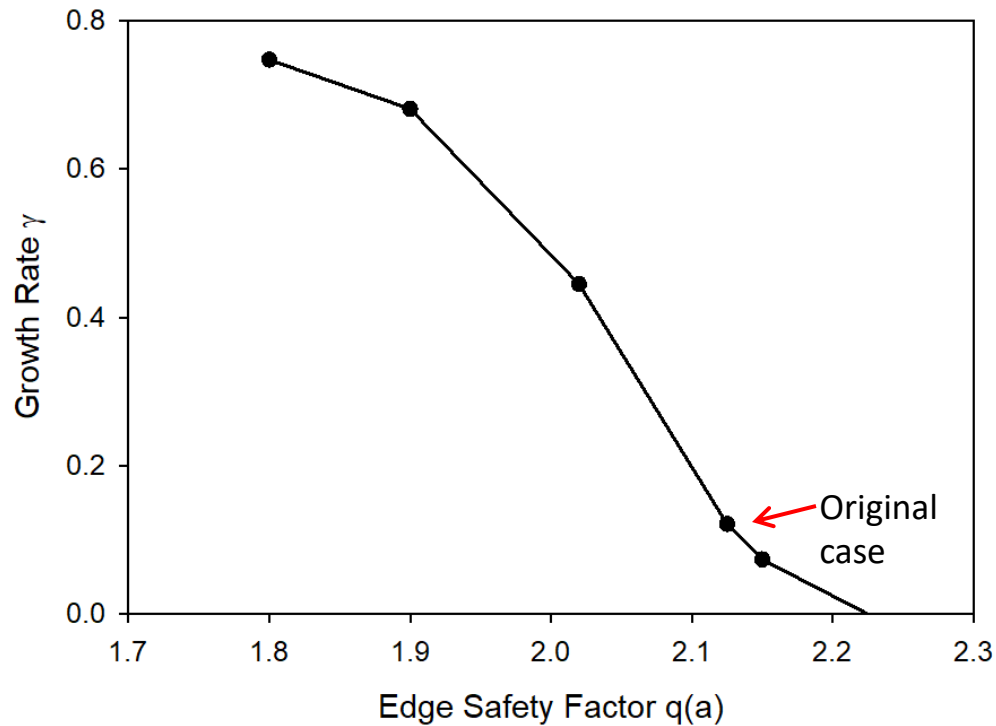


Dependence of Growth Rate on Plasma and Vacuum resistivity



Growth Rate vs $q(a)$

Bateman Scaling: isplitstep=0, dt=1, eta0=1, etar=3.E-9

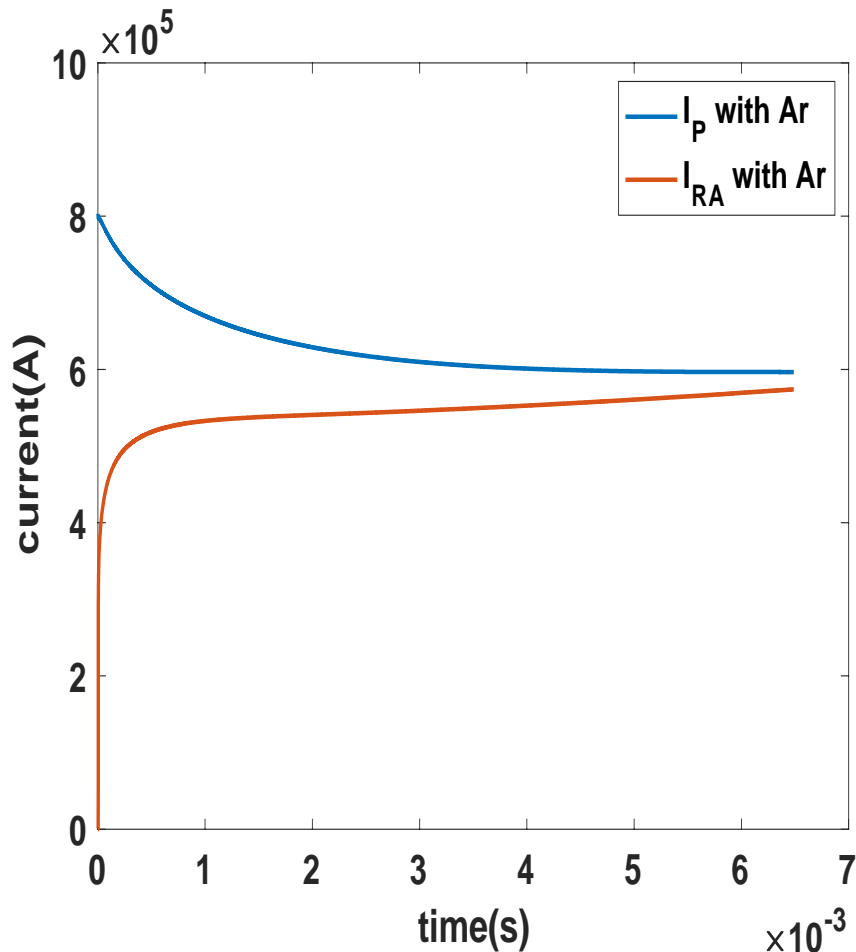


Batemanscale modifies $F(1)$ but keeps p' and FF' fixed.

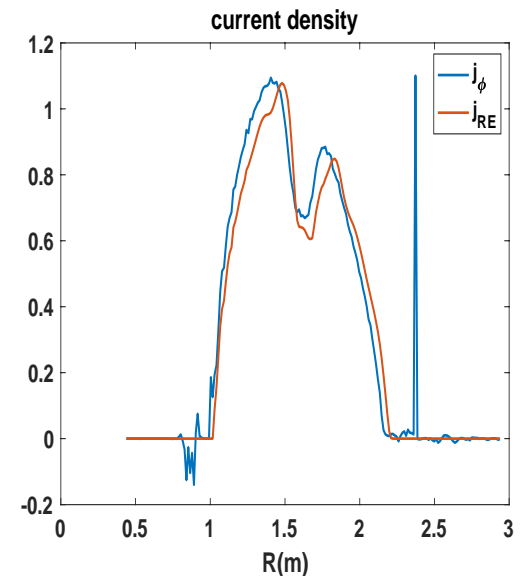
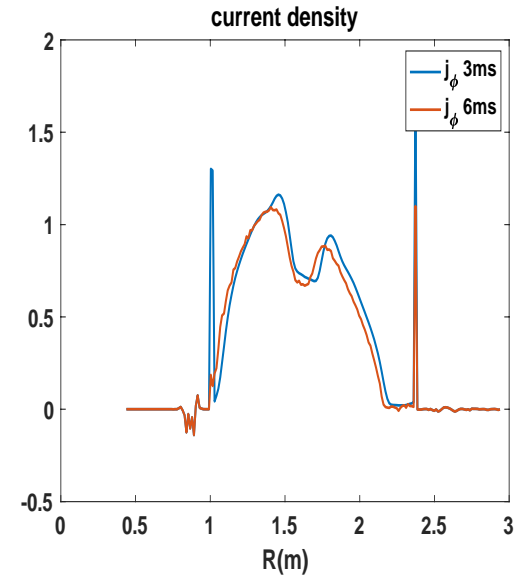
We should ask MARS to make a similar plot.

177053 case with Argon

10/12/20



- The growth rate of plasma current was at about 6ms larger than 0 and both the runaway current and the plasma current increased.
- The reason is that the plasma current shifted towards so that at last the runaway current at the low field side was higher than plasma current and caused an increasing of plasma current

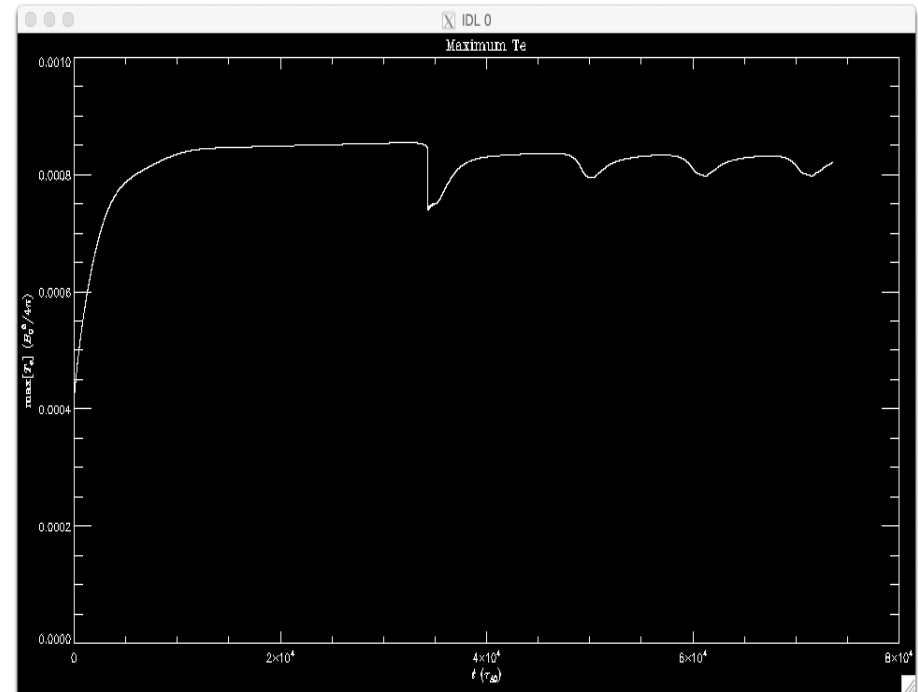
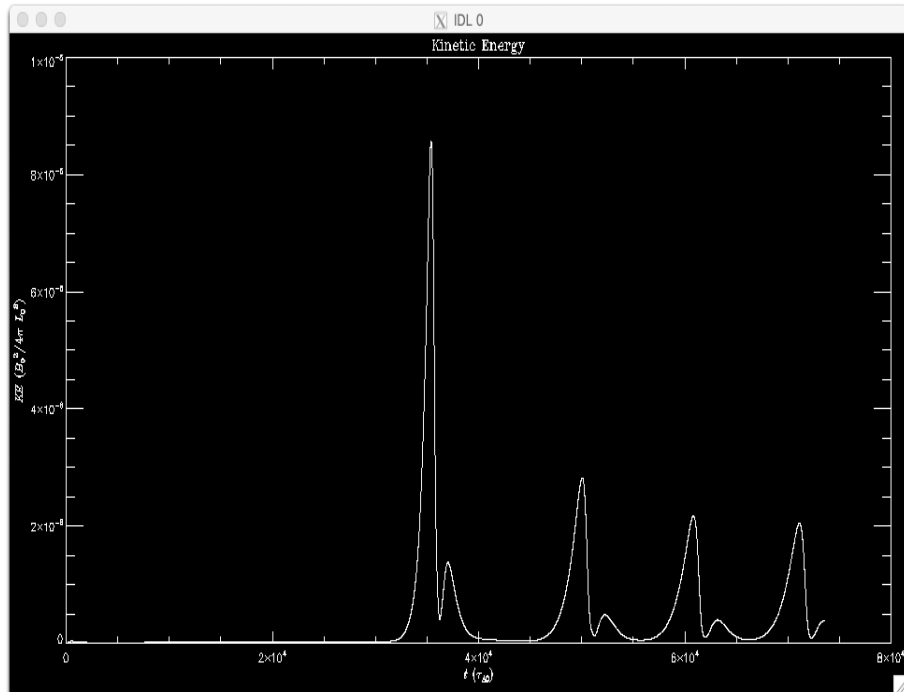


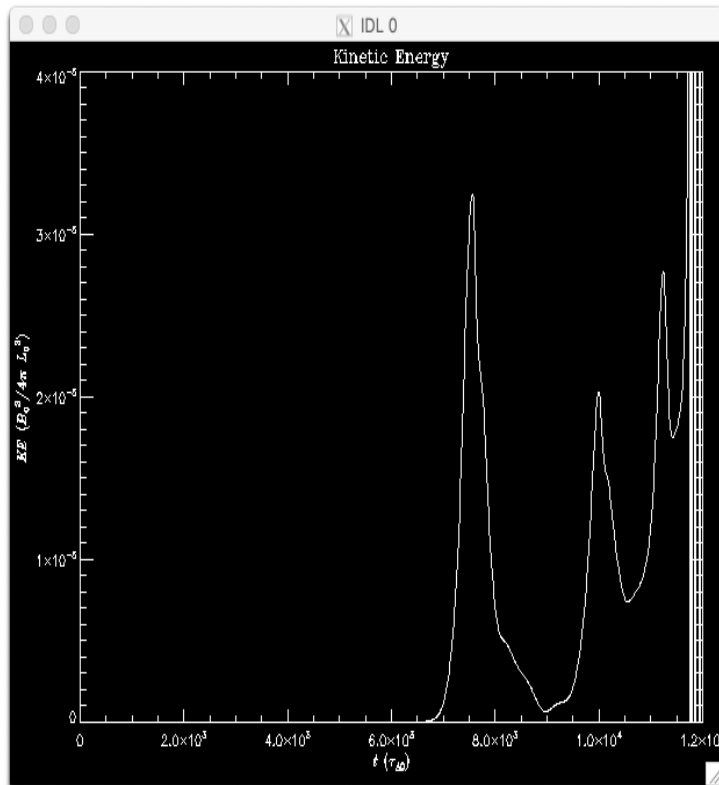
This only happened in DIII-D geometry, not in cylinder.
If no runaways, the plasma current also did not shift

Sawtooth case with Isabel's Equilibrium

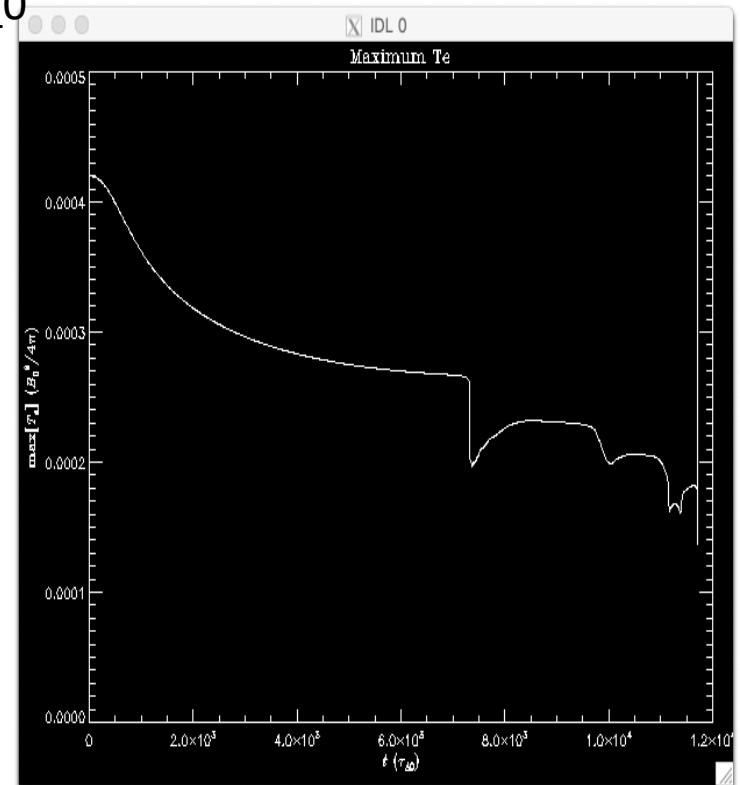
10/12/20

No runaways





$$I_{RA} = 0.8 I_0$$



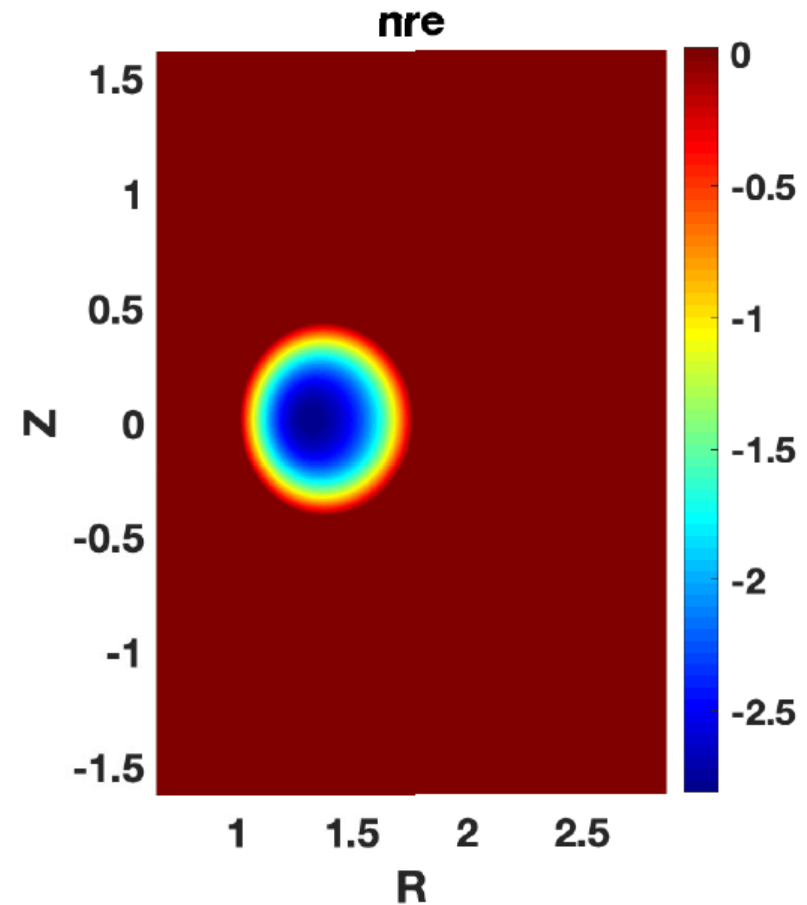
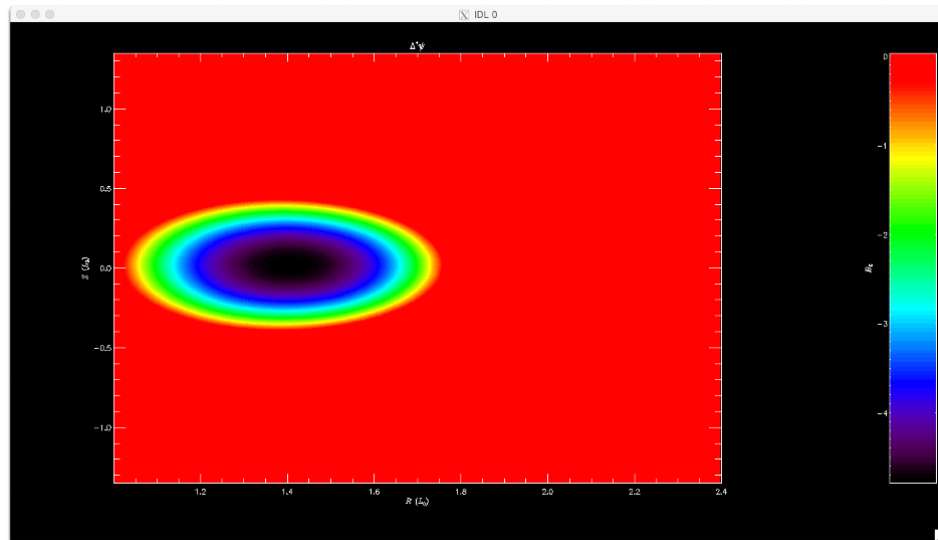
The temperature was dropped when the 1/1 mode growing, and since the runaway current do not have ohmic heating effect, the temperature would not come back to the previous value when the 1/1 mode disappear, the resistivity become larger, and because the resistive kink mode do not have the resistivity correction effect, the growth rate will become higher, and finally the case crashed when the temperature was too low.

I think after the first sawtooth phase, the current drive should disappear by the large runaway current effect according to Cai & Fu paper's opinion.

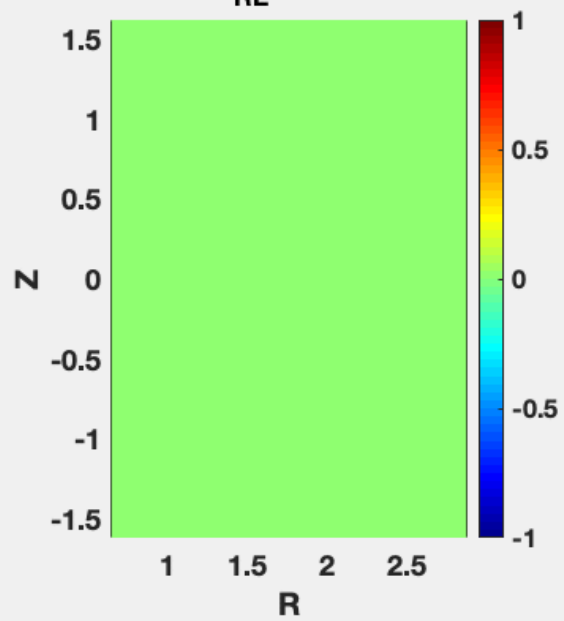
That's All I have

Anything Else ?

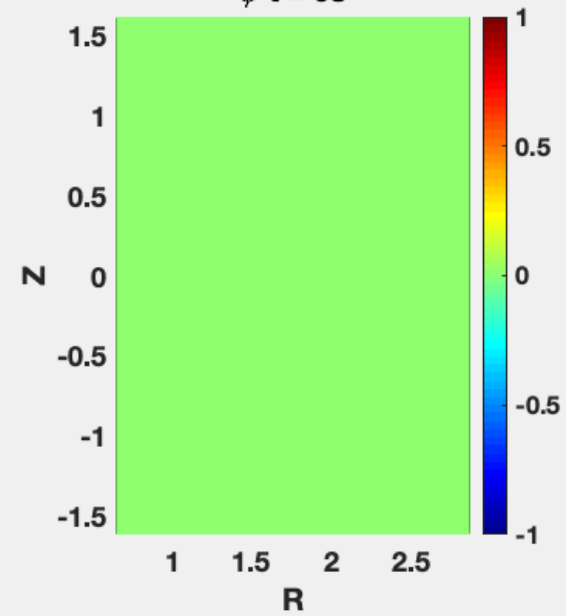
9/18/20 Chen Zhao: Shot 177040 with RE



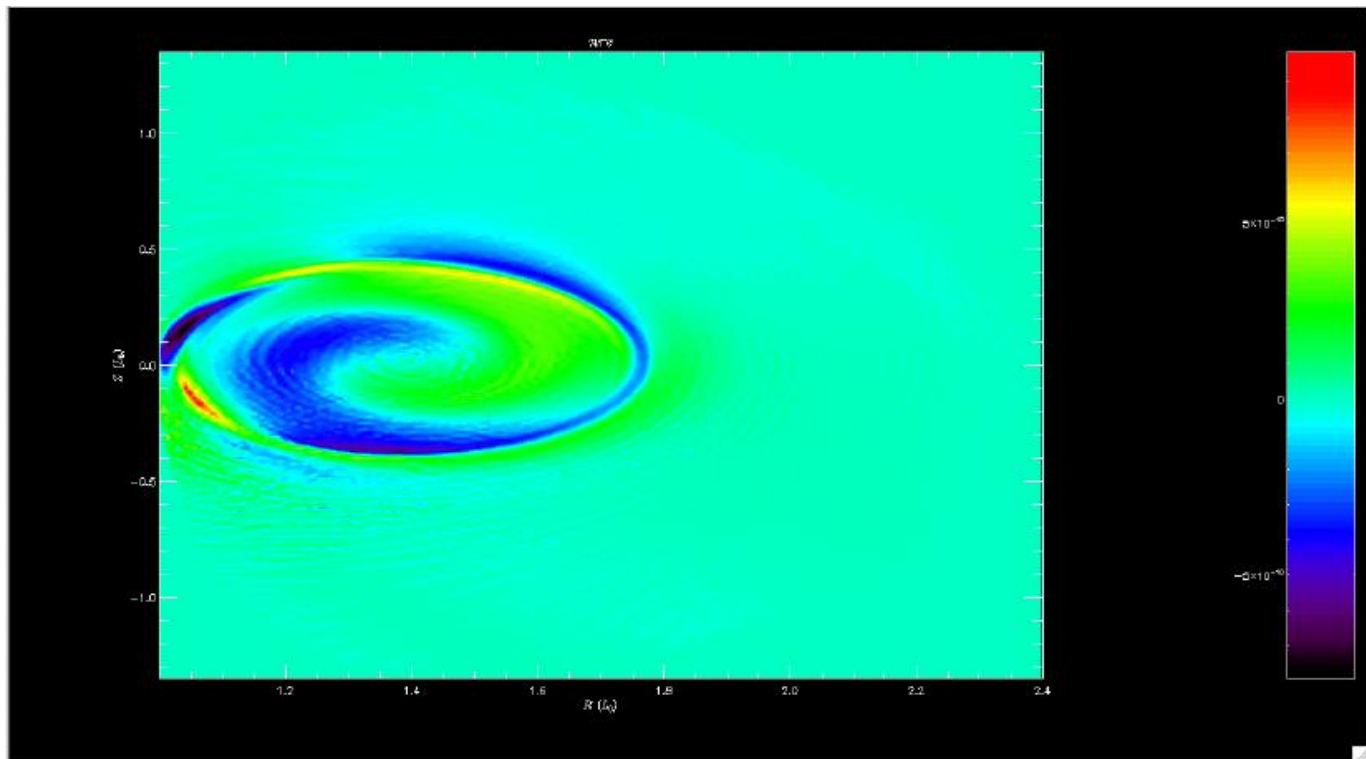
n_{RE} $t = 0s$



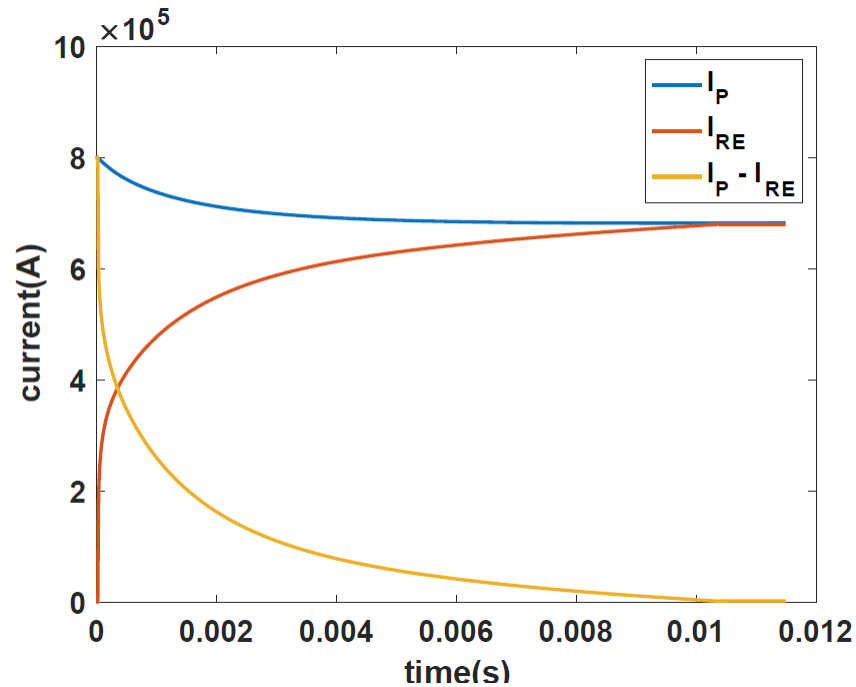
ψ $t = 0s$



With ExB drift terms removed



DIII-D Shot 177053 (with RE sources) – Chen Zhao

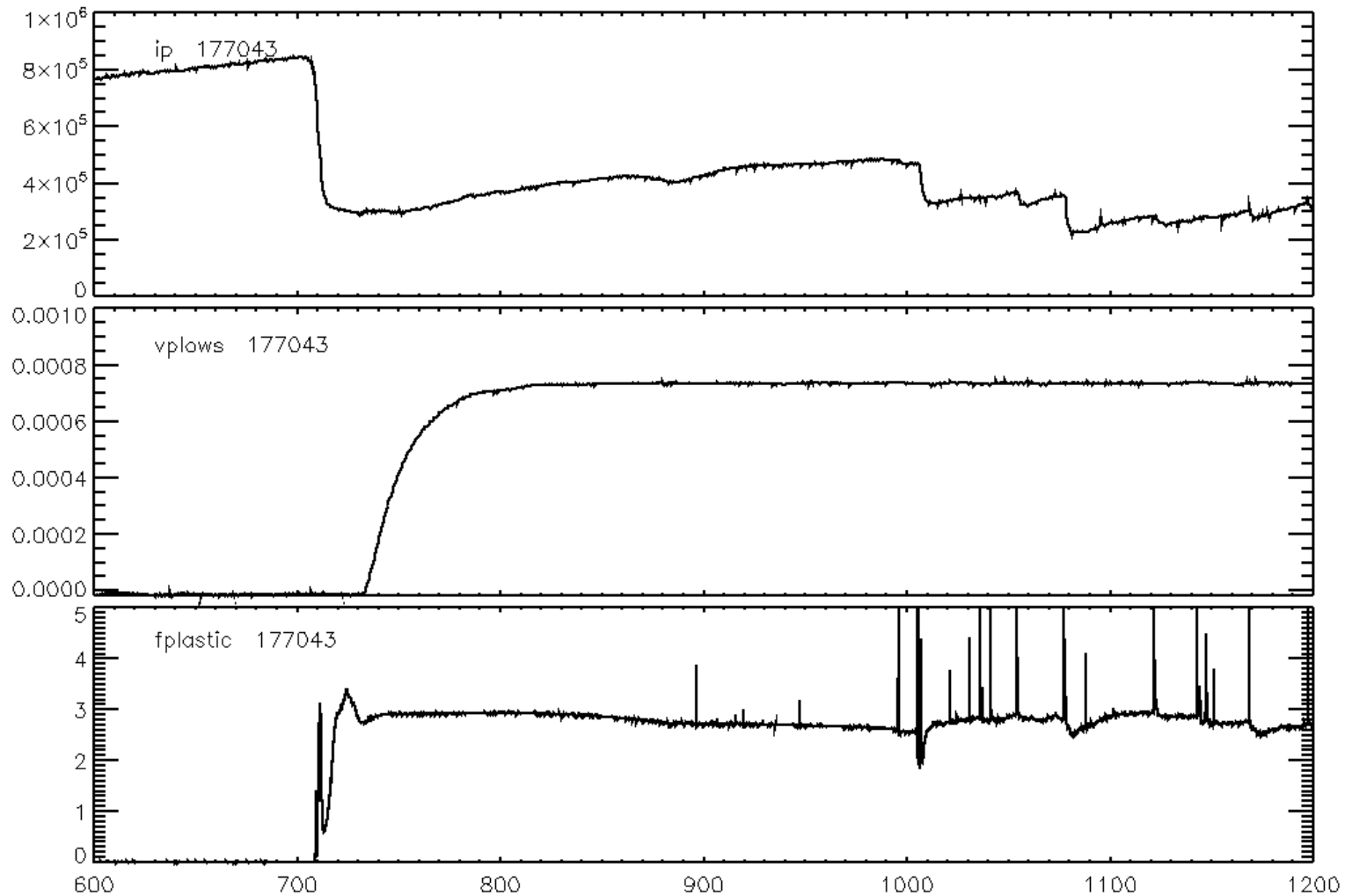


Code changes now committed to GIT

NEXT Steps:

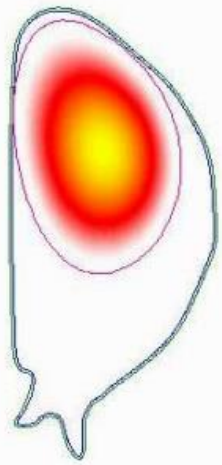
1. Study of sensitivity to κ_{\perp} ?
2. Comparison with shot results
3. Presentation at DIII-D disruption meeting

177053 Exp. Traces (Lyons 06/08/20)



Test of Boozer Theory for Cold VDE (Clauser)

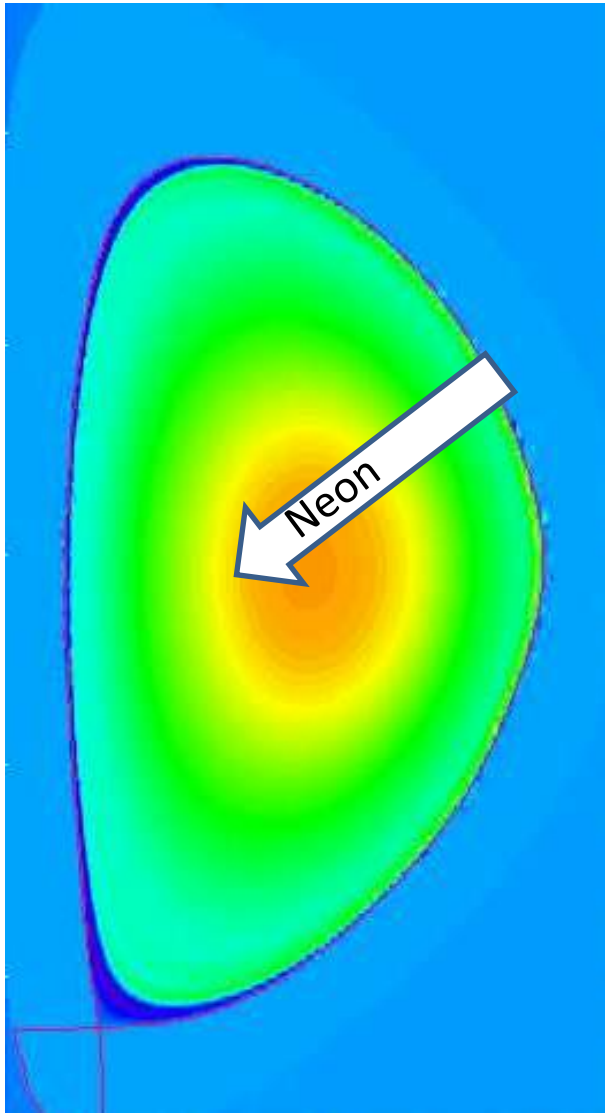
- Boozer's analytic theory that if ITER suffers a disruption on the mid-plane, such that the current decreases to $I = 0.83 I_0$, vertical stability will be lost, even for an ideally conducting wall.
- Cesar has tried to verify this, and finds the plasma is still VDE stable with $I = 0.3 I_0$
- Difference is likely the wall model, Cesar to confirm.



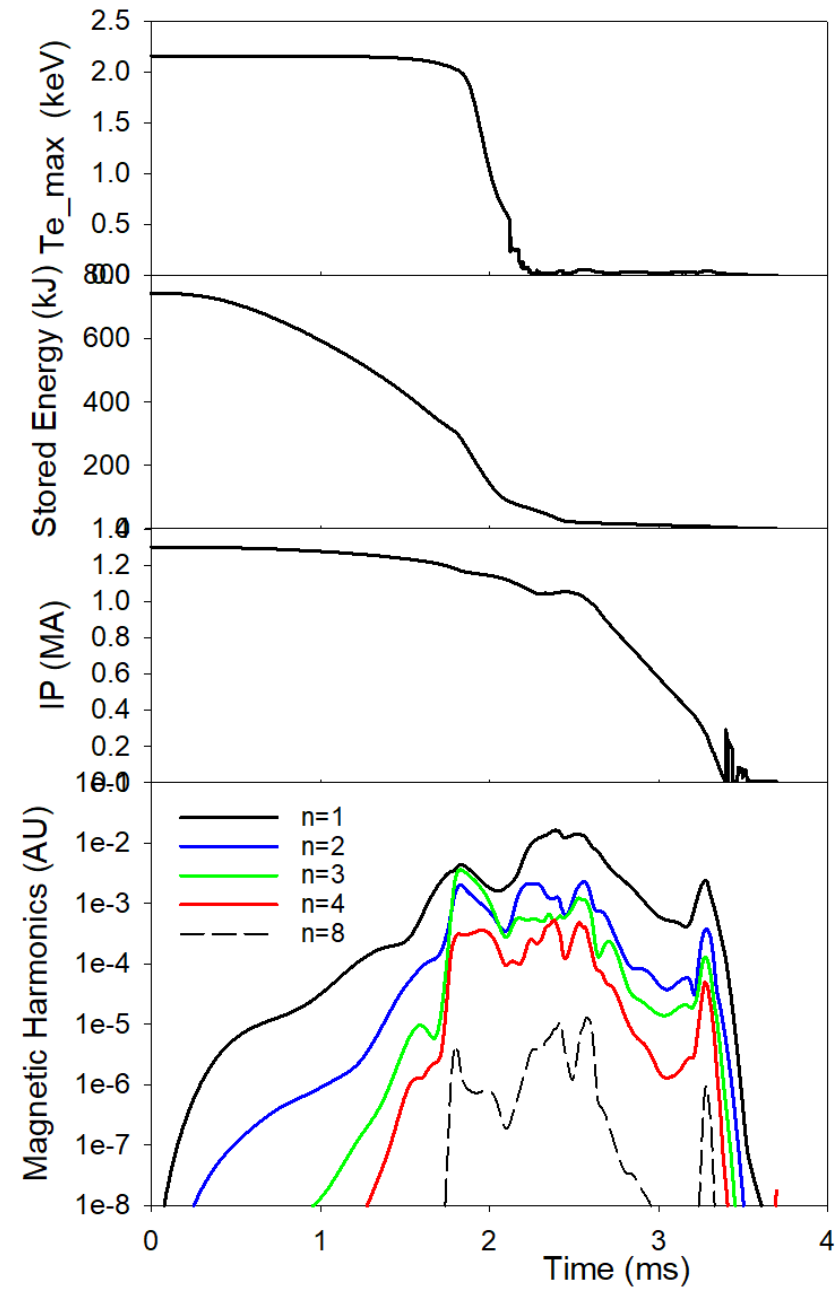
Vs.

Boozer, "Halo currents and vertical displacements after ITER disruptions", Phys. Plasmas 26, 114501 (2019)

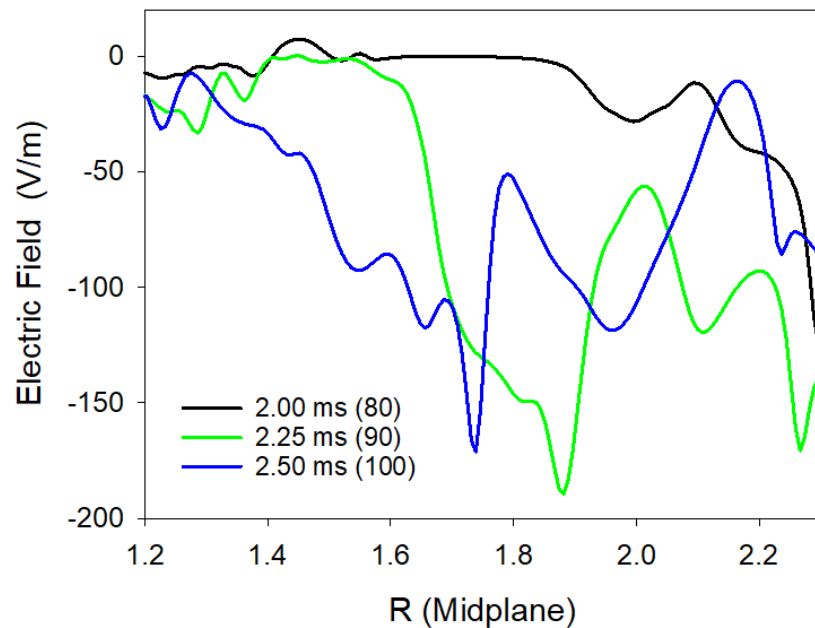
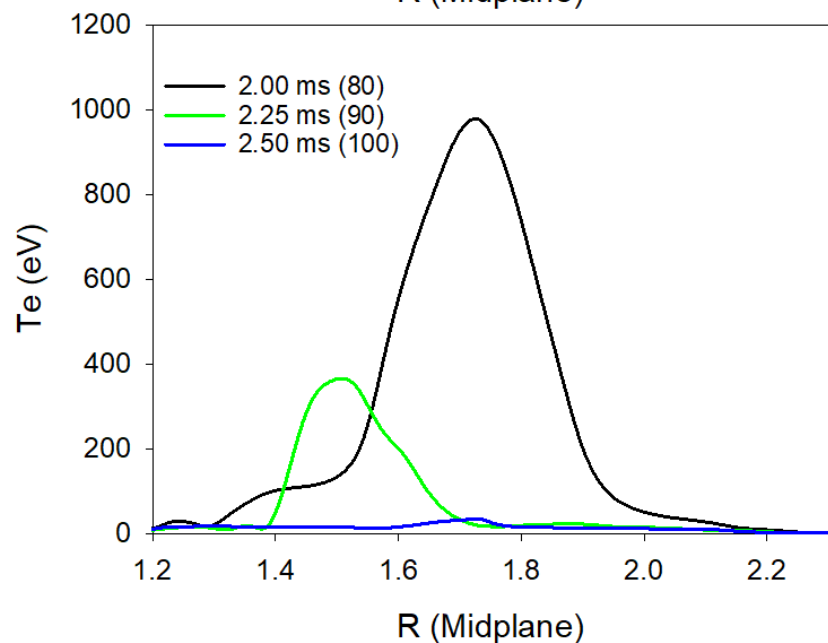
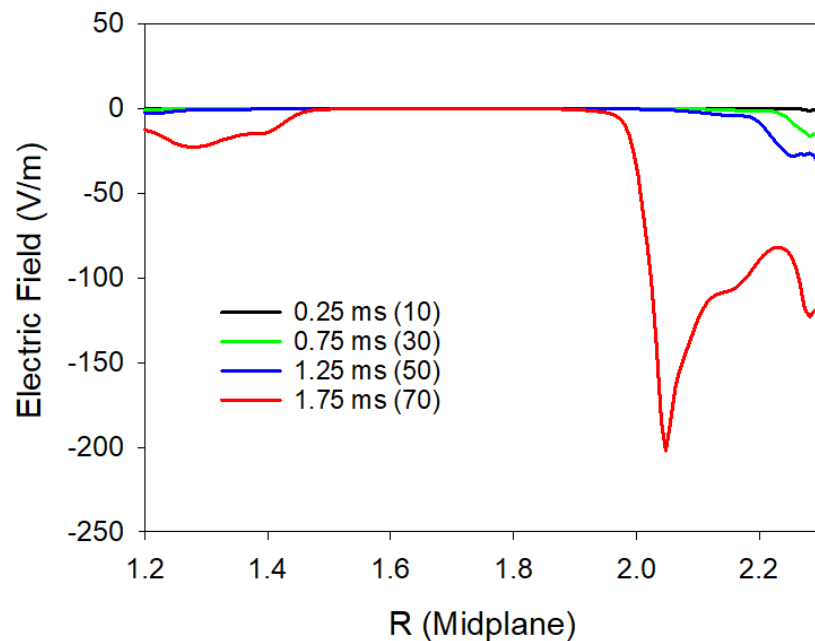
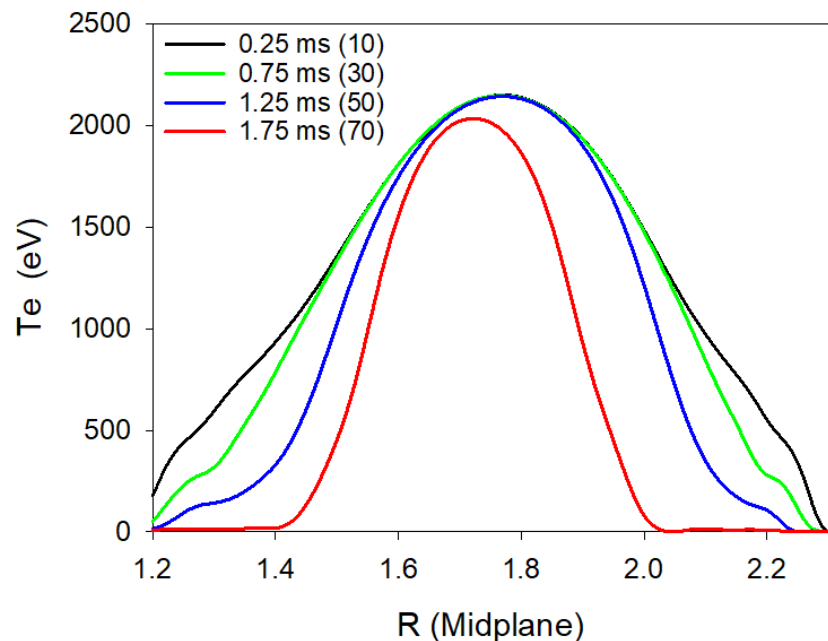
DIII-D Neon pellet mitigation simulation (for KORC)



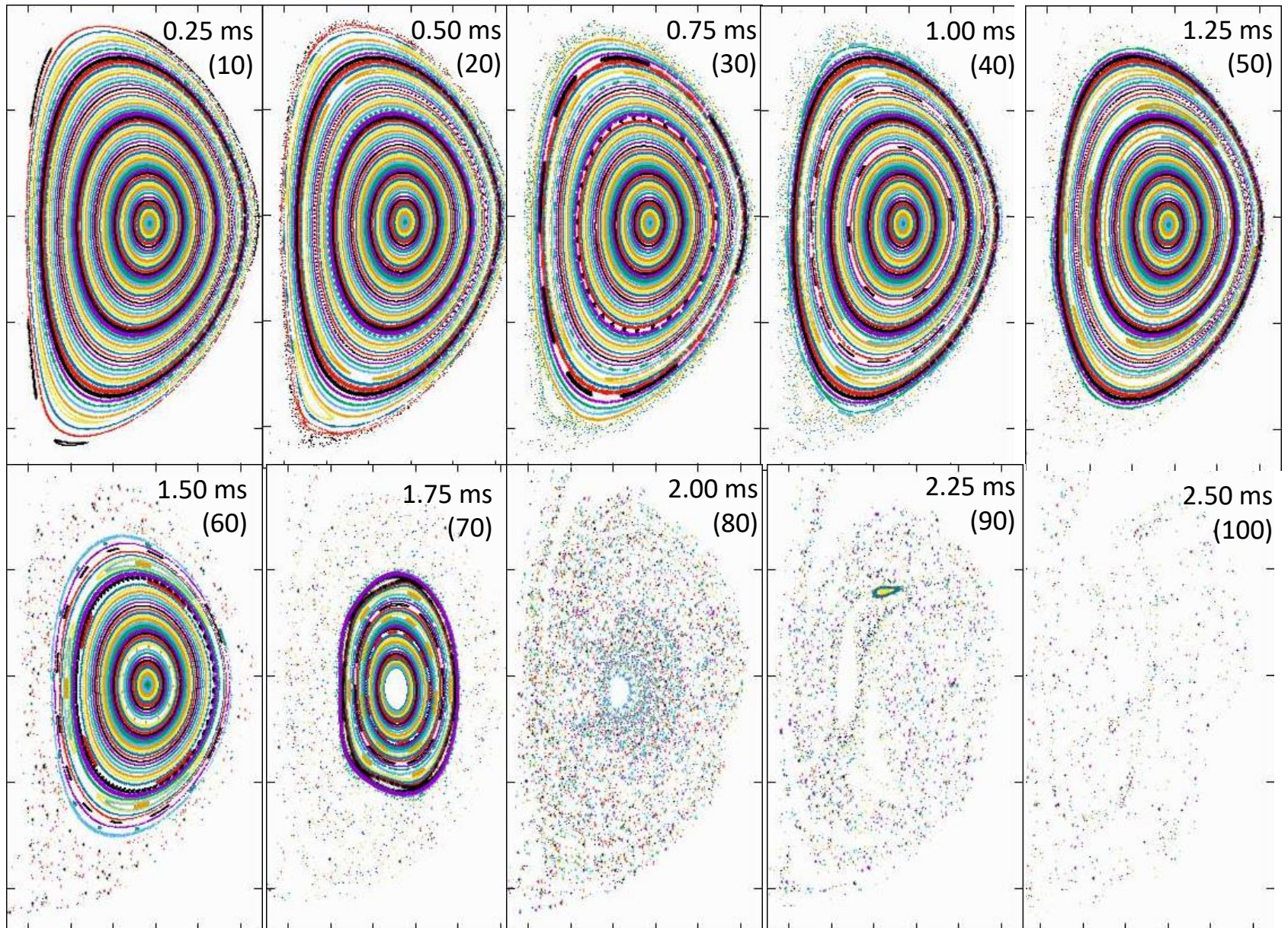
/global/cscratch1/sd/blyons/C1_33984065



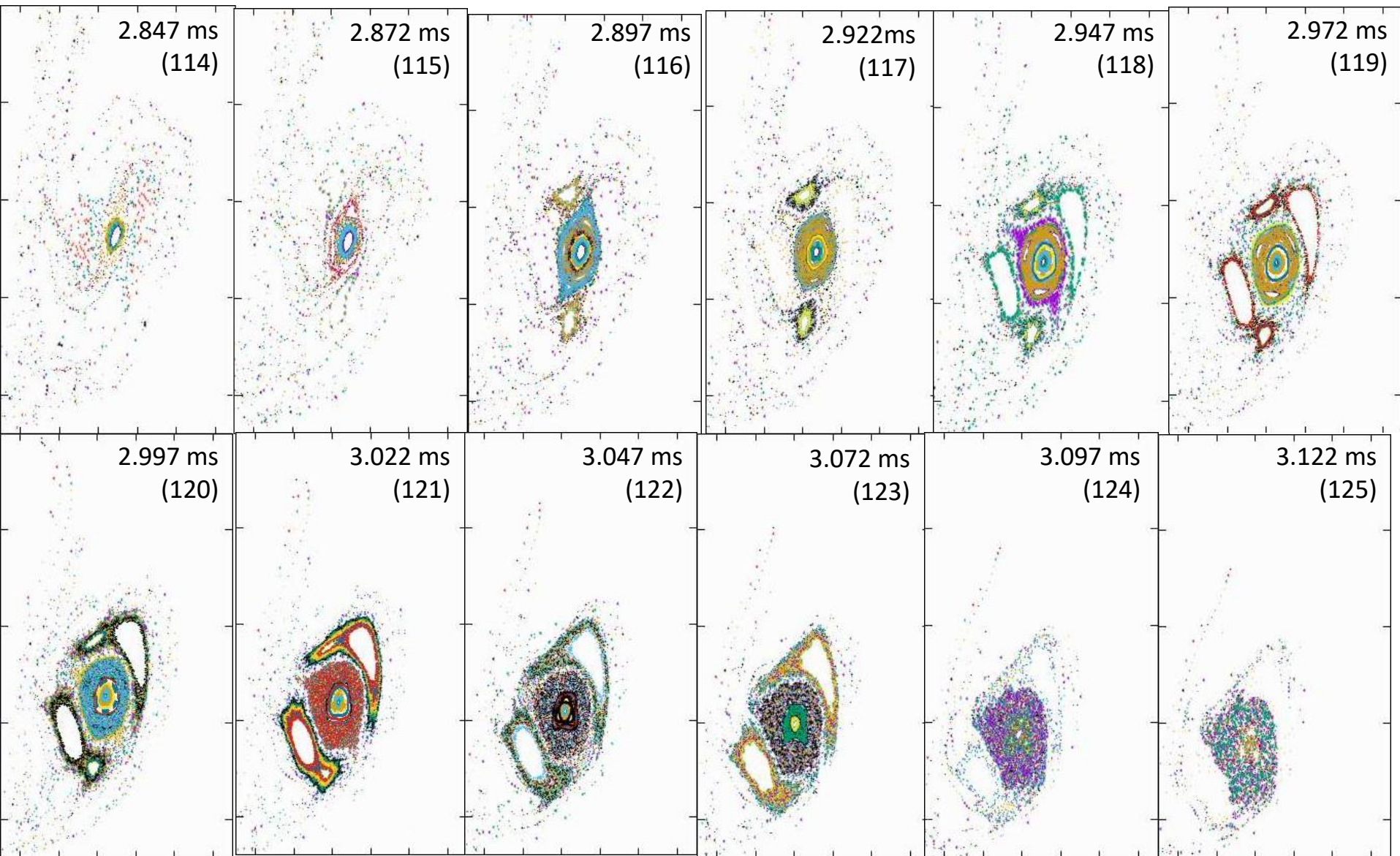
Mid-plane Electron Temperature and Electric Field



Magnetic Surface Breakup

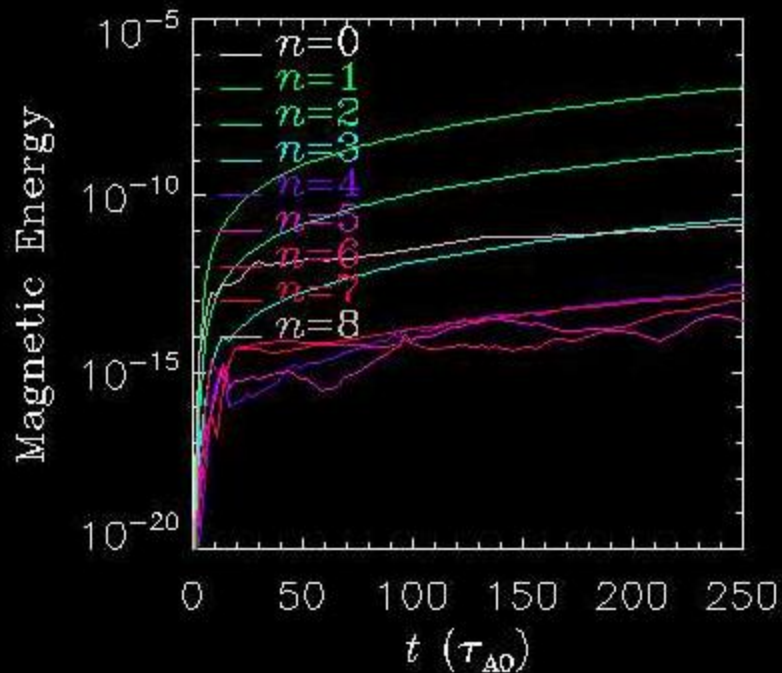


Partial Surfaces Reforming

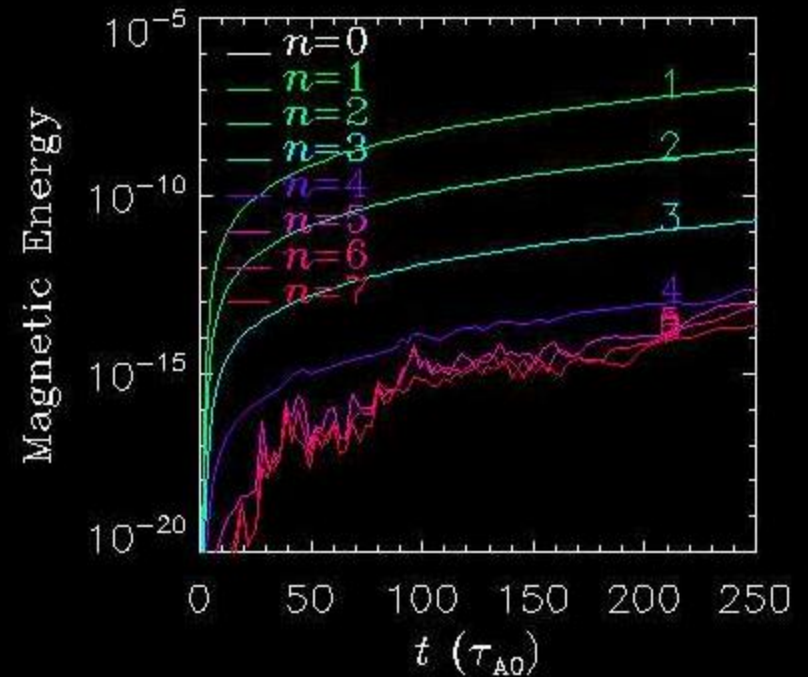


32 plane rerun now in progress

8 planes

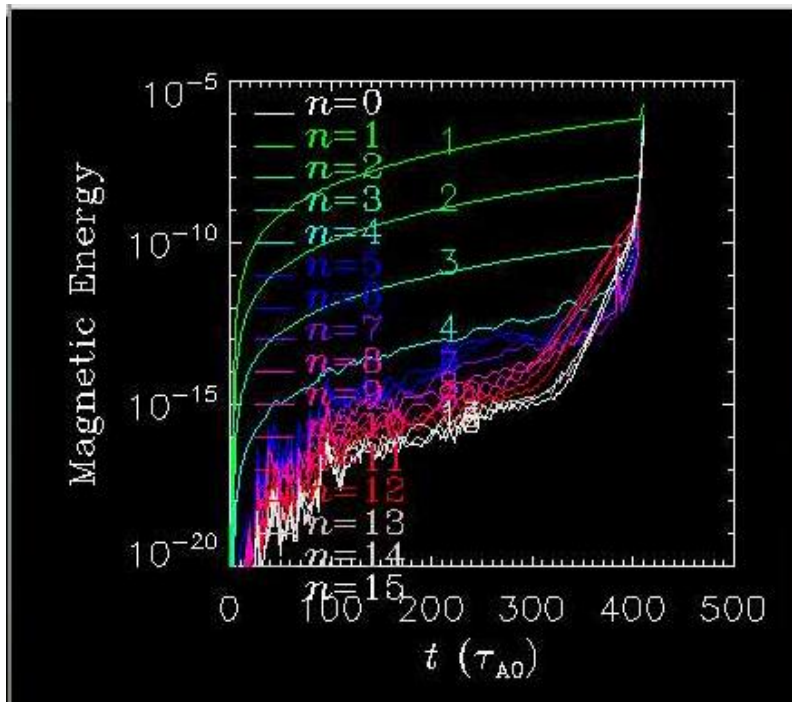


32 planes

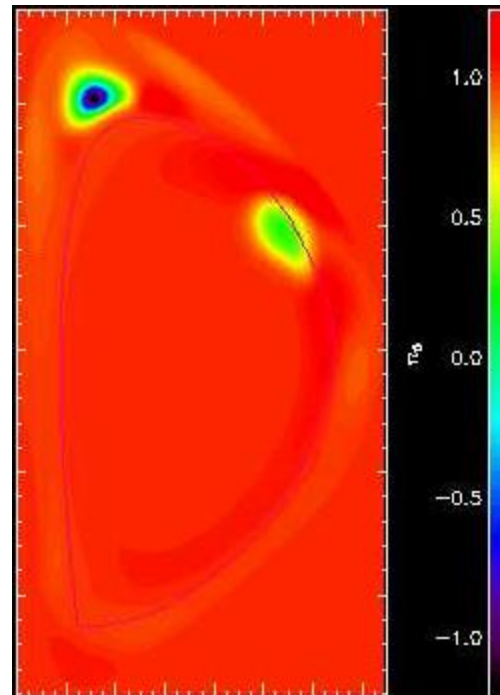


32 plane case crashed with negative density

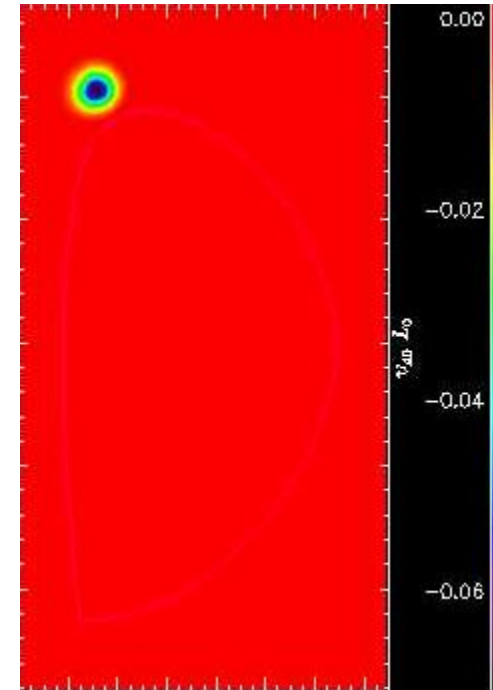
N=32 blows up at later time



density



Toroidal velocity @ 18



Near the end, $dt > dx / V$ in the toroidal direction, which can lead to oscillations. Recommendations:

(1) iupstream=1, (2) smaller dt, (3) increase hyperv, (4) increase denm

Energy conservation

6% error does not depend on:

- $dt=0.5, 1.0, 2.0$
 - $inocurrent_pol=0,1$
 - $inocurrent_tor=0,1$
 - $ltemp = 0.1$
 - $jadv = 0,1$
 - $etar = 1.e-7, 1.e-9$
 - $idents = 0,1$
-
- Now checking dependence on magnetic boundary conditions and form of Poynting Flux divergence: $\nabla \cdot (\mathbf{E} \times \mathbf{B})$