My professional path

- Physics major at UT-Austin. Largely unaware of plasma research there.
- Multiple graduate school options, chose fusion b/c it might be useful
- Also studied international science policy while in grad school
- Postdoc back at UT-Austin, because plasma physics is good there!
- My wife wanted to go to graduate school at Johns Hopkins, so we moved to Maryland. I joined the staff at UMD.
- Was an associate professor at Imperial College (London)
- For the last 20 years, professor in the physics department at UMD. Ran the Honors College. JPP editor. Presently joint UMD-PPPL.
- **I have worked with lots of terrific undergraduates. My advisees have won some nice awards. This is the first year I’ve worked with SULI.**

Computing for Science

William Dorland
UMD, PPPL

Algorithms, solutions, simulations, insight, prediction, discovery, …

Scientific computing != computer science

- Computer science: the study of computers and computing
- Scientific computing: using computers and computing to advance science

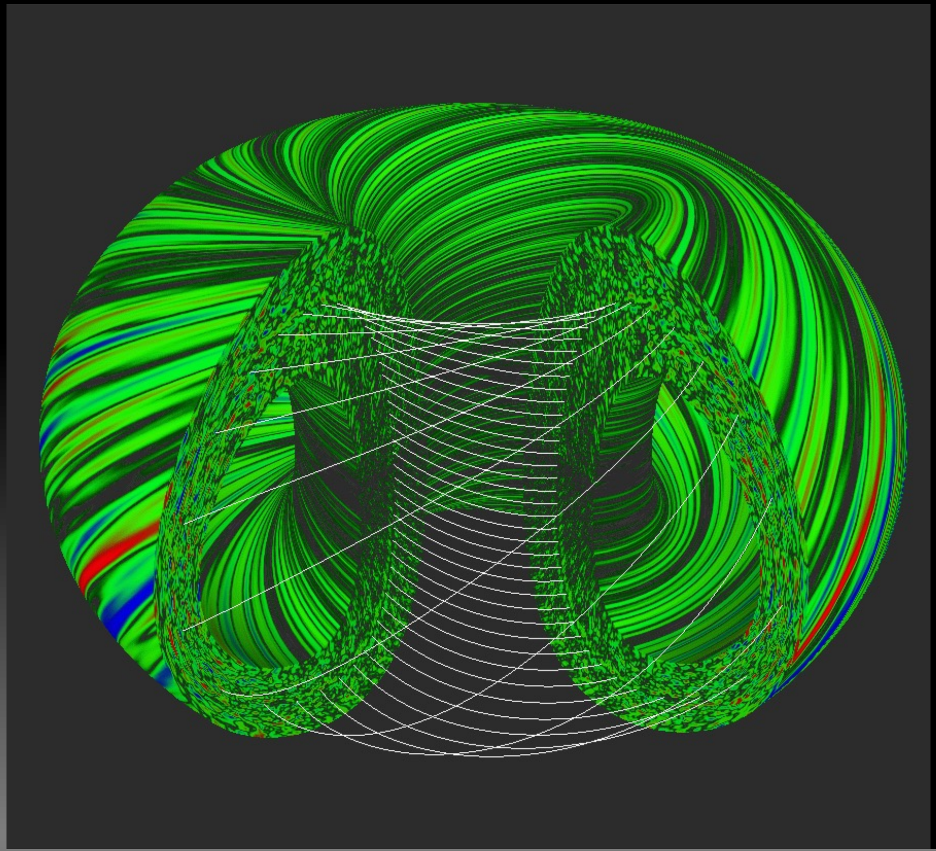
- Scientific computing is one kind of “applied computer science” but it is not the dominant application by many measures

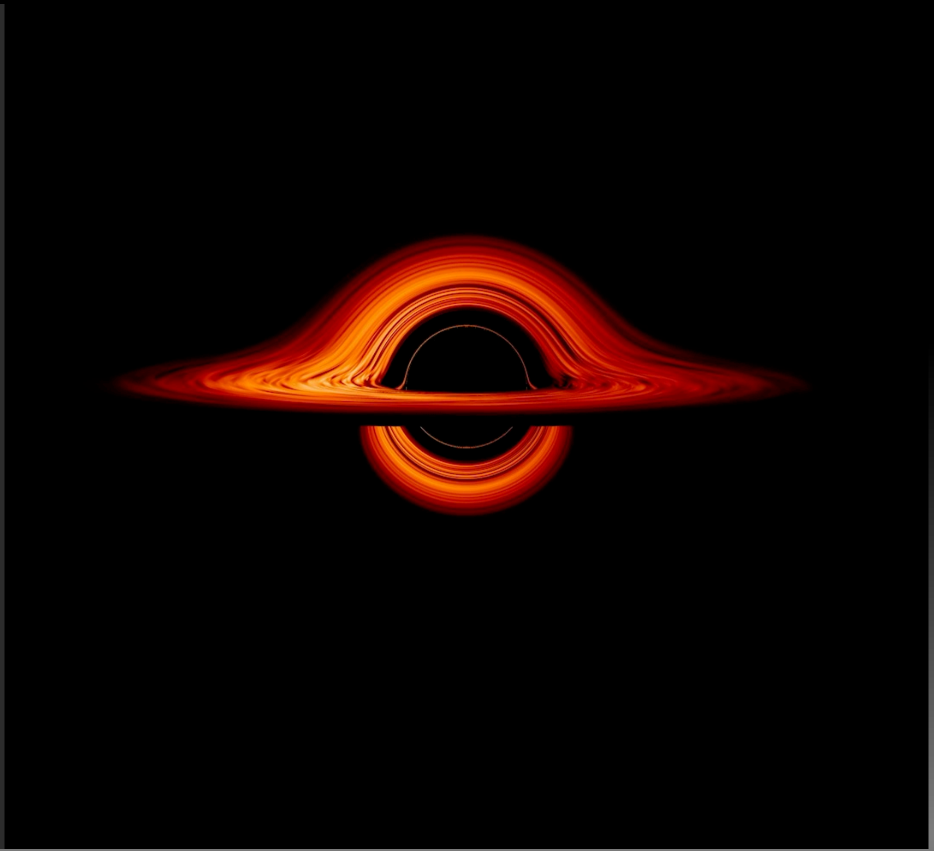
- People who do scientific computing have a wide range of skills and talents. The primary activity is not the act of writing code, though this is important.

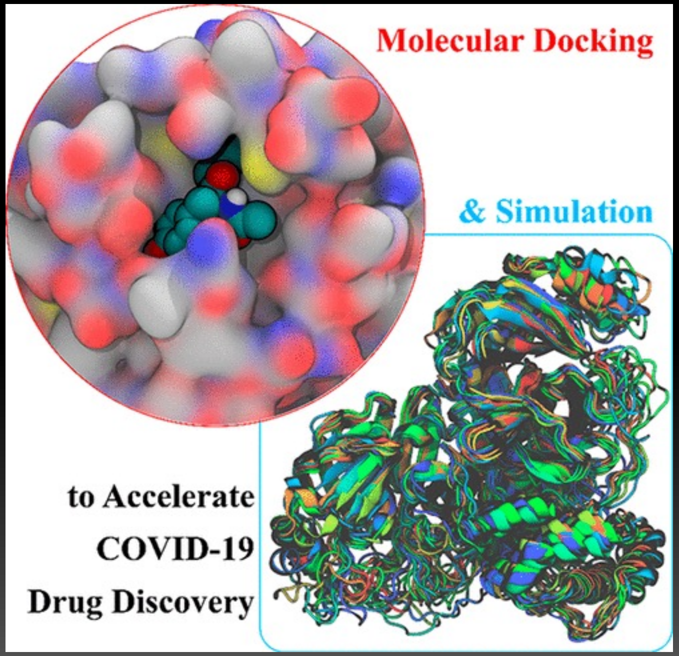
Conceptualize; model; design; enable; explore; discover; predict

What is the most exciting thing one can do with a supercomputer?

- Enable fusion? Visualize black holes? Discover cures for diseases?



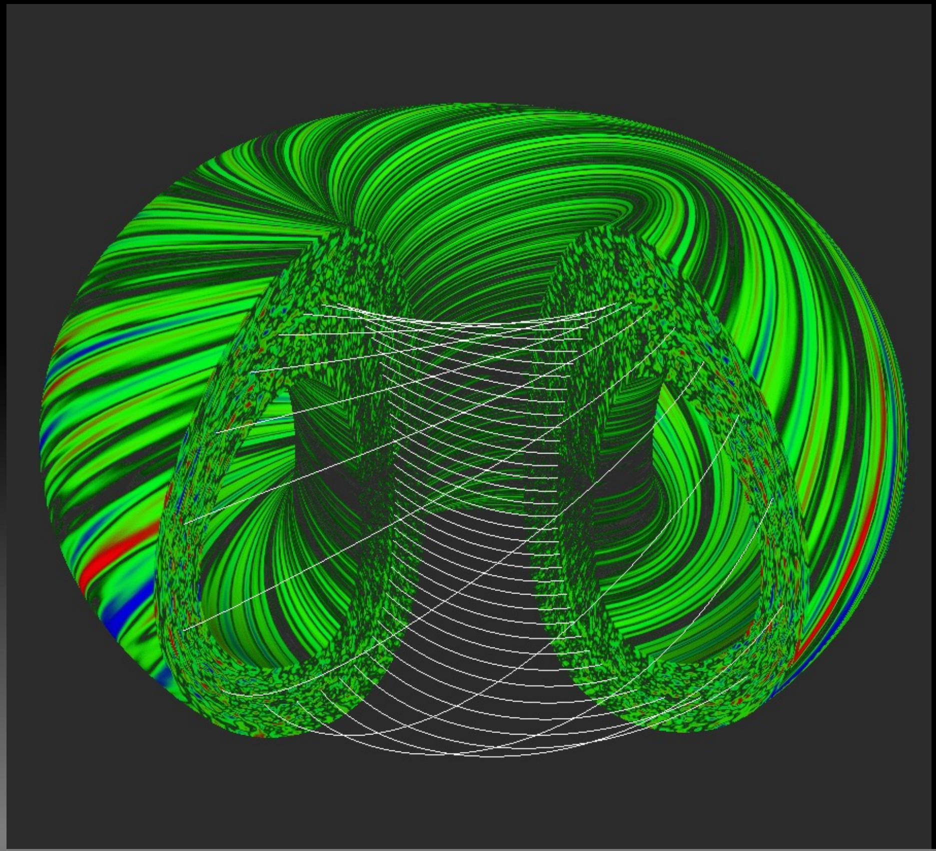


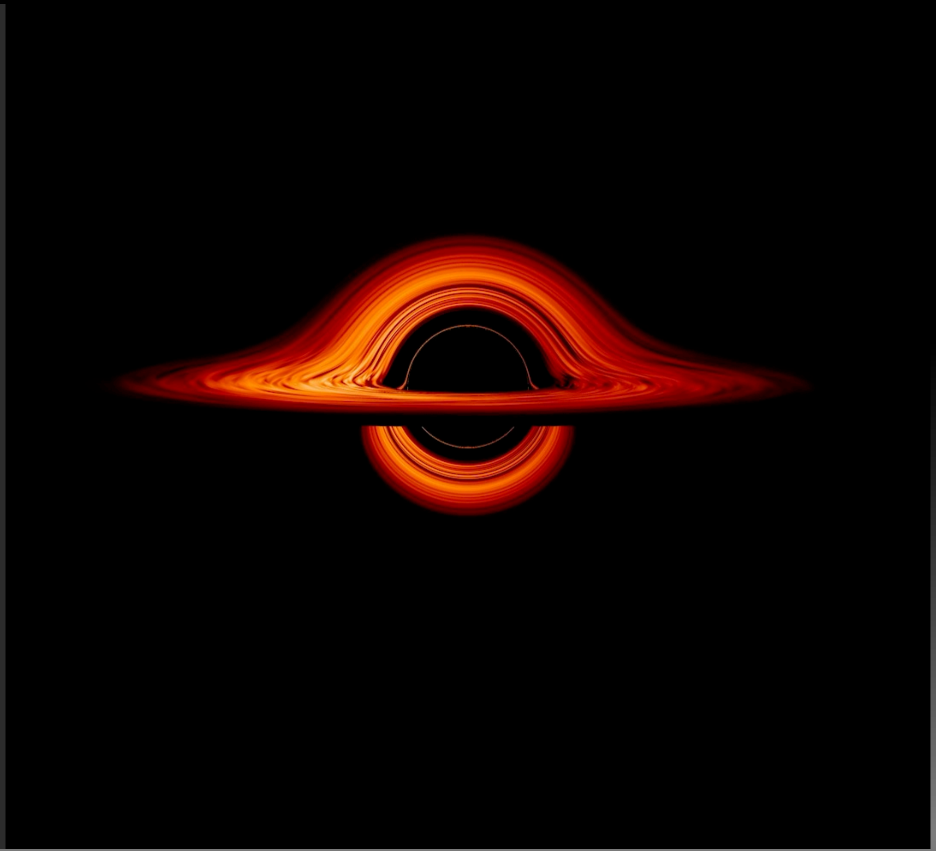


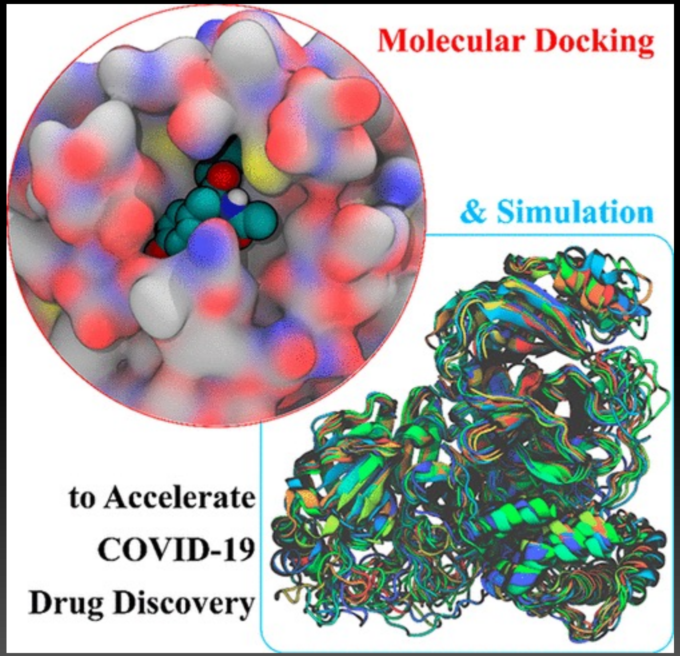
Acharya, et al

What is the most exciting thing one can do with a supercomputer?

TURN IT OFF!







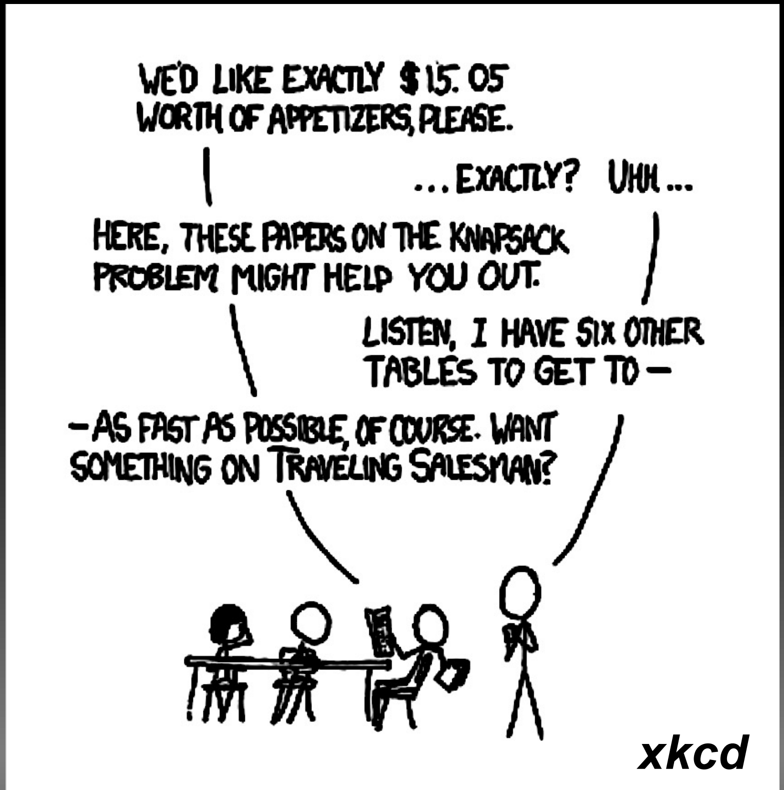
Acharya, et al

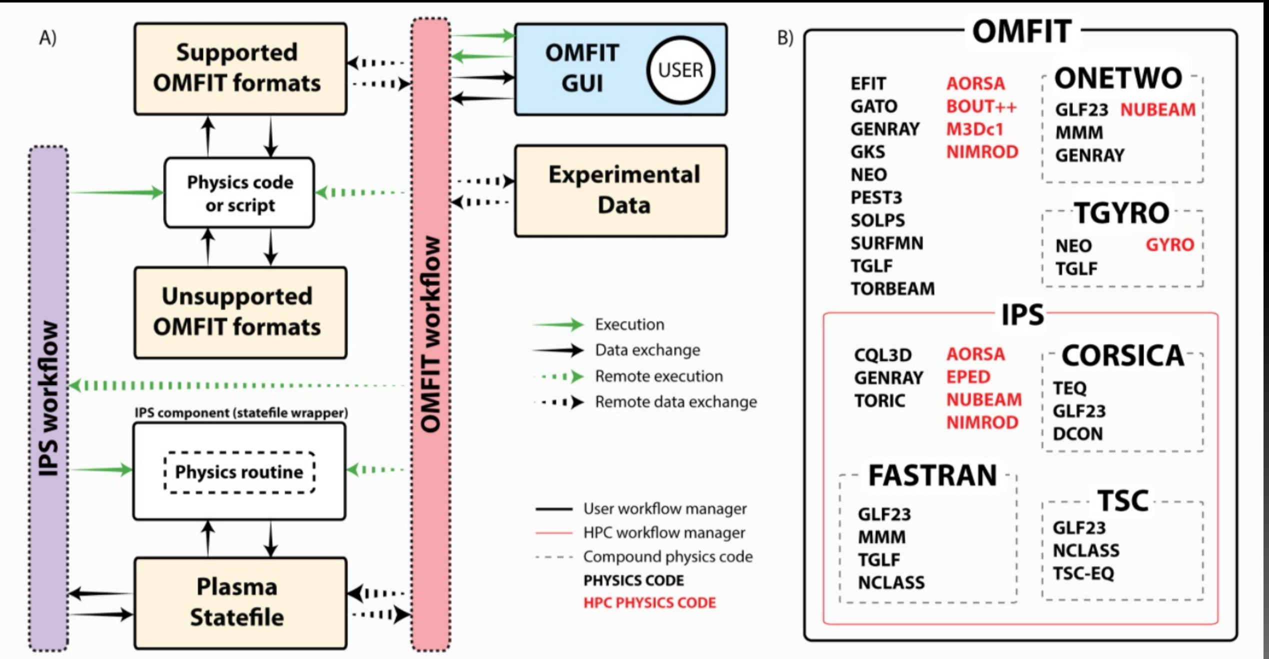
Five-step method for ending your dysfunctional supercomputer relationship

1. Conceptualize the problem in **mathematical** terms $$\frac{\partial \mathbf{E}}{\partial t} = -\nabla \times \mathbf{B}$$

Five-step method for ending your dysfunctional supercomputer relationship

1. Conceptualize the problem in **mathematical** terms
2. Design **algorithms** and **interfaces.** Typically more than once.





A schematic of the current inter-relationship and nesting of the frameworks and components selected for inclusion into AToM

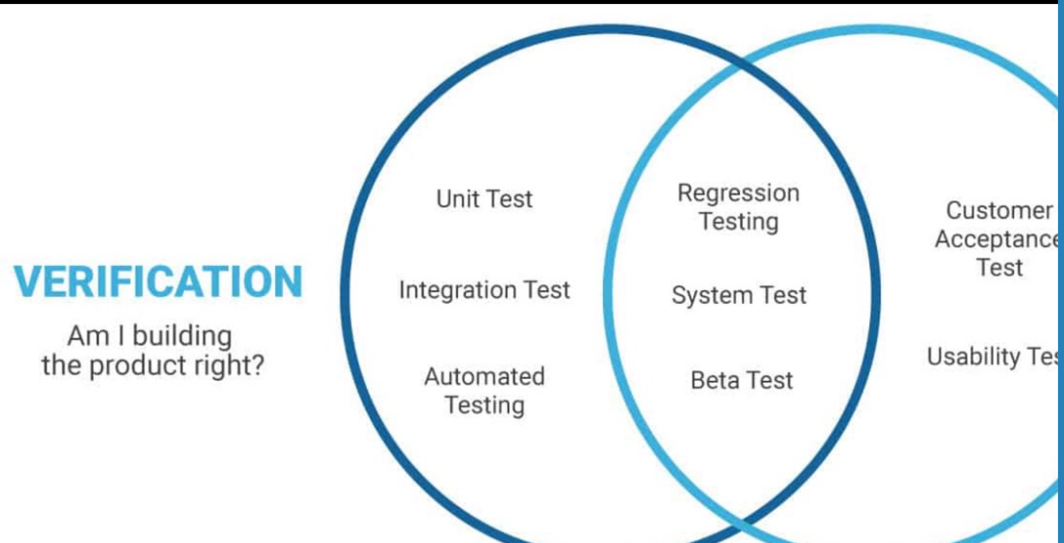
Five-step method for ending your dysfunctional supercomputer relationship

1. Conceptualize the problem in **mathematical** terms
2. Design **algorithms** and **interfaces.** Typically more than once. **Code it.**

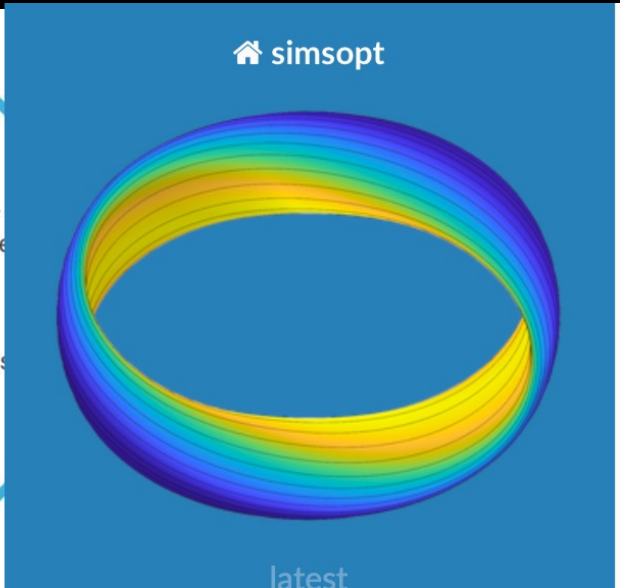
```
for (int m = threadIdx.z; m < nm; m += blockDim.z) {
  for (int l = threadIdx.y; l < nl; l += blockDim.y) {
    unsigned int globalIdx = idxyz + nR*(l + nl*(m + nm*is));
    int sl = l + 1; // offset to get past ghosts
    int sm = m + 2; // offset to get past ghosts

    rhs[globalIdx] = rhs[globalIdx]
      - vt_ * bgrad_ * ( - sqrtf(m+1)*(l+1)*S_G(sl,sm+1) - sqrtf(m+1)* l   *S_G(sl-1,sm+1)
                         + sqrtf(m  )* l   *S_G(sl,sm-1) + sqrtf(m  )*(l+1)*S_G(sl+1,sm-1) )

      - icv_d_s * ( sqrtf((m+1)*(m+2))*S_G(sl,sm+2) + (2*m+1)*S_G(sl,sm) + sqrtf(m*(m-1))*S_G(sl,sm-2) )
      - igb_d_s * (                  (l+1)*S_G(sl+1,sm) + (2*l+1)*S_G(sl,sm)                 + l*S_G(sl-1,sm) )

      - (nu_ + nuei_) * ( b_s + 2*l + m ) * ( S_G(sl,sm) );
```

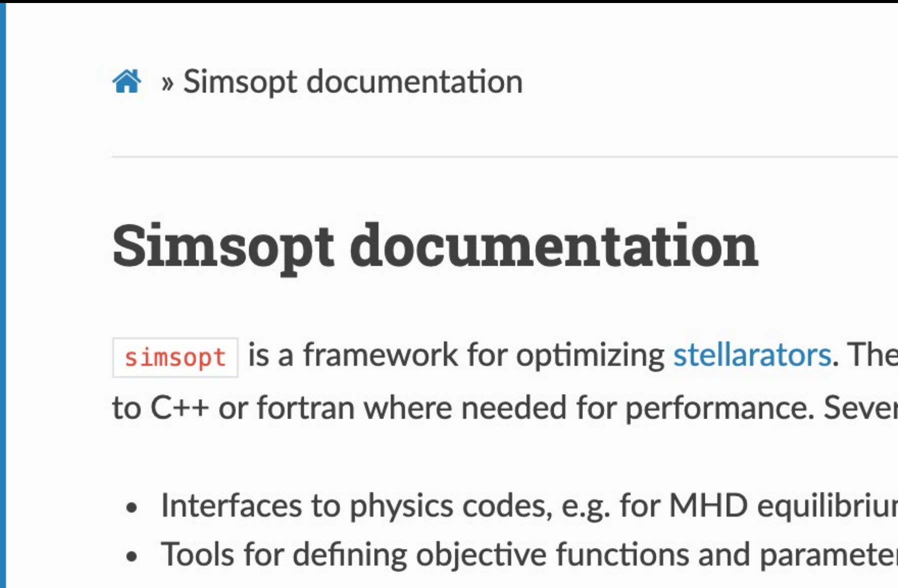
Five-step method for ending your dysfunctional supercomputer relationship

1. Conceptualize the problem in **mathematical** terms
2. Design **algorithms** and **interfaces.** Typically more than once. **Code it.**
3. **Verify** that the model being solved correctly. **Validate** the model against reality. **Document your work** – enabling future you and other users.



plutora.com

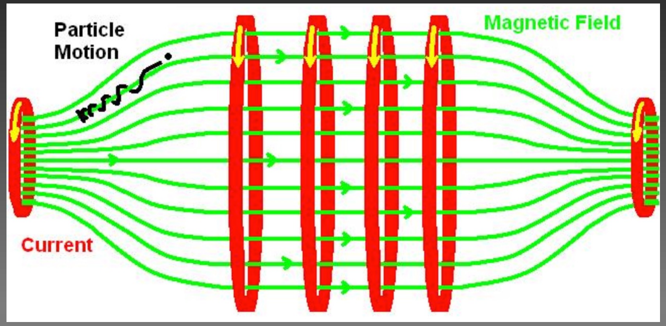


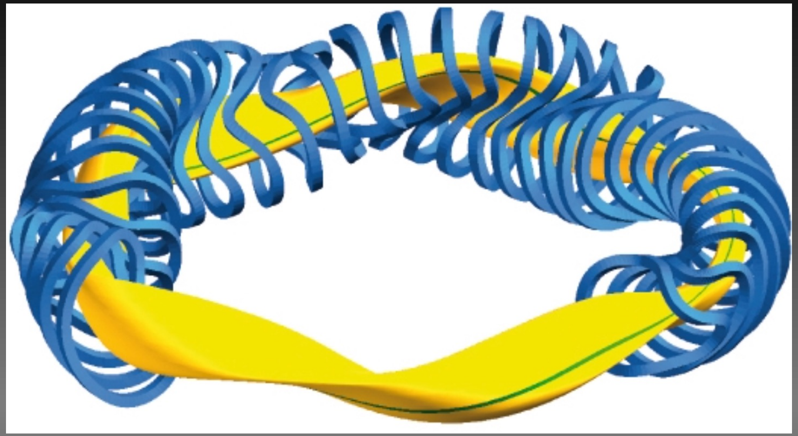


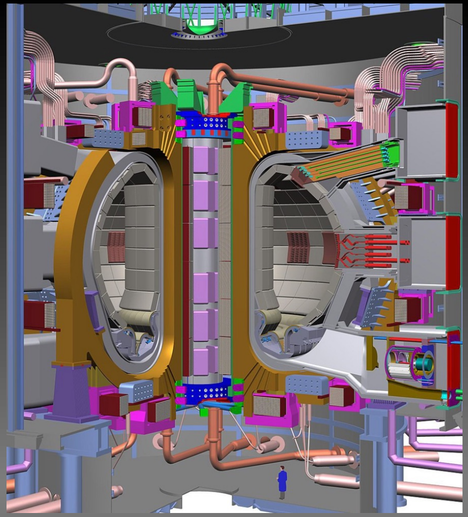
Five-step method for ending your dysfunctional supercomputer relationship

NASA

1. Conceptualize the problem in
2. Design **algorithms** and **inter**
3. **Verify** that the model being s
   reality. **Document your work**
4. Use the computer to study cases of interest. Develop **insights** and new, simpler models. Make **predictions**.







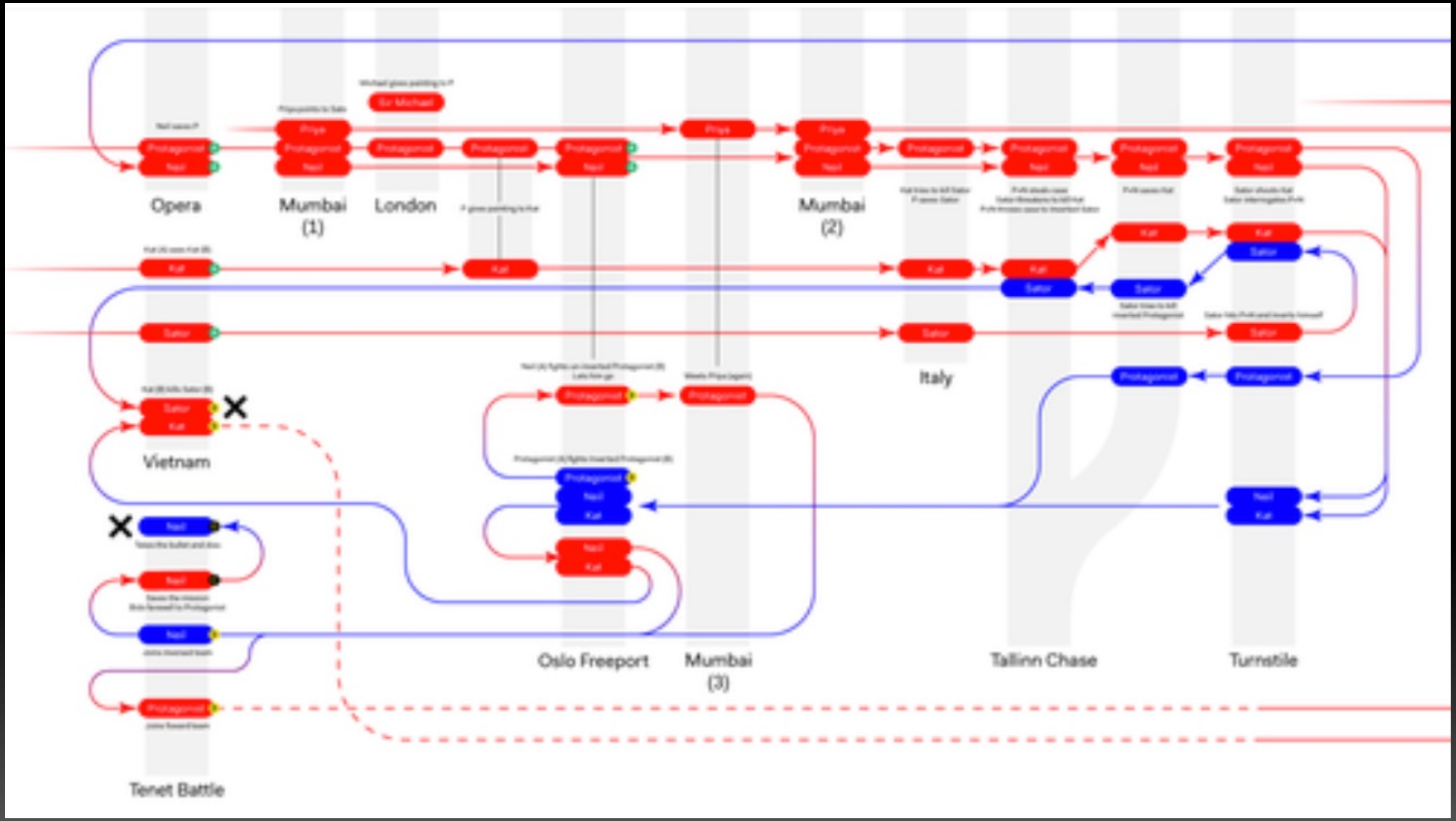
Five-step method for ending your dysfunctional supercomputer relationship

1. Conceptualize the problem in **mathematical** terms
2. Design **algorithms** and **interfaces.** Typically more than once. **Code it.**
3. **Verify** that the model being solved correctly. **Validate** the model against reality. **Document your work** – enabling future you and other users.
4. Use the computer to study cases of interest. Develop **insights** and new, simpler models. Make **predictions**.
5. **TURN IT OFF** *because you don't need that supercomputer anymore.*

Okay, this is really a twelve step method

12-step method for ending your dysfunctional supercomputer relationship

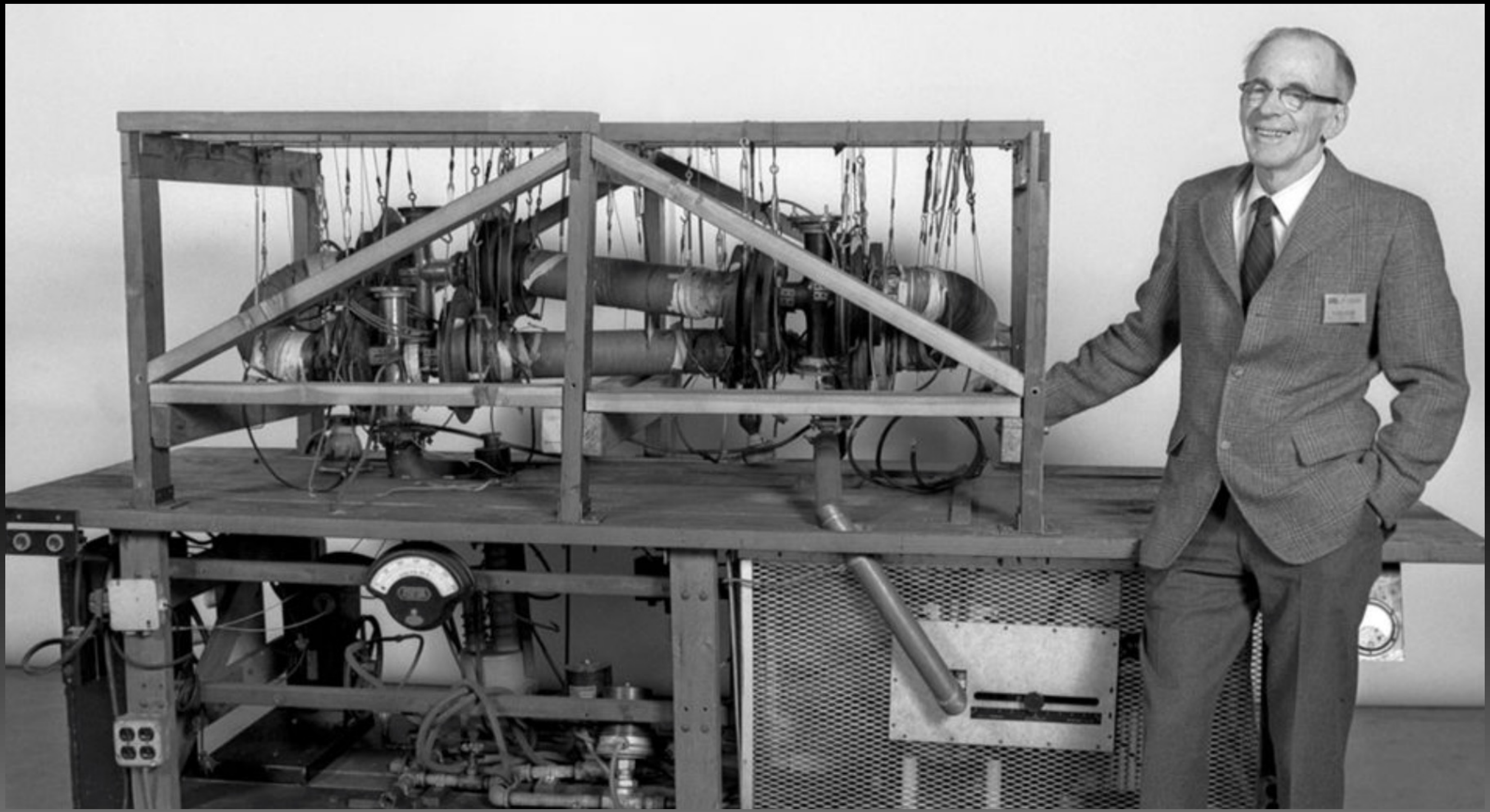
In the real world, the timeline of a scientific computation project is more like the timeline of Tenet



D Paul on Quora

Mathematical conceptualization can take years

- Fusion confinement concepts at first: Maxwell, Newton, and geometry



L Spitzer

Mathematical conceptualization can take years

- Fusion confinement concepts at first: Maxwell, Newton, and geometry
- Magnetohydrodynamic stability, collisional thermal conductivity next
- Instabilities that lead to turbulence identified, solved in simple limits
- Equations describing the turbulence derived, algorithms invented
- Computers used to model turbulence w/ Maxwell, Newton, and geometry

- From conceptualization to implementation took around thirty years!
- There are people who feel that the development of the mathematical and physical foundations is what defines physics. They might be right.

Algorithms: creativity in scientific computation

- An algorithm is a mechanical (unthinking) method for solving a problem.
- Since Turing we have known that finite algorithms can be appropriately general.
- Algorithms have requirements like *memory, minimum number of steps, maximum number of steps,* and *complexity.*
- Balancing these requirements against each other and matching them to the computing hardware and languages that are available now or in the future requires creativity. One of my favorite parts of scientific computing.
- **Implementation of an algorithm requires careful planning and attention to detail. This is the perspiration that goes with the inspiration.**

Writing and debugging code is the realm of software engineering

- Hard-won experience: bugs (*defects*) are inevitable. The best way to write correct code is to detect and fix defects when they first appear, or as soon thereafter as possible.
- Good languages help with debugging, typically by making every character count. Be sure you understand the symbols, the spaces, and the rules.
- Code that hasn't been tested and isn't regularly tested is probably defective.
- Writing good documentation is as important as writing good code.
- Back up your work, using with professional revision control software.
- **Continuous integration, clear interfaces, and lots of documentation *add* to the time spent in early development but *save* time overall.**

Good workflows and visualizations are essential

- For all but the simplest calculations, it is important to automate/simplify the setup of a problem instance and to make the output easy to understand.
- If you find you are repeating the same steps in a calculation, figure out how to automate those steps so that you can do more, more quickly.
- Visualization (or generally, the presentation of results) is important.

We tend to write tables and lists at first

1 2000PhPl....7..969D 2000/03 cited: 716
Comparisons and physics basis of tokamak transport models and turbulence simulations
Dimits, A. M.; Bateman, G.; Beer, M. A. *and 15 more*

2 2000PhPl....7.1904J 2000/05 cited: 700
Electron temperature gradient driven turbulence
Jenko, F.; Dorland, W.; Kotschenreuther, M. *and 1 more*

3 2009ApJS..182..310S 2009/05 cited: 574
Astrophysical Gyrokinetics: Kinetic and Fluid Turbulent Cascades in Magnetized Weakly Collisional Plasmas
Schekochihin, A. A.; Cowley, S. C.; Dorland, W. *and 4 more*

4 2000PhRvL..85.5579D 2000/12 cited: 453
Electron Temperature Gradient Turbulence
Dorland, W.; Jenko, F.; Kotschenreuther, M. *and 1 more*

5 1997PhPl....4.2482W 1997/07 cited: 410
A gyro-Landau-fluid transport model
Waltz, R. E.; Staebler, G. M.; Dorland, W. *and 3 more*

6 1995PhPl....2.2381K 1995/06 cited: 275
Quantitative predictions of tokamak energy confinement from first-principles simulations with kinetic effects
Kotschenreuther, M.; Dorland, W.; Beer, M. A. *and 1 more*

7 1993PhFlB...5..812D 1993/03 cited: 253
Gyrofluid turbulence models with kinetic effects
Dorland, W.; Hammett, G. W.

8 2008JGRA..113.5103H 2008/05 cited: 240
A model of turbulence in magnetized plasmas: Implications for the dissipation range in the solar wind
Howes, G. G.; Cowley, S. C.; Dorland, W. *and 3 more*

9 2006ApJ...651..590H 2006/11 cited: 229
Astrophysical Gyrokinetics: Basic Equations and Linear Theory
Howes, Gregory G.; Cowley, Steven C.; Dorland, William *and 3 more*

10 2008PhRvL.100f5004H 2008/02 cited: 221
Kinetic Simulations of Magnetized Turbulence in Astrophysical Plasmas
Howes, G. G.; Dorland, W.; Cowley, S. C. *and 4 more*

11 1993PPCF...35..973H 1993/08 cited: 214
Developments in the gyrofluid approach to Tokamak turbulence simulations
Hammett, G. W.; Beer, M. A.; Dorland, W. *and 2 more*

12 2001PhPl....8.4096J 2001/09 cited: 191
Critical gradient formula for toroidal electron temperature gradient modes
Jenko, F.; Dorland, W.; Hammett, G. W.

13 2000PhRvL..85.5336R 2000/12 cited: 189
Generation and Stability of Zonal Flows in Ion-Temperature-Gradient Mode Turbulence
Rogers, B. N.; Dorland, W.; Kotschenreuther, M.

14 2011PhRvL.107c5004H 2011/07 cited: 178
Gyrokinetic Simulations of Solar Wind Turbulence from Ion to Electron Scales
Howes, G. G.; Tenbarge, J. M.; Dorland, W. *and 4 more*

15 1997PhPl....4.3974S 1997/11 cited: 163
Landau fluid models of collisionless magnetohydrodynamics
Snyder, P. B.; Hammett, G. W.; Dorland, W.

16 2004PhRvL..93k4502S 2004/09 cited: 143
Experimental Observation and Characterization of the Magnetorotational Instability
Sisan, Daniel R.; Mujica, Nicolás; Tillotson, W. Andrew *and 5 more*

17 1997PhPl....4.1792B 1997/05 cited: 134
Gyrofluid simulations of turbulence suppression in reversed-shear experiments on the Tokamak Fusion Test R
Beer, M. A.; Hammett, G. W.; Rewoldt, G. *and 3 more*

18 2004PhPl...11.2637E 2004/05 cited: 126
Role of trapped electron mode turbulence in internal transport barrier control in the Alcator C-Mod Tokamak
Ernst, D. R.; Bonoli, P. T.; Catto, P. J. *and 10 more*

19 1992PhFlB...4.2052H 1992/07 cited: 120
Fluid models of phase mixing, Landau damping, and nonlinear gyrokinetic dynamics
Hammett, G. W.; Dorland, W.; Perkins, F. W.

20 2002PhRvL..89v5001J 2002/11 cited: 111
Prediction of Significant Tokamak Turbulence at Electron Gyroradius Scales
Jenko, F.; Dorland, W.

21 2008PhPl...15l2509A 2008/12 cited: 102
Linearized model Fokker-Planck collision operators for gyrokinetic simulations. I. Theory
Abel, I. G.; Barnes, M.; Cowley, S. C. *and 2 more*

22 2002ApJ...577..524Q 2002/09 cited: 99
The Magnetorotational Instability in a Collisionless Plasma
Quataert, Eliot; Dorland, William; Hammett, Gregory W.

23 2006PhPl...13l2306N 2006/12 cited: 96
Characterizing electron temperature gradient turbulence via numerical simulation
Nevins, W. M.; Candy, J.; Cowley, S. *and 8 more*

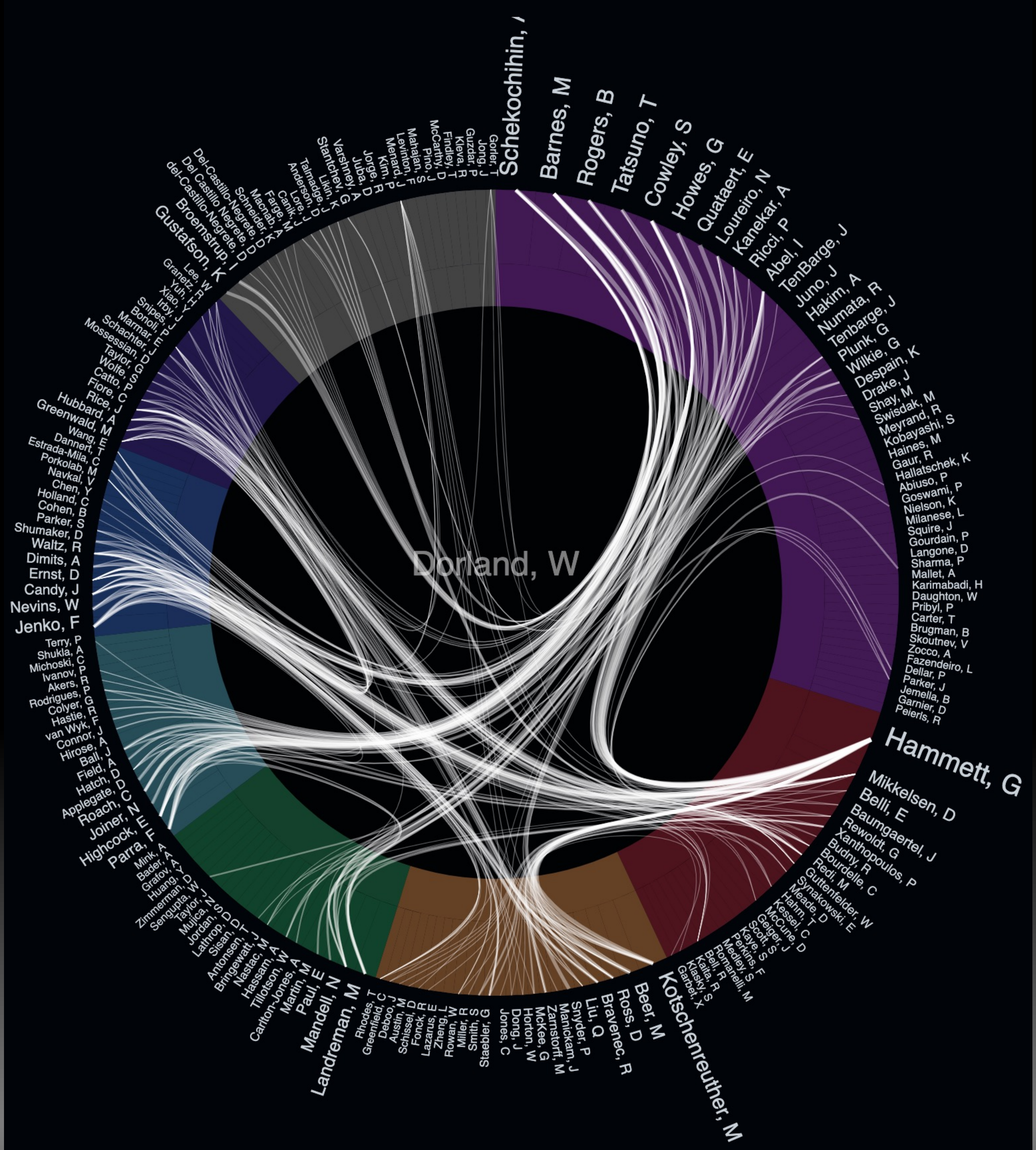
24 2009PhRvL.103a5003T 2009/07 cited: 93
Nonlinear Phase Mixing and Phase-Space Cascade of Entropy in Gyrokinetic Plasma Turbulence
Tatsuno, T.; Dorland, W.; Schekochihin, A. A. *and 4 more*

25 2005PhRvL..95w5003L 2005/12 cited: 88
X-Point Collapse and Saturation in the Nonlinear Tearing Mode Reconnection
Loureiro, N. F.; Cowley, S. C.; Dorland, W. D. *and 2 more*

26 2008PPCF...50l4024S 2008/12 cited: 87
Gyrokinetic turbulence: a nonlinear route to dissipation through phase space
Schekochihin, A. A.; Cowley, S. C.; Dorland, W. *and 5 more*

Visualizations can be more effective

Good visualizations can even be beautiful



NASA/ADS

What are categories of “scientific computation”?

- Equilibrium or steady state; leads to time-independent equations.

$$\nabla p = \mathbf{j} \times \mathbf{B}$$

Grad-Shafranov eq pressure force balanced by electromagnetic force

$$C(F_0) = 0 \rightarrow F_0 \propto \frac{n \exp[-m(v-u)^2/2T]}{T^{3/2}}$$

Strong collisions produce a distribution function that is a Maxwell-Boltzmann distribution

What are categories of “scientific computation”?

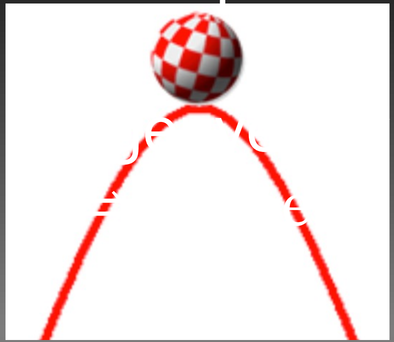
- Equilibrium or steady state; leads to time-independent equations.
- Stability calculations; typically linear integro-differential eqns.

stable

roblems

c polynomial

⇔ dispersion relation

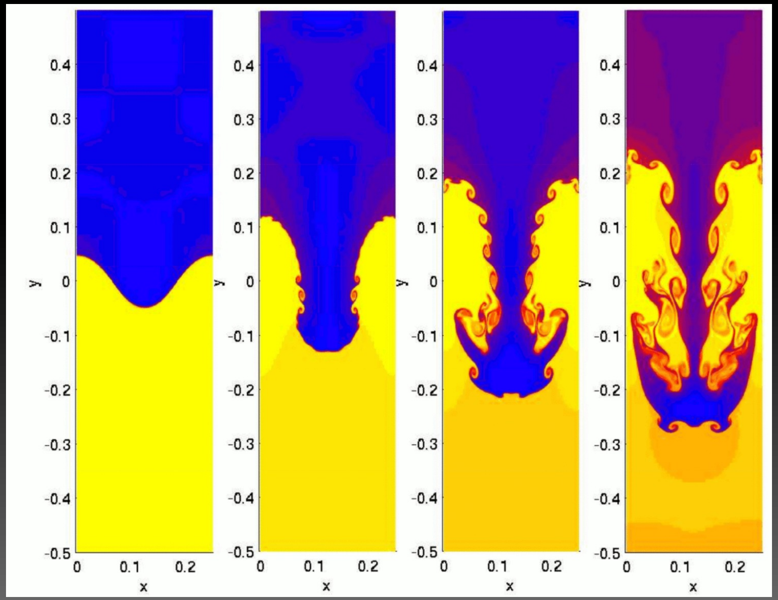


unstable

perturbation

$$\mathbf{Lx} = \lambda \mathbf{x}$$

Heavy fluid



Light fluid

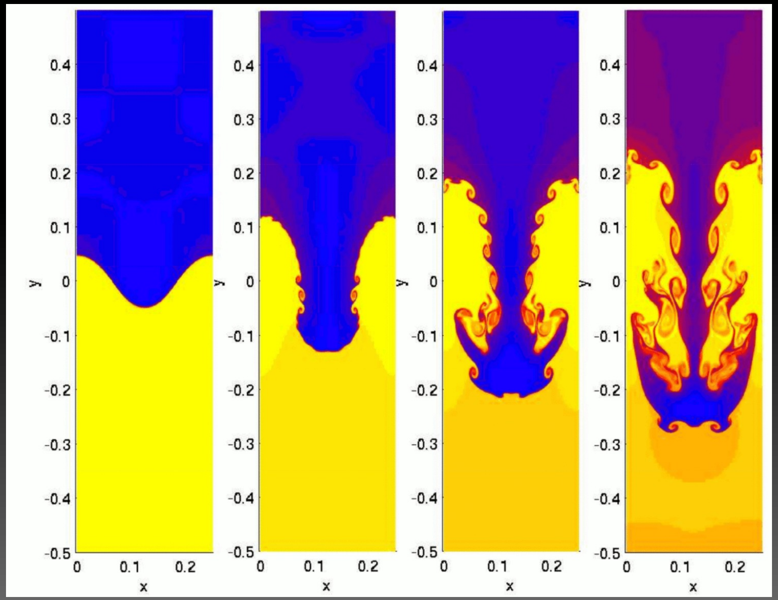
↑ this panel

What are categories of “scientific computation”?

- Equilibrium or steady state; leads to time-independent equations.
- Stability calculations; typically linear integro-differential eqns.
- Turbulence simulations. These are nonlinear systems of equations.

Heavy fluid

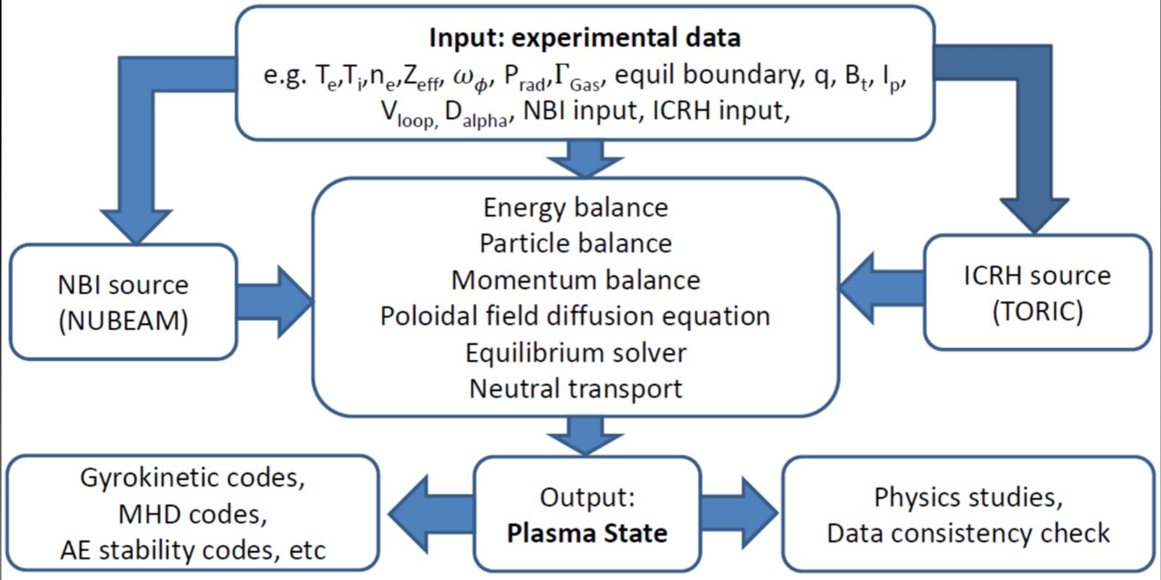
Light fluid



this panel ↑

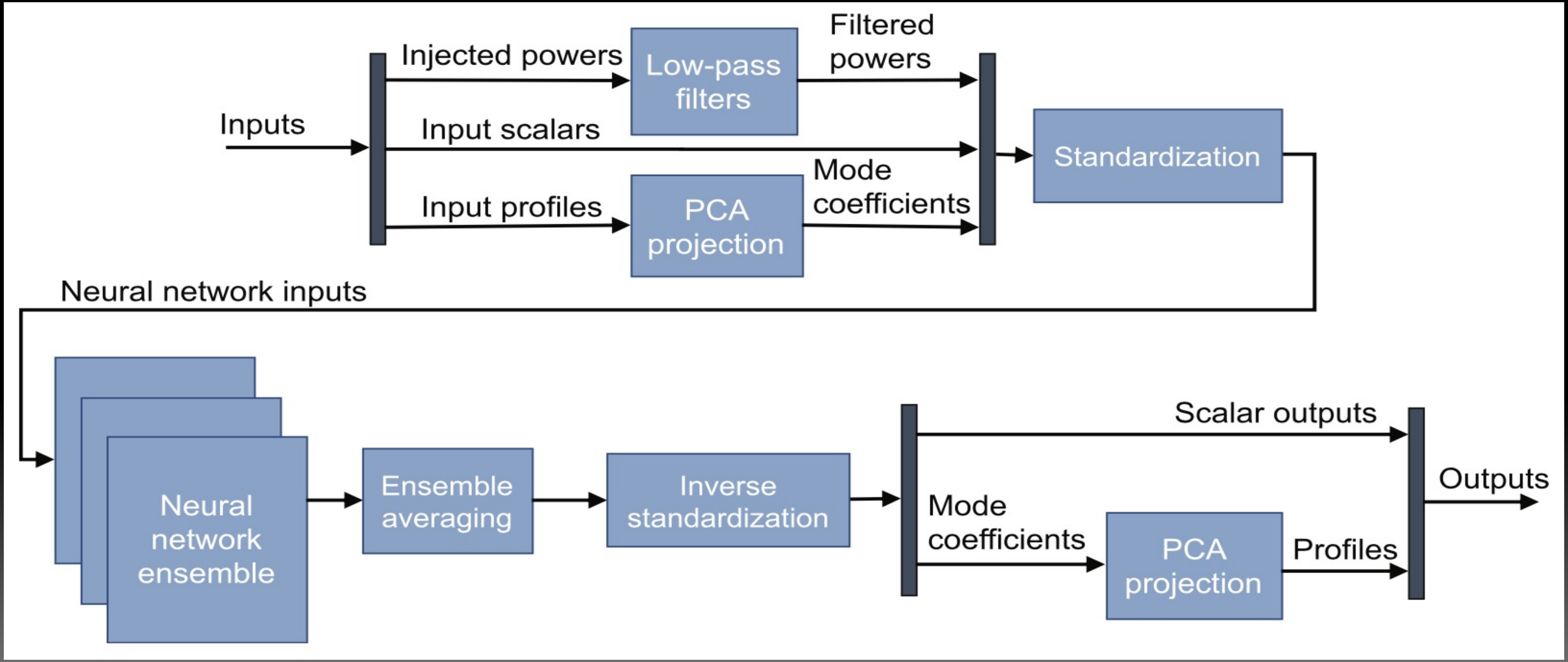
What are categories of "scientific computation"?

- Equilibrium or steady state; leads to time-independent equations.
- Stability calculations; typically linear integro-differential eqns.
- Turbulence simulations. These are nonlinear systems of equations.
- Transport equations; typically nonlinear but not necessarily turbulent



TRANSP, EUROfusion

What are categories of “scientific computation”?



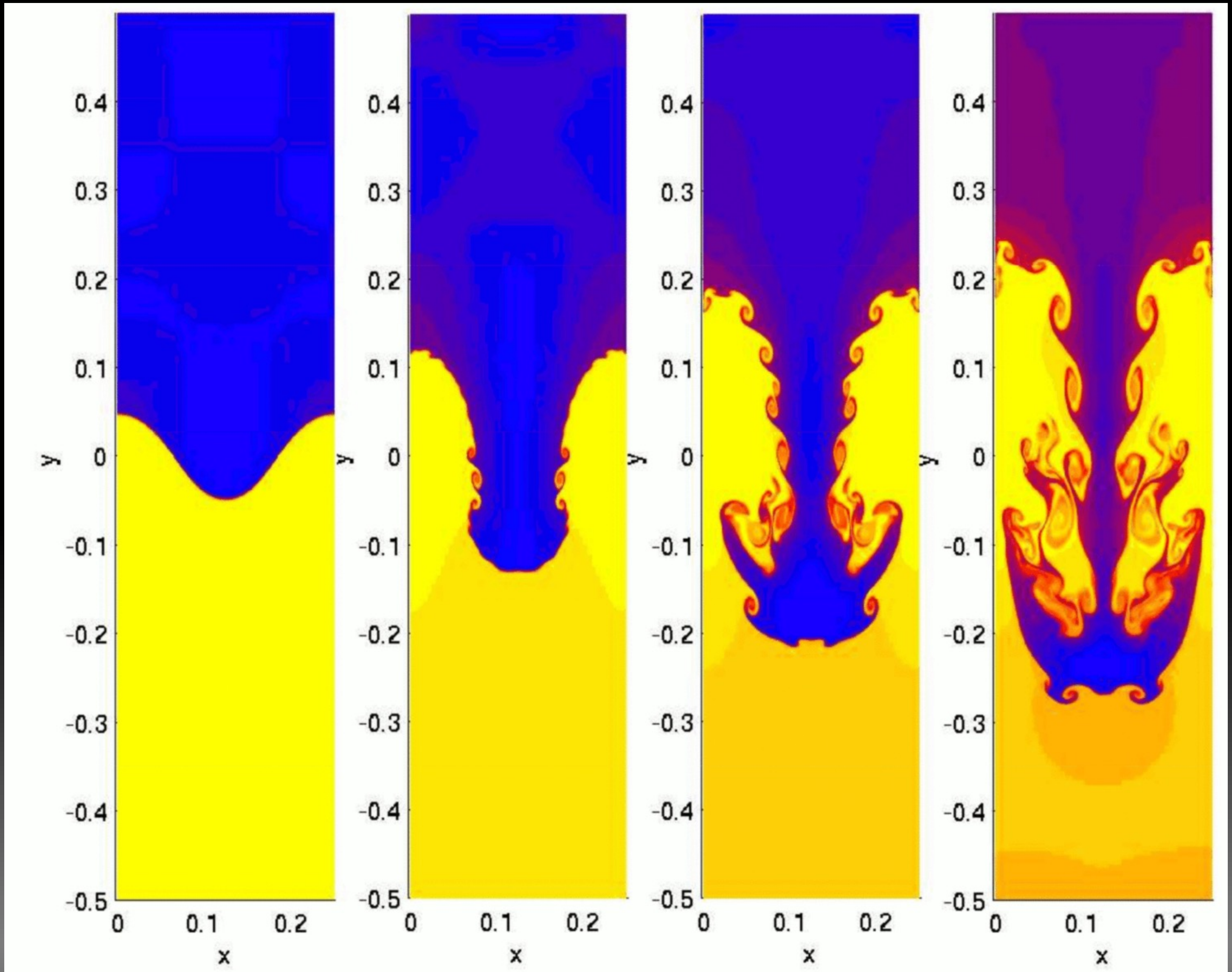
Boyer

Let’s go back a couple of slides…

- There are interesting patterns in this picture.

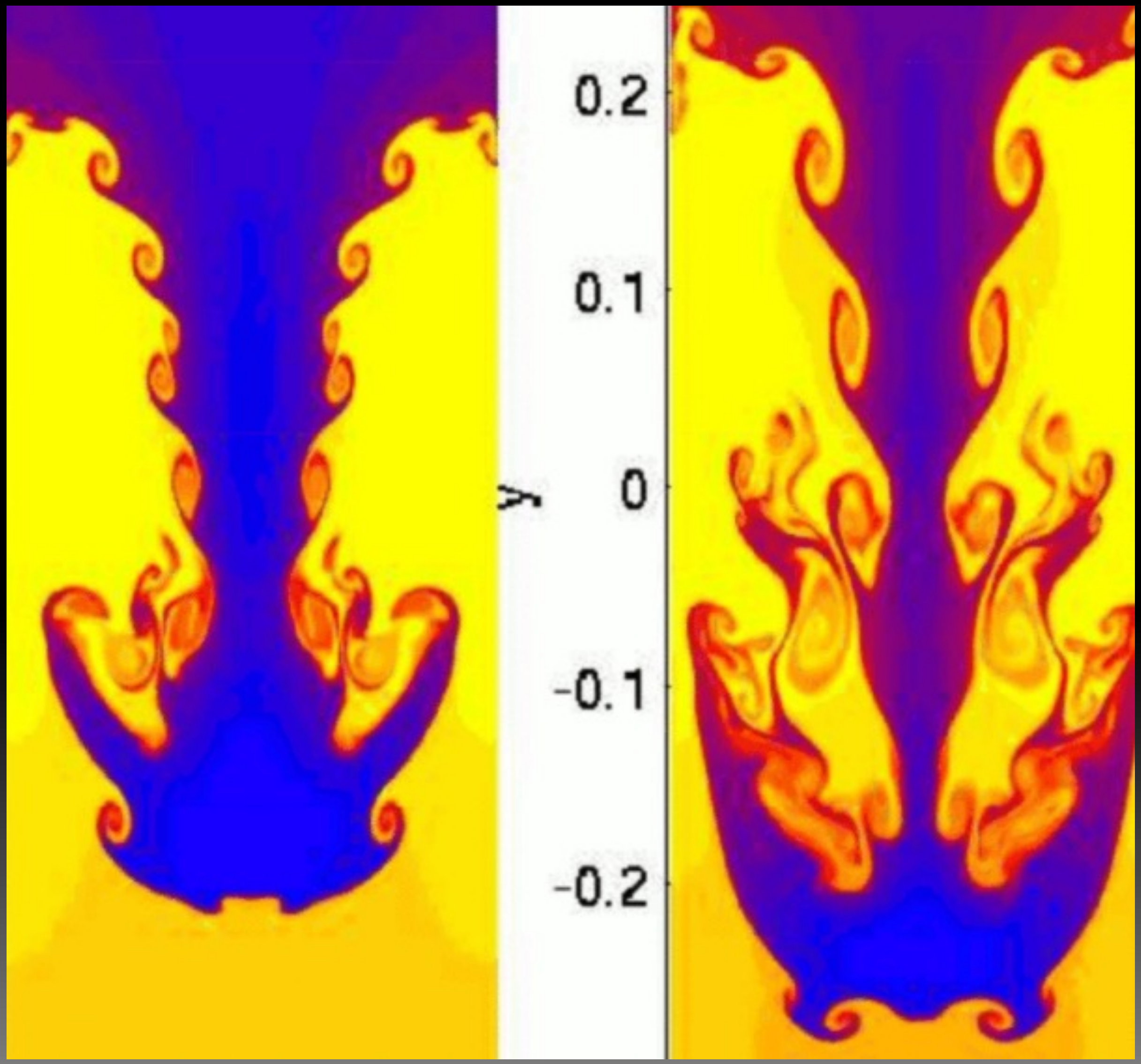
Heavy fluid

Light fluid



Zooming in, one sees whorls (vortices, eddies)

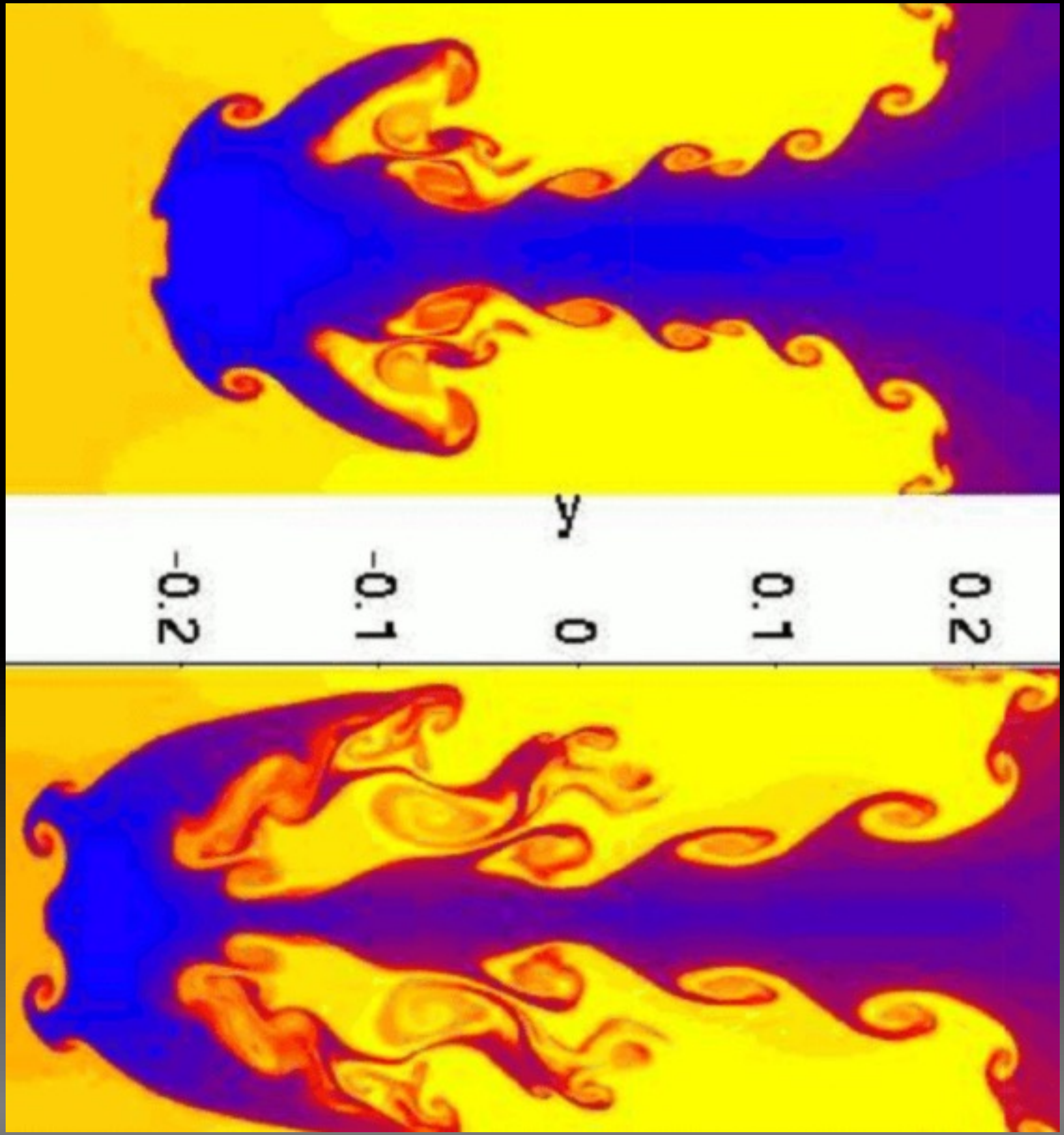
- There are interesting patterns in this picture.



Zooming in, one sees whorls (vortices, eddies)

- There are interesting patterns in this picture.
- Compare with this pattern someone saw in Portland (KGW8)

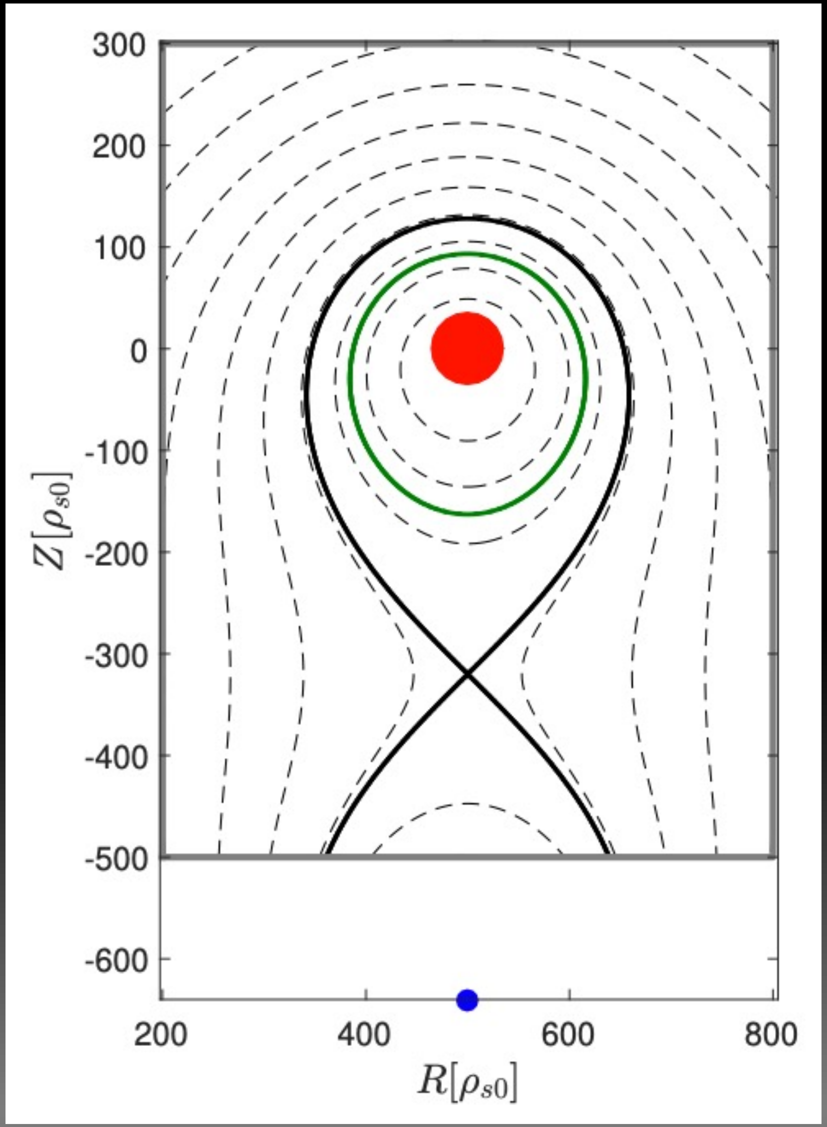
Zooming in, one sees whorls (vortices, eddies)



Some of us get captured by these ideas

- Similar patterns of fingers, whorls, eddies, jets, and more are seen in simulations of plasma turbulence
- Examples from simulations of Giacomi and Ricci

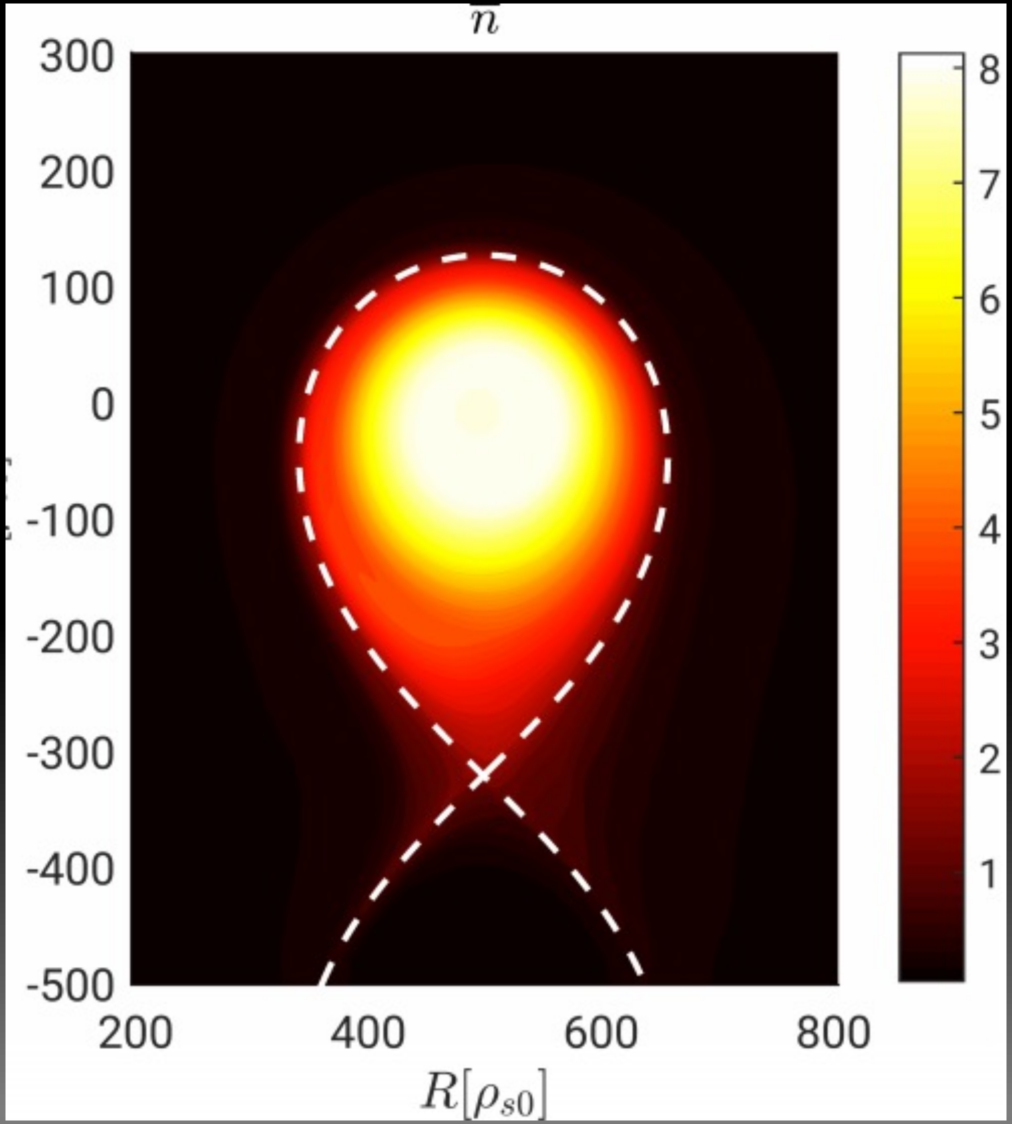
- Flux surfaces; includes a separatrix and an X-point



Some of us get captured by these ideas

- Similar patterns of fingers, whorls, eddies, jets, and more are seen in simulations of plasma turbulence
- Examples from simulations of Giacomi and Ricci

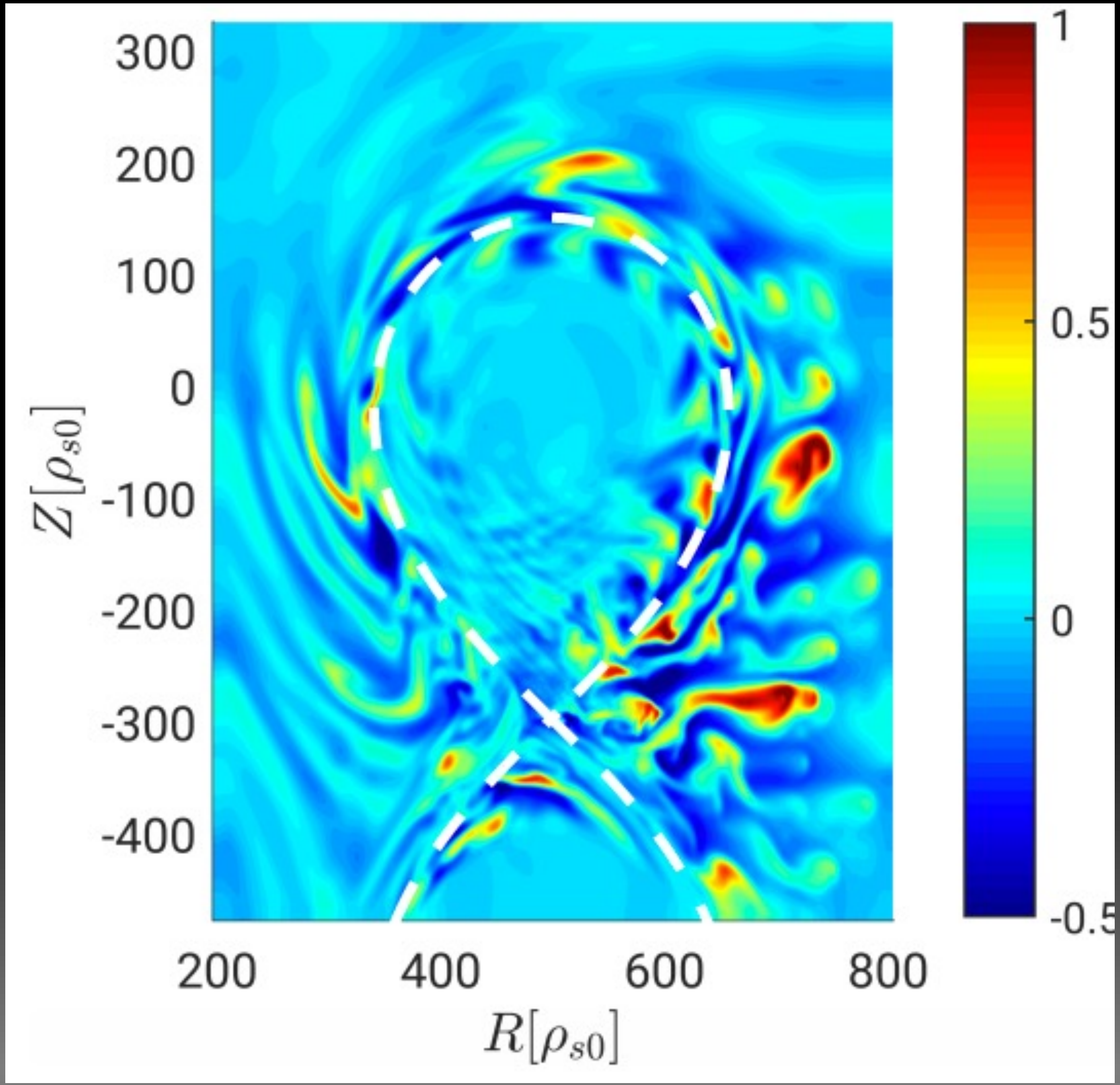
- This is the plasma density of the equilibrium



Some of us get captured by these ideas

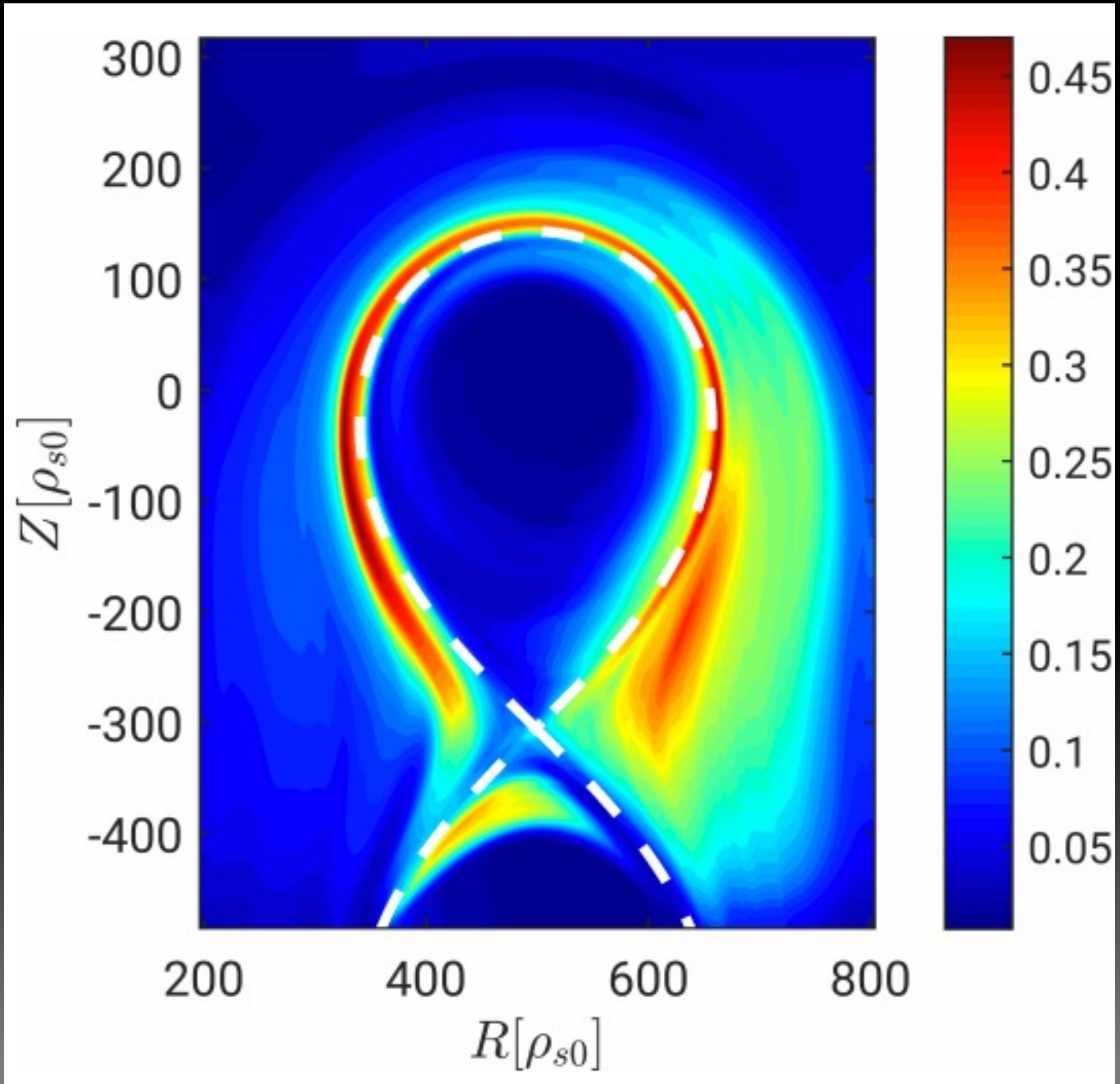
- Similar patterns of fingers, whorls, eddies, jets, and more are seen in simulations of plasma turbulence
- Examples from simulations of Giacomi and Ricci

- Here is a snapshot of the density fluctuations



Some of us get captured by these ideas

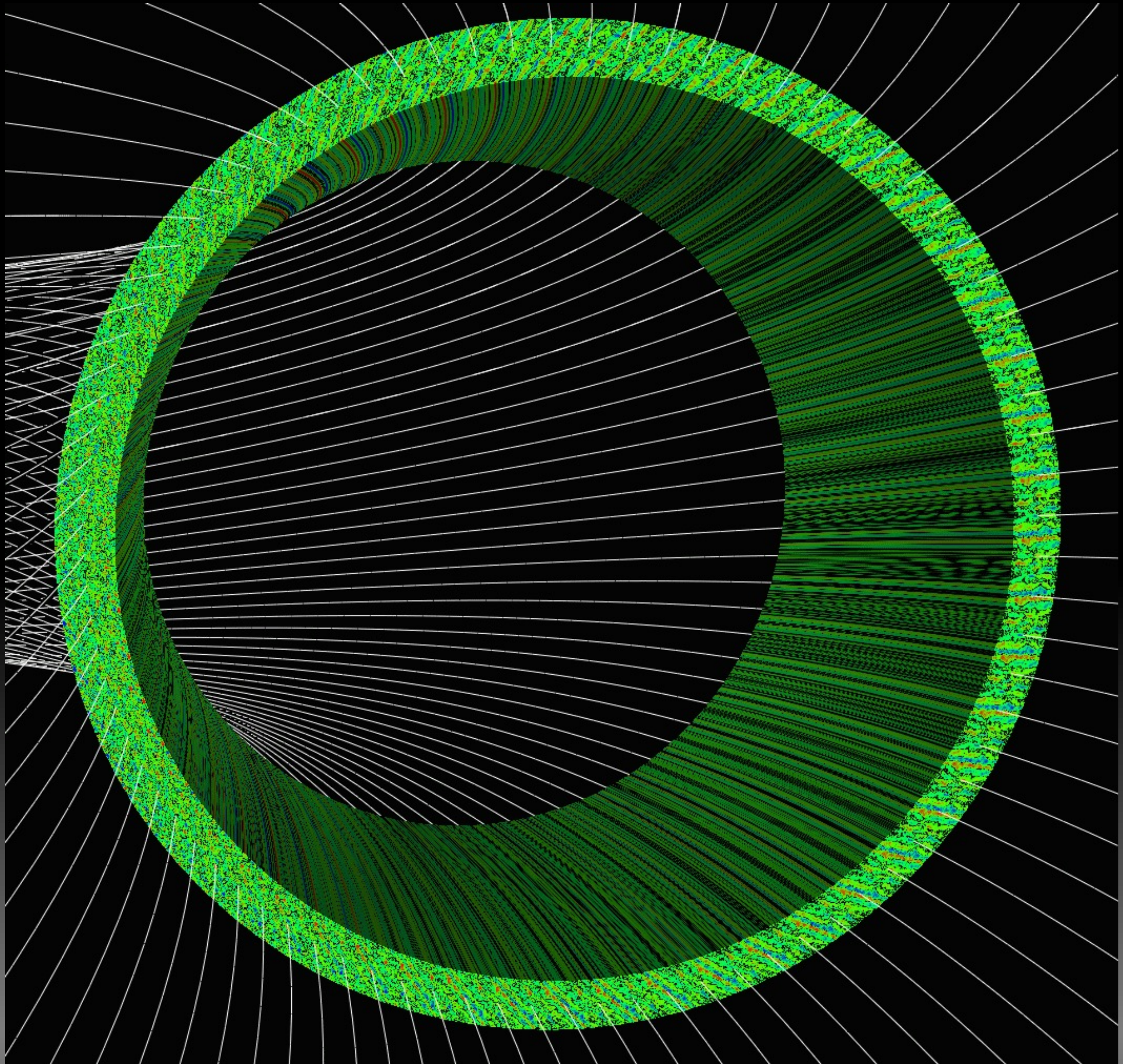
- Similar patterns of fingers, whorls, eddies, jets, and more are seen in simulations of plasma turbulence
- Examples from simulations of Giacomi and Ricci
- This is the variance of the density fluctuations



Parasitic instability model: tokamak turbulence

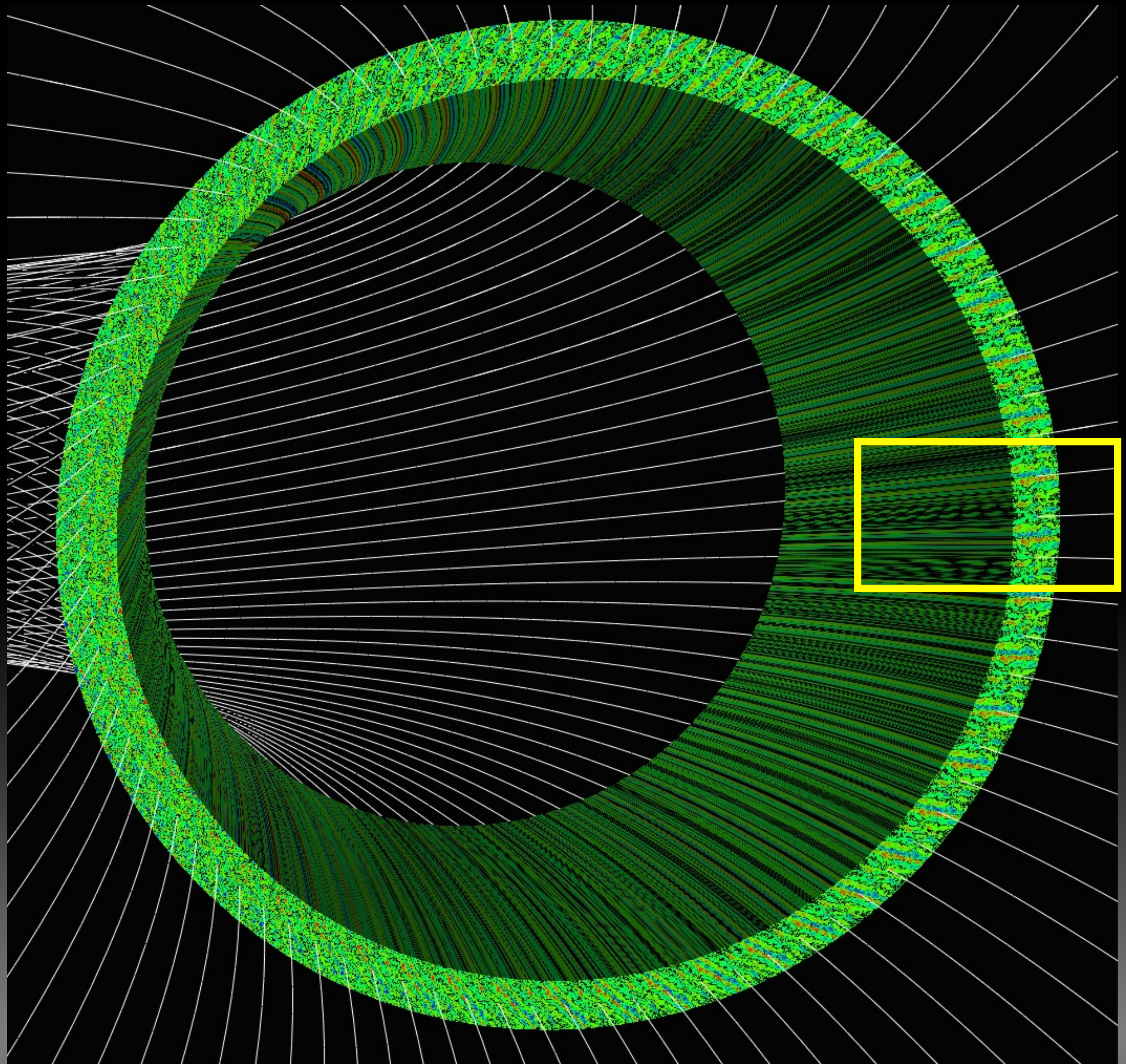
- Equilibrium unstable to *primary* instabilities which form radial jets
- Jets are unstable to *secondary* instabilities associated with velocity shear. Jets bend and break up.
- Secondary instabilities tend to form quasi-steady flow patterns in symmetry direction, called "*zonal flows*"
- Zonal flows are subject to further instabilities, associated with the velocity shear and the (modified) equilibrium gradients
- The turbulence that limits the performance of fusion devices can be understood in these terms.
- The identification of the key processes came from sims and theory

Parasitic instability model: tokamak turbulence



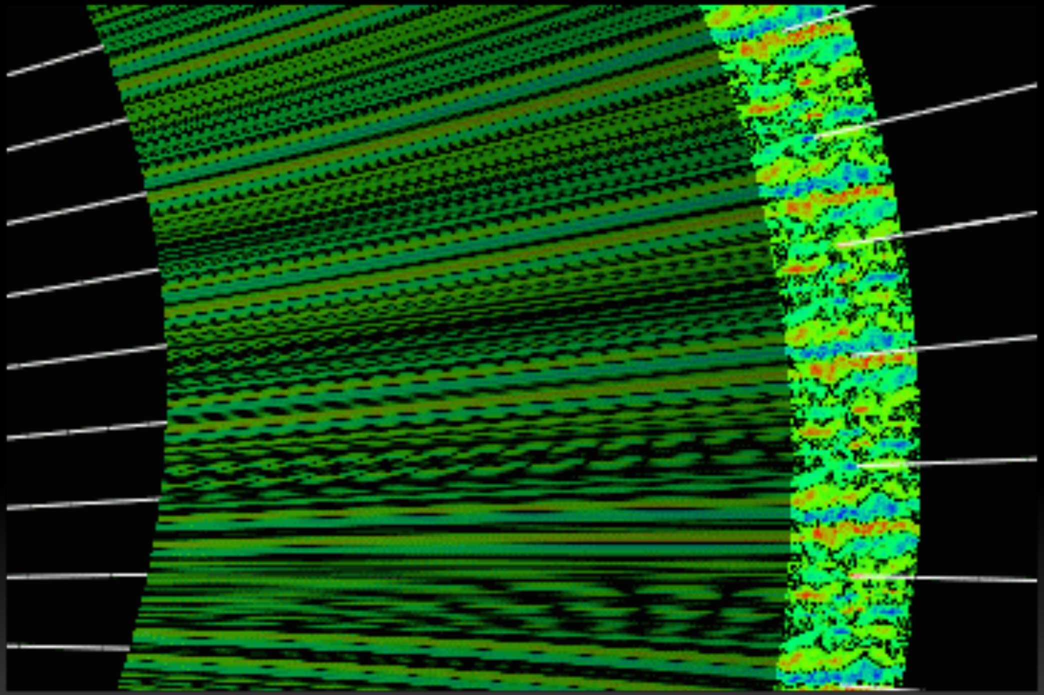
- Turbulence here is driven by electron temperature gradient
- Contours of the flow fields with green as zero flow, red and blue are maxima and minima
- White lines are the equilibrium magnetic field lines
- Instabilities most intense on the outside of the torus, where curvature and gradients conspire

Parasitic instability model: tokamak turbulence



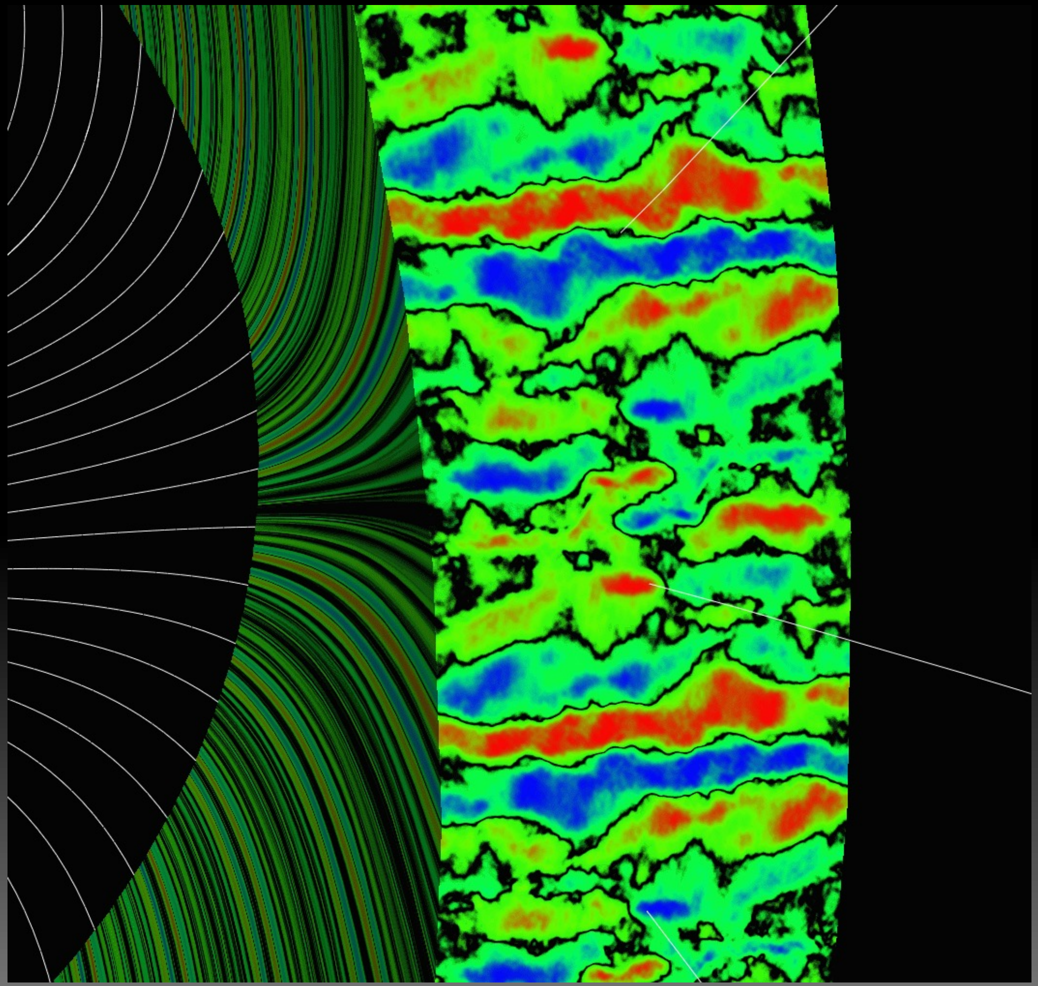
- Turbulence here is driven by electron temperature gradient
- Contours of the flow fields with green as zero flow, red and blue are maxima and minima
- White lines are the equilibrium magnetic field lines
- Instabilities most intense on the outside of the torus, where curvature and gradients conspire

Parasitic instability model: tokamak turbulence



- Zooming in…

Parasitic instability model: tokamak turbulence

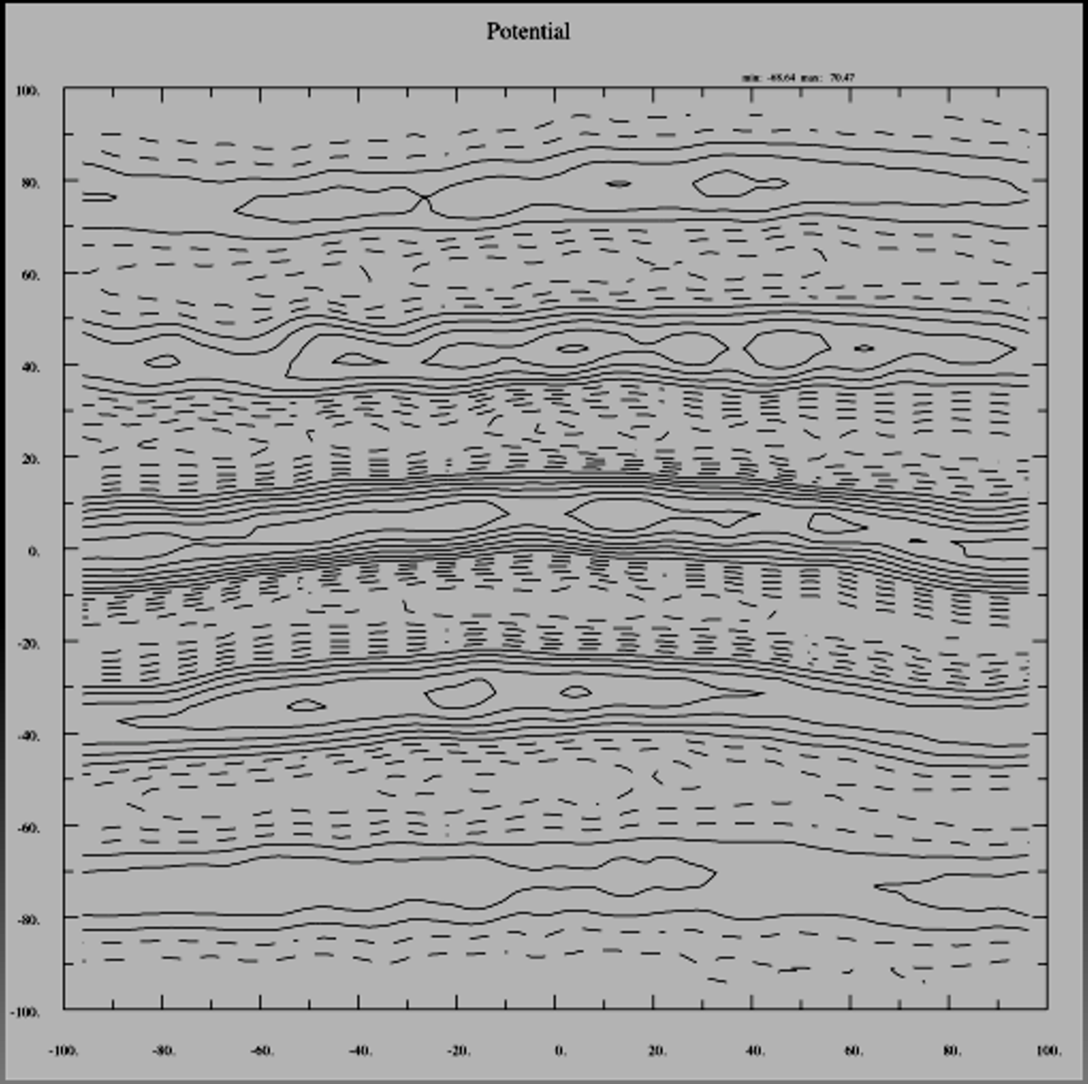


- The primary instability's jets are visible at this scale (typically sub-millimeter in size!)

Parasitic instability model: tokamak turbulence

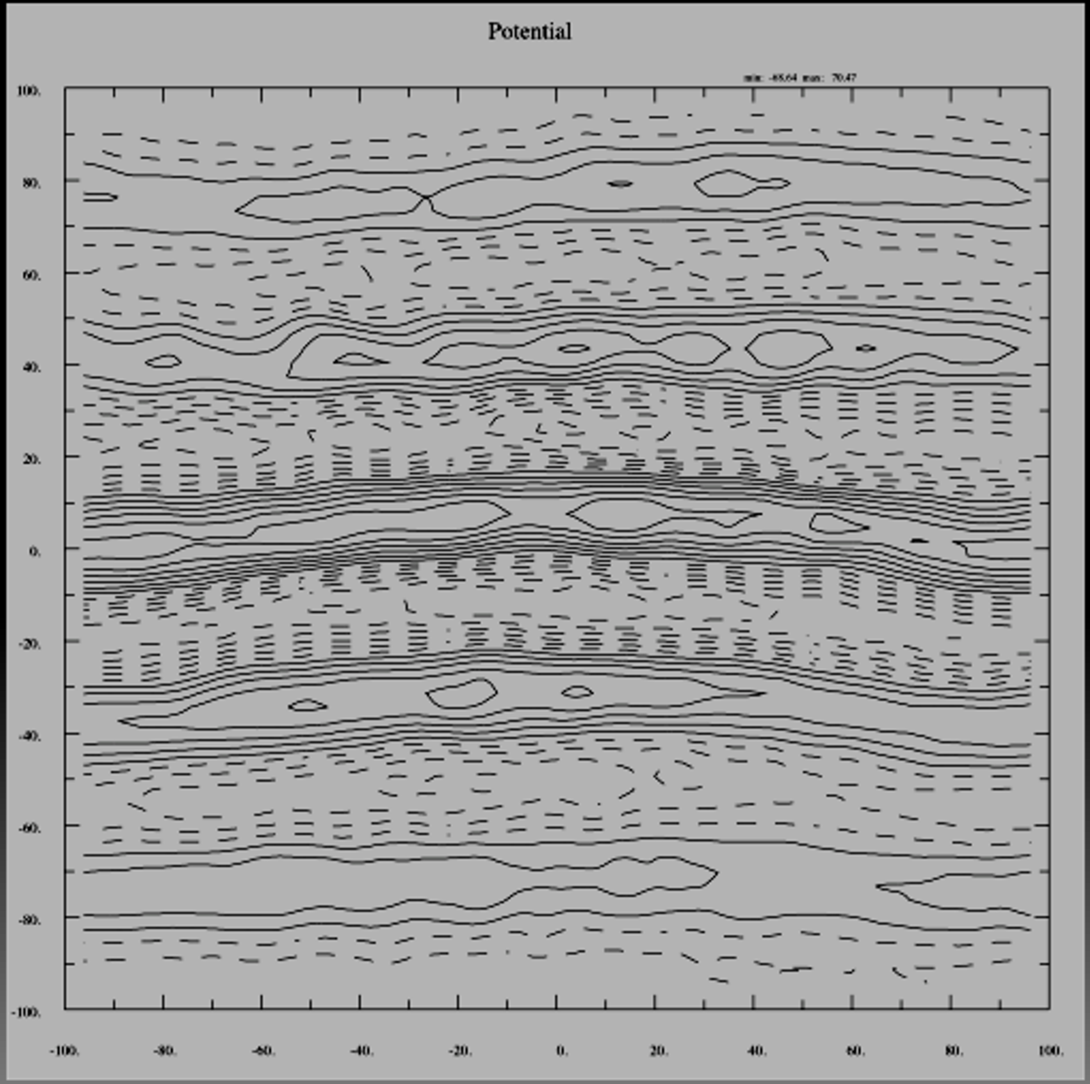
- What follows next are some images from a calculation of the dynamics of eddies that are larger (cm scales) but still very small compared to the size of the device
- Before, we were looking at eddies that are roughly the size of the electron gyroradius. Now, switching to eddies that are roughly the size of the ion gyroradius.

Parasitic instability model: tokamak turbulence



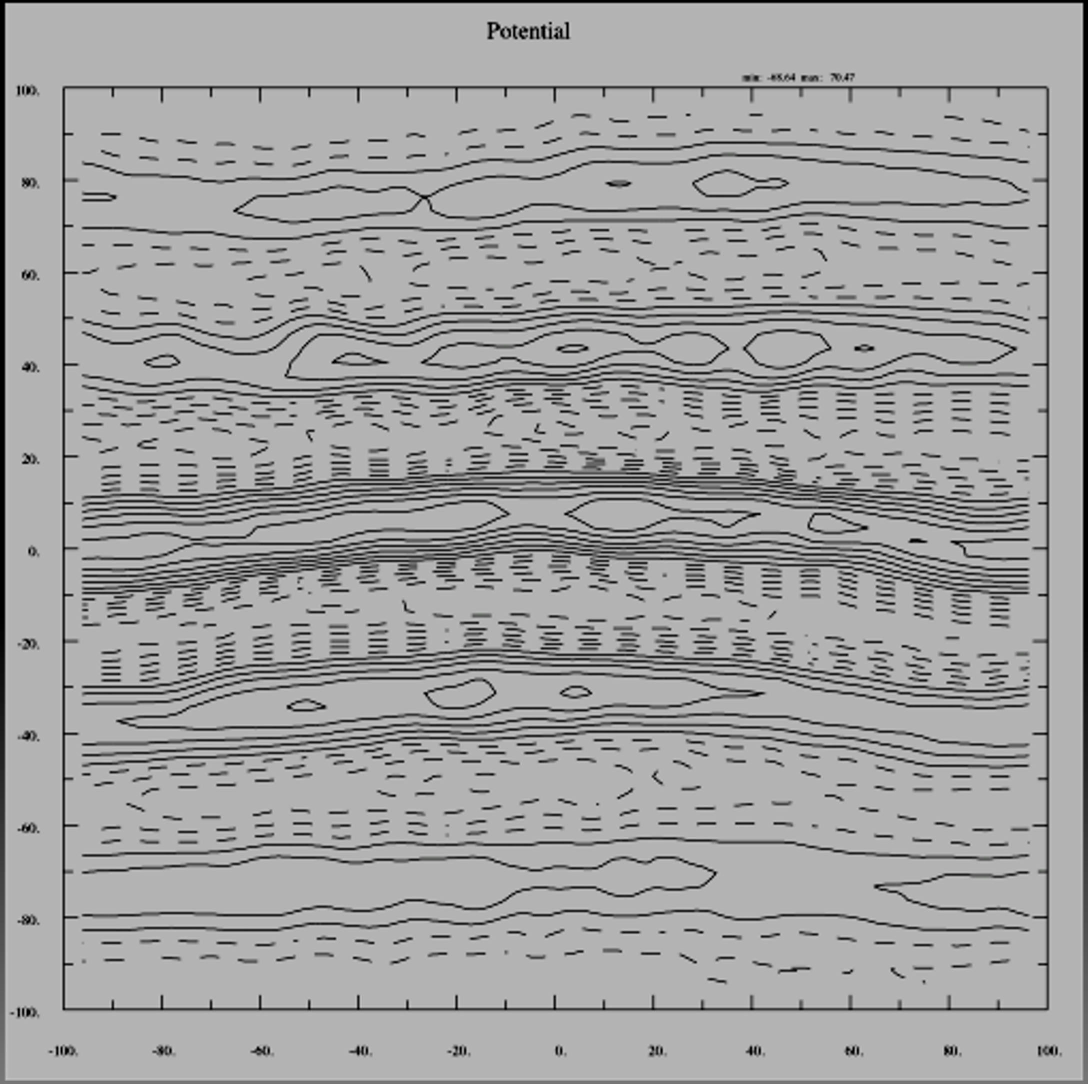
- Focus on the jets, switching to 2D contour plots

Parasitic instability model: tokamak turbulence



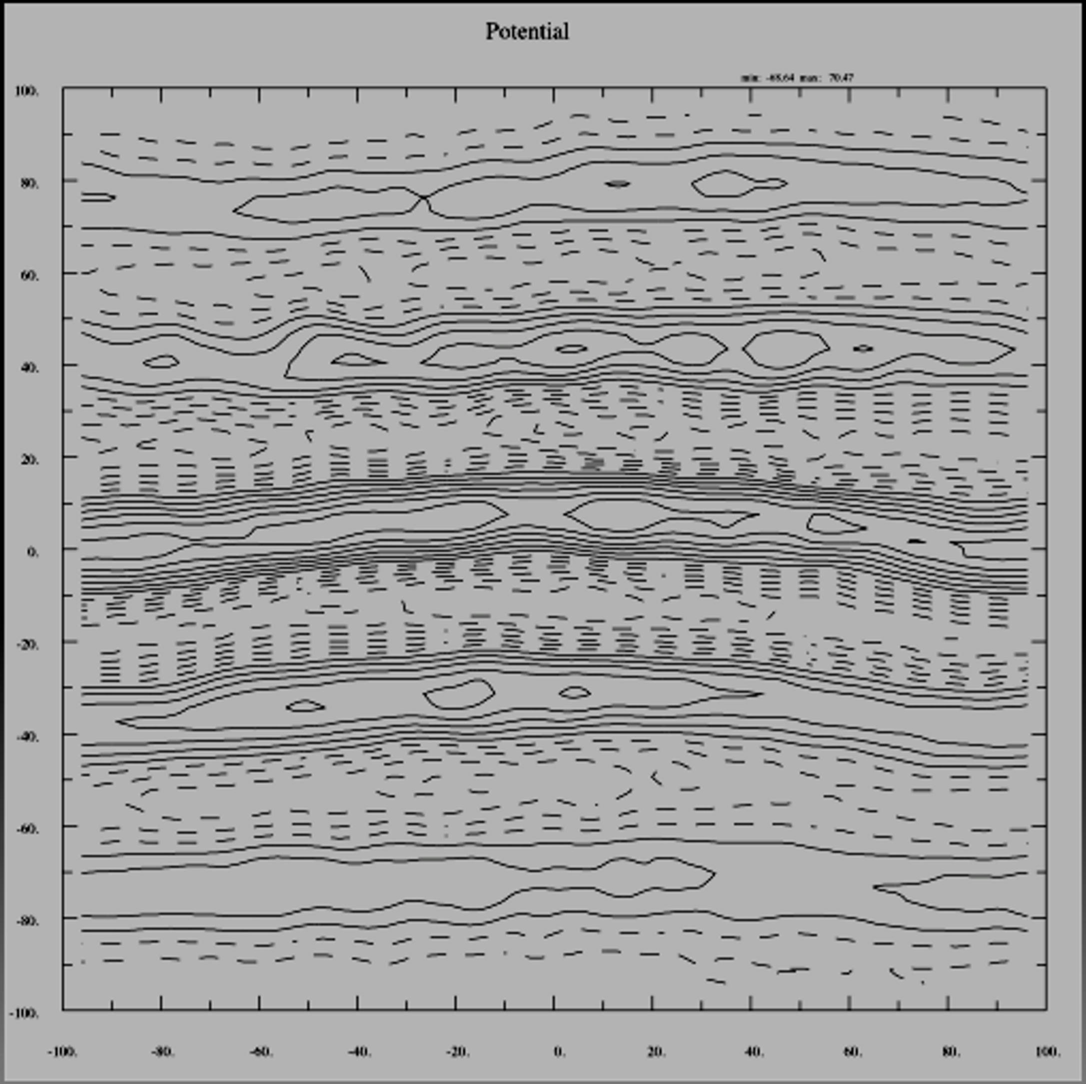
- Focus on the jets, switching to 2D contour plots
- Formation of skinny jets follows from shapes of most unstable linear instabilities + relatively weak nonlinear interactions in an appropriate sense

Parasitic instability model: tokamak turbulence



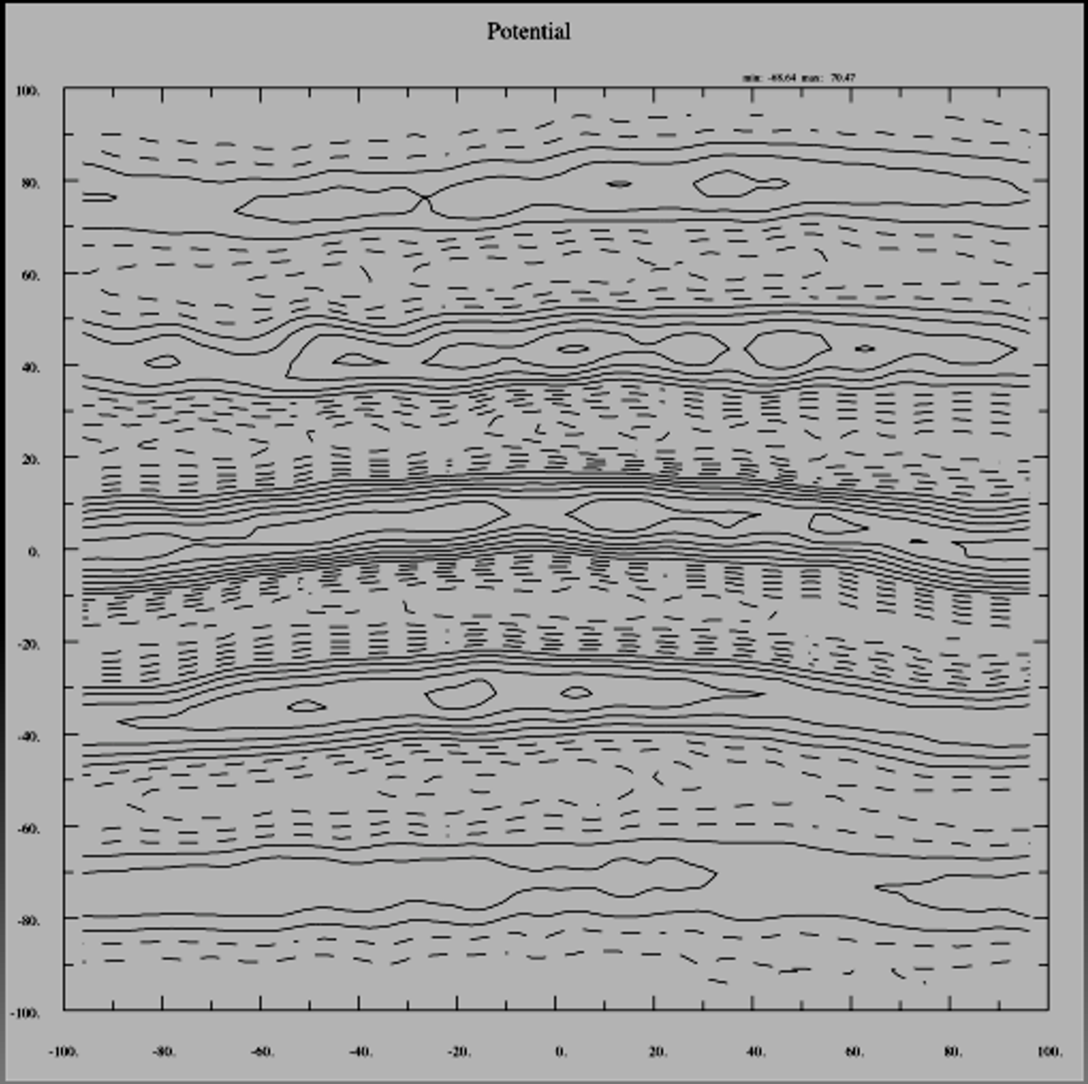
- Focus on the jets, switching to 2D contour plots
- Formation of skinny jets follows from shapes of most unstable linear instabilities + relatively weak nonlinear interactions in an appropriate sense
- Because this is an instability, the jet flows speed up exponentially in time. They get intense very quickly.

Parasitic instability model: tokamak turbulence



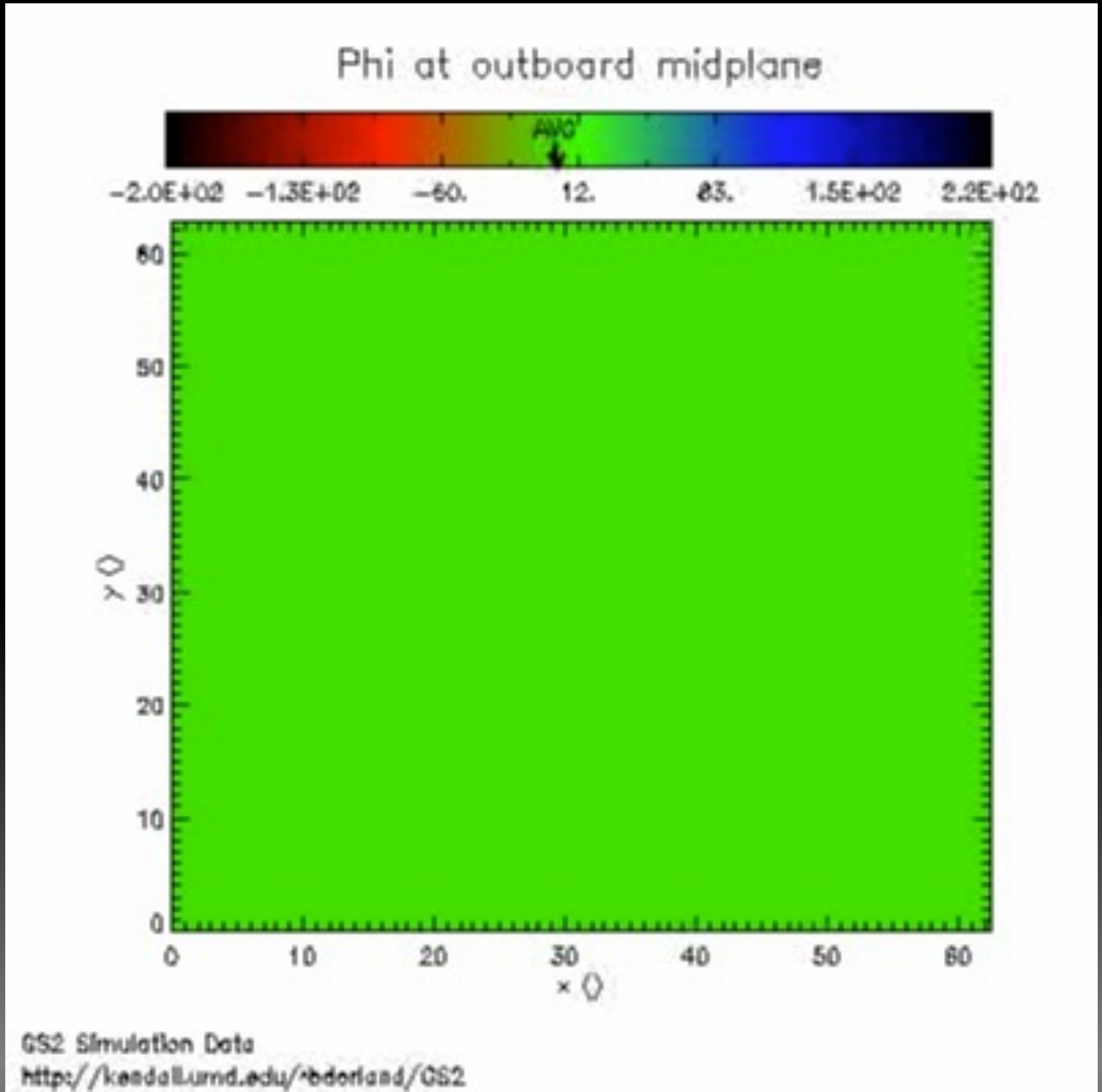
- Formation of skinny jets follows from shapes of most unstable linear instabilities + relatively weak nonlinear interactions in an appropriate sense
- Because this is an instability, the jet flows speed up exponentially in time. They get intense very quickly.
- These flows are themselves subject to instabilities like the ones we saw earlier

Parasitic instability model: tokamak turbulence



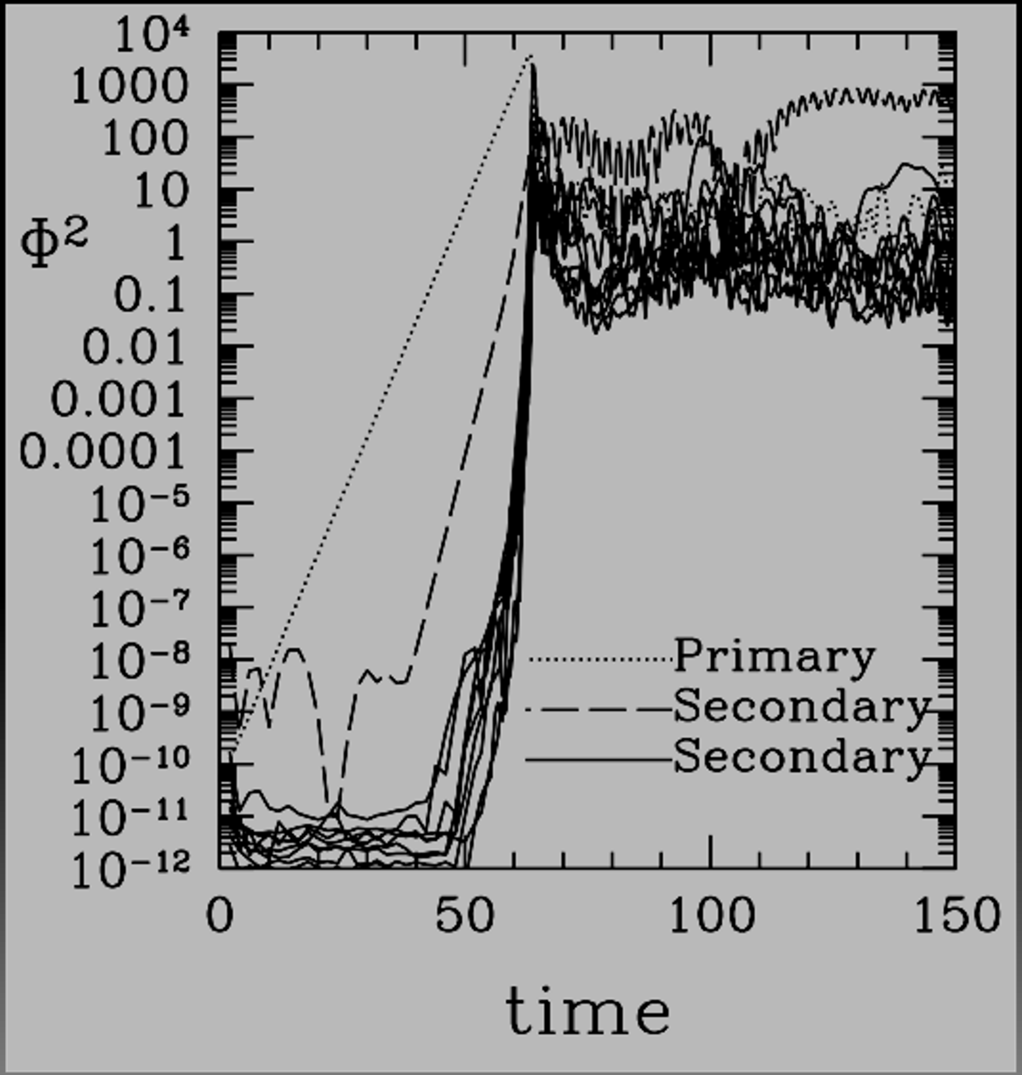
- Because this is an instability, the jet flows speed up exponentially in time. They get intense very quickly.
- These flows are themselves subject to instabilities like the ones we saw earlier
- Flows associated with the parasitic *secondary* instabilities speed up super-exponentially and quickly break up the jets

Parasitic instability model: tokamak turbulence



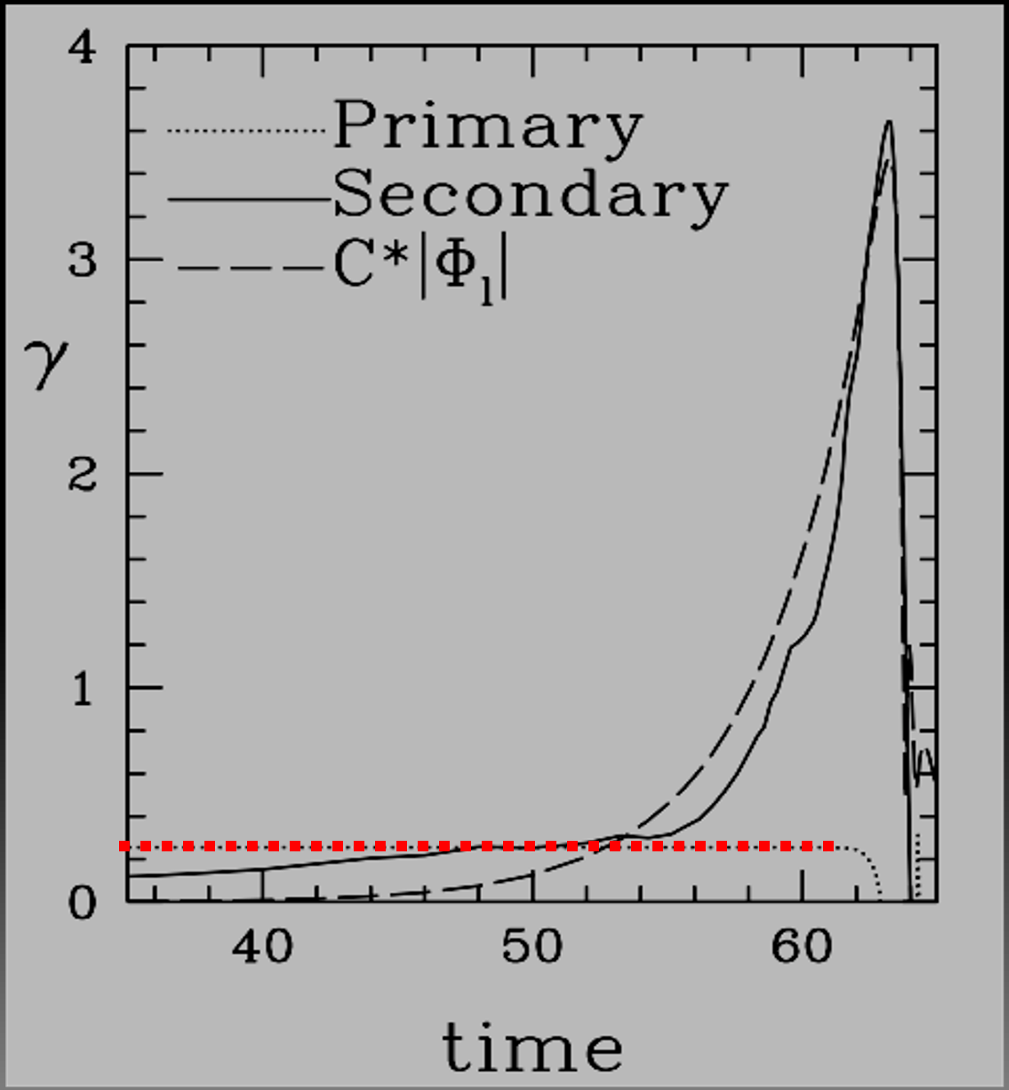
- Because this is an instability, the jet flows speed up exponentially in time. They get intense very quickly.
- These flows are themselves subject to instabilities like the ones we saw earlier
- Flows associated with the parasitic *secondary* instabilities speed up super-exponentially and quickly break up the jets

Parasitic instability model: tokamak turbulence



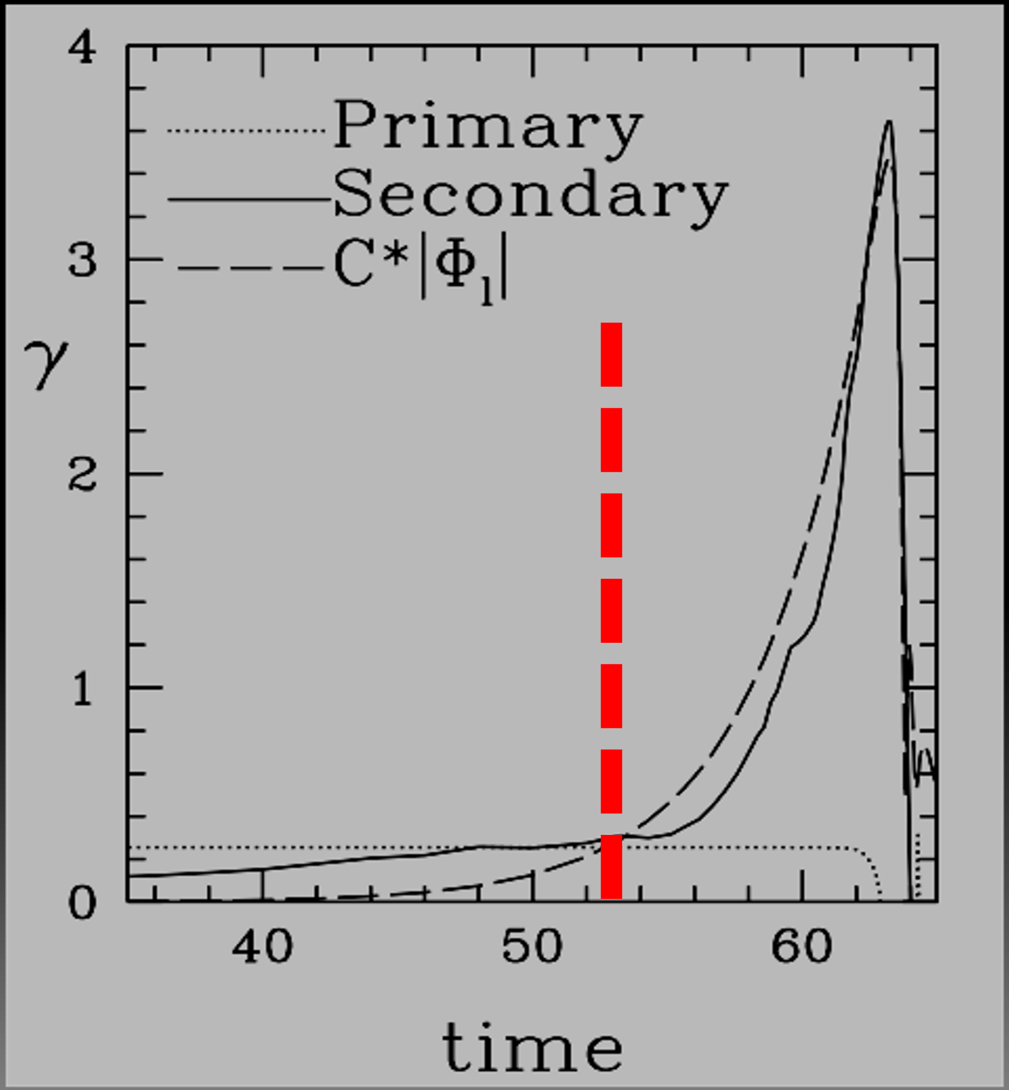
- Shown: various Fourier harmonics of the velocity stream function (squared) as a function of time
- Primary jets grow in intensity like $e^{\gamma t}$
- Secondary whorls are initially stable, then growing exponentially, then super-exponentially, like $e^{e^{\gamma t}}$
- Turbulent state results

Parasitic instability model: tokamak turbulence



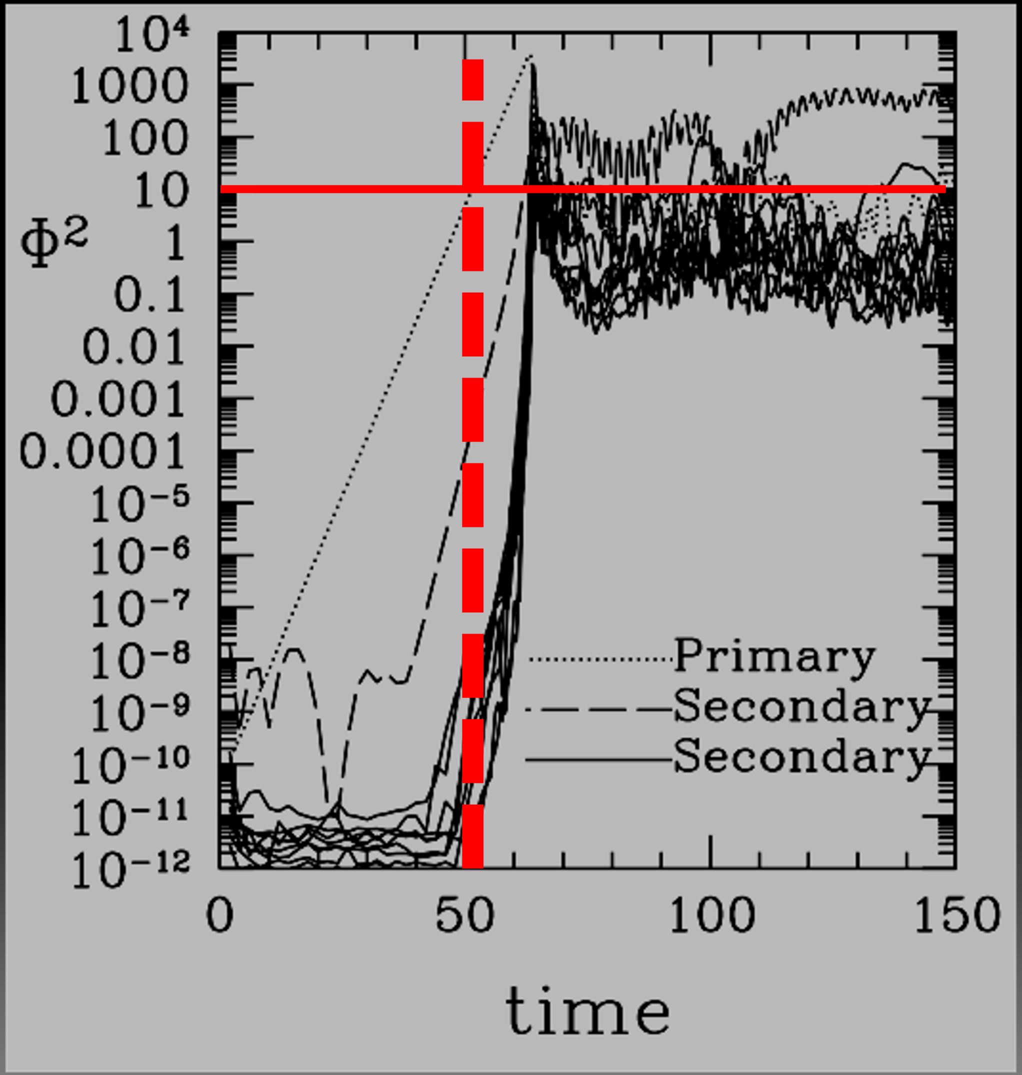
- Slopes of previous curves are the growth rates – a measure of the pace of intensification of the flow patterns
- Primary jets grow in intensity like $e^{\gamma t}$ so the growth rate does not change in time
- Secondary whorls have growth rates that are proportional to the intensity of the primary jet
- This is super-exponential intensification

Parasitic instability model: tokamak turbulence



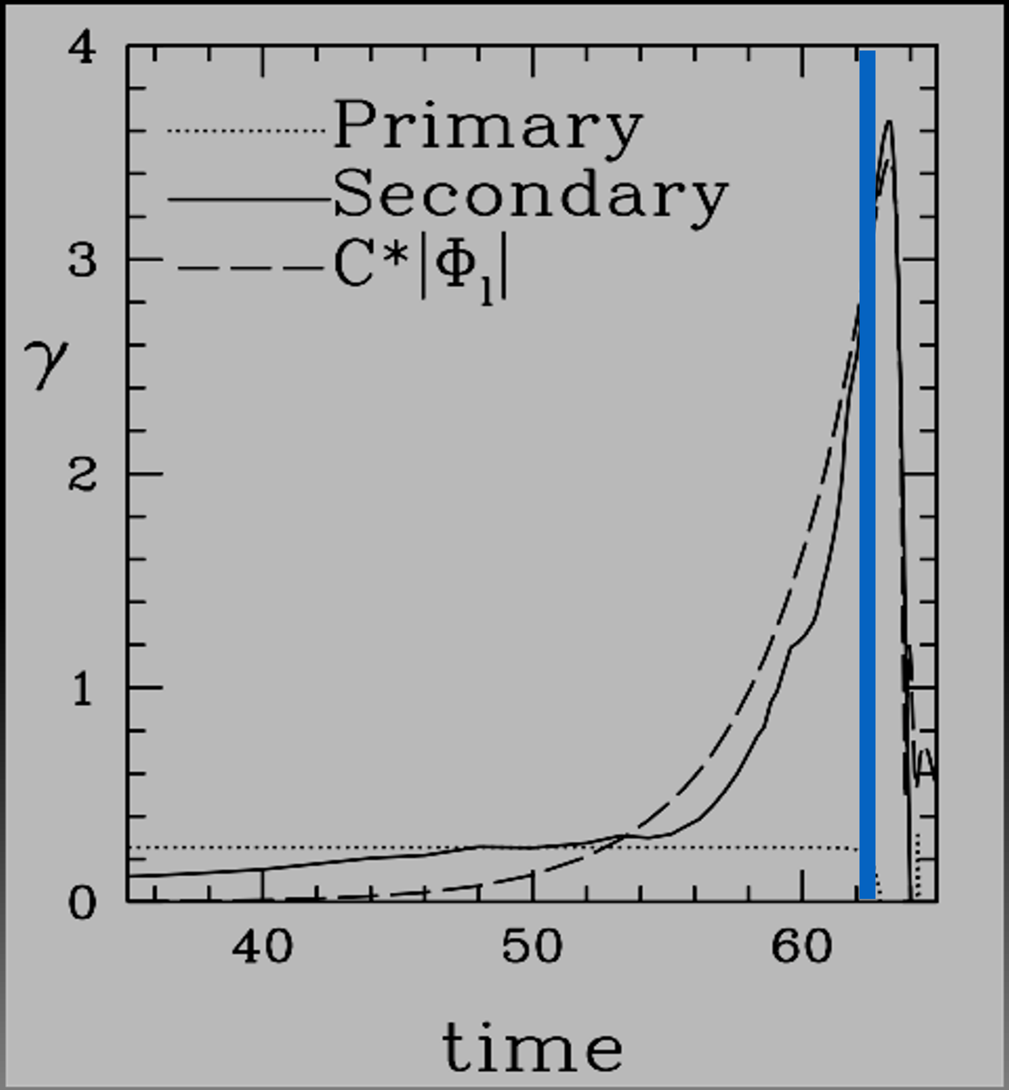
- Identify the time when the primary and secondary growth rates are equal

Parasitic instability model: tokamak turbulence



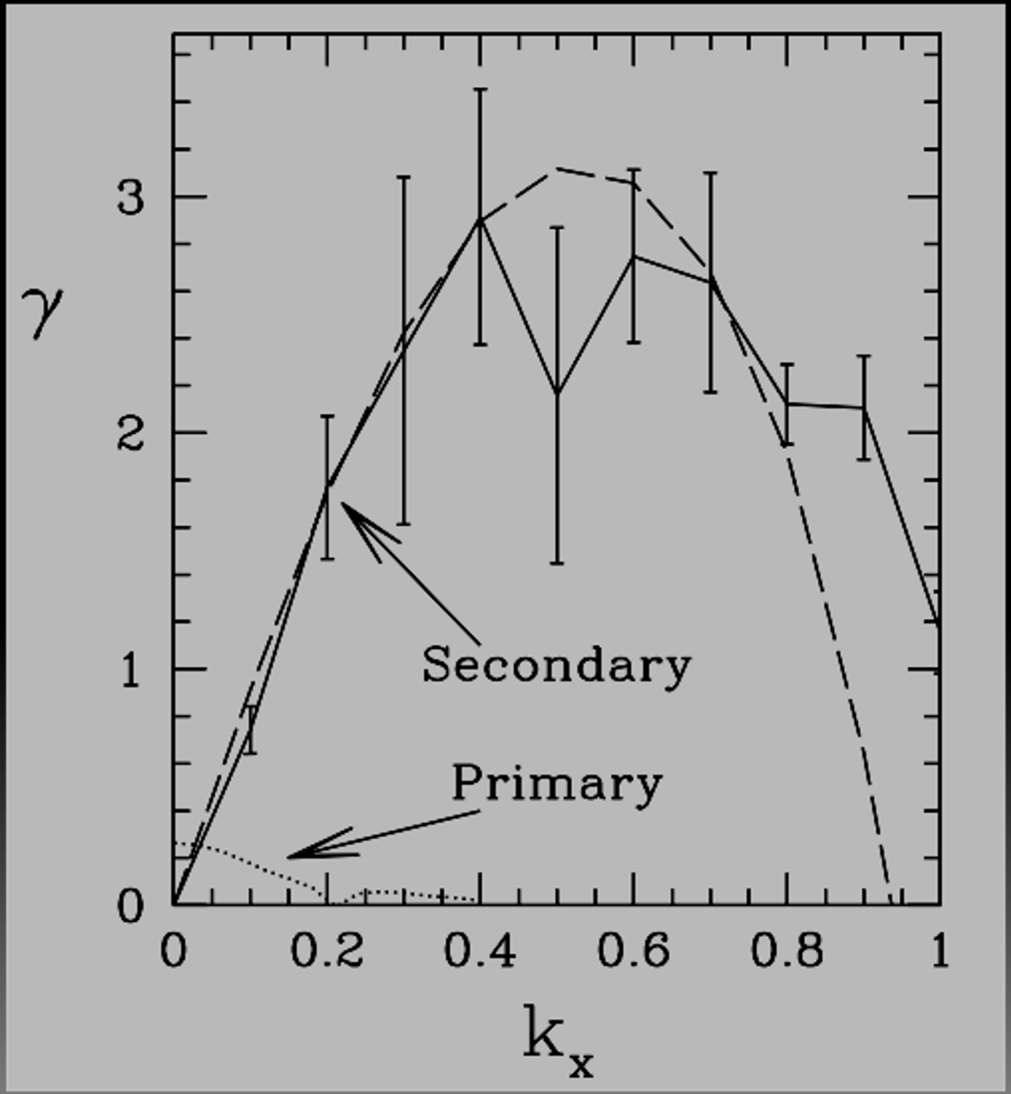
- Work backward to get an estimate of the amplitude of the jets when they are breaking up at the same pace they are intensifying
- Not a bad estimate of the intensity of the turbulence

Parasitic instability model: tokamak turbulence



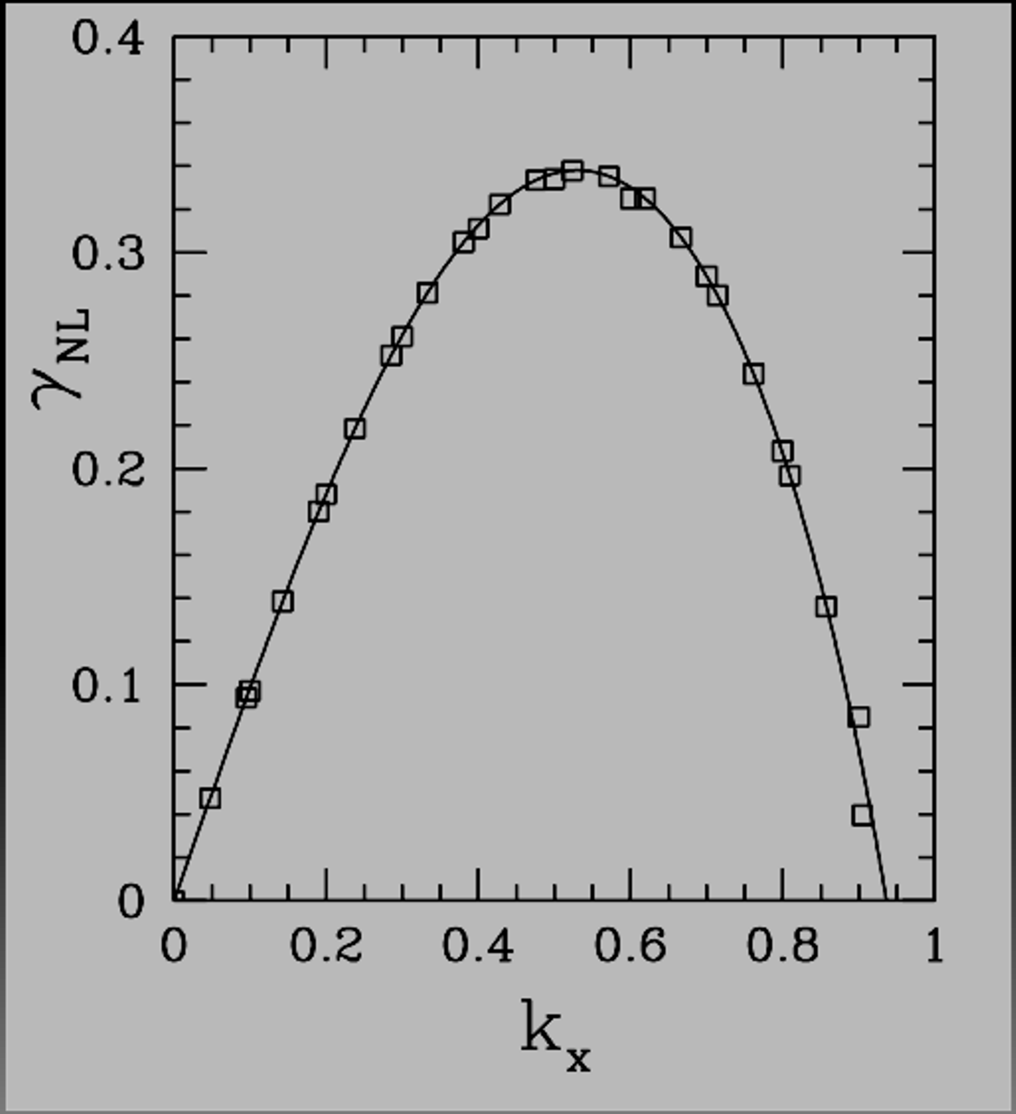
- Returning to the growth rates of the secondary. Can an analytic estimate of this growth be found?
- Yes, when the processes are proceeding at very different rates
- Focus on the time when the secondary whorls are intensifying must faster than the primary jets

Parasitic instability model: tokamak turbulence



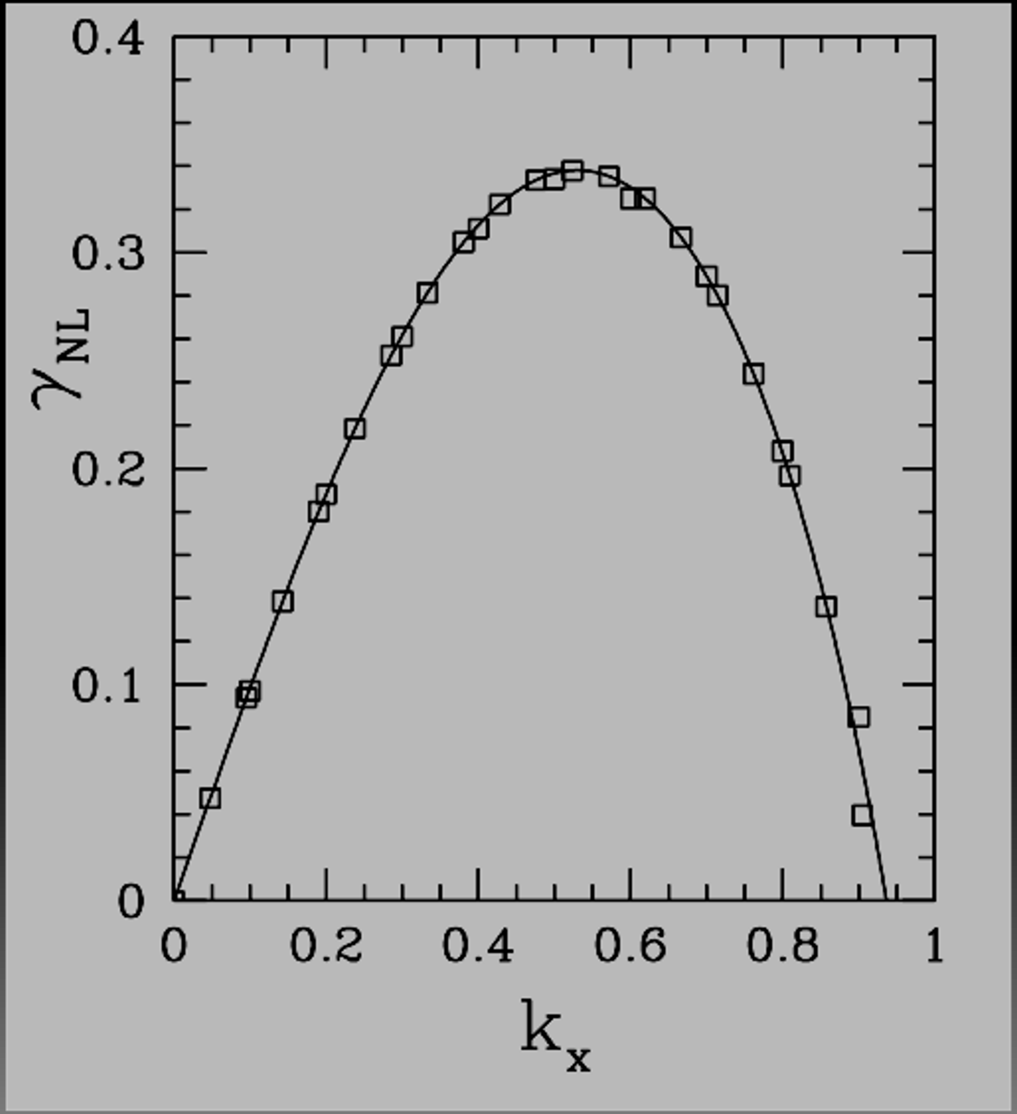
- As a function of the inverse radial wavelength, the observed growth rate agrees an analytic calculation

Parasitic instability model: tokamak turbulence



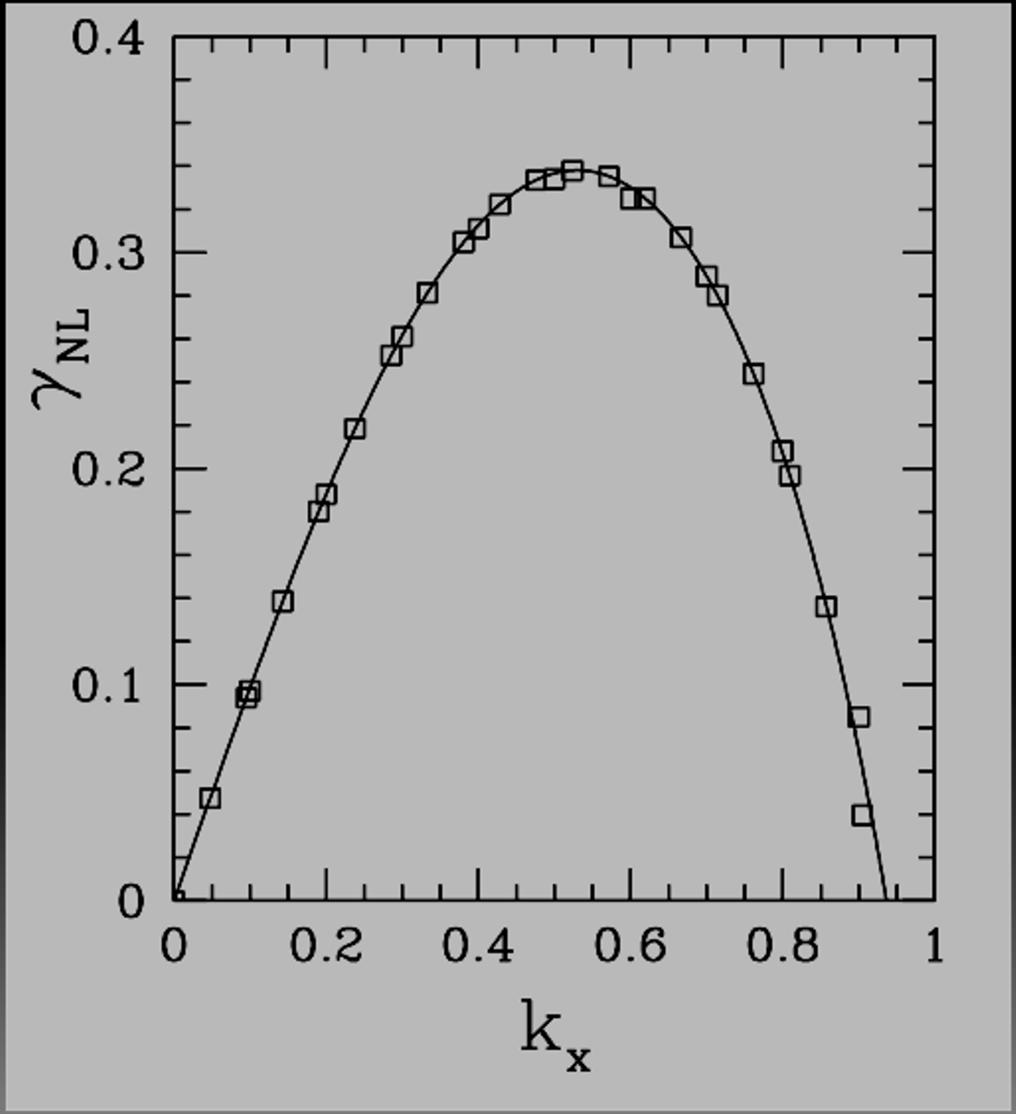
- As a function of the inverse radial wavelength, the observed growth rate agrees an analytic calculation
- In idealized case (just one jet, etc) the agreement is very good

Parasitic instability model: tokamak turbulence



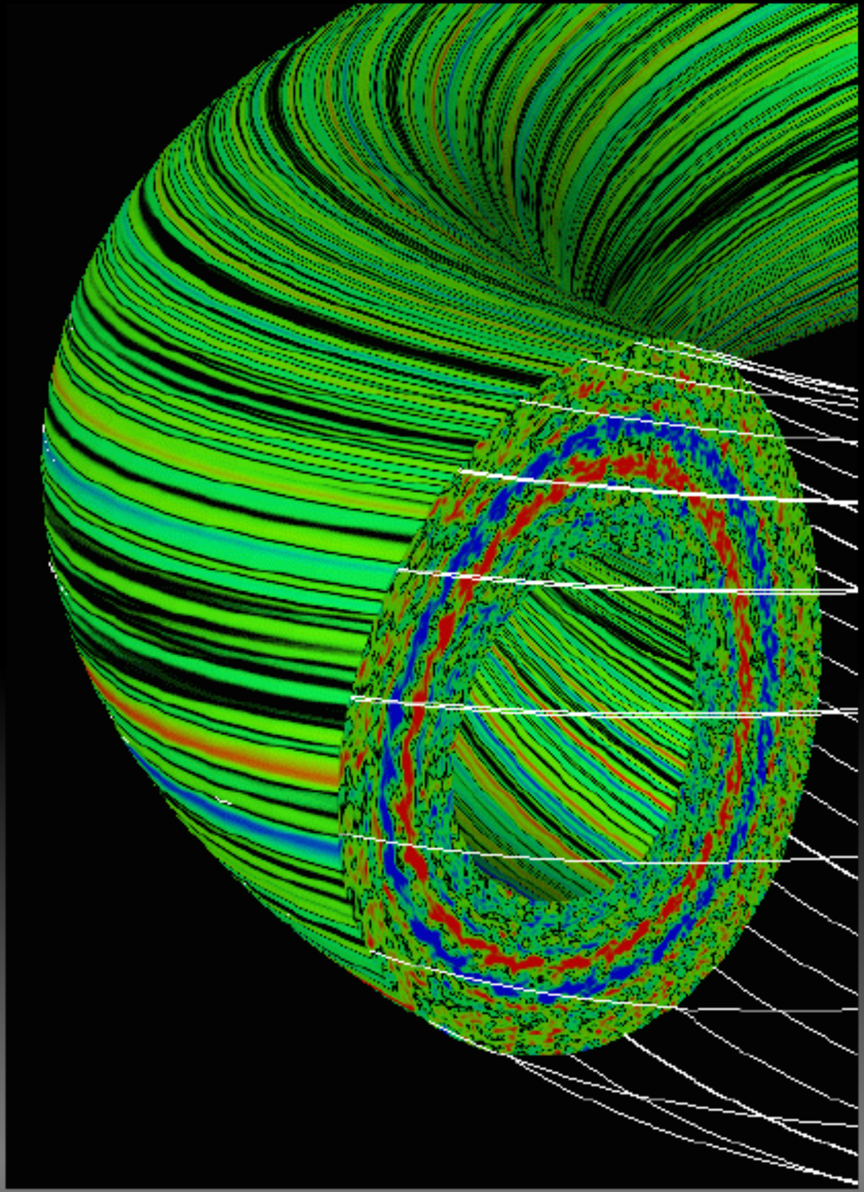
- As a function of the inverse radial wavelength, the observed growth rate agrees an analytic calculation
- In idealized case (just one jet, etc) the agreement is very good
- The calculations here are once again eigenvalue calculations, this time with two spatial dimensions

Parasitic instability model: tokamak turbulence



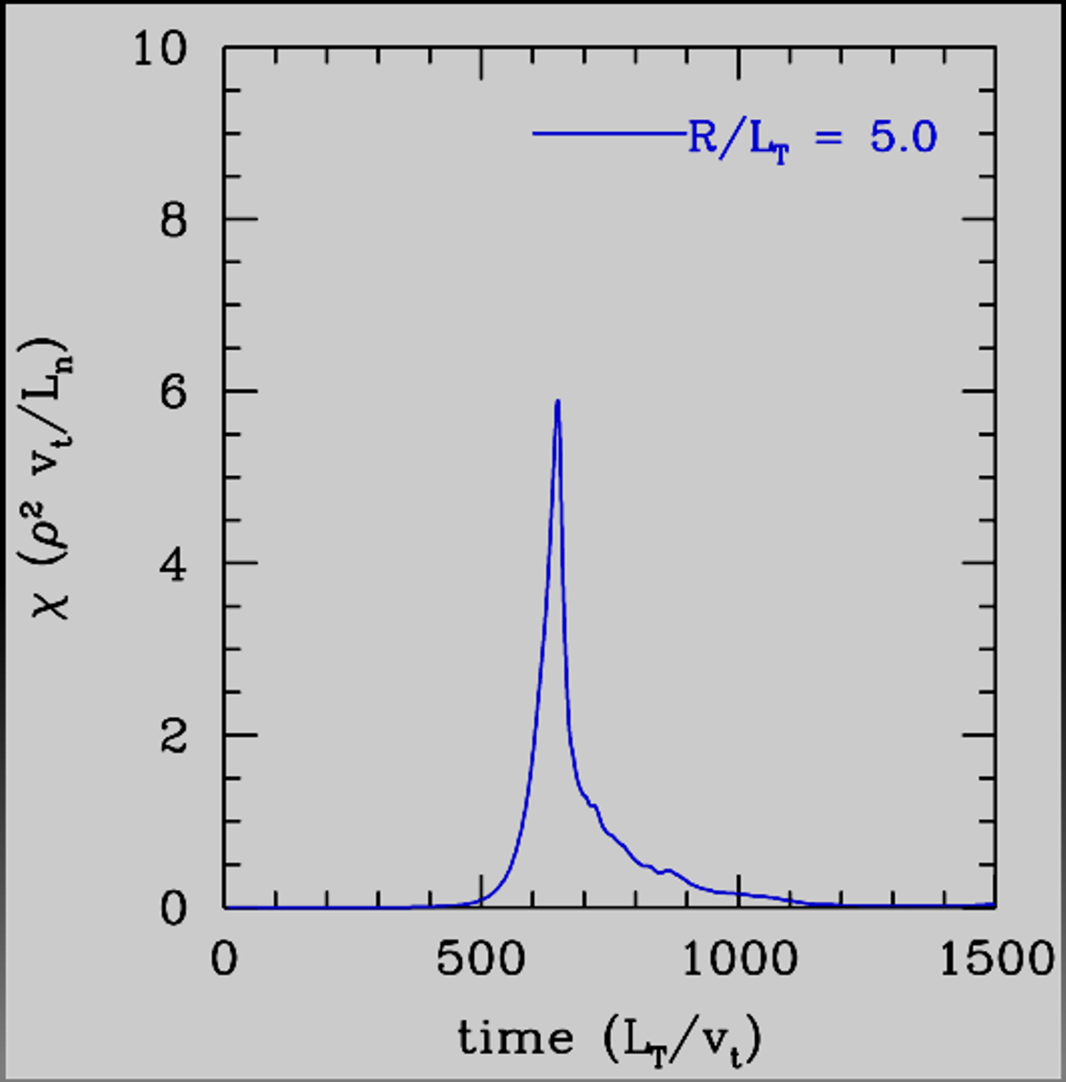
- As a function of the inverse radial wavelength, the observed growth rate agrees an analytic calculation
- In idealized case (just one jet, etc) the agreement is very good
- The calculations here are once again eigenvalue calculations, this time with two spatial dimensions
- **The component of the secondary eigenfunction that is constant in *y*-direction is special: this is a "zonal flow"**

Parasitic instability model: tokamak turbulence



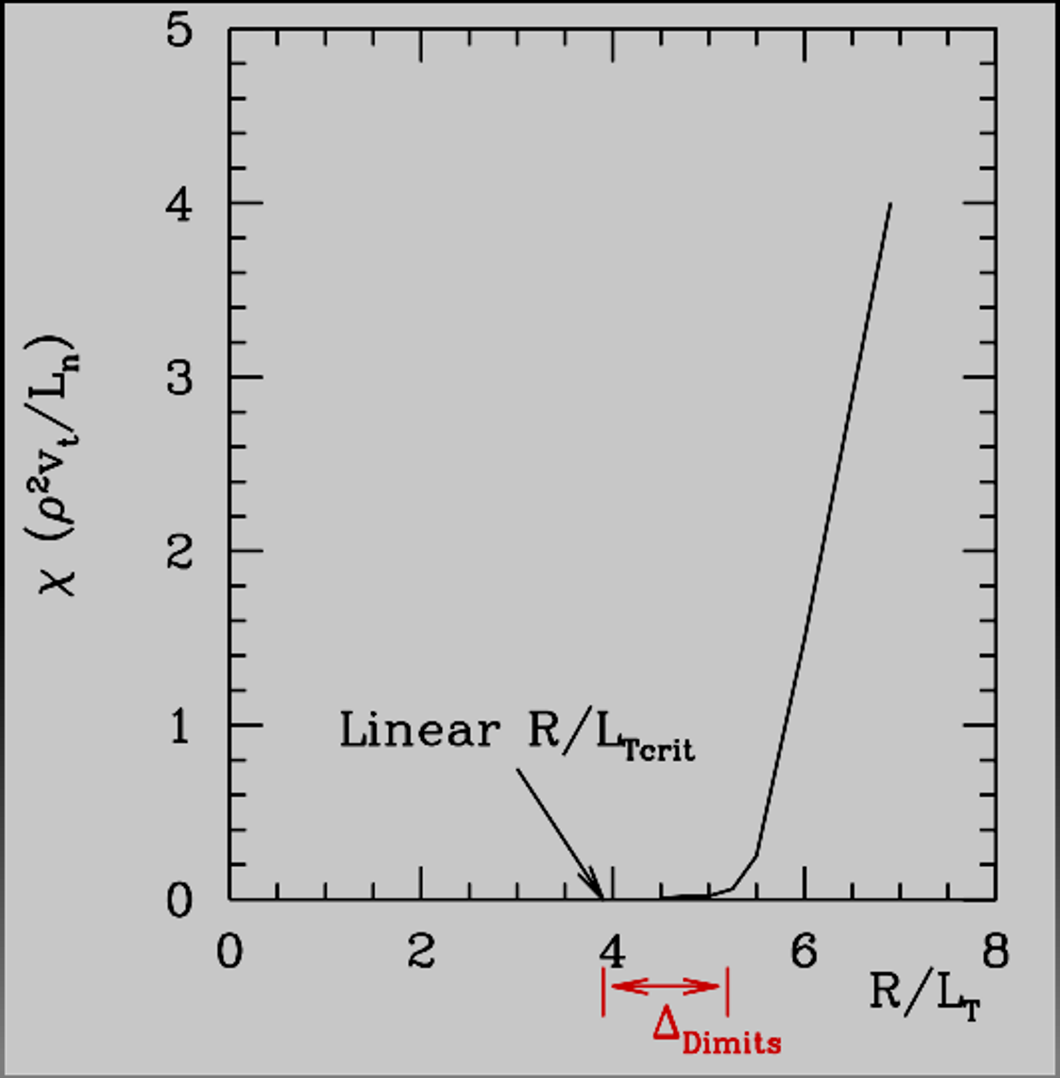
- An example zonal flow pattern is shown here (in red and blue)
- The speed of the zonal flow is a function of only radius (and time)

Parasitic instability model: tokamak turbulence



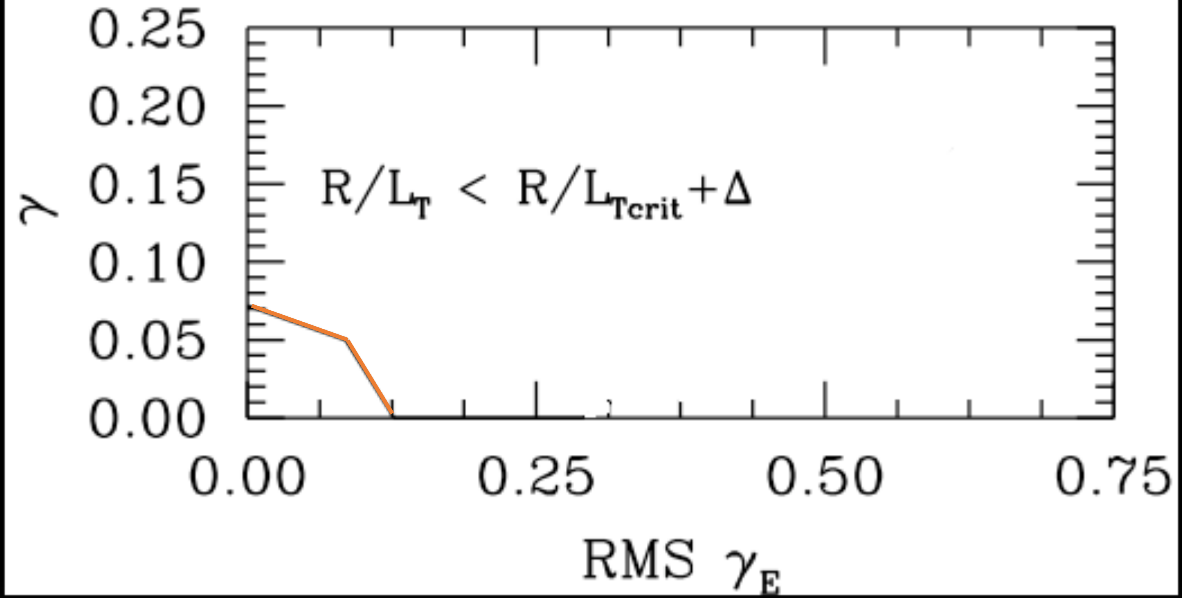
- This figure shows the turbulent diffusivity (a measure of the turbulence-induced thermal losses) as a function of time.
- It starts out growing but then is quenched by the zonal flows

Parasitic instability model: tokamak turbulence



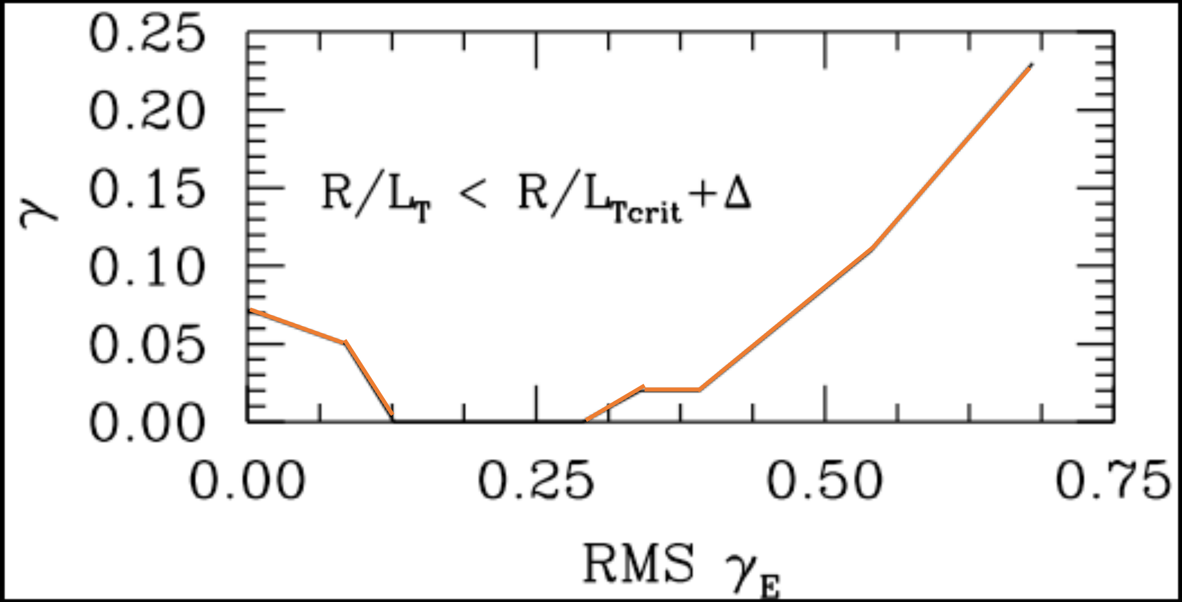
- Changing the gradient and repeating the calculation several times, we can display the turbulence-induced diffusivity as a function of how steep the temperature gradient is.
- There is a super-critical threshold I named the “Dimits shift”, because the quench was first observed by Andris Dimits (LLNL)
- What determines the super-critical threshold?

Parasitic instability model: tokamak turbulence



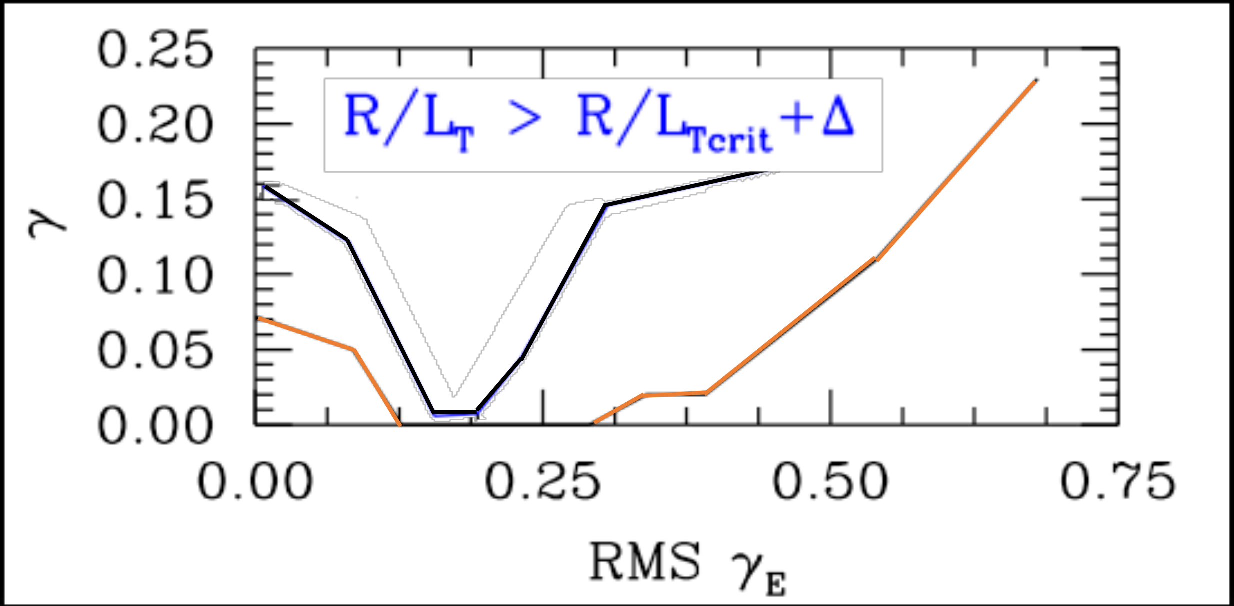
- Artificially increase the zonal flow intensity, which thereby increases the shearing that the zonal flows impose on the jets

Parasitic instability model: tokamak turbulence



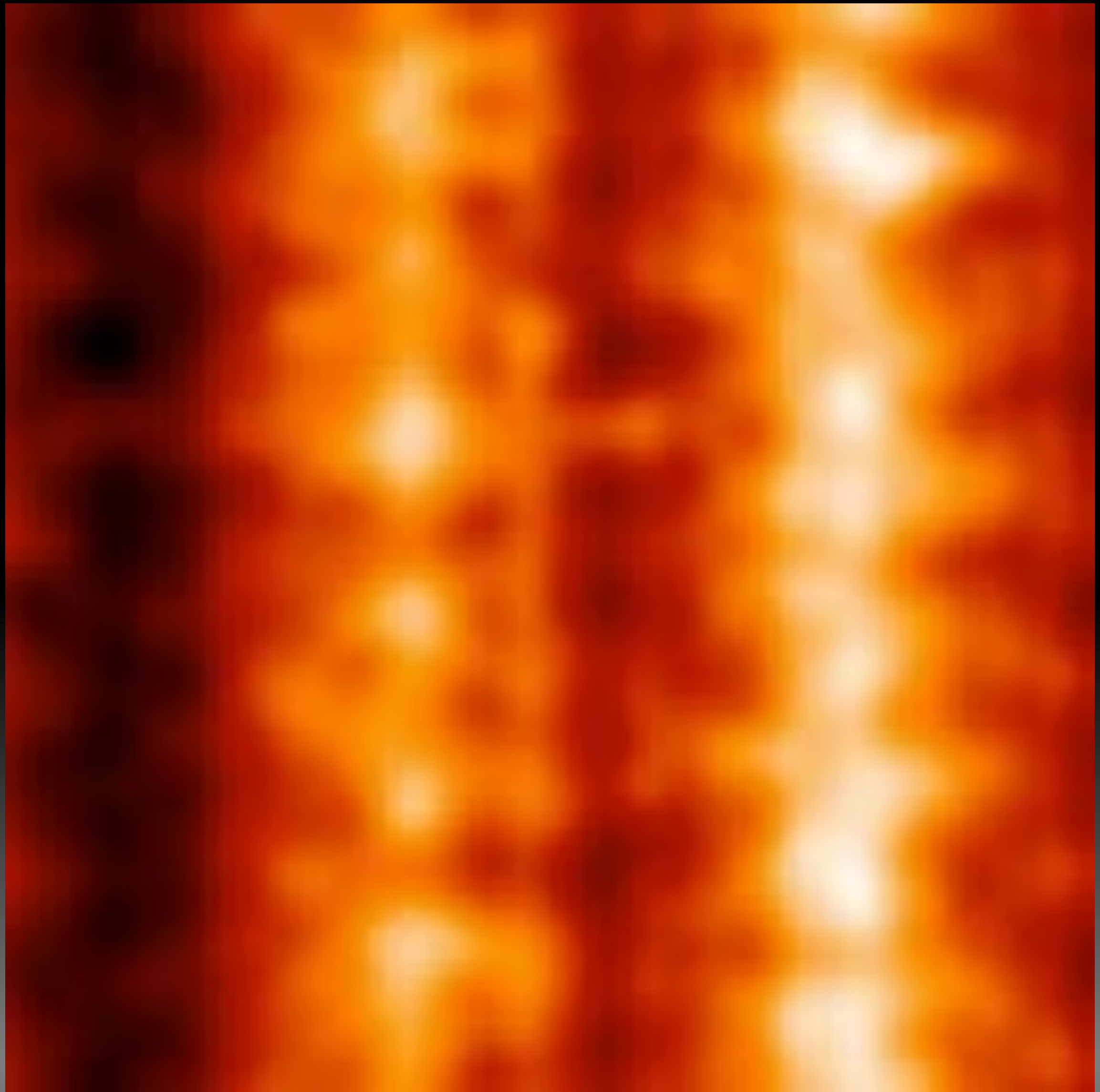
- Artificially increase the zonal flow intensity, which thereby increases the shearing that the zonal flows impose on the jets
- Eventually, further increases in the zonal flow intensity causes new instabilities

Parasitic instability model: tokamak turbulence



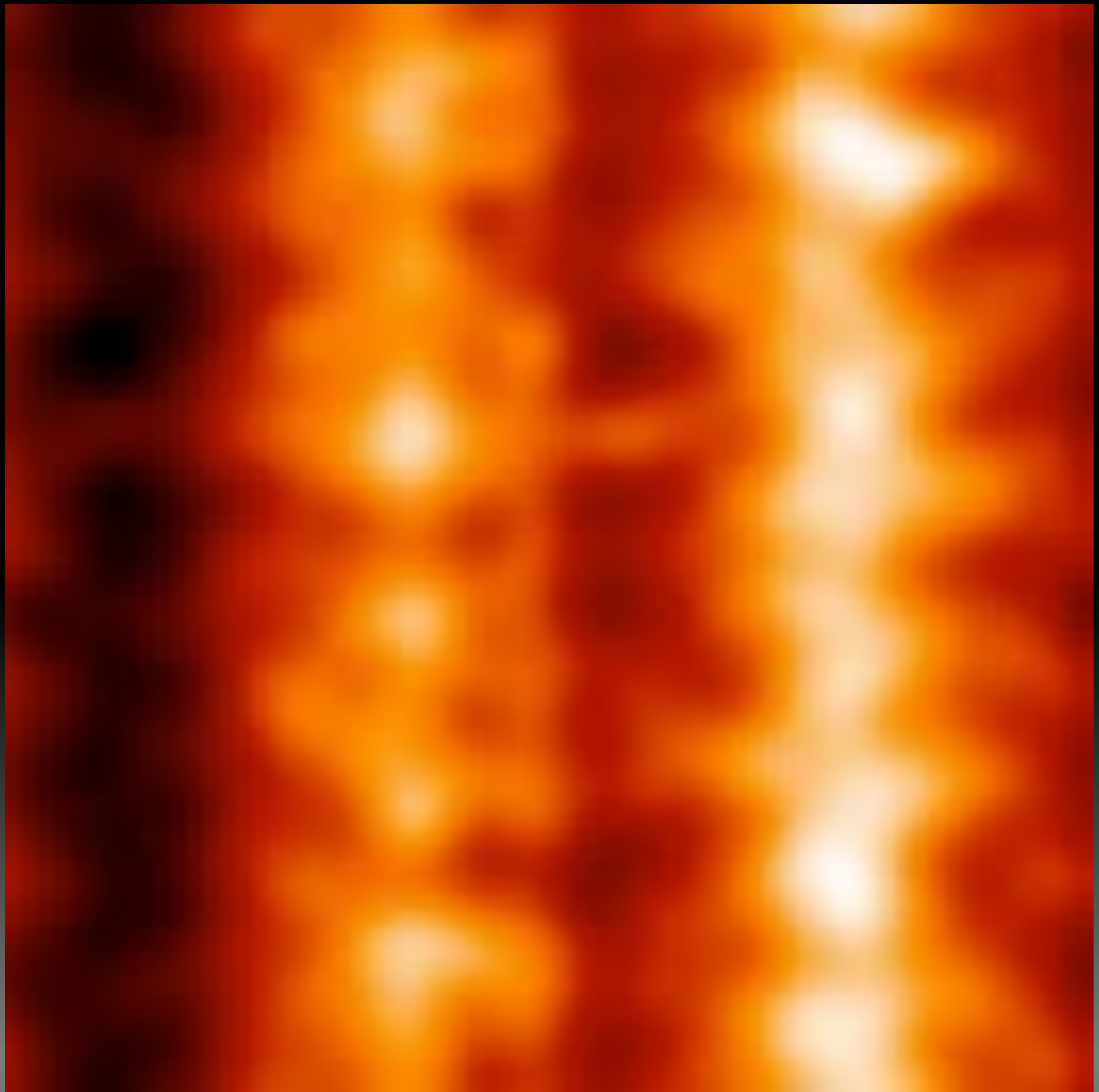
- Artificially increase the zonal flow intensity, which thereby increases the shearing that the zonal flows impose on the jets
- Eventually, further increases in the zonal flow intensity causes new instabilities
- Increasing the steepness of the temperature gradient removes the stable region altogether.

What processes govern the zonal flows?



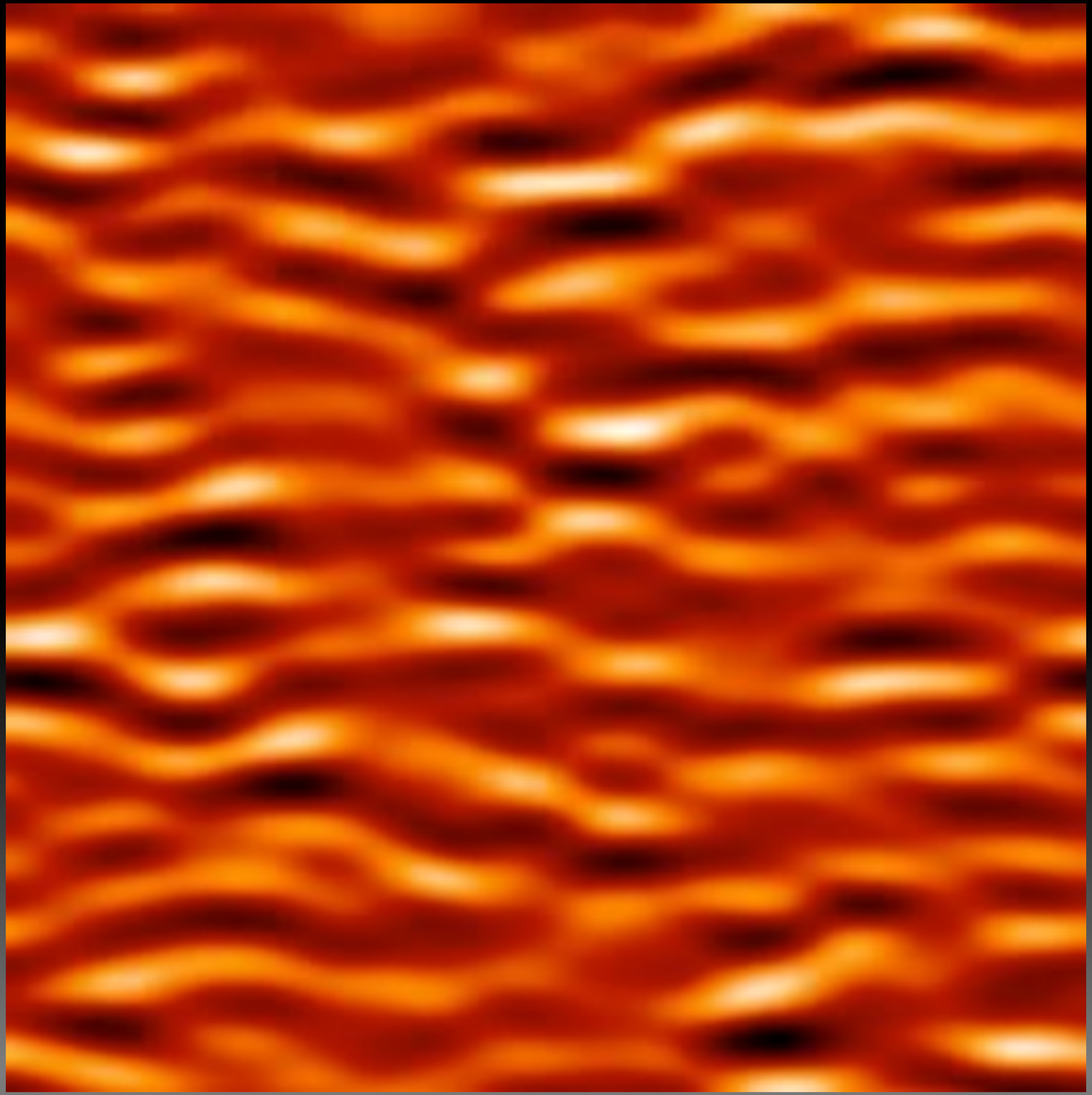
- Movies of a simulated velocity stream function for a Z-pinch geometry
- Long mean free path limit first

What processes govern the zonal flows?



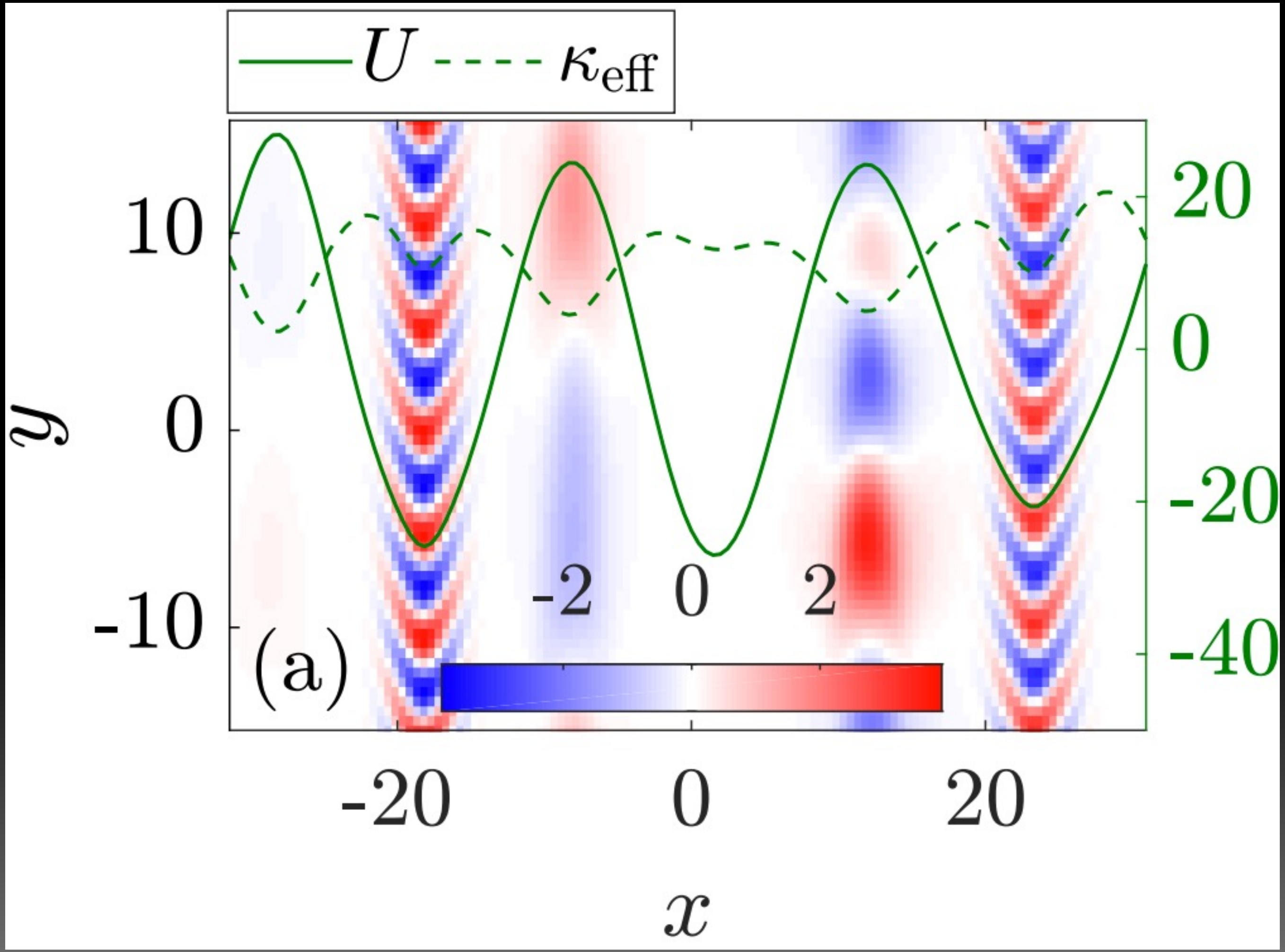
- Movies of a simulated velocity stream function for a Z-pinch geometry
- Shorter mean free path (theory is easier)

What processes govern the zonal flows?



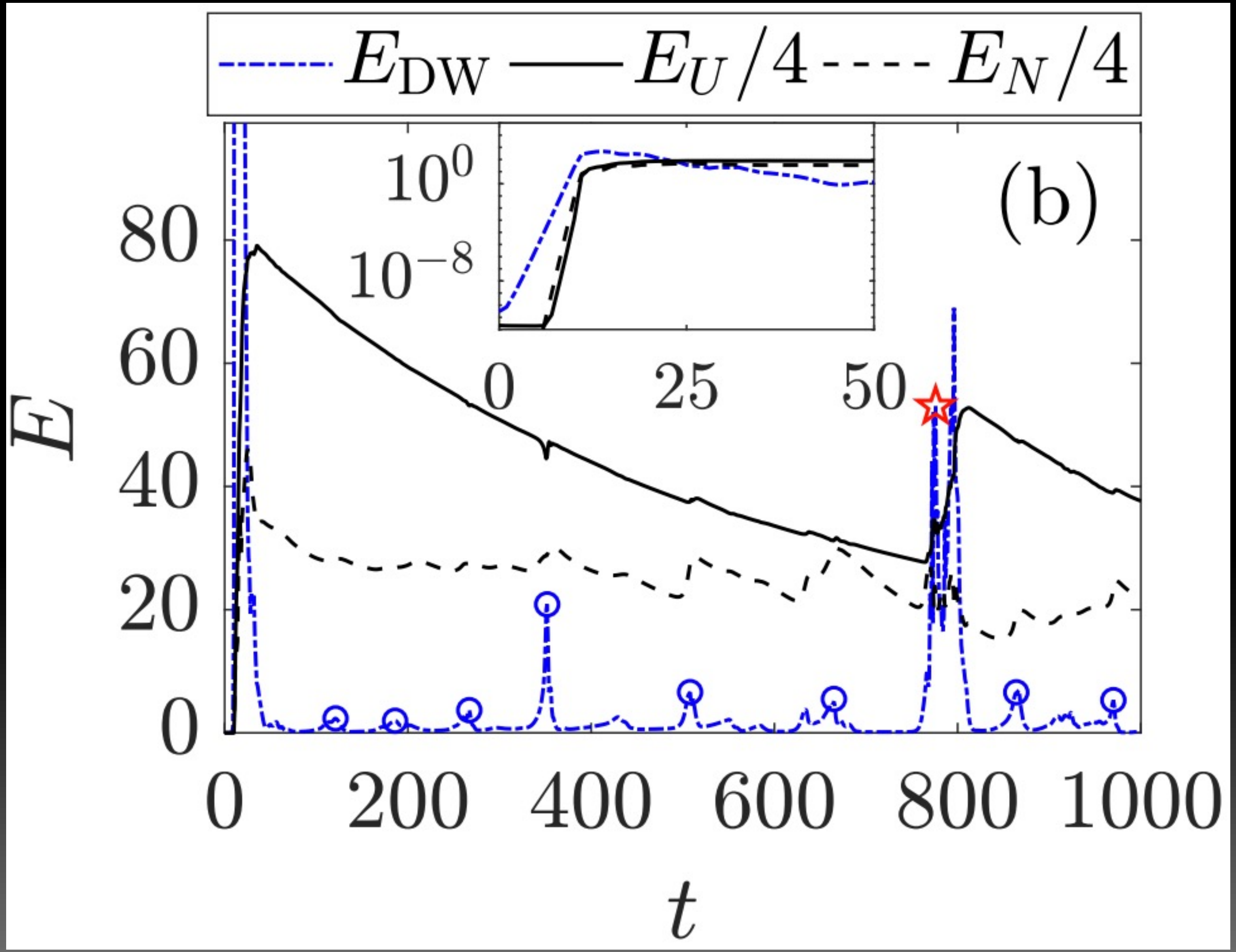
- Movies of a simulated velocity stream function for a Z-pinch geometry
- Shorter mean free path (theory is easier)
- Increase the equilibrium gradient

What processes govern the zonal flows?



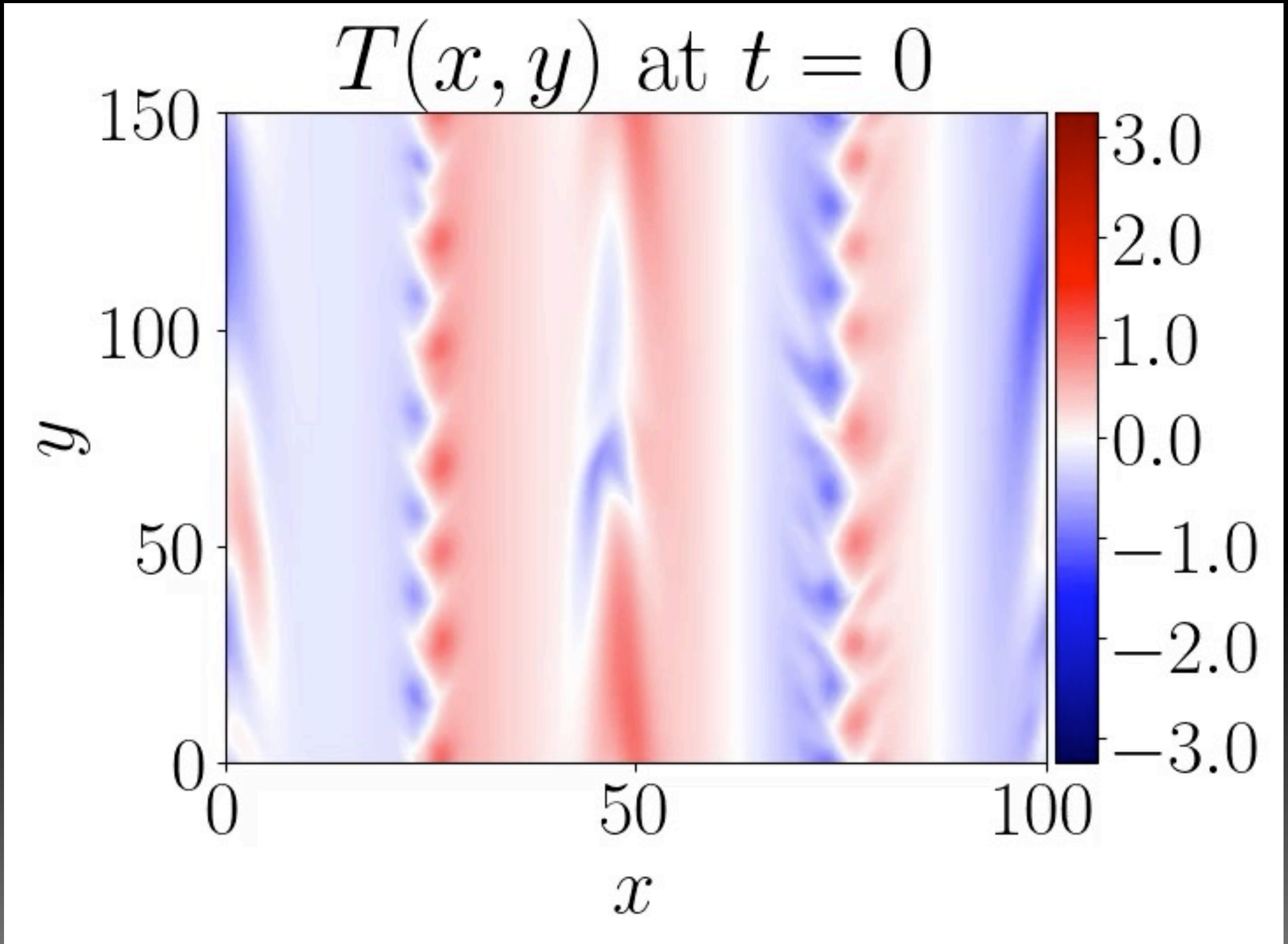
- Hongxuan Zhu, *et al*, recent work
- Sinusoidal flow has instabilities concentrated where the shear is weak
- There are major differences between flow peaks and flow valleys
- Theory for these patterns worked out, focusing on the flows themselves

What processes govern the zonal flows?



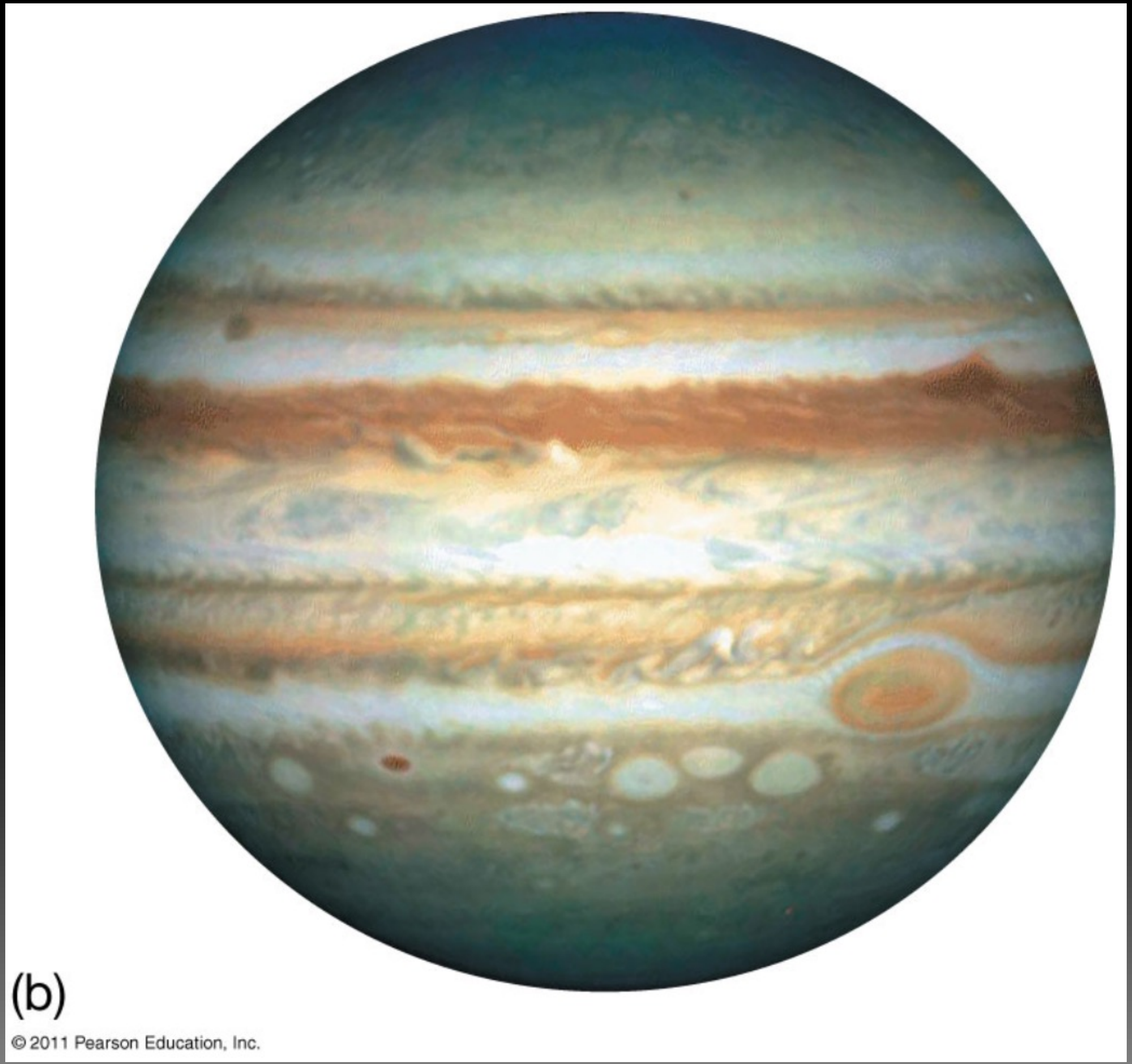
- Hongxuan Zhu, *et al*, recent work
- Patterns of rapid bursts and slow decay
- Similar to predator-prey dynamics (rabbits and foxes)

What processes govern the zonal flows?



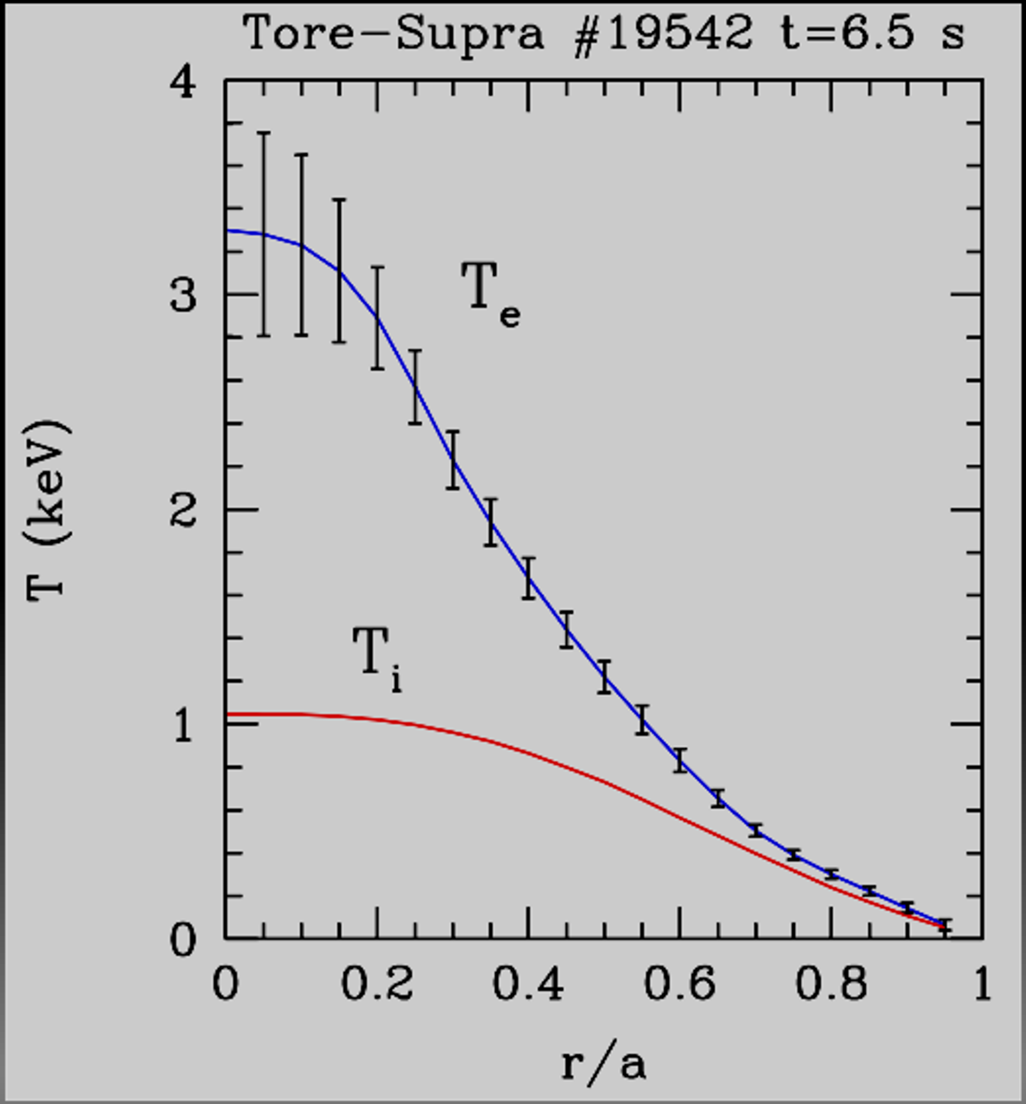
- Plamen Ivanov, *et al,* developed a theory that includes the temperature fluctuations
- With the temperature fluctuations included, one can see the influence of diamagnetic flows (in addition to ***E*** *x* ***B*** flows)
- When the temperature gradient is strong, the diamagnetic flows break up the zonal flows

Many papers have been written on zonal flows



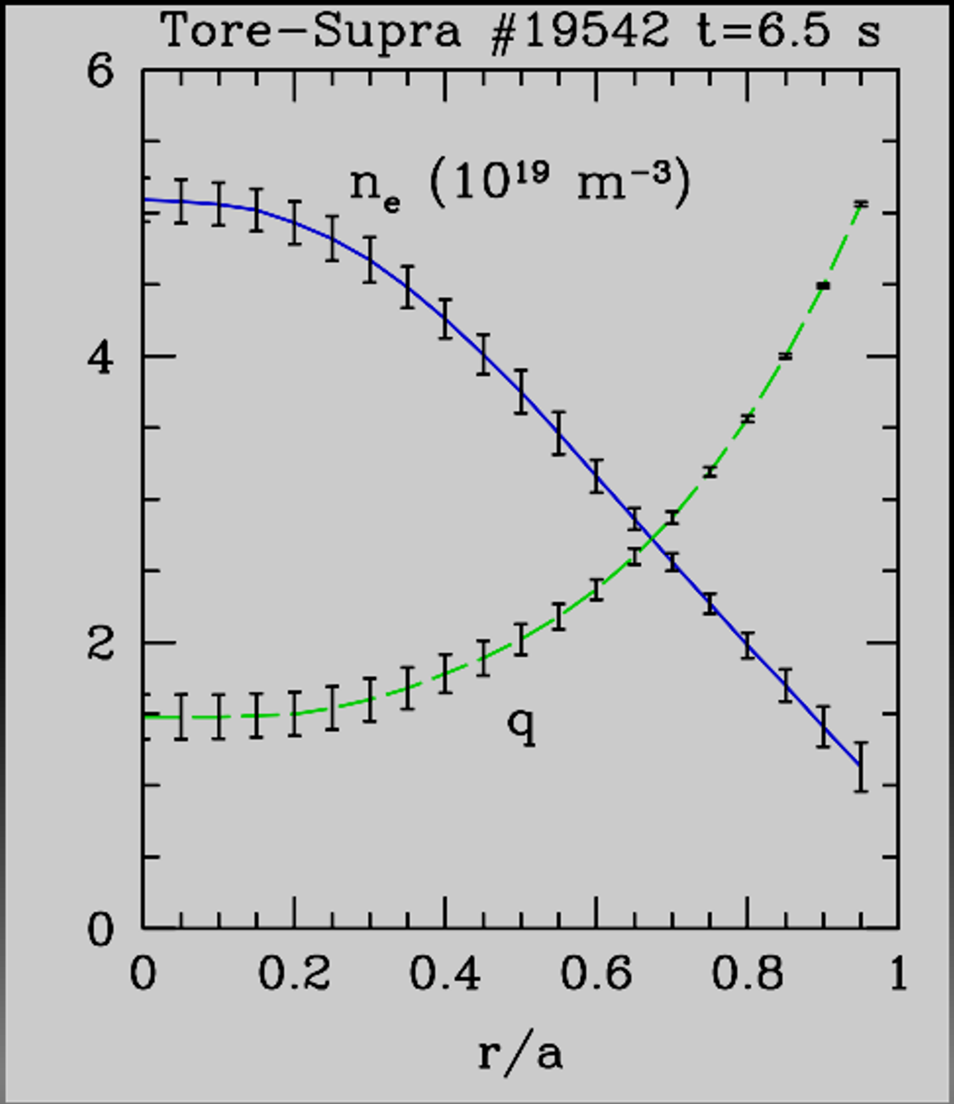
- I have been talking about zonal flows in plasma systems
- There are zonal flows in neutral fluids as well
- The dynamics of thin atmospheres on spheres is remarkably similar to plasma systems.

Simulations are phenomenologically rich



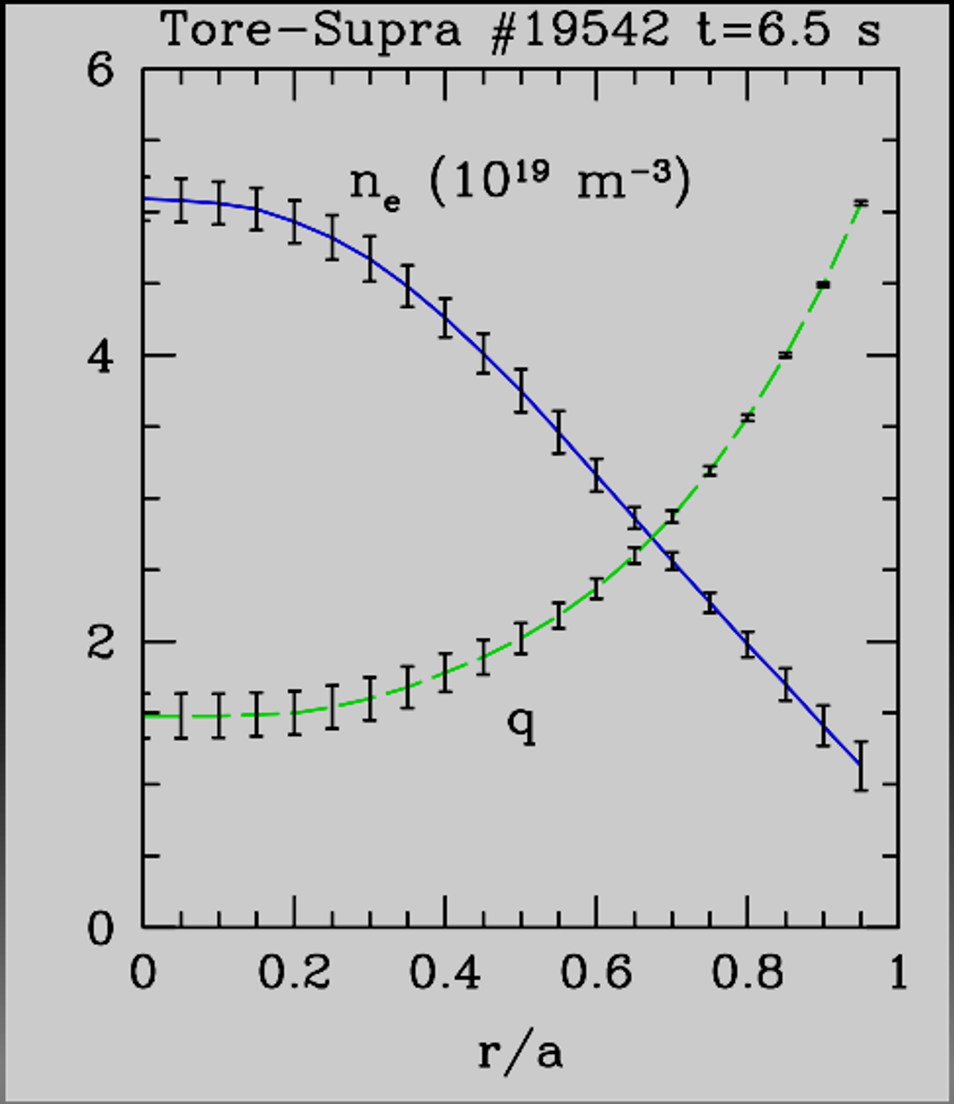
- Here are some temperature profiles from the Tore-Supra tokamak

Simulations are phenomenologically rich



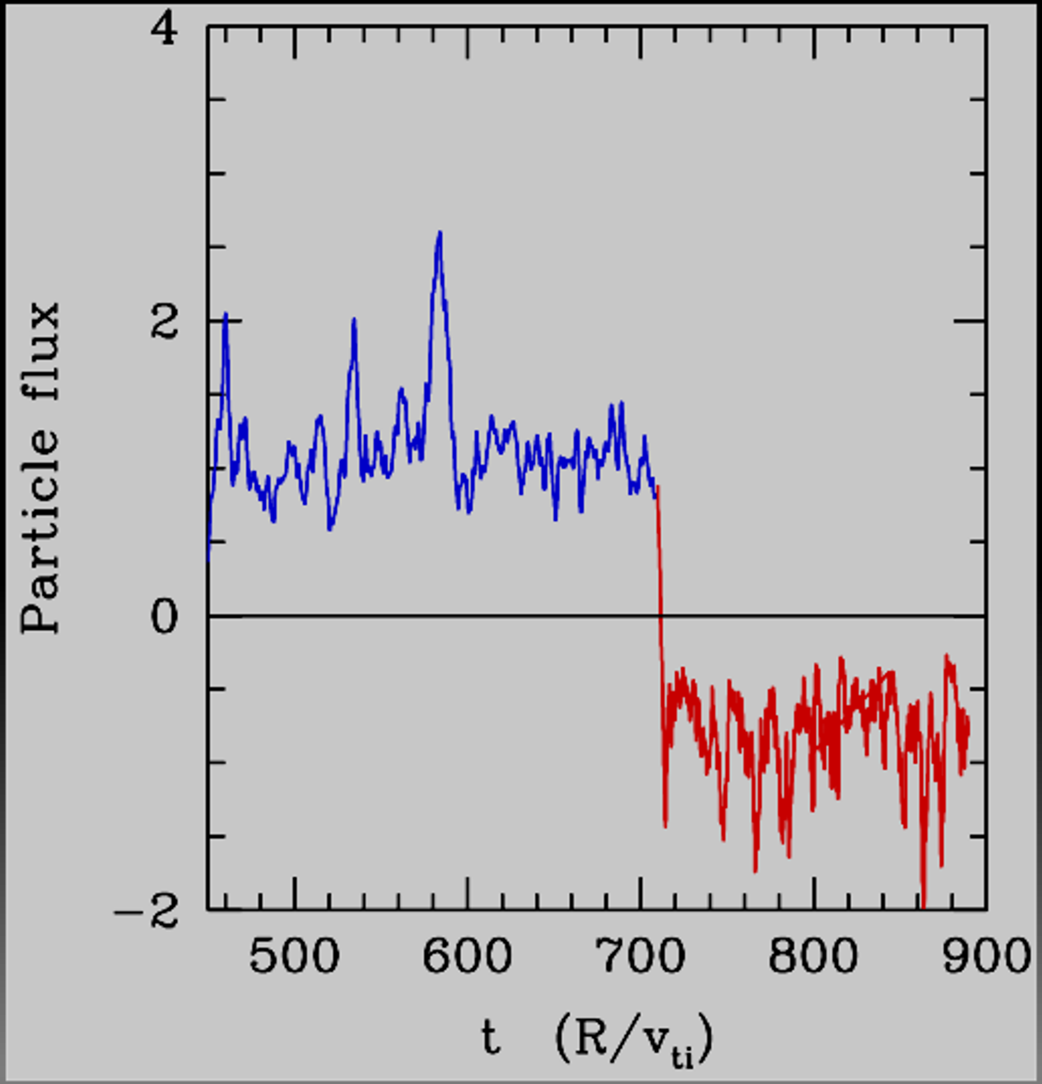
- Here are some temperature profiles from the Tore-Supra tokamak
- In this case, there are no sources of particles except at the edge

Simulations are phenomenologically rich



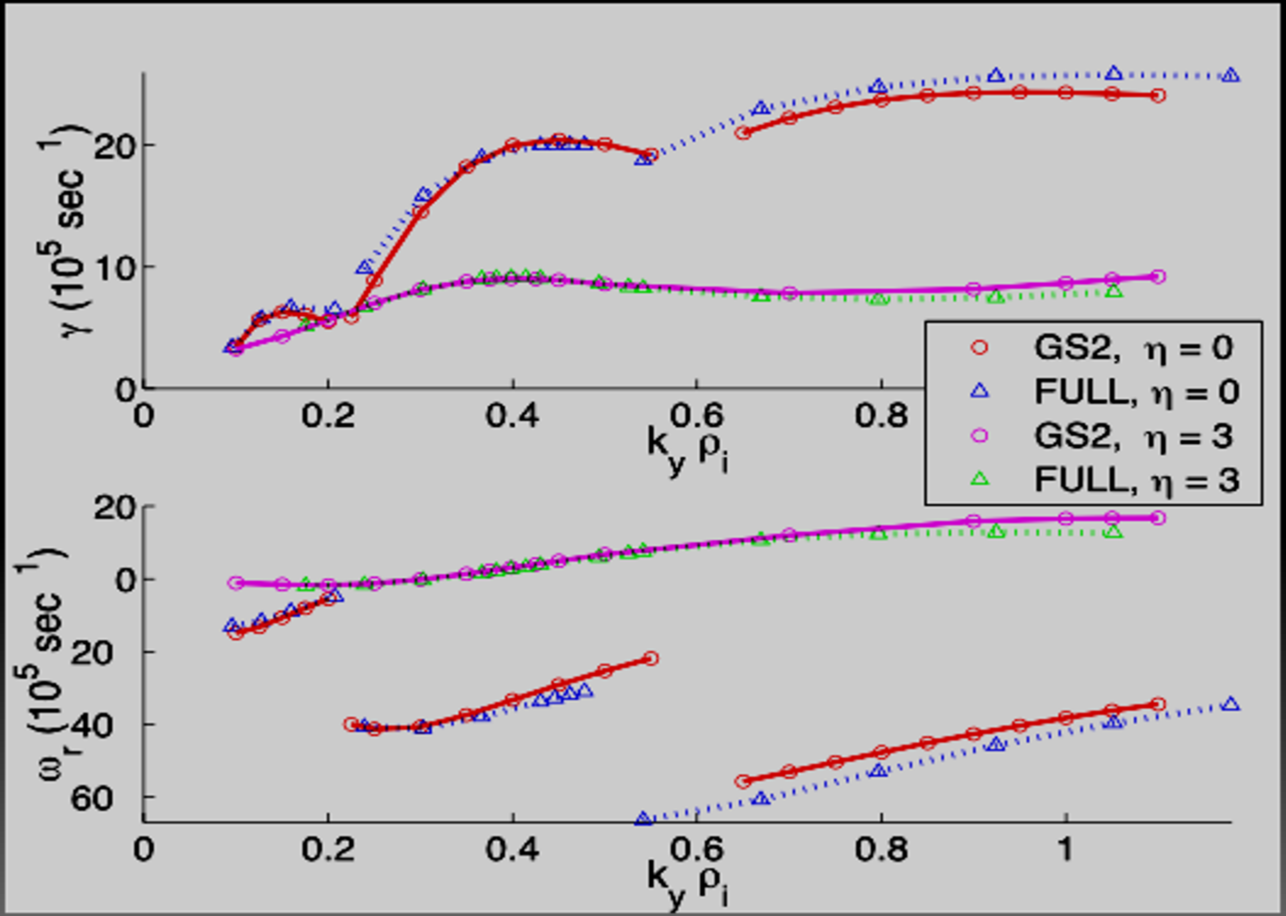
- Here are some temperature profiles from the Tore-Supra tokamak
- In this case, there are no sources of particles except at the edge
- And yet the density is highest in the center
- Turbulence must be concentrating the plasma in the center.

Simulations are phenomenologically rich



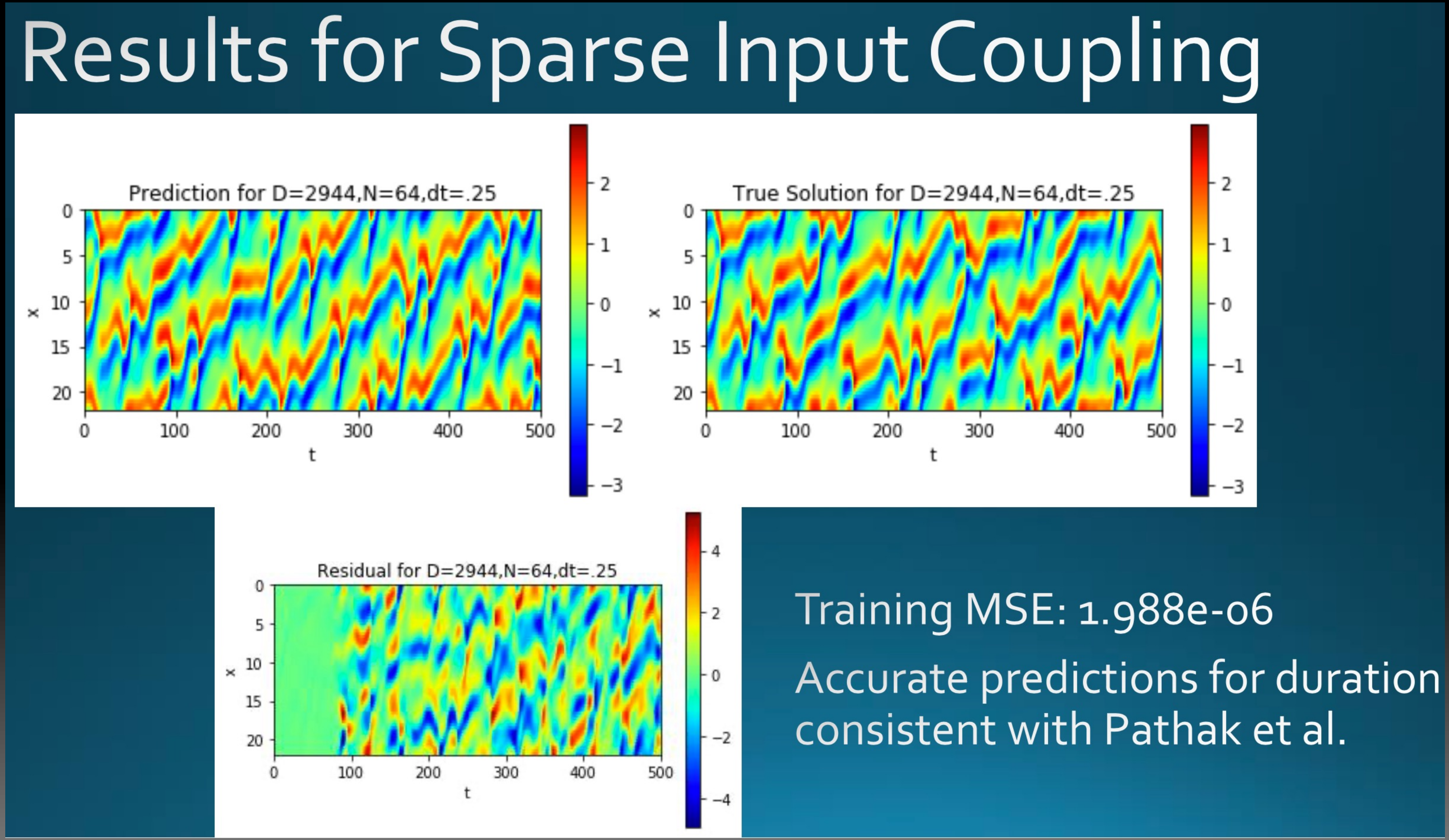
- In this case, there are no sources of particles except at the edge
- And yet the density is highest in the center
- Turbulence must be concentrating the plasma in the center.
- Simulations predict this kind of thing (change of temperature gradient from red to blue causes particle flux to reverse).
- This is a turbulence-induced plasma concentration.

Can study stellarators, tokamaks, mirrors, etc



- Here are some primary growth rates and frequencies as calculated by two different methods
- Geometry is the NCSX stellarator

Using machine learning to model turbulence



- Upper right: a flow field that is chaotic
- Upper left: a machine-learning prediction of the chaotic flow
- Bottom: the difference between the two
- J Pathak originally, N Barbour here

Summary

- I’ve given you some general thoughts about what scientific computing is and how to approach it if you wish to be successful
- I showed you some examples of scientific computing in the form of “elucidation of physics” and how that kind of work leads to the generation of new, simplified models that are faster to compute (or even closed-form)
- The goal of scientific computation is not always to generate insights and simpler models. Sometimes we need to wire up ever more complicated whole-device models.
- Machine learning can be used in many interesting ways. I flashed two examples. ML is a rapidly growing field with many good ideas in the air.