Computational Plasma Physics: Powerful New Tools of Scientific Discovery

Greg Hammett, Princeton Plasma Physics Lab (PPPL)
http://w3.pppl.gov/~hammett

DOE National Undergraduate Fellowship Program
In Plasma Physics and Fusion Sciences, 1-week course, June 2006

Acknowledgments:

Center for Multiscale Plasma Dynamics

& Plasma Microturbulence Project

(General Atomics, U. Maryland, LLNL, PPPL,
U. Colorado, UCLA, U. Texas)

DOE Scientific Discovery Through
Advanced Computing

http://fusion.gat.com/theory/pmp

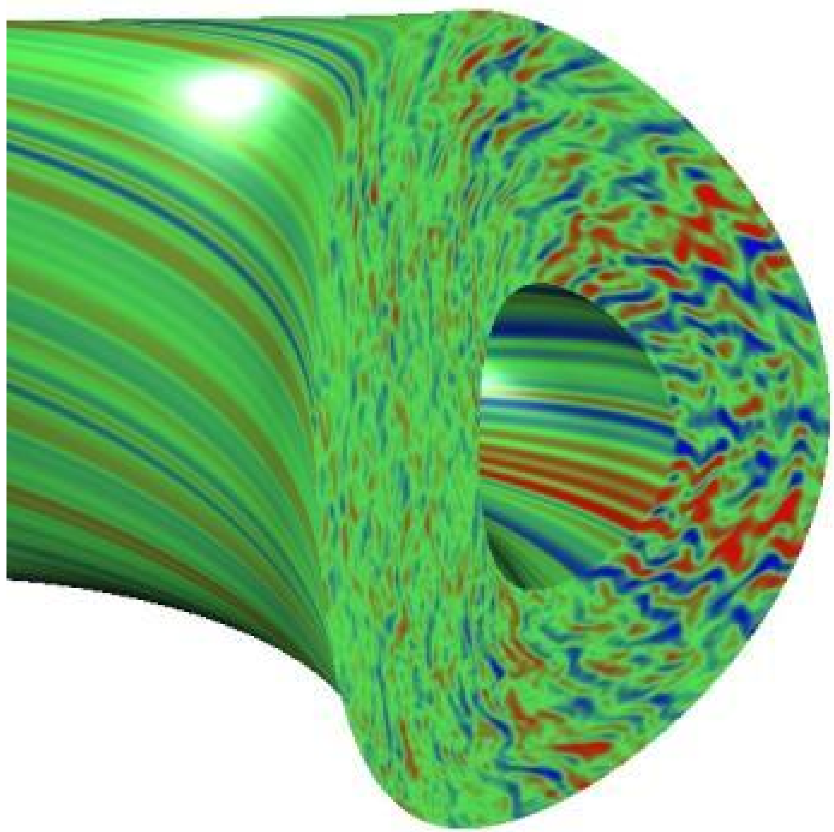
J. Candy, R. Waltz (General Atomics)

W. Dorland (Maryland) W. Nevins (LLNL)

R. Nazikian, D. Meade, E. Synakowski (PPPL)

J. Ongena (JET),

S. Jardin, D. Keyes, et al.



Candy, Waltz (General Atomics)

Computational Plasma Physics: Powerful New Tools of Scientific Discovery

- Brief intro to computational science

- The importance of good numerical algorithms

    Pitfall of naive algorithm for diffusion equation

- Examples of cutting edge computational plasma physics, such as:

    Simulating 5-dimensional plasma turbulence in fusion devices

But first another topic:

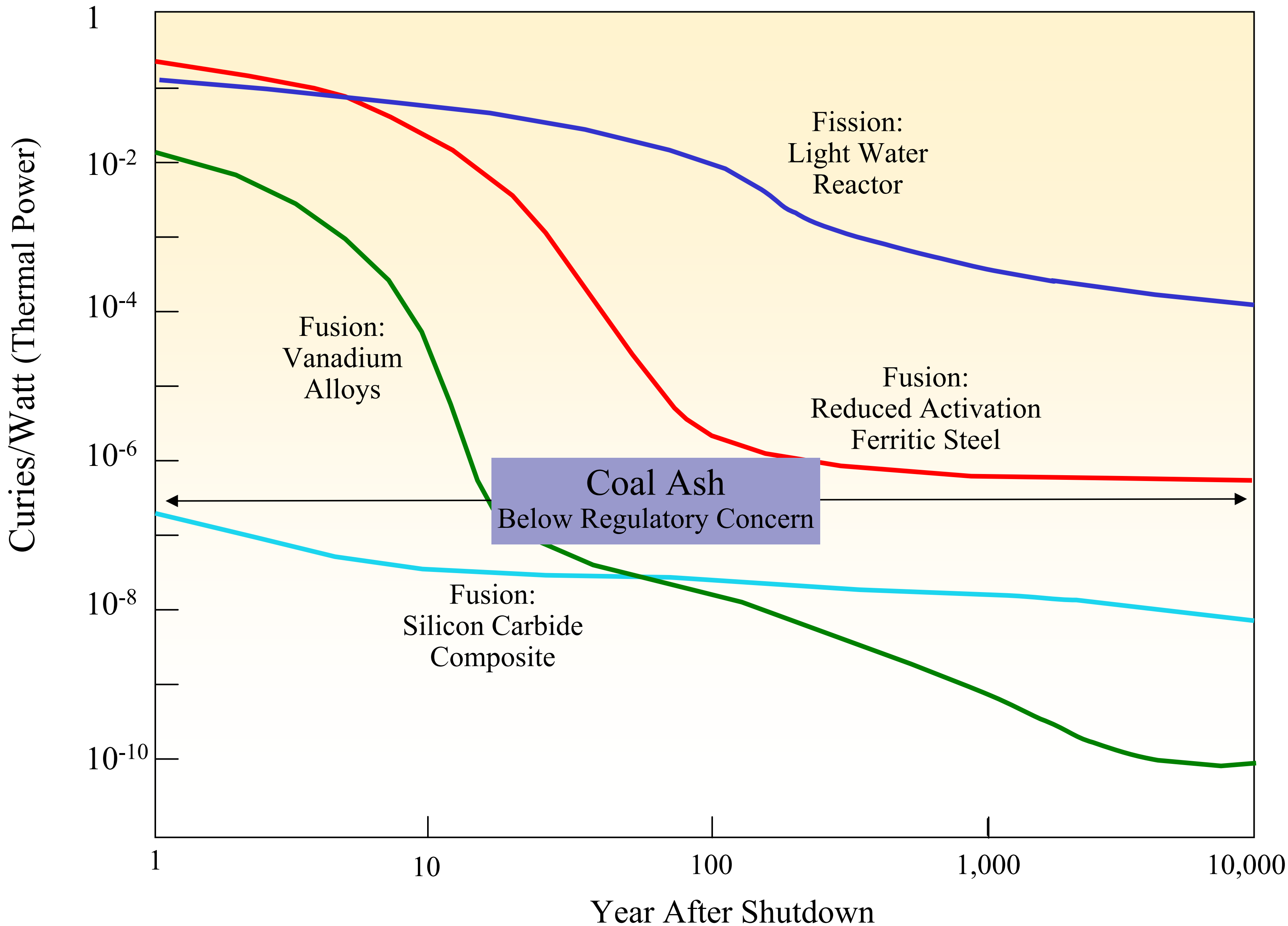
Brief Review:

Progress is being made in Fusion Energy research, well worth continuing.

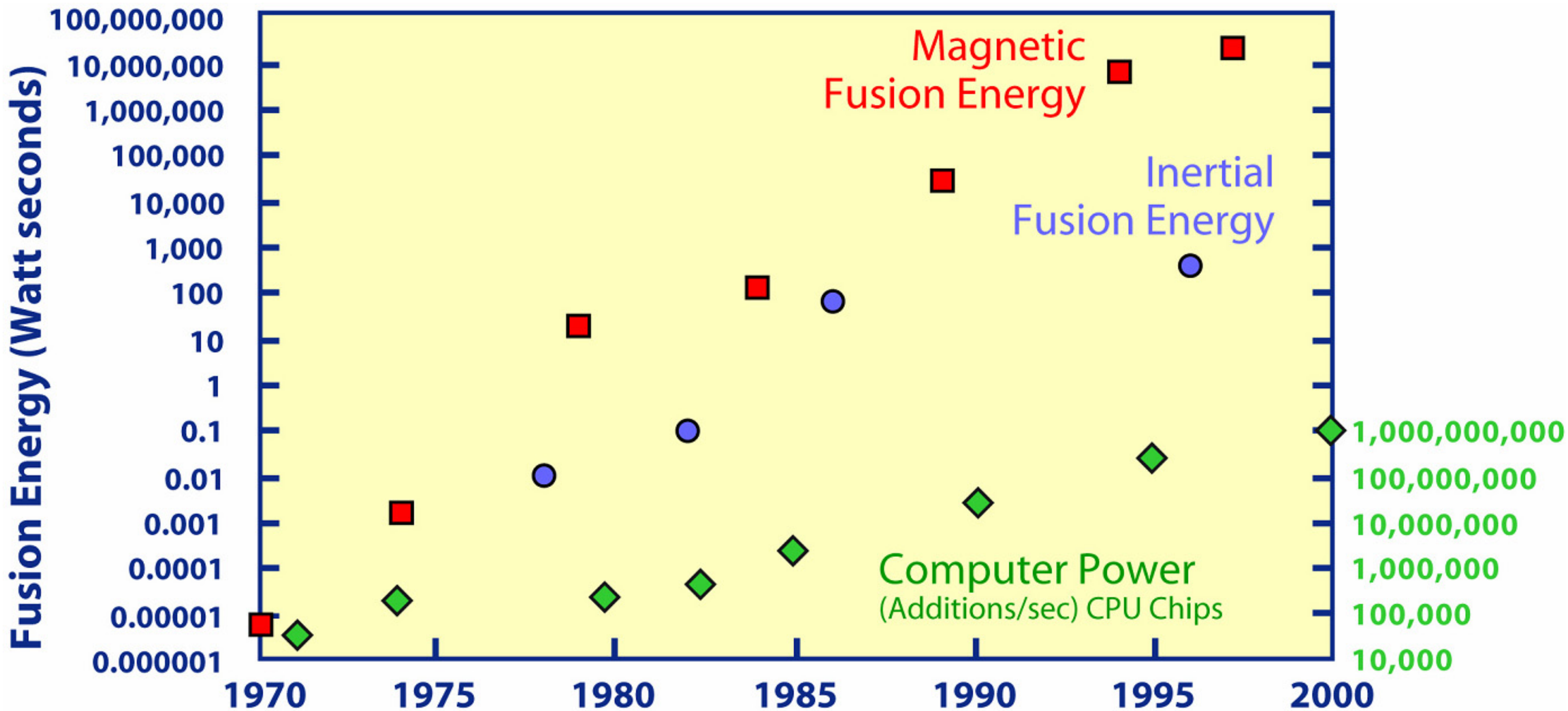
Fusion can be an Attractive Domestic Energy Source

- Abundant fuel, available to all nations
  - Deuterium and lithium easily available for thousands of years
- Environmental advantages
  - No carbon emissions, short-lived radioactivity
- Can't blow up, resistant to terrorist attack
  - Less than a minute's worth of fuel in the chamber
- Low risk of nuclear materials proliferation
  - No fissile or fertile materials required
- Compact relative to solar, wind and biomass
  - Modest land usage
- Not subject to daily, seasonal or regional weather variation, no requirement for local CO_2 sequestration.
  - Not limited in its contribution by need for large-scale energy storage or extreme-distance transmission
- Cost of power estimated similar to coal, fission
- Can produce electricity and hydrogen
  - Complements other nearer-term energy sources

Comparison of Fission and Fusion Radioactivity After Shutdown

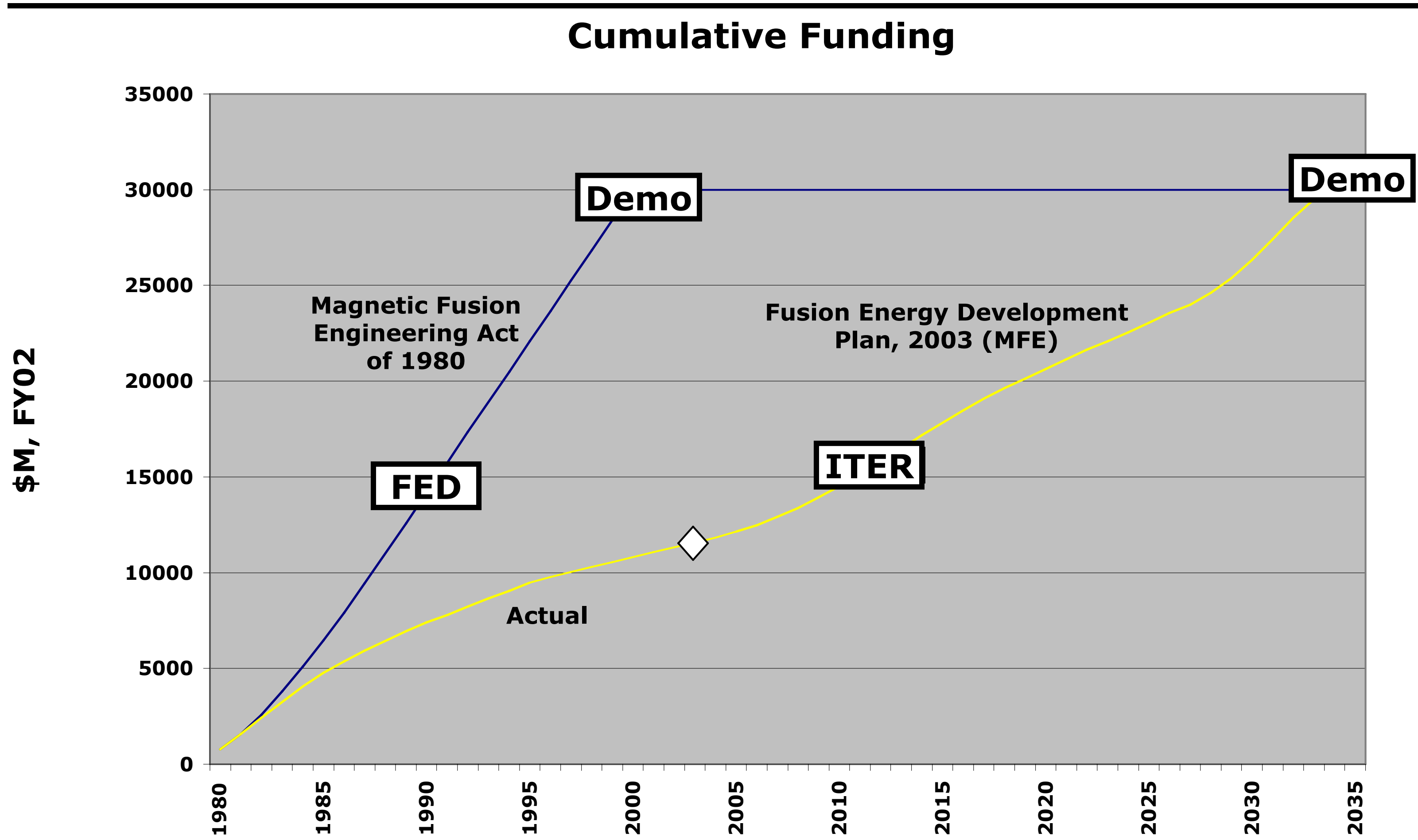


Progress in Fusion Energy has Outpaced Computer Speed



Some of the progress in computer speed can be attributed to plasma science.

The Estimated Development Cost for Fusion Energy is Essentially Unchanged since 1980



On budget,
if not on time.

$30B development cost tiny compared to >$100 Trillion energy needs of 21st century and potential costs of global warming. Still 40:1 payoff after discounting 50+ years.

Introduction to Scientific Computing

and the Importance of

Good Numerical Algorithms

Computing has become a powerful 3rd way of Scientific Discovery

- Traditional view: Experiments or Theory
  Scientific method is all about the interaction between the two:

Experiments
Reality
observation
hypothesis testing

Theory
hypothesis formulation
Insights, systematic laws

- New view: Experiments and Theory and Computing:

Experiments
Reality

Theory
Exact solutions to approximate Eqs.
(or exact Eqs. for simplified systems)

Computational Science
Approximate solutions to exact Eqs.
(or more complicated systems)
numerical experiments / computational theory

(oversimplified)

Computational Physics & Numerical Algorithms are Interesting and Rich Fields

- Amazing exponential growth in computer power means we can now solve many problems that were thought impossible 30 years ago. Computers being applied to many problems of human importance and interest (physics, astrophysics, biology, climate modelling, engineering…)
  (Careful: there are also many problems that can never be directly solved on computers…)

- Computational work very interesting: sometimes you don't really understand equations until you get up close & personal with them to solve them numerically. Boundary conditions, conservation laws, other properties that should be preserved…

- Study of Numerical algorithms is a large and rich field: Huge bag of numerical tricks: Many different algorithms highly optimized for different applications. Choice of which features of the original equations you want to preserve most accurately in the discrete numerical approximations:
  - Highly accurate solutions in some regimes but large errors in other regimes vs.
  - Fast and Robust algorithms that have somewhat larger but manageable errors over a wider range of parameters
  - Preserve exact conservation laws or other important properties of real solution? Conserve momentum or energy to round-off error (but sometimes can't conserve both). Preserve $\nabla \cdot \mathbf{B}=0$ exactly?

Pitfalls of Naive Numerical Algorithms: A Simple Diffusion Eq. Example

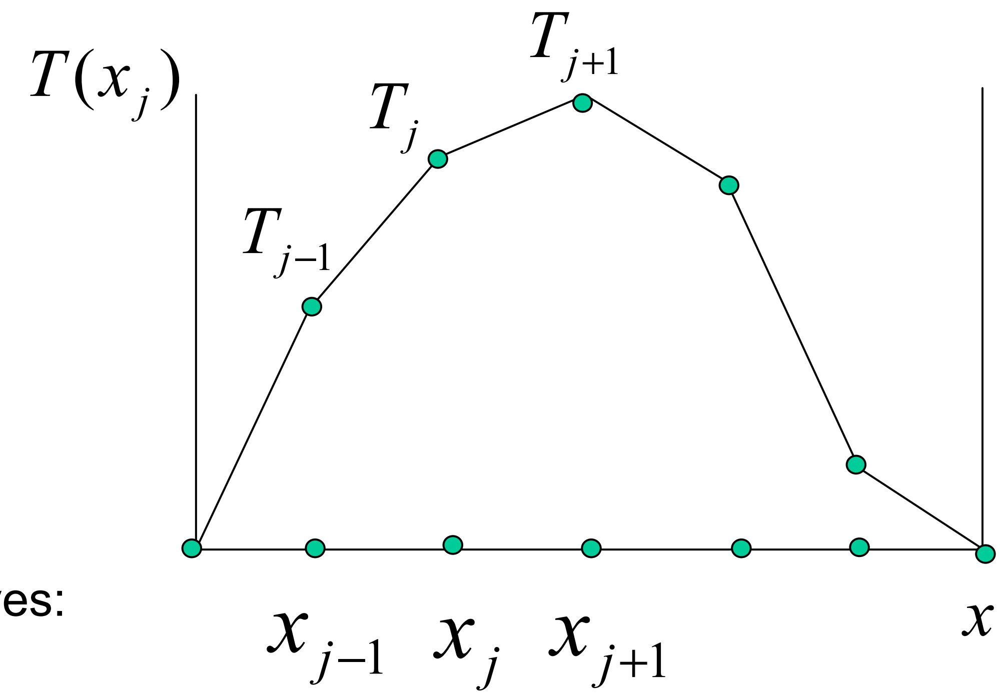
$$\frac{\partial}{\partial t}T(x,t) = \frac{\partial^2 T}{\partial x^2} + \sin(x)$$

Discretizing a Diffusion Eq.

$$\frac{\partial}{\partial t}T(x,t)=\frac{\partial^2 T}{\partial x^2}+\sin(x)$$

Discretize T(x) onto a grid:

$$T(x_j,t)=T_j(t)$$



Discrete analogs of 1st & 2cd derivatives:

$$\left(\frac{\partial T}{\partial x}\right)_{j+1/2}=\frac{T_{j+1}-T_j}{\Delta x}$$

$$x_j = j\,\Delta x$$

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_j=\left(\frac{\partial}{\partial x}\frac{\partial T}{\partial x}\right)=\frac{(T_{j+1}-T_j)-(T_j-T_{j-1})}{(\Delta x)^2}$$

Simple Discretization of a Diffusion Eq.

$$\frac{\partial}{\partial t} T(x,t) = \frac{\partial^2 T}{\partial x^2} + \sin(x)$$

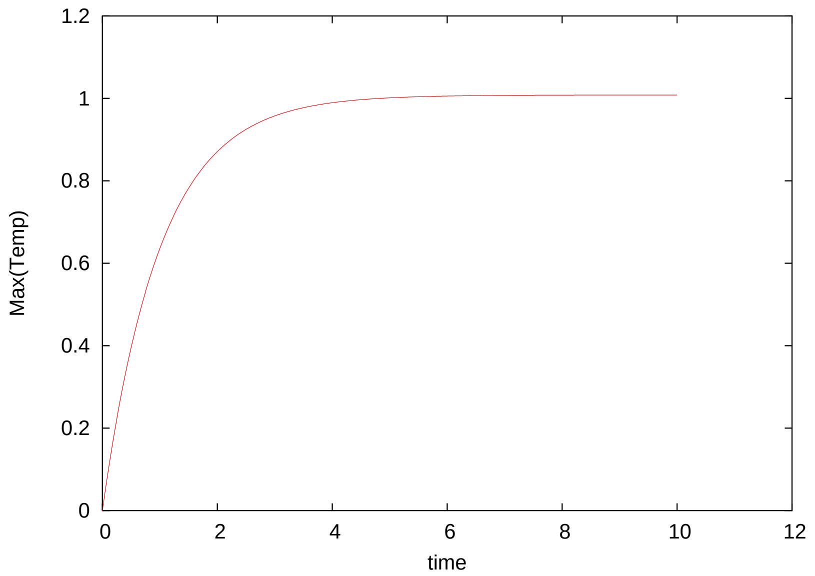
$$\frac{T_j(t+\Delta t) - T_j(t)}{\Delta t} = \frac{(T_{j+1} - T_j) - (T_j - T_{j-1})}{(\Delta x)^2} + \sin(x_j)$$

Given $T_j(t)$, loop over all positions j to get T_j (t+dt).
Repeat to find T(t+2dt) from T(t+dt)…

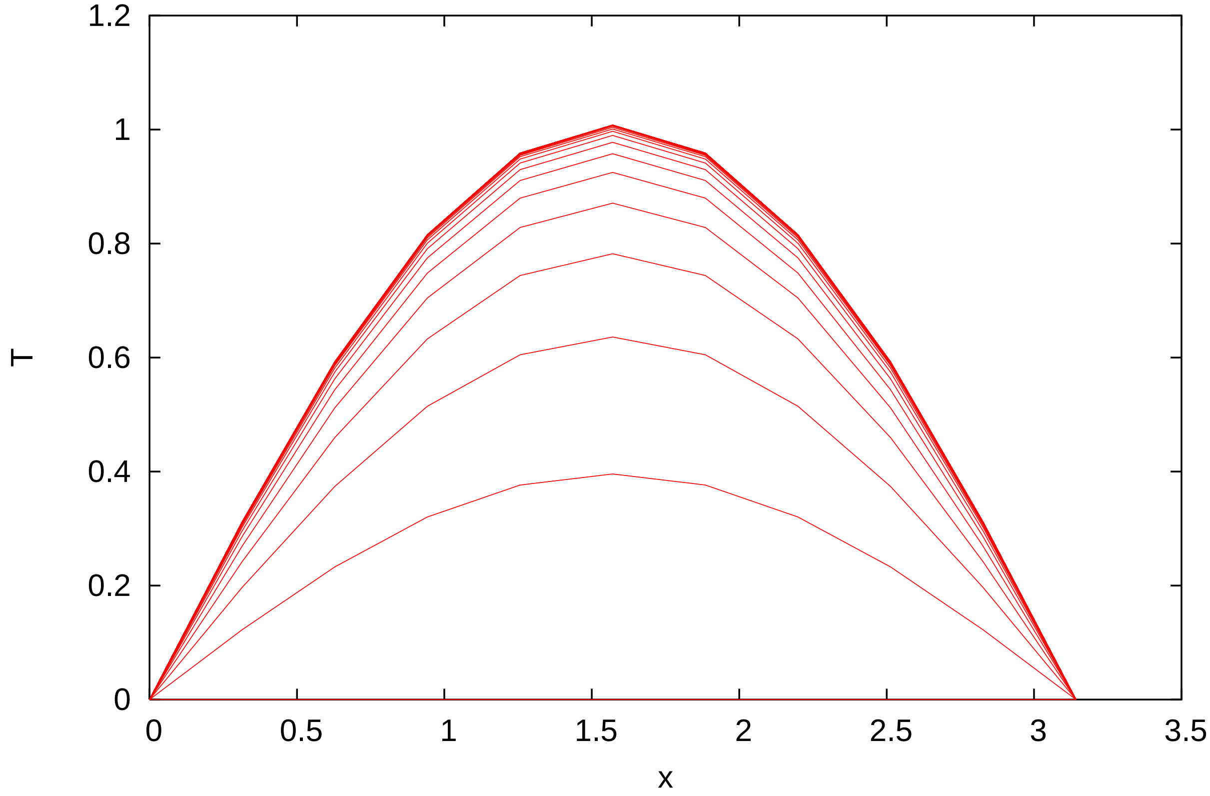
Test simple diffusion algorithm on a coarse mesh,
10 points for x=0 to x=pi,
dt=0.01

matches exact solution fairly well.

Everything “seems fine”, just want to use finer spatial mesh…



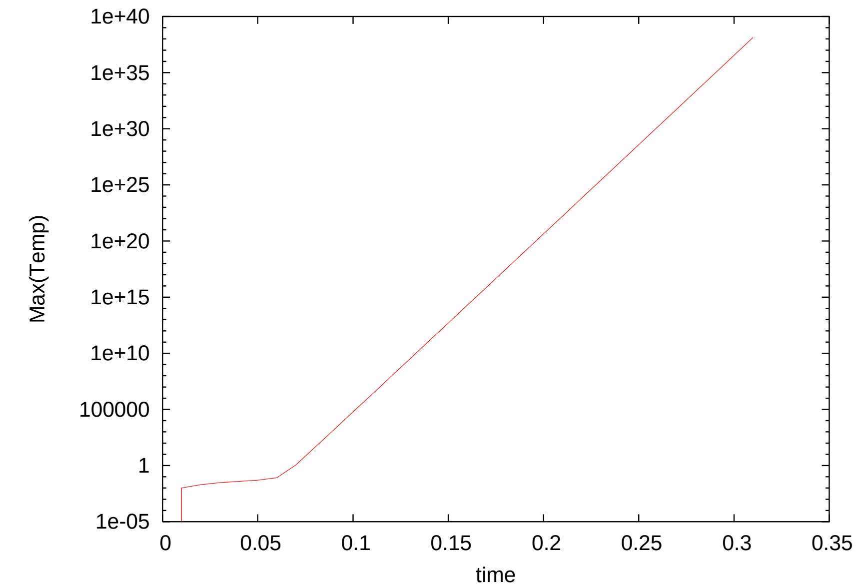
T(x) at t=0, 0.5, 1.0, ...

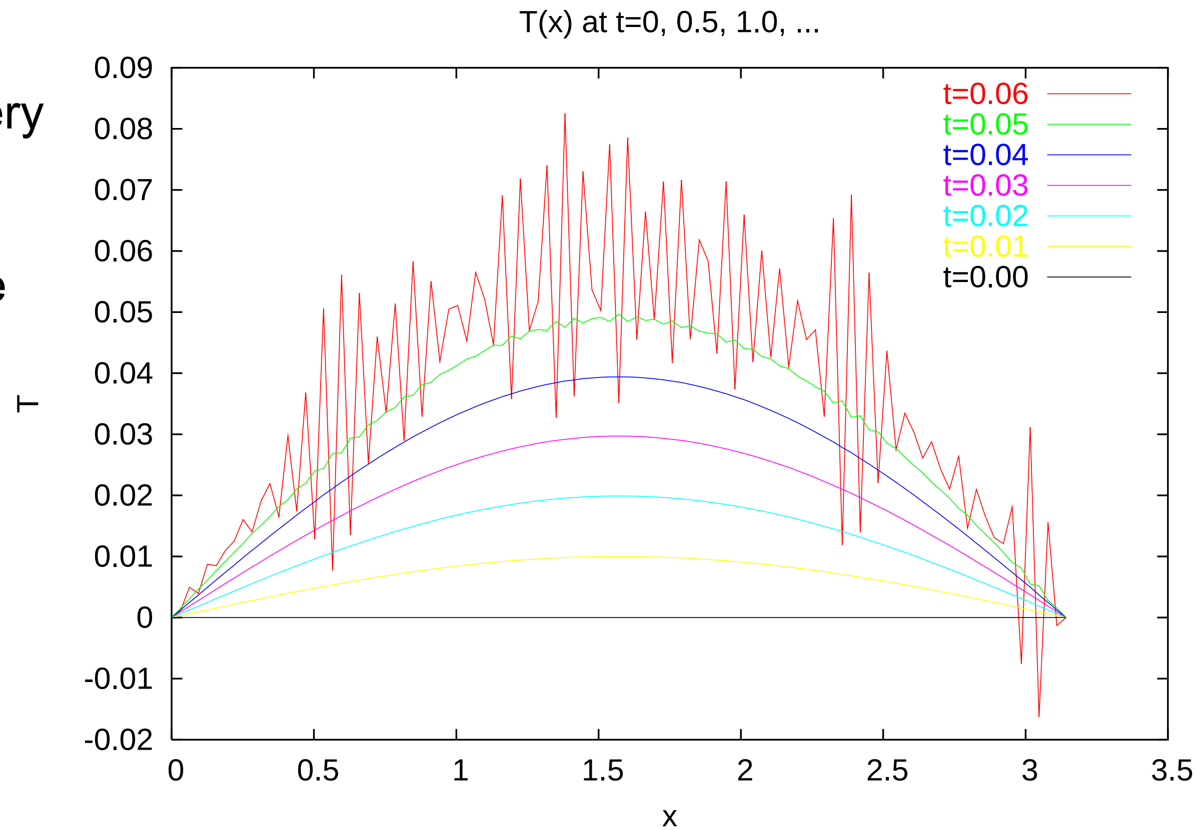


Test simple diffusion algorithm on a finer mesh,
increase from 10 to 100 points for x=0 to x=pi,
dt=0.01 (unchanged)

Within 6 time steps the solution becomes garbage

Maximum temperature grows very quickly due to numerical instability, exceeds biggest number representable on the computer in just a few dozen iterations, T = “NaN”.





The root of the problem

$$\frac{\partial}{\partial t}T(x,t) = D\frac{\partial^2 T}{\partial x^2} + \sin(x)$$

Fourier Transform, look at k ≠ 1 modes

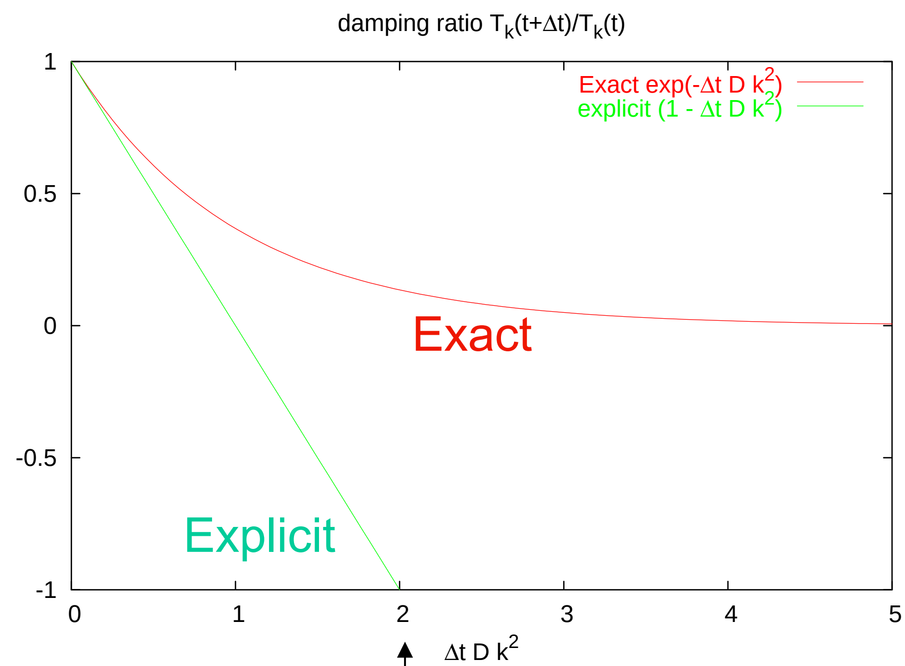
$$\frac{\partial}{\partial t}T_k = -Dk^2 T_k$$

Explicit integration (1st order “Euler”):

$$\frac{T_k(t+\Delta t) - T_k(t)}{\Delta t} = -Dk^2 T_k(t)$$

$$T_k(t+\Delta t) = \underbrace{\left(1 - \Delta t\; Dk^2\right)T_k(t)}$$

1st order Taylor
series approx. to
exact result $e^{-\Delta t\, D k^2}$



Stability limit: $\Delta t\ D\ k^2 < 2$
for all k modes in simulation

Fix with a more robust Implicit algorithm

$$\frac{\partial}{\partial t} T_k = -Dk^2 T_k$$

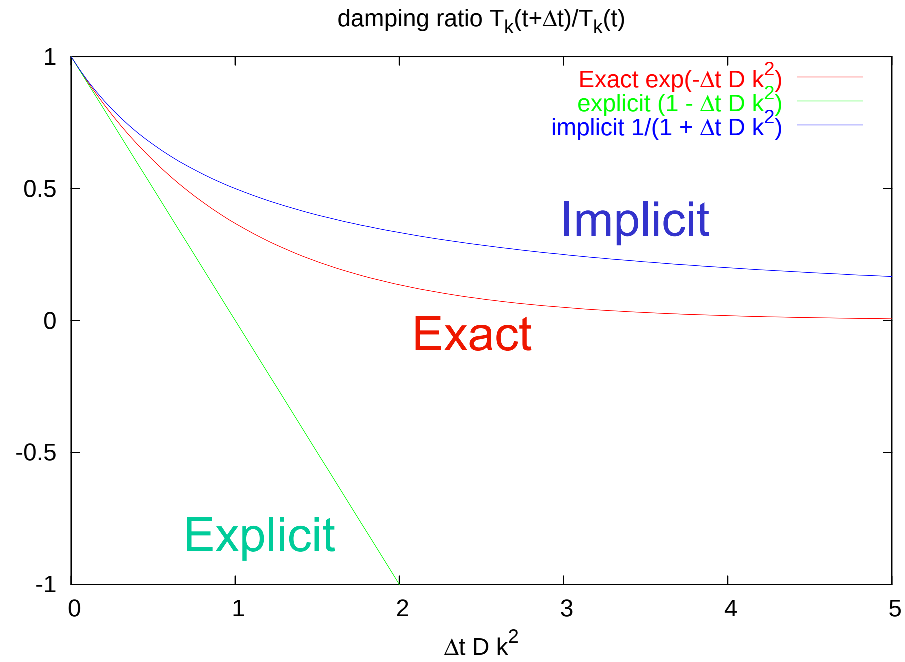
Explicit integration (1st order "Euler"):

$$\frac{T_k(t+\Delta t) - T_k(t)}{\Delta t} = -Dk^2 T_k(t)$$

Implicit integration (1st order "Backwards Euler"):

$$\frac{T_k(t+\Delta t) - T_k(t)}{\Delta t} = -Dk^2 T_k(t+\Delta t)$$

$$T_k(t+\Delta t) = \underbrace{\frac{1}{\left(1+\Delta t\ Dk^2\right)}}_{\text{1st order Padé approx. to exact result } e^{-\Delta t\, D k^2}} T_k(t)$$



Implicit algorithm is accurate for modes with small $\Delta t\ D\ k^2$, while robustly stable for all $\Delta t\ D\ k^2$, giving qualitatively correct damping for all modes.

(higher order implicit algorithms exist: Crank-Nicolson, Backward Differentiation Formulas)

Geometrical Interpretation of Implicit Algorithm as Integrating Backwards in Time

$$\frac{\partial y}{\partial t} = F(y) = -y$$

Explicit integration (1st order "Euler"):

$$y(\Delta t) = y(0) + \Delta t\, F(y(0))$$

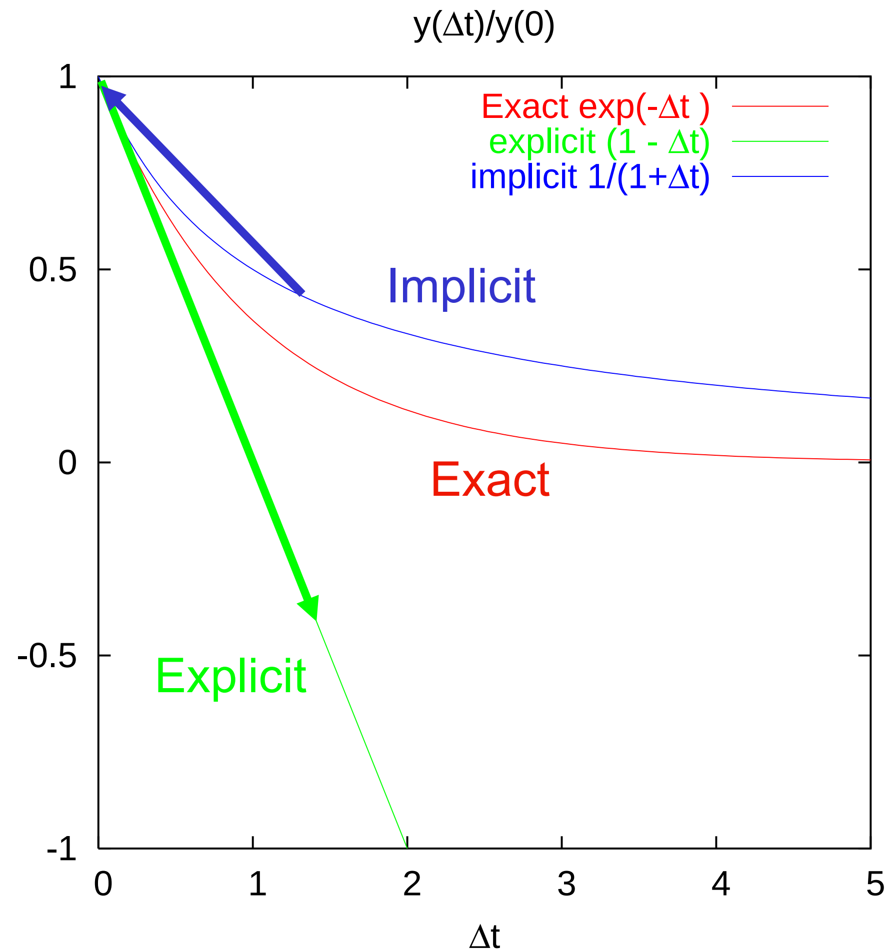
Implicit integration (1st order "Backwards Euler"):

$$y(\Delta t) = y(0) + \Delta t\, F(y(\Delta t))$$

Rearrange as:

$$y(0) = y(\Delta t) - \Delta t\, F(y(\Delta t))$$

Thus implicit algorithm
= integrating backwards in time,
to find what y(Δt) at the future time is
neded to give y(0) at the current time.
(Requires inverting operator 1-ΔtF …)



Implicit algorithms more complex, require inversions

$$\frac{\partial}{\partial t}T(x,t) = \frac{\partial^2 T}{\partial x^2} + \sin(x)$$

$$\frac{T_j(t+\Delta t) - T_j(t)}{\Delta t} = \frac{T_{j+1}(t+\Delta t) - 2T_j(t+\Delta t) + T_{j-1}(t+\Delta t)}{(\Delta x)^2} + \sin(x_j)$$

Can rearrange this in the form:

$$M_{ij}T_j(t+\Delta t) = T_i(t) + \Delta t \sin(x_i)$$

Requires inverting the matrix M to find the vector T at the future time

Sometimes this can be hard: General NxN matrix requires $O(N^3)$ operations to invert. In this case, M is "tridiagonal" and solution can be found quickly in O(N) operations. For many PDE's **M** is "sparse", and fast solution methods exist… In general:

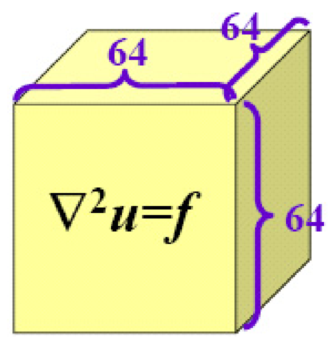
$$\frac{\partial \mathbf{Y}}{\partial t} = \mathbf{F}(\mathbf{Y})$$

Things can get very interesting, if **F** is a nonlinear integro-differential operator that has to be inverted. Recent Newton-Krylov algorithms?

The power of optimal algorithms

- Advances in algorithmic efficiency can rival advances in hardware architecture
- Consider Poisson's equation on a cube of size $N=n^3$

Year	*Method*	*Reference*	*Storage*	*Flops*
1947	GE (banded)	Von Neumann & Goldstine	n^5	n^7
1950	Optimal SOR	Young	n^3	$n^4 \log n$
1971	CG	Reid Conjugate Gradients w/ modified ILU preconditioner	n^3	$n^{3.5} \log n$
1984	Full MG	Brandt Multigrid	n^3	n^3



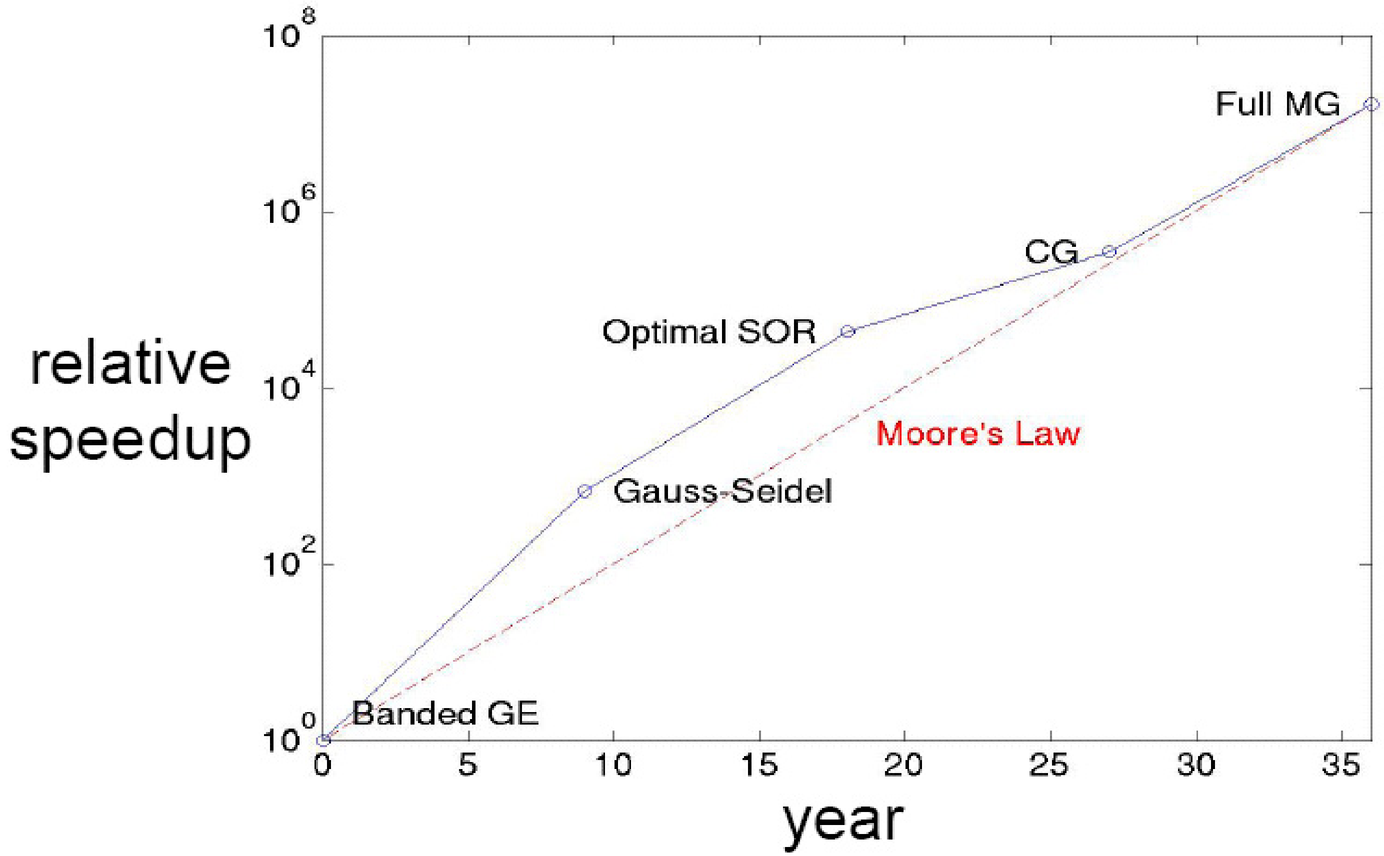
- If n=64, this implies an overall reduction in flops of ~16 million *

*Six-months is reduced to 1 s

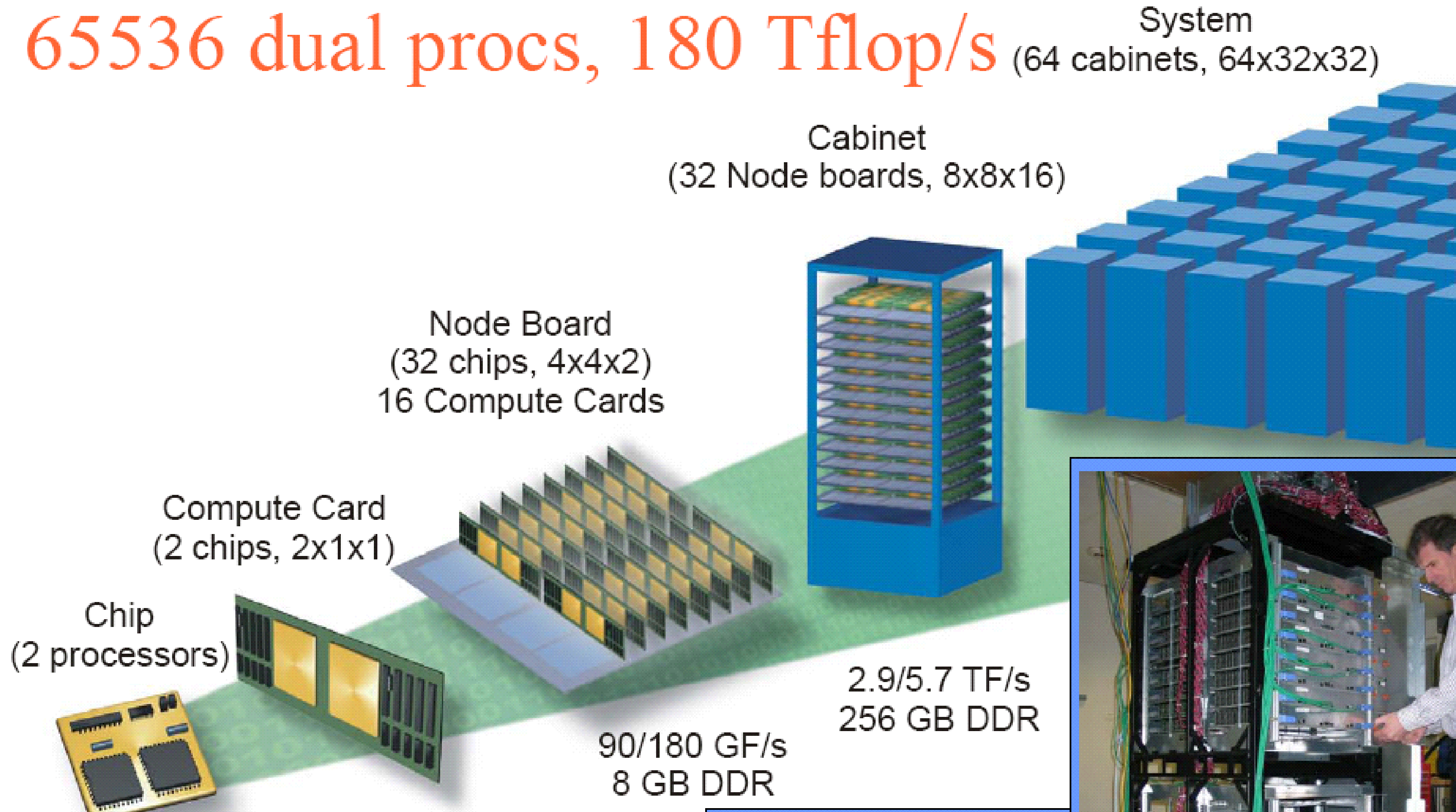
David Keyes, Columbia Univ.

Algorithms and Moore's Law

- This advance took place over a span of about 36 years, or 24 doubling times for Moore's Law
- $2^{24} \approx 16$ million $\Rightarrow$ the same as the factor from algorithms alone!



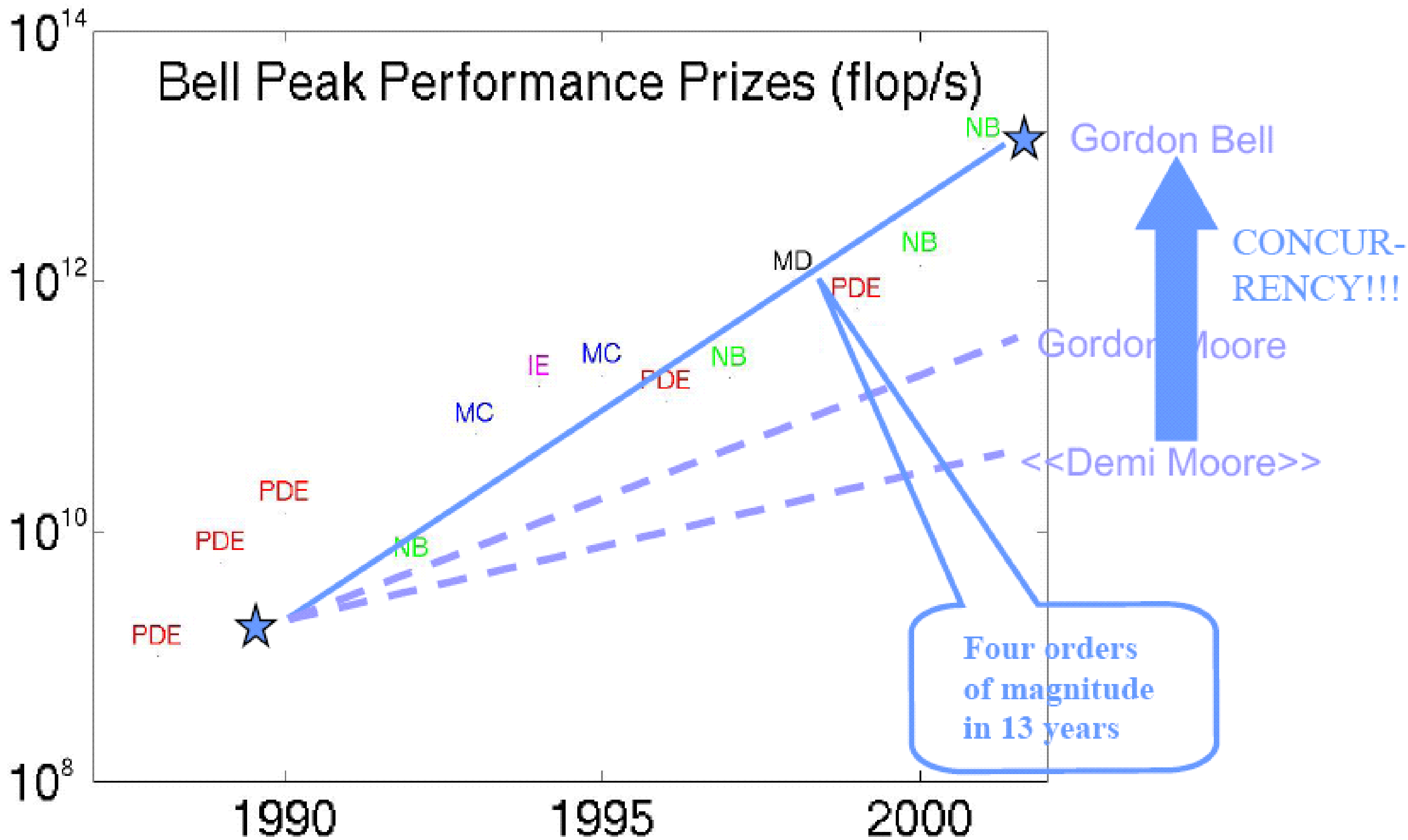
IBM's BlueGene/L:
65536 dual procs, 180 Tflop/s



Present offer from IBM

Single cabinet
5.7 TFlop/s peak
$2M in acad. consortium

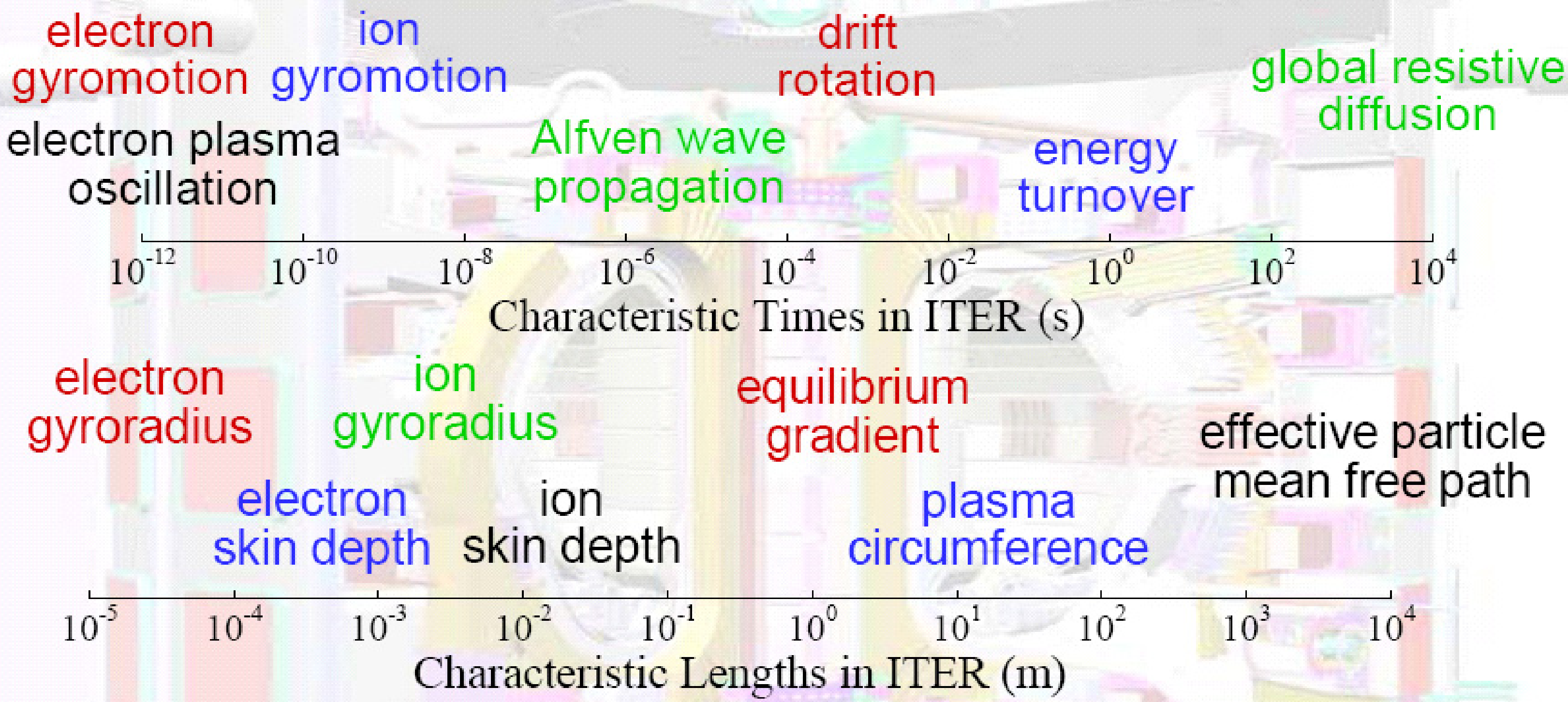
Gordon Bell Prize outpaces Moore's Law



Whimsical remarks on simulation progress, 1988-2005

- If similar improvements in speed (10^5) had been realized in the airline industry, a 3-hour flight would require one-tenth of a second today
- If similar improvements in storage (10^4) had been realized in the publishing industry, our office bookcases could hold the book portion of the collection of the Library of Congress (15M volumes)
- If similar reductions in cost (10^4) had been realized the higher education, tuition room and board would cost about $2 per year

Fusion plasmas exhibit enormous ranges of temporal and spatial scales.



- **Nonlinear MHD-like behavior couples many of the time- & length-scales.**
- **Even within the context of resistive MHD modeling, there is stiffness and anisotropy in the system of equations.**

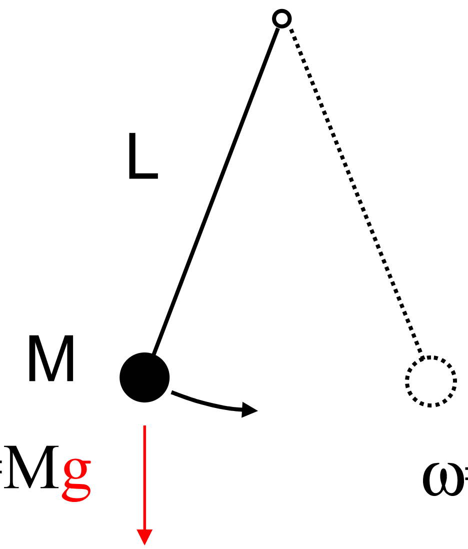
Even with the most powerful computers expected in the next 20 years, there are many problems with such an extreme range of scales that they can't be directly solved…

Carl Sovinec, http://www.csm.ornl.gov/workshops/SciDAC2005/SovinecTalk/scidac05_talk.pdf

The Plasma Microturbulence Project

- A DOE, Office of Fusion Energy Sciences, SciDAC (Scientific Discovery Through Advanced Computing) Project
- devoted to studying plasma microturbulence through direct numerical sumulation
- National Team (& four codes):
  - GA (Waltz, Candy)
  - U. MD (Dorland)
  - U. CO (Parker, Chen)
  - UCLA (Lebeouf, Decyk)
  - LLNL (Nevins P.I., Cohen, Dimits)
  - PPPL (Lee, Lewandowski, Ethier, Rewoldt, Hammett, …)
  - UCI (Lin)
- They've done all the hard work …

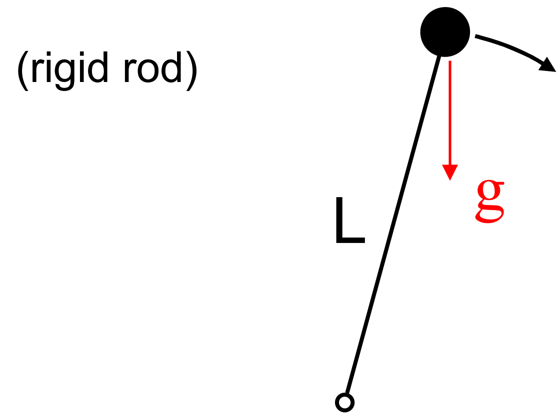
Stable Pendulum



$F = M\mathbf{g}$

$\omega = (g/L)^{1/2}$

Unstable Inverted Pendulum

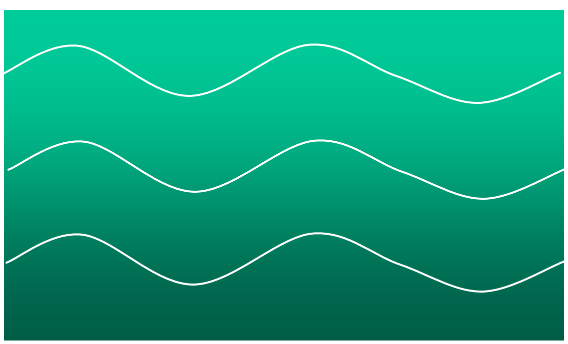


$\omega = (-g/|L|)^{1/2} = i(g/|L|)^{1/2} = i\gamma$

Instability

Density-stratified Fluid

$\rho = \exp(-y/L)$

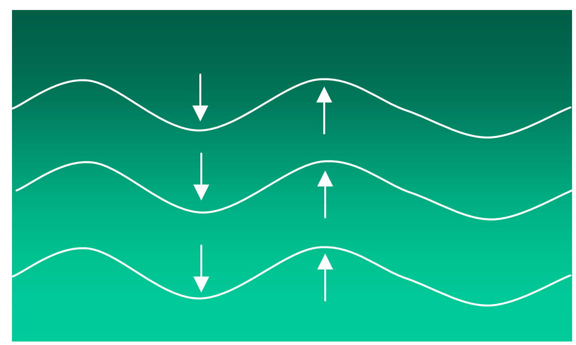


stable $\omega = (g/L)^{1/2}$

Inverted-density fluid

$\Rightarrow$Rayleigh-Taylor Instability

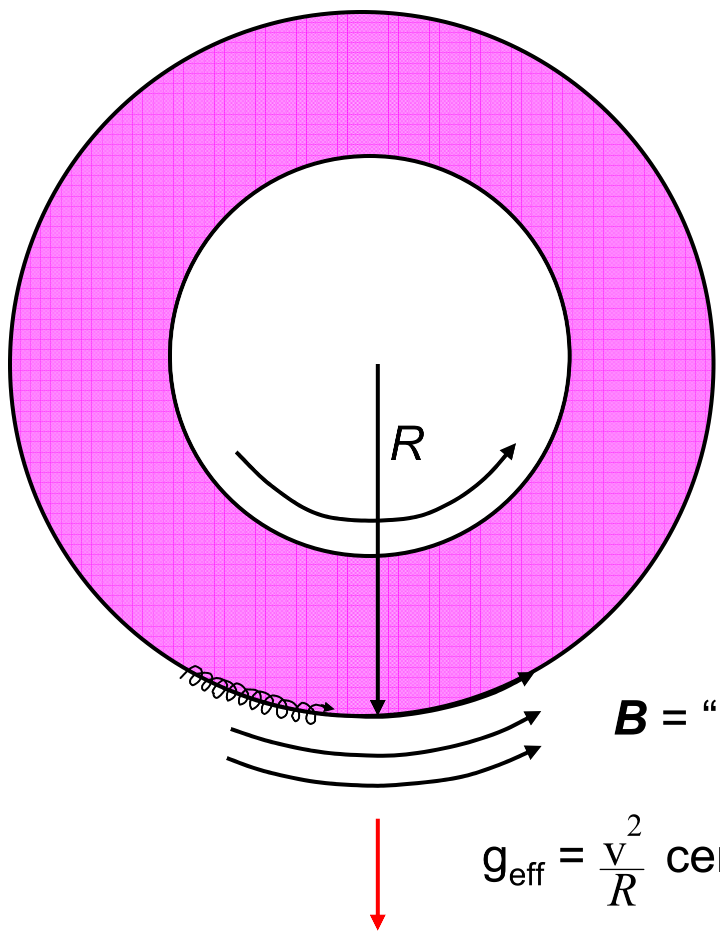
$\rho = \exp(y/L)$



Max growth rate $\gamma = (g/L)^{1/2}$

"Bad Curvature" instability in plasmas
≈ Inverted Pendulum / Rayleigh-Taylor Instability

Top view of toroidal plasma:



plasma = heavy fluid

B = "light fluid"

$g_{eff} = \frac{v^2}{R}$ centrifugal force

Growth rate:

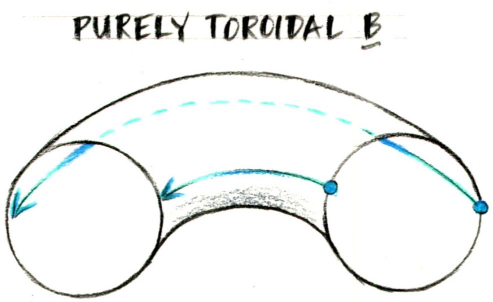
$$\gamma = \sqrt{\frac{g_{eff}}{L}} = \sqrt{\frac{v_t^2}{RL}} = \frac{v_t}{\sqrt{RL}}$$

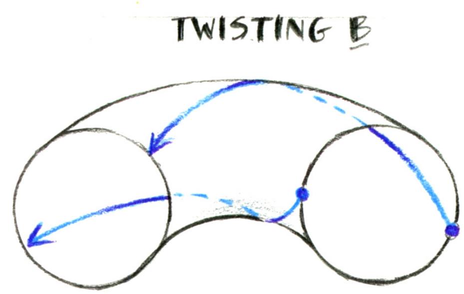
Similar instability mechanism
in MHD & drift/microinstabilities

1/L = ∇p/p in MHD,
∝ combination of ∇n & ∇T
in microinstabilities.

The Secret for Stabilizing Bad-Curvature Instabilities

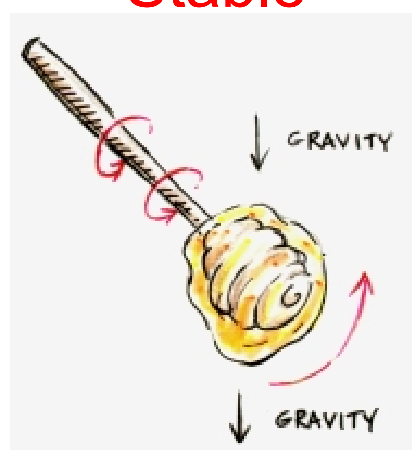
Twist in ***B*** carries plasma from bad curvature region to good curvature region:





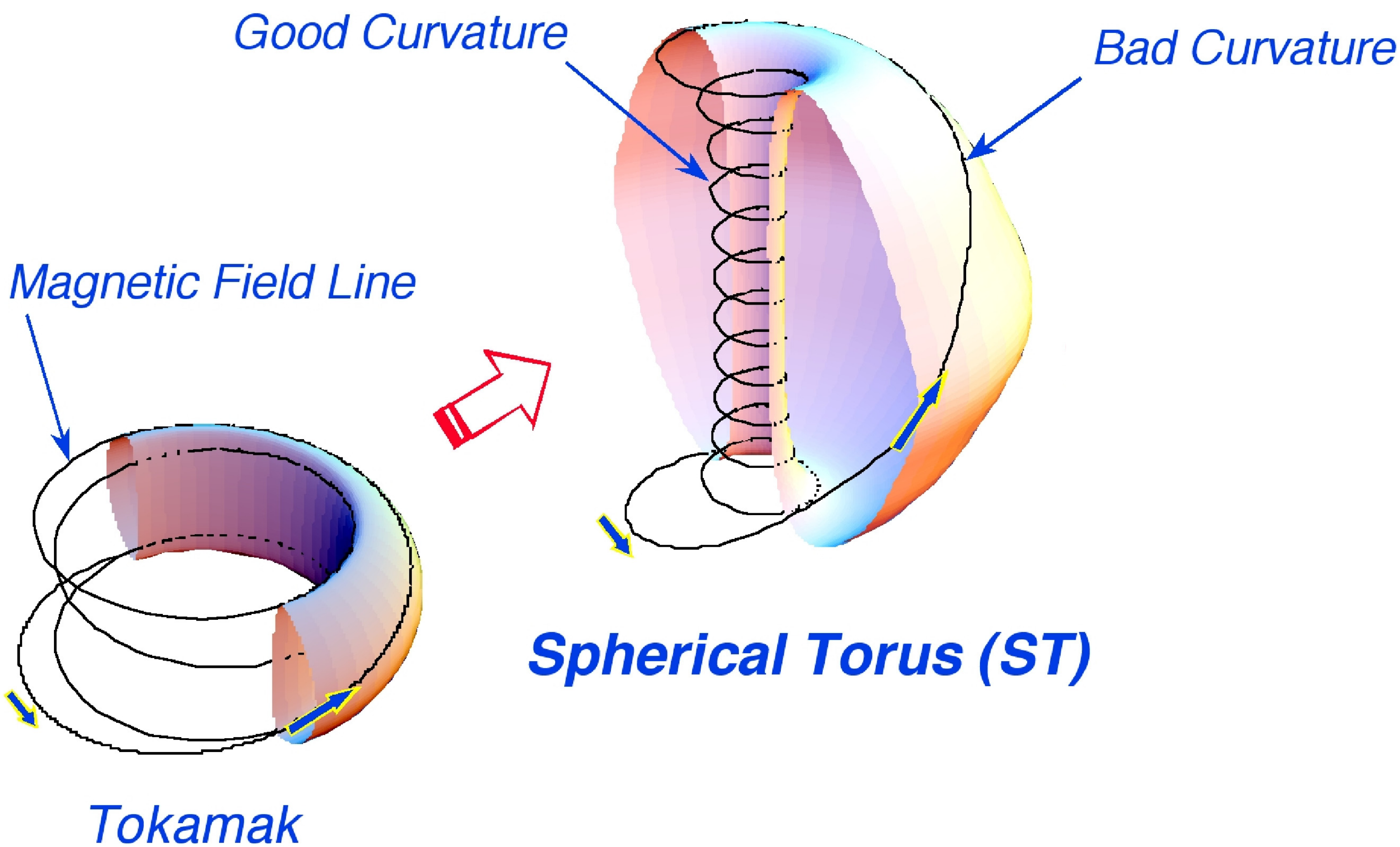
Unstable

Stable

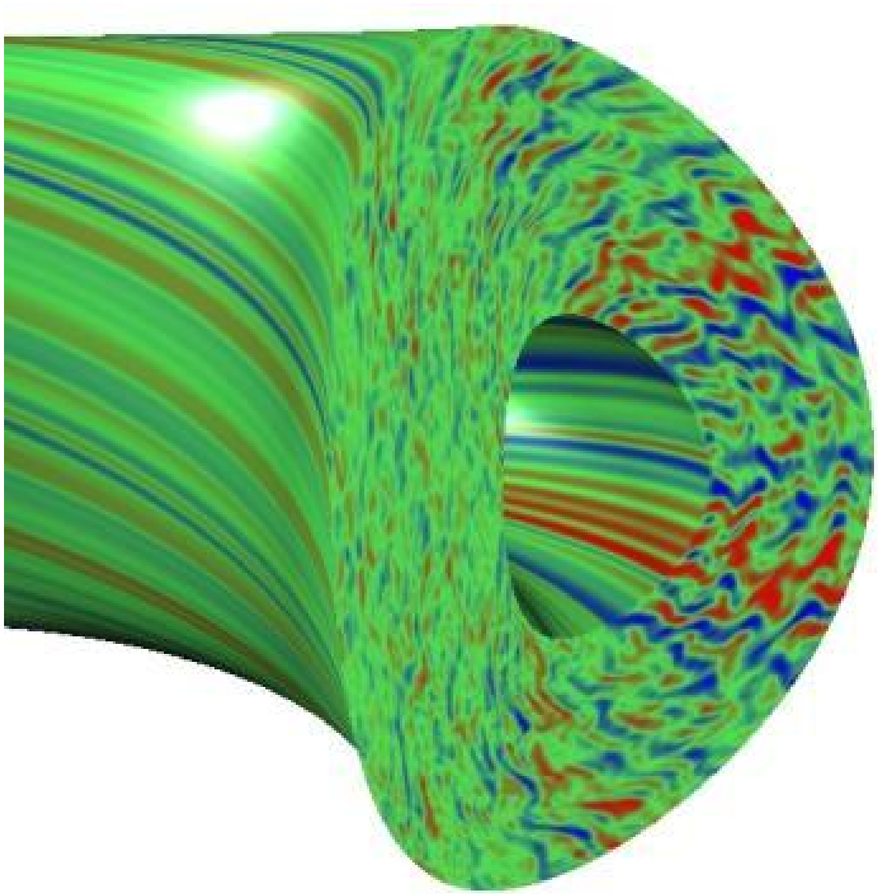


Similar to how twirling a honey dipper can prevent honey from dripping.

Spherical Torus has improved confinement and pressure limits (but less room in center for coils)



Understanding Turbulence That Affects the Performance of Fusion Device

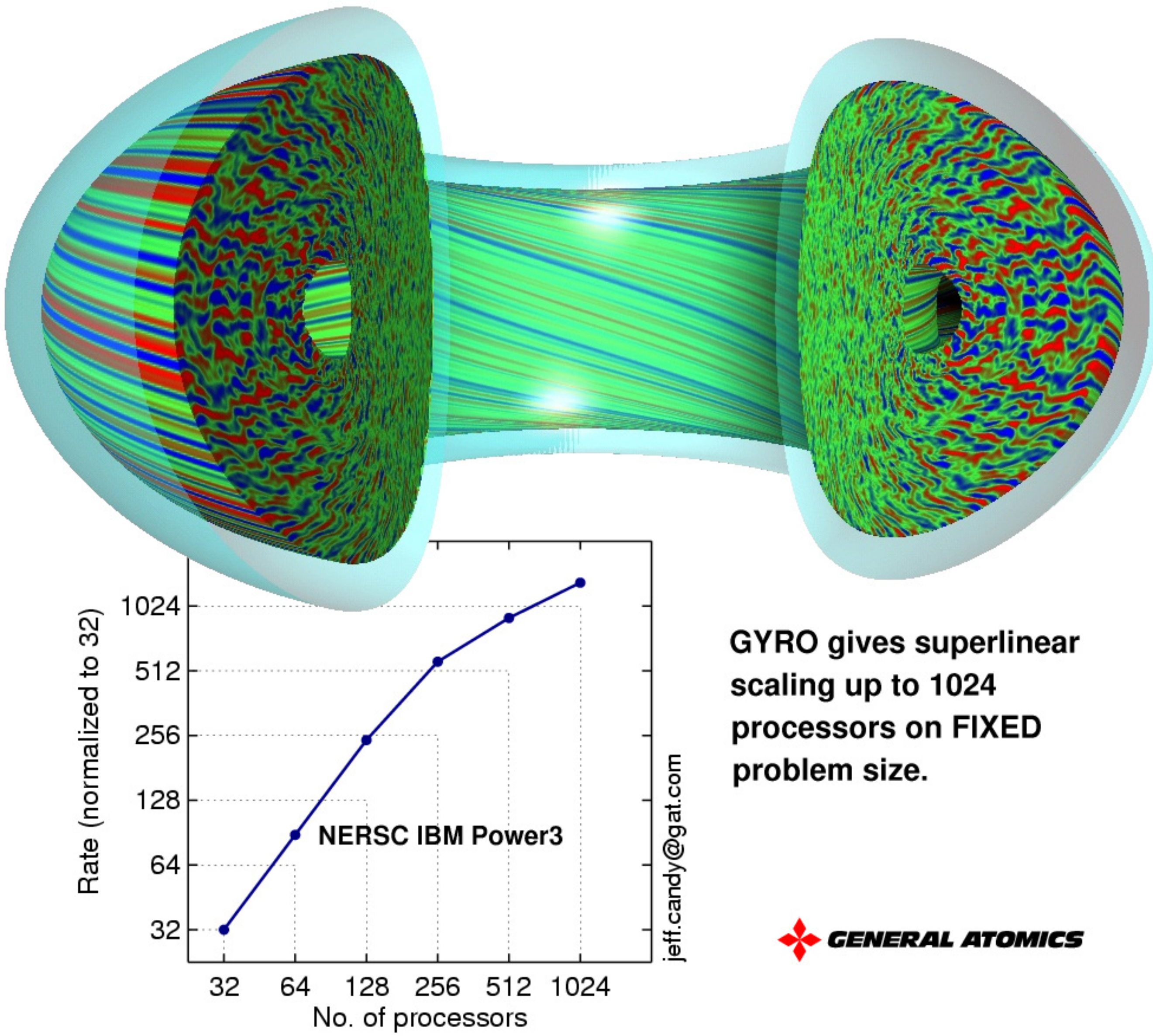


(Candy & Waltz, GA 2003)

Central temp ~ 10 keV ~ 10^8 K

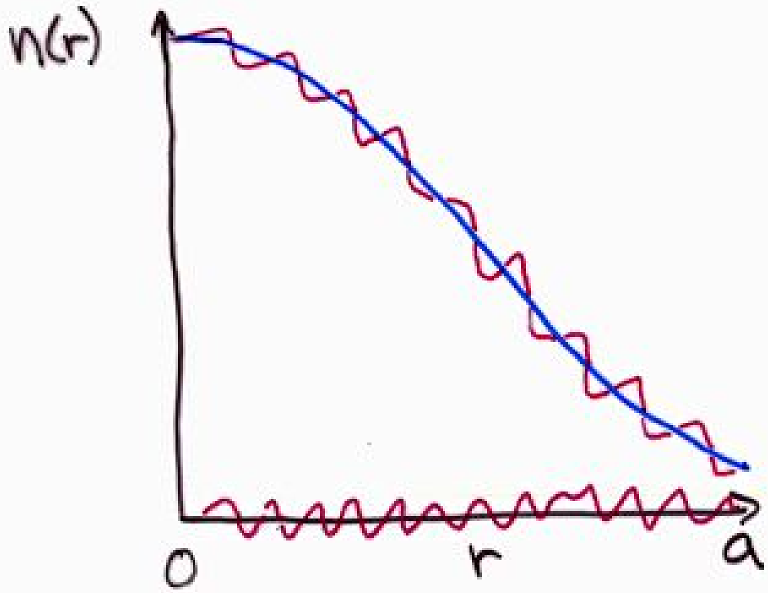
Large temperature gradient
$\rightarrow$ turbulent eddies
$\rightarrow$ cools plasmas
$\rightarrow$ determines plasma size needed for fusion ignition

Major progress in last decade: detailed nonlinear simulations (first 3-D fluid approximations, now 5-D $f(\vec{x}, v_{||}, v_{\perp}, t)$)
& detailed understanding



Comprehensive 5-D computer simulations of core plasma turbulence being developed by Plasma Microturbulence Project. Candy & Waltz (GA) movies shown: d3d.n16.2x_0.6_fly.mpg & supercyclone.mpg, from http://fusion.gat.com/comp/parallel/gyro_gallery.html (also at http://w3.pppl.gov/~hammett/refs/2004).

Microinstabilities are small-amplitude but still nonlinear



$$n = n_0(r) + \tilde{n}(\underset{\sim}{x}, t)$$

$$n_0 \gg \tilde{n}$$

but $\nabla n_0 \sim \nabla \tilde{n}$

Can locally flatten or reverse total gradient that was driving instability.

* Turbulence causes loss of plasma to the wall, but confinement still $\times 10^5$ better than without $\underset{\sim}{B}$.

If no $\underset{\sim}{B}$, loss time $\sim \frac{a}{v_t} \sim$ 1 μsec

with $\underset{\sim}{B}$, expts. measure $\sim$ 0.1 – 1.0 sec.

Define Problem in Mathematical Terms

Define Problem in Mathematical Terms

"Gyrokinetic equations" describe small scale turbulence in magnetized plasma; written down from 1978-1982

Define Problem in Mathematical Terms

"Gyrokinetic equations" describe small scale turbulence in magnetized plasma; written down from 1978-1982

Equations describe self-consistent evolution of 5-dimensional distribution functions (one for each plasma species) in time:

$$h = h_s(x, y, z, \varepsilon, \mu; t)$$

Define Problem in Mathematical Terms

"Gyrokinetic equations" describe small scale turbulence in magnetized plasma; written down from 1978-1982

Equations describe self-consistent evolution of 5-dimensional distribution functions (one for each plasma species) in time:

$$h = h_s(x, y, z, \varepsilon, \mu; t)$$

$$\boxed{\frac{\partial h}{\partial t} + \left(\mathrm{v}_{\parallel}\hat{b} + \vec{\mathrm{v}}_d\right) \bullet \nabla h + \hat{b} \times \nabla \chi \bullet \nabla\left(h + F_0\right) + q\frac{\partial F_0}{\partial \varepsilon}\frac{\partial \chi}{\partial t} = C(h)}$$

Define Problem in Mathematical Terms

"Gyrokinetic equations" describe small scale turbulence in magnetized plasma; written down from 1978-1982

Equations describe self-consistent evolution of 5-dimensional distribution functions (one for each plasma species) in time:

$$h = h_s(x, y, z, \varepsilon, \mu; t)$$

$$\boxed{\frac{\partial h}{\partial t} + \left(\mathrm{v}_{\parallel}\hat{b} + \vec{\mathrm{v}}_d\right) \bullet \nabla h + \hat{b} \times \nabla \chi \bullet \nabla\left(h + F_0\right) + q\,\frac{\partial F_0}{\partial \varepsilon}\frac{\partial \chi}{\partial t} = C(h)}$$

Plus Maxwell's Eqs to get gyro-averaged potential:

$$\boxed{\chi = J_0\left(k_{\perp}\rho\right)\left(\Phi - \frac{\mathrm{v}_{\parallel}}{c}A_{\parallel}\right) + \frac{J_1\left(k_{\perp}\rho\right)}{k_{\perp}\rho}\frac{m\mathrm{v}_{\perp}^2}{\mathrm{q}}\frac{\delta B_{\parallel}}{B_0}}$$

Typical Resolution for Continuum Gyrokinetics

Typical moderate-resolution parameters for GYRO:

$$f\,(\,r,\quad n_\phi,\quad \tau_{||},\quad E,\quad \lambda\,)$$

$$140 \times 32 \times 12 \times 8 \times 16 \quad \times\ 2 \text{ species}$$

Much larger simulations also done.

Continuum/Eulerian Approach to Electromagnetic Gyrokinetic Turbulence

GS2 (Dorland & Kotschenreuther), GENE (Jenko), and GYRO (Candy & Waltz) have demonstrated that direct Eulerian simulations of microturbulence using the 5-D electromagnetic gyrokinetic equations can be effective, by

(1) Using modern massively parallel supercomputers and clusters, and

(2) Using modern advanced algorithms, including

- implicit / semi-implicit methods (or carefully designed explicit methods)

- pseudo-spectral and/or Arakawa treatment of nonlinearities (preserves all 3 conservation properties of Poisson bracket nonlinearities)

- pseudo-spectral and/or high-order upwind advection algorithms: very low dissipation at long wavelengths, effective sink at small scales.

- high-order velocity-space integration algorithms,

- efficient field-aligned coordinate systems, …

Continuum/Eulerian Approach to Electromagnetic Gyrokinetic Turbulence

GS2 (Dorland & Kotschenreuther) http://gs2.sourceforge.net
GENE (Jenko) http://www.ipp.mpg.de/~fsj/
GYRO (Candy & Waltz) http://fusion.gat.com/comp/parallel/

These codes widely used by many to study plasma turbulence in fusion devices.
GYRO is currently the most comprehensive gyrokinetic code available:

- Gyrokinetic ions (multiple species) & adiabatic/drift-kinetic/gyrokinetic electrons
- Trapped and passing electrons (and ions) for Trapped Electron Mode
- Pitch-angle scattering collision operator (TEM / neoclassical effects)
- Finite beta magnetic fluctuations as well as electrostatic fluctuations (important for kinetic-ballooning modes, magnetic flutter contribution to transport)
- General shaped tokamak geometry
- Equilibrium ExB and parallel velocity shear
- Finite-ρ_* effects (profile shear stabization, nonlocal transport)…

Nevertheless, a lot of interesting work remains to be done: more tests against experiments, particle transport, transport barrier formation, shaping effects, understand scalings, couple to transport codes for complete predictive ability, &:

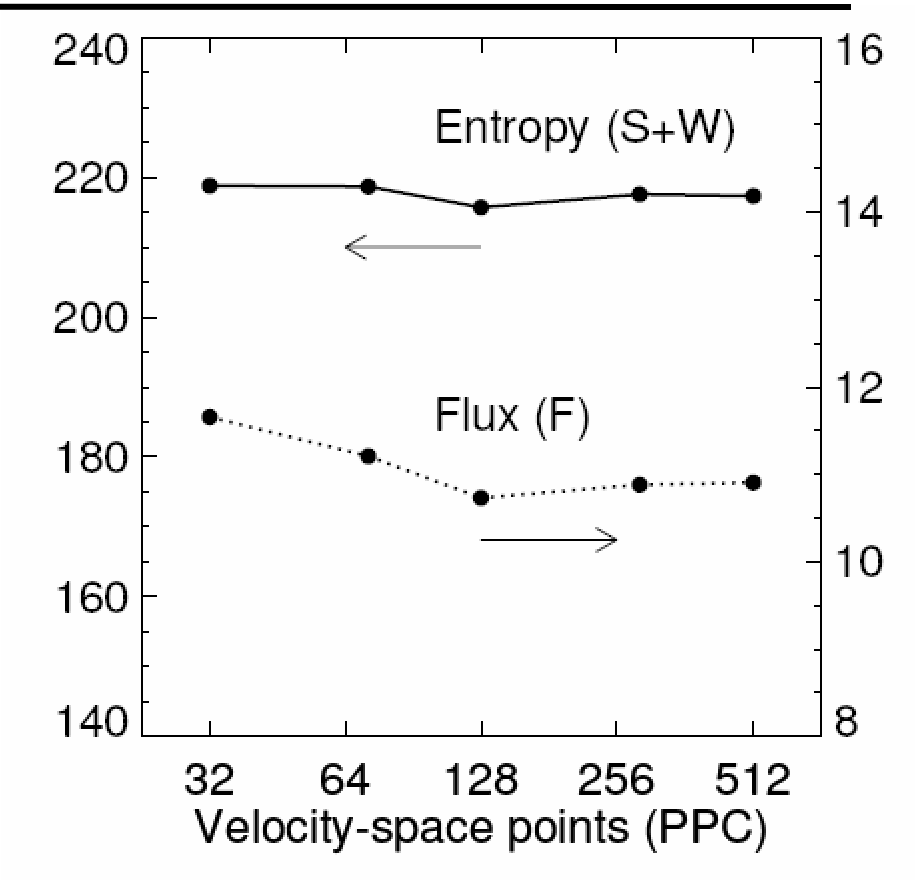
edge simulations (new codes needed to do gyrokinetics in the edge, challenging…)

GYRO well-converged w/ velocity resolution

Fluctuations in *f* cascade to small spatial scales where dissipation needed for steady state is provided by hyperdiffusion from high-order upwind algorithms.

Fluctuations in *f* also cascade to small velocity scales where collisions (or hypercollisions) are also a sink.

(Hyperdiffusion was more important in this low ν case.)



Note suppressed zeros!

Candy & Waltz, PoP 2006, “Velocity-space resolution, entropy production & upwind dissipation in Eulerian gyrokinetic simulations”
http://fusion.gat.com/comp/parallel/gyro_publications.html

Complex 5-dimensional Computer Simulations being developed

- Solving gyro-averaged kinetic equation to find time-evolution of particle distribution function

  $$f(\vec{\mathbf{x}}, E, v_{||}/v, t)$$

- Gyro-averaged Maxwell's Eqs. determine Electric and Magnetic fields
- "typical" grid 96x32x32 spatial, 10x20 velocity, x 3 species for 10^4 time steps.
- Various advanced numerical methods: implicit, semi-implicit, pseudo-spectral, high-order finite-differencing and integration, efficient field-aligned coordinates, Eulerian (continuum) & Lagrangian (particle-in-cell).

Gyrokinetic Eq. Summary

Gyro-averaged, non-adiabatic part of 5-D particle distribution function: $f_s = f_s(\mathbf{x}, \varepsilon, \mu, t)$ determined by gyrokinetic Eq. (in deceptively compact form):

$$\frac{\partial f}{\partial t} + \left(v_{\|}\hat{\mathbf{b}} + \mathbf{v}_d\right)\cdot \nabla f + \underbrace{\hat{\mathbf{b}} \times \nabla \chi \cdot \nabla\left(f + F_0\right)} + q\frac{\partial F_0}{\partial \varepsilon}\frac{\partial \Phi}{\partial t} = C(f)$$

Generalized Nonlinear ExB Drift
Incl. Magnetic fluctuations

$\chi(\mathbf{x},t)$ is gyro-averaged, generalized potential. Electric and magnetic fields from gyro-averaged Maxwell's Eqs.

$$\chi = J_0\left(\tfrac{k_\perp v_\perp}{\Omega}\right)\left(\phi - \frac{v_\|}{c}A_\|\right) + \frac{J_1\left(\tfrac{k_\perp v_\perp}{\Omega}\right)}{\tfrac{k_\perp v_\perp}{\Omega}}\frac{m v_\perp^2}{q}\frac{\delta B_\|}{B}$$

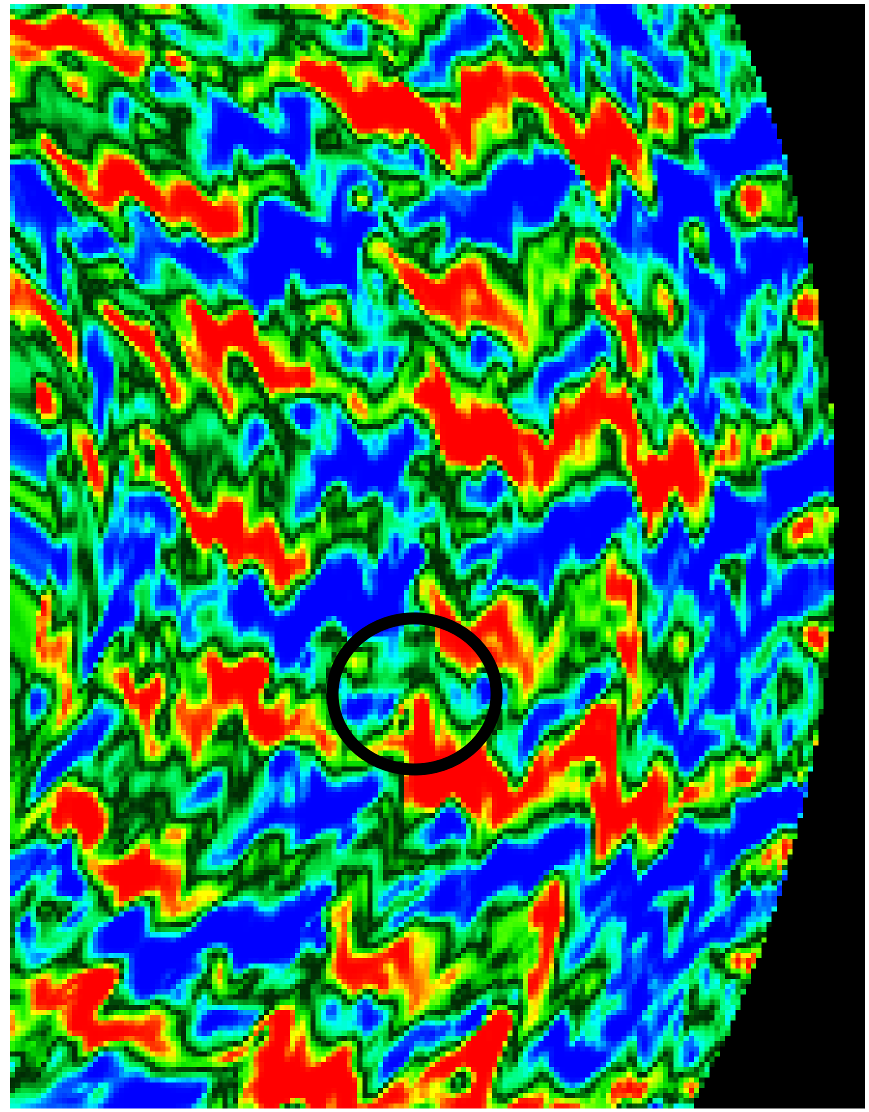
Bessel Functions represent averaging around particle gyro-orbit

Gyroaveraging eliminates fast time scales of particle gyration (10 MHz- 10 GHz)

Easy to evaluate in pseudo-spectral codes. Fast multipoint Padé approx. in other codes.

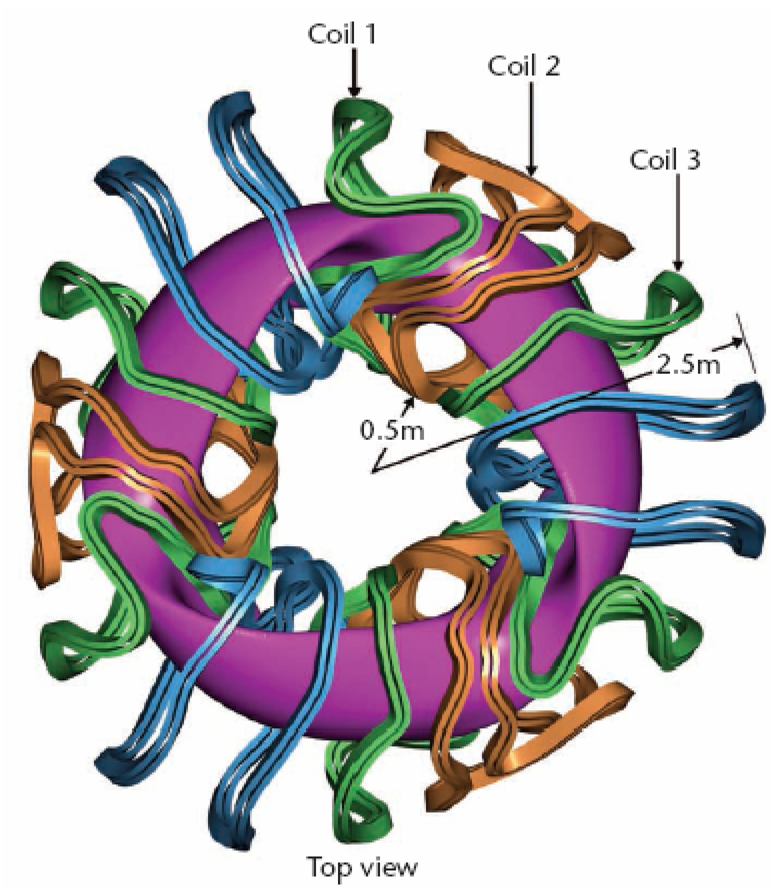
$$\chi = J_0(k_\perp \rho)\,\Phi$$

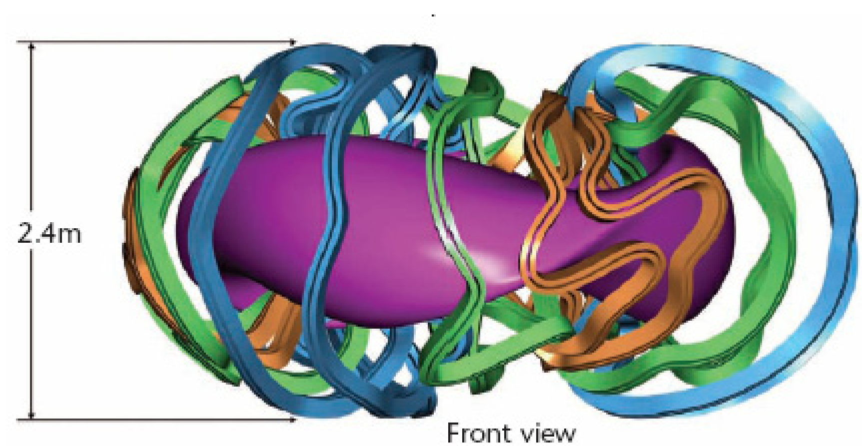
$$\chi(\vec{\mathbf{x}}) = \oint d\theta \;\Phi(\vec{\mathbf{x}} + \vec{\boldsymbol{\rho}}(\theta))$$



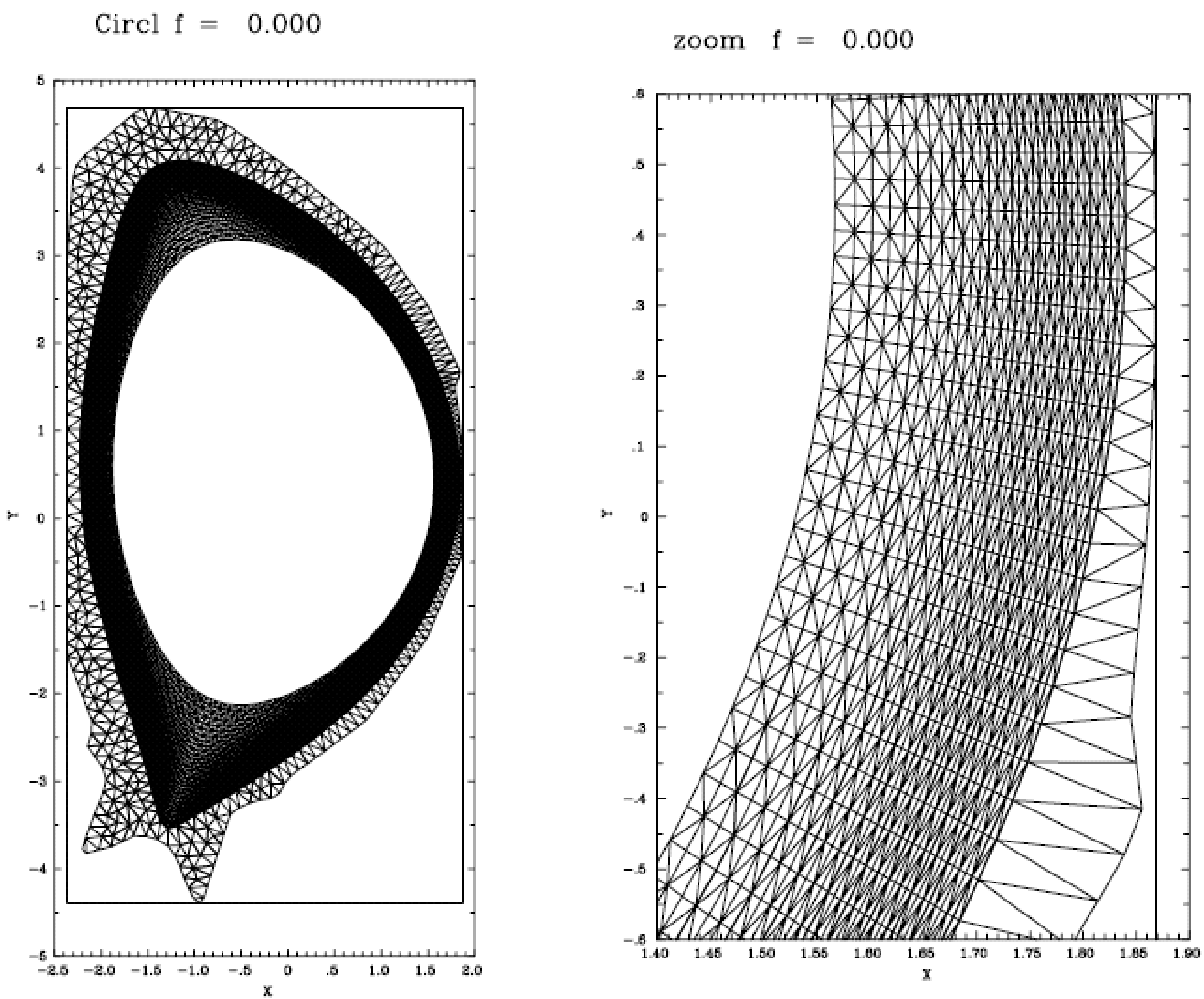
Improved Stellarators Being Built

- Magnetic field twist and shear provided by external coils, not plasma currents. More stable?
- Computer optimized designs much better than 1950-60 slide rules?
- Quasi-toroidal symmetry allows plasma to spin toroidally: shear flow stabilization?



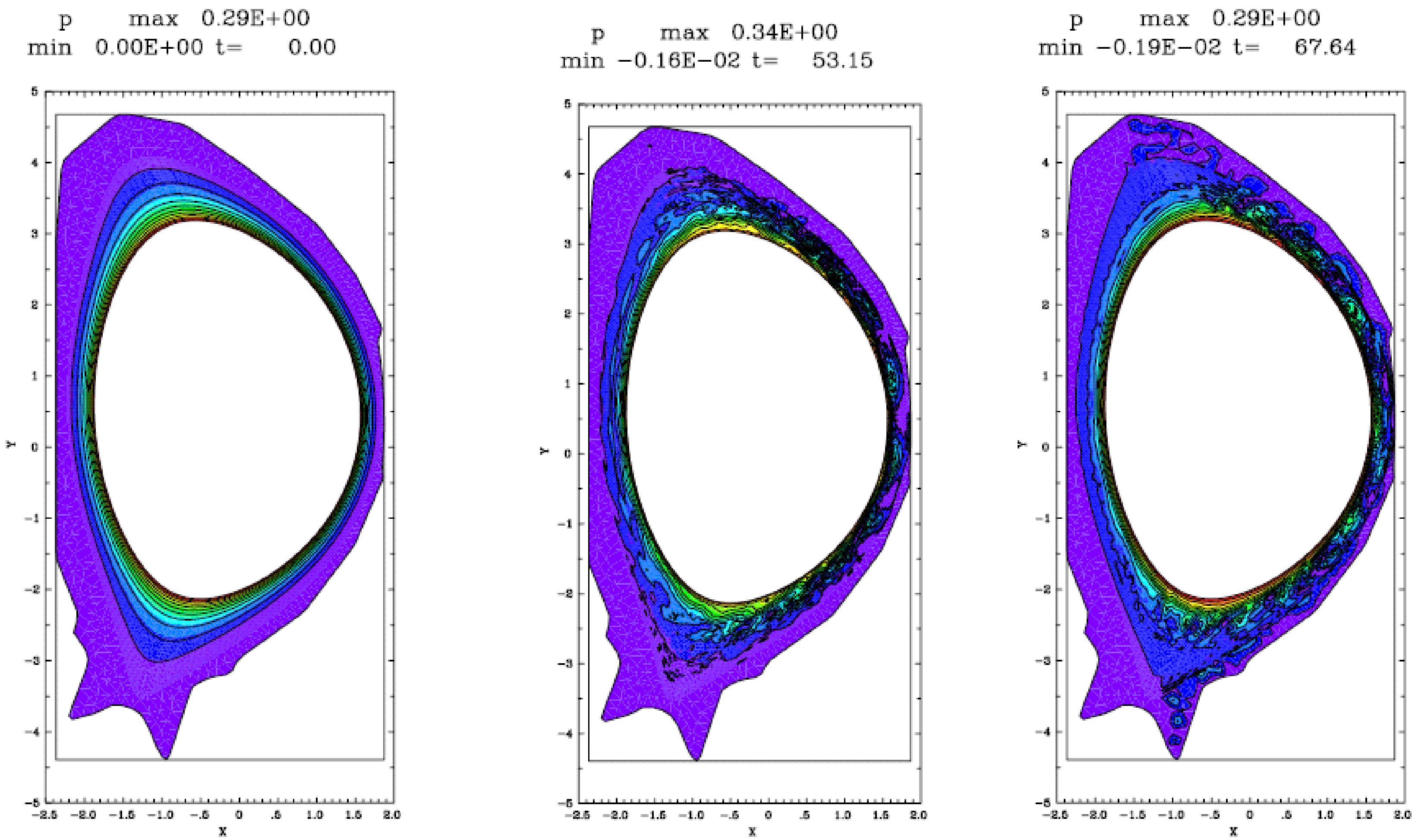


ITER mesh



Strauss

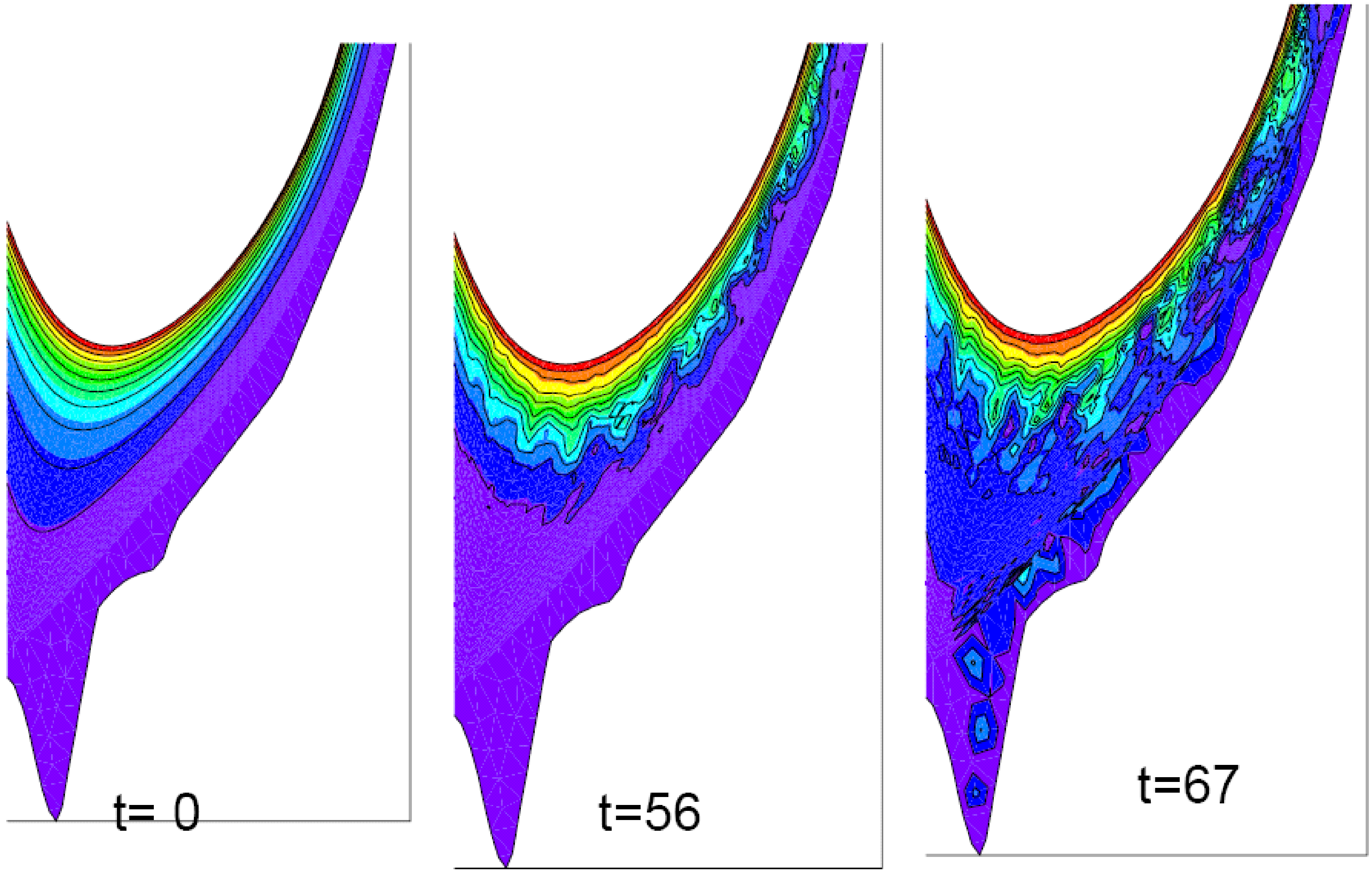
ITER ELM: pressure time evolution



Strauss

Center for Extended Magnetohydrodynamic Modelling (CEMM), S. Jardin et al., 2006 http://w3.pppl.gov/theory/PSACI%2006/Jardin%20CEMM(2006)-rev.pdf

ELM pressure: initial, mode growth, outflow



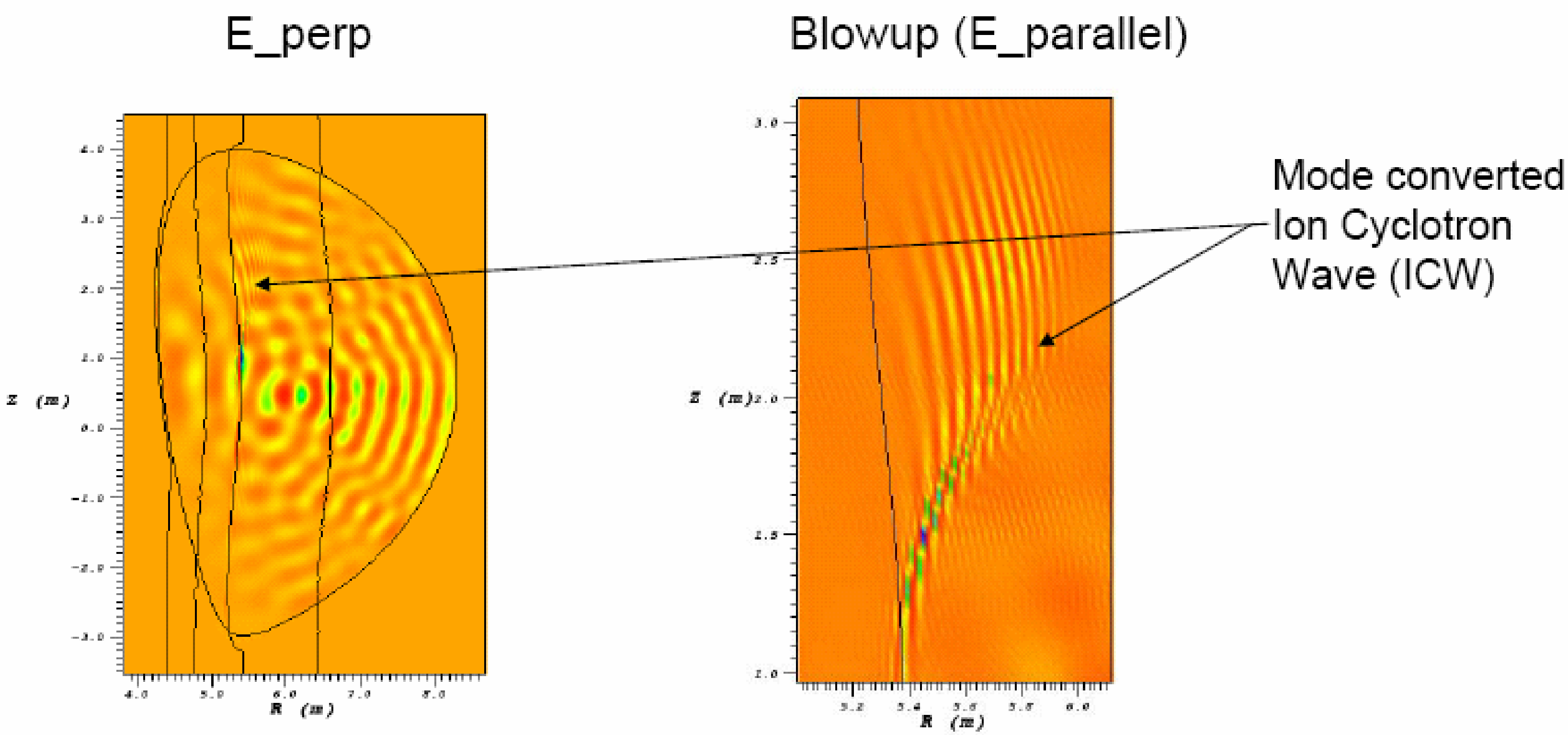
Strauss

Center for Extended Magnetohydrodynamic Modelling (CEMM), S. Jardin et al., 2006 http://w3.pppl.gov/theory/PSACI%2006/Jardin%20CEMM(2006)-rev.pdf

Calculations on the Cray XT-3 have allowed the first simulations of mode conversion in ITER

ITER with D:T:HE3 = 20:20:30 with $N_R = N_Z = 350$, $f = 53$ MHz, $n = 2.5 \times 10^{19}$ m^{-3}

(4096 processors for 1.5 hours on the Cray XT-3)



Future Work – Will extend this MC scenario to more ITER relevant densities ($\approx 7 \times 10^{19}$m^{-3})

Paul Bonoli et al., 2006, Center for Simulation of Wave-Plasma Interactions http://w3.pppl.gov/theory/PSACI%2006/Bonoli%20PSAC_RFSciDAC_Rev.pdf

Computational Plasma Physics: Powerful New Tools of Scientific Discovery

- Exponential growth of computer power means that a lot of important and interesting problems are becoming tractable by computer solutions.

- The importance of good numerical algorithms

  Pitfall of naive algorithm for diffusion equation

- Examples of cutting edge computational plasma physics, such as:

  Simulating 5-dimensional plasma turbulence in fusion devices

- Computer simulations can be fun!

Computational References

- **Numerical Recipes**, W.H. Press et al., www.nr.com (in C/C++, Fortran 77/90) entertainingly written, many insights, good place to start
- Durran, Numerical Methods for Wave Equations in Geophysical Fluid Dynamics
- LeVeque, Finite Volume Methods for Hyperbolic Problems
- Gershenfeld, The Nature of Mathematical Modeling
- "Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws," Chi-Wang Shu, NASA/CR-97-206253, ICASE Report No. 97-65, http://library-dspace.larc.nasa.gov/dspace/jsp/handle/2002/14650
- High Resolution Non-Oscillatory Central Schemes: http://www.cscamm.umd.edu/people/faculty/tadmor/centralstation/

- Useful web sites / high quality numerical software:
- www.netlib.org Vast repository of high quality (& free) numerical software
- PETSC (library of optimized parallelized algorithms for scientific computing, PDEs and Linear solvers)
- FFTW (Fastest FFT in the West)
- www.scidac.org DOE Scientific Discovery Through Advanced Computing Initiative

Selected Further References

- This talk: http://fire.pppl.gov & http://w3.pppl.gov/~hammett
- Plasma Microturbulence Project http://fusion.gat.com/theory/pmp
- GYRO code and movies http://fusion.gat.com/comp/parallel/gyro.html
  & http://w3.pppl.gov/~hammett/talks/2004
- GS2 gyrokinetic code http://gs2.sourceforge.net
- Center for Multiscale Plasma Dynamics http://cmpd.umd.edu/
- My gyrofluid & gyrokinetic plasma turbulence references: http://w3.pppl.gov/~hammett/papers/
- "ENDING THE ENERGY STALEMATE: A Bipartisan Strategy to Meet America's Energy Challenges", The National Commission on Energy Policy, December 2004. http://www.energycommission.org/
- "Anomalous Transport Scaling in the DIII-D Tokamak Matched by Supercomputer Simulation", Candy & Waltz, Phys. Rev. Lett. 2003
- "Burning plasma projections using drift-wave transport models and scalings for the H-mode pedestal", Kinsey et al., Nucl. Fusion 2003
- "Electron Temperature Gradient Turbulence", Dorland, Jenko et al. Phys. Rev. Lett. 2000
- "Comparisons and Physics Basis of Tokamak Transport Models and Turbulence Simulations", Dimits et al., Phys. Plasmas 2000.