SAND2007-3360C

General-Purpose Algorithms for Large-Scale Time-Periodic Flow Problems

Andy Salinger, Eric Phipps

Sandia National Laboratories

Albuquerque, New Mexico, USA

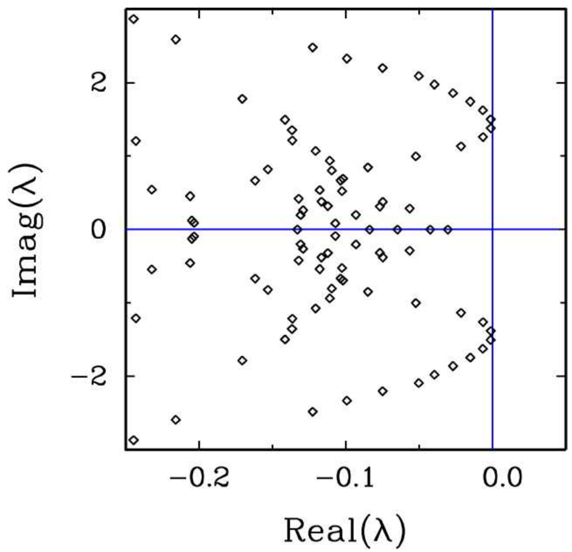
SIAM Dynamical Systems'07 May 28, 2007 Snowbird Utah

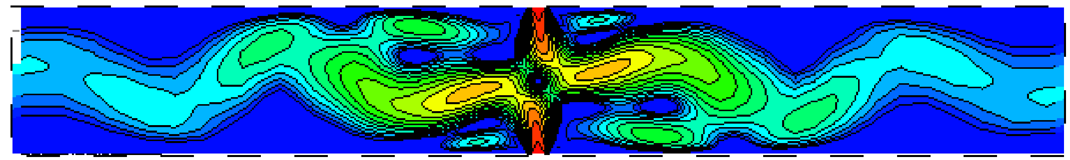
MS7 Bifurcation Analysis of Incompressible Flow Problems

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

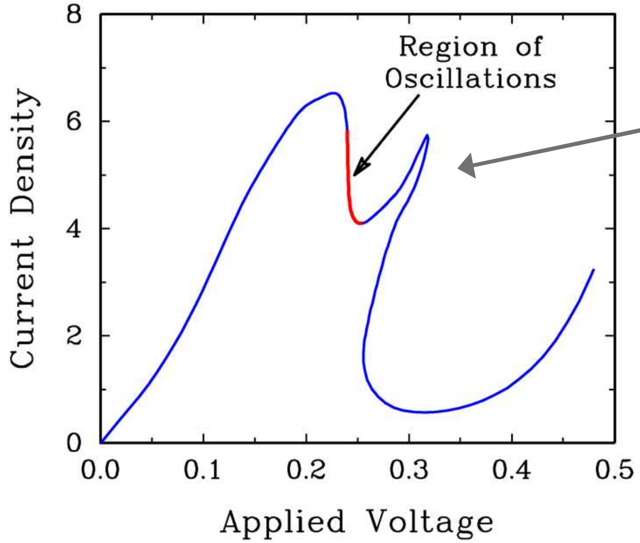
Sandia National Laboratories

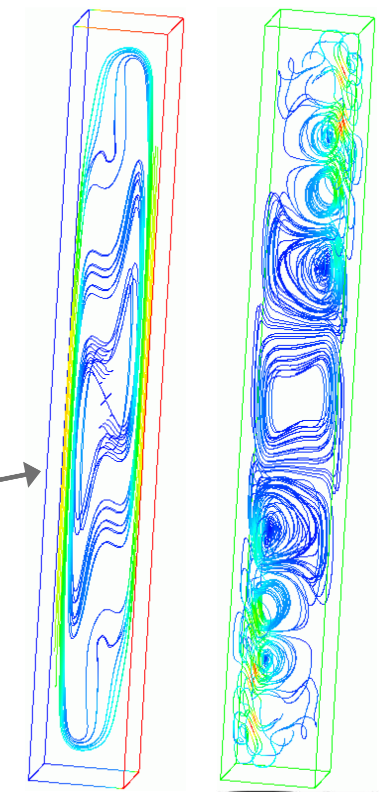
Hopf Bifurcations and Oscillatory Solutions Occur in Many Physical Systems





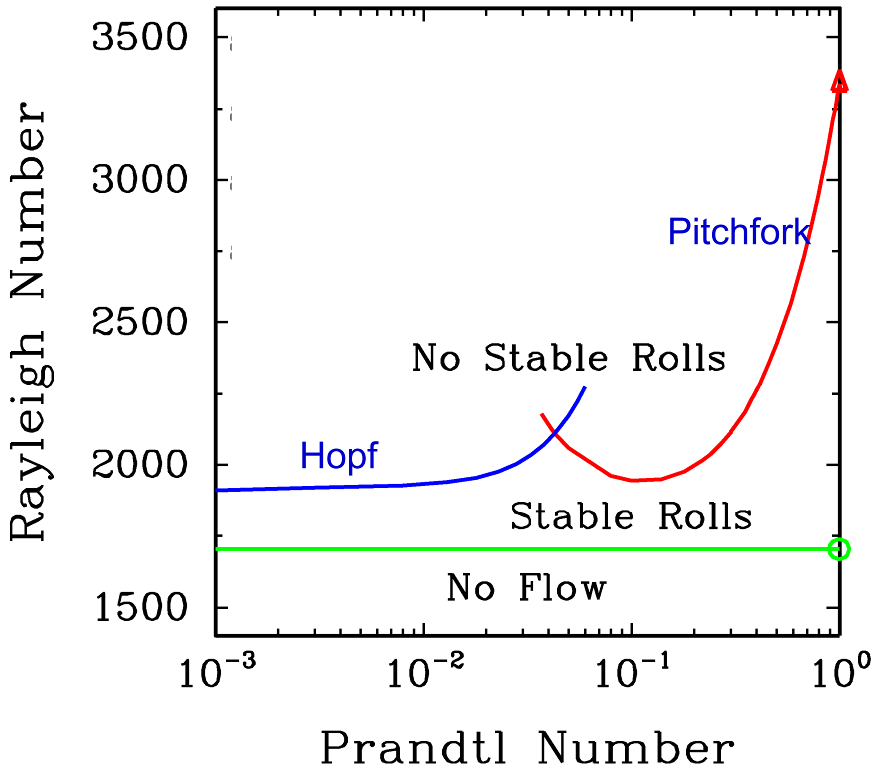
- Opposing Jets
- Vortex Shedding
- Predator-Prey models
- Flutter
- Resonant Tunneling Diodes
- Natural convection
- Reverse Flow Reactors
- Electronic Circuits

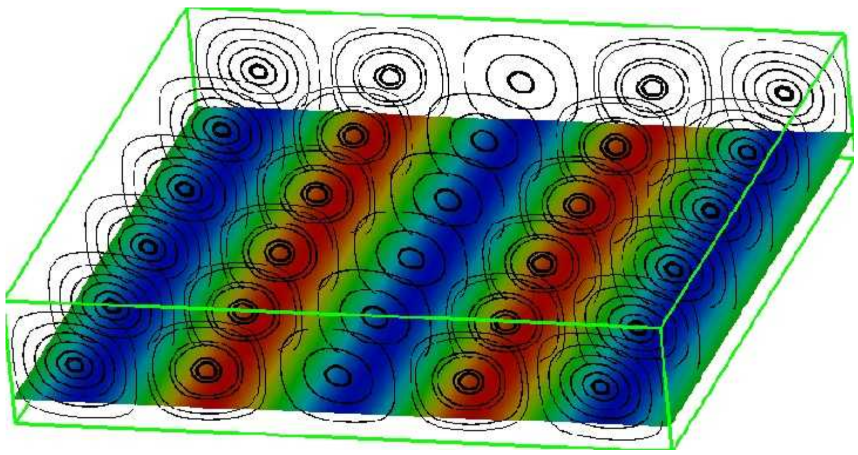


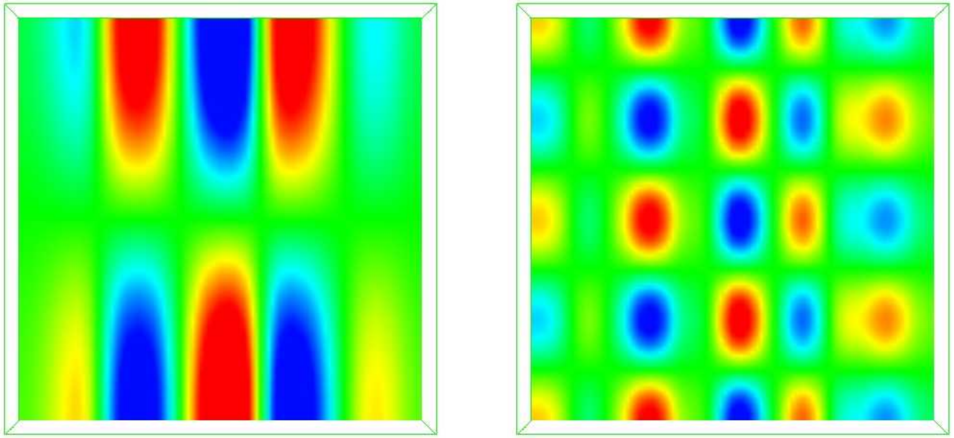


Sandia National Laboratories

Bifurcation Analysis of 3D Flows using Scalable Algorithms in LOCA/Trilinos







Destabilizing Eigenvectors

Space-Time "4D" Approach to Periodic Orbit Tracking

Transient Simulation of: $\mathbf{B}\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \lambda)$

First solve: $\mathbf{B}\frac{\mathbf{x}_1 - \mathbf{x}_0}{\Delta t} - \mathbf{f}(\mathbf{x}_1, \lambda) = 0$

Then solve: $\mathbf{B}\frac{\mathbf{x}_2 - \mathbf{x}_1}{\Delta t} - \mathbf{f}(\mathbf{x}_2, \lambda) = 0$

Then solve: $\mathbf{B}\frac{\mathbf{x}_3 - \mathbf{x}_2}{\Delta t} - \mathbf{f}(\mathbf{x}_3, \lambda) = 0$

Instead, solve for all solutions at once:

$$\mathbf{g}(\mathbf{y}, \lambda) = 0$$

where

$$\mathbf{y} = \begin{vmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{vmatrix}$$

$$\mathbf{g}_i = \mathbf{B}\mathbf{x}_i - \mathbf{B}\mathbf{x}_{i-1} - \Delta t\mathbf{f}(\mathbf{x}_i, \lambda)$$

… and with Newton solve:

$$\begin{vmatrix} (\mathbf{B} - \Delta t\mathbf{J}) & 0 & 0 & 0 & -\mathbf{B} \\ -\mathbf{B} & (\mathbf{B} - \Delta t\mathbf{J}) & 0 & 0 & 0 \\ 0 & -\mathbf{B} & (\mathbf{B} - \Delta t\mathbf{J}) & 0 & 0 \\ 0 & 0 & -\mathbf{B} & (\mathbf{B} - \Delta t\mathbf{J}) & 0 \\ 0 & 0 & 0 & -\mathbf{B} & (\mathbf{B} - \Delta t\mathbf{J}) \end{vmatrix} \begin{vmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ \Delta\mathbf{x}_3 \\ \Delta\mathbf{x}_4 \\ \Delta\mathbf{x}_5 \end{vmatrix} = \begin{vmatrix} -\mathbf{g}_1 \\ -\mathbf{g}_2 \\ -\mathbf{g}_3 \\ -\mathbf{g}_4 \\ -\mathbf{g}_5 \end{vmatrix}$$

$$\text{"}\mathbf{K}\Delta\mathbf{y} = -\mathbf{g}\text{"}$$

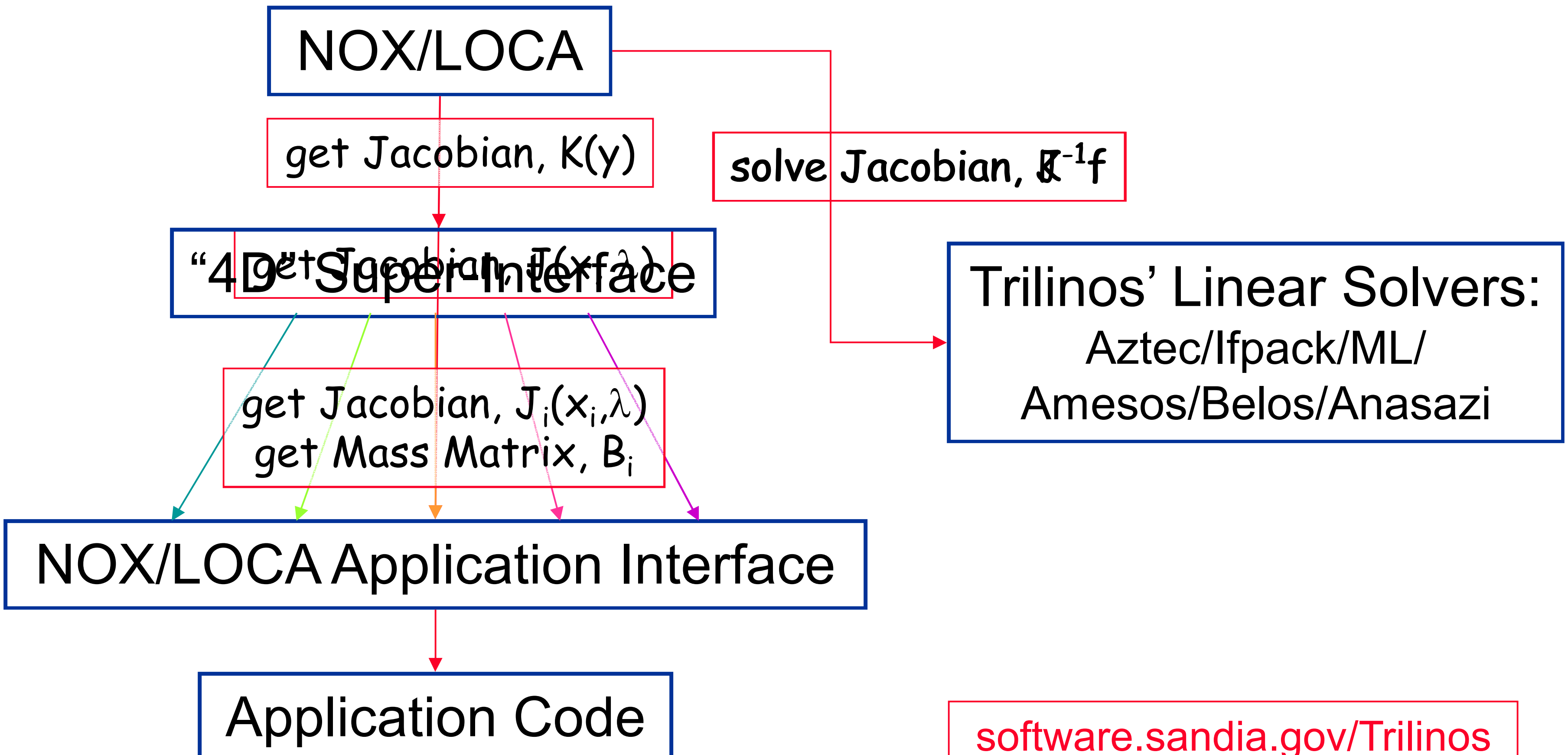
Why try this Space-Time "4D" Approach for Periodic Orbit Tracking?

1. **General purpose interface:** finite difference discretization of the time derivative does not require any changes to the physics package (not true of spectral methods)
2. **Analytic, Sparse Jacobian:** ... for codes that can produce an analytic Jacobian for an implicit time step (not true for shooting method)
3. **Time parallelism:** the matrix structure makes it relatively straightforward to parallelize over the time axis
4. **Straightforward to add Constraints:** design constraints or phase condition for autonomous systems

$$
\begin{vmatrix}
(\mathrm{B}-\Delta t\mathrm{J}) & 0 & 0 & 0 & -\mathrm{B} & \mathrm{g}_{1_\tau} \\
-\mathrm{B} & (\mathrm{B}-\Delta t\mathrm{J}) & 0 & 0 & 0 & \mathrm{g}_{2_\tau} \\
0 & -\mathrm{B} & (\mathrm{B}-\Delta t\mathrm{J}) & 0 & 0 & \mathrm{g}_{3_\tau} \\
0 & 0 & -\mathrm{B} & (\mathrm{B}-\Delta t\mathrm{J}) & 0 & \mathrm{g}_{4_\tau} \\
0 & 0 & 0 & -\mathrm{B} & (\mathrm{B}-\Delta t\mathrm{J}) & \mathrm{g}_{5_\tau} \\
0 & 0 & 0 & 0 & h_{x_5} & 0
\end{vmatrix}
\begin{vmatrix}
\Delta\mathrm{x}_1 \\ \Delta\mathrm{x}_2 \\ \Delta\mathrm{x}_3 \\ \Delta\mathrm{x}_4 \\ \Delta\mathrm{x}_5 \\ \Delta\tau
\end{vmatrix}
=
\begin{vmatrix}
-\mathrm{g}_1 \\ -\mathrm{g}_2 \\ -\mathrm{g}_3 \\ -\mathrm{g}_4 \\ -\mathrm{g}_5 \\ -h(x_5)
\end{vmatrix}
$$

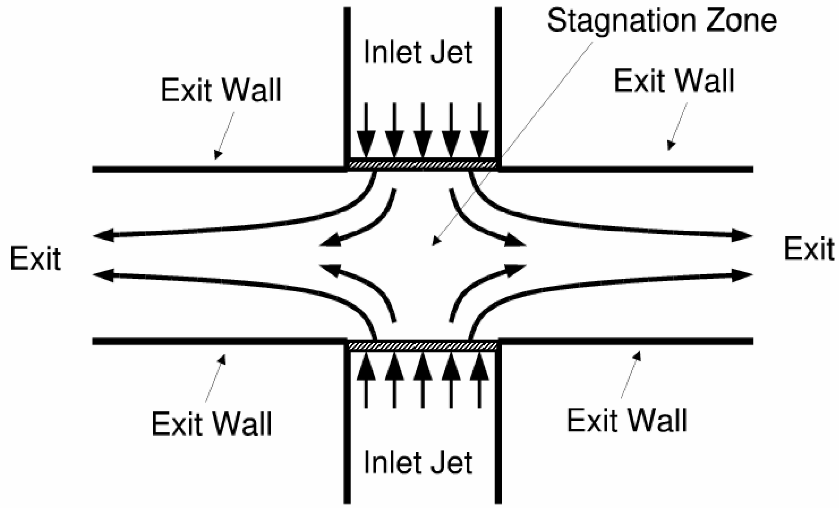
Sandia National Laboratories

Trilinos Implementation of Space-Time Capability: create a “Super” NOX/LOCA interface

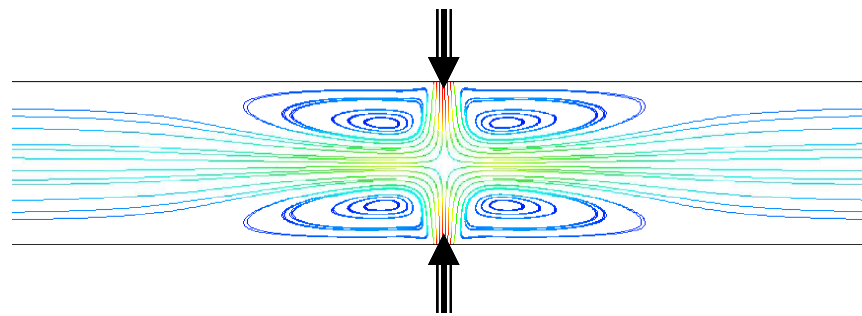


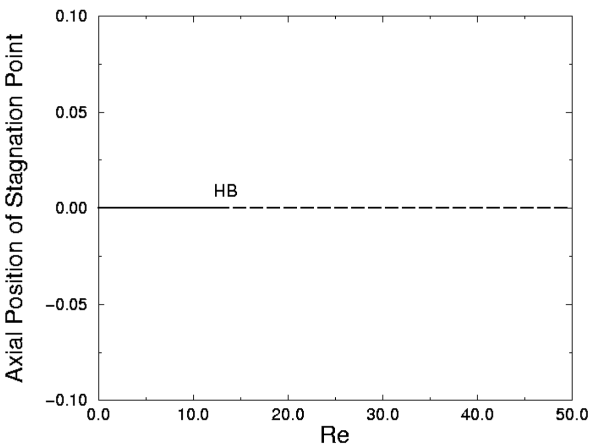
software.sandia.gov/Trilinos

LOCA has been used to analyze 2D and 3D PDE discretizations of over a Million unknowns

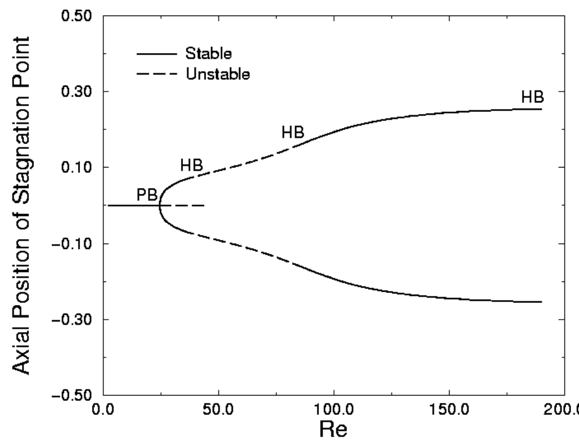


Stability Analysis of Impinging Jets, Pawlowski, Salinger, Shadid, Mountziaris, JFM (2005)

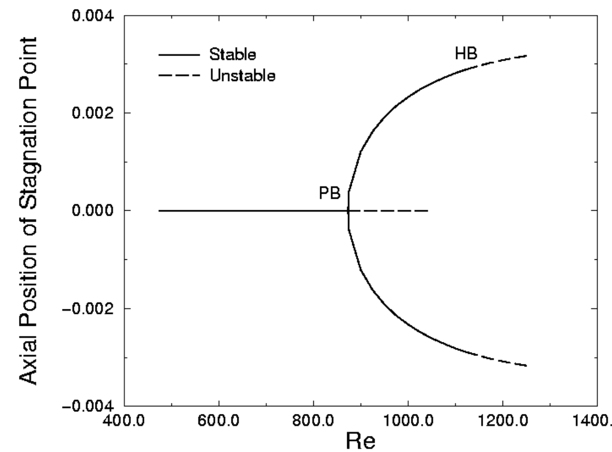




aspect ratio=0.05

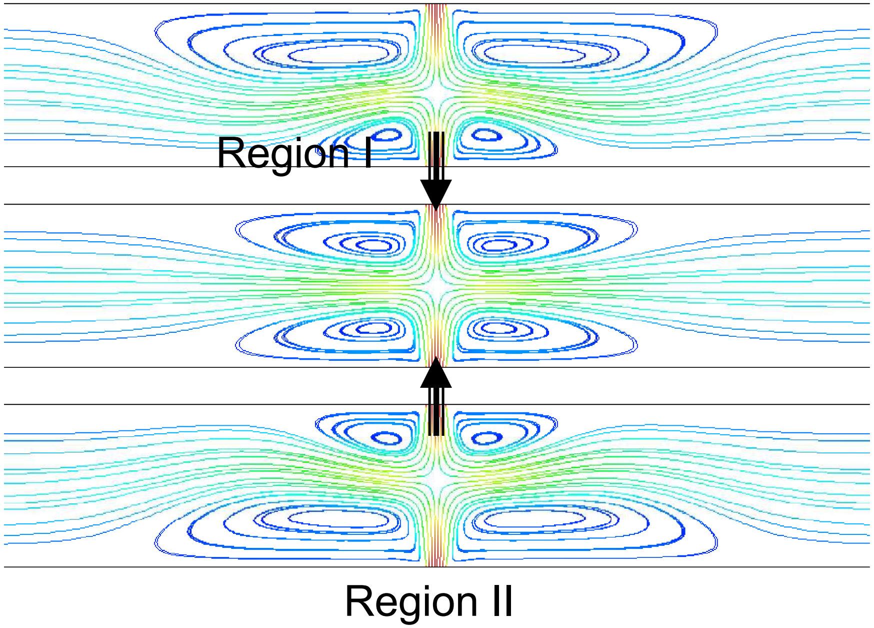


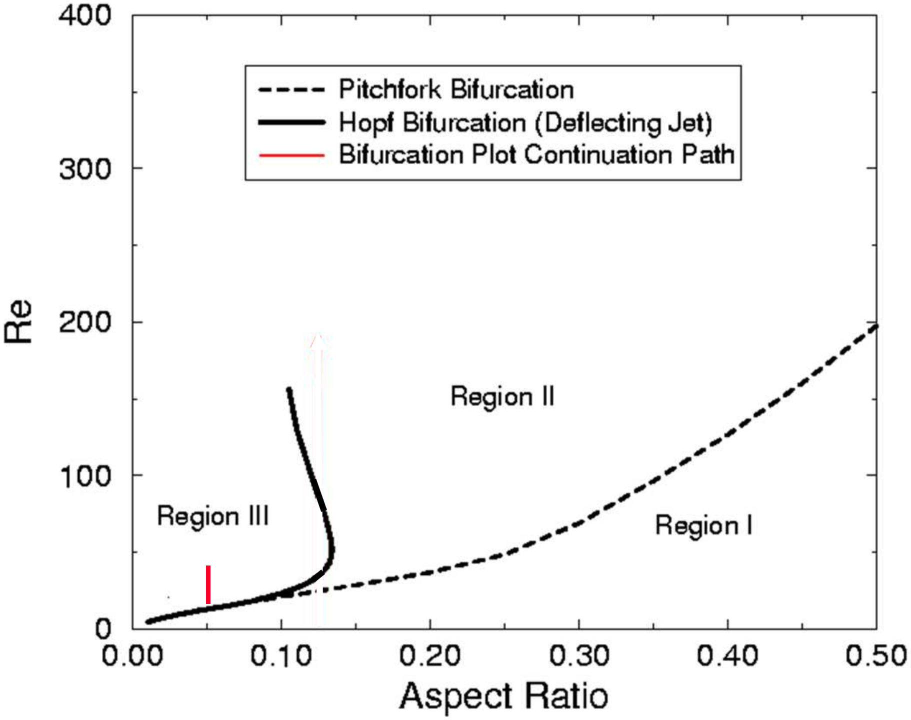
aspect ratio=0.125



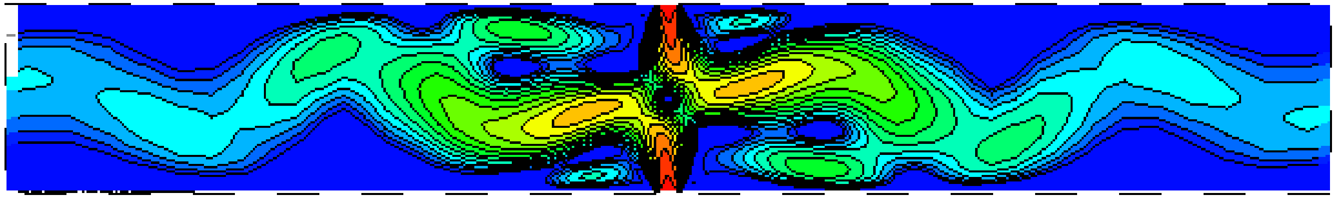
aspect ratio=1.0

Bifurcation tracking routines quickly delineate three flow regimes in 2-parameter space



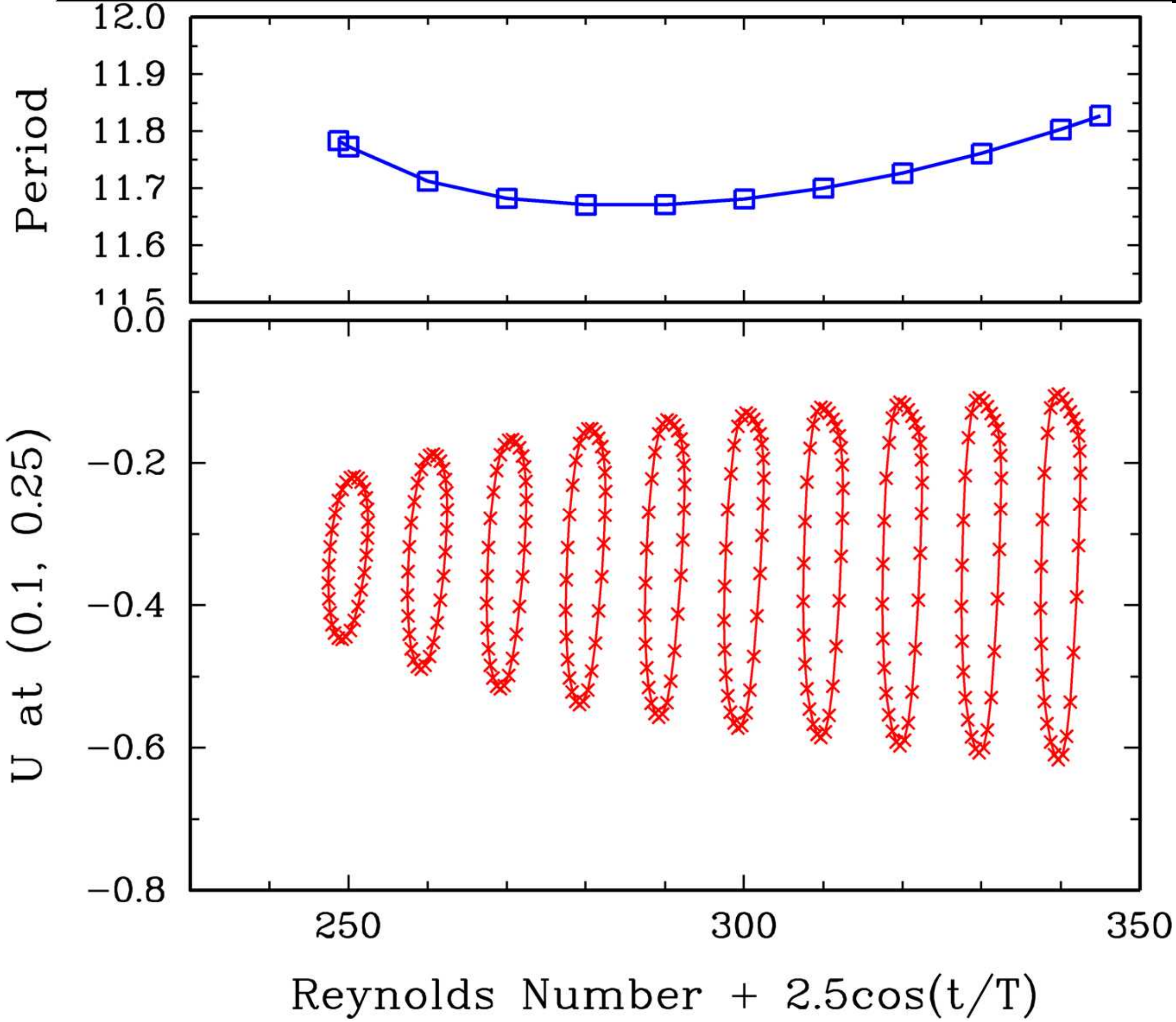


Region III



Sandia National Laboratories

Periodic Orbit Tracking for Impinging Jet Reactor



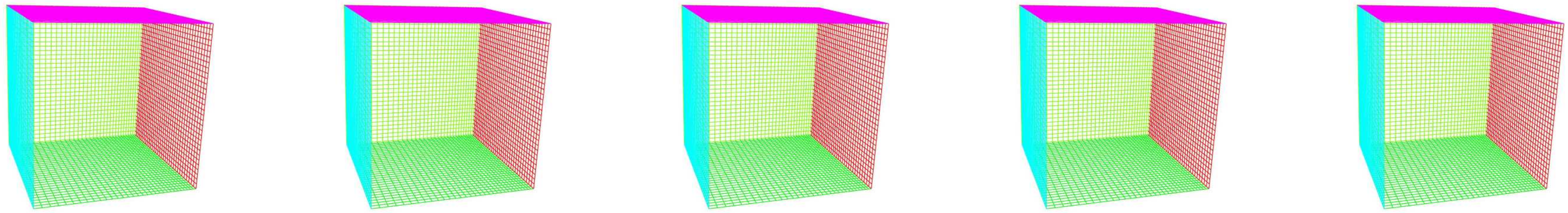
Phase condition: "Newton update is orthogonal to the flow":

$$M\dot{x} \cdot \Delta x = 0$$

Details:
7K nodes
21K spatial unknowns
30 time steps/period
620K total unknowns
30 Processors
~2 min / Newton iter

Sandia National Laboratories

Two-Level Parallelism: Space and Time



$\mathbf{J}(\mathbf{x}_1), \mathbf{B}(\mathbf{x}_1)$ $\quad \mathbf{J}(\mathbf{x}_2), \mathbf{B}(\mathbf{x}_2)$ $\quad \mathbf{J}(\mathbf{x}_3), \mathbf{B}(\mathbf{x}_3)$ $\quad \mathbf{J}(\mathbf{x}_4), \mathbf{B}(\mathbf{x}_4)$ $\quad \mathbf{J}(\mathbf{x}_5), \mathbf{B}(\mathbf{x}_5)$

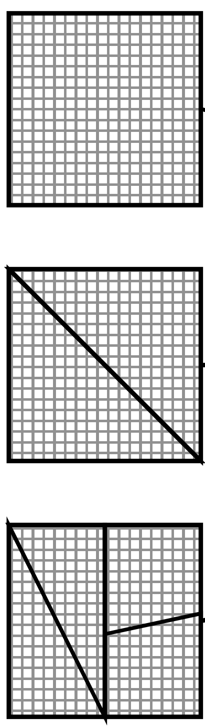
$$\begin{vmatrix} (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 & 0 & -\mathbf{B} \\ -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 & 0 \\ 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 \\ 0 & 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 \\ 0 & 0 & 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) \end{vmatrix} \begin{vmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ \Delta\mathbf{x}_3 \\ \Delta\mathbf{x}_4 \\ \Delta\mathbf{x}_5 \end{vmatrix} = \begin{vmatrix} -\mathbf{g}_1 \\ -\mathbf{g}_2 \\ -\mathbf{g}_3 \\ -\mathbf{g}_4 \\ -\mathbf{g}_5 \end{vmatrix}$$

Sandia National Laboratories

Space and Time can be partitioned independently, Ex: 4 Time Steps on 4 Procs

Spatial Partition

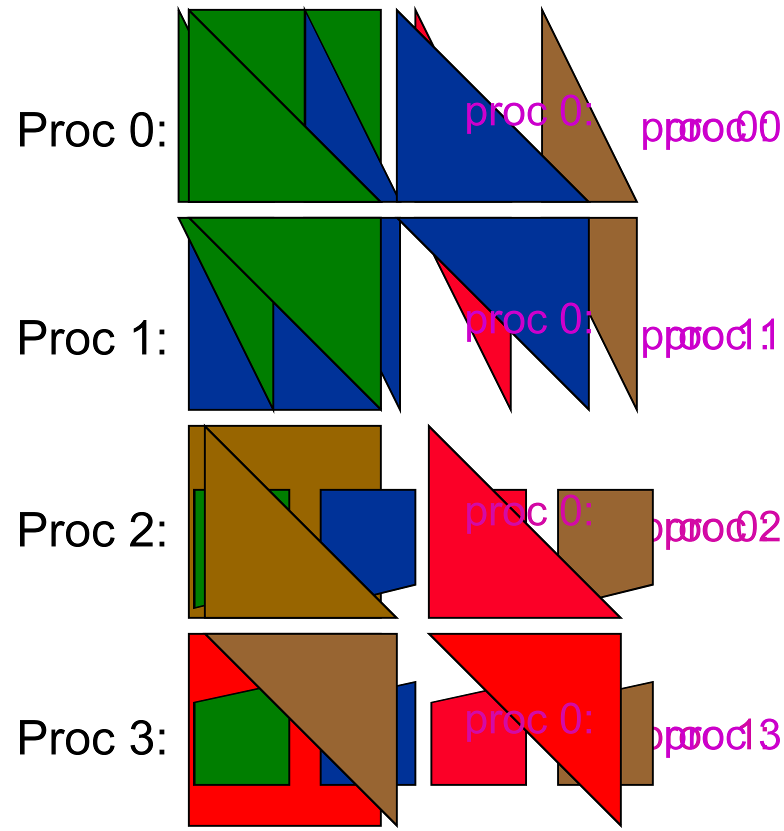
Space-Time Partition



Each processor owns 1 time step for the entire spatial domain

Each processor owns 2 time steps for $\frac{1}{2}$ of the spatial domain

Each processor owns 4 time steps for $\frac{1}{4}$ of the spatial domain

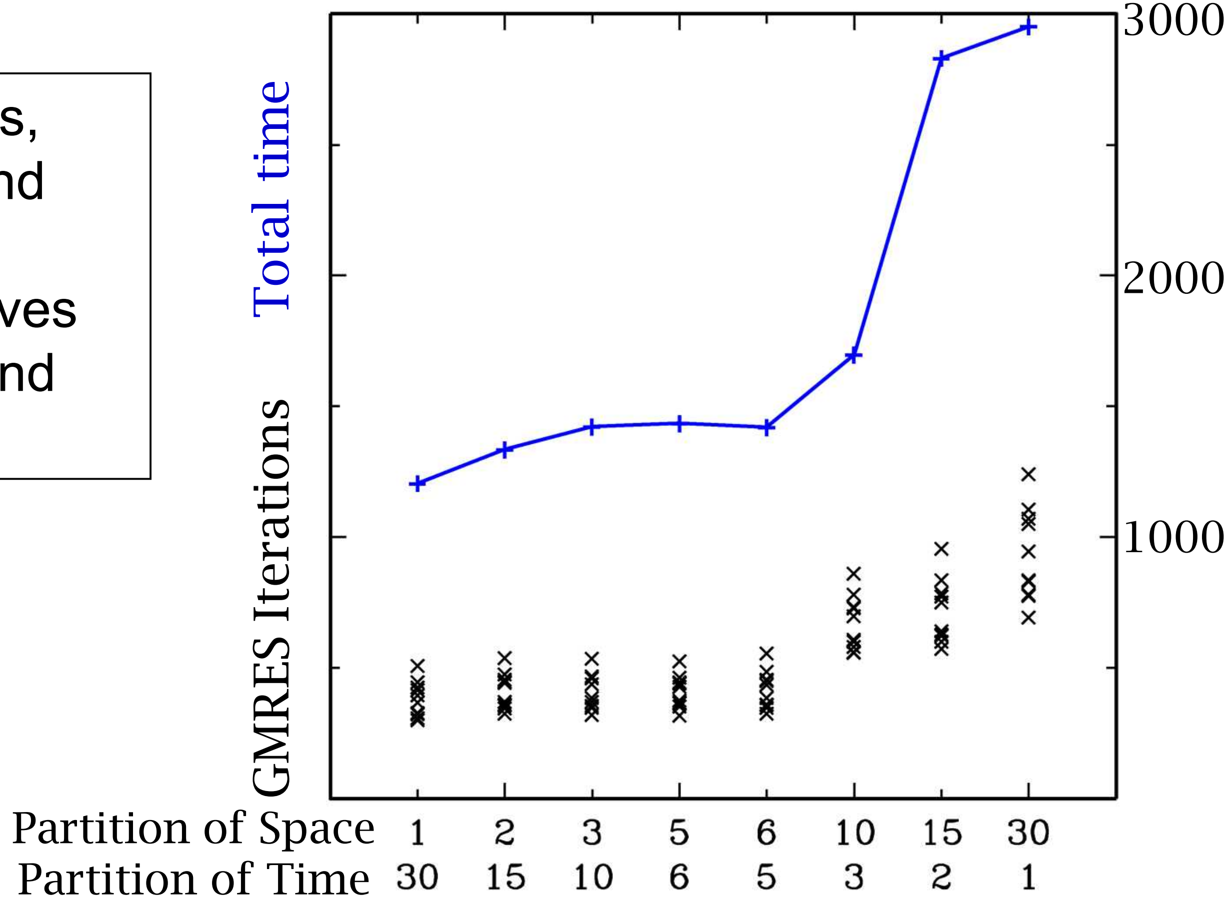


mpirun -np 4 salsa infile 1 4

Sandia National Laboratories

What is the most efficient parallel partitioning of space and time?

1. 30 total Processors, changing space and time partitioning
2. Only the linear solves matter for timing and parallel efficiency

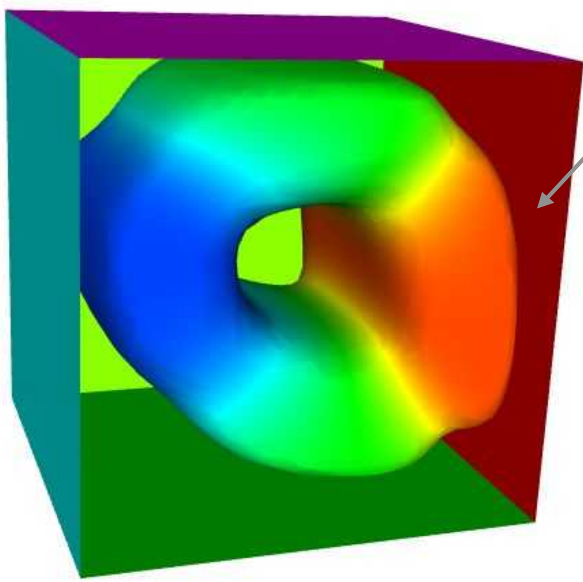


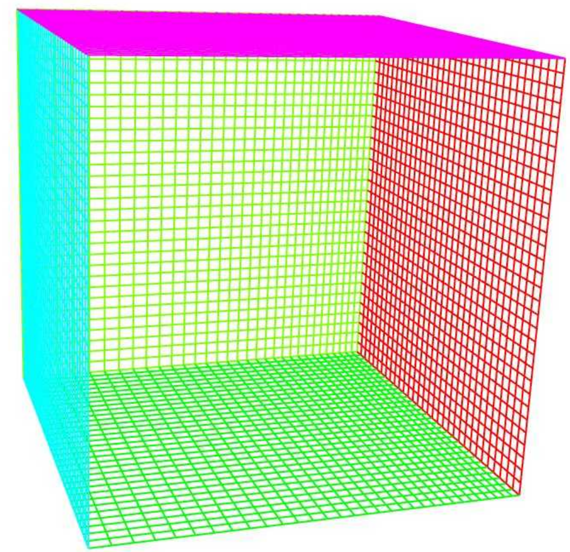
Sandia National Laboratories

4D Demonstration Problem

3D Buoyancy-driven flow in a cube with oscillatory heating

$$T(1.0, y, z, t) = sin(2\pi t)$$

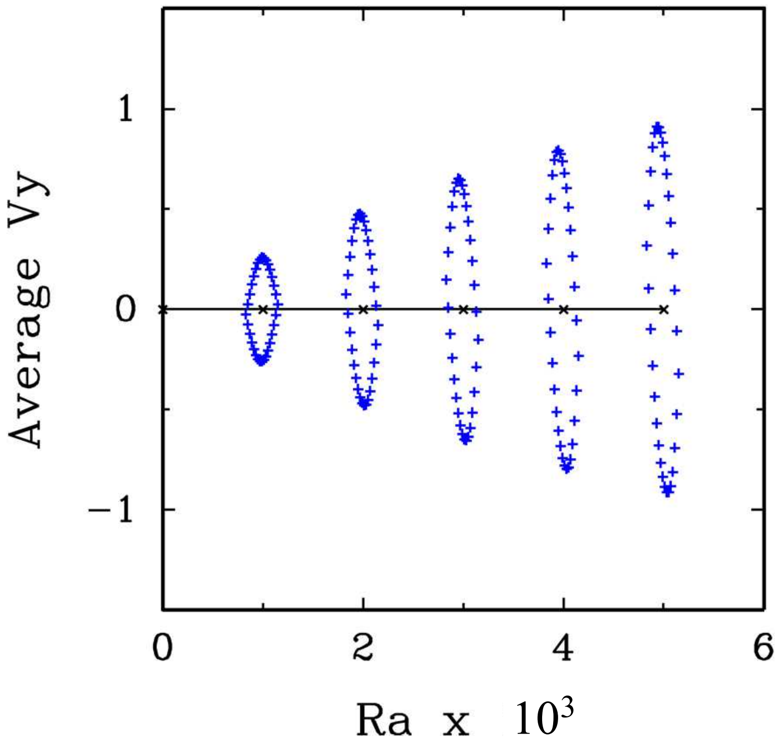


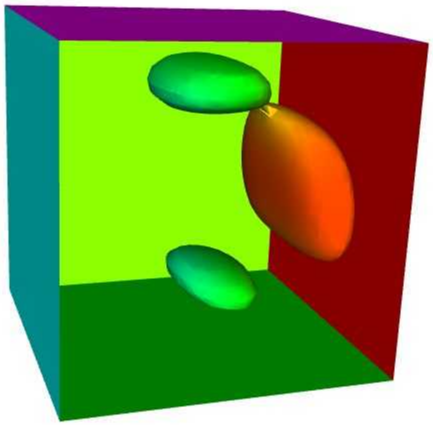


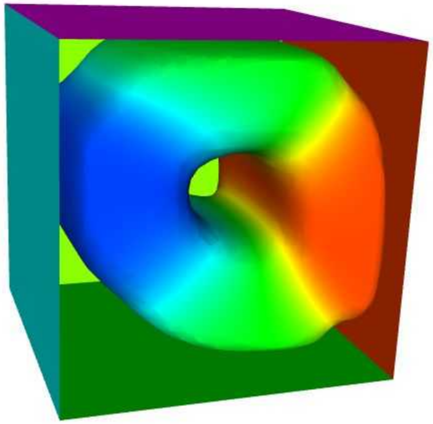
$$Pr = 0.71$$

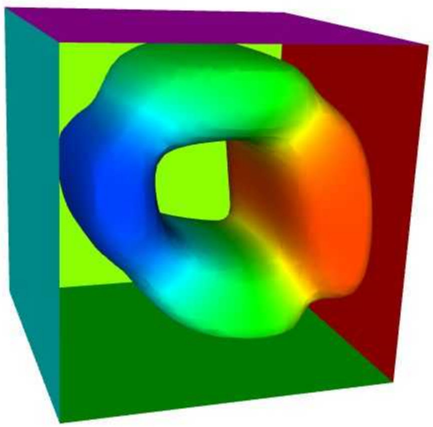
$$Ra = \text{Continuation parameter}$$

Natural Convection in a Cube with Oscillatory Heating: Continuation Run









Sandia National Laboratories

Natural Convection in a Cube with Oscillatory Heating: Successfully Continued up to Ra=10^6

{64 x 64 x 64} x 32 grid, with 5 PDEs = 43M unknowns,
(with 270 nonzeroes/row)

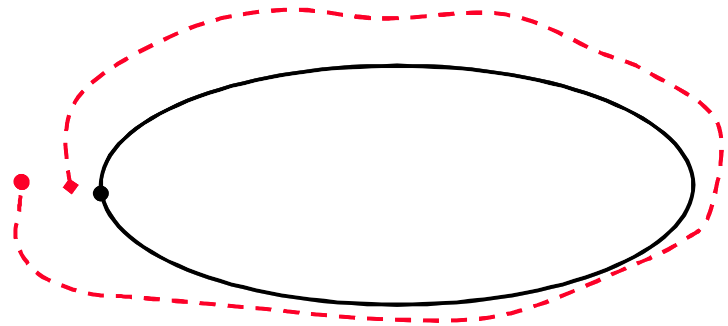
```
--------------XYZT Partition Info----------------
        NumProcs                    = 256
        Spatial Decomposition       = 8
        Number of Time Domains      = 32
        Time Steps on Domain 0      = 1
        Number of Time Steps        = 32
        -->Solving for a Periodic Orbit!
-----------------------------------------------
```

Preconditioner calc (ILU[1,0]): 5-8 mins
GMRES: 250-400 iters, 10-15 mins
Newton iters/step: 4-8
Arnoldi iterations/eigensolve 15-20
Time/Arnoldi iteration, 3-4 mins
Total Time per continuation step: 2-5 hours

Floquet Stability Analysis of Periodic Orbits

Floquet Multipliers:

- Linearize around a periodic solution
- Integrate perturbations through 1 period
- Eigenvalues of this operator are called Floquet Multipliers: σ_i
- Orbits are stable if, forall i: $\|\sigma_i\| < 1$



Krylov iteration for Anasazi eigensolver (Thornquist, Baker, Lehoucq):

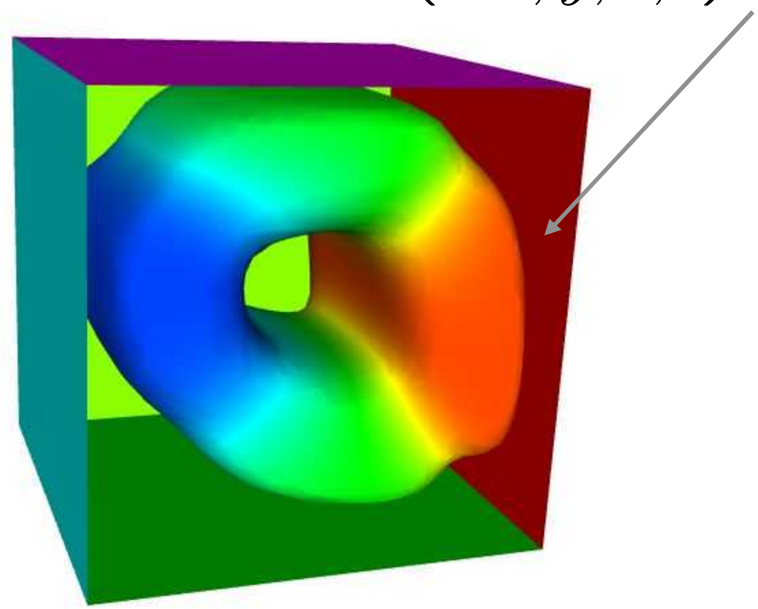
$$\begin{vmatrix} (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 & 0 & 0 \\ -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 & 0 \\ 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 & 0 \\ 0 & 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) & 0 \\ 0 & 0 & 0 & -\mathbf{B} & (\mathbf{B}-\Delta t\mathbf{J}) \end{vmatrix} \begin{vmatrix} \ldots \\ \ldots \\ \ldots \\ \ldots \\ \mathbf{v}_{i+1} \end{vmatrix} = \begin{vmatrix} \mathbf{B}\mathbf{v}_i \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

Sandia National Laboratories

Is the Periodic Orbit Stable at Ra=8.8x10^5?

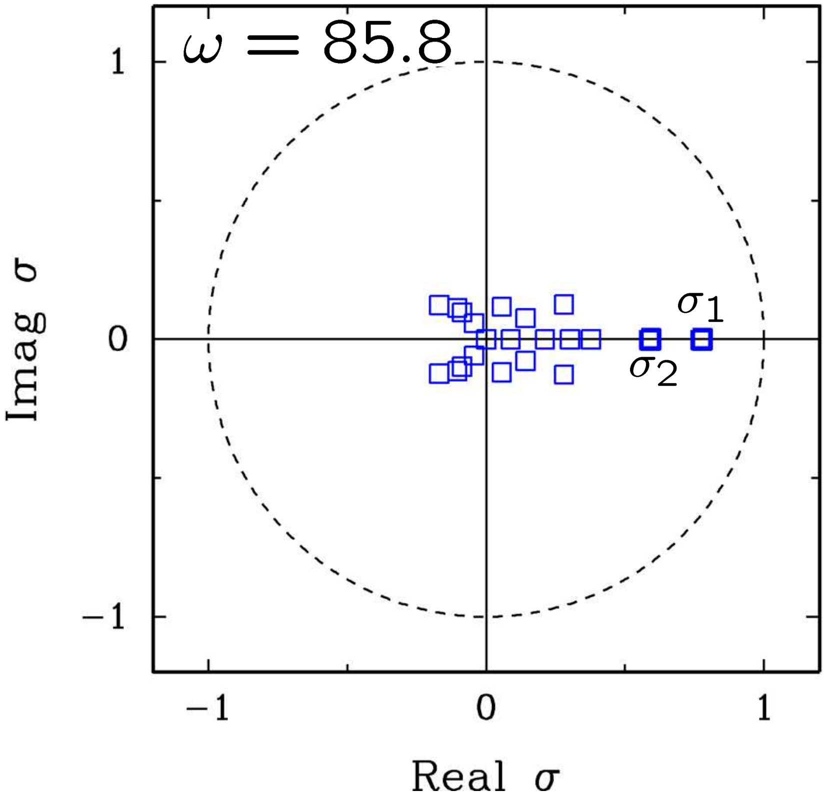
$$\sigma_1 = 2.3 \times 10^{-10}$$

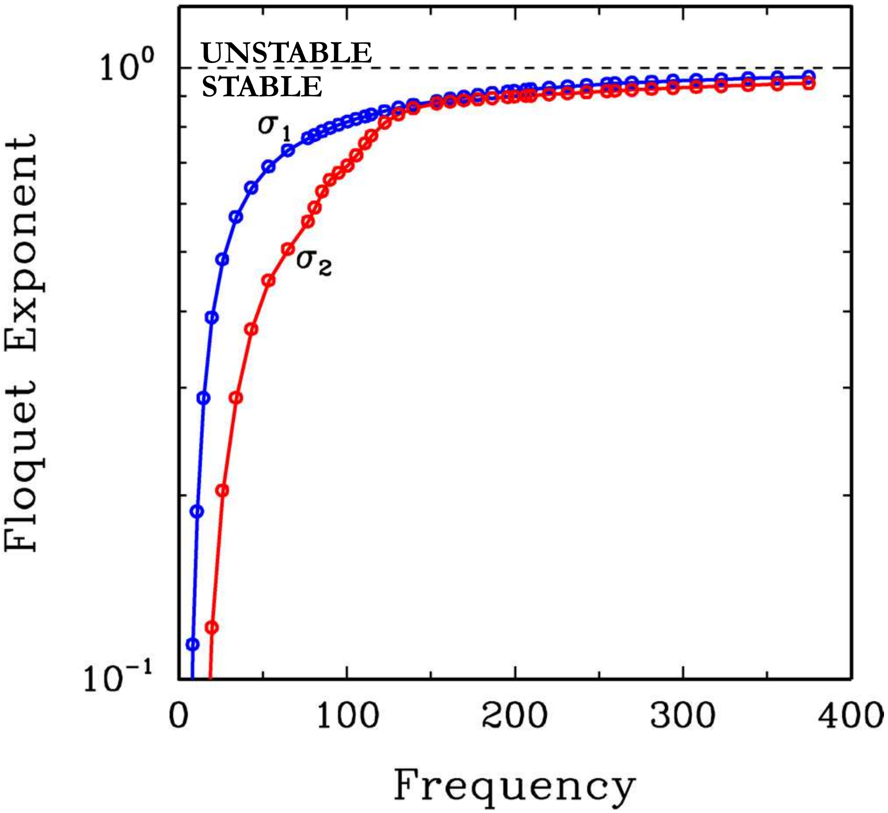
$$T(1.0, y, z, t) = sin(2\pi\omega t)$$



Forcing Frequency is additional dimensionless parameter

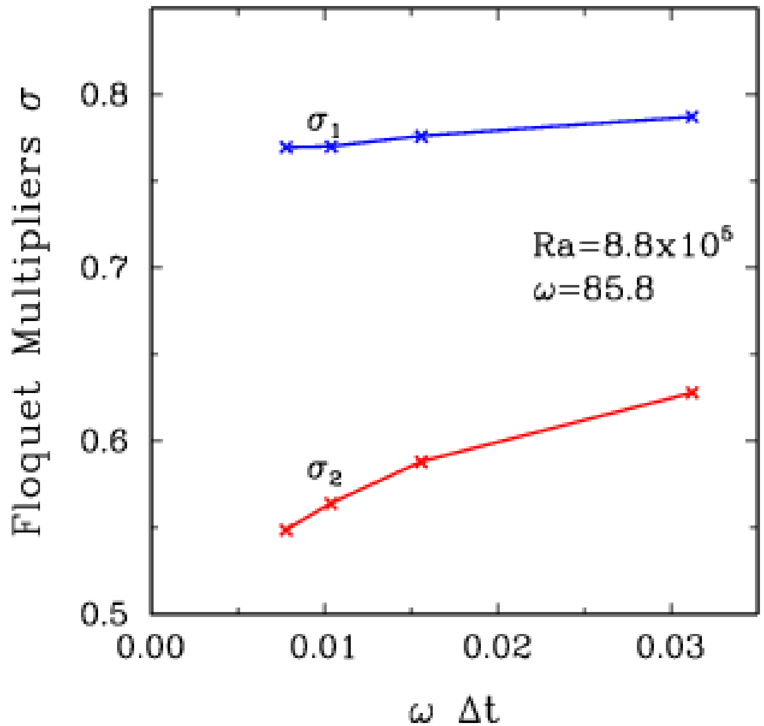
Continuation run in frequency ω





Ra=8.8x10^5

Convergence of Floquet Multipliers with Time Discretization



Time Steps	σ_1	σ_2	Total Unks	Procs
32	0.7876	0.6283	43M	256
64	0.7765	0.5884	87M	512
96	0.7705	0.5645	131M	768
128	0.7700	0.5491	175M	1024

Linear Solve Time Disparity: 175M Unknowns (128x1.37M) on 1024 Procs

$$\begin{vmatrix} (B-\Delta t J) & 0 & 0 & 0 & -B \\ -B & (B-\Delta t J) & 0 & 0 & 0 \\ 0 & -B & (B-\Delta t J) & 0 & 0 \\ 0 & 0 & -B & (B-\Delta t J) & 0 \\ 0 & 0 & 0 & -B & (B-\Delta t J) \end{vmatrix} \begin{vmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \\ \Delta x_4 \\ \Delta x_5 \end{vmatrix} = \begin{vmatrix} -g_1 \\ -g_2 \\ -g_3 \\ -g_4 \\ -g_5 \end{vmatrix}$$

540 GMRES iterations, 0.1-0.01 drop, 2800 Sec

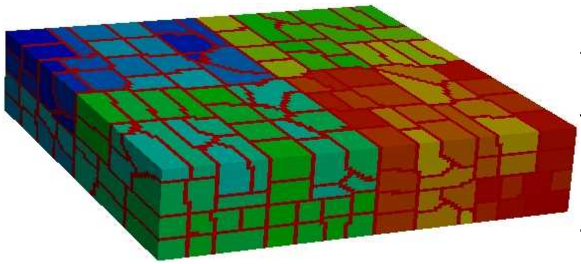
$$\begin{vmatrix} (B-\Delta t J) & 0 & 0 & 0 & 0 \\ -B & (B-\Delta t J) & 0 & 0 & 0 \\ 0 & -B & (B-\Delta t J) & 0 & 0 \\ 0 & 0 & -B & (B-\Delta t J) & 0 \\ 0 & 0 & 0 & -B & (B-\Delta t J) \end{vmatrix} \begin{vmatrix} \dots \\ \dots \\ \dots \\ \dots \\ v_{i+1} \end{vmatrix} = \begin{vmatrix} Bv_i \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

200 GMRES iterations, 0.0001 drop , 440 Sec

Future Work

- Finish verification of new Floquet analysis capability
  - Search for period-doubling; bifurcations to a torus
- Add higher-order discretizations for time domain, such as midpoint rule, Implicit RK
- Work on scalability to 100M+ unknown problems on 1000+ processors
  - Preconditioning: Multilevel, block based

MPSalsa used as flow code; All solver algorithms in Trilinos



$$\frac{\partial v}{\partial t} + (v \cdot \nabla)v = -\nabla P + \frac{1}{Pr}\nabla^2 v + RaTe_z$$

$$\nabla \cdot v = 0$$

$$\frac{\partial T}{\partial t} + (v \cdot \nabla)T = \nabla^2 T$$

$$Pr = 0.71$$

MPSalsa (Shadid *et al.*, SNL):

- Incompressible Navier-Stokes
- Heat and Mass Transfer, Reactions
- Unstructured Finite Element (TriLinear with PSPG stabilization)
- Distributed Memory Parallelism
- Analytic, Sparse Jacobian

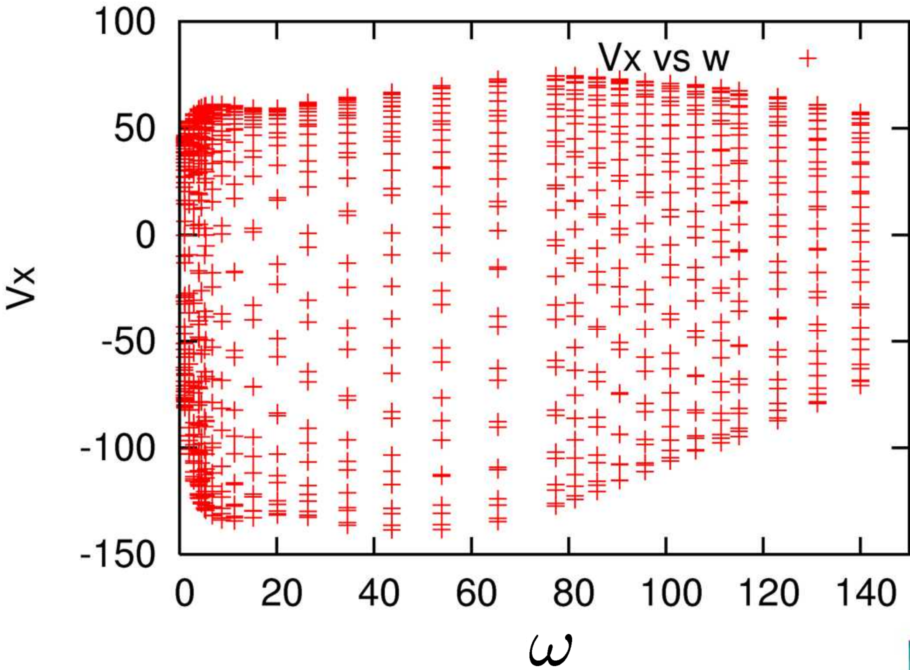
Trilinos Packages used in MPSalsa:

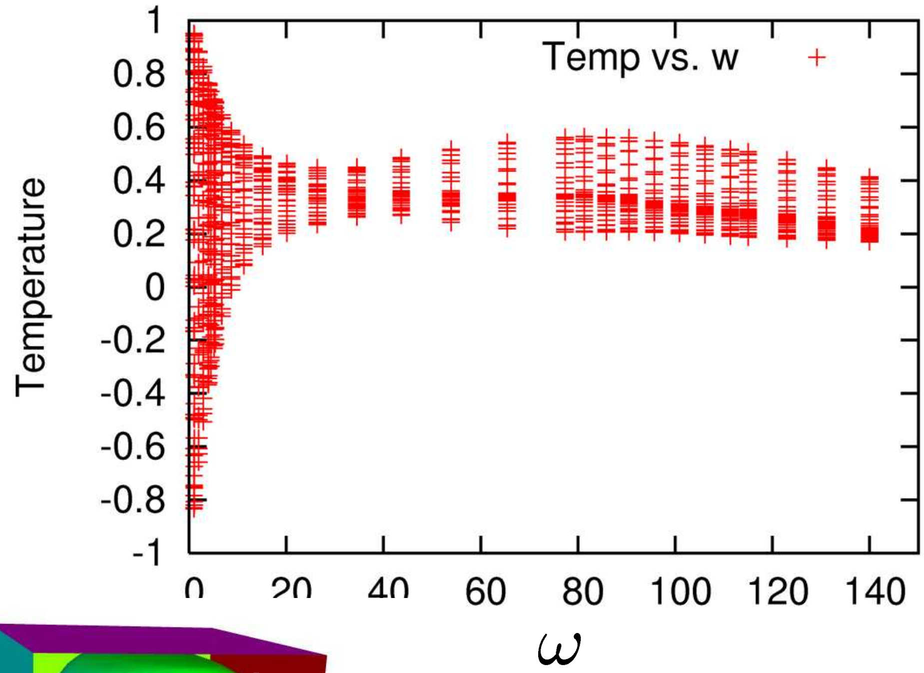
- Aztec and Ifpack, GMRES with ILU Domain Decomp Preconditioners
- NOX Nonlinear Solver
- LOCA Continuation/Bifurcation
- Anasazi Eigensolver
- Epetra sparse matrix structures
- Meros block preconditioner
- ML multilevel solver
- Moocho PDE-Const. Optimization

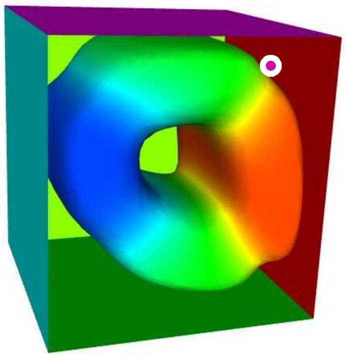
Continuation run in frequency ω

$$T(1.0, y, z, t) = sin(2\pi\omega t)$$

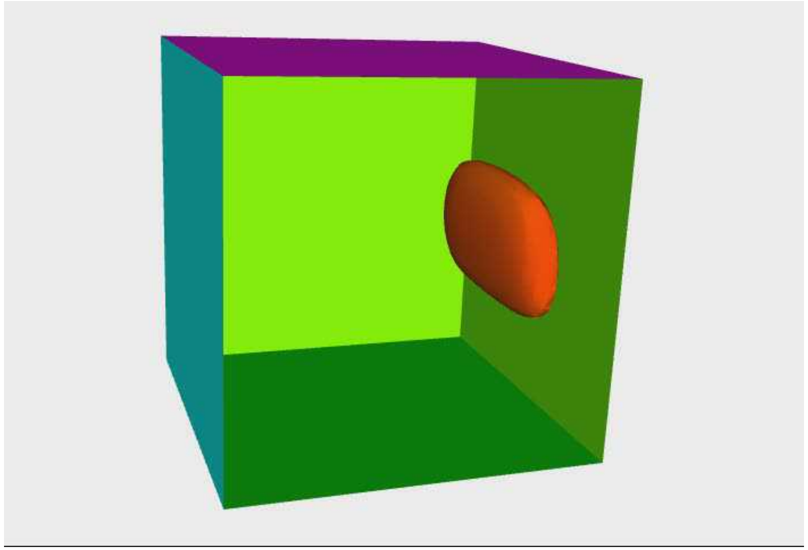
Ra=8.8x10^5



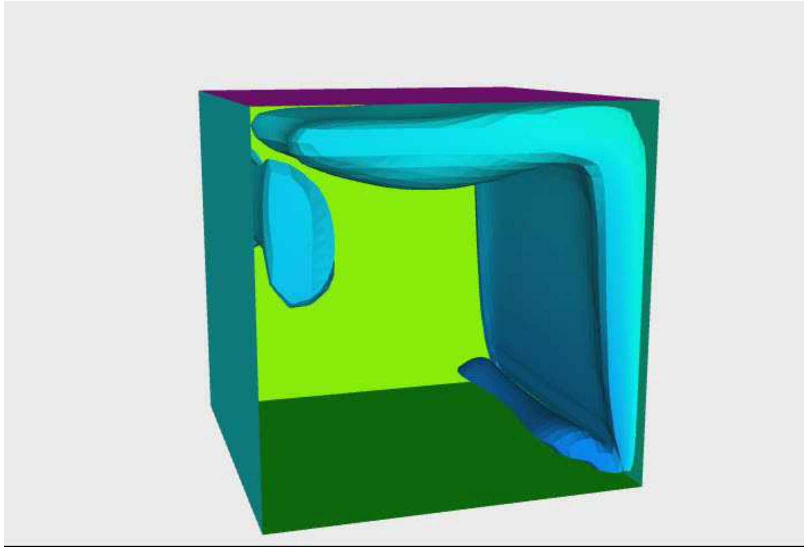




Movies



$Ra=3x10^3$



$Ra=2x10^6$

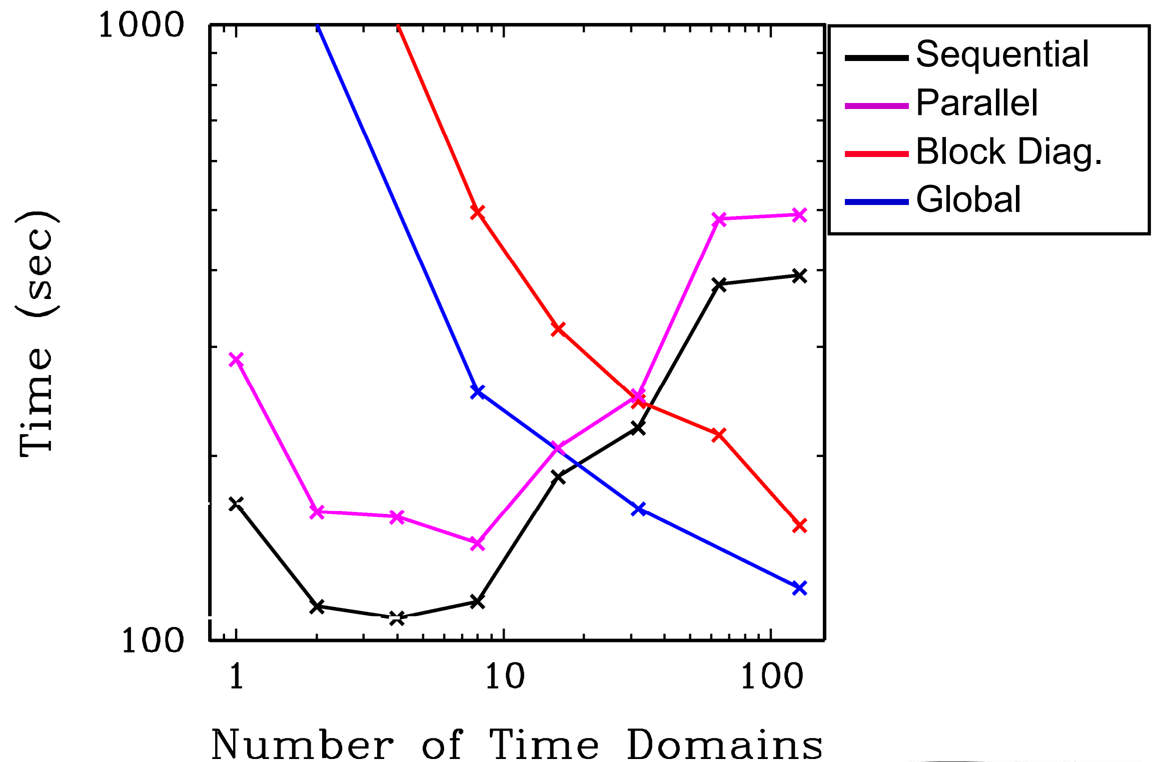
Numerical Experiment

Can space-time parallelism be more effective than just spatial parallelism?

- **Fixed Number of Processors (128)**
  - **Spatial domains: 1 2 4 8 16 32 64 128**
  - **Time domains: 128 64 32 16 8 4 2 1**

MPSalsa:
PDEs: 5
FEM: 2D, 64 x 48 elements
Time steps: 128
Unknowns: 2,038,400

Trilinos:
Newton (NOX) : 4–6 iterations
GMRES (Aztec) : <200 iters
ILUk (Ifpack) : overlap=0, fill=1
Continuation steps (LOCA): 1



Numerical Experiment #1

How much can parallelism in time speed up the solve?

- **Fixed Number of Spatial Domains (4)**
  - **Processors: 4 8 16 32 64 128**
  - **Time Domains: 1 2 4 8 16 32**

MPSalsa:
PDEs: 5
FEM: 2D, 64 x 48 elements
Time steps: 32
Unknowns: 509,600

Trilinos:
Newton (NOX) : 3-6 iterations
GMRES (Aztec) : <200 iters
ILUk (Ifpack) : overlap=0, fill=1
Continuation steps (LOCA): 1