SAND2006-4454C

National Nuclear Security Administration

Nonlinear Solution Techniques for the Analysis of Large-Scale Reaction/Transport Systems

Roger Pawlowski, Andrew Salinger, and John Shadid

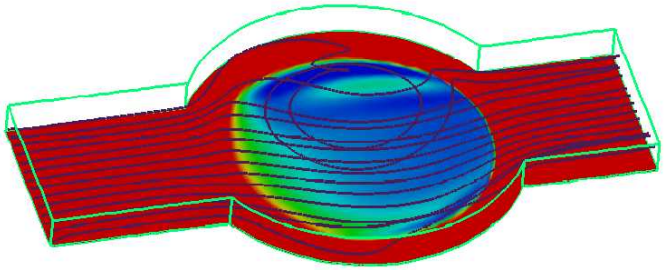
Sandia National Laboratories
Computer Science and Mathematics Group

Tuesday August 1st, 2006

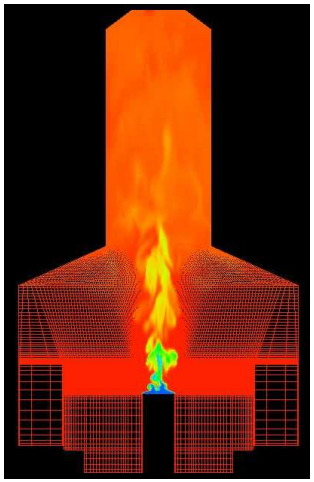
Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

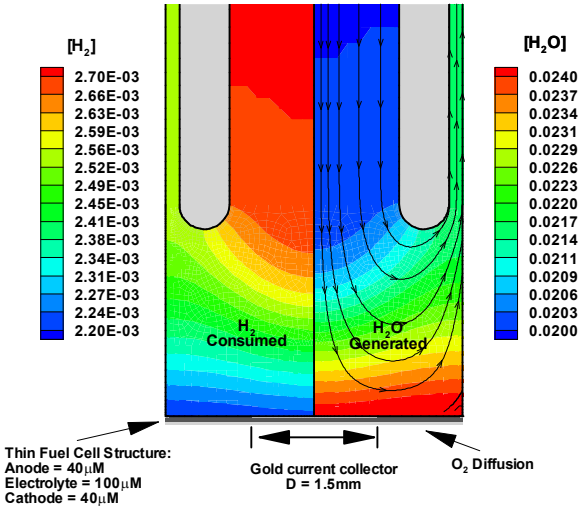
Transport/Reaction Simulation is Critical to Sandia's Mission



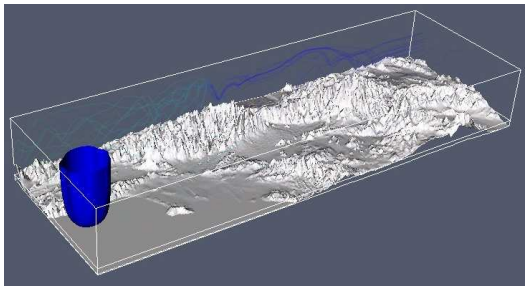
Chemical Reactor Design



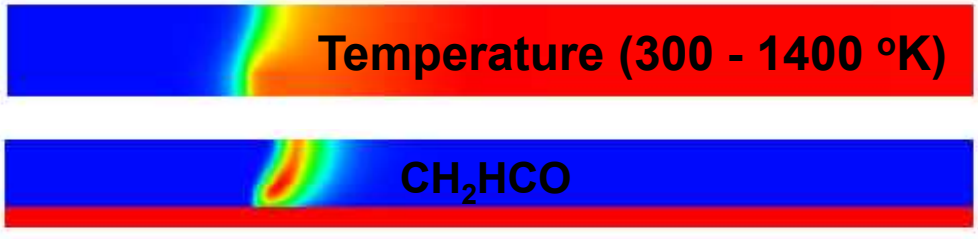
Combustion



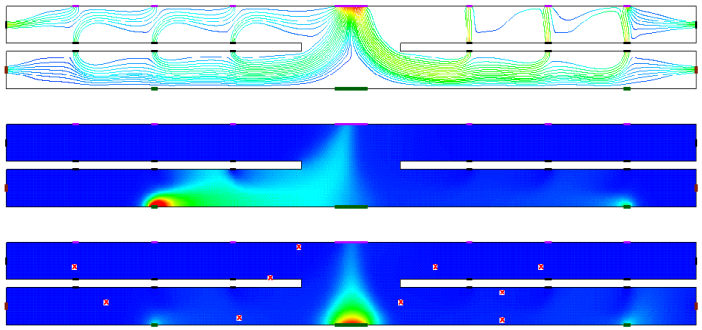
H2 Fuel Cells



Contamination Events for External Flows



Partial Catalytic Oxidation



Chem/Bio Atack on Airport

Outline

- PDE Equations and Discretization
  - Incompressible N-S, Energy, and Mass Transfer
  - Stabilized Galerkin Finite Elements
- Linear Solver Overview
  - Preconditioning
- Nonlinear Solution Techniques for Robust Simulation
  - Line Search
  - Trust Region
  - Homotopy/Continuation
- Analysis of CVD Reactors
  - Bifurcation and Stability
  - Optimization

Computational Simulation of Transport/Reaction Systems

Requirements:

- Complex Geometry
- 2D/3D Transport
  - Variable density / subsonic compressible flow
  - Multi-component diffusion / thermal diffusion
  - Full property variation $f(T,Y_i)$
- Non-equilibrium Chemistry
  - Gas phase and surface reactions
- Steady State & Transient Solution
- Mapping of Complex Solution Spaces
- Bifurcation and Stability Analysis
- Design Optimization

MP Supercomputers:

3D Unstructured FE,

Fully-Implicit Methods,

Direct-to-Steady-State,

Robust and Scalable

Iterative Methods,

h - adaptivity,

Dynamic Load Balancing

Sandia National Laboratories

Equations and Discretization

Sandia National Laboratories

Incompressible Navier-Stokes, Energy and Mass Transport

Equations

Continuity $$\nabla \cdot \mathbf{u} = 0$$

Momentum $$\rho \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla \cdot \sigma - \rho g = 0$$

Energy $$\rho C_p \frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T + \nabla \cdot q - \sum_{i=1}^{K} h_k \dot{\omega}_k W_k = 0$$

Mass $$\rho \frac{\partial y_k}{\partial t} + \rho(\mathbf{v} \cdot \nabla) y_k + \nabla \cdot \mathbf{j}_k + \dot{\omega}_k W_k$$

Constitutive Models

Stress Tensor $$\sigma = -P\mathbf{I} - \frac{2}{3}\mu(\nabla \cdot \mathbf{u})\mathbf{I} + \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^{\mathrm{T}})$$

Energy Flux $$q = -k \nabla T$$

Mass Flux $$\mathbf{j}_k = \rho y_k \frac{1}{x_k \bar{W}} \sum_{j=1}^{K} W_j D_{kj} \nabla x_j$$

Sandia National Laboratories

Stabilized Finite Elements
(Hughes and coworkers)

The choice of discretization is critical in achieving efficient, robust, and accurate prediction.

Continuity Equation: $\nabla \cdot (\rho \mathbf{u}) = 0$

Residual Form: $\mathbf{R}_p = \nabla \cdot (\rho \mathbf{u})$

Stabilized FE Variational Formulation:

$$F_p = \underbrace{\int_\Omega \Phi \mathbf{R}_p d\Omega}_{\text{Galerkin Term (Eqns. of interest)}} + \underbrace{\sum_e \int_{\Omega_e} \rho \tau_m \nabla \Phi \cdot \mathbf{R}_m d\Omega}_{\text{PSPG Term}}$$

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{N} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\mathbf{P}} \end{bmatrix} + \begin{bmatrix} \mathbf{A} & -\mathbf{B}^T \\ \mathbf{BR} & \mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{P} \end{bmatrix}$$

Where $\mathbf{K} = \sum_e \int_{\Omega^e} \rho \tau_m \nabla \Phi \bullet \nabla \Phi d\Omega$

Sandia National Laboratories

Stabilized Finite Elements

Momentum Equation: $$\rho\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla\cdot\sigma - \rho g = 0$$

Residual Form: $$\mathbf{R}_m = \rho\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla\cdot\tau + \nabla P - \rho g$$

Stabilized FE Variational Formulation:

$$F_{m,i} = \underbrace{\int_\Omega \Phi \mathbf{R}_{m,i} d\Omega}_{\text{Galerkin Term (Eqns. of interest)}} + \underbrace{\sum_e \int_{\Omega_e} \rho\tau_m(\mathbf{u}\cdot\nabla\Phi)\mathbf{R}_{m,i} d\Omega}_{\text{SUPG Term (convection dominated)}}$$

$$+ \underbrace{\sum_e \int_{\Omega_e} \nu((R_{m,i})\nabla\Phi\cdot\mathbf{G}^{\mathbf{c}}\nabla\mathbf{u} d\Omega}_{\text{DCTerm (Shocks)}}$$

Sandia National Laboratories

Nonlinear Problem

Find x_* such that $F(x_*) = 0$ where $F : \Re^n \rightarrow \Re^n$

$$F(x) = \begin{Bmatrix} F_{m_x} \\ F_{m_y} \\ F_{m_z} \\ F_p \\ F_{y_1} \\ F_{y_2} \\ \vdots \\ F_{y_N} \end{Bmatrix}$$

We define the Jacobian as:

$$F'(x) \in \Re^{m \times n}$$

$$F'(x)_{ij} = J(x)_{IJ} = \frac{\partial F_i}{\partial x_j}$$

Newtons Method

Until converged:

$$J(x_i)\Delta x = -F(x_i)$$
$$x = x_i + \Delta x$$

Sandia National Laboratories

Newton-Krylov Methods

Issues:
Very Large Problems -> Iterative Solution of Sub-problems
Krylov Methods - Robust, Scalable and Efficient Preconditioners

Iterative Linear Solver – GMRES

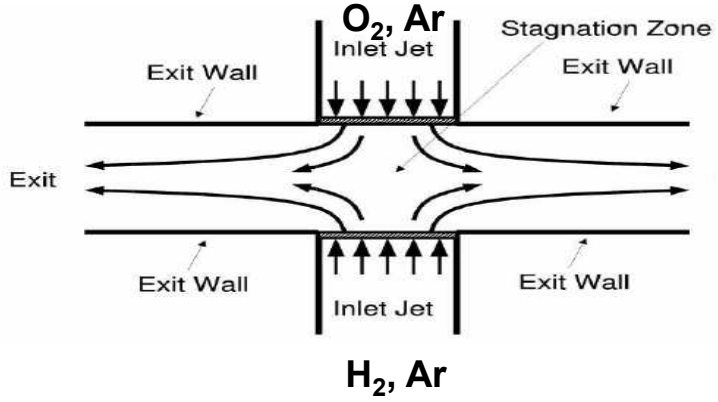
Krylov Subspace of the form:

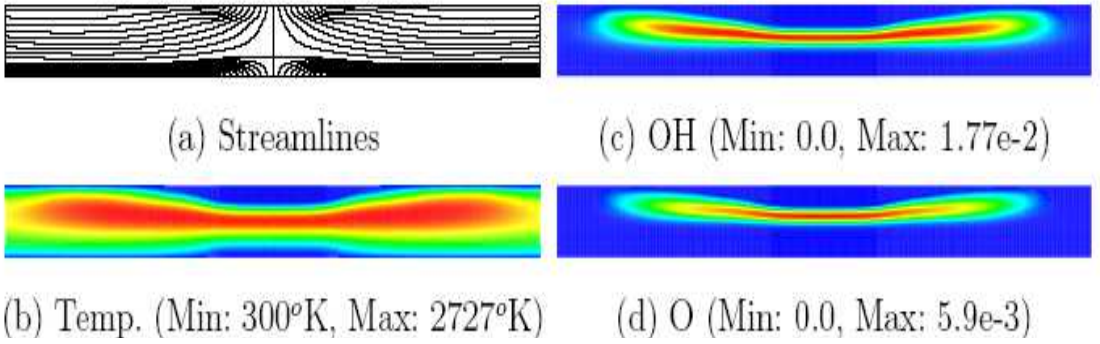
$$\mathcal{K}(A, v) \equiv \text{span}\{v, Av, A^2v, ..., A^{m-1}v\}$$

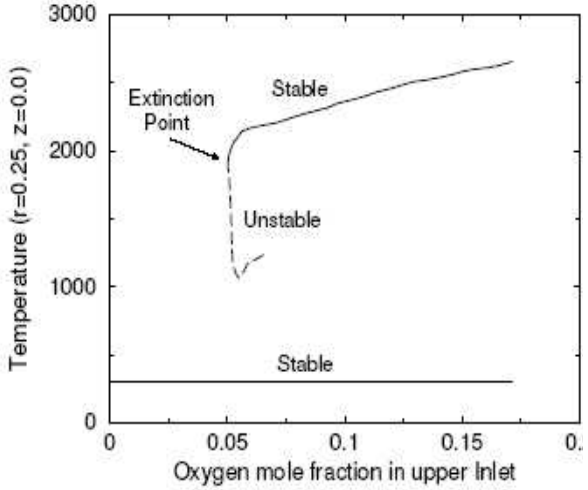
Preconditioning

Right preconditioning
Domain Decomposition
MultiLevel (algebraic and geometric)

Parallel Scaled Efficiency of 1- level DD Preconditioners: Steady Reacting H_2, O_2 Opposed Jet Reactor





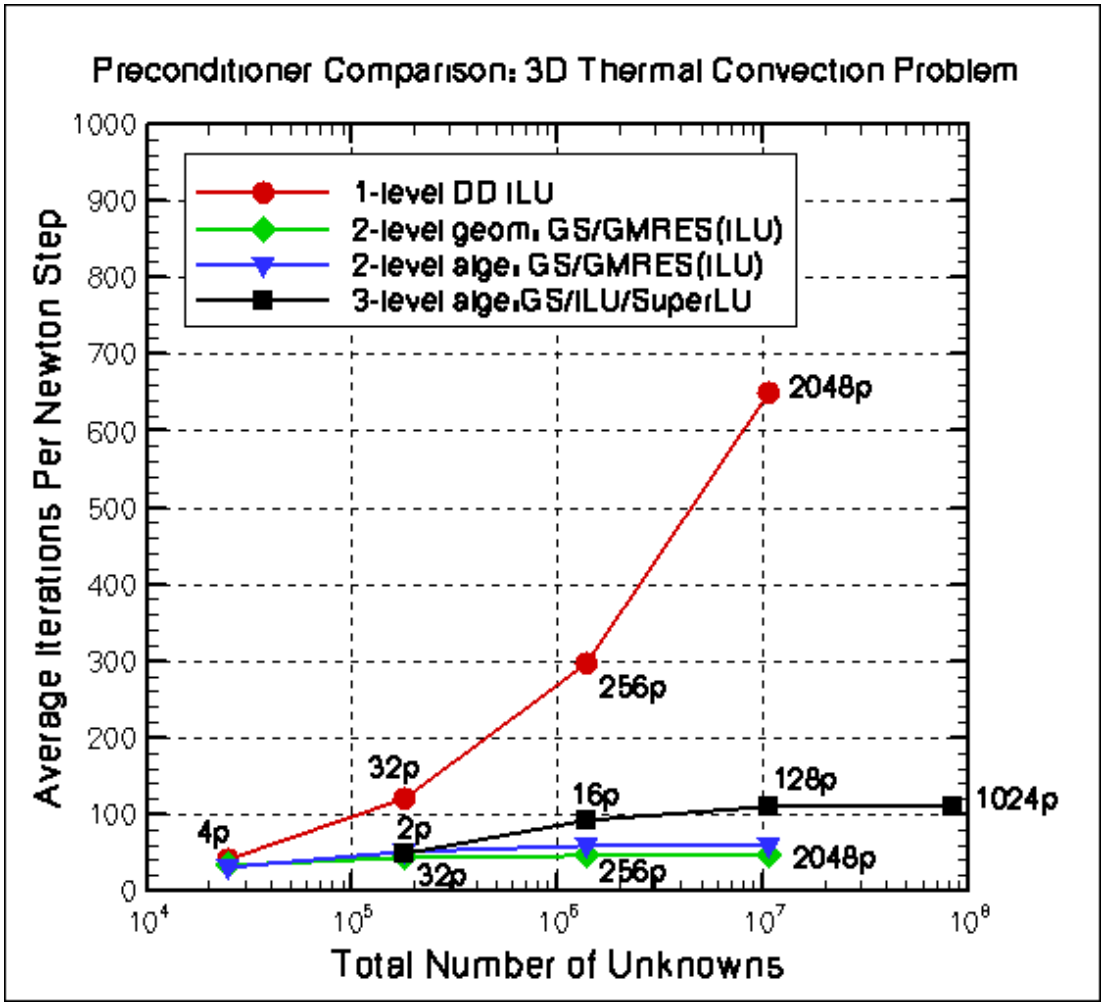


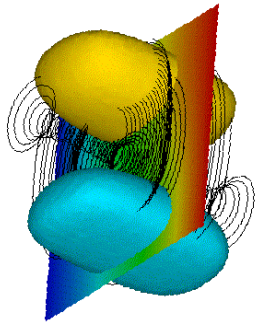
Num Procs	rf	Num. unknowns	avg time /matrix fill (sec)	scaled eff.	avg time /linear iter (sec)	scaled eff.
16	0	199,374	13.94	—	0.5219	—
64	1	790,734	13.86	1.01	0.5286	0.99
256	2	3,149,454	14.06	0.99	0.5363	0.97
1024	3	12,570,894	13.99	1.00	0.5369	0.97

Table 8
Scaled Efficiency of hydrogen/oxygen diffusion flame simulation with DD-ILU TFQMR. **(10 species, 19 reactions)**

Sandia National Laboratories

Multilevel Preconditioner Scaling Study: 3D Thermal Buoyancy Driven Convection





Sandia National Laboratories

Robust Nonlinear Solution Techniques

Sandia National Laboratories

Nonlinear Solution Algorithms

Tensor Method

$$M_T = f(x_c) + J_c d + \frac{1}{2} T_c dd$$

Newton's Method

$$M_N = f(x_c) + J_c d$$

Broyden's Method

$$M_B = f(x_c) + B_c d$$

Globalizations

Line Search
Interval Halving
Quadratic
Cubic
More'-Thuente

Trust Region
Dogleg
Inexact Dogleg

Homotopy
Artificial Parameter Continuation
Natural Parameter Continuation

Iterative Linear Solvers, Adaptive Forcing Terms

Globalized Methods

$F(x^*) = 0 \Rightarrow \min f(x) = \tfrac{1}{2}\,\|F(x)\|^2$

Newton's Method

$$M_k(s) = F(x_k) + J(x_k)s_k$$

$$J(x_k)s_k = -F(x_k)$$

$$x_{k+1} = x_k + s_k$$

Line Search

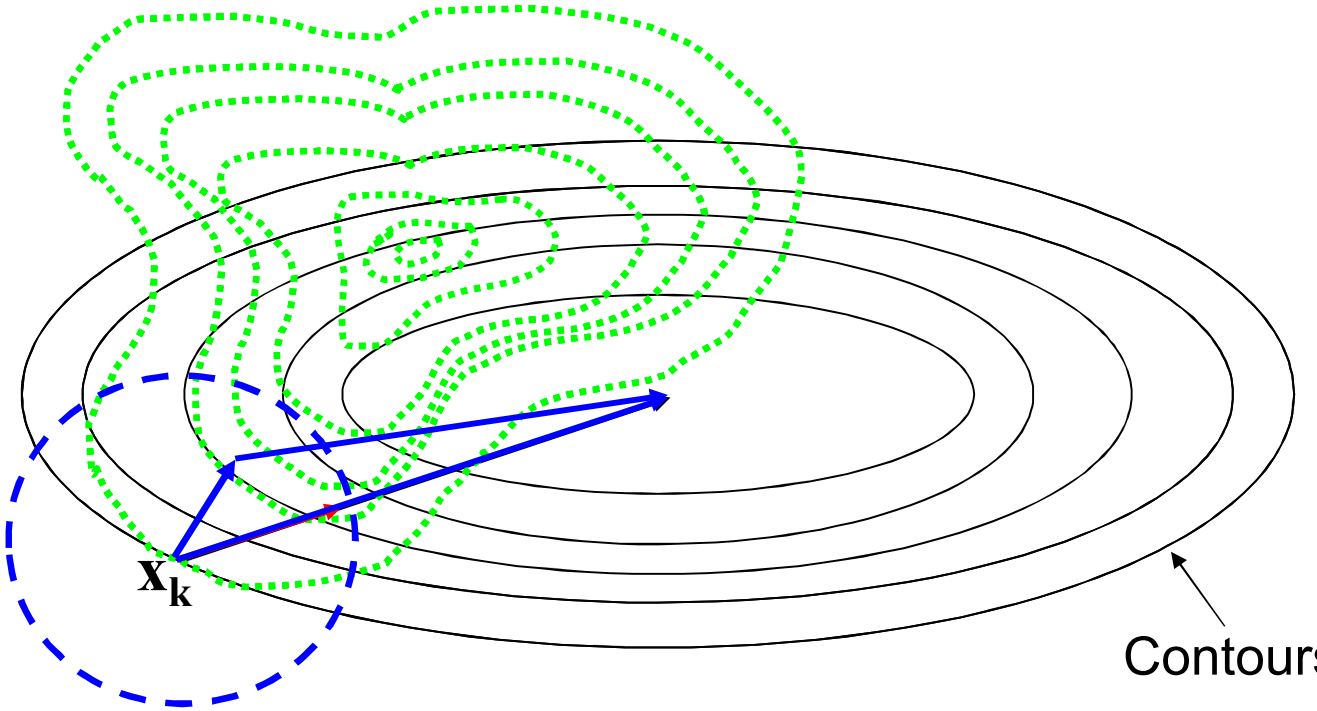
1. Compute direction, s_k
2. Compute distance, λ_k

$$x_{k+1} = x_k + \lambda_k s_k$$

Trust Region

1. Compute distance, δ_k
2. Compute direction

$$x_{k+1} = x_k + s_k$$



Sandia National Laboratories

Line Search Algorithms

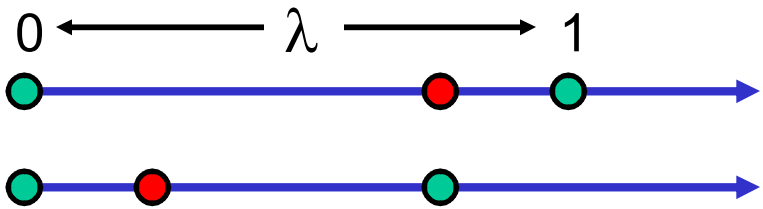
1. Compute a descent direction
2. Compute a distance

While (not converged) k=0, 1, …

Compute s_k

While (LS Criteria not conv.) i=0, 1, …

Compute λ_i via interpolation



"Sufficient Decrease"

$$f(x_k + \lambda_k s_k) \leq f(x_k) + \alpha \lambda_k \nabla f_k^T s_k$$

"Curvature"

$$\nabla f(x_k + \lambda_k s_k)^T s_k \geq \beta \nabla f(x_k)^T s_k$$

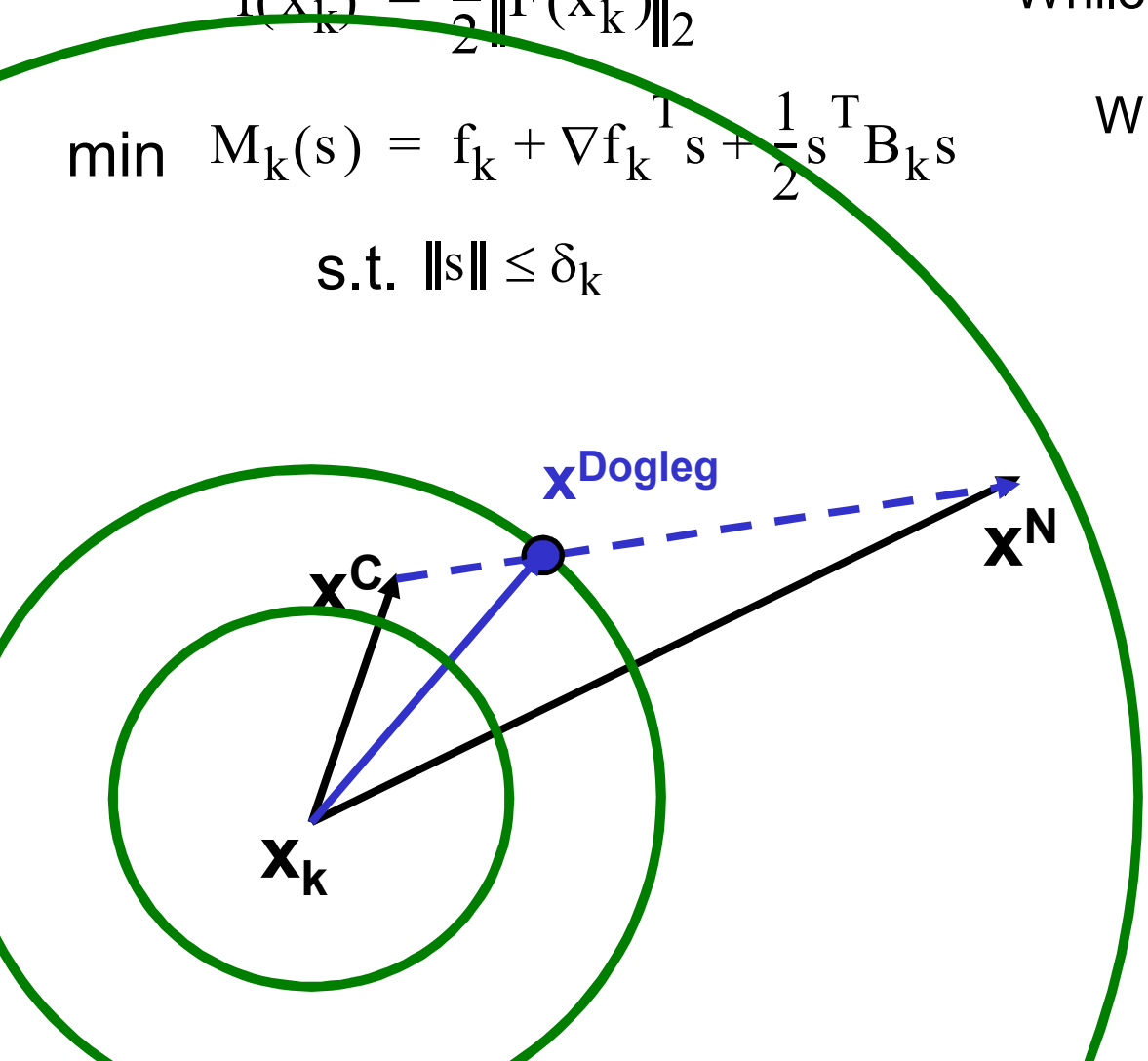
Sandia National Laboratories

Trust Region Algorithm

$$f(x_k) = \frac{1}{2}\|F(x_k)\|_2^2$$

$$\min \; M_k(s) = f_k + \nabla f_k^T s + \frac{1}{2} s^T B_k s$$

$$\text{s.t. } \|s\| \le \delta_k$$



While (not converged) k=0,1, …

While $\rho_k < \rho_{min}$ and $\delta > \delta_{min}$

Calculate s_k based on δ_k

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{M_k(0) - M_k(s_k)}$$

Adjust δ_k

Parameters
Contraction Trigger
Expansion Trigger
Contraction Factor
Expansion Factor
Ratio min/max
Radius min/max

Robustness Study on MPSalsa Benchmarks

Number of failures for 2D test problems

Method	Forcing Term	2D Thermal Convection		2D Lid Driven Cavity		2D Backward Facing Step	
		Easy	Difficult	Easy	Difficult	Easy	Difficult
Full Step	constant	0	1	5	5	3	4
	adaptive	0	1	4	5	1	4
Quadratic Line Search	constant	0	0	4	5	1	1
	adaptive	0	0	0	0	0	0
Cubic Line Search	constant	0	0	4	5	0	0
	adaptive	0	0	0	0	1	1
More'-Thuente Line Search	constant	0	0	4	5	0	0
	adaptive	0	0	0	0	1	1
Dogleg Trust Region	constant	0	1	4	5	0	0
	adaptive	0	1	1	4	0	0

- All globalizations of interest improve robustness.
- All globalizations are equally effective.
- Adaptive forcing terms improve robustness dramatically (resolving less, but more important modes in GMRES loop)!
- Overall, the combination of adaptive forcing term and globalization is very effective, however no combination succeeded in every case.

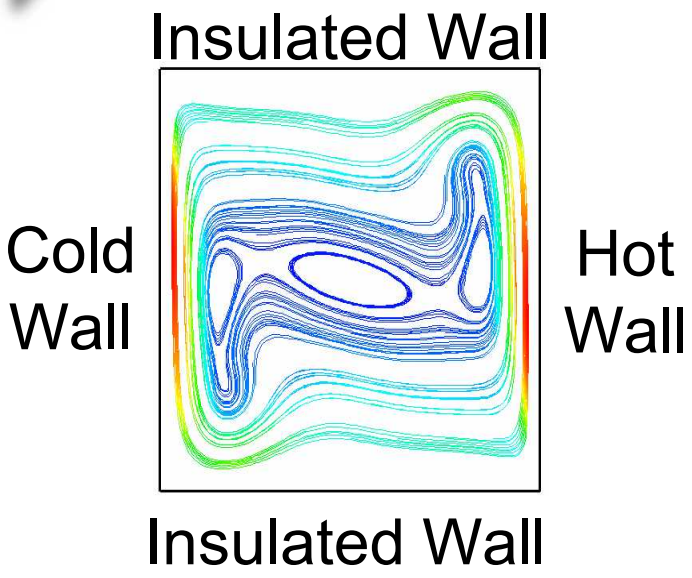
Efficiency

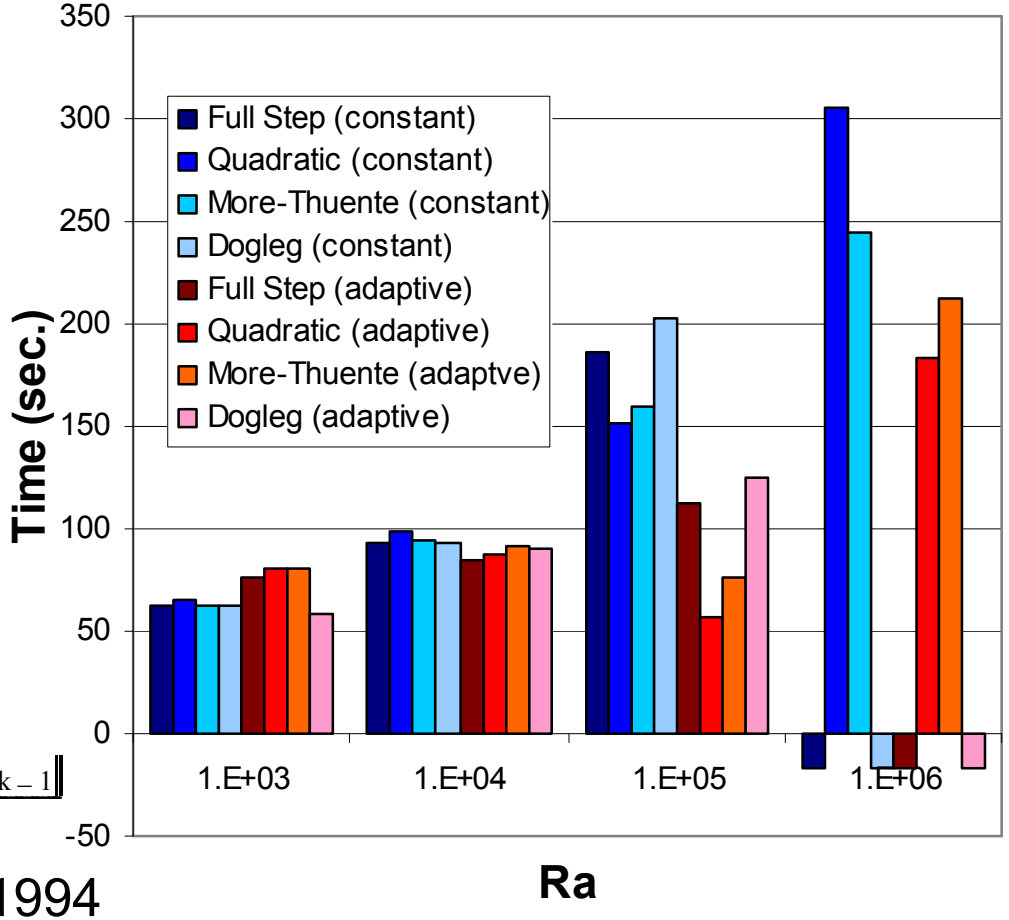
Method	Forcing Term	Inexact Newton Steps	Backtracks per INS	GMRES iterations per INS	Normalized Time
Quadratic Line Search	constant	8.90	0.17	126.48	1.00
	adaptive	16.42	0.14	46.49	0.79
Cubic Line Search	constant	8.86	0.18	126.44	1.00
	adaptive	16.37	0.14	46.26	0.78
More'-Thuente Line Search	constant	8.43	0.18	126.14	0.93
	adaptive	15.25	0.17	50.70	0.93
Dogleg Trust Region	constant	9.99	0.29	131.09	1.11
	adaptive	14.04	0.19	63.00	0.80

- For a particular choice in forcing term, most globalizations performed similarly.
- Adaptive forcing terms reduced the mean # of GMRES iterations (reduced oversolving).
- Small constant forcing terms required fewer inexact Newton steps (achieved greater residual reduction).
- Adaptive forcing terms yielded better run times.

Adaptive Forcing Terms

(MPSalsa - 2D Thermal Convection)





- Constant: $\eta_k = 1.0e\text{-}4$
- Adaptive: $\eta_k = \dfrac{\left\|F(x_k)\right\| - \left\|F(x_{k-1}) + F'(x_{k-1})s_{k-1}\right\|}{\left\|F(x_{k-1})\right\|}$

"Type 1" – Eisenstat and Walker 1994

In General:

- **Adaptive forcing terms are less efficient for EASY problems.**
- **Adaptive forcing terms are more efficient for DIFFICULT problems.**

Numerical Issues

Inexact Newton with adaptive forcing:

$$\eta_k = \frac{\|F(x_k)\| - \|F(x_{k-1}) + J(x_{k-1})s_{k-1}\|}{\|F(x_{k-1})\|}$$

$$\|F(x_k) + J(x_k)s_k\| \leq \eta_k \|F(x_k)\|$$

MPSalsa 2D Lid driven cavity:

- Precondition via right scaling
- Unknowns scaled via left scaling with Jacobian row sums.

$$\|D^{-2}F(x_k) + D^{-2}J(x_k)s_k\| \leq \eta_k \|D^{-2}F(x_k)\|$$

- Globalizations are sensitive to scaling.
  - Get a different direction out of linear solver
  - Sufficient decrease condition may not be satisfied anymore!
- Users can override the norms and merit functions used throughout NOX!
  - NOX::Parameter::Arbitrary::UserNorm
  - NOX::Parameter::Arbitrary::MeritFunction
  - NOX::Parameter::Arbitrary::PrePostIterate

L-2 Norms

Re	Quadratic	Cubic
1000	21	21
2000	30	30
3000	F	F
4000	66	66
5000	85	85
6000	F	186
7000	F	F
8000	136	136
9000	F	F
10000	179	172

Row Sum Scaled Norms

Re	Quadratic	Cubic
1000	24	24
2000	33	33
3000	52	52
4000	52	57
5000	60	58
6000	76	75
7000	101	94
8000	140	106
9000	143	150
10000	163	160

Part 3: Enabling Technology

Sandia National Laboratories

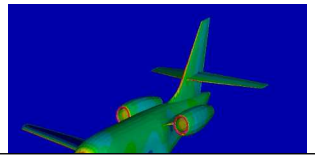
Solver Technology

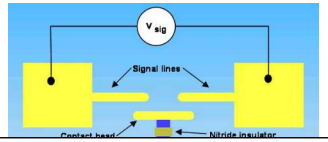
Sandia's MP Solver Suite

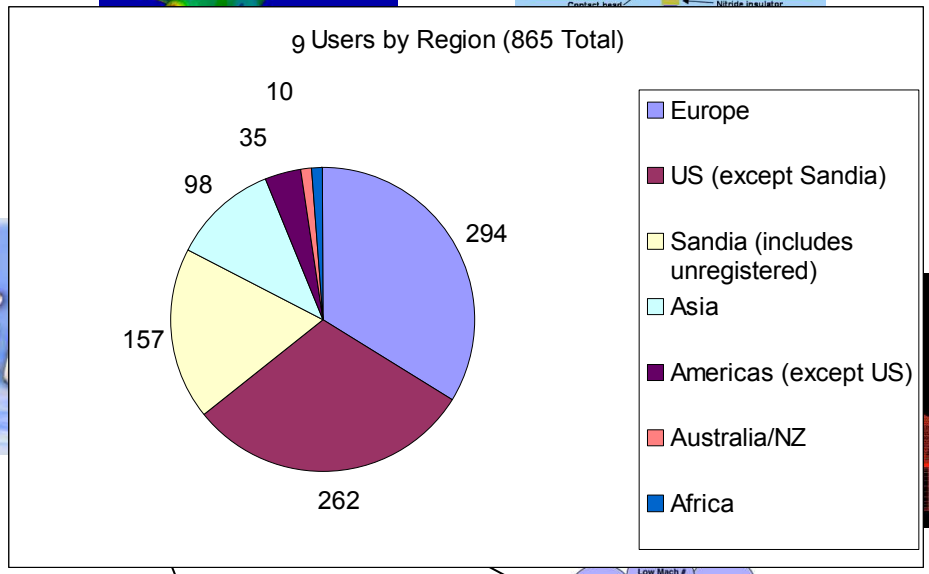
Two-level design:
- **Self-contained packages**
- **Leveraged common tools.**

Allows rapid algorithmic development and delivery

Leveraging investments in software infrastructure without compromising individual package autonomy





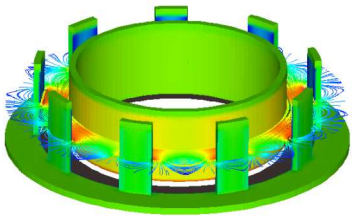


Statistics:

Trilinos 6.0 released 9/05, >1000 downloads

Focus on continued package development and software engineering

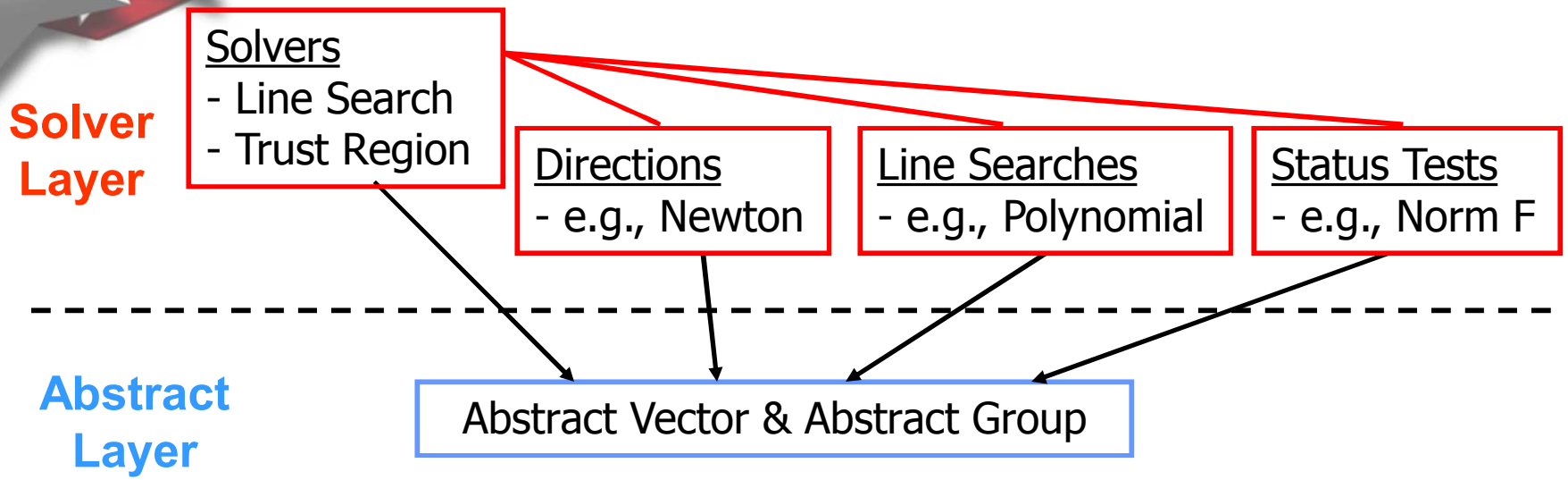
Awards
- **R&D 100 (in 2004)**
- **IEEE HPC Software Challenge Award**
- **Sandia ERA team award and LM NOVA nomination**



Full Vertical Solver Coverage

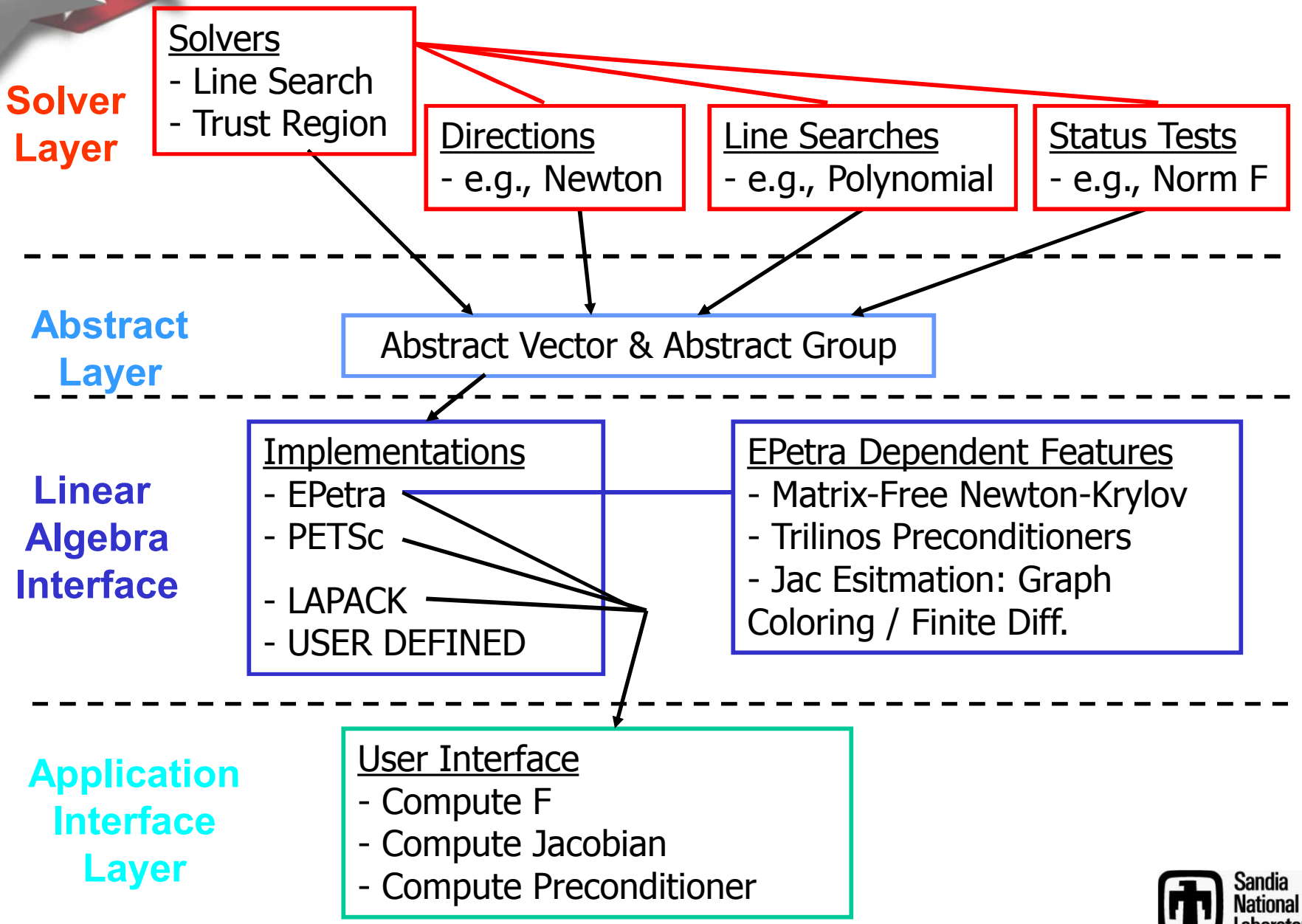
Optimization Unconstrained: Constrained:	Find $u \in \Re^n$ that minimizes $g(u)$ Find $x \in \Re^m$ and $u \in \Re^n$ that minimizes $g(x,u)$ s.t. $f(x,u) = 0$	**MOOCHO**
Bifurcation Analysis	Given nonlinear operator $F(x,u) \in \Re^{n+m} \rightarrow \Re^n$ For $F(x,u) = 0$ find space $u \in U \ni \frac{\partial F}{\partial x}$ singular	**LOCA**
Transient Problems DAEs/ODEs:	Solve $f(\dot{x}(t), x(t), t) = 0$ $t \in [0,T], x(0) = x_0, \dot{x}(0) = x_0'$ for $x(t) \in \Re^n, t \in [0,T]$	**Rhythmos**
Nonlinear Problems	Given nonlinear operator $F(x,u) \in \Re^{n+m} \rightarrow \Re^n$ Solve $F(x) = 0$ $x \in \Re^n$	**NOX**
Linear Problems Linear Equations: Eigen Problems:	Given Linear Ops (Matrices) $A, B \in \Re^{m \times n}$ Solve $Ax = b$ for $x \in \Re^n$ Solve $A\nu = \lambda B\nu$ for (all) $\nu \in \Re^n$, $\lambda \in \Re$	**AztecOO** **Belos** **Ifpack, ML, etc...** **Anasazi**
Distributed Linear Algebra Matrix/Graph Equations: Vector Problems:	Compute $y = Ax; A = A(G); A \in \Re^{m \times n}, G \in \Im^{m \times n}$ Compute $y = \alpha x + \beta w; \alpha = \langle x, y \rangle; x, y \in \Re^n$	**Epetra** **Tpetra**

NOX Interface



- Don't need to directly access the vector or matrix entries, only manipulate the objects.
- NOX uses an abstract interface to manipulate linear algebra objects.
- Isolate the Solver layer from the linear algebra implementations used by the application.
- This approach means that NOX does NOT rely on any specific linear algebra format.
- Allows the apps to tailor the linear algebra to their own needs!
  - Serial or Parallel
  - Any Storage format: User Defined, LAPACK, PETSc, Epetra

NOX Framework



Sandia National Laboratories

Bifurcation and Stability Analysis

Sandia National Laboratories

Why Do We Need Stability Analysis Algorithms for Large-Scale Applications?

Nonlinear systems exhibit instabilities, *e.g.*:

- Multiple steady states
- Ignition
- Symmetry Breaking
- Onset of Oscillations
- Phase Transitions

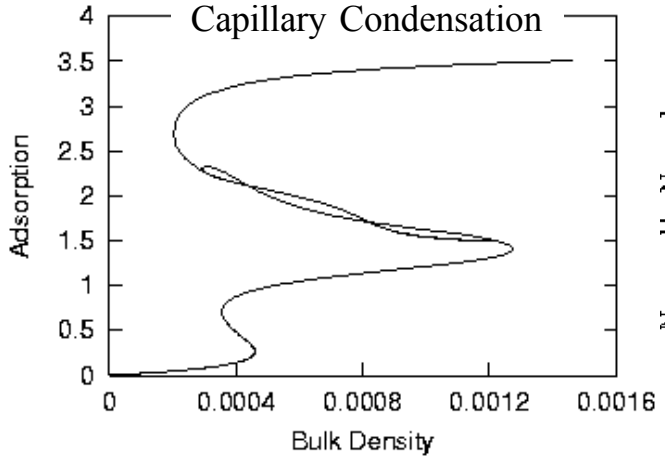
→ These phenomena must be understood in order to perform computational design and optimization.

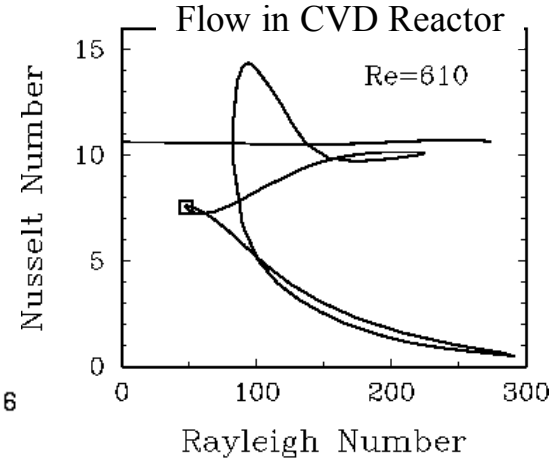
Established bifurcation analysis libraries exist:

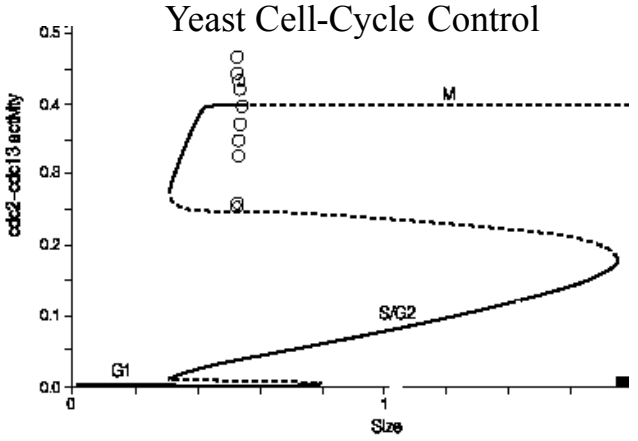
- AUTO (Doedel)
- CONTENT (Kuznetsov)

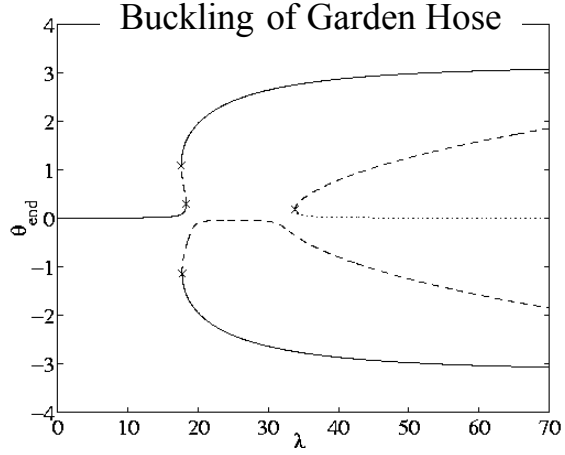
→ We need algorithms, software, and experience to impact ASCI- and SciDAC-sized applications.

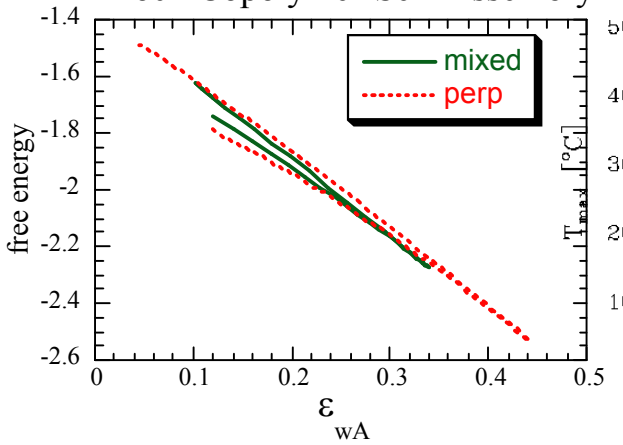
Examples of Hysteresis / Turning Point Bifurcations (Eigenvalue λ=0)

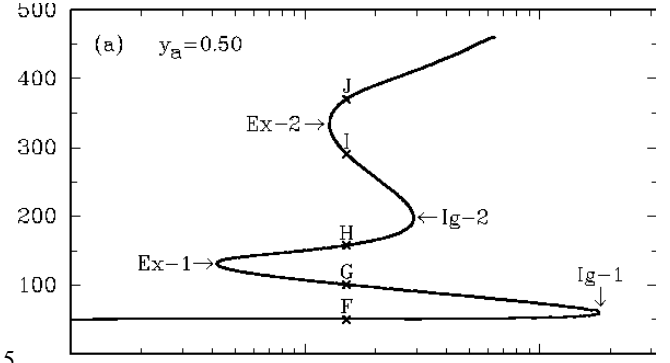












Sandia National Laboratories

LOCA: Library of Continuation Algorithms

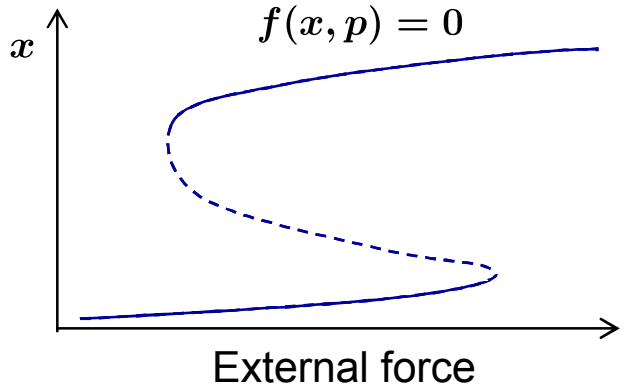
Salinger, Phipps, et. al.

Application code provides:

- Nonlinear steady-state residual and Jacobian fill:

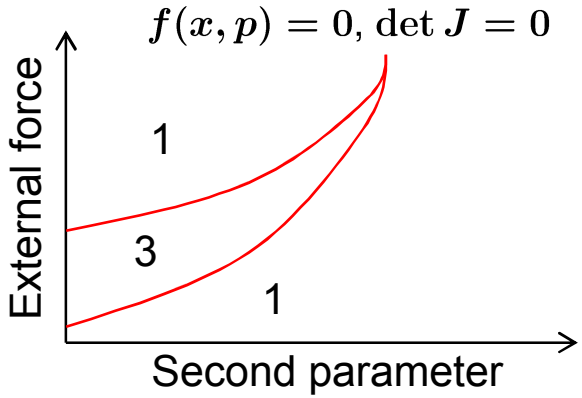
$$f(x,p) = \text{internal} - \text{external force}, \quad J = \frac{\partial f}{\partial x}$$

- Newton-like linear solves: $J\Delta x = -f$



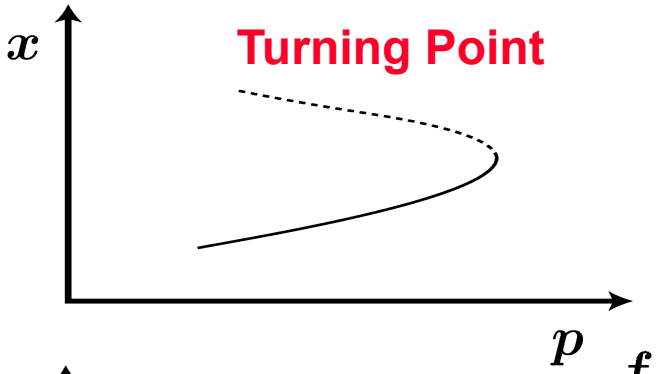
LOCA provides:

- Parameter Continuation: Tracks a family of steady state solutions with parameter
- Linear Stability Analysis: Calculates leading eigenvalues via Anasazi (Thornquist, Lehoucq)
- Bifurcation Tracking: Locates neutral stability point (x,p) and tracks as a function of a second parameter



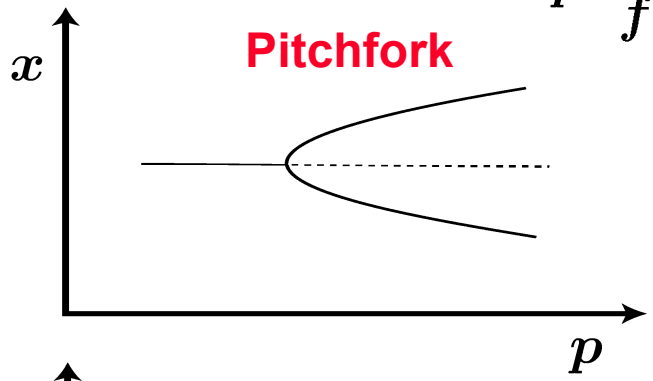
Sandia National Laboratories

Codimension 1 Bifurcations



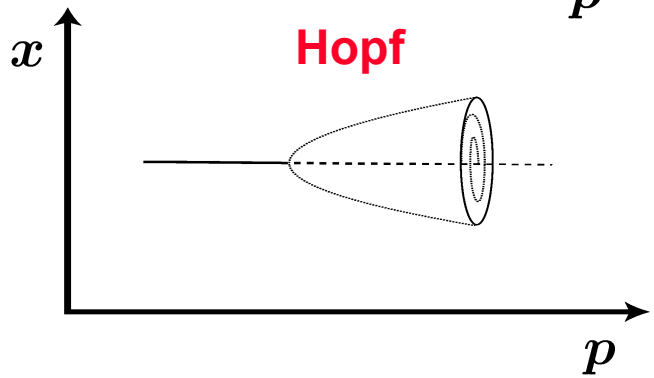
$$f(x,p) = 0$$
$$Jv = 0$$
$$\phi^T v = 1$$

- Combustion
- Buckling of an Arch



$$f(x,p) + s\psi = 0$$
$$Jv = 0$$
$$\langle x, \psi \rangle = 0$$
$$\phi^T v = 1$$

- Buckling of a Beam
- Pattern formation
- Cell differentiation (morphogenesis)



$$f(x,p) = 0$$
$$Jy - \omega M z = 0$$
$$Jz + \omega M y = 0$$
$$\phi^T y = 1$$
$$\phi^T z = 0$$

- Vortex Shedding
- Predator-Prey models
- Flutter

Sandia National Laboratories

Parameter Continuation, Linear Stability Analysis and Bifurcation Point Equations

Steady Solution (N)

$$R(x, p) = 0$$

Arclength Continuation (N+1)

$$R(x, p) = 0$$

$$\frac{dx}{ds} \cdot (x - x_o) + \theta^2 \frac{dp}{ds}(p - p_o) - s = 0$$

Eigenvalue Problem (N)

$$\mathbf{J}z = \lambda \mathbf{M}z$$

Generalized Cayley Transformation (N)

$$(\mathbf{J} - \sigma\mathbf{M})^{-1}(\mathbf{J} - \mu\mathbf{M})z = \left(\frac{\lambda - \mu}{\lambda - \sigma}\right)z$$

Turning Point (2N+1)

$$R = 0$$

$$\mathbf{J}n = 0$$

$$\phi \cdot n = 1$$

Pitchfork (2N+2)

$$R + \varepsilon\Psi = 0$$

$$\mathbf{J}n = 0$$

$$\langle x, \Psi \rangle = 0$$

$$\phi \cdot n = 1$$

Hopf (3N+2)

$$R = 0$$

$$\mathbf{J}y + \omega\mathbf{M}z = 0$$

$$\mathbf{J}z - \omega\mathbf{M}y = 0$$

$$\phi \cdot y = 1$$

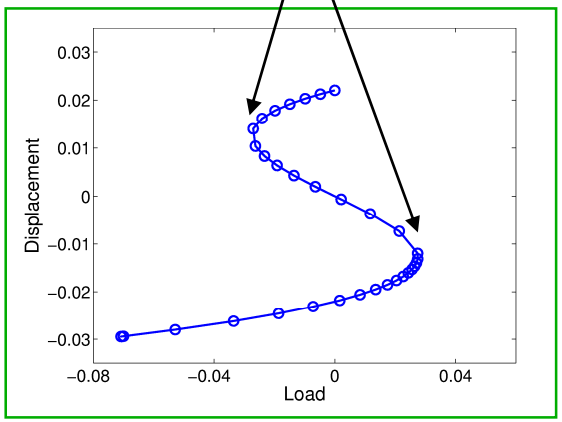
$$\phi \cdot z = 0$$

Sandia National Laboratories

Turning Point Identification

Turning Point Bifurcation

$$f(x,p) = 0$$
$$Jv = 0$$
$$\phi^T v = 1$$



Full Newton Algorithm

$$\begin{bmatrix} J & 0 & f_p \\ (Jv)_x & J & (Jv)_p \\ 0 & \phi^T & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta v \\ \Delta p \end{bmatrix} = - \begin{bmatrix} f \\ Jv \\ \phi^T v - 1 \end{bmatrix}$$

Bordering Algorithm

$$Ja = -f$$
$$Jb = -f_p$$
$$Jc = -(Jv)_x a - Jv$$
$$Jd = -(Jv)_x b - (Jv)_p$$

$$\Delta p = \frac{1 - \phi^T v - \phi^T c}{\phi^T d}$$
$$\Delta x = a + \Delta p b$$
$$\Delta v = c + \Delta p d$$

… but 4 solves of J per Newton iteration are used to drive J singular!

Sandia National Laboratories

Bordering Algorithm for Hopf Tracking

$$
\begin{aligned}
R(x,p) &= 0 \\
Jy + \omega Mz &= 0 \\
Jz - \omega My &= 0 \\
\phi \cdot y - 1 &= 0 \\
\phi \cdot z &= 0
\end{aligned}
$$

$$
\begin{bmatrix}
J & 0 & 0 & 0 & \frac{\partial R}{\partial p} \\
\frac{\partial (Jy)}{\partial x} + \frac{\partial (\omega Mz)}{\partial x} & J & \omega M & Mz & \frac{\partial (Jy)}{\partial p} + \frac{\partial (\omega Mz)}{\partial p} \\
\frac{\partial (Jz)}{\partial x} - \frac{\partial (\omega My)}{\partial x} & -\omega M & J & -My & \frac{\partial (Jz)}{\partial p} - \frac{\partial (\omega My)}{\partial p} \\
0 & \phi^t & 0 & 0 & 0 \\
0 & 0 & \phi^t & 0 & 0
\end{bmatrix}
\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta \omega \\ \Delta p \end{bmatrix}
=
\begin{bmatrix} -R \\ -Jy - \omega Mz \\ -Jz + \omega My \\ 1 - \phi \cdot y \\ \phi \cdot z \end{bmatrix}
$$

$$Ja = -R$$

$$Jb = -\frac{\partial R}{\partial p}$$

$$
\begin{bmatrix} J & \omega M \\ -\omega M & J \end{bmatrix}
\begin{bmatrix} c \\ d \end{bmatrix}
=
\begin{bmatrix} Mz \\ -My \end{bmatrix}
$$

$$
\begin{bmatrix} J & \omega M \\ -\omega M & J \end{bmatrix}
\begin{bmatrix} e \\ f \end{bmatrix}
=
\begin{bmatrix} \frac{\partial (Jy)}{\partial x} a + \omega \frac{\partial (Mz)}{\partial x} a \\ \frac{\partial (Jz)}{\partial x} a - \omega \frac{\partial (My)}{\partial x} a \end{bmatrix}
$$

$$
\begin{bmatrix} J & \omega M \\ -\omega M & J \end{bmatrix}
\begin{bmatrix} g \\ h \end{bmatrix}
=
\begin{bmatrix} \frac{\partial (Jy)}{\partial p} + \frac{\partial (Jy)}{\partial x} b + \omega \frac{\partial (Mz)}{\partial p} + \omega \frac{\partial (Mz)}{\partial x} b \\ \frac{\partial (Jz)}{\partial p} + \frac{\partial (Jz)}{\partial x} b - \omega \frac{\partial (My)}{\partial p} - \omega \frac{\partial (My)}{\partial x} b \end{bmatrix}
$$

$$\Delta p = \frac{\phi \cdot d(\phi \cdot e) + \phi \cdot d - \phi \cdot f(\phi \cdot e)}{\phi \cdot c(\phi \cdot h) - \phi \cdot e(\phi \cdot g)}$$

$$\Delta \omega = -\frac{1 + (\phi \cdot e) + (\phi \cdot g)\Delta p}{(\phi \cdot c)}$$

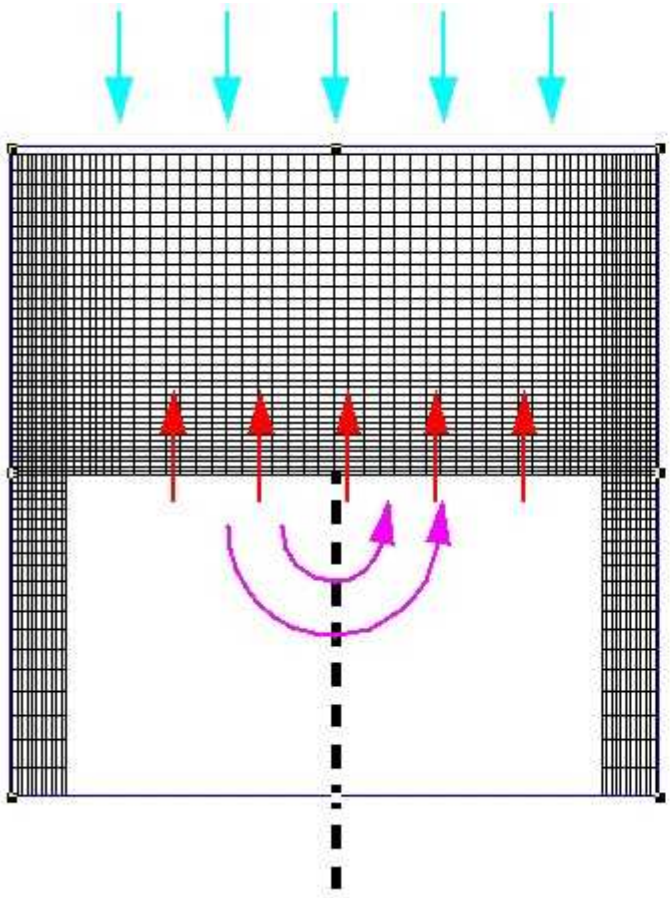
$$\Delta x = a + b\Delta p$$

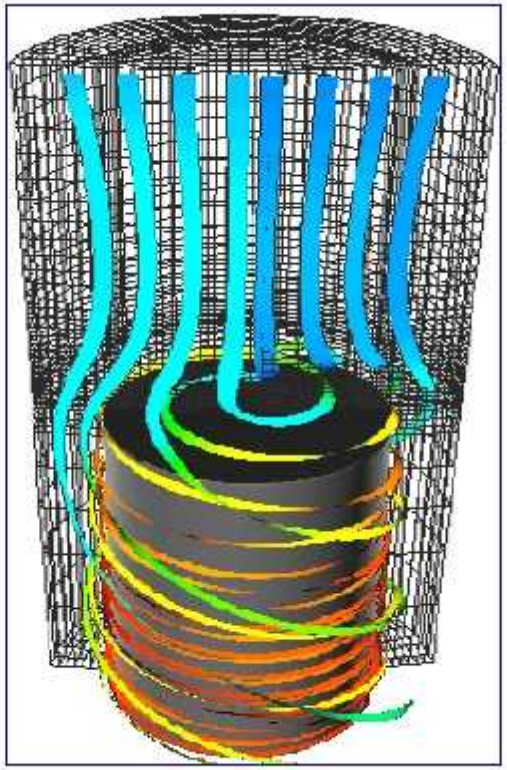
$$\Delta y = -y - e - g\Delta p - c\Delta \omega$$

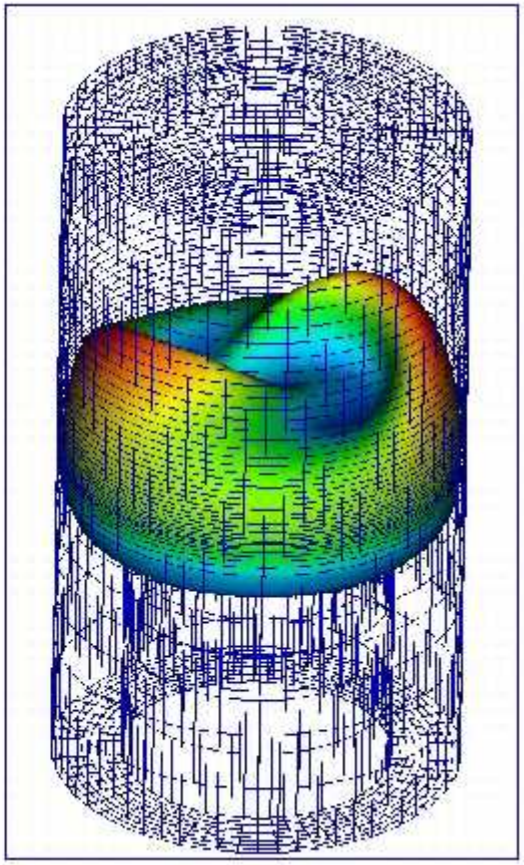
$$\Delta z = -z - f - h\Delta p - d\Delta \omega$$

Sandia National Laboratories

CVD Reactor Design and Scale-up:
Buoyancy force can lead to undesirable flows







Chemical Vapor Deposition of Semiconductors: GaN, GaAs

Sandia National Laboratories

Model Parameters

- Real Variables
  - Pressure
  - Susceptor Temperature
  - Inlet Temperature
  - Rotation Rate
  - Inlet Velocity
- Dimensionless Variables
  - Reynolds Number
  - Rayleigh Number
  - Nusselt Number
  - Prandtl Number
- Geometry

1. Velocity is tied to rotation rate:

$$u_z \rightarrow u_{zo} = 0.885\sqrt{\omega_o \upsilon}$$

2. Rotational Reynolds Number:

$$Re = \frac{\omega R_i^2}{\upsilon}$$

3. Length scale for Rayleigh number is the von Karman boundary layer thickness:

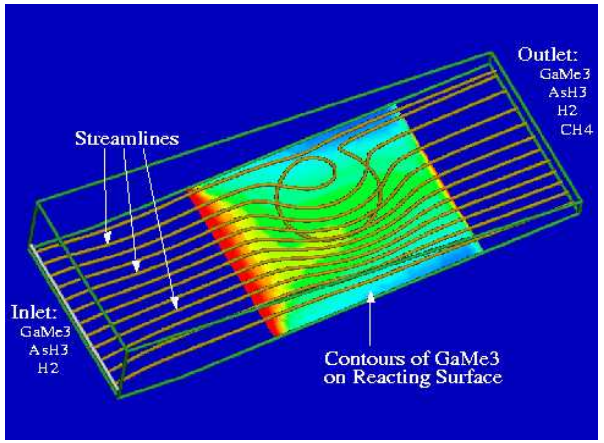
$$Ra = \frac{g\beta\Delta T\delta^2 R_i}{\alpha\upsilon} = \frac{g\beta\Delta T R_i}{\alpha\omega}$$

$$\delta = \sqrt{\upsilon/\omega}$$

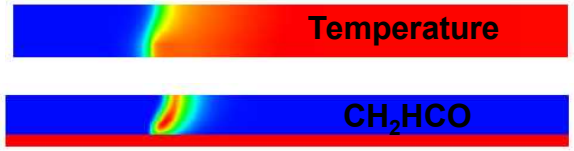
Sandia National Laboratories

MPSalsa: 2D and 3D Parallel Reacting Flow Code

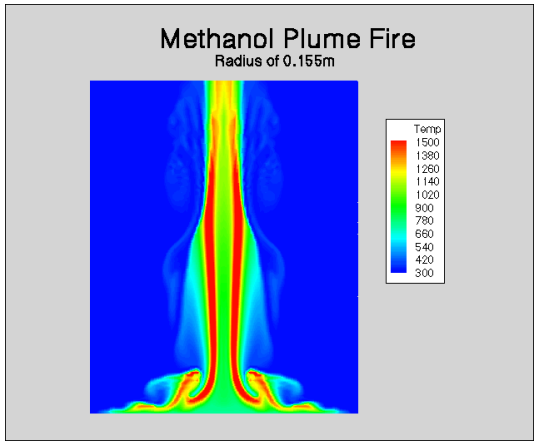
GaAs CVD



Ethane Catalysis

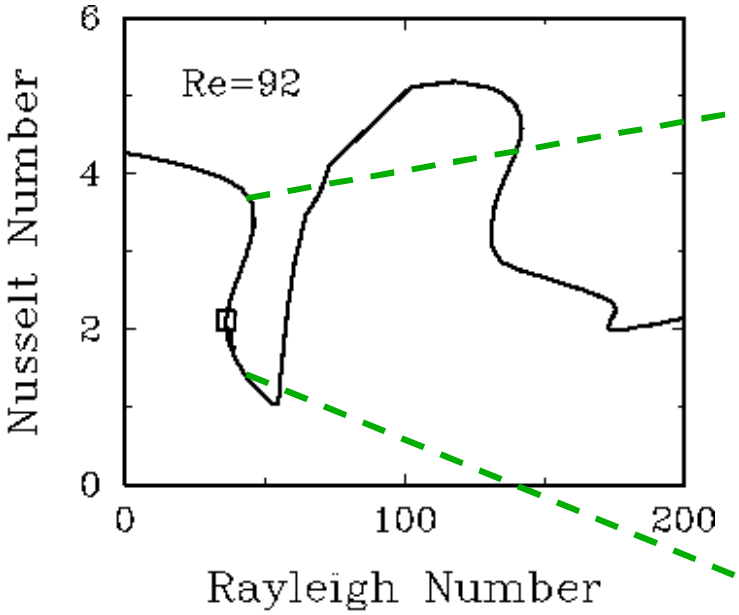


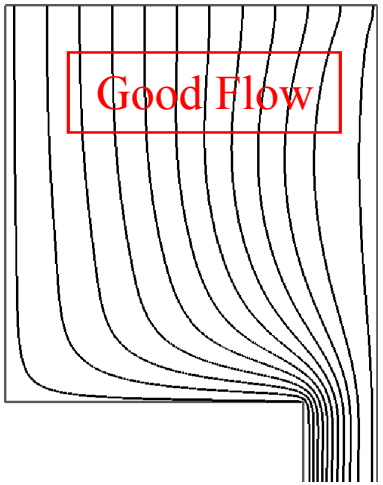
Turbulent Combustion

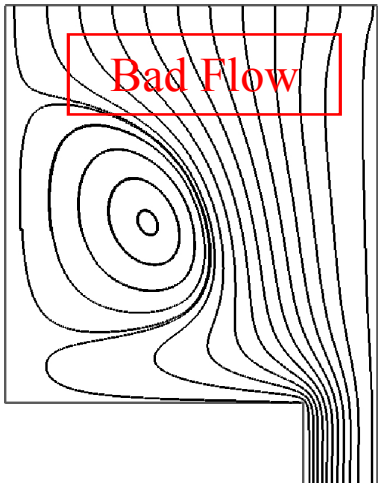


- **Formulation:** Galerkin/Least-Squares Finite Elements (GLSFEM)
  - Adaptive Mesh refinement
  - Static and Dynamic Load Balancing (**Chaco, Zoltan**)
  - Parallel FE Database (**Nemesis**)
- **Physics:** Laminar and turbulent, incompresible and ideal gas, full multicomponent and mixture averaged tranport, finite rate chemical reactions (**CHEMKIN, Surface CHEMKIN, and TRANFIT**).
- **Transient Solver:** Fully implicit 1st and 2nd order (alpha method), operator splitting for multiple time scales.
- **Steady State Solver:** fully coupled inexact Newton method w/ backtracking
- **Linear solvers:** iterative methods based preconditioned Krylov methods. Multilevel preconditioning (**Aztec/ML**).
- **Bifurcation Analysis:** Zero, first, and arc-length continuation. Turning point, pitchfork and Hopf bifurcation tracking (**LOCA**).
- **Scalable Linearized Stabiity Analysis:** Arnoldi's method with Cayley transforms (**P_ARPACK** - being generalized in **Anasazi**).
- **Optimization:** **rSQP** and **DAKOTA**.

Good and bad flows are found to coexist at certain values of (Ra, Re)



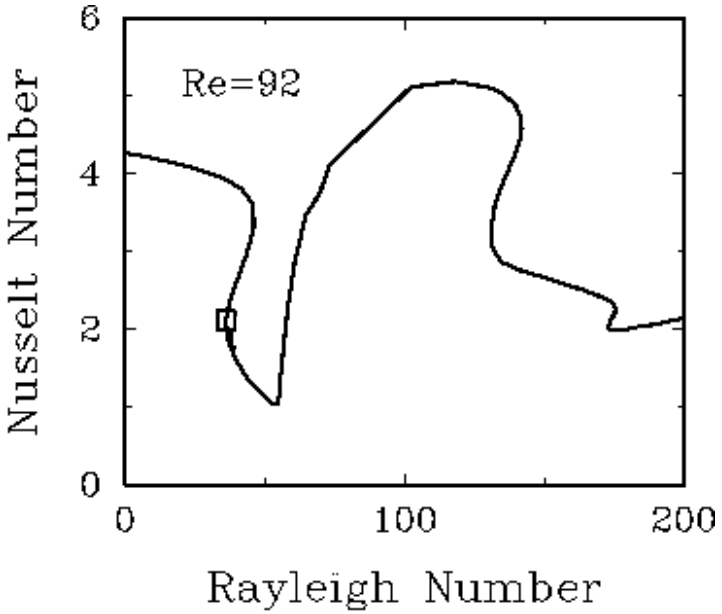


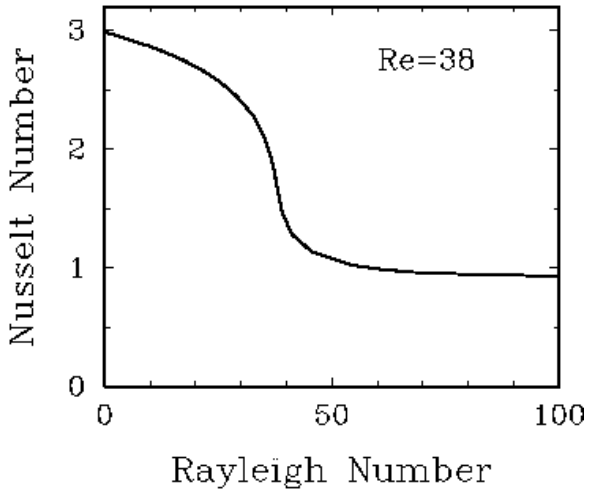


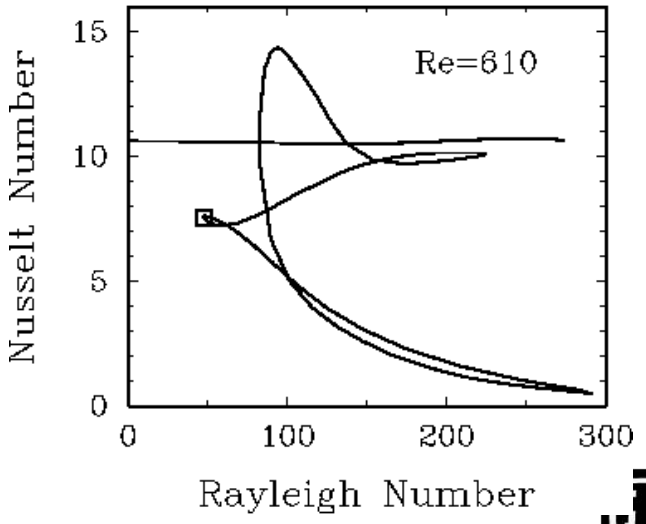
30500 Unknowns

Sandia National Laboratories

Good and bad flows are found to coexist at certain values of (Ra, Re)



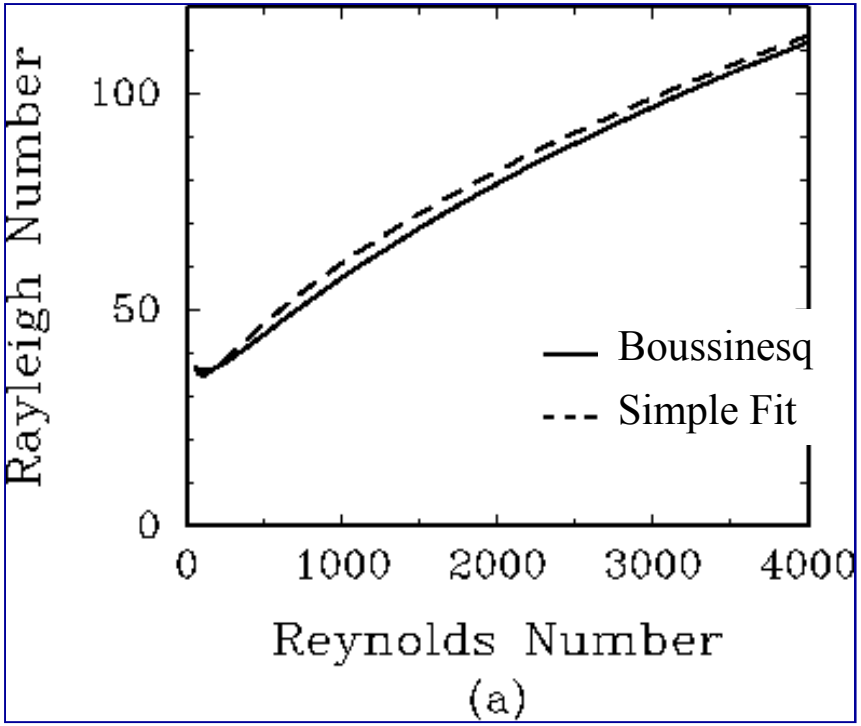


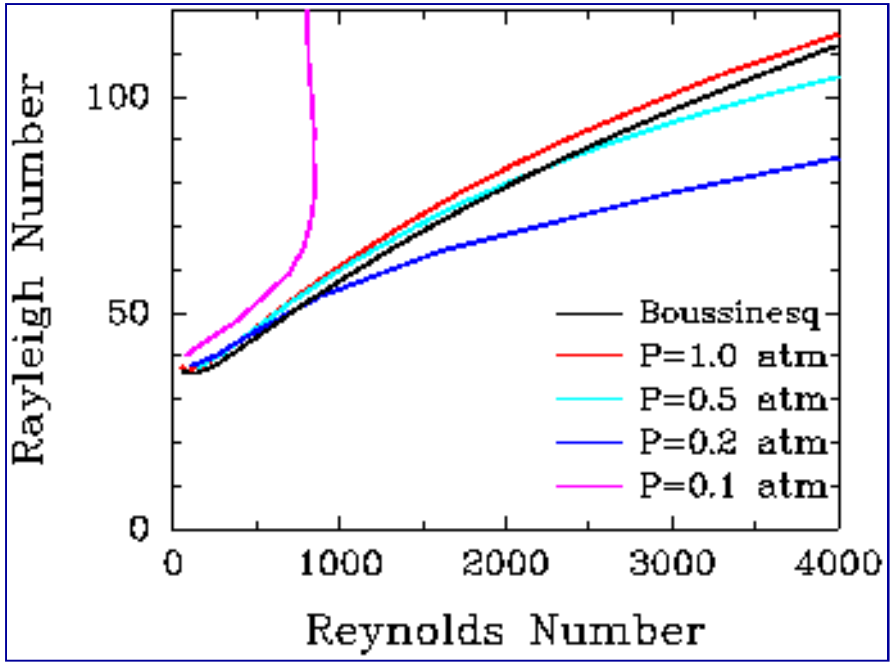


30500 Unknowns

Sandia National Laboratories

Tracking of bifurcation leads to design rule





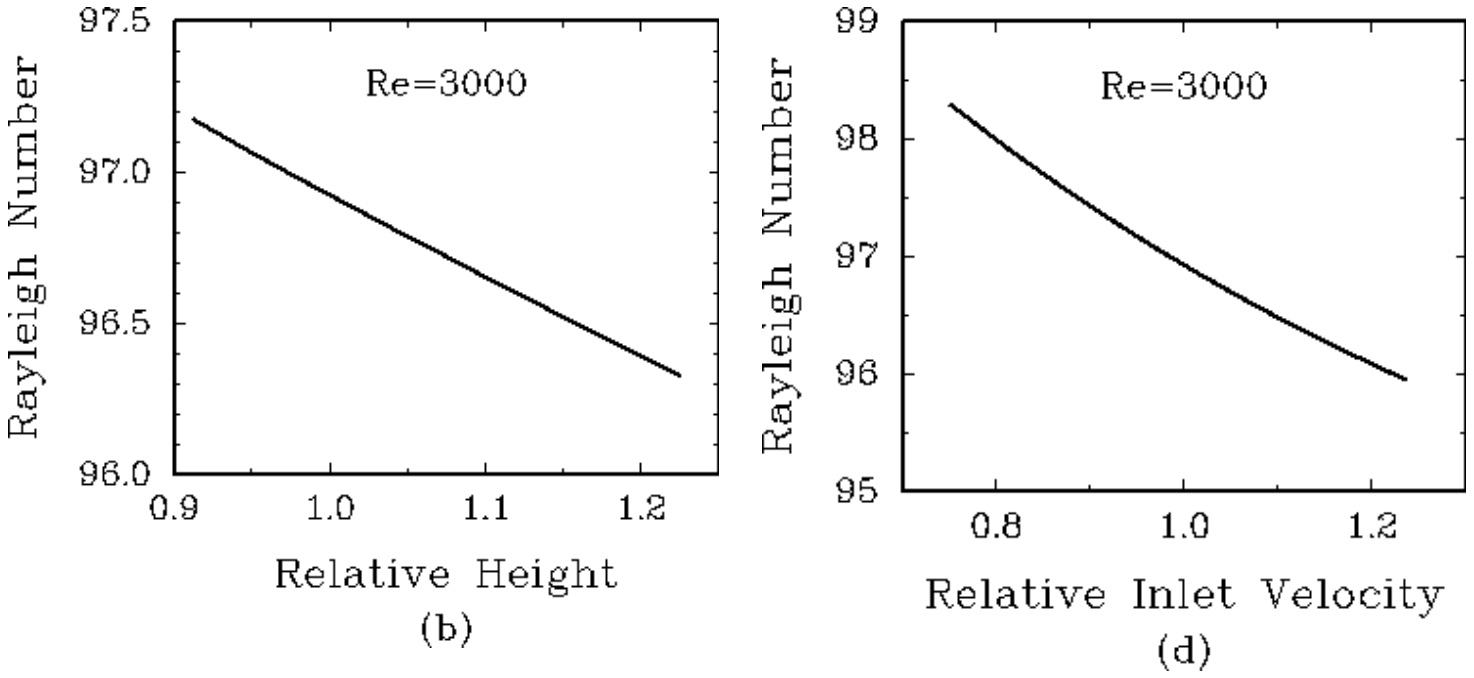
$$Ra = 1.75 Re^{0.5}\left(1 + \frac{100}{Re}\right)$$

Ideal gas curves collapse onto Boussinesq for good choice of T_o

Pawlowski, Salinger, Romero, Shadid 2001

Sandia National Laboratories

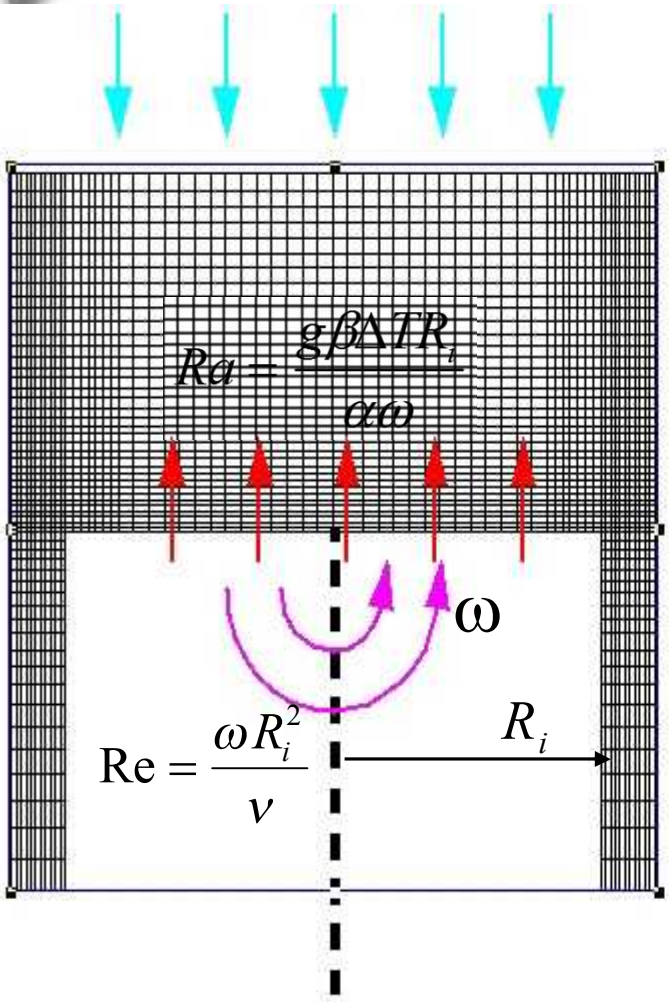
Results Determined to be Insensitive with respect to Two Key Parameters

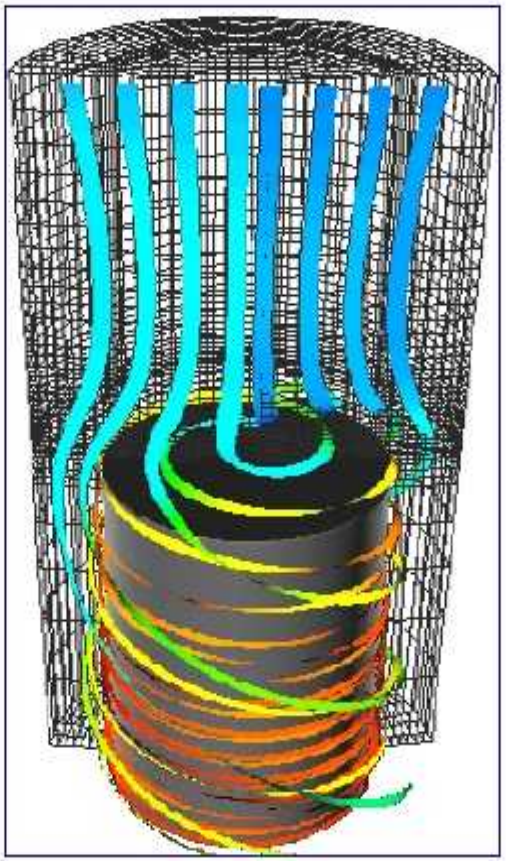


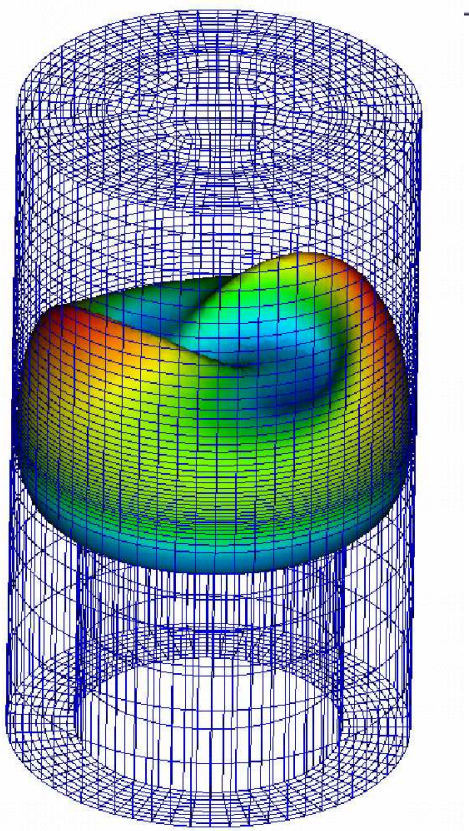
Turning point tracking runs at Re=3000

Sandia National Laboratories

Mapping Complex Solution Spaces: CVD Reactor Design and Scale-up





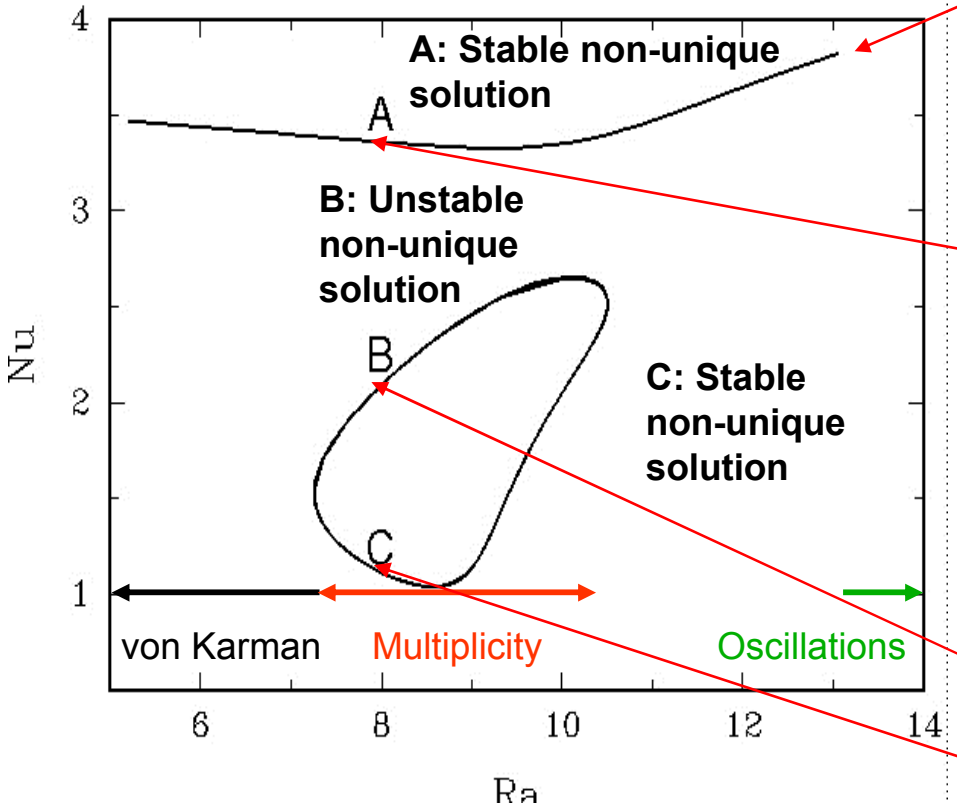


M = 2 instability

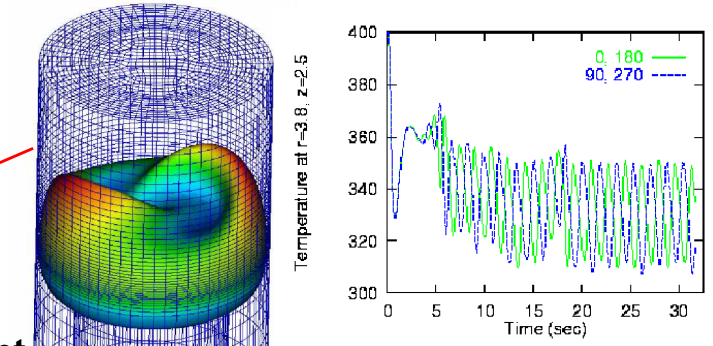
Chemical Vapor Deposition of Semiconductors: GaN, GaAs

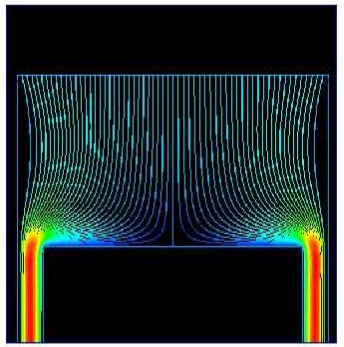
Sandia National Laboratories

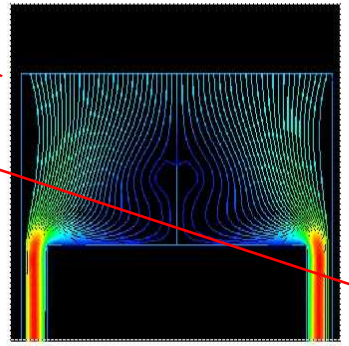
Stability of Rotating Disk Reactor: Global stability limit and linear stability results

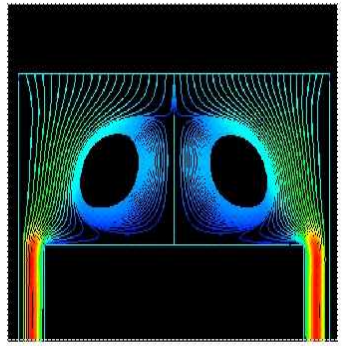


Time dependent solution







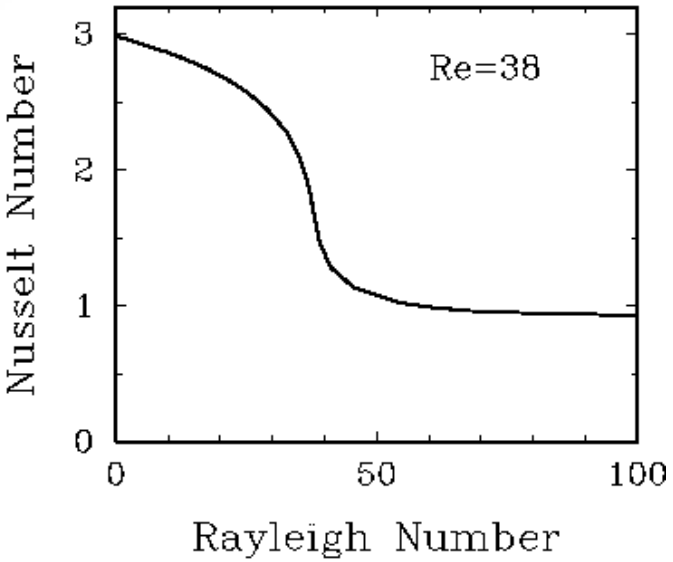


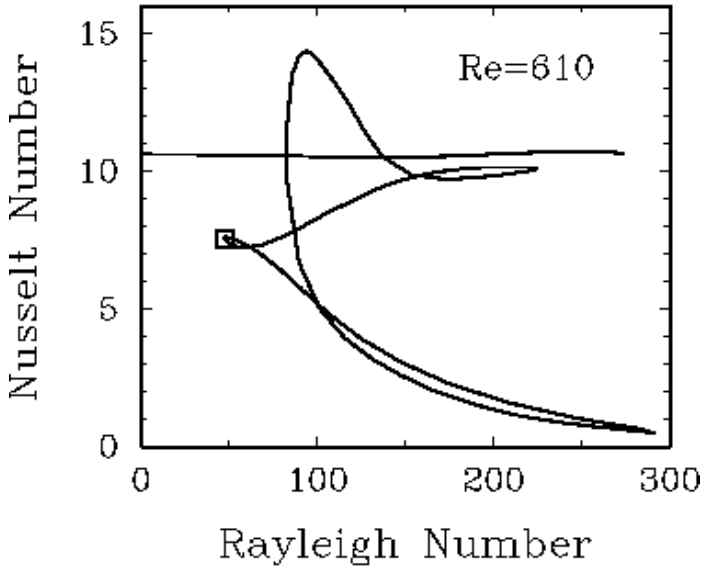
Fixed Re

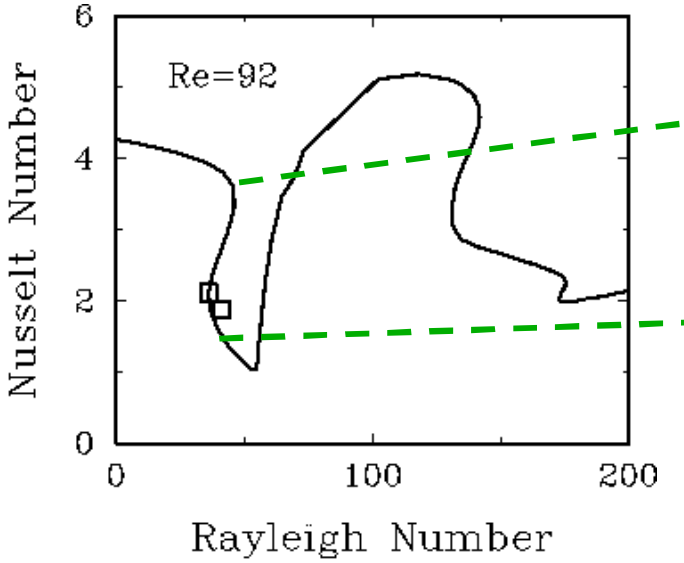
LOCA - Salinger, Pawlowski, Phipps

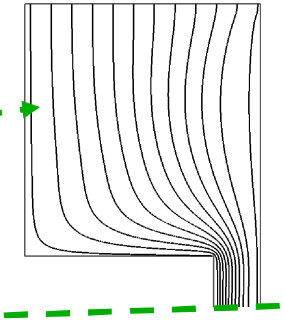
Sandia National Laboratories

Structure of Solution Space with Variation of 2 Parameters (Ra, Re) is Complex

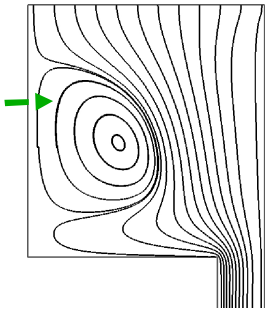








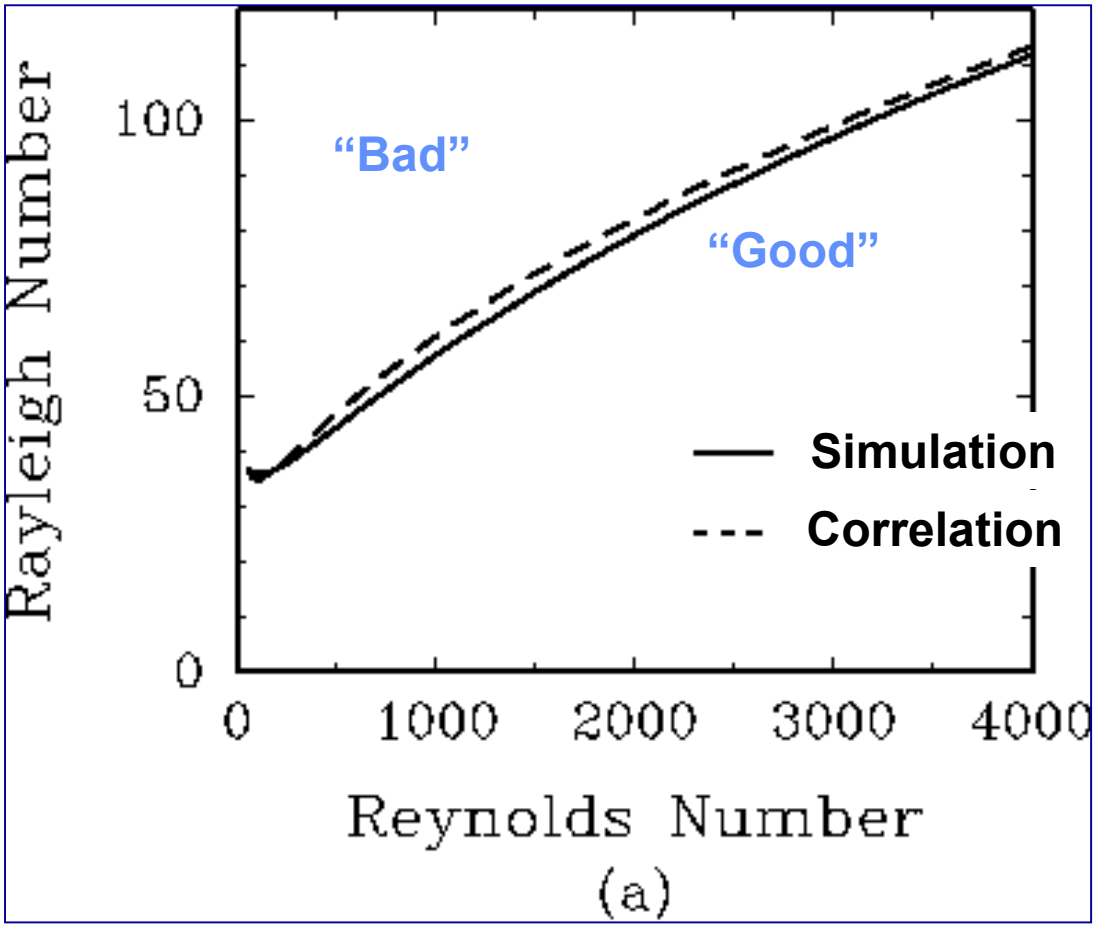
Bad Flow



Good Flow

Sandia National Laboratories

Direct Tracking of Turning Point Leads to Scale-up Stability Design Rule



$$\mathbf{F}(\mathbf{x}, \mathrm{Re}^*, \mathrm{Ra}^*) = \mathbf{0}$$

$$\mathbf{F}'\mathbf{v} = \mathbf{0}$$

$$\Upsilon^T\mathbf{v} - 1 = 0$$

Solve extended system with Newton's method

Power Law Correlation Fit to Simulation Results:

$$\mathrm{Ra} = 1.75\mathrm{Re}^{0.5}\left(1 + \frac{100}{\mathrm{Re}}\right)$$

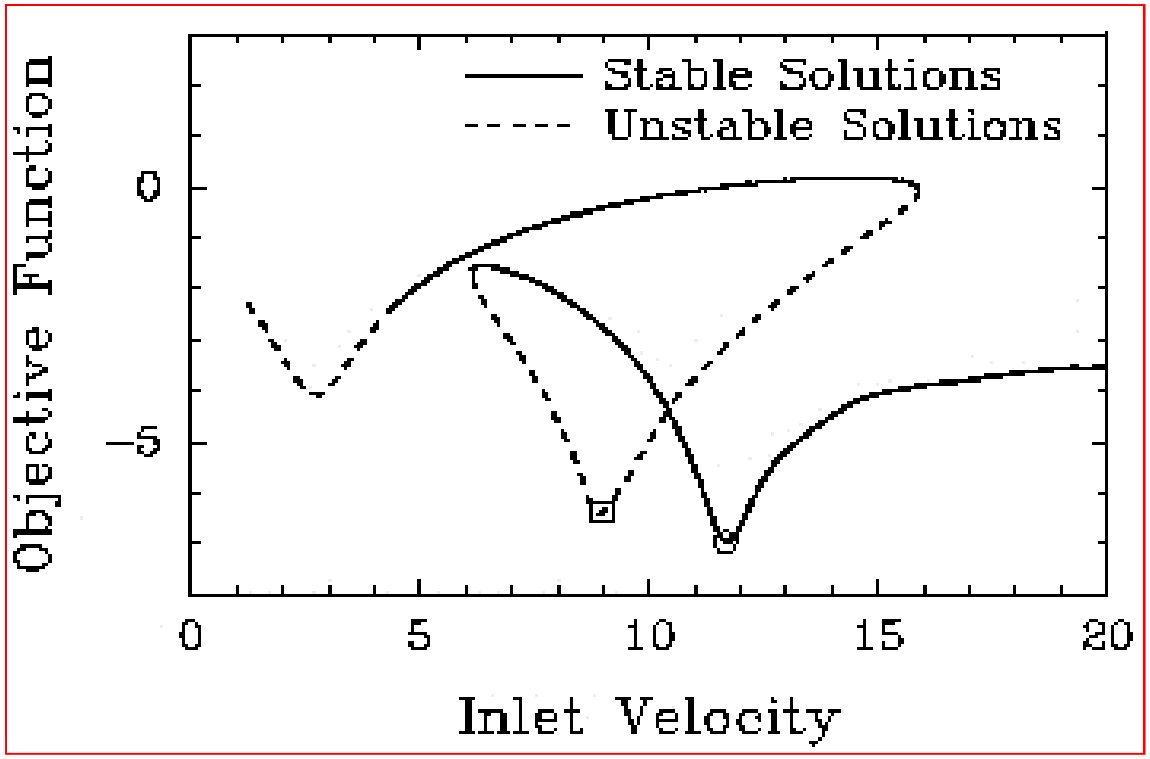
Pawlowski, Salinger, Romero, Shadid 2001

Sandia National Laboratories

Optimization

Sandia National Laboratories

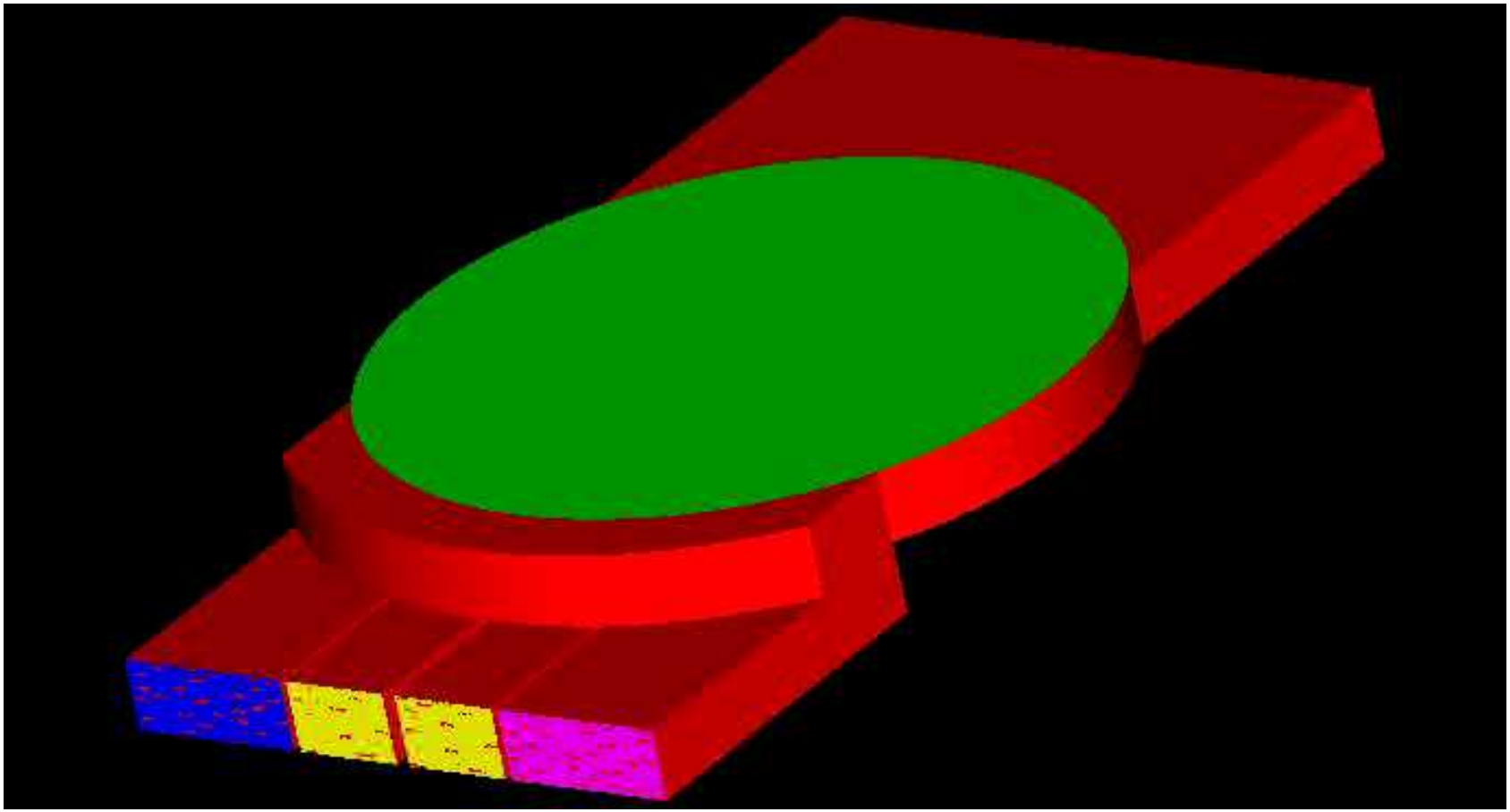
Optimization Algorithms, such as rSQP, Need Same Calls as Bifurcation Algs



Collaboration with Biegler, CMU

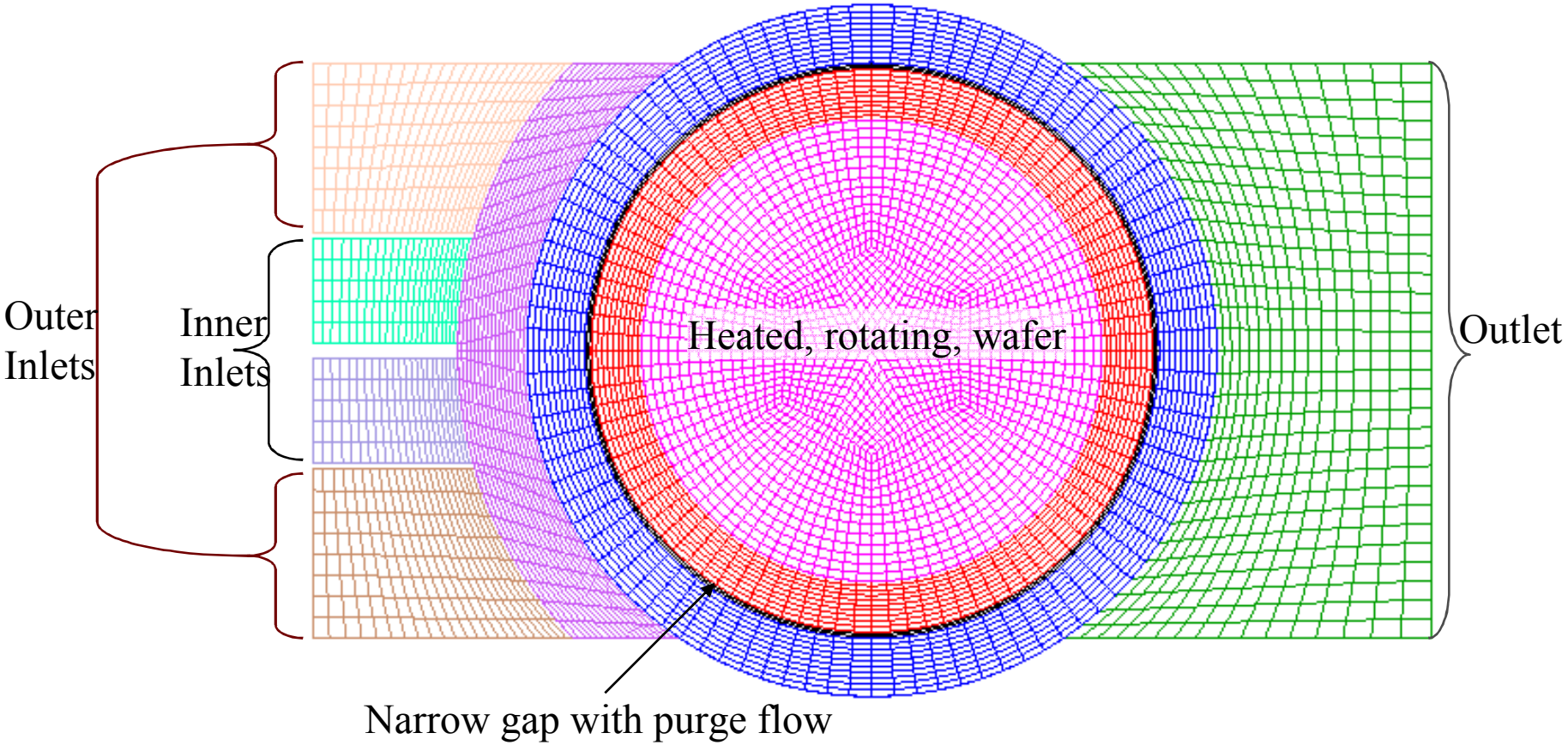
Sandia National Laboratories

Chemical Vapor Deposition Reactor for Growing Epitaxial Silicon on 8” Wafer



Sandia National Laboratories

150K Node Mesh: Bottom View



Sandia National Laboratories

Chemistry Model for Silicon Epitaxy from Trichlorosilane in Hydrogen Carrier

3 Species Model

$SiCl_3H$	$X_{TCS}=0.024$	$Y_{TCS}=0.625$	
HCl	$X_{HCl}=0.000$	$Y_{HCl}=0.000$	T_{in} 300K
H_2	$X_{H2}=0.976$	$Y_{H2}=0.375$	

Physical Properties are a function of local $\underline{Y}$ and T, using Chemkin

ρ, μ, Cp, D_k, D_k^T

Reaction Model of Kommu, Wilson, and Khomami (2000):

Langmuir-Hinshelwood Surface Reactions:

T_{wafer}=1398K

$$SiCl_3H(g) + \Theta \Rightarrow SiCl_2^{\Theta} + HCl(g)$$

$$SiCl_2^{\Theta} + H_2(g) \Rightarrow Si(s) + 2HCl(g) + \Theta$$

No Gas Phase Reactions:

Sandia National Laboratories

Nonlinear Solution Strategy and Details

First solution at realistic conditions: ~30 steady state solves:

(1) Solve for flow at **low P, const T, const Y, coarse mesh**. (2) Ramp up **P**.

(3) Ramp up **T**. (4) Ramp up **Surf Reaction**.

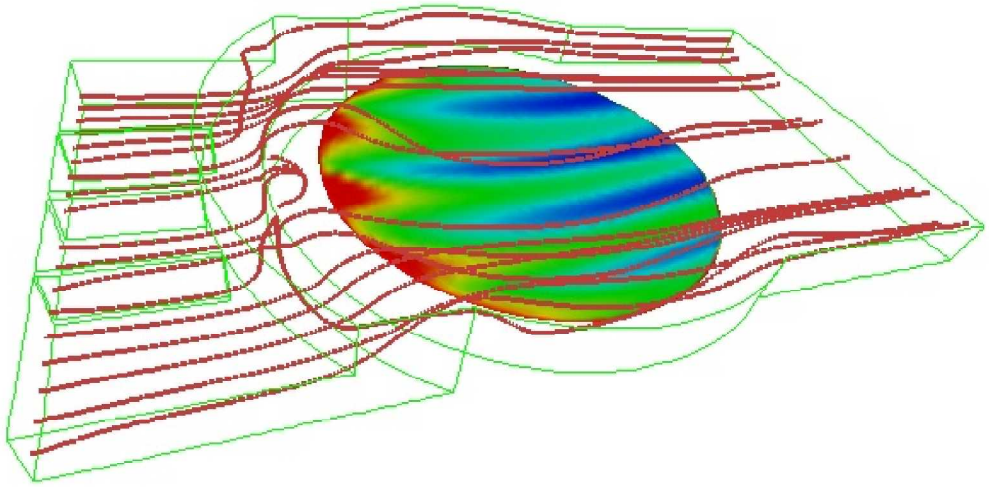
(5) Turn on **Thermal Diffusion.** (6) Reconverge on **fine mesh** (1.2M unks).

Typical "forward" nonlinear solve on 48-(3GHz) Processors: 18 Minutes

Newton Iterations: 5

Linear Solve (ILU, GMRES, 300iter; tol=10^{-4}): 160sec

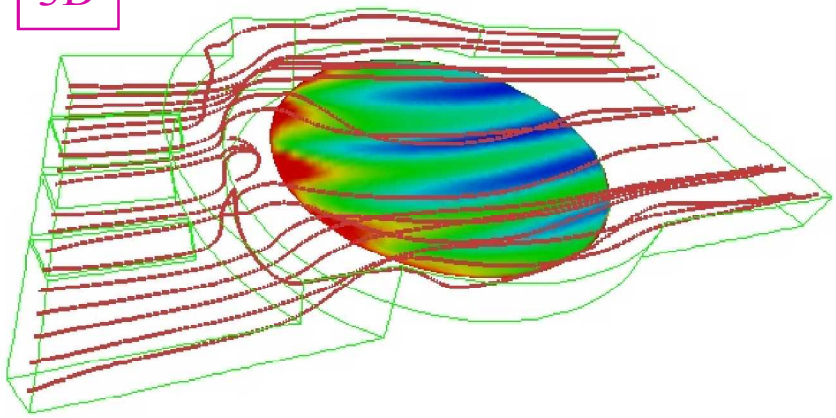
Jacobian Fill (Element-level Numerical): 100sec



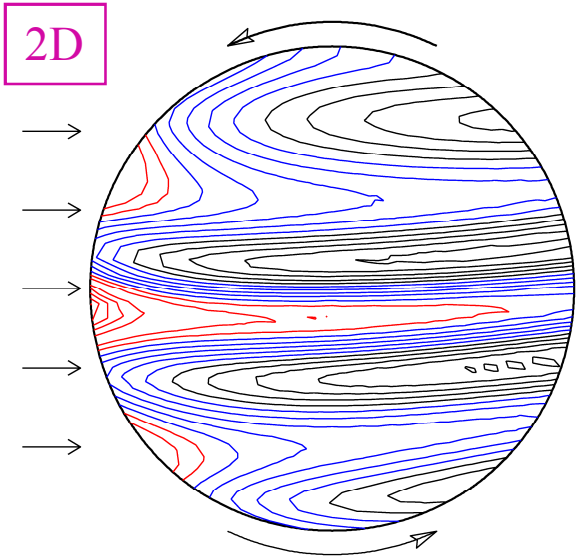
Sandia National Laboratories

Uniform Deposition Profiles are Desirable for High Quality Silicon Films

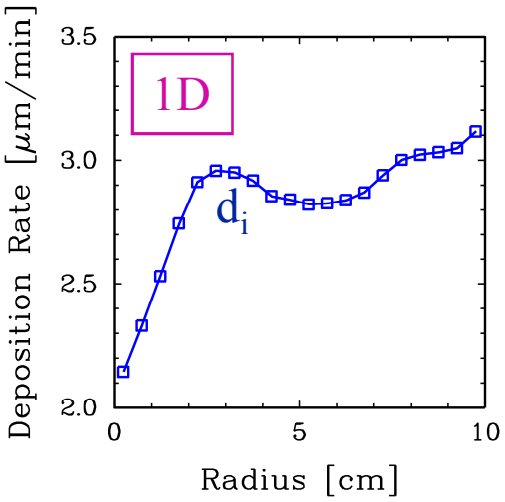
3D



2D



1D

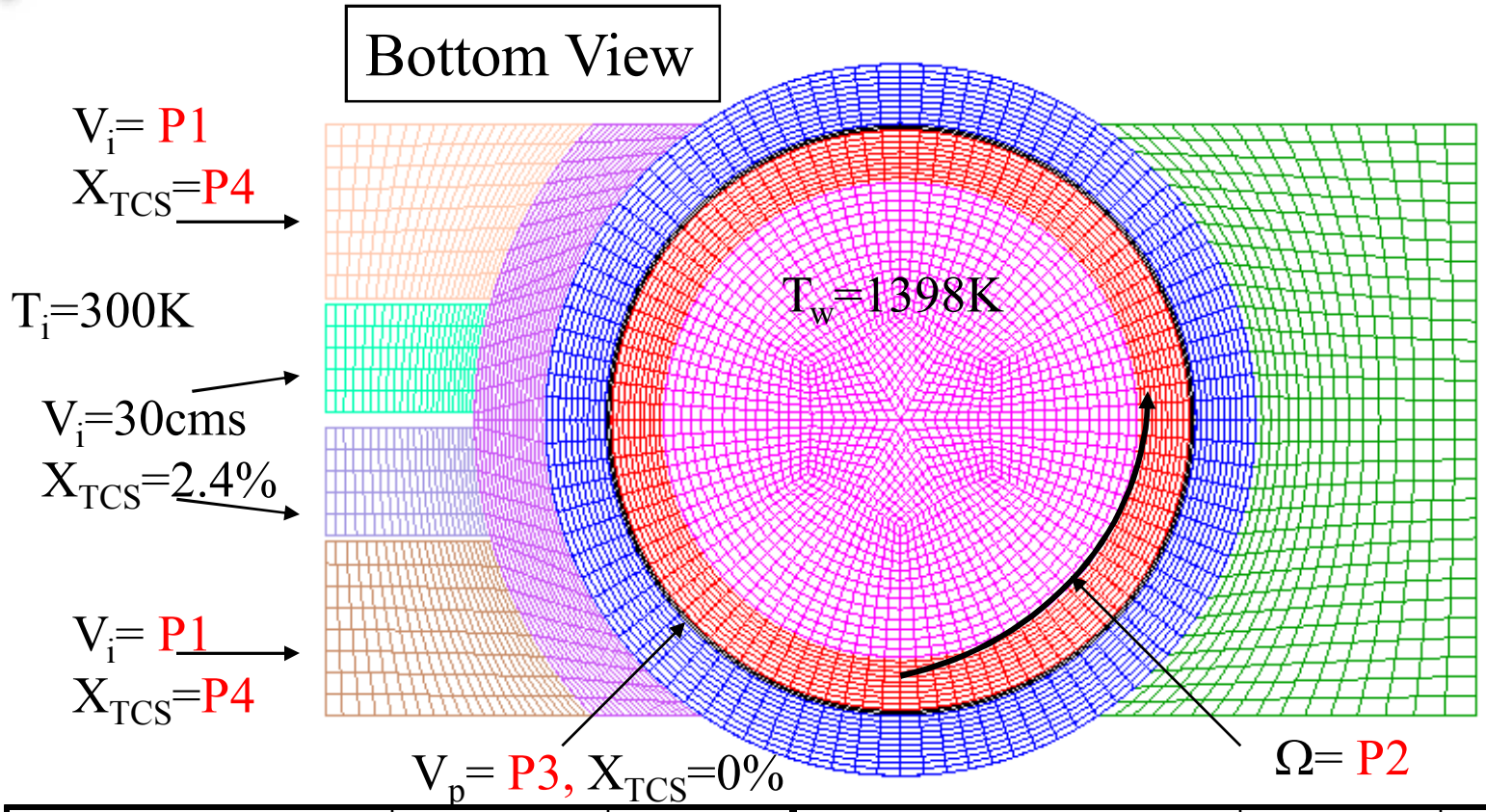


0D Objective Function:

$$f = \frac{1}{2}\sum_{\text{radii}} (d_i/d_{ave} - 1)^2$$

Sandia National Laboratories

Operating Parameters and Other B.C.'s



P1: Side Inlets Velocity V_i	30cms	15-60	P3: Purge Flow Velocity V_p	10cms	0-50
P2: Wafer Rotation Ω	10RPM	0-150	P4 : Side Inlets Mole Frac. X_i	**0.024**	**0.018-0.30**

Comparison Optimization Algorithms Interface Requirements

NAND{Dakota(Dot-SQP)} SAND{Moocho(rSQP)}

$$\begin{aligned} \min \quad & f(y,u) \\ & c(y,u) = 0 \end{aligned}$$

Requirements from Application Code

1. Nonlinear Solve (c=0) with convergence criterion
2. Objective Function (f)
3. 1 Convergence Criterion (f)

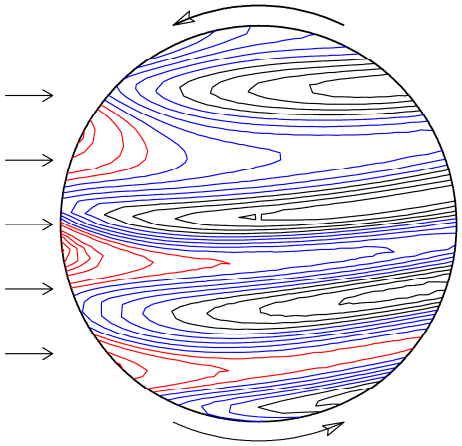
1. Constraint Vector (c)
2. Jacobian Solve (J^{-1} to 10^{-8})
3. Sensitivity vectors (dc/du)
4. Objective Function (f)
5. Obj Fn Gradient (df/dy)
6. Adjoint J Solve (J^{-T} to 10^{-8})
7. 2 Convergence Criteria (c,f)
8. Needs Good Scaling, c *vs* f!

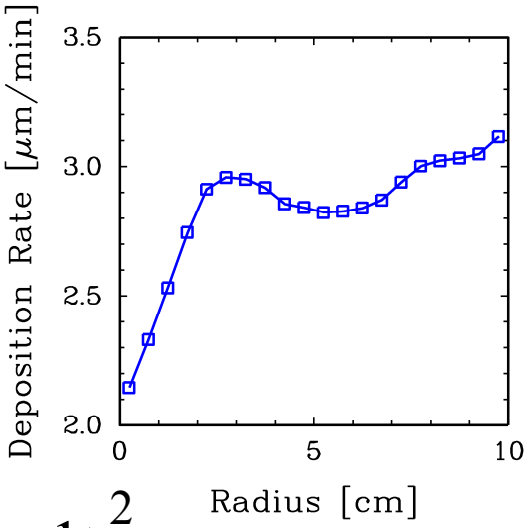
CPU time versus Developer time.

Sandia National Laboratories

Implementation of SAND Derivatives can be difficult and time consuming

df/dy calc difficult: changed f(y) from previous NAND work



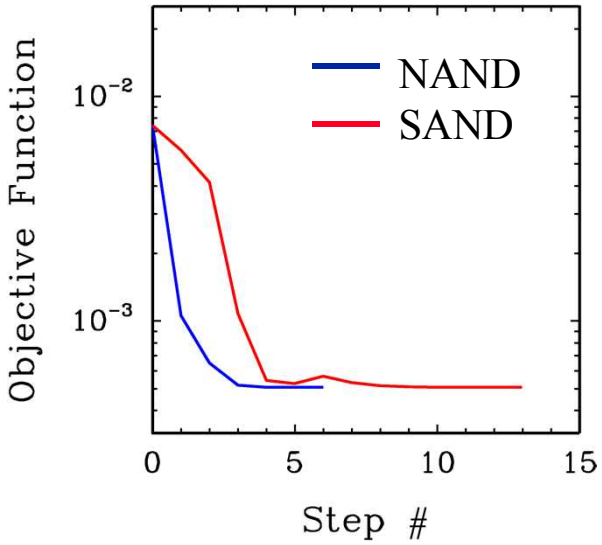


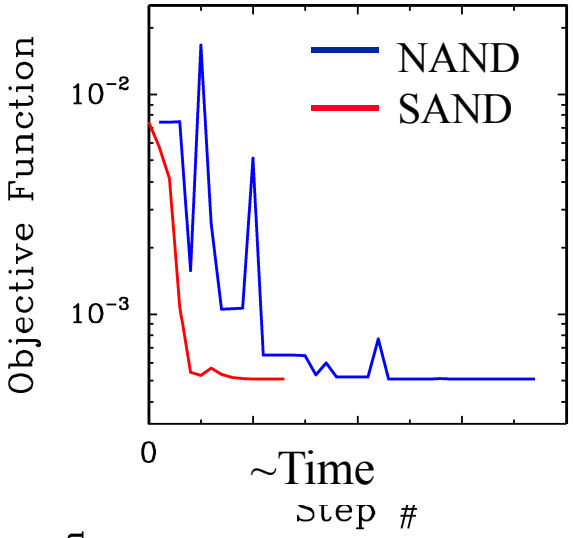
$$f = \frac{1}{2}\sum_{\text{radii}} \left(d_i / d_{ave} - 1 \right)^2$$

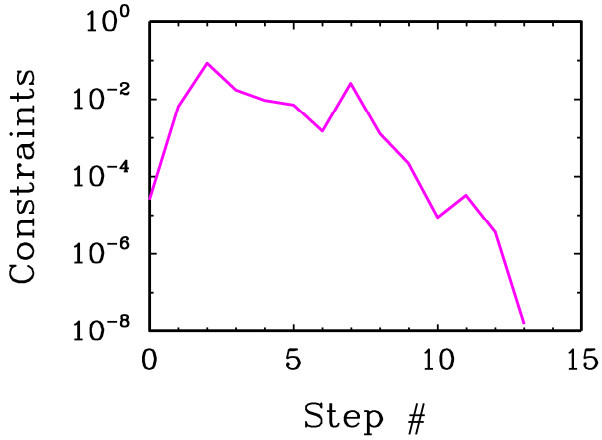
---

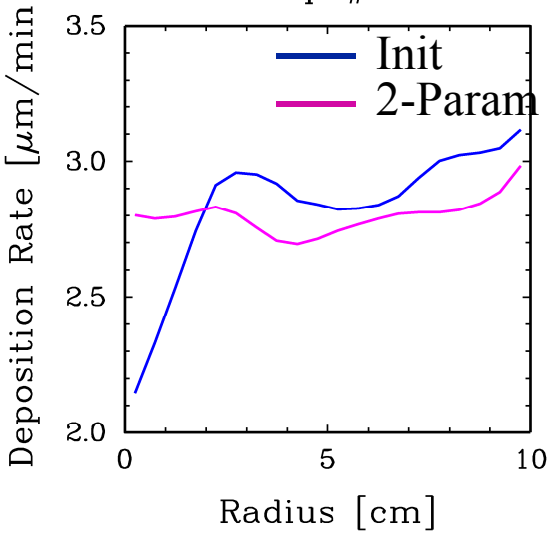
dc/du calculations: All Dirichlet BCs => c(u) is linear
dc/du precalculated with F.D.

2-Parameter Results



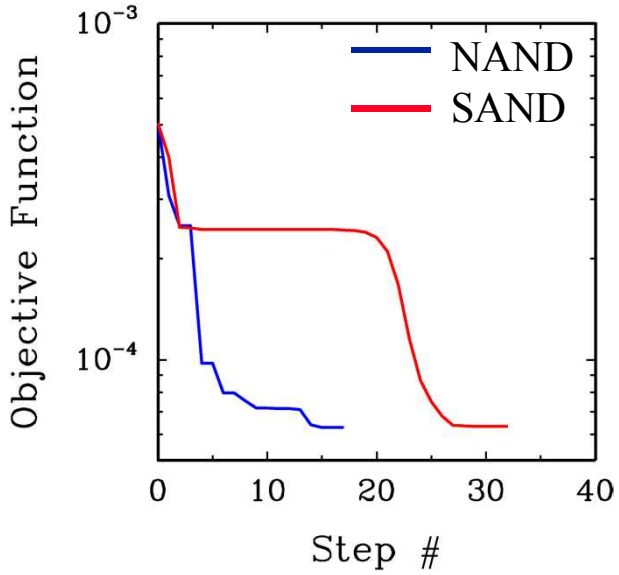


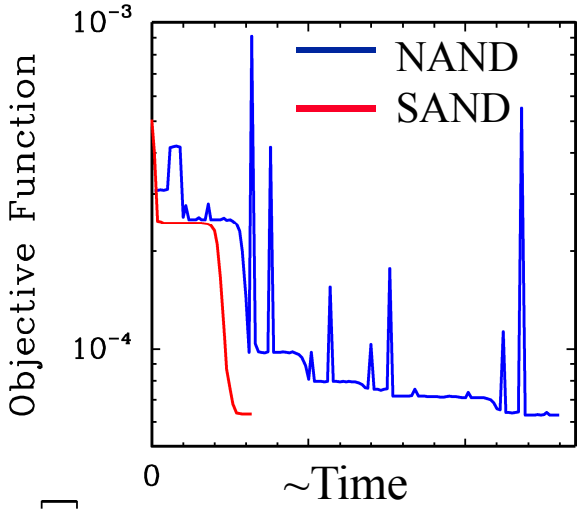


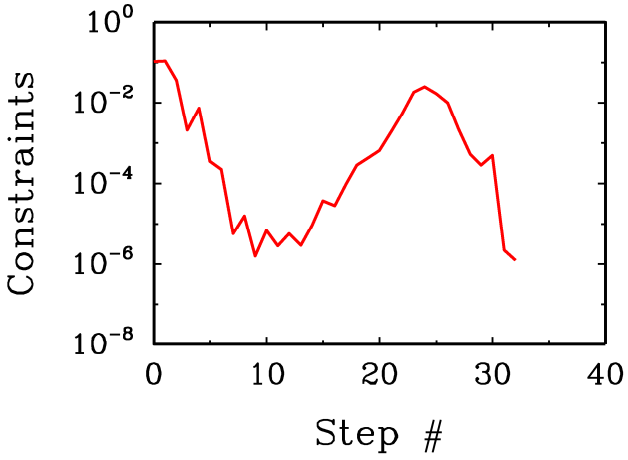


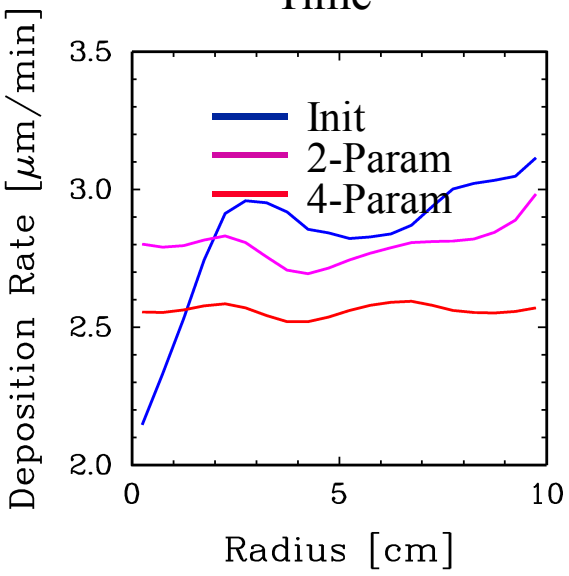
Sandia National Laboratories

4-Parameter Results
(starting near 2-parameter optimum)





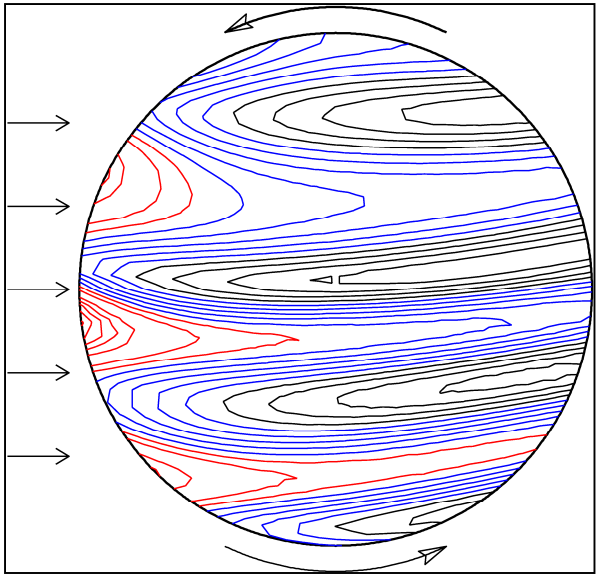




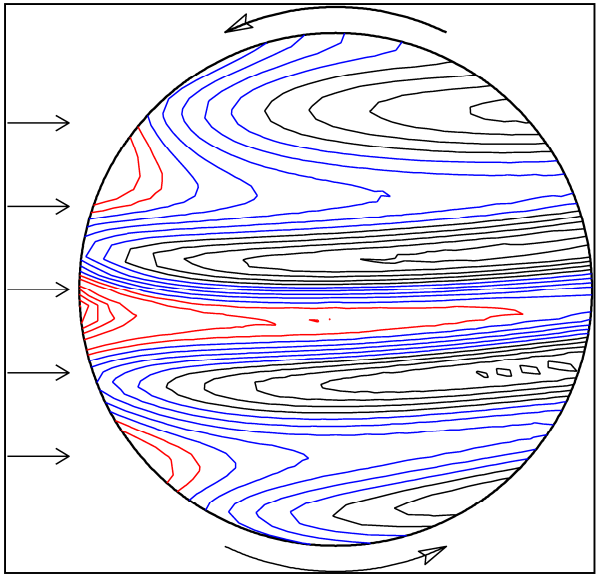
Sandia National Laboratories

Comparison of 2D Profiles

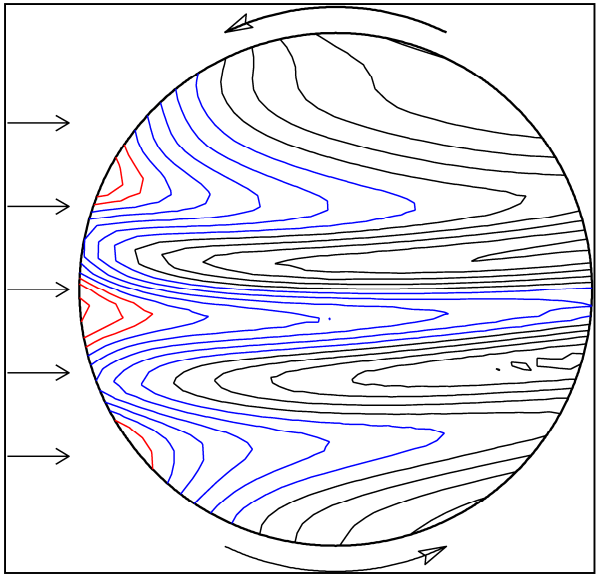
Initial Solution

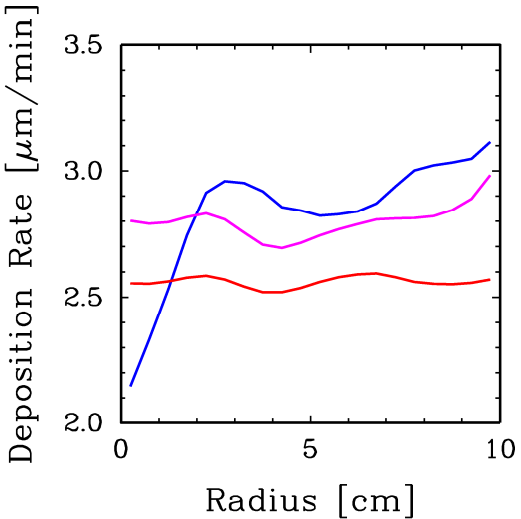


2-Parameter Opt.



4-Parameter Opt





Sandia National Laboratories

Summary of Results: SAND 4x faster then NAND, bigger difference with more params

Method	NAND	SAND	NAND	SAND
Params	2	2	4	4
Iterations	6	14	17	33
Jacobian Fills	120	15	391	34
Jacobian Solves	120	43	391	107
Total Time	**7.2hrs**	**2.2hrs**	**25.2hrs**	**6.2hrs**
Final Objective Function	5.0638	5.0635	0.6301	0.6338
P1: Side Inlets Velocity V_i	24.51	24.46	24.62	24.92
P2: Wafer Rotation Ω	31.26	31.25	28.18	28.55
P3: Purge Flow Velocity V_p	Set: 10.0		16.16	17.34
P4 : Side Inlets Mole Frac. X_i	Set: 0.024		0.018*	0.018*

*bound

Sandia National Laboratories

Comparison of More SAND and NAND Levels

$$\frac{df}{du} = \frac{\partial f}{\partial y}\left(-\frac{\partial c^{-1}}{\partial y}\frac{\partial c}{\partial u}\right) + \frac{\partial f}{\partial u}$$

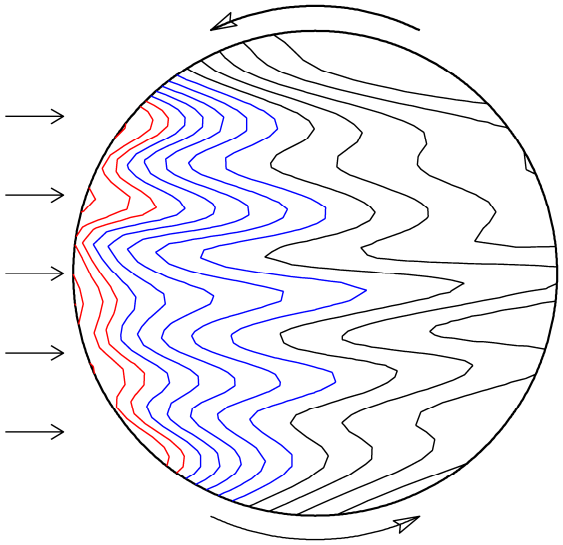
NAND (17 SQP iters)			SAND (33 SQP iters)		
Level 1 Black Box	Level 2 Direct	Level 3 Adjoint	Level 4 Direct	Level 5 Adjoint	Level 5^S
391	303	264	165	107	74

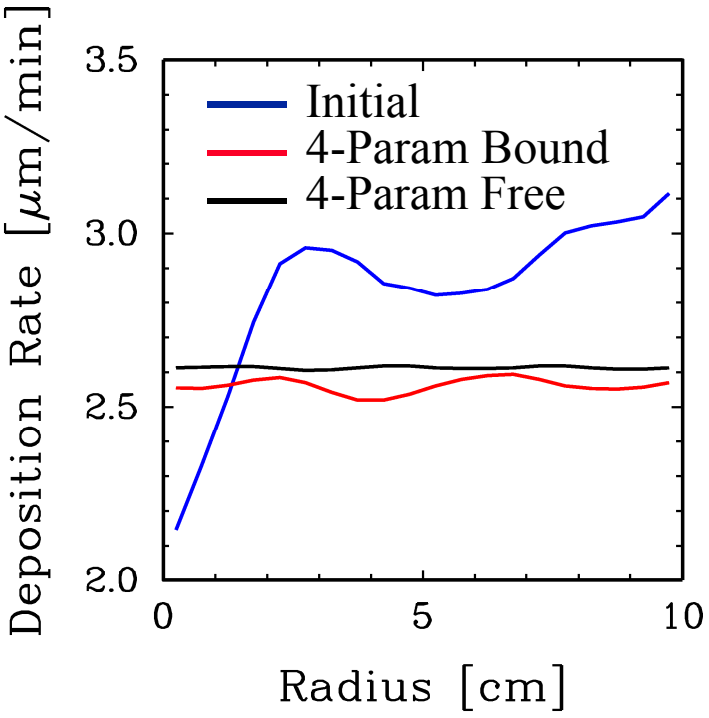
Linear Solves

Second order corrections with SAND can bridge the methods

Sandia National Laboratories

Freeing parameter improves uniformity considerably: 0.5% non-uniformity





	Bound	Free
Final Objective Function	0.6338	0.0207
P1: Side Inlets Velocity V_i	24.92	44.77
P2: Wafer Rotation Ω	28.55	31.87
P3: Purge Flow Velocity V_p	17.34	4.81
P4 : Side Inlets Mole Frac X_i	**1.80%***	1.08%

Sandia National Laboratories

Conclusions

- Stabilized FE methods can provide an effective formulation for a number of challenging transport / reaction applications.
- Scalable algorithms are required for efficient performance.
- Globalized Newton-Krylov methods are required to produce robust, accurate, and efficient codes.
- Stability, Bifurcation Analysis, and Optimization are critical tools for effective use of application codes.

The End

Sandia National Laboratories

Stopping Criteria
(StatusTests)

Highly Flexible Design: Users build a convergence test hierarchy and registers it with the solver (via solver constructor or reset method).

- Norm F: {Inf, One, Two} {absolute, relative} $\|F\| \leq \text{tol}$
- Norm Update ΔX: {Inf, One, Two} $\|x_k - x_{k-1}\| \leq \text{tol}$
- Norm Weighted Root Mean Square (WRMS):

$$C\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i^k - x_i^{k-1}}{\text{RTOL}|x_i^{k-1}| + \text{ATOL}_i}\right)^2} \leq \text{tol}$$

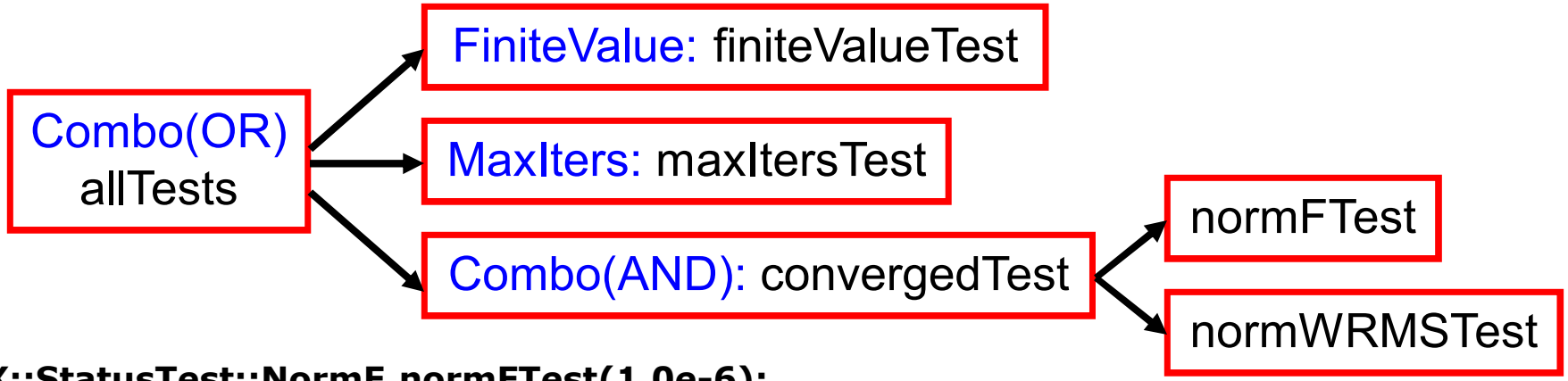
- Max Iterations: Failure test if solver reaches max # iters
- FiniteValue: Failure test that checks for NaN and Inf on $\|F\|$
- Stagnation: Failure test that triggers if the convergence rate fails a tolerance check for n consecutive iterations.

$$\frac{\|F_k\|}{\|F_{k-1}\|} \geq \text{tol}$$

- Combination: {AND, OR}
- Users Designed: Derive from NOX::StatusTest::Generic

Building a Status Test

- Converge if both: $\|F\| \leq 1.0E-6$ $\|\delta x\|_{WRMS} \leq 1.0$
- Fail if value of $\|F\|$ becomes Nan or Inf
- Fail if we reach maximum iterations




```
NOX::StatusTest::NormF normFTest(1.0e-6);
NOX::StatusTest::NormWRMS normWRMSTest();
NOX::StatusTest::Combo convergedTest(NOX::StatusTest::Combo::AND);
convergedTest.addStatusTest(normFTest);
convergedTest.addStatusTest(normWRMSTest);
NOX::StatusTest::FiniteValue finiteValueTest;
NOX::StatusTest::MaxIters maxItersTest(200);
NOX::StatusTest::Combo allTests(NOX::StatusTest::Combo::OR);
allTests.addStatusTest(finiteValueTest);
allTests.addStatusTest(maxItersTest);
allTests.addStatusTest(convergedTest);
```

Status Tests Continued

```
-- Final Status Test Results --
Converged....OR Combination ->
  Converged....AND Combination ->
    Converged....F-Norm = 3.567e-13 < 1.000e-08
                  (Length-Scaled Two-Norm, Absolute Tolerance)
    Converged....WRMS-Norm = 1.724e-03 < 1
                  (Min Step Size:  1.000e+00 >= 1)
                  (Max Lin Solv Tol:  4.951e-14 < 0.5)
  ??..........Finite Number Check (Two-Norm F) = Unknown
  ??..........Number of Iterations = -1 < 200
```

User Defined are Derived from NOX::StatusTest::Generic

NOX::StatusTest::StatusType **checkStatus**(const NOX::Solver::Generic &problem)

NOX::StatusTest::StatusType
checkStatusEfficiently(const NOX::Solver::Generic &problem,
NOX::StatusTest::CheckType checkType)

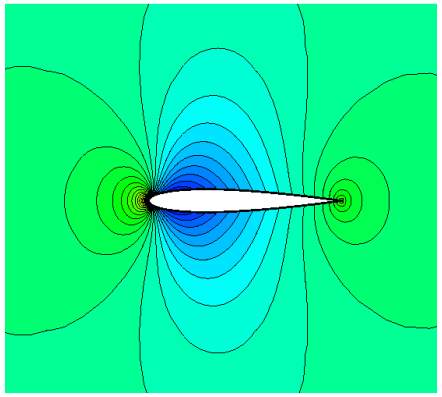
NOX::StatusTest::StatusType **getStatus**() const

ostream& **print**(ostream &stream, int indent=0) const

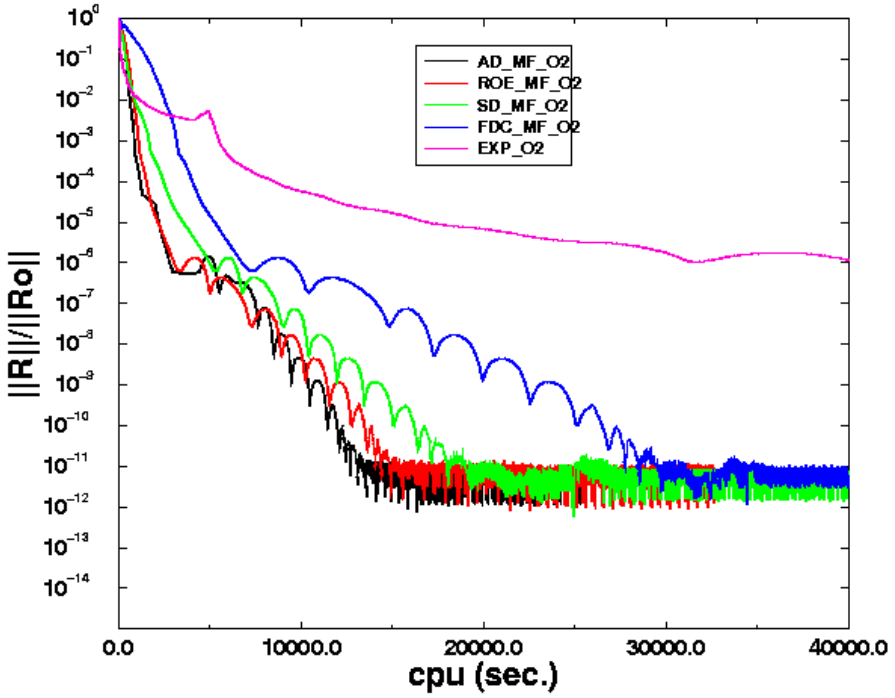
Interface Benefits

Premo – Compresible Flow

2D and 3D Finite Volume Code



NACA 0012 Airfoil



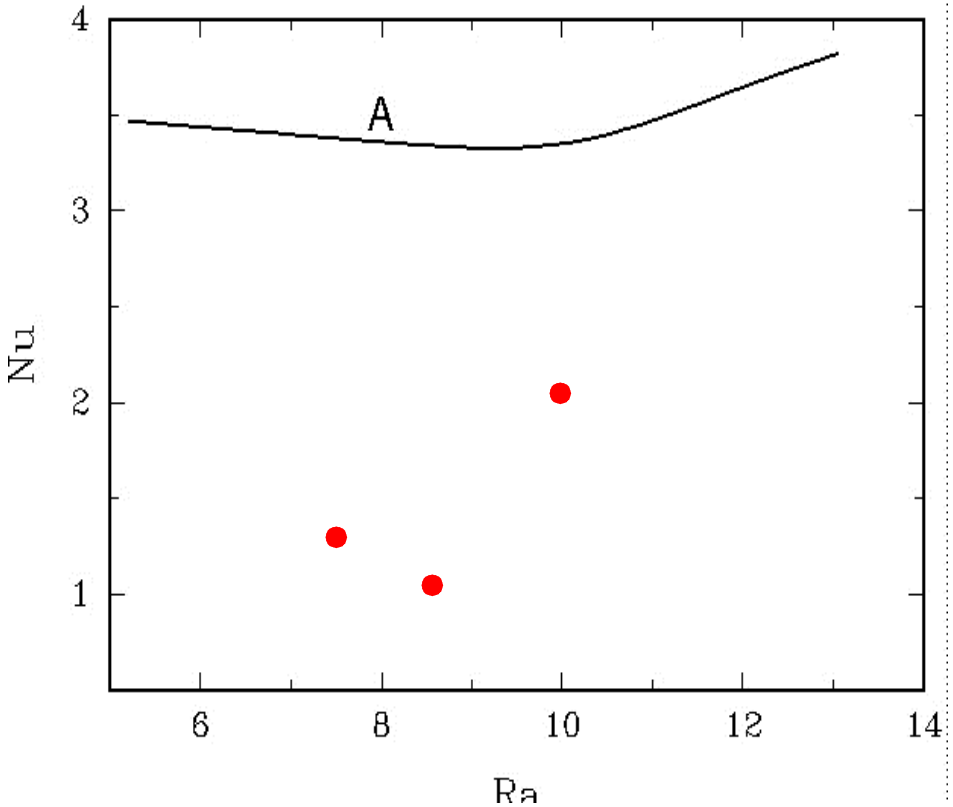
Method	Time	‖F‖
Explicit	**32.1 hrs**	**1.80E-10**
Peudo-Transient NOX (Colored FD)	**49.5 min**	**8.90E-13**
LOCA Homotopy NOX (Colored FD)	**34.4 min**	**6.50E-16**
LOCA Homotopy Automatic Differentiation	**14.2 min**	**6.50E-14**

Interfacing to LOCA took 1hr!

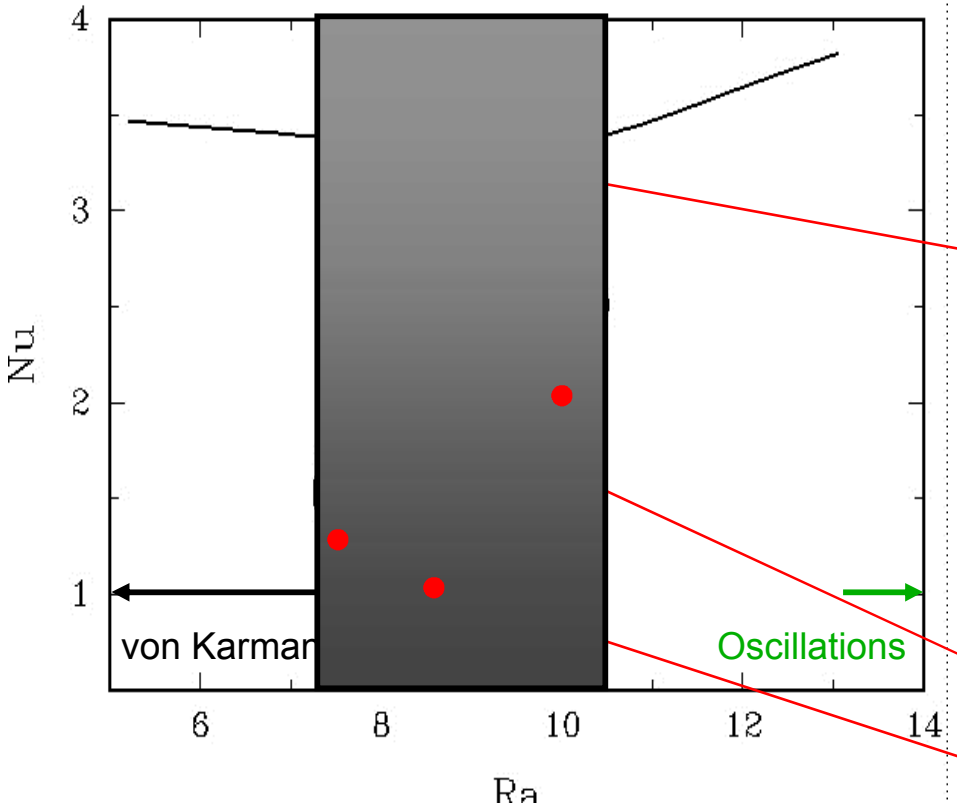
Sandia National Laboratories

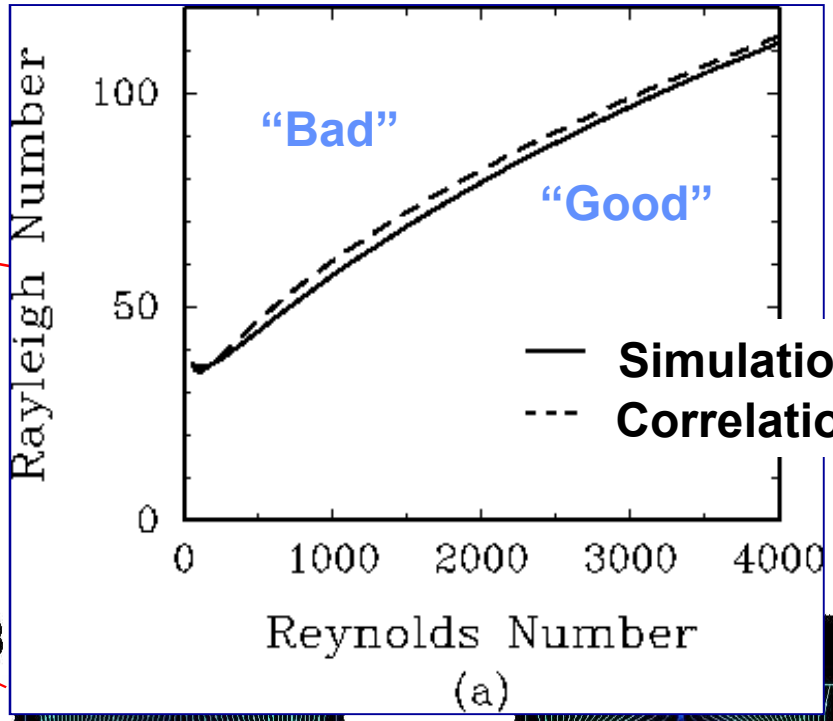
Bifurcation = Instability

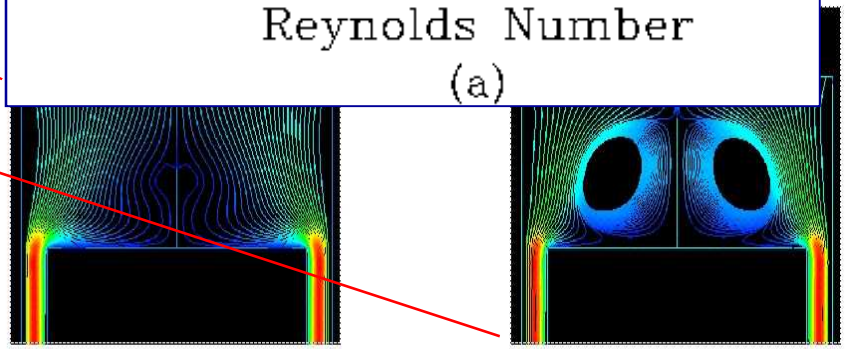
Erratic Operation could be due to physical Instability



Stability of Rotating Disk Reactor: Global stability limit and linear stability results







Fixed Re

LOCA - Salinger, Pawlowski, Phipps

Sandia National Laboratories