SAND2009-7538P

NOX: Nonlinear Object-oriented Solutions

Roger Pawlowski, Russell Hooper, Andrew Salinger, Eric Phipps, Tammy Kolda
Sandia National Laboratories

Trilinos User Group Meeting
November 3rd, 2009

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.

Motivation

- NOX is designed to provide robust and efficient solvers for sets of nonlinear equations
- NOX targets large-scale parallel computations but works just as well on serial problems
- Nonlinear problem definition:

Find $x_* \in \mathbb{R}^n$ such that $F(x_*) = 0$ where

$$F : \mathbb{R}^n \rightarrow \mathbb{R}^n$$

We define the Jacobian:

$$J(x) \in \mathbb{R}^{n \times n}$$

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

NOX Solver Basics

- Goals: ROBUST and EFFICIENT
- Use Newton's Method: $M(x) = F(x) + J\Delta x$

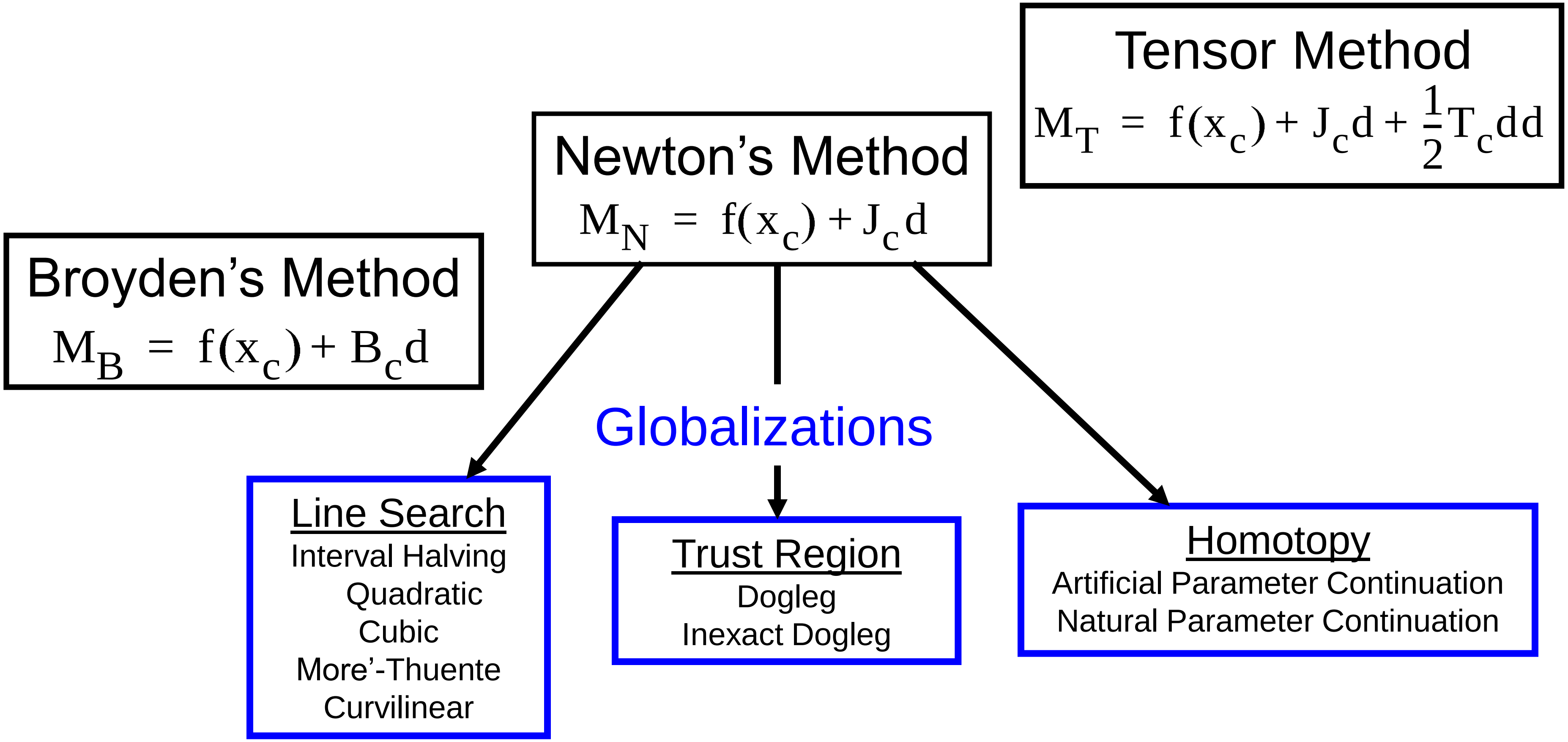
Until converged:

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

$$x_{i+1} = x_i + \Delta x$$

- Efficiency :
  - q-quadratic convergence rate
  - Fast linear system solver (iterative linear solvers, preconditioners)
- Robust solvers due to globalizations:
  - Line search
  - Trust region
  - Homotopy

Nonlinear Solution Strategies



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

Curvilinear

Trust Region

Dogleg

Inexact Dogleg

Homotopy

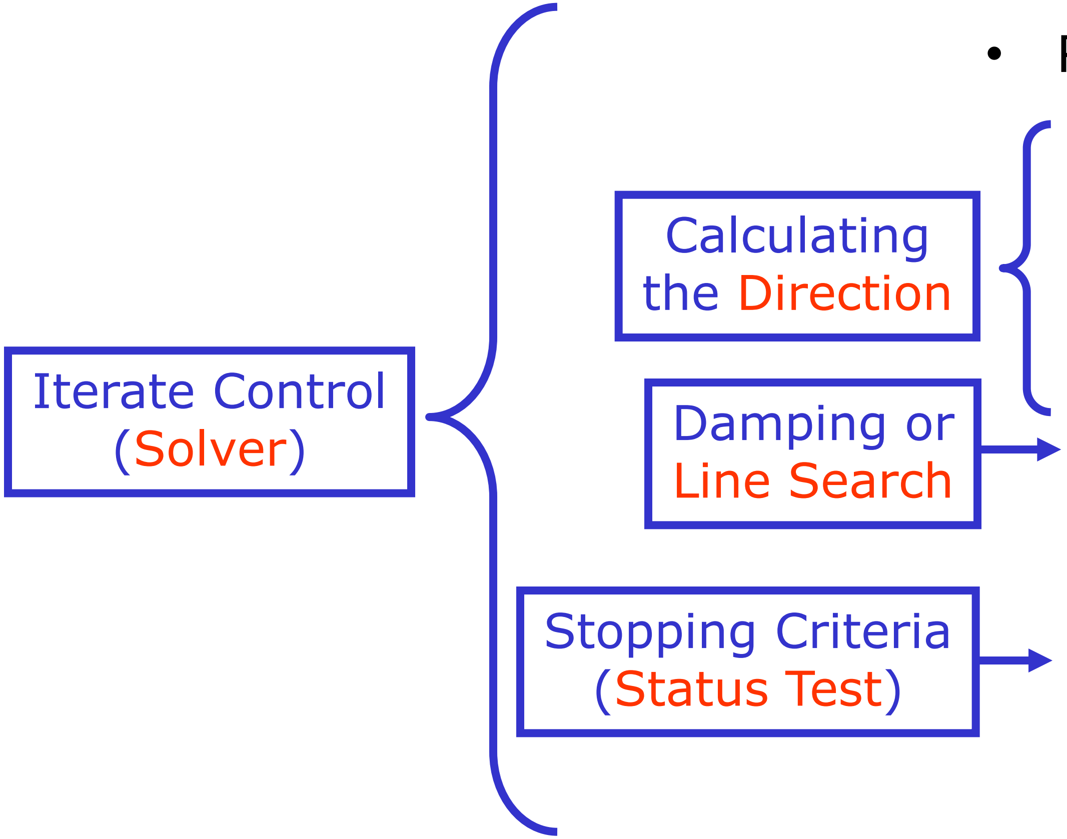
Artificial Parameter Continuation

Natural Parameter Continuation

Sandia National Laboratories

Building Blocks of NOX

Example: Newton's Method for $F(x) = 0$



Iterate Control (Solver)

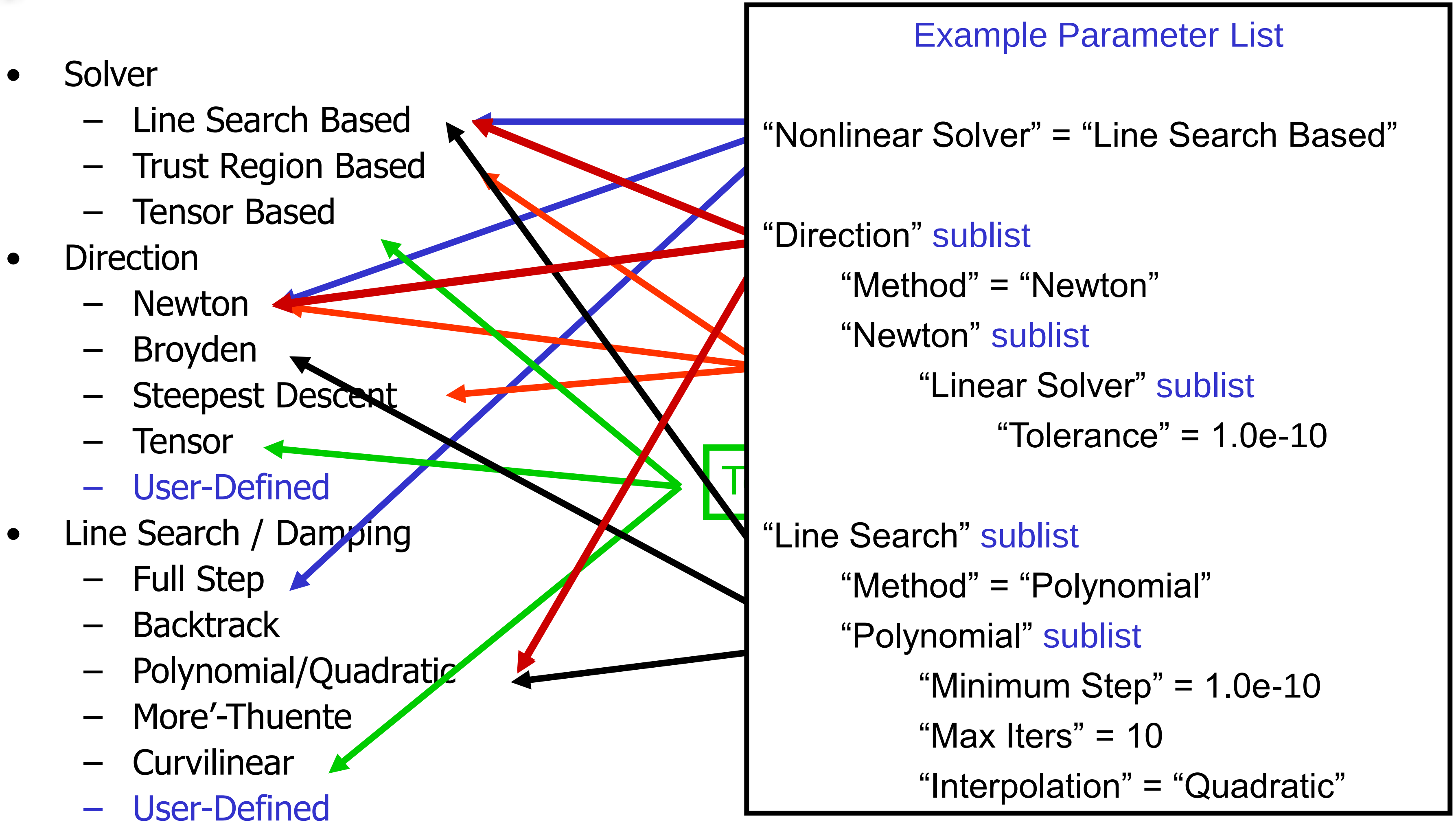
Calculating the Direction

Damping or Line Search

Stopping Criteria (Status Test)

- Choose an initial guess x_0
- For k = 0,1,2,...
  - Compute $F_k = F(x_k)$
  - Compute J_k where $(J_k)_{ij} = \partial F_i(x_k)/\partial x_j$
  - Let $d_k = -J_k^{-1} F_k$
  - (Optional) Let λ_k be a calculated step length
  - Set $x_{k+1} = x_k + \lambda_k d_k$
  - Test for Convergence or Failure

Mix-n-Match Solver Algorithms



- Solver
  - Line Search Based
  - Trust Region Based
  - Tensor Based
- Direction
  - Newton
  - Broyden
  - Steepest Descent
  - Tensor
  - User-Defined
- Line Search / Damping
  - Full Step
  - Backtrack
  - Polynomial/Quadratic
  - More'-Thuente
  - Curvilinear
  - User-Defined

Example Parameter List

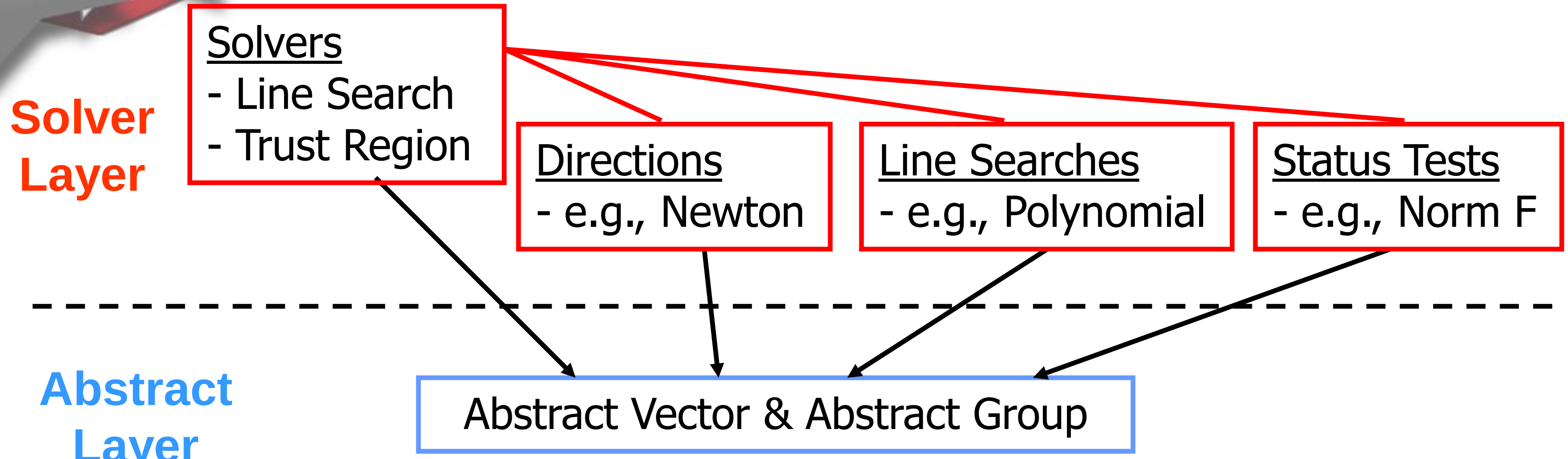
"Nonlinear Solver" = "Line Search Based"

"Direction" sublist
    "Method" = "Newton"
    "Newton" sublist
        "Linear Solver" sublist
            "Tolerance" = 1.0e-10

"Line Search" sublist
    "Method" = "Polynomial"
    "Polynomial" sublist
        "Minimum Step" = 1.0e-10
        "Max Iters" = 10
        "Interpolation" = "Quadratic"

NOX Documentation: Related Pages

NOX Framework



Solver Layer

Solvers
- Line Search
- Trust Region

Directions
- e.g., Newton

Line Searches
- e.g., Polynomial

Status Tests
- e.g., Norm F

Abstract Layer

Abstract Vector & Abstract Group

- Don’t need to directly access the vector or matrix entries, only manipulate the objects!

Abstract Vector

- Init
- Abs
- Scale
- Update
- Norm
- Dot
- Clone

Abstract Group

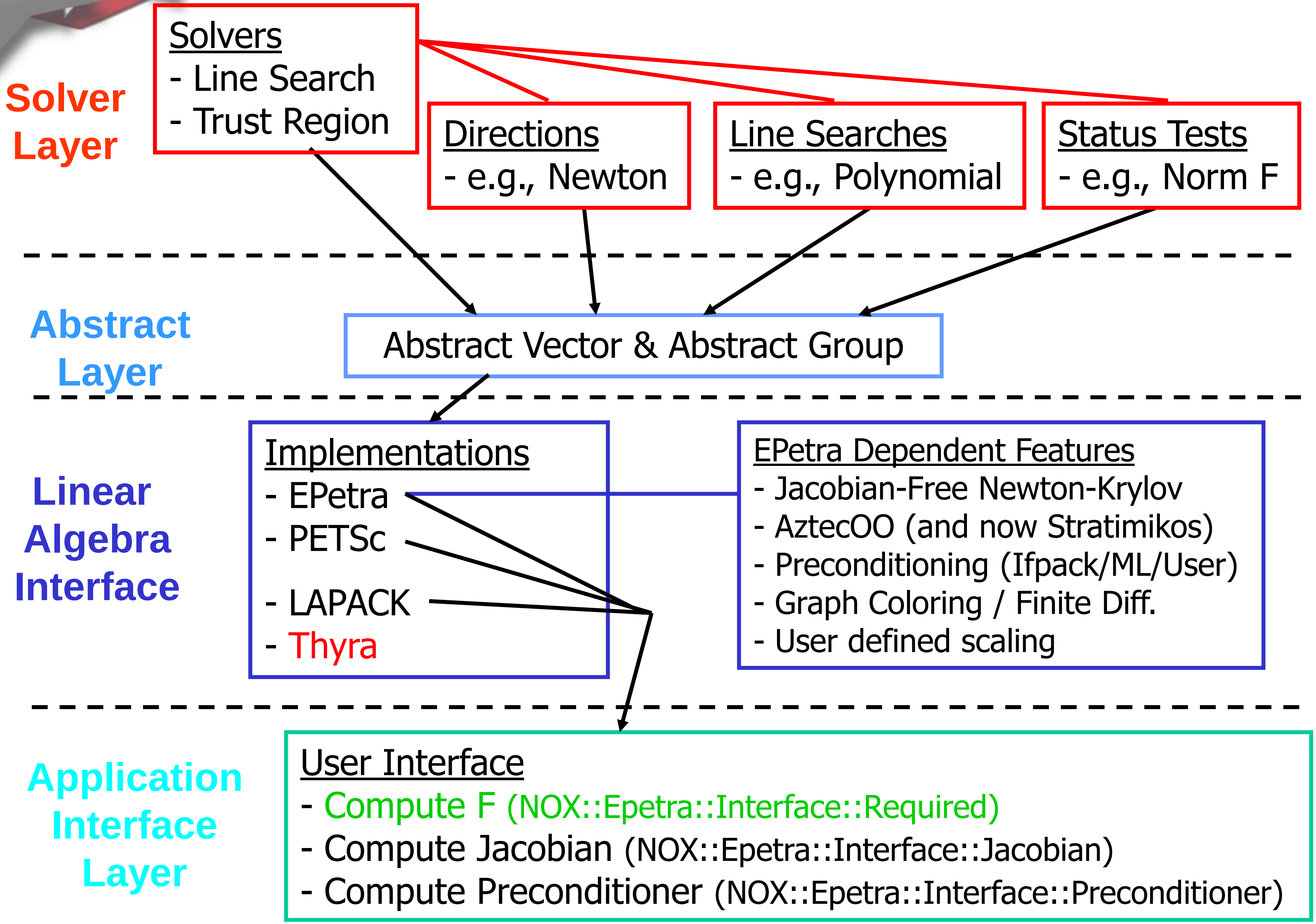
- Compute F
- Compute J
- Compute Grad F
- Clone
- apply J
- apply J^T
- apply J^{-1}

Wraps user interface
Stores: x, F, J

- Implementation is independent of the linear algebra storage format and parallel services.

NOX Framework

Online Docs



Solver Layer

Solvers
- Line Search
- Trust Region

Directions
- e.g., Newton

Line Searches
- e.g., Polynomial

Status Tests
- e.g., Norm F

Abstract Layer

Abstract Vector & Abstract Group

Linear Algebra Interface

Implementations
- EPetra
- PETSc
- LAPACK
- Thyra

EPetra Dependent Features
- Jacobian-Free Newton-Krylov
- AztecOO (and now Stratimikos)
- Preconditioning (Ifpack/ML/User)
- Graph Coloring / Finite Diff.
- User defined scaling

Application Interface Layer

User Interface
- Compute F (NOX::Epetra::Interface::Required)
- Compute Jacobian (NOX::Epetra::Interface::Jacobian)
- Compute Preconditioner (NOX::Epetra::Interface::Preconditioner)

Thyra Support

- Eventually all other support should be deprecated for Thyra support
  - Use Stratimikos to build Linear solver/Preconditioner combinations
- Only requires a Thyra (or Epetra) Model Evaluator

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):

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

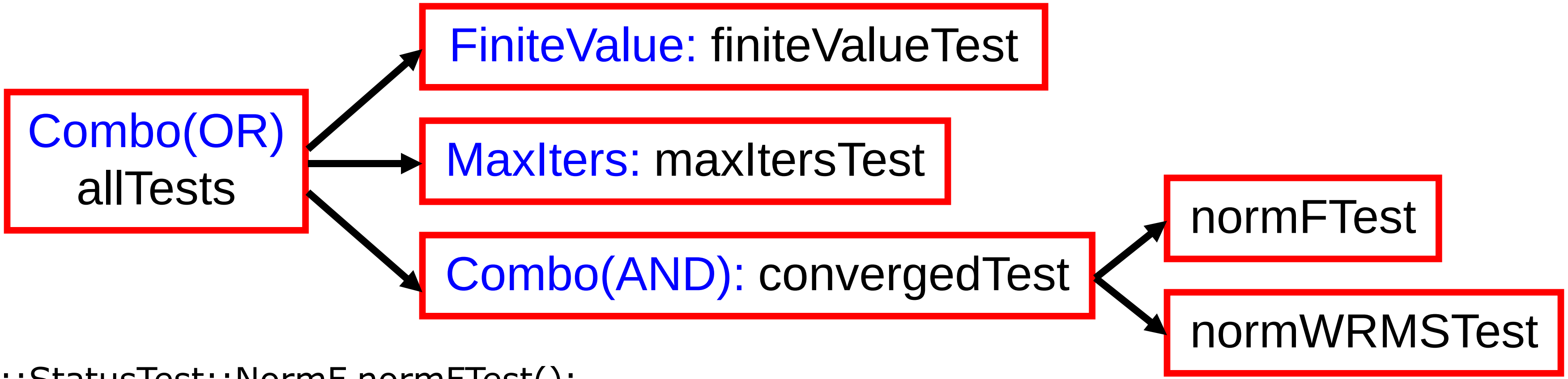
- Max Iterations: Failure test if solver reaches max # iters
- FiniteValue: Failure test that checks for NaN and Inf on $\|F\|$
- Stagnation: Failure test that triggers if the convergence rate fails a tolerance check for n consecutive iterations.

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

- 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



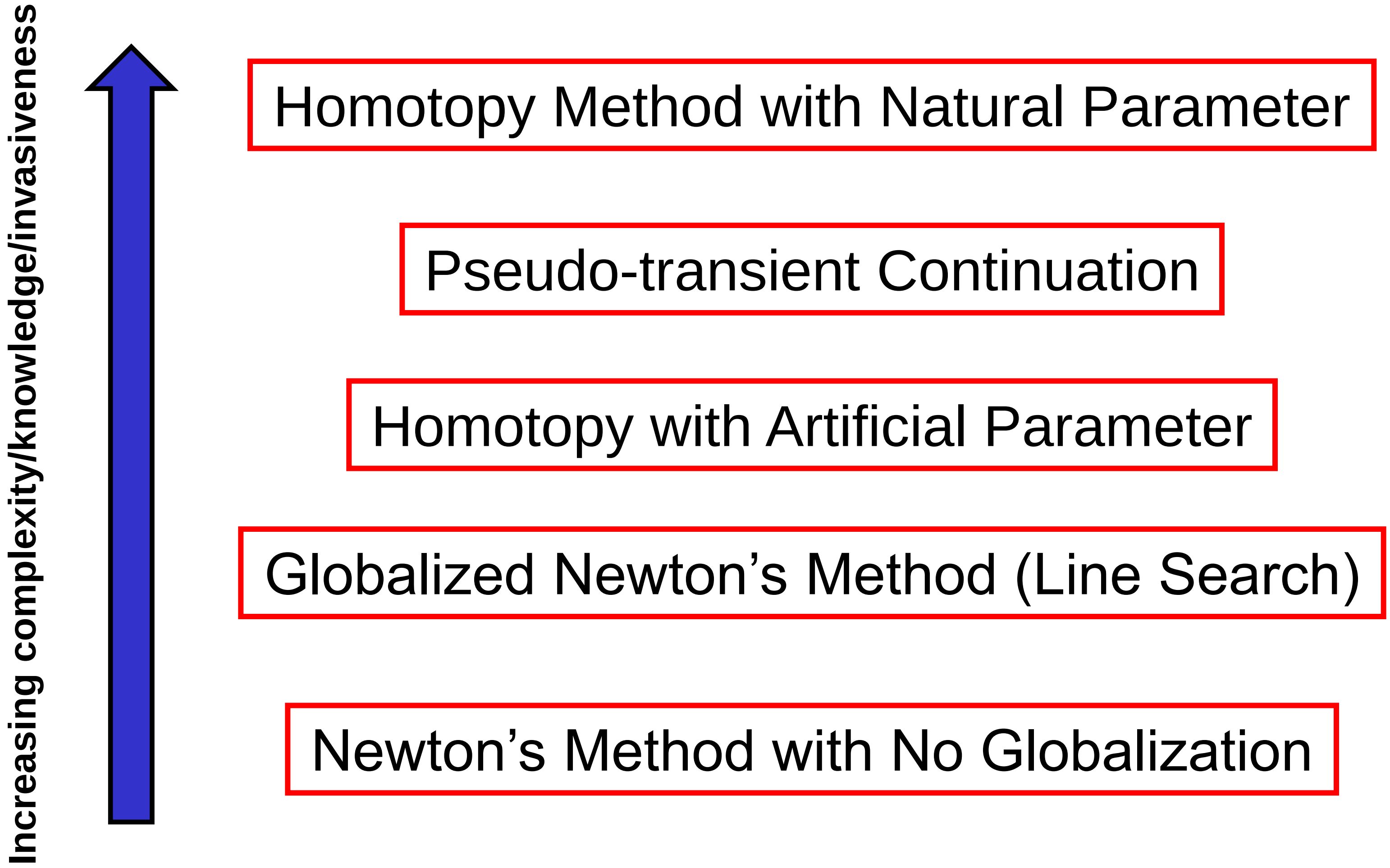
```
NOX::StatusTest::NormF normFTest();
NOX::StatusTest::NormWRMS normWRMSTest();
NOX::StatusTest::Combo convergedTest(NOX::StatusTest::Combo::AND);
convergedTest.addStatusTest(normFTest);
convergedTest.addStatusTest(normWRMSTest);
NOX::StatusTest::FiniteValue finiteValueTest;
NOX::StatusTest::MaxIters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

Status Tests Continued

```
-- Status Test Results --
**...........OR Combination ->
  **...........AND Combination ->
    **...........F-Norm = 5.907e-01 < 1.000e-08
                 (Length-Scaled Two-Norm, Absolute Tolerance)
    **...........WRMS-Norm = 4.794e+01 < 1
                 (Min Step Size:  1.000e+00 >= 1)
                 (Max Lin Solv Tol:  1.314e-15 < 0.5)
  **...........Finite Number Check (Two-Norm F) = Finite
  **...........Number of Iterations = 2 < 200
```

```
-- 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
```

Recommendations for Robust Solves



Increasing complexity/knowledge/invasiveness

Homotopy Method with Natural Parameter

Pseudo-transient Continuation

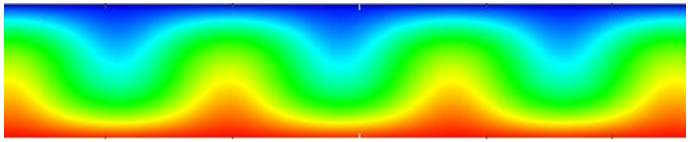
Homotopy with Artificial Parameter

Globalized Newton’s Method (Line Search)

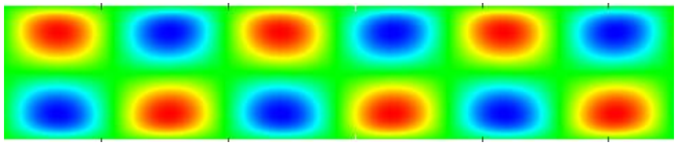
Newton’s Method with No Globalization

Hydro-Magnetic Rayleigh-Bernard Stability: Direct Determination of Nonlinear Equilibrium Solutions (Steady State Solves, Ra=2500, Q=4)

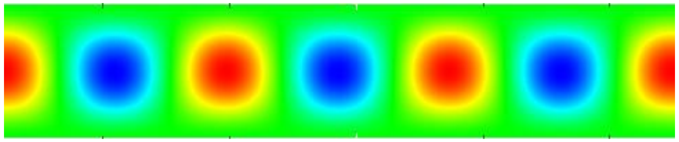
Temp



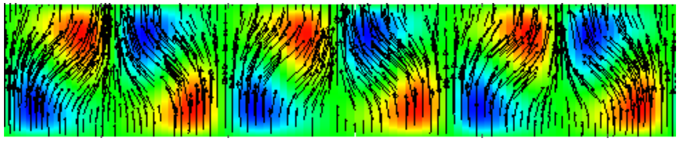
Vx



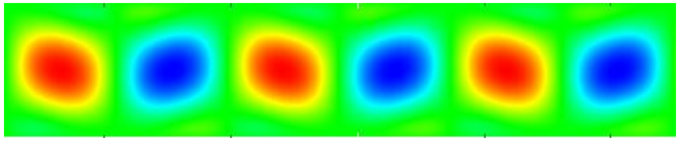
Vy



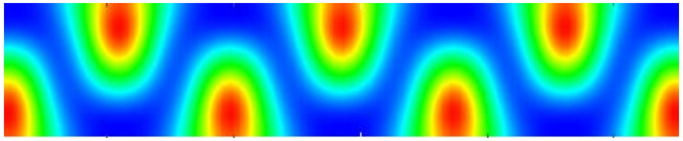
Jz

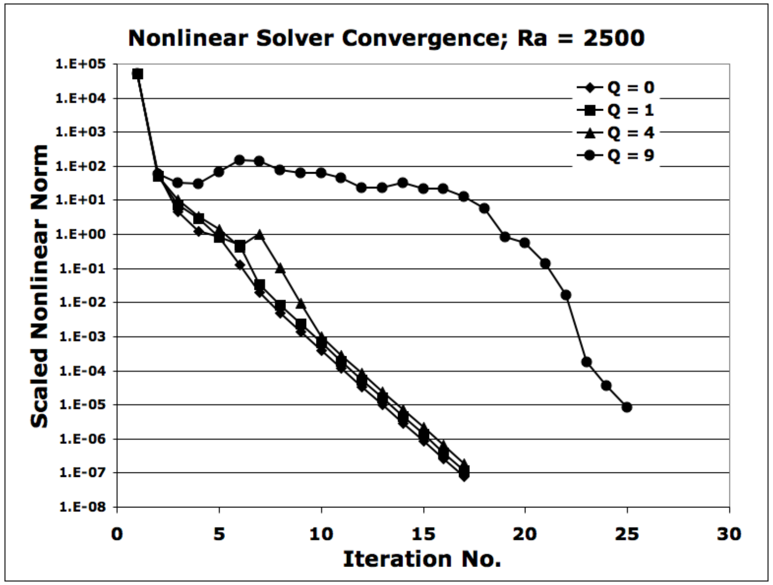


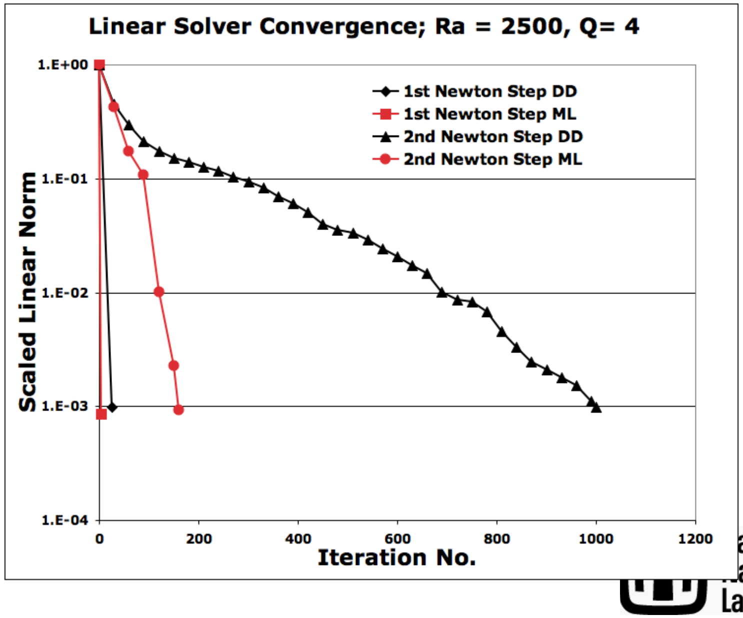
Bx



By







Observations/Suggestions

- THERE IS NO SILVER BULLET for nonlinear solvers!
  - Useful to throw a lot of methods at a problem => NOX!!
  - One interface -> Many methods
  - Users can supply their own solvers, directions, and line searches

- Where to start:
  - **Best examples in: trilinos/packages/nox/test/epetra/1Dfem**

- Largest problem to date: 1 Billion unknowns on 24,000 cores (Cray XT3/4).

Selected References

- *Globalization Techniques for Newton–Krylov Methods and Applications to the Fully Coupled Solution of the Navier–Stokes Equations*
  Roger P. Pawlowski, John N. Shadid, Joseph P. Simonis, and Homer F. Walker
  SIAM Rev. 48 700 (2006)
- *Inexact Newton dogleg methods*
  Pawlowski, RP ; Simonis, JP ; Walker, HF ; Shadid, JN
  SIAM JOURNAL ON NUMERICAL ANALYSIS (2007) Vol.46, iss.4, p.2112-2132
- *Large-scale stabilized FE computational analysis of nonlinear steady-state transport/reaction systems*
  Shadid, J.N. ; Salinger, A.G. ; Pawlowski, R.P. ; Lin, P.T. ; Hennigan, G.L. ; et al.
  Computer Methods in Applied Mechanics and Engineering (Feb 15 2006) Vol.195, iss.13-16, p.1846-1871