A Residual Smoothing Strategy for Accelerating Newton Method Continuation

Dimitri Mavriplis

University of Wyoming

Motivation

- Newton-Krylov methods have become popular for solving difficult/stiff CFD problems
  - Krylov methods provide robust linear system convergence
  - Newton method provides quadratic convergence enabling convergence to low residual tolerances
- Newton methods require continuation for most problems
- Most of the time spent for solving CFD problems is spent in the continuation process
- Continuation methods can stall due to local effects
  - “Unbalanced nonlinearities”
  - Attempts made to break up into smaller nonlinear problems
    - ASPIN, RASPIN

Newton Method

- To solve: $R(w) = 0$
- Linearize to get Jacobian dR/dw
- Take Newton steps as:

$$[dR(w^n)/dw^n]\,\Delta w^n = -\,R(w^n)$$

$$w^{n+1} = w^n + \alpha\,\Delta w^n$$

with $0 < \alpha < 1$ as determined by (backtracking) line search to minimize $||R(w^{n+1})||_2$

Pseudo-Transient Continuation Newton Method

- Introduce pseudo-time term and solve

$$[M\,(w^{n+1} - w^n) / \Delta\tau + R(w^{n+1}) = R_t(w^{n+1}) = 0$$

- Take Newton steps as:

$$[M/\Delta\tau + dR(w^n)/dw^n]\,\Delta w^n = -\,R(w^n)$$

$$w^{n+1} = w^n + \alpha\,\Delta w^n$$

with $0 < \alpha < 1$ as determined by (backtracking) line search to minimize $||R_t(w^{n+1})||_2$

M is a suitable mass matrix

$\Delta\tau$ is the pseudo time step (local time step = CFL $\Delta\tau_{explicit}$)

Pseudo-Transient Continuation Newton Method

- Introduce pseudo-time term and solve

$$[M (w^{n+1} - w^n) / \Delta\tau + R(w^{n+1}) = R_t(w^{n+1}) = 0$$

- Take Newton steps as:

$$[M/\Delta\tau + dR(w^n)/dw^n]\, \Delta w^n = -\, R(w^n)$$

$$w^{n+1} = w^n + \alpha\, \Delta w^n$$

with $0 < \alpha < 1$ as determined by (backtracking) line search to minimize $||R_t(w^{n+1})||_2$

Note: $R(w^n) = R_t(w^n)$ ...... but $R(w^{n+1}) \neq R_t(w^{n+1})$

Pseudo-Transient Continuation Newton Method

- Introduce pseudo-time term and solve

  $[M (w^{n+1} – w^n) / \Delta\tau + R(w^{n+1}) = R_t(w^{n+1})=0$

- Take Newton steps as:

$$[M/\Delta\tau + dR(w^n)/dw^n] \Delta w^n = - R(w^n)$$

$$w^{n+1} = w^n + \alpha \Delta w^n$$

with $0 < \alpha < 1$ as determined by (backtracking) line search to minimize $||R_t(w^{n+1})||_2$

Δw is guaranteed to be a descent direction for $||R_t||_2$ provided $[M/\Delta\tau + dR/dw]$ is an exact linearization of R_t

Pseudo-Transient Continuation Newton Method

$$[M/\Delta\tau + dR(w^n)/dw^n]\,\Delta w^n = -\,R(w^n)$$

- Limit as $\Delta t >> 1$: Recover Newton scheme

$$[dR(w^n)/dw^n]\,\Delta w^n = -\,R(w^n)$$

- Limit as $\Delta\tau << 1$: Recover point explicit scheme

$$[M/\Delta\tau]\,\Delta w^n = -\,R(w^n) \quad \text{or} \quad \Delta w^n = -\,\Delta\tau\,R(w^n)$$

M is simply cell volume for finite-volume scheme and is absorbed in $\Delta\tau$ above for simplicity

Pseudo-Transient Controller

- Magnitude of $\Delta\tau$ (or CFL) controlled by success/failure of line search
  - Initial CFL ~ 1
  - Line search result: $\alpha = 1$ CFL = CFL * 1.5
  - Line search result: $\alpha < 0.1$ CFL = CFL / 10
  - Otherwise CFL = constant

  - Common failure mode: CFL $\rightarrow$ 0

$$\Delta w^n = - \Delta\tau \, R(w^n) \quad \text{also} \rightarrow 0$$

- Observation:
  - Common local nonlinear smoothers (block Jacobi, line Jacobi, Gauss-Seidel) have no difficulties reducing residuals in cases where PTC fails in above mode
  - Explicit scheme is poor choice for anisotropic problems (line smoothers preferred)

Desired Behavior

good for rapid initial residual reduction

- In the limit CFL << 1

  $\Delta w^n = - D^{-1} R(w^n)$

  – where D is some preconditioner/smoother
    - possibly nonlinear
    - Independent of CFL or $\Delta\tau$

- Possible formulation:

$$\left[\alpha(\Delta\tau) D + \beta(\Delta\tau) \frac{\partial R}{\partial w} \right] \Delta w^n = -R(w^n)$$

with, for example....

$$\alpha(\Delta\tau) = \frac{1}{1+\Delta\tau} \qquad \beta(\Delta\tau) = \frac{\Delta\tau}{1+\Delta\tau}$$

Still recovers Newton scheme for $\Delta\tau >> 1$

Disadvantages

$$\left[\alpha(\Delta\tau) D + \beta(\Delta\tau) \frac{\partial R}{\partial w} \right] \Delta w^n = -R(w^n)$$

- Left-hand side matrix is modified
  - May require modification of linear solver techniques especially for intermediate values of $\Delta\tau$
- Left-hand side matrix is no longer exact linearization of RHS
  - Descent direction for line search not guaranteed

Alternate Approach

- Leave LHS (Jacobian) unchanged
- Modify RHS as:

$$\left[\frac{M}{\Delta\tau}+\frac{\partial R}{\partial w}\right]\Delta w^{n}=-R(w^{n})-D^{-1}\frac{M}{\Delta\tau}R(w^{n})$$

  - For Δτ << 1: $$\frac{M}{\Delta\tau}\Delta w^{n}=-D^{-1}\frac{M}{\Delta\tau}R(w^{n})$$
  - For Δτ >> 1: $$\frac{\partial R}{\partial w}\Delta w^{n}=-R(w^{n})$$

Alternate Approach

- Leave LHS (Jacobian) unchanged
- Modify RHS as:

$$\left[\frac{M}{\Delta\tau}+\frac{\partial R}{\partial w}\right]\Delta w^n = -R(w^n) - D^{-1}\frac{M}{\Delta\tau}R(w^n)$$

  - For $\Delta\tau \ll 1$: $\cancel{\frac{M}{\Delta\tau}}\Delta w^n = -D^{-1}\cancel{\frac{M}{\Delta\tau}}R(w^n)$
  - For $\Delta\tau \gg 1$: $\frac{\partial R}{\partial w}\Delta w^n = -R(w^n)$

Residual Smoothing Interpretation

$$\left[\frac{M}{\Delta\tau} + \frac{\partial R}{\partial w}\right]\Delta w^n = -R(w^n) - D^{-1}\frac{M}{\Delta\tau}R(w^n)$$

- $D^{-1}M/\Delta\tau$ is a non-dimensional operator with a non-trivial stencil (due to D^{-1})
- RHS may be interpreted as a smoothed residual vector

$$\left[\frac{M}{\Delta\tau} + \frac{\partial R}{\partial w}\right]\Delta w^n = -\underbrace{\left[I + D^{-1}\frac{M}{\Delta\tau}\right]}_{\textit{smoothing operator}}R(w^n) = R_{sm}(w^n)$$

Residual Smoothing Advantages

$$\left[\frac{M}{\Delta\tau}+\frac{\partial R}{\partial w}\right]\Delta w^n = -\left[I + D^{-1}\frac{M}{\Delta\tau}\right]R(w^n) = R_{sm}(w^n)$$

– Simple to implement:
  - Add precomputed correction $\Delta w=-D^{-1}R(w)$ to RHS and scale by $M/\Delta\tau$
– LHS Jacobian is unchanged from original scheme
  - Make use of existing linear solvers
– LHS Jacobian is exact linearization of RHS
  - Line search descent direction is guaranteed

$$R_{sm}(w^{n+1}) = \frac{M}{\Delta\tau}(w^{n+1}-w^n) + R(w^{n+1}) + D^{-1}\frac{M}{\Delta\tau}R(w^n)$$

Smoothing term always evaluated at w^n
(vanishes in linearization)

Residual Smoothing Advantages

- Line search minimizes $||R_{sm}||_2$ instead of $||R_t||_2$
- For $\Delta\tau$ >>1 these are the same
- For $\Delta\tau$ << 1 Line search usually takes full update since we have:

$$R_{sm}(w^{n+1}) \sim \frac{M}{\Delta\tau}(\Delta w^n) + R(w^{n+1}) + D^{-1}\frac{M}{\Delta\tau}R(w^n)$$

small wrt to other terms

– and the the solution $\Delta w^n = -D^{-1}R(w^n)$ implies $R_{sm}(w^n+\Delta w^n) \sim 0$

Generalization and Implementation

$$\left[\frac{M}{\Delta\tau}+\frac{\partial R}{\partial w}\right]\Delta w^n = -R(w^n) - D^{-1}\frac{M}{\Delta\tau}R(w^n)$$

– Implement by adding precomputed update as source term on RHS: $\Delta w^{sm} = -D^{-1}R(w^n)$

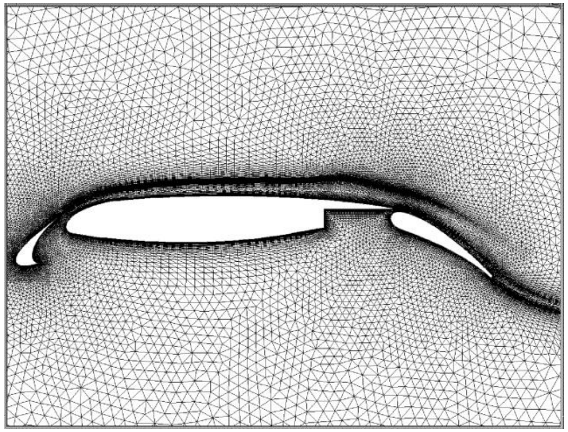
  – and rescale by $M/\Delta\tau$

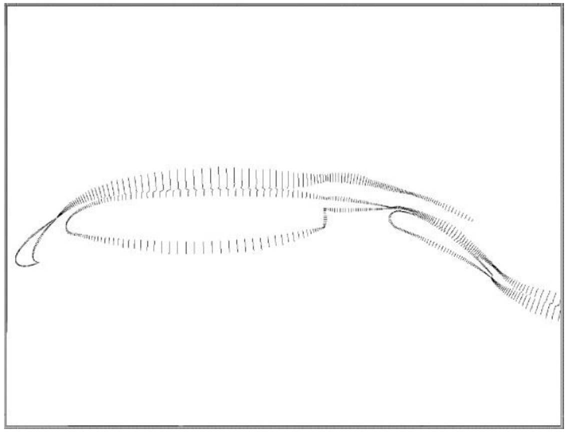
– In practice Δw^{sm} can be the result of any sequence of nonlinear smoothing operations

  - Multistage Runge-Kutta designed for smoothing (Jameson 1981)
  - Any number of nonlinear (FAS) multigrid cycles

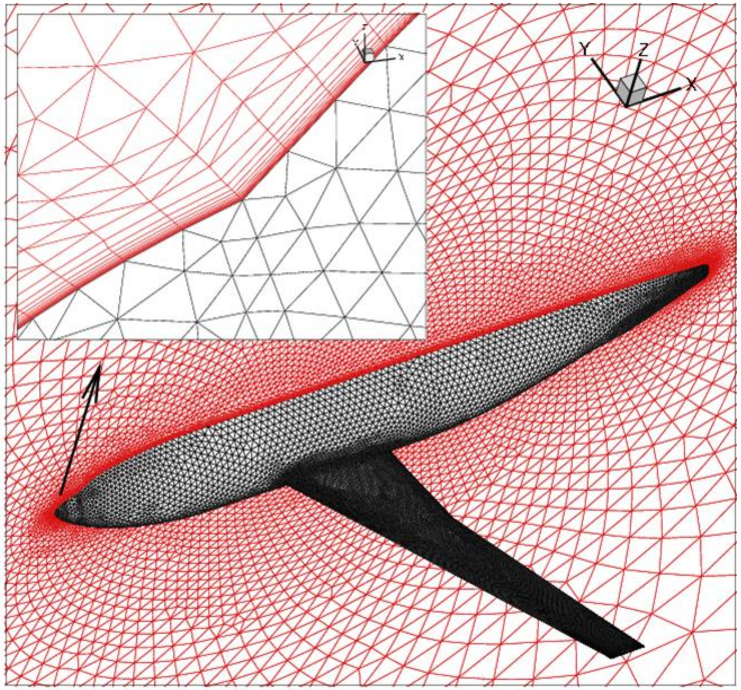
Results

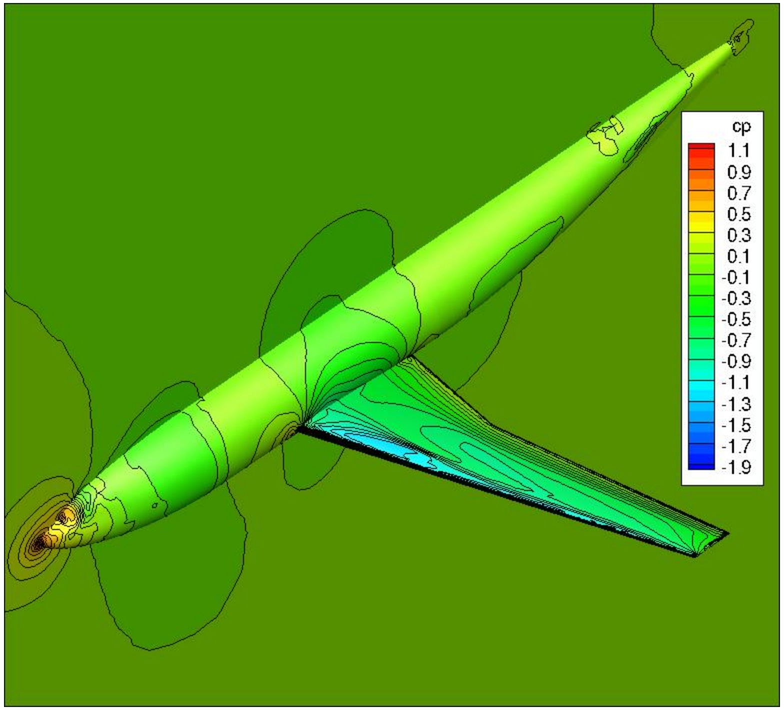
- Implemented in unstructured mesh CFD code NSU3D
    - Highly anisotropic meshes in near wall region
    - Extract line structures for implicit line solve
  - Nonlinear solver:
    - 3 stage line-implicit Runge-Kutta
    - Used as solver, or smoother for agglomeration Multigrid
  - Newton-Krylov Solver
    - Pseudo-transient continuation with line search and CFL controller
    - Linear system solved by linear MG: Linear residual reduction = 0.01
    - Original version (unsmoothed)
    - Smoothed version: 5 cycles of 3-stage line RK to compute smoothing term





Results: Test Case 1

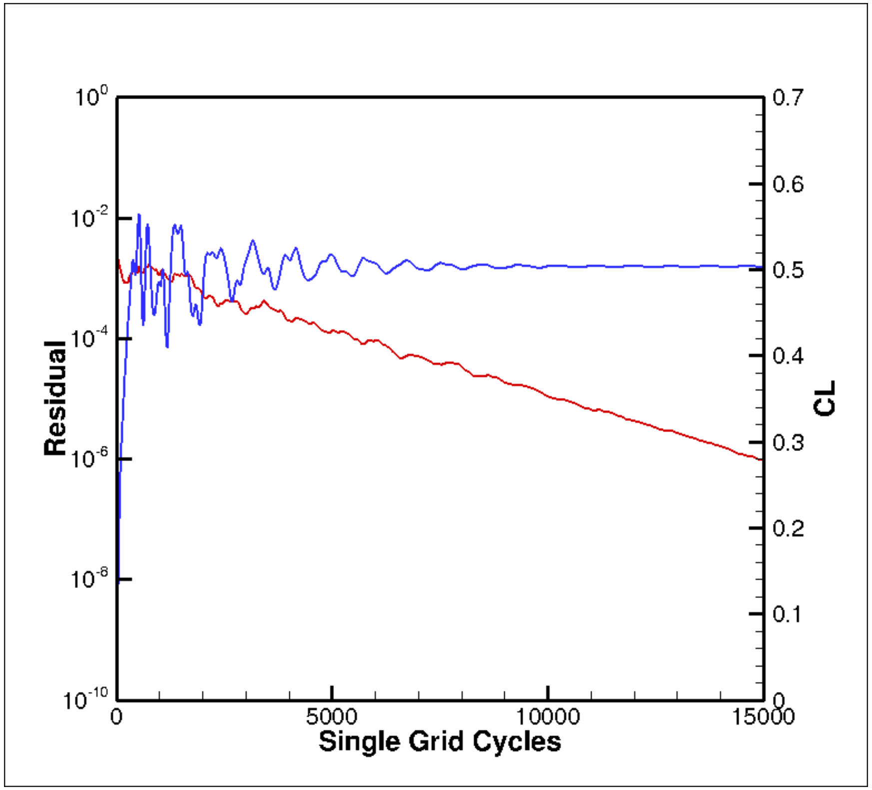




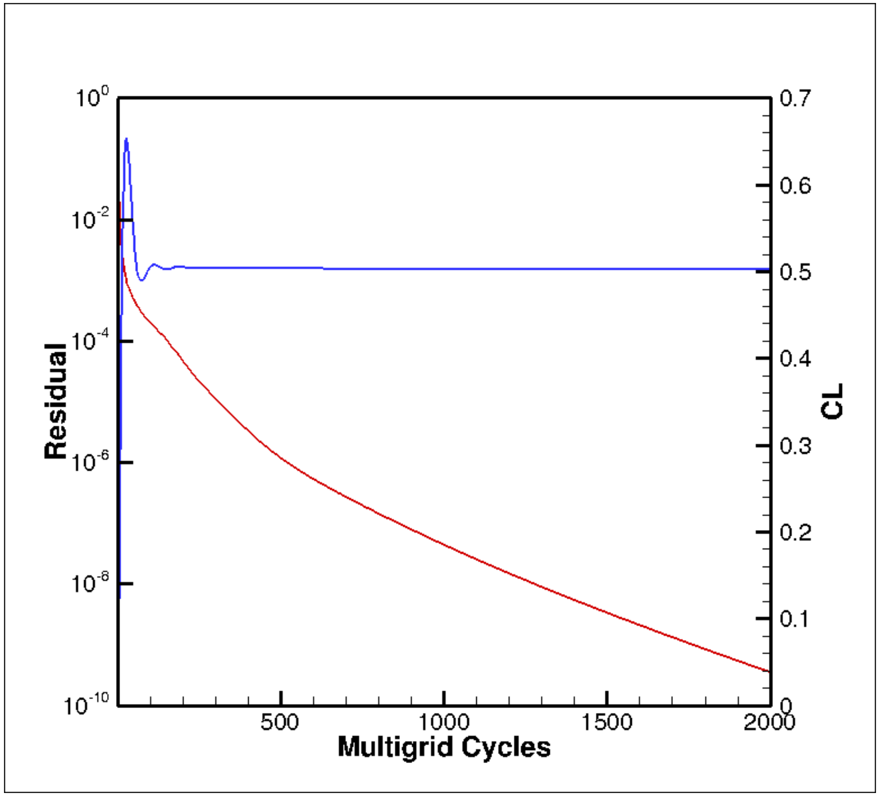
- Transonic flow over wing-body configuration
- Solution of Reynolds-Averaged Navier-Stokes Equations (RANS):
  - 2nd order finite-volume
  - Mach=0.75, Incidence=0°, Re=3 million, Spalart-Allmaras Turbulence model
  - 1.2 million point mesh (mixed tets, prisms)
  - Highly anisotropic (1:10,000) near wall

Convergence of Nonlinear Solvers

- 3 stage line-implicit Runge Kutta smoother



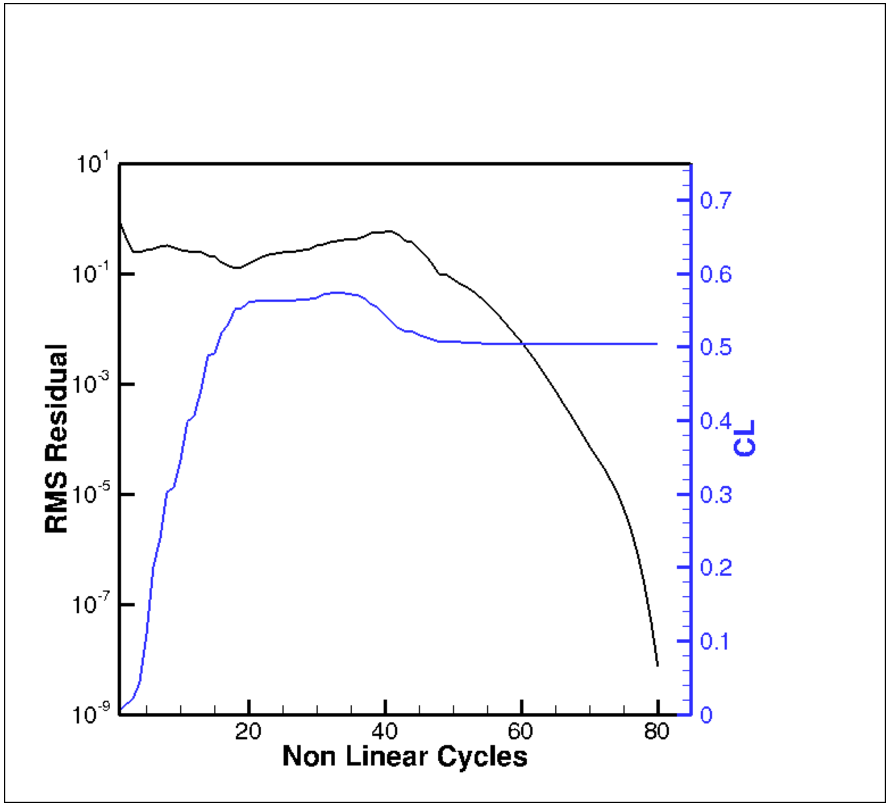
Single Grid Solver

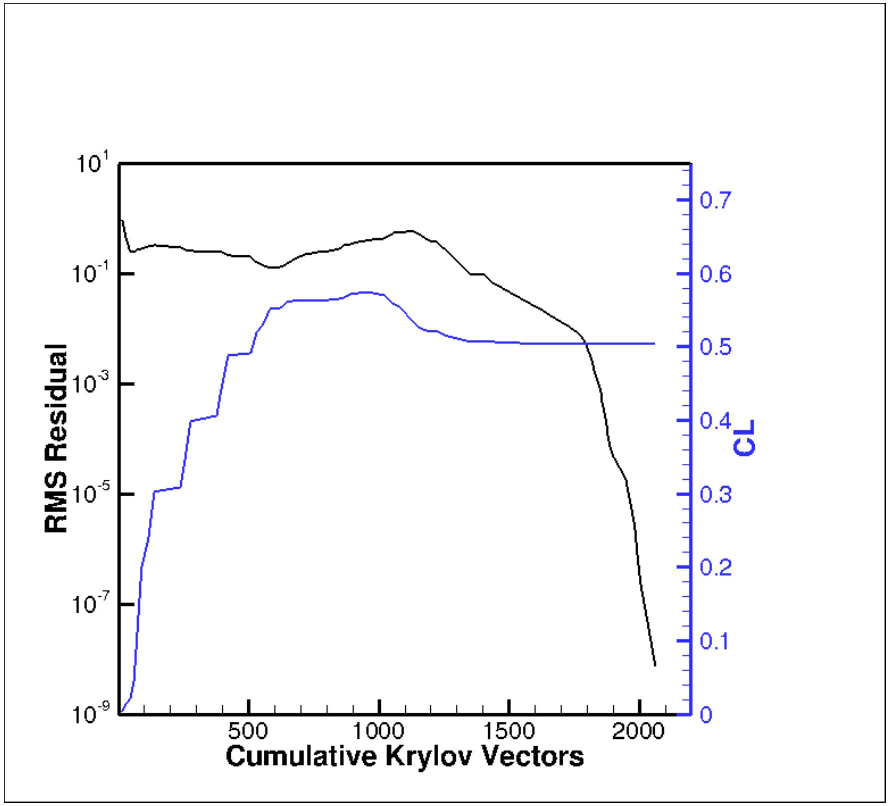


4-Level FAS Multigrid Solver

- Relatively monotone convergence in both cases
- As expected, multigrid solver 10X faster

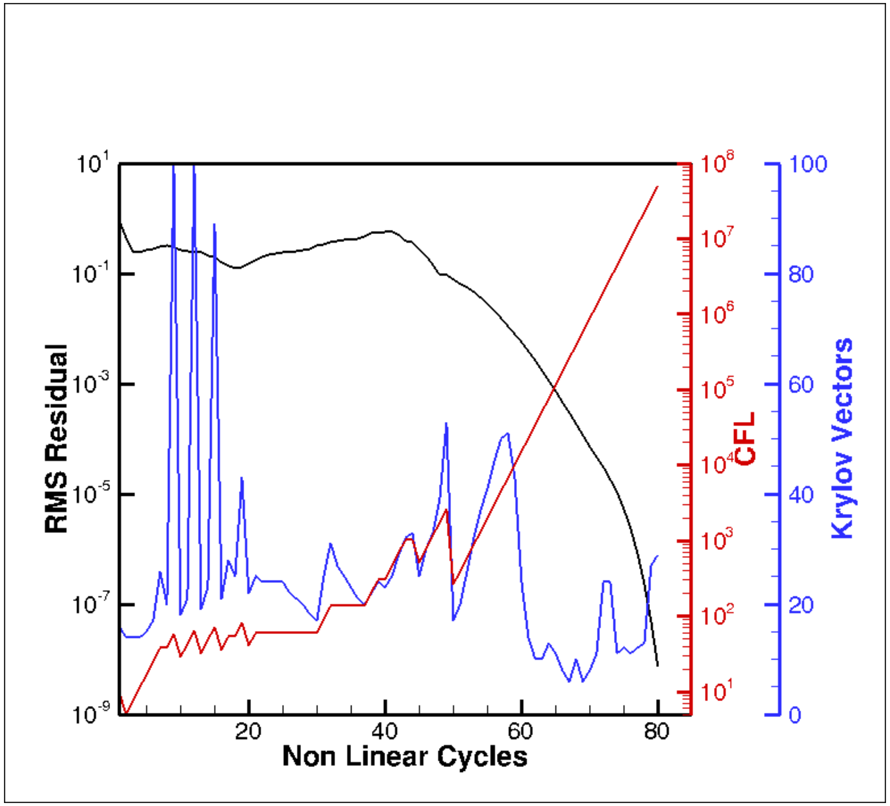
Convergence of PTC Newton-Krylov Original (Unsmoothed)

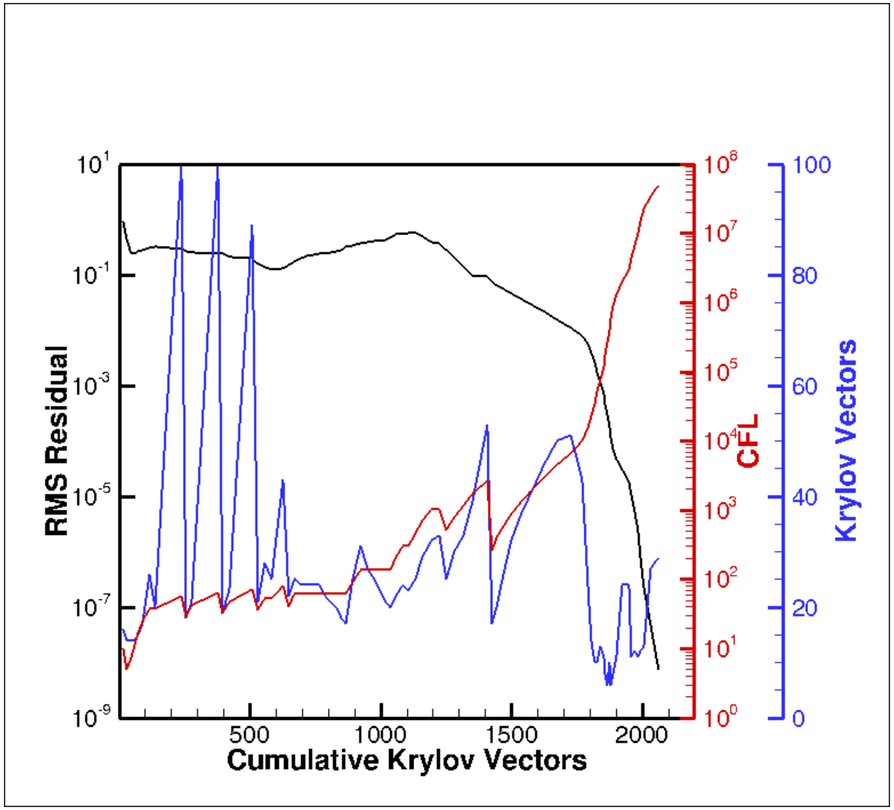




- 80 nonlinear cycles, 2063 total Krylov vectors
- Achieves quadratic convergence at end

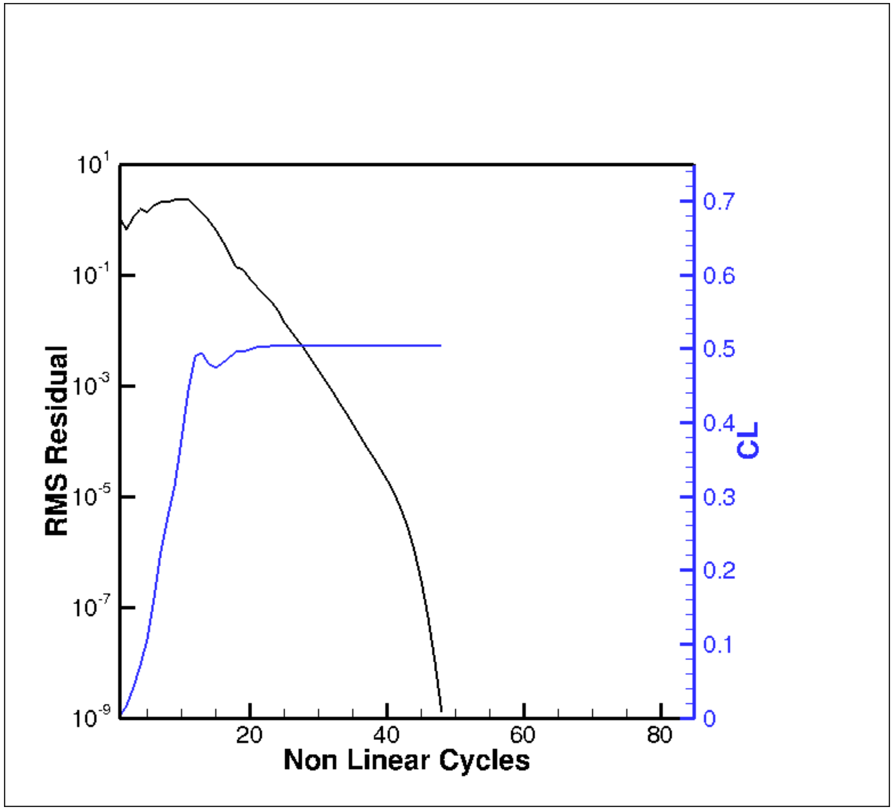
Convergence of PTC Newton-Krylov Original (Unsmoothed)

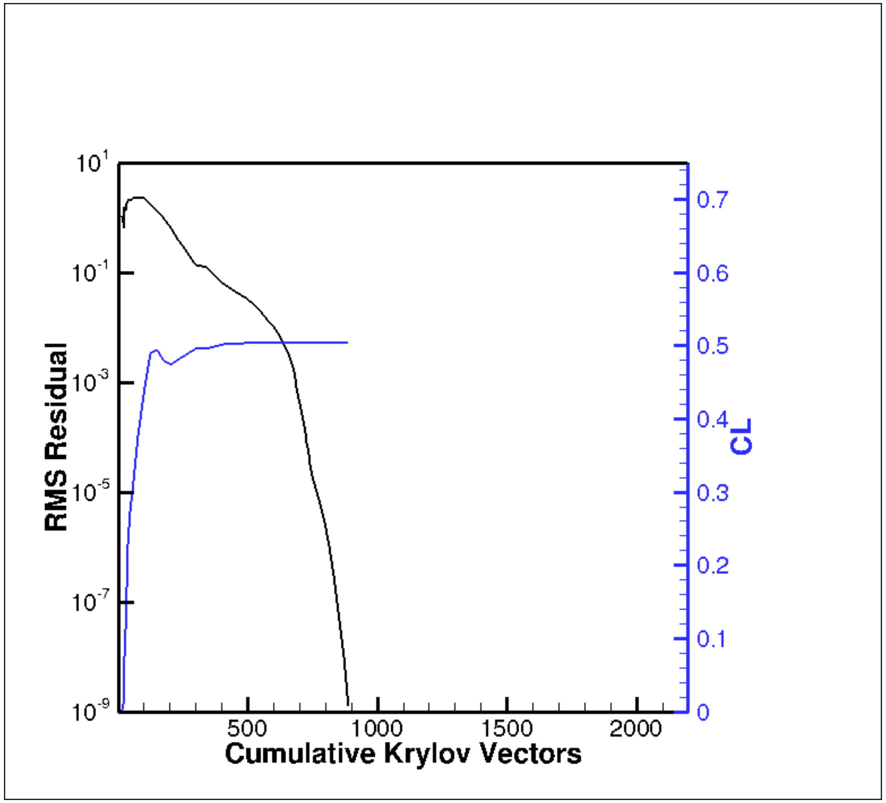




- Some linear systems at startup (low CFL) are difficult to solve !
- CFL only climbs rapidly after ~50 nonlinear cycles (out of 80)

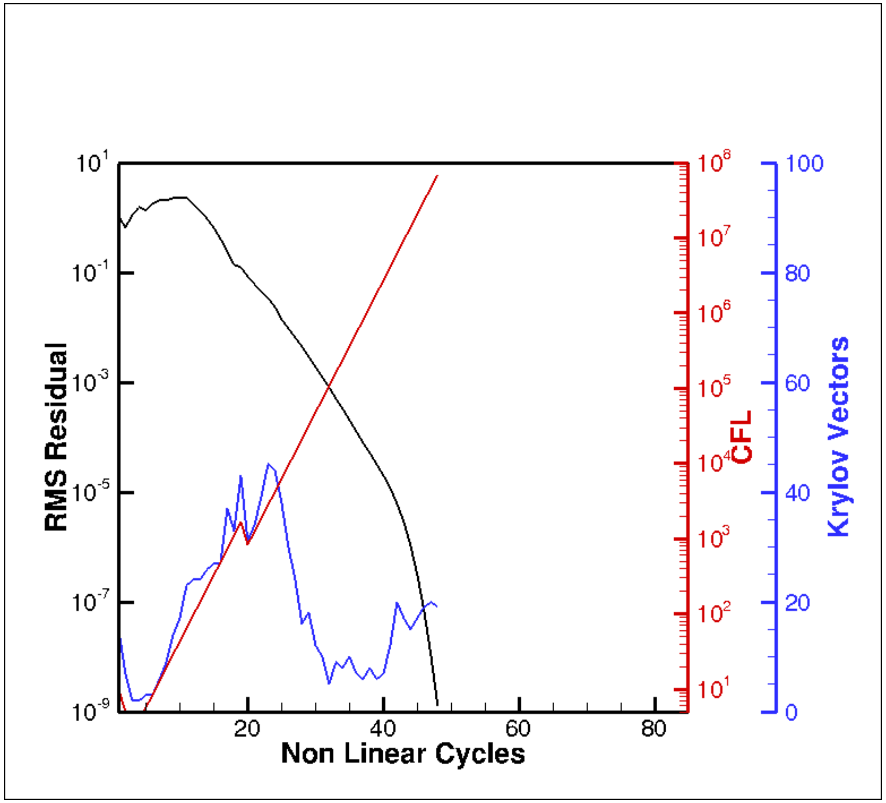
Convergence of PTC Smoothed Newton-Krylov

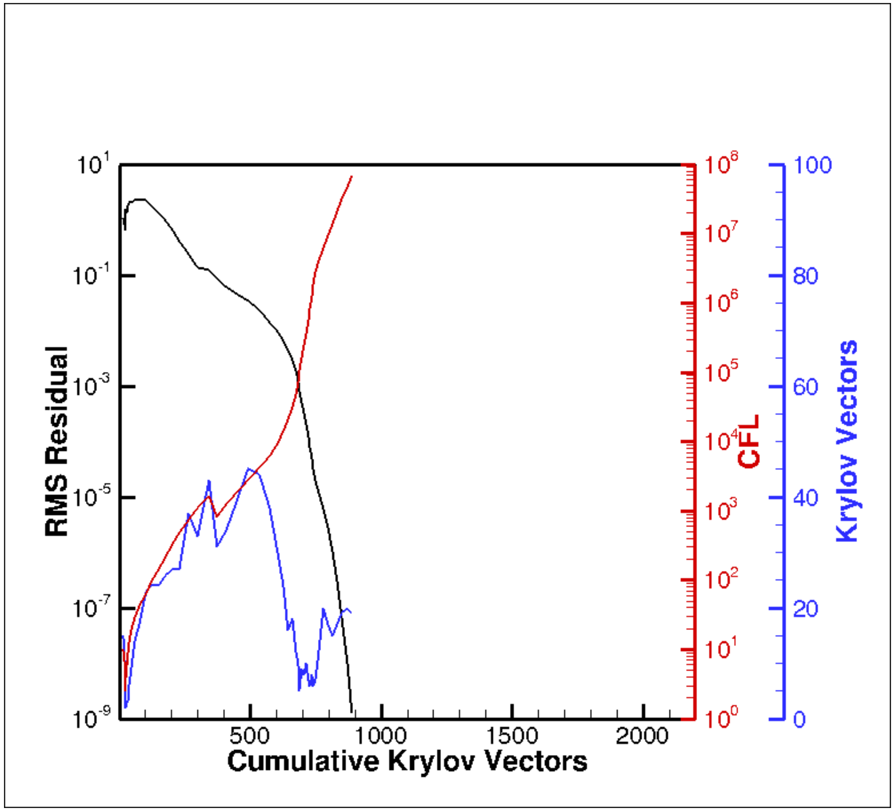




- All settings identical to previous case
- Smoothing constructed using 5 nonlinear cycles of 3-stage line-RK
  - Requires 10% of overall solution time
- Nonlinear cycles reduced from 80 to 43
- Cumulative Krylov vectors reduces from 2068 to 888

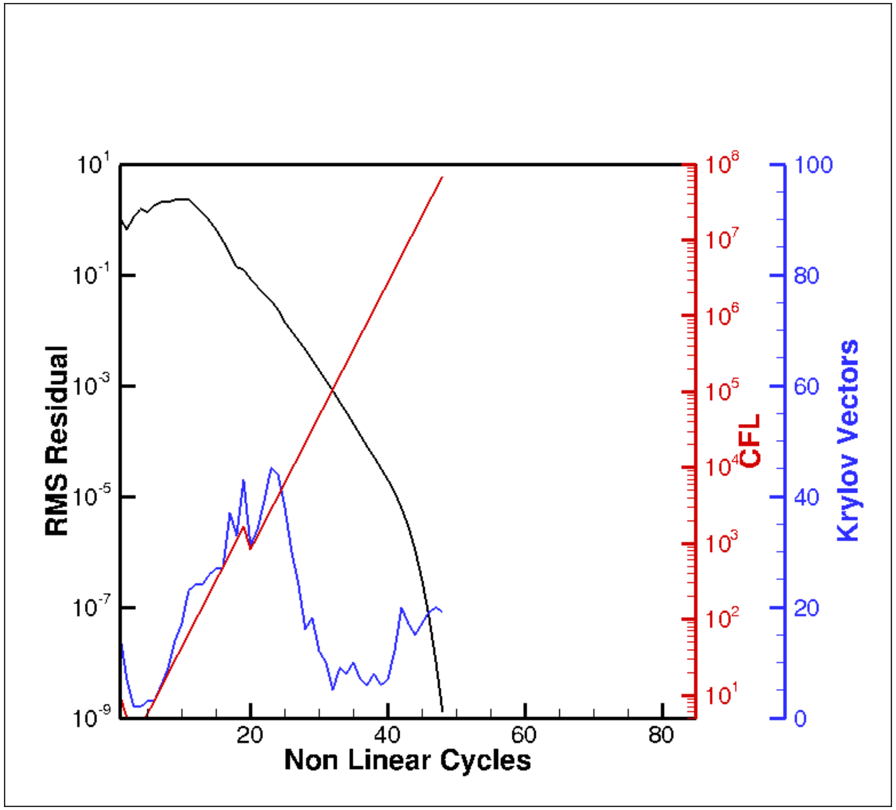
Convergence of PTC Smoothed Newton-Krylov



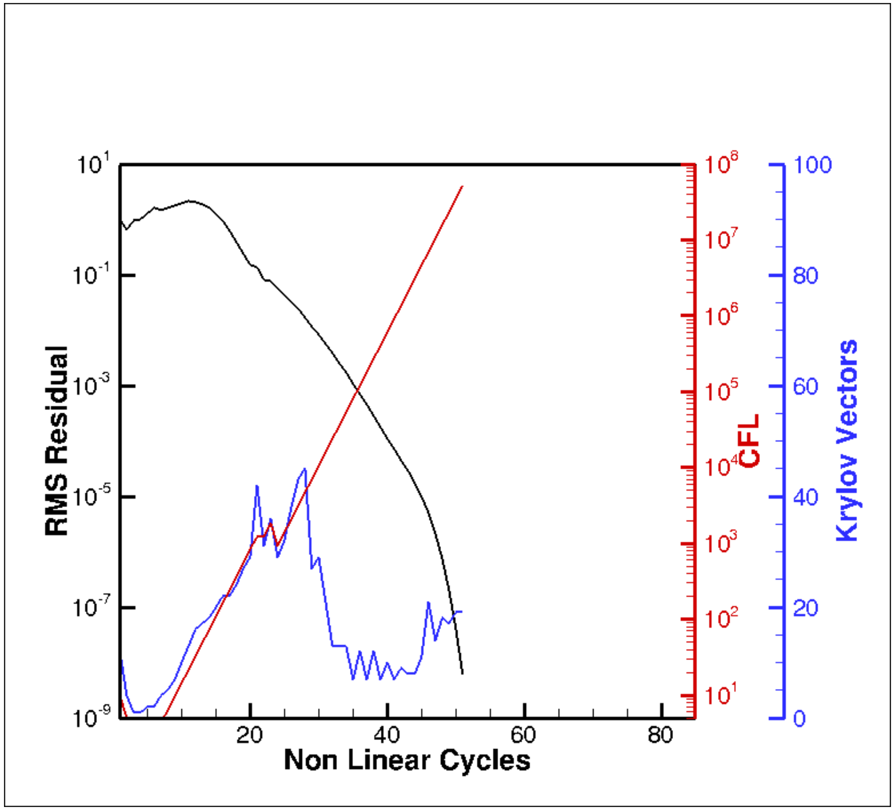


- Near monotonic rise of CFL in continuation process
- No difficult linear systems (as determined by number of Krylov vectors)

Is it Smoothing or Solving ?



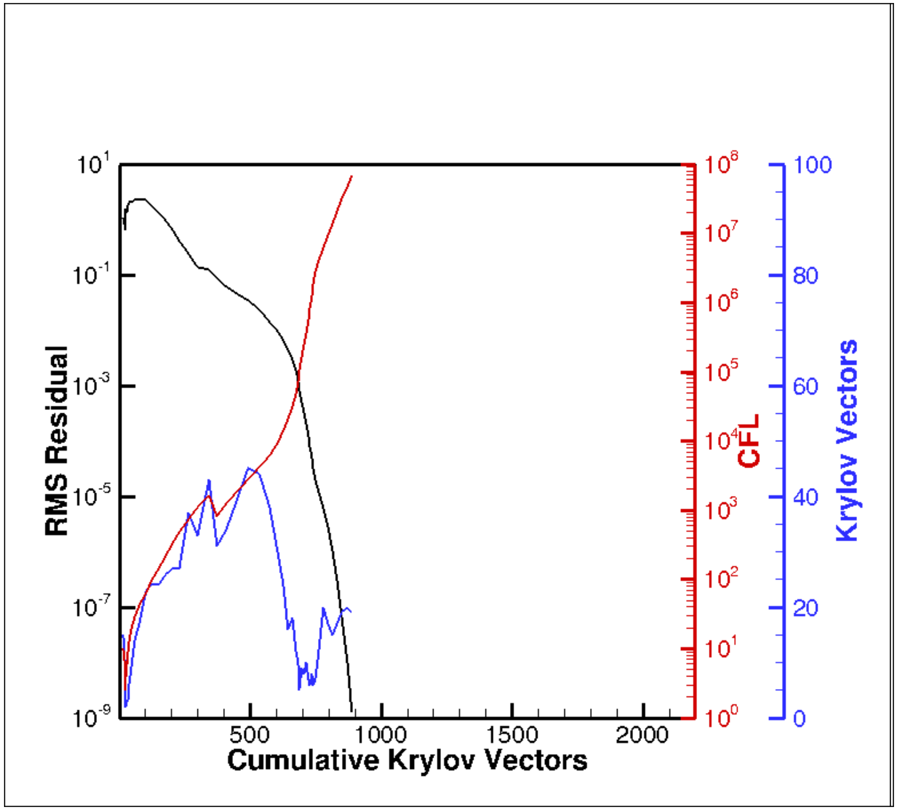
Smoothing = 5 single grid nonlinear cycles



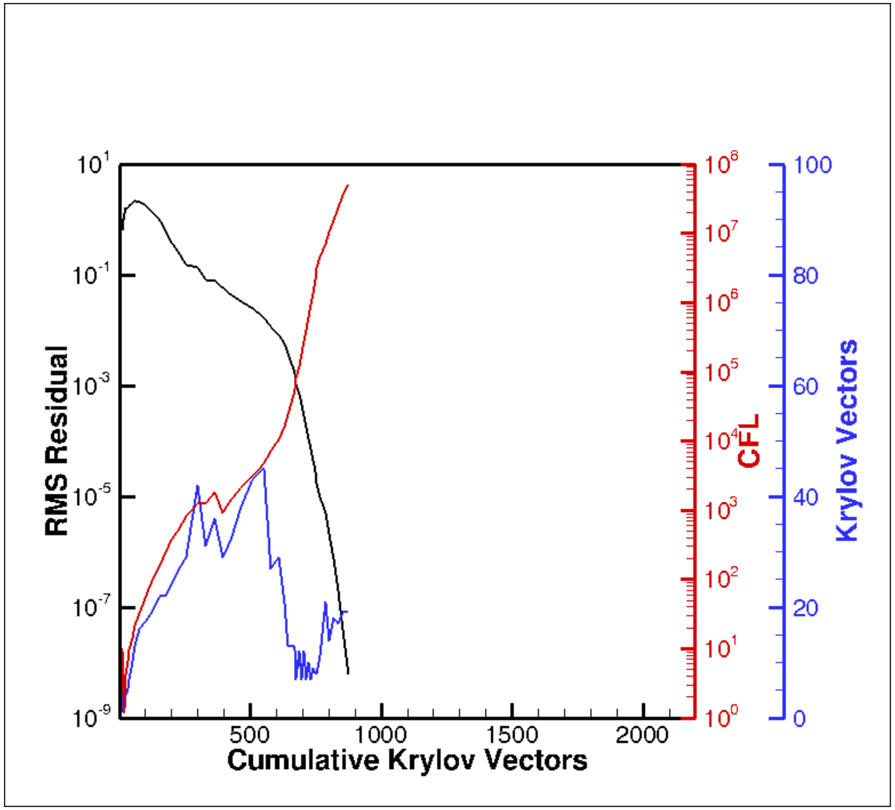
Smoothing = 5 multigrid nonlinear cycles

- Multigrid and single grid smoothing produce similar overall convergence
- Supporting evidence that smoothing is effective mechanism
  - Recall: FAS MG 10X faster than single grid nonlinear solver

Is it Smoothing or Solving ?



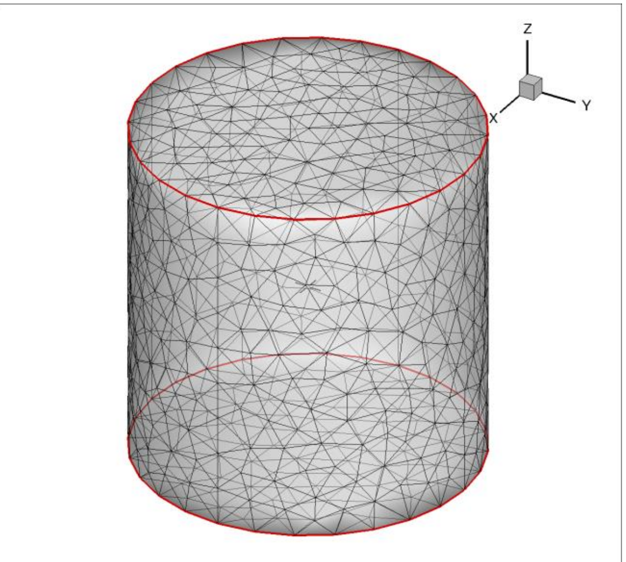
Smoothing = 5 single grid nonlinear cycles

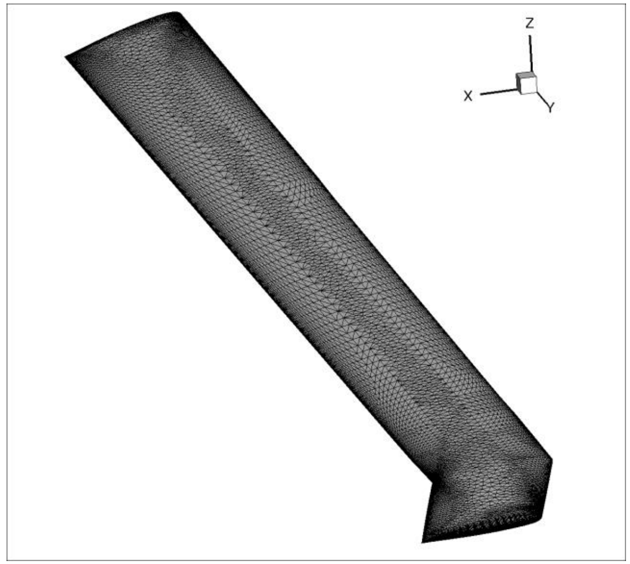


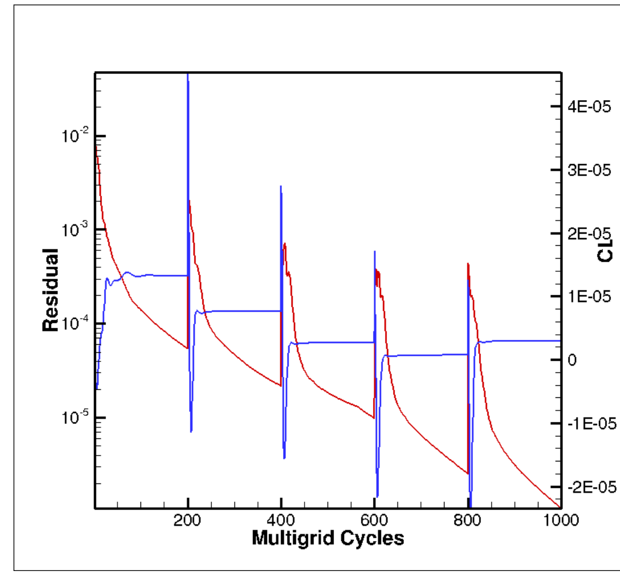
Smoothing = 5 multigrid nonlinear cycles

- Multigrid and single grid smoothing produce similar overall convergence
- Supporting evidence that smoothing is effective mechanism
  - Recall: FAS MG 10X faster than single grid nonlinear solver

Test Case 2: Time-Dependent 4-Bladed Rotor





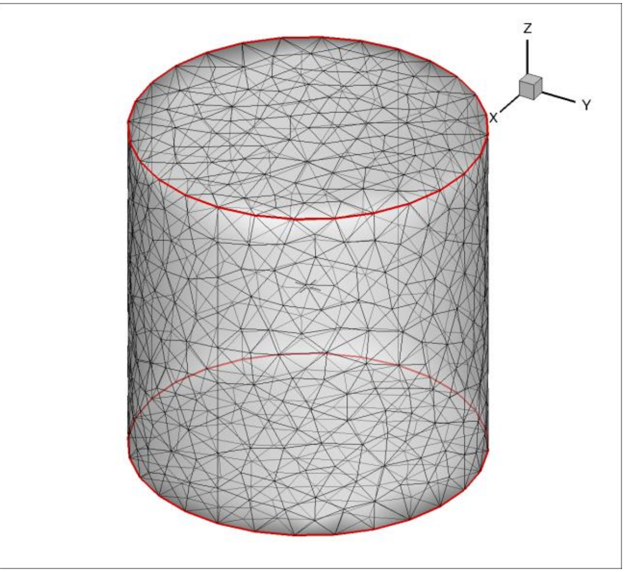


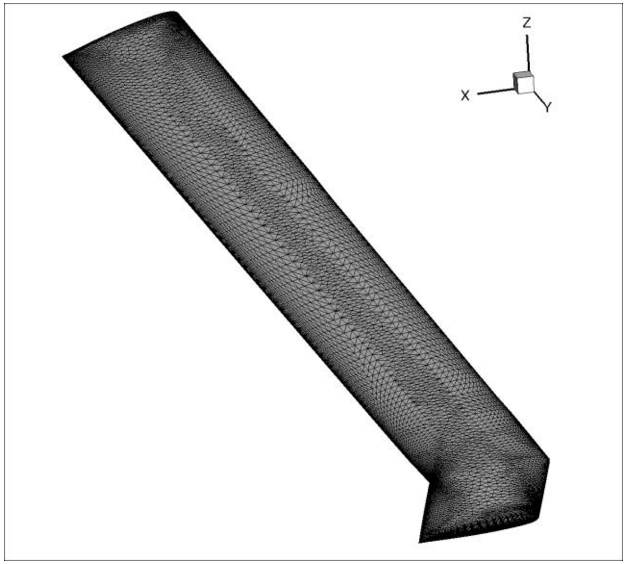
FAS Multigrid Solver
first 5 time steps

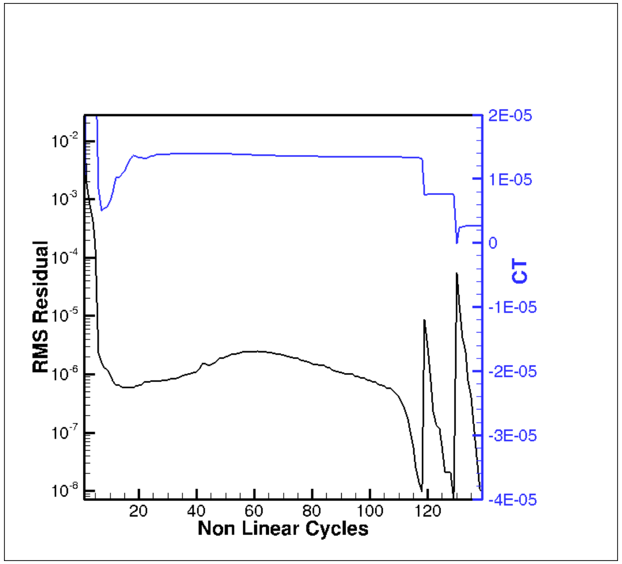
- RANS equations with SA turbulence model
- 2 million point mesh with highly anisotropic prisms near blade surfaces
- BDF2 time discretization: 1 degree time step
- Rotor started impulsively in freestream flow (tip Mach number ~ 0.9)

- FAS Multigrid converges initial and subsequent time steps at similar rates

Time-Dependent Test Case



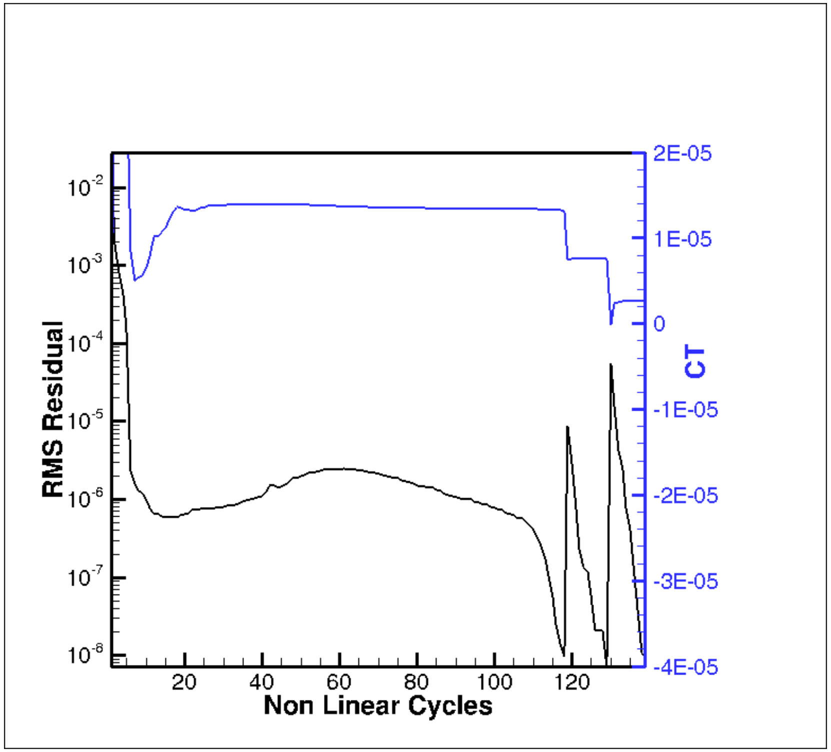


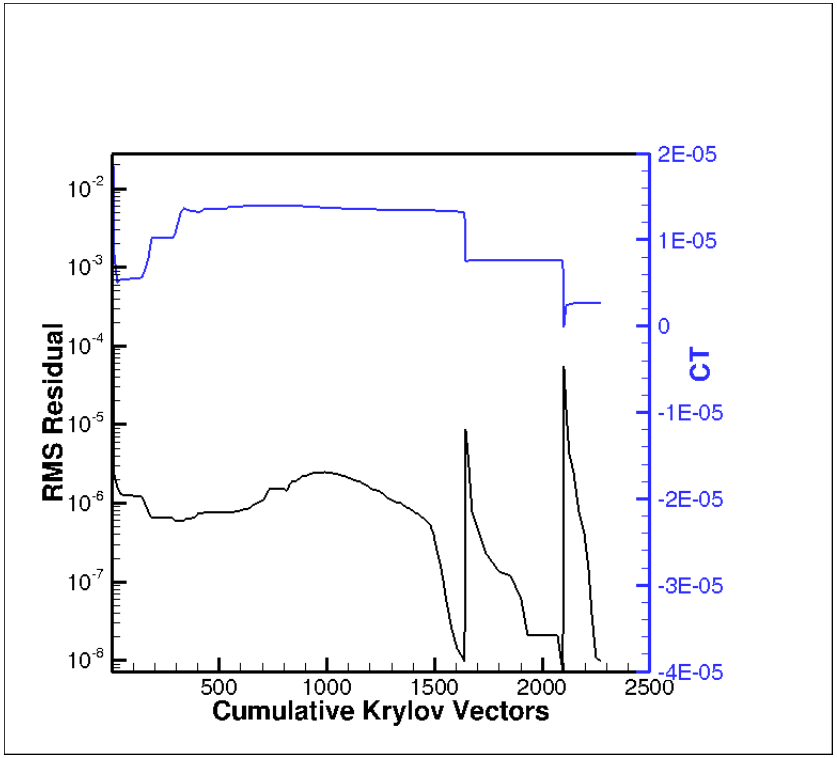


Original (unsmoothed) Newton

- Newton-Krylov method requires lengthy continuation to converge first time step: 120 nonlinear cycles
  - Impulsively started rotor
- Subsequent time steps converge rapidly: < 10 nonlinear cycles
  - Good initial guess from previous time step

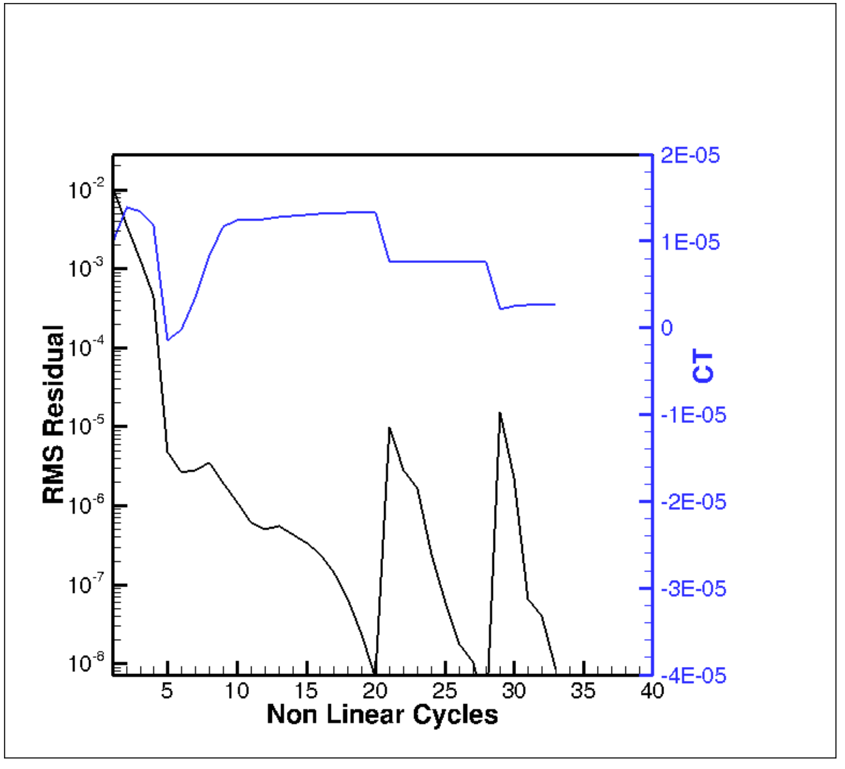
Original (unsmoothed) Newton-Krylov

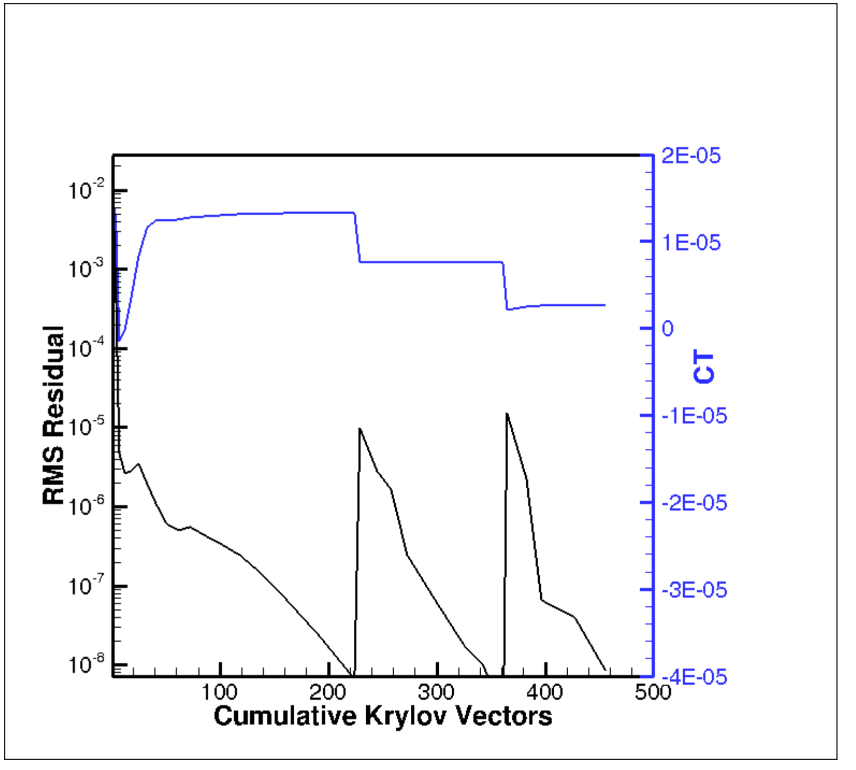




- First time step
  - 120 nonlinear steps, 1600 Krylov vectors
- Third time step
  - 9 nonlinear steps, 150 Krylov vectors

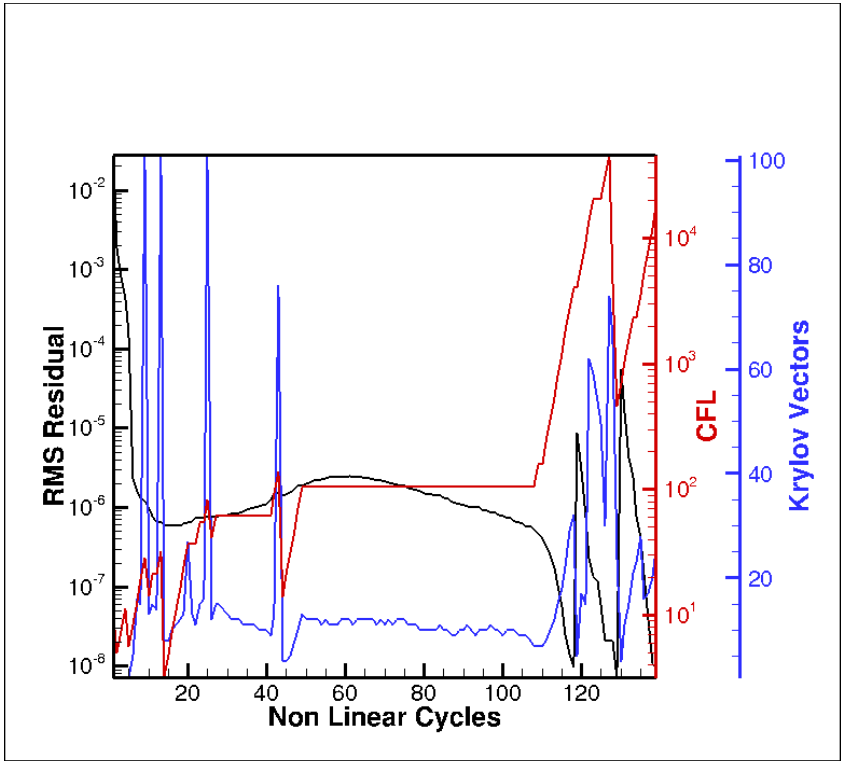
Smoothed Newton-Krylov

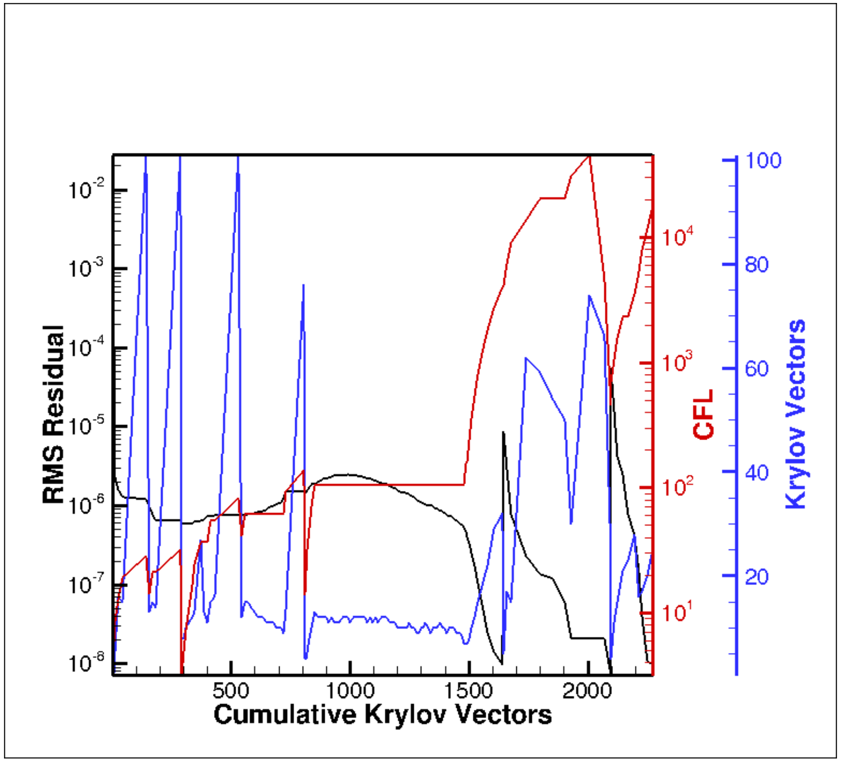




- Smoothing constructed using 5 cycles of 3-stage line RK
- First time step solution reduced from
  - 120 to 20 nonlinear cycles
  - 1600 to 220 Krylov vectors
- Subsequent time steps similar to unsmoothed case
- Convergence of all time steps is more consistent

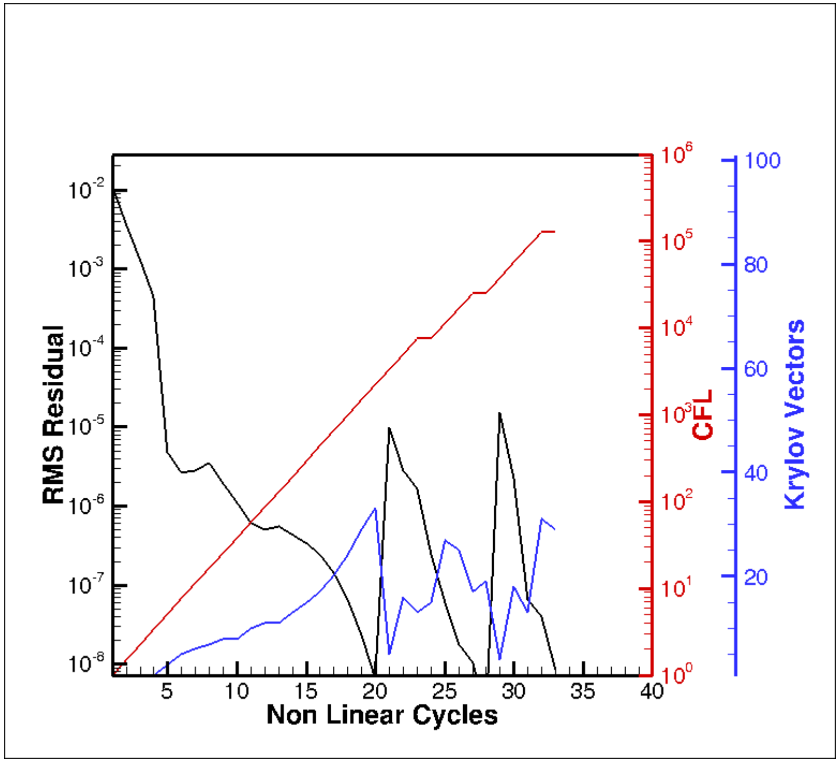
Original (unsmoothed) Newton-Krylov

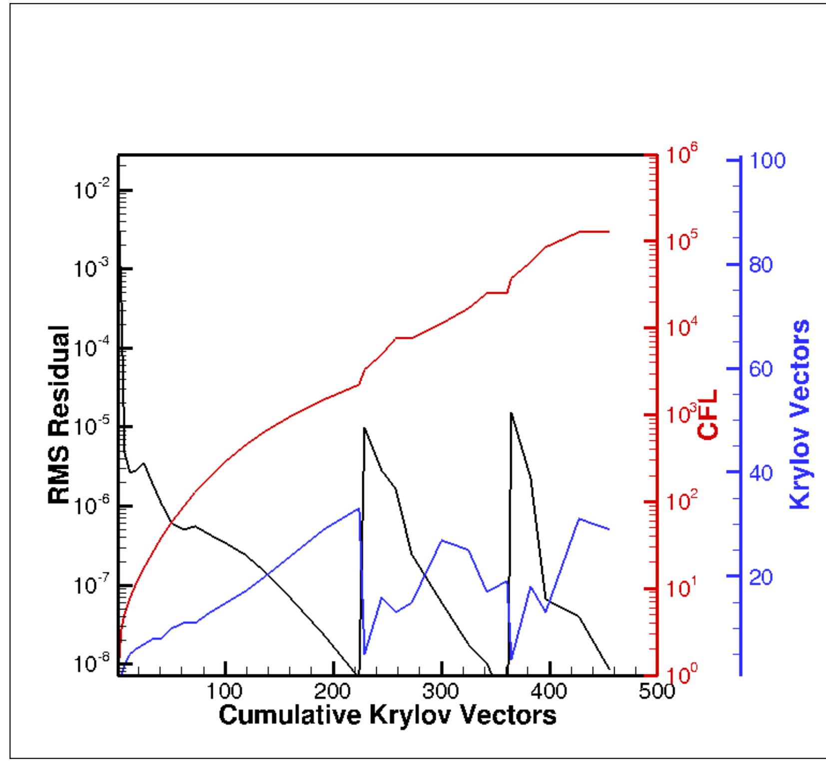




- First time step generates
  - Difficult linear systems
  - Slow CFL growth

Smoothed Newton-Krylov





- Smoothed solver produces monotonic CFL growth
- More similar convergence for all time steps

Conclusions

- Continuation for Newton methods in CFD are often problematic
  - Majority of solver time spent far from domain of quadratic convergence
  - Pseudo-transient continuation can lead to ill-conditioned systems generated by “bad” solution states
- Addition of source term based on nonlinear smoothing can accelerate PTC-Newton schemes
  - Empirical evidence points to smoothing (vs. solving) as dominant mechanism
- Formulation prevents stalling due to small CFL values
  - Reverts to local nonlinear smoother in limit CFL << 1
- Difficulties may still occur if strong nonlinearities arise in intermediate regions $1 \ll \mathrm{CFL} \ll \infty$
  - Future work…

Acknowledgements

- Sandia National Laboratory Contract 1852733
- NASA Grant NNX15AU23A under the Transformational Tools andTechnologies (T^3) project
- University of Wyoming Advanced Research Computing Center (ARCC)