Nonlinear Preconditioning Methods and Applications

Xiao-Chuan Cai

Department of Mathematics
University of Macau
&
Department of Computer Science
University of Colorado Boulder
Outline of the talk

- Root Finding Problem: Find $X \in \mathbb{R}^n$, such that $F(X) = 0$

- Motivating applications
- Nonlinearly preconditioned Newton methods
- Some nonlinearly difficult problems in fluid and solid problems in biomechanics
- Final remarks
Nonlinear problems with local singularity: Blood flows in arteries with aneurysm or stenosis
Nonlinear problems with local singularity: Stroke

View from the front

View from the back
Nonlinear problems with local singularity: Arterial plaques
Motivation from linear preconditioning

- Iterative methods for solving large system of linear equations
  \[ Ax = b \]

  Roughly speaking, if \( h \) is the mesh size and \( np \) is the number of processors of a parallel machine
  - The discretization accuracy \( \approx h^\alpha \)
  - The number of iterations \( \approx h^{-\beta} \cdot (np)^\gamma \)

- Preconditioned iterative methods for solving large system of linear equations
  \[ M^{-1} Ax = M^{-1} b \]

- A good preconditioner reduces the impact of certain parameters on the convergence. Parameters: mesh size, number of processors, jump coefficients, ...
• Preconditioned iterative methods for solving large system of nonlinear equations

\[ F(X) = 0 \iff G(F(X)) = 0 \iff F(G(X)) = 0 \]

• A good nonlinear preconditioner can sometimes drastically improve the robustness of the nonlinear convergence by reducing the impact of certain parameters on the nonlinear convergence

• Parameters: Reynolds number, Mach number, Grashof number, ..., plus mesh size, number of processors, ...

• Note: Linear preconditioning doesn’t help much with the “nonlinear parameters”
A little history of nonlinear preconditioning


In general, we can use the classical Newton’s method for

\[ F(X) = 0, \]

where \( X = (x_1, x_2, \ldots, x_n)^T \) and \( F = (f_1, f_2, \ldots, f_n)^T \)

\[ X^{k+1} = X^k - F'(X^k)^{-1} F(X^k) \]

or, equivalently,

\[ X^{k+1} = X^k + H^k \]

\[ F'(X^k)H^k = -F(X^k), \]

which requires the solving of a large linear system of equations at every iteration.

**Inexact Newton’s method:** \( H^k \) is chosen such that

\[ \|F'(X^k)H^k + F(X^k)\| \leq \eta^k \|F(X^k)\|, \]

for a given \( \eta^k > 0 \) (which may or may not depend on \( k \))
Globally convergent Newton’s methods

- Linesearch: Changing the steplength by a factor of $\lambda^k \in (0, 1]$
  \[ X^{k+1} = X^k - \lambda^k H^k \]
  In other words, $H^k$ is a good search direction if a non-zero $\lambda^k$ can be found such that
  \[ \frac{1}{2} \| F(X^{k+1}) \|^2 \leq \frac{1}{2} \| F(X^k) \|^2 - \alpha \lambda^k (H^k)^T JF(X^k) \]
  where $\alpha$ is small positive parameter

- Trust region: Changing the search direction $H^k$
What happens when inexact Newton is applied to a large system with unbalanced nonlinearities?

- The convergence, or fast convergence, happens only if a good initial guess is available.
- Generally it is very difficult to obtain such an initial guess especially for nonlinear equations that have unbalanced nonlinearities.
- The step length $\lambda^k$ is often determined by the components with the strongest nonlinearities, and this may lead to an extended period of stagnation in the nonlinear residual curve.
- We say that the nonlinearities are “unbalanced” when $\lambda^k$ is, in effect, determined by a subset of the overall degrees of freedom.
We consider a one-dimensional compressible flow problem described by the full potential equation in a variable-area duct. The problem is to determine the solution potential $u(x)$ satisfying

$$(A \rho u_x)_x = 0,$$

for $0 < x < 2$ and $u(0) = 0$ and $u(2) = u_R$ given. The duct area

$$A = A(x) = 0.4 + 0.6(x - 1)^2,$$

and the density $\rho$ is given by

$$\rho = \rho(v) = (c^2)^{1/(\gamma-1)} = \left(1 + \frac{\gamma - 1}{2}(1 - v^2)\right)^{1/(\gamma-1)}.$$

Here $v = u_x$ is the velocity. $\gamma = 1.4$ is the ratio of specific heat and $c$ is the speed of sound.
Some observations

• To advance from $X^k$ to $X^{k+1}$, all $n$ variables and equations need to be updated even though in many situations $n$ can be very large, but only a small number of components of $X^k$ receive significant updates.

• If a small number of components of the initial guess $X^0$ are not acceptable, the entire $X^0$ is declared bad.

• There are two global control variables $\eta^k$ and $\lambda^k$. Any slight change of $F(\cdot)$ may result in the change of $\eta^k$ or $\lambda^k$, and any slight change of $\eta^k$ or $\lambda^k$ may result in some global function evaluations and/or the solving of global Jacobian systems.
Can we remove those bad components, just like Gaussian elimination?

For example

\[
\begin{align*}
F_1(x_1, x_2) &= 0 \\
F_2(x_1, x_2) &= 0
\end{align*}
\]

Eliminate (implicit function theorem) the bad component \(x_2\)

\[x_2 = G(x_1)\]

The leftover is a slightly smaller and easier to solve nonlinear system

\[F_1(x_1, G(x_1)) = 0\]
A simple example with one bad component

- Consider a $2 \times 2$ system

$$F(x_1, x_2) \equiv \begin{bmatrix} F_1(x_1, x_2) \\ F_2(x_1, x_2) \end{bmatrix} = \begin{bmatrix} (x_1 - x_2^3 + 1)^m - x_2^m \\ x_1 + 2x_2 - 3 \end{bmatrix} = 0$$

where $m = 1, 3, 5$. $x^* = [1, 1]^T$ is the root

<table>
<thead>
<tr>
<th>$x^{(0)}$</th>
<th>IN</th>
<th>PIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x^{(0)}$</td>
<td>$m=1$</td>
<td>$m=3$</td>
</tr>
<tr>
<td>$(0, 0)^T$</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>$(0, 2)^T$</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>$(2, 0)^T$</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>$(2, 2)^T$</td>
<td>5</td>
<td>12</td>
</tr>
</tbody>
</table>

- As $m$ increases, one of the equations is more nonlinear than the other; the number of iterations increases and is also more sensitive to the choice of the initial guess
Nonlinear elimination – peak removing

Consider a nonlinear problem \( F(x) = 0 \) defined on \( \Omega \) with the current approximate solution \( x_c \).

- Identify the worst region. A peak of \( F \) is a region \( \omega \in \Omega \) such that \( \| F(x_c) \|_2(\omega) \) is large.
- Solve a local nonlinear problem

\[
F|_{\omega}(x_\omega) = 0, \text{ with boundary condition } x_\omega|_{\partial\omega} = x_c|_{\partial\omega}
\]

- Locally correct the solution

\[
x_{\text{new}} = \begin{cases} 
  x_\omega & \text{in } \omega \\
  x_c & \text{in } \Omega \setminus \omega
\end{cases}
\]
Linear and nonlinear Schwarz preconditioners

• $R_i^\delta$, $R_i^0$ restriction with and without overlap. $\delta$ is the overlapping size

• Additive Schwarz

\[
\text{Linear: } \sum_{i=1}^{N} (R_i^\delta)^T A_i^{-1} R_i^\delta \quad \text{Nonlinear: } \sum_{i=1}^{N} (R_i^\delta)^T F_i^{-1} (R_i^\delta x)
\]

• Restricted additive Schwarz

\[
\text{Linear: } \sum_{i=1}^{N} (R_i^0)^T A_i^{-1} R_i^\delta \quad \text{Nonlinear: } \sum_{i=1}^{N} (R_i^0)^T F_i^{-1} (R_i^\delta x)
\]
A general scalable nonlinear equation solver: RAS-NKS

• **Step 1 (The Nonlinearity Checking Step):** Check stopping conditions.
  - If the global condition is satisfied, **stop**.
  - If the local nonlinearities are not balanced, go to **Step 2**.
  - If the local conditions indicate the nonlinearities are balanced, set \( \tilde{u}^{(k)} = u^{(k)} \), go to **Step 3**.

• **Step 2 (The RAS Step):** Solve local nonlinear problems on the overlapping subdomains to obtain the subdomain correction \( \nu^\delta_i \)

\[
R^\delta_i F(u^{(k)} + \nu^\delta_i) = 0, \quad \text{for } i = 1, \ldots, N
\]

Drop the solution in the overlapping part of the subdomain and compute the global function \( G(u^{(k)}) \)

\[
G(u^{(k)}) = \sum_{i=1}^{N} R^0_i (u^{(k)} + \nu^\delta_i), \quad \text{and set } \tilde{u}^{(k)} = G(u^{(k)}).\]

• **Step 3 (The NKS Step):** Compute the next approximate solution \( u^{(k+1)} \) by solving the following equation

\[
F(u) = 0
\]

with one step of NKS iteration using \( \tilde{u}^{(k)} \) as the initial guess.

Go to **Step 1**
Comparing NKS and RAS-NKS

8x8 = 64 processors on 128x128 grids

Step
Residual
Re = 10^3
Re = 5 \times 10^3
Re = 10^4
Re = 5 \times 10^4
Re = 10^5
Re = 5 \times 10^5
Re = 10^6
Blood flow in the cerebral artery of a stroke patient

Time-dependent incompressible Navier-Stokes equations

\[
\begin{align*}
\rho \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) - \nabla \cdot \mathbf{\sigma} &= 0, \\
\nabla \cdot \mathbf{u} &= 0
\end{align*}
\]

Here \( \rho \) is the blood density, \( \mu \) is the viscosity, and \( \mathbf{\sigma} = -p \mathbf{I} + \mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) \) is the Cauchy stress tensor, \( \mathbf{u} \) is the velocity, \( p \) is the pressure.
Two cases of patient-specific arteries

<table>
<thead>
<tr>
<th>Case</th>
<th>Mean Reynolds</th>
<th># of inlets</th>
<th># of outlets</th>
<th># of nodes</th>
<th># of elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>One-inlet</td>
<td>443.2</td>
<td>1</td>
<td>6</td>
<td>437,538</td>
<td>2,190,164</td>
</tr>
<tr>
<td>Two-inlet</td>
<td>262.6</td>
<td>2</td>
<td>15</td>
<td>1,069,767</td>
<td>5,225,949</td>
</tr>
</tbody>
</table>
Figure: Numerical solutions at $t = 0.7\text{s}$.
We form the following nonlinear system in the region with large residuals

\[ \mathcal{G}(\mathbf{x}) \equiv \mathcal{R}_b^k (\mathcal{F}(\mathbf{x})) + \mathcal{R}_g^k (\mathbf{x} - \mathbf{x}_n^k) = 0. \]

\( \mathbf{x}_k^* \) is accepted as the approximate solution if the stopping condition \( \| \mathcal{G}(\mathbf{x}_k^*) \| \leq \gamma_r^{NE} \| \mathcal{G}(\mathbf{x}_n^k) \| \) is satisfied.

(a) Before the subspace correction  
(b) After the subspace correction

Figure: The one-inlet case: residual contours for the \( u \) component at the second nonlinear step during the second time step.
Nonlinear residual history

![Graph showing nonlinear residual history for different methods: INB, INB-NE (Component-wise), INB-NE (Point-wise), INB-NE (Field-split), INB-NE (Region-based). The x-axis represents the global nonlinear step, and the y-axis represents the residual. The graph compares the convergence of these methods over the nonlinear steps.]
Parallel scalability of INB-NE

Table: Scalability test for the two-inlet case obtained using the INB-NE method with different fill-in levels of the ILU subsolve. A fixed mesh with 5,225,949 elements and 1,069,767 nodes is used. The overlap size of RAS preconditioner is $\delta = 1$. The time step size is 0.0025s.

<table>
<thead>
<tr>
<th>np</th>
<th>Subsolve</th>
<th>$N_{global}$</th>
<th>$L_{global}$</th>
<th>$N_{ne}$</th>
<th>$L_{ne}$</th>
<th>$Time_{ne}(s)$</th>
<th>$Time_{total}(s)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>240</td>
<td>ILU(0)</td>
<td>3.83</td>
<td>1068.52</td>
<td>1.33</td>
<td>6.75</td>
<td>11.04</td>
<td>243.79</td>
</tr>
<tr>
<td></td>
<td>ILU(1)</td>
<td>3.67</td>
<td>668.59</td>
<td>1.20</td>
<td>3.66</td>
<td>10.42</td>
<td>171.54</td>
</tr>
<tr>
<td></td>
<td>ILU(2)</td>
<td>3.83</td>
<td>525.70</td>
<td>1.20</td>
<td>3.33</td>
<td>10.82</td>
<td>171.83</td>
</tr>
<tr>
<td></td>
<td>ILU(3)</td>
<td>3.50</td>
<td>488.47</td>
<td>1.20</td>
<td>3.00</td>
<td>11.54</td>
<td>179.63</td>
</tr>
<tr>
<td>480</td>
<td>ILU(0)</td>
<td>3.83</td>
<td>1115.04</td>
<td>1.33</td>
<td>6.50</td>
<td>6.19</td>
<td>133.81</td>
</tr>
<tr>
<td></td>
<td>ILU(1)</td>
<td>3.50</td>
<td>683.47</td>
<td>1.20</td>
<td>4.00</td>
<td>5.83</td>
<td>87.84</td>
</tr>
<tr>
<td></td>
<td>ILU(2)</td>
<td>3.67</td>
<td>532.91</td>
<td>1.20</td>
<td>3.67</td>
<td>6.05</td>
<td>88.66</td>
</tr>
<tr>
<td></td>
<td>ILU(3)</td>
<td>3.50</td>
<td>491.91</td>
<td>1.20</td>
<td>3.50</td>
<td>6.47</td>
<td>96.40</td>
</tr>
<tr>
<td>960</td>
<td>ILU(0)</td>
<td>3.83</td>
<td>1179.70</td>
<td>1.33</td>
<td>6.63</td>
<td>3.72</td>
<td>77.45</td>
</tr>
<tr>
<td></td>
<td>ILU(1)</td>
<td>3.50</td>
<td>743.81</td>
<td>1.20</td>
<td>4.00</td>
<td>3.69</td>
<td>52.88</td>
</tr>
<tr>
<td></td>
<td>ILU(2)</td>
<td>3.50</td>
<td>541.76</td>
<td>1.20</td>
<td>3.67</td>
<td>3.76</td>
<td>48.08</td>
</tr>
<tr>
<td></td>
<td>ILU(3)</td>
<td>3.67</td>
<td>505.50</td>
<td>1.20</td>
<td>3.50</td>
<td>4.02</td>
<td>56.74</td>
</tr>
<tr>
<td>1920</td>
<td>ILU(0)</td>
<td>3.83</td>
<td>1436.56</td>
<td>1.20</td>
<td>5.00</td>
<td>2.27</td>
<td>53.17</td>
</tr>
<tr>
<td></td>
<td>ILU(1)</td>
<td>3.67</td>
<td>900.36</td>
<td>1.20</td>
<td>4.16</td>
<td>2.30</td>
<td>38.33</td>
</tr>
<tr>
<td></td>
<td>ILU(2)</td>
<td>3.67</td>
<td>653.05</td>
<td>1.20</td>
<td>3.67</td>
<td>2.34</td>
<td>33.31</td>
</tr>
<tr>
<td></td>
<td>ILU(3)</td>
<td>3.67</td>
<td>630.18</td>
<td>1.20</td>
<td>3.83</td>
<td>2.43</td>
<td>38.58</td>
</tr>
</tbody>
</table>
A carotid artery with plaques
Hyperelastic model for arterial wall

- **Energy functional**

\[ \psi = \psi^{iso}(C) + \psi^{vol}(C) + \psi^{ti}(C, M^{(i)}), \]

where \( C \) is Cauchy-Green tensor, \( M^{(i)} \) are the structural tensors

- **Momentum equation**

\[ \text{div}P = -f \]

where \( P = FS, S = \frac{\partial \psi}{\partial C} \)

**Figure:** Cross-section of artery [Klawonn et al., 2008]

**Figure:** Collagen fibre reinforced [Holzapfel et al., 2000]
Computational difficulties

Standard NKS doesn’t work well under the following 3 conditions, and the issue is not the linear solver

- Large deformation
- Nearly incompressible
- Highly anisotropic

<table>
<thead>
<tr>
<th>Material</th>
<th>Poisson’s Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>rubber</td>
<td>0.4999</td>
</tr>
<tr>
<td>soft tissue</td>
<td>0.45 - 0.49</td>
</tr>
<tr>
<td>gold</td>
<td>0.42 - 0.44</td>
</tr>
<tr>
<td>clay</td>
<td>0.3 - 0.45</td>
</tr>
<tr>
<td>stainless steel</td>
<td>0.3 - 0.31</td>
</tr>
<tr>
<td>glass</td>
<td>0.18 - 0.3</td>
</tr>
<tr>
<td>concrete</td>
<td>0.1 - 0.2</td>
</tr>
</tbody>
</table>
3 situations

From Klawonn, Rheinbach et al 2008

\[
\psi_A = \psi^{isocho} + \psi^{volumetric} + \psi^{ti} = c_1 \left( \frac{l_1}{l_3^{1/3}} - 3 \right) + \epsilon_1 \left( l_3^{\epsilon_2} + \frac{1}{l_3^{\epsilon_2}} - 2 \right) + \sum_{i=1}^2 \alpha_1 \langle l_1 J_4^{(i)} - J_5^{(i)} - 2 \rangle \alpha_2
\]

<table>
<thead>
<tr>
<th>Set</th>
<th>Layer</th>
<th>(c_1)</th>
<th>(\epsilon_1)</th>
<th>(\epsilon_2)</th>
<th>(\alpha_1)</th>
<th>(\alpha_2)</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>-</td>
<td>1.0e3</td>
<td>1.0e3</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>Deformations by different forces</td>
</tr>
<tr>
<td></td>
<td>C1</td>
<td>-</td>
<td>1.0e3</td>
<td>1.0e3</td>
<td>1.0</td>
<td>0.0</td>
<td>Different compressibility</td>
</tr>
<tr>
<td></td>
<td>C2</td>
<td>-</td>
<td>1.0e3</td>
<td>1.0e4</td>
<td>1.0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>C3</td>
<td>-</td>
<td>1.0e3</td>
<td>1.0e5</td>
<td>1.0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>Adv.</td>
<td>7.5</td>
<td>100.0</td>
<td>20.0</td>
<td>1.5e10</td>
<td>20.0</td>
<td>Highly anisotropic arterial walls (^1)</td>
</tr>
<tr>
<td></td>
<td>Med.</td>
<td>17.5</td>
<td>100.0</td>
<td>50.0</td>
<td>5.0e5</td>
<td>7.0</td>
<td></td>
</tr>
<tr>
<td>A2</td>
<td>Adv.</td>
<td>6.6</td>
<td>23.9</td>
<td>10</td>
<td>1503.0</td>
<td>6.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Med.</td>
<td>17.5</td>
<td>499.8</td>
<td>2.4</td>
<td>30001.9</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>Adv.</td>
<td>7.8</td>
<td>70.0</td>
<td>8.5</td>
<td>1503.0</td>
<td>6.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Med.</td>
<td>9.2</td>
<td>360.0</td>
<td>9.0</td>
<td>30001.9</td>
<td>5.1</td>
<td></td>
</tr>
</tbody>
</table>

Table: Model parameter sets of \(\psi_A\)

\(^1\) kPa for \(c_1, \epsilon_1, \alpha_1\).
Test for large deformation

Figure: Convergence history of IN and IN-NE.

Figure: Deformations by different pulls.
For consistency with linear elasticity,

\[ C_1 = \frac{\mu}{2}, \quad \epsilon_1 = \frac{\kappa}{2}, \]

where \( \mu, \kappa \) and the shear and bulk modulus. The Poisson ratio can be computed by

\[ \nu = \frac{3K - 2G}{2(3K + G)} \]

<table>
<thead>
<tr>
<th>Set</th>
<th>Poisson’s Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>0.125</td>
</tr>
<tr>
<td>C2</td>
<td>0.452</td>
</tr>
<tr>
<td>C3</td>
<td>0.495</td>
</tr>
</tbody>
</table>
Test for anisotropic arterial wall

Figure: Convergence history of IN and IN-NE.

Figure: von Mises stress
Some final remarks

• For problems with uniform global nonlinearity, NKS is a good general purpose parallel solver. Multilevel maybe necessary if the number of processors is large
• For problems with unbalanced global and local nonlinearities, a combination of full space Newton and subspace Newton, in the form of a nonlinear elimination preconditioned Newton, offers a good strategy
• It may not be easy to identify the local ‘bad’ region
• For problems with only local nonlinearities, subspace Newton is often sufficient
• It is often difficult to tell what types of nonlinearities a problem may have
• The norm of the residual function, $\|F(x)\|_2$, is often not a good monitor, unfortunately all existing nonlinear theories and algorithms are based on $\|F(x)\|_2$
• Many parameters (stopping conditions)

Some recent publications