Arnoldi Algorithm

Nguyễn, Thanh Sơn

Universitaet Bremen
Zentrum fuer Technomathematik

25th March 2009
Outline

1. Introduction
2. Applications
3. Remarks
4. Examples
5. References
Arnoldi algorithm/Arnoldi process is used to produce an orthonormal basis for a Krylov subspace. Given a square matrix $A$, a non-zero vector $x$ and an integer number $m$, find a matrix $V$ s.t. $V^TV = I$ and

$$\text{colspan}(V) = \text{span}(x, Ax, A^2x, ..., A^{m-1}x).$$

- Invented by W. E. Arnoldi in 1951 for eigenvalue problem.
- Wide-used in approximate solvers.
Arnoldi algorithm/Arnoldi process is used to produce an orthonormal basis for a Krylov subspace. Given a square matrix $A$, a non-zero vector $x$ and an integer number $m$, find a matrix $V$ s.t. $V^TV = I$ and

$$
\text{colspan}(V) = \text{span}(x, Ax, A^2x, ..., A^{m-1}x).
$$

- Invented by W. E. Arnoldi in 1951 for eigenvalue problem.
- Wide-used in approximate solvers.
Arnoldi algorithm/Arnoldi process is used to produce an orthonormal basis for a Krylov subspace. Given a square matrix $A$, a non-zero vector $x$ and an integer number $m$, find a matrix $V$ s.t. $V^TV = I$ and

$$colspan(V) = span(x, Ax, A^2x, ..., A^{m-1}x).$$

Invented by W. E. Arnoldi in 1951 for eigenvalue problem.

Wide-used in approximate solvers.
Arnoldi algorithm/Arnoldi process is used to produce an orthonormal basis for a Krylov subspace. Given a square matrix $A$, a non-zero vector $x$ and an integer number $m$, find a matrix $V$ s.t. $V^T V = I$ and

$$colspan(V) = \text{span}(x, Ax, A^2x, ..., A^{m-1}x).$$

Invented by W. E. Arnoldi in 1951 for eigenvalue problem.

Wide-used in approximate solvers.
function \( [V, H] = \text{Arnoldi}(A, b, m, \text{tol}) \)

\[
V = \frac{b}{\beta};
\]

for \( j = 1 : m \)

\[
w = AV(:, j);
\]

for \( i = 1 : j \)

\[
H(i, j) = w' V(:, i);
\]

end

end

\[
w = w - H(i, j) \ast V(:, i);
\]

end

end

\[
H(j + 1, j) = \text{norm}(w);
\]

if \( H(j + 1, j) < \text{tol} \)

break

end

end

\[
V = [V w \div H(j + 1, j)];
\]
function \([V, H] = \text{Arnoldi}(A, b, m, \text{tol})\)

\[ H = \text{zeros}(m + 1, m); \]
\[ \beta = \text{norm}(b); \]
\[ V = b/\beta; \]

for \(j = 1 : m\)

\[ w = AV(:, j); \]

for \(i = 1 : j\)

\[ H(i, j) = w' V(:, i); \]

end

end

for \(i = 1 : j\)

\[ w = w - H(i, j) \ast V(:, i); \]

end

\[ H(j + 1, j) = \text{norm}(w); \]

if \(H(j + 1, j) < \text{tol}\)

break

\[ H = H(1 : j, 1 : j); \]

end

\[ V = [V \, w/H(j + 1, j)]; \]

end

end
Three matrices $A$, $V$, $H$ satisfy the relations

\[ AV_m = V_{m+1}H_{(m+1)xm}; \quad (1) \]
\[ V_m^T AV_m = H_{mxm}. \quad (2) \]

$H_{mxm}$ is a Hessenberg matrix, $V_m^T V_m = I_m$.

This structure is useful in many situations in linear algebra.
GMRES (Generalized Minimum RESidual method): Given a square (usually large, sparse) system $Ax = b$ of order $N$, initial guess $x_0$, $r_0 = b - Ax_0$ called initial residual. One finds the approximate solution in affine subspace $x_0 + \mathcal{K}_m(A, r_0)$.

Let $x = x_0 + V_m y, y \in \mathbb{R}^m$, the minimization problem $\|b - Ax\|_2$ subject to $x \in \mathbb{R}^N$ is converted to minimizing $\|r_0\| e_1 - H_{(m+1)}x_m y \|$ subject to $y \in \mathbb{R}^m$.

If $A$ is nonsingular, GMRES breaks down at $m^{th}$ iff $x_m = x_0 + V_m y$ is the exact solution.

Some other methods or variations: FOM, GCR, GMRES-DR, c.f [4, 6, 7, 8]
GMRES (Generalized Minimum RESidual method): Given a square (usually large, sparse) system $Ax = b$ of order $N$, initial guess $x_0$, $r_0 = b - Ax_0$ called initial residual. One finds the approximate solution in affine subspace $x_0 + \mathcal{K}_m(A, r_0)$.

Let $x = x_0 + V_m y, y \in \mathbb{R}^m$, the minimization problem $\|b - Ax\|_2$ subject to $x \in \mathbb{R}^N$ is converted to minimizing $\|r_0\| e_1 - H_{(m+1)} x_m y \| \text{ subject to } y \in \mathbb{R}^m$.

If $A$ is nonsingular, GMRES breaks down at $m^{th}$ iff $x_m = x_0 + V_m y$ is the exact solution.

Some other methods or variations: FOM, GCR, GMRES-DR, c.f [4, 6, 7, 8]
GMRES (Generalized Minimum RESidual method): Given a square (usually large, sparse) system $Ax = b$ of order $N$, intial guess $x_0$, $r_0 = b - Ax_0$ called intial residual. One finds the approximate solution in affine subspace $x_0 + \mathcal{K}_m(A, r_0)$.

Let $x = x_0 + V_m y$, $y \in \mathbb{R}^m$, the minimization problem $\|b - Ax\|_2$ subject to $x \in \mathbb{R}^N$ is converted to minimizing $\|r_0\|_e_1 - H_{(m+1)}x_m y\|$ subject to $y \in \mathbb{R}^m$.

If $A$ is nonsingular, GMRES breaks down at $m^{th}$ iff $x_m = x_0 + V_m y$ is the exact solution.

Some other methods or variations: FOM, GCR, GMRES-DR, c.f [4, 6, 7, 8]
GMRES (Generalized Minimum RESidual method): Given a square (usually large, sparse) system $Ax = b$ of order $N$, initial guess $x_0$, $r_0 = b - Ax_0$ called initial residual. One finds the approximate solution in affine subspace $x_0 + \mathcal{K}_m(A, r_0)$.

Let $x = x_0 + V_m y$, $y \in \mathbb{R}^m$, the minimization problem $\|b - Ax\|_2$ subject to $x \in \mathbb{R}^N$ is converted to minimizing $\||r_0||e_1 - H(m+1)x_m y||$ subject to $y \in \mathbb{R}^m$.

If $A$ is nonsingular, GMRES breaks down at $m^{th}$ iff $x_m = x_0 + V_m y$ is the exact solution.

Some other methods or variations: FOM, GCR, GMRES-DR, c.f [4, 6, 7, 8]
The original eigenvalue problem $Ax = \lambda x$ is replaced by the "reduced" eigenvalue problem $H_{mxm}z_m = \theta z_m$ where the Arnoldi basis $V_m$ is started with a unit-normed initial eigenvector $v_1$. Then, the approximate eigenpairs of $A$ are selected from the Ritz pairs $\{(\theta_i, V_mz_{m,i})\}$.

This approach yields a better result in comparison with the Power method which only uses $A^{m-1}v_1$ for approximation. c.f.[5]
The original eigenvalue problem $Ax = \lambda x$ is replaced by the "reduced" eigenvalue problem $H_{mxm}z_m = \theta z_m$ where the Arnoldi basis $V_m$ is started with a unit-normed initial eigenvector $v_1$. Then, the approximate eigenpairs of $A$ are selected form the Ritz pairs $\{(\theta_i, V_mz_{m,i})\}$.

This approach yields a better result in compare with *Power method* which only uses $A^{m-1}v_1$ for approximation. c.f.[5]
Consider a LTI dynamical system

\[ Ex'(t) = Ax(t) + bu(t), \]  
\[ y(t) = cx(t) + du(t); \]  

The transfer function is \( H(s) = c(sE - A)^{-1}b + d \). Moments matching method approximates \( H(s) \) near some point, say \( s_0 \), by matching a few leading coefficients of Taylor expansion at \( s_0 \).

\[ H(s) = -c \sum_{i=0}^{\infty} ((A - s_0E)^{-1}E)^i(A - s_0E)^{-1}b(s - s_0)^i + d \]
This can be done by projecting the system (3)-(4) onto Krylov subspace \( V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b) \) or both
\( V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b) \) and
\( W = \mathcal{K}_{i+1}(((A - s_0E)^{-1}E)^T, C^T) \). The reduced systems are then,

**One-sided**

\[
V^T EVx' = V^T AVx + V^T bu,
\]
\[
y = cVx + du;
\]

**Two-sided**

\[
W^T EVx' = W^T AVx + W^T bu,
\]
\[
y = cVx + du;
\]
Model reduction

This can be done by projecting the system (3)-(4) onto Krylov subspace $V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b)$ or both $V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b)$ and $W = \mathcal{K}_{i+1}(((A - s_0E)^{-1}E)^T, C^T)$. The reduced systems are then,

One-sided

$$V^T EVx' = V^T AVx + V^T bu,$$
$$y = cVx + du;$$

Two-sided

$$W^T EVx' = W^T AVx + W^T bu,$$
$$y = cVx + du;$$
This can be done by projecting the system (3)-(4) onto Krylov subspace \( V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b) \) or both
\( V = \mathcal{K}_m((A - s_0E)^{-1}E, (A - s_0E)^{-1}b) \) and
\( W = \mathcal{K}_{i+1}(((A - s_0E)^{-1}E)^T, C^T) \). The reduced systems are then,

**One-sided**

\[
V^T EVx' = V^T AVx + V^T bu, \\
y = cVx + du;
\]

**Two-sided**

\[
W^T EVx' = W^T AVx + W^T bu, \\
y = cVx + du;
\]
Deflation may occur, if the process has not been convergent, restarting is required.

Restarting is also needed when the order of Krylov subspace is big, (but under-convergent) since this requires much computer memories.

Multiple starting vectors leads to so called block Arnoldi algorithm which requires more efforts in implementation.
- **Deflation** may occur, if the process has not been convergent, restarting is required.

- **Restarting** is also needed when the order of Krylov subspace is big, (but under-convergent) since this requires much computer memories.

- Multiple starting vectors leads to so called *block Arnoldi algorithm* which requires more efforts in implementation.
Deflation may occur, if the process has not been convergent, restarting is required.

Restarting is also needed when the order of Krylov subspace is big, (but under-convergent) since this requires much computer memories.

Multiple starting vectors leads to so called block Arnoldi algorithm which requires more efforts in implementation.
Two applications of Arnoldi algorithm in solving linear equations and model reduction are provided.


