Arnoldi Algorithm

Nguyễn, Thanh Sơn

Universitaet Bremen
Zentrum fuer Technomathematik

25th March 2009

Outline

Introduction

- 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

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

- Invented by W. E. Arnoldi in 1951 for eigenvalue problem.
- Wide-used in approximate solvers.

Introduction

- 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.

Introduction

- 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

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

- Invented by W. E. Arnoldi in 1951 for eigenvalue problem.
- Wide-used in approximate solvers.

Introduction

- 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

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

- Invented by W. E. Arnoldi in 1951 for eigenvalue problem.
- Wide-used in approximate solvers.

Coding

```
function [V,H] = Arnoldi(A,b,m,tol)
H = zeros(m+1,m);
beta = norm(b);
V = b/beta;
for j = 1:m
  w = AV(:,j);
  for i = 1:j
    H(i,j) = w'V(:,i);
  end
  for i = 1:j
    w = w - H(i,j)*V(:,i);
  end
  H(j+1,j) = norm(w);
  if H(j+1,j) < tol
    break
    H = H(1:j,1:j);
  end
  V = [V w/H(j+1,j)];
end
end
```

Coding

```
function [V,H] = Arnoldi(A,b,m,tol)
H = zeros(m+1,m);
beta = norm(b);
V = b/beta;
for j = 1:m
  w = AV(:,j);
  for i = 1:j
    H(i,j) = w'V(:,i);
  end
  for i = 1:j
    w = w - H(i,j)*V(:,i);
  end
  H(j+1,j) = norm(w);
  if H(j+1,j) < 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

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

  H_{mxm} is a Hessenberg matrix, $V_m^T V_m = I_m$.
- This structure is useful in many situations in linear algebra.

Linear equations

- 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 $\|\|r_0\|e_1 - H_{(m+1)xm}y\|$ 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]

Linear equations

- 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 $\|\|r_0\|e_1 - H_{(m+1)xm}y\|$ 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]

Linear equations

- 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 $\|\|r_0\|e_1 - H_{(m+1)xm}y\|$ 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]

Linear equations

- 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 $\| \|r_0\| e_1 - H_{(m+1)xm} y\|$ 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]

Eigenvalues problem

- 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 $\{(\theta_i, V_m z_{m,i})\}$.
- This approach yields a better result in compare with *Power method* which only uses $A^{m-1} v_1$ for approximation. c.f.[5]

Eigenvalues problem

- 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 $\{(\theta_i, V_m z_{m,i})\}$.
- This approach yields a better result in compare with *Power method* which only uses $A^{m-1} v_1$ for approximation. c.f.[5]

Model reduction

Consider a LTI dynamical system

$$Ex'(t) = Ax(t) + bu(t), \quad (3)$$

$$y(t) = cx(t) + du(t); \quad (4)$$

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.

$$H(s) = -c\sum_{i=0}^{\infty}((A - s_0E)^{-1}E)^i(A - s_0E)^{-1}b(s - s_0)^i + d$$

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

$$V^TEVx' = V^TAVx + V^Tbu,$$
$$y = cVx + du;$$

Two-sided

$$W^TEVx' = W^TAVx + W^Tbu,$$
$$y = cVx + du;$$

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

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

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

$$V^TEVx' = V^TAVx + V^Tbu,$$
$$y = cVx + du;$$

Two-sided

$$W^TEVx' = W^TAVx + W^Tbu,$$
$$y = cVx + du;$$

- 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.

Examples

Two applications of Arnoldi algorithm in solving linear equations and model reduction are provided.

- [Antoulas05] A. C. Antoulas, *Approximation of Large-scale Dynamical Systems,* Advances in Design and Control DC-06, SIAM, Philadenphia, 2005.
- [Chahlaoui02] Y. Chahlaoui, P. V. Doreen, "A collection of benchmark examples for model reduction of linear time invariant dynamical systems", *Technical report*, SILICOT Working Note 2002-2, 2002.
- [Freund99] R. W. Freund, "Krylov-subspace methods for reduced-order modeling in circuit simulation", *Numerical Analysis Manuscript*, No. 99-3-17, Bell Laboratories.
- [Morgan02] R. B. Morgan, "GMRES with deflated restarting", *SIAM J. Sci. Comput.*, Vol. 24, No. 1, pp. 20-37, 2002.

- [Saad92] Y. Saad, *Numerical Methods for Eigenvalues Problems*, Manchester University Press, 1992.
- [Saad96] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, 1996.
- [Sturler96], E. de Sturler, "Nested Krylov methods based on GRC", *J. Computational and Applied Mathematics*, 67, pp. 15-41, 1996.
- [Sturler99], E. de Sturler, "Truncation strategies for optimal Krylov subspace methods", *SIAM J. Numer. Anal.*, Vol. 36, No. 3, pp. 864-889, 1999.