Finding Rightmost Eigenvalues of Large, Sparse, Nonsymmetric Parameterized Eigenvalue Problems

AMSC 663-664 Final Report

Minghao Wu
AMSC Program
mwu@math.umd.edu

Dr. Howard Elman
Department of Computer Science
elman@cs.umd.edu

Problem Statement

To find the rightmost eigenvalues of:

$$\boldsymbol{Ax} = \boldsymbol{\lambda Bx}.$$

where matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are

- Real ***N**-by-**N***
- Large
- Sparse
- Nonsymmetric
- Depend on one or several parameters

Application

- To determine the stability of the linearized system of the form[1]:

$$B\dot{x} = Ax$$

- The steady state solution $\boldsymbol{x^*}$ is

  - ***stable***, if all the eigenvalues of $\boldsymbol{Ax = \lambda Bx}$ have negative real parts;

  - ***unstable***, otherwise.

Basic *Arnoldi* Algorithm and Implicitly Restarted *Arnoldi*

Properties of basic *Arnoldi* algorithm

Matrix transformation

Implicitly restarted *Arnoldi*

Eigensolver

- *Arnoldi* algorithm[1]

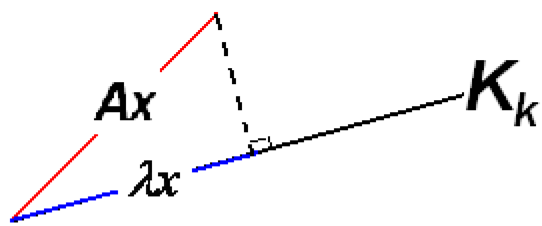
  - iterative method

  - based on *Krylov* subspace

    k - dimensional *Krylov* subspace:

$$\boldsymbol{K_k(A,u_1)} = \mathrm{span}\{\boldsymbol{u_1}\ \boldsymbol{Au_1}\ \boldsymbol{A^2u_1} \ldots \boldsymbol{A^ku_1}\}$$

  - residual of computed eigenpairs is orthogonal to $\boldsymbol{K_k}$



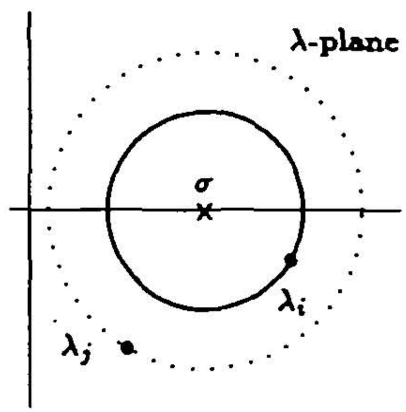
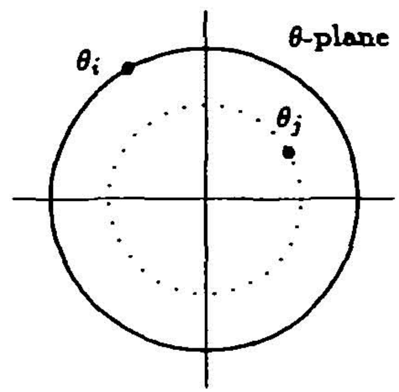
  - only solves standard eigenvalue problem $\boldsymbol{Ax = \lambda x}$

  - converges to well-separated extremal eigenvalues

Matrix Transformation

- Shift – Invert Transformation[1]

$$T_{SI} = (A-\sigma B)^{-1}B$$

$Ax = \lambda Bx$	$T_{SI}x = \theta x$
λ	$\theta = 1/(\lambda-\sigma)$
	

Computational Result

- Example: Olmstead model[2]

$$\begin{cases} u_t = S_{xx} + cu_{xx} + Ru - u^3 \\ bS_t = (1-c)u - S \end{cases}$$

$$u = S = 0 \text{ at } x = 0,\ \pi.$$

u: velocity, ***S***: a quantity related to viscoelastic forces

b, ***c***, ***R***: parameters. ***R***: Rayleigh number

- $b = 2$, $c = 0.1$, $R = 0.6$, $N = 1000$, $k = 20$, $\sigma = 0$:

rightmost eigenvalues: $\lambda_{1,2} = 0 \pm 0.4472i$
residual: $||Ax_i - \lambda_i x_i||_2 / ||x_i||_2 = 1.5659 \times 10^{-11}$, $i = 1,2$

Implicitly Restarted *Arnoldi*

- Motivation
  - large *Krylov* subspace is not practical
  - singular *B* gives rise to spurious eigenvalues

- Basic idea of IRA

  Filter out unwanted eigendirections from the starting vector of *Arnoldi* algorithm by applying shifted *QR* algorithm to a small upper *Hessenberg* matrix[3]

Computational Result

- Example[4]:

$$\begin{bmatrix} K & C \\ C^T & 0 \end{bmatrix} x = \lambda \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix} x$$

- K (200 × 200), C (200 × 100) and M (200 × 200) are of full rank

- eigenvalues lie between -3 and 50

- mimics the eigenproblem arises from Navier-Stokes equation

$k = 10$, $\sigma = 60$

Exact Eigenvalues	**Computed Eigenvalues (IRA)**	**Computed Eigenvalues (basic Arnoldi)**
49.9129 + 0i	49.9129 + 0i	**193.8412 ± 7113830.9524i (residual ≈ 10^{-4})**
2.9112 ± 1.1256i	2.9112 ± 1.1256i	49.9129 + 0i
2.5036 ± 0.0624i	2.5036 ± 0.0624i	3.0891 + 0i
2.3792 + 0i	2.3792 + 0i	2.6112 + 0i
2.1318 ± 0.9356i	2.1318 ± 0.9356i	-0.5752 ± 2.7079i
2.1081 ± 1.3539i	2.1081 ± 1.3539i	-1.0901 ± 0.7532i

Arnoldi Algorithm with Iterative Linear System Solver

How to solve $(A - \sigma B)w = b$ efficiently?

($A - \sigma B$: **large**, **sparse**, **nonsymmetric**)

GMRES Method

- GMRES stands for the ***G***eneralized ***M***inimal ***RES***idual method.
- Based on *Krylov* subspace
- Solves the following minimization problem

$$\min_{x \in x_0 + K_m} \left\| b - Ax \right\|_2$$

where $\boldsymbol{x_0}$ is the initial guess and $\boldsymbol{K_m}$ is the $\boldsymbol{m}$ – dimensional *Krylov* subspace

Preconditioning of $Ax = b$

- Goal

  cluster the eigenvalues of the original matrix

- Basic idea

  solve $M^{-1}Ax = M^{-1}b$ instead of $Ax = b$ where

  1) M approximates A

  2) M^{-1} is easy to apply

- Example: incomplete LU factorization (ILU)

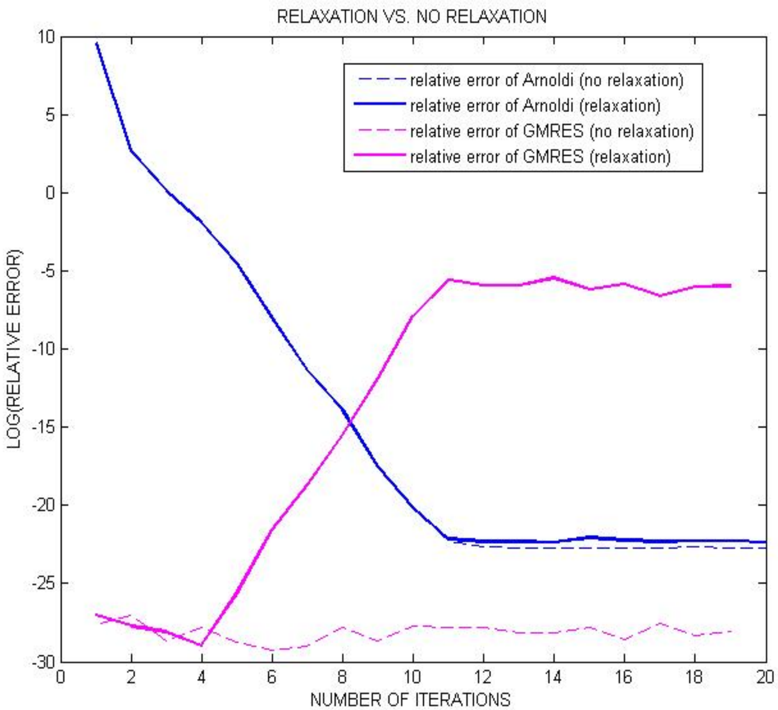
  - Basic idea: drop certain entries in the complete LU factors of the matrix, eg., entries < some threshold

  - ILU(0): select the allowed fill to exactly match the sparsity pattern of the original matrix $\boldsymbol{A}$

Computational Result

Example: Olmstead model, $N = 5000$, $R = 4.7$, $\lambda_1 = 4.5102$, $\sigma = 5$

solver: GMRES with ILU(0)



Bouras and *Fraysse*'s relaxation strategy[6]:

$$tolerance_{GMRES} = 10^{-2}\boldsymbol{\varepsilon} / \|r_{k-1}\|_2$$

Method	No relaxation	Relaxation with BF
Number of GMRES iteration at every outer iteration	9	11
	9	10
	9	9
	9	9
	9	8
	9	7
	9	6
	9	5
	9	4
	9	3
	9	2
	8	2
	9	2
	9	2
	9	2
	9	2
	8	2
	8	2
	8	2
Run time (sec.)	2.25	1.87
Relative error	1.3918e-010	1.9595e-010

Application in The Study of Dynamical Equilibrium

The detection of:

multiple steady states

change of stability

Hopf bifurcation phenomena

Tubular Reactor Model

- Tubular reactor model[7]

$$\begin{cases} \dfrac{\partial y}{\partial t} = \dfrac{1}{Pe_m}\dfrac{\partial^2 y}{\partial s^2} - \dfrac{\partial y}{\partial s} - Dy\exp(\gamma - \gamma/\theta) \\ \dfrac{\partial \theta}{\partial t} = \dfrac{1}{Pe_h}\dfrac{\partial^2 \theta}{\partial s^2} - \dfrac{\partial \theta}{\partial s} - \beta(\theta - \theta_0) + BDy\exp(\gamma - \gamma/\theta) \end{cases} \quad on \quad s \in (0,1)$$

with boundary condition

$$\frac{\partial y}{\partial s} = Pe_m(y-1), \frac{\partial \theta}{\partial s} = Pe_h(\theta - 1) \quad at\ s = 0$$

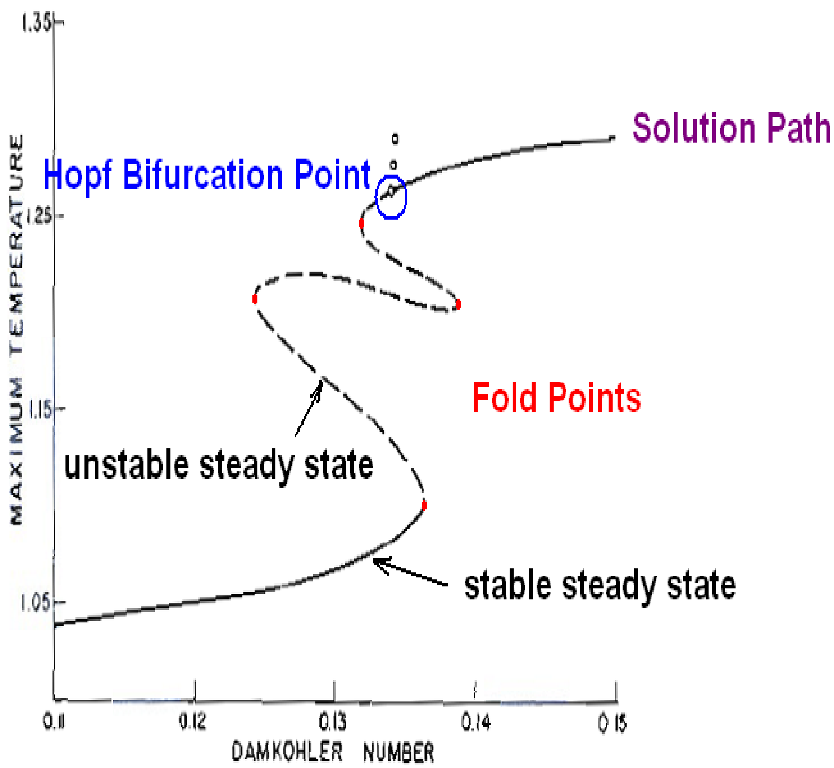
$$\frac{\partial y}{\partial s} = \frac{\partial \theta}{\partial s} = 0 \quad at\ s = 1$$

$\boldsymbol{y}$: velocity; $\boldsymbol{\theta}$: temperature;

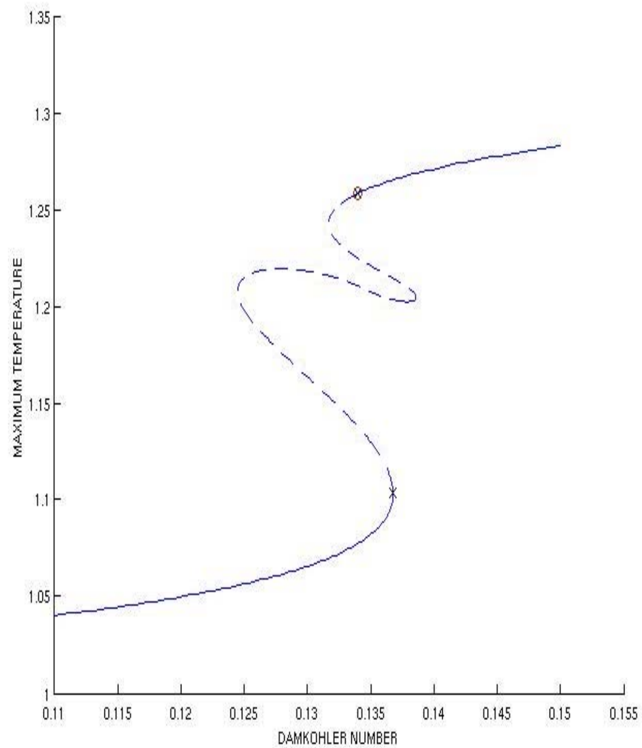
$\boldsymbol{Pe_m}$, $\boldsymbol{Pe_h}$, $\boldsymbol{B}$, $\boldsymbol{D}$, $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$: parameters. $\boldsymbol{D}$: *Damkohler* number

Computation of Solution Path[8]

Published Result[7]

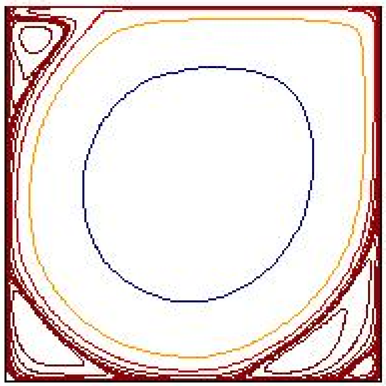


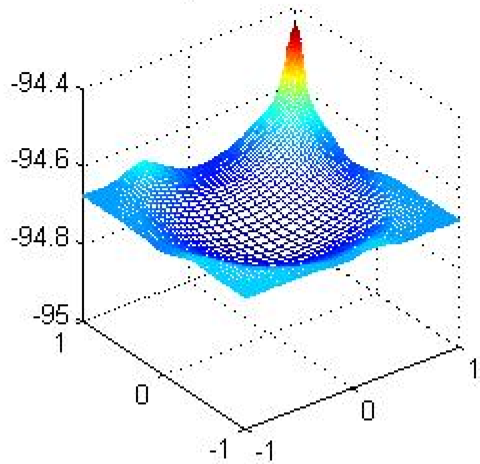
Computed Result



$$Pe_m = Pe_h = 5, B = 0.5, \gamma = 25, \beta = 25$$

Eigenvalue Problem From *Navier-Stokes* equations





2D driven-cavity problem

Reynolds number and stability

Detection of eigenvalues with large imaginary part

2D Driven-Cavity Problem

- Equations[5]:

$$u_t - \upsilon \nabla^2 \vec{u} + \vec{u} \cdot \nabla \vec{u} + \nabla p = 0$$

$$\nabla \cdot \vec{u} = 0$$

$u_x = 1 - x^4$ at $y = 1$.

$\vec{u} = \left(u_x, u_y\right)$: velocity, p: pressure, $\upsilon > 0$: kinematic viscosity.

- Reynolds number:

$$R = \frac{UL}{\upsilon}$$

A quantitative measure of the relative contributions of viscous diffusion and convection.

Reynolds Number And Stability

Reynolds Number	Rightmost Eigenvalues
7500	-0.0044
7600	-0.0043
7700	-0.0043
7800	-0.0042
7900	-0.0042
8000	**-0.0041**
8100	**-0.0038 ± 1.2926i**
8200	-0.0020 ± 1.2932i
8300	**-0.0002 ± 1.2937i**
8400	**0.0017 ± 1.2932i**
8500	0.0037 ± 1.2947i

Q_2-Q_1 macroelement, $h = 2^{-4}$, nonlinear residual<10^{-10}

- Rightmost eigenvalues change from real to complex at R around 8100.
- As R increases, steady state solution loses its stability.
- *Hopf* bifurcation happens at R around 8300.
- Imaginary parts are much larger than real parts (in modulus).

Conclusions

- What do we have
  - *Arnoldi* and Implicitly Restarted *Arnoldi* code
  - *Arnoldi* with iterative linear system solver
  - Codes for discretizing several commonly used benchmark problems
  - Continuation code for single-parameter nonlinear dynamical systems
  - Validated results for various computational tasks performed on these benchmark problems

- What's next
  - Detection of eigenvalues with large imaginary parts
  - Preconditioning of *Arnoldi* applied to *Navier-Stokes* equations
  - Look at a *Navier-Stokes* equation with low critical Reynolds number

Acknowledgement

- The Implicitly Restarted *Arnoldi* code is written by ***Fei Xue***, a PhD student of Department of Mathematics, University of Maryland.

- The software we used to discretize the Navier-Stokes equation and compute its steady state solution is **IFISS**, Incompressible Flow Iterative Solution Software, developed by:

Howard Elman

Department of Computer Science, University of Maryland

David Silvester

Department of Mathematics, University of Manchester

Andy Wathern

Oxford University, Computing Laboratory

Reference

[1] Meerbergen, K & Roose, D 1996 Matrix transformation for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems. *SIAM J. Numer. Anal.* 16, 297-346.

[2] Olmstead, W. E., Davis, W. E., Rosenblat, S. H., & Kath, W. L. 1986 Bifurcation with memory. *SIAM J.Appl. Math.* 40, 171-188.

[3] Stewart, G. W. 2001 Matrix algorithms, volume II: eigensystems. *SIAM*.

[4] Meergergen, K & Spence, A 1997 Implicitly restarted Arnoldi with purification for the shift-invert transformation. *Math. Comput.* 66, 667-698.

[5] Elman, H & Silvester, D & Wathen, A 2005 Finite elements and fast iterative solvers with applications in incompressible fluid dynamics. Oxford University Press

[6] Simoncini, V 2005 Variable accuracy of matrix-vector products in projection methods for eigencomputation. *SIAM J. Numer. Anal.* 43, 1155-1174.

[7] Heinemann, R & Poore, A 1981 Multiplicity, stability, and oscilatory dynamics of the tubular reactor.Chemical Engineering Science, 36 1411-1419.

[8] Spence, A & Graham, Ivan G., Numerical Methods for Bifurcation Problems.