Theory and Numerics of Model Reduction

Web page:
http://www.math.ethz.ch/~kressner/modred.php

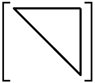
Martin Gutknecht and Daniel Kressner

Existing solution methods for Lyapunov equations

- Medium/dense: Schur form + substitution [Bartels/Stewart'72] [Hammarling'82], [Jonsson/Kågström'02], [Sorensen/Zhou'03], [Quintana-Ortí/van de Geijn'03], [Granat'05], [Kressner'06].
- Medium/dense: sign function iteration [Roberts'71], [Beavers/Denman'75], [Byers'87], [Benner/Quintana-Ortí'99], [Grasedyck et al.'03], [Baur/Benner'04].
- Large/sparse, factorizing $A - \sigma I$ is affordable: ADI [Wachspress'88], [Penzl'99], [Li/White'02], [Gugercin et al.'03], [Simoncini'06] a.m.o.; API [Hodel et al.'96].
- Large/sparse, factorizing $A - \sigma I$ is *not* affordable: Krylov subspace methods [Saad'90], [Hu/Reichel'92], [Jaimoukha/Kasenally'94], [Hochbruck/Starke'95], a.o.

Bartels/Stewart method

1. Schur decomposition: Compute orthogonal matrix Q such that

   $$Q^T A Q =$$ 

   ($\Lambda(A) \subset \mathbb{R}$ to simplify presentation). Then $AX + XA^T = -C \rightsquigarrow$

   $$(Q^T AQ)(Q^T XQ) + (Q^T XQ)(Q^T AQ)^T = -Q^T CQ.$$

2. Substitution: Solve triangular Lyapunov equation

   $$\widetilde{A}\widetilde{X} + \widetilde{X}\widetilde{A}^T = -\widetilde{C}$$

   where $\widetilde{A} = Q^T AQ$ and $\widetilde{C} = Q^T CQ$.
3. Back transform $X = Q\widetilde{X}Q^T$.

Solution of triangular Lyapunov equation

If $n = 1$:

$$X = -\frac{C}{2A}.$$

If $n > 1$: Partitioning

$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} + \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \begin{bmatrix} A_{11}^T & 0 \\ A_{12}^T & A_{22}^T \end{bmatrix} = - \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

yields coupled equations (solved recursively)

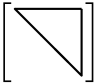
$$\begin{aligned} A_{22}X_{22} + X_{22}A_{22}^T &= -C_{22}, & (1) \\ A_{22}X_{21} + X_{21}A_{11}^T &= -C_{21} - X_{22}A_{12}^T, & (2) \\ A_{11}X_{12} + X_{12}A_{22}^T &= -C_{12} - A_{12}X_{22}, & (3) \\ A_{11}X_{11} + X_{11}A_{11}^T &= -C_{11} - A_{12}X_{21} - X_{12}A_{12}^T. & (4) \end{aligned}$$

SLICOT/MATLAB's `lyap` use $A_{11} \in \mathbb{R}^{(n-1)\times(n-1)}$, $A_{22} \in \mathbb{R}^{1\times 1}$.
RECSY (much more efficient) uses $A_{11}, A_{22} \in \mathbb{R}^{n/2\times n/2}$.

Hammarling's method

Computes factorized solution $X = RR^T$ of $AX + XA^T = -BB^T$ directly.
(Requires less operations and is more numerically stable.)

1. Schur decomposition: Compute orthogonal matrix Q such that

   $$Q^T AQ =$$ 

   ($\Lambda(A) \subset \mathbb{R}$ to simplify presentation).
2. Substitution: Solve factorized triangular Lyapunov equation

   $$\widetilde{A}\widetilde{R}\widetilde{R}^T + \widetilde{R}\widetilde{R}^T\widetilde{A}^T = -\widetilde{B}\widetilde{B}^T,$$

   where $\widetilde{A} = Q^T AQ$ and $\widetilde{B} = Q^T B$.
3. Back transform $R = Q\widetilde{R}$.

Solution of factorized triangular Lyapunov equation

If $n = 1$:

$$\widetilde{R} = \sqrt{-\frac{\widetilde{B}\widetilde{B}^T}{2\widetilde{A}}}.$$

If $n > 1$: Partitioning

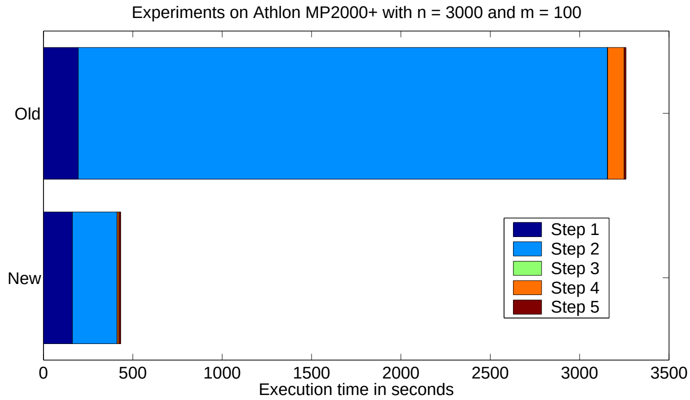
$$\widetilde{A} = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}, \quad \widetilde{R} = \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

yields coupled equations (solved recursively)

$$\begin{aligned} A_{22}R_{22}R_{22}^T + R_{22}R_{22}^TA_{22}^T &= -B_2B_2^T, & (5) \\ A_{11}R_{12} + R_{12}(R_{22}^{-1}A_{22}R_{22})^T &= -B_1B_2^TR_{22}^{-T} - A_{12}R_{22}, & (6) \\ A_{11}R_{11}R_{11}^T + R_{11}R_{11}^TA_{11}^T &= -\widetilde{B}_1\widetilde{B}_1^T. & (7) \end{aligned}$$

SLICOT/MATLAB's `lyapchol` use $A_{11} \in \mathbb{R}^{(n-1)\times(n-1)}$, $A_{22} \in \mathbb{R}^{1\times 1}$.

Overall execution time



Step 1 $A \rightsquigarrow$ Hessenberg H (Old: LAPACK 3, New: [Quintana-Ortí/Van de Geijn'06])

Step 2 $H \rightsquigarrow$ Schur $\widetilde{A}$ (Old: LAPACK 3, New: [Braman/Byers/Mathias'02])

Step 4 Solution of $\widetilde{A}X + \widetilde{X}\widetilde{A}^T = \widetilde{B}\widetilde{B}^T$ (Old: SLICOT, New: [Kressner'06])

The sign function iteration

Sign function iteration for

$$AX + XA^T = -BB^T$$

with stable matrix A.

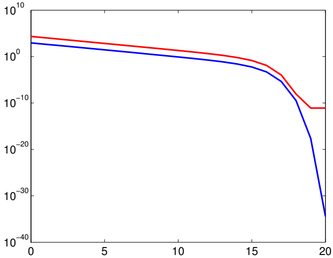
$A_0 \leftarrow A, \quad B_0 \leftarrow B$
for $j \leftarrow 0, 1, 2, \ldots$

1. $A_{j+1} = \frac{1}{2}(A_j + A_j^{-1})$
2. $B_{j+1} = \frac{1}{\sqrt{2}}[B_j, A_j^{-1}B_j]$
3. If $\|A_j - I\| < \text{tol}$ then exit.

Converges globally and locally quadratic.

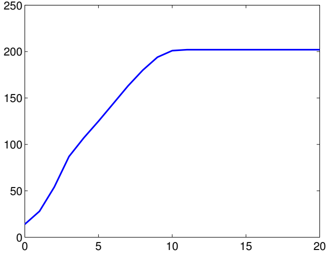
To avoid excessive growth of columns of B_j: truncate by low-rank approximation.

Convergence of sign function iteration: steel ex.



$\|X - B_j B_j^T/2\|_F / \|X\|_F$

$\|A + I\|_F / \sqrt{n}$



columns of B_j

The globalized sign function iteration

Slow initial convergence.

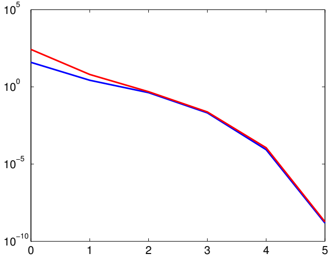
Basic idea: choose c_j such that spectrum of $\frac{1}{2}(c_j A_j + \frac{1}{c_j} A_j^{-1})$ is as narrow as possible $\rightsquigarrow c_j = \sqrt{\|A_j^{-1}\|_2 / \|A_j\|_2}$ (suboptimal choice).

$A_0 \leftarrow A, \quad B_0 \leftarrow B$
for $j \leftarrow 0, 1, 2, \ldots$

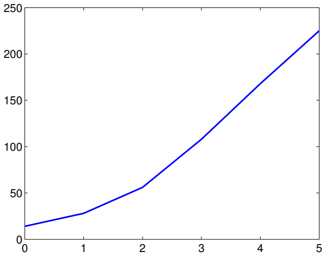
1. $A_{j+1} = \frac{1}{2}(c_j A_j + \frac{1}{c_j} A_j^{-1})$
2. $B_{j+1} = \frac{1}{\sqrt{2}} \left[\sqrt{c_j} B_j, \frac{1}{\sqrt{c_j}} A_j^{-1} B_j \right]$
3. If $\|A_j - I\| < \text{tol}$ then exit.

Converges globally (better) and locally quadratic.

Convergence of sign function iteration: steel ex.



$\|X - B_j B_j^T/2\|_F / \|X\|_F$
$\|A + I\|_F / \sqrt{n}$



$\#$ columns of B_j