ADI-BASED METHODS FOR ALGEBRAIC LYAPUNOV AND RICCATI EQUATIONS

Peter Benner

Professur Mathematik in Industrie und Technik
Fakultät für Mathematik
Technische Universität Chemnitz

Partially based on joint work with Jens Saak, Martin Köhler (both TU Chemnitz), Ninoslav Truhar (U Osijek), and Ren-Cang Li (UT Arlington)

CICADA/MIMS Workshop on
Numerics for Control and Simulation
Manchester, June 17, 2009

Overview

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n\times n}$ given and $X \in \mathbb{R}^{n\times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \implies$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\implies 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^TC\}$, where $B \in \mathbb{R}^{n\times m}$, $m \ll n$, $C \in \mathbb{R}^{p\times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n \times n}$ given and $X \in \mathbb{R}^{n \times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \Longrightarrow$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\Longrightarrow 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^T C\}$, where $B \in \mathbb{R}^{n \times m}$, $m \ll n$, $C \in \mathbb{R}^{p \times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n \times n}$ given and $X \in \mathbb{R}^{n \times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \Longrightarrow$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\Longrightarrow 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^T C\}$, where $B \in \mathbb{R}^{n \times m}$, $m \ll n$, $C \in \mathbb{R}^{p \times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n\times n}$ given and $X \in \mathbb{R}^{n\times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \Longrightarrow$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\Longrightarrow 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^TC\}$, where $B \in \mathbb{R}^{n\times m}$, $m \ll n$, $C \in \mathbb{R}^{p\times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n \times n}$ given and $X \in \mathbb{R}^{n \times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \Longrightarrow$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\Longrightarrow 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^T C\}$, where $B \in \mathbb{R}^{n \times m}$, $m \ll n$, $C \in \mathbb{R}^{p \times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

General form of algebraic Riccati equation (ARE) for $A, G = G^T, W = W^T \in \mathbb{R}^{n\times n}$ given and $X \in \mathbb{R}^{n\times n}$ unknown:

$$0 = \mathcal{R}(X) := A^T X + XA - XGX + W.$$

$G = 0 \Longrightarrow$ Lyapunov equation:

$$0 = \mathcal{L}(X) := A^T X + XA + W.$$

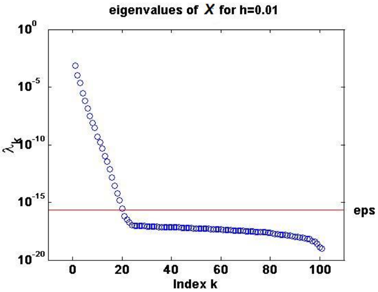
Typical situation in model reduction and optimal control problems for semi-discretized PDEs:

- $n = 10^3 - 10^6$ ($\Longrightarrow 10^6 - 10^{12}$ unknowns!),
- A has sparse representation ($A = -M^{-1}S$ for FEM),
- G, W low-rank with $G, W \in \{BB^T, C^T C\}$, where $B \in \mathbb{R}^{n\times m}$, $m \ll n$, $C \in \mathbb{R}^{p\times n}$, $p \ll n$.
- Standard (eigenproblem-based) $\mathcal{O}(n^3)$ methods are not applicable!

Consider spectrum of ARE solution (analogous for Lyapunov equations).

Example:

- Linear 1D heat equation with point control,
- $\Omega = [0, 1]$,
- FEM discretization using linear B-splines,
- $h = 1/100 \implies n = 101$.



Idea: $X = X^T \geq 0 \implies$

$$X = ZZ^T = \sum_{k=1}^{n} \lambda_k z_k z_k^T \approx Z^{(r)}(Z^{(r)})^T = \sum_{k=1}^{r} \lambda_k z_k z_k^T.$$

$\implies$ Goal: compute $Z^{(r)} \in \mathbb{R}^{n \times r}$ directly w/o ever forming X!

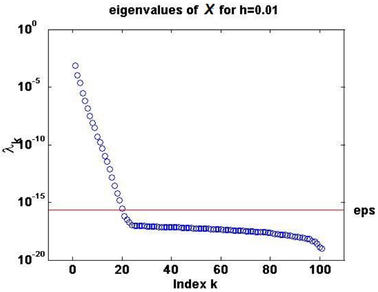
Large-Scale Matrix Equtions

Low-Rank Approximation

Consider spectrum of ARE solution (analogous for Lyapunov equations).

Example:

- Linear 1D heat equation with point control,
- $\Omega = [0, 1]$,
- FEM discretization using linear B-splines,
- $h = 1/100 \Longrightarrow n = 101$.



Idea: $X = X^T \geq 0 \;\Longrightarrow$

$$X = ZZ^T = \sum_{k=1}^{n} \lambda_k z_k z_k^T \approx Z^{(r)}(Z^{(r)})^T = \sum_{k=1}^{r} \lambda_k z_k z_k^T .$$

$\Longrightarrow$ Goal: compute $Z^{(r)} \in \mathbb{R}^{n \times r}$ directly w/o ever forming X!

Numerical solution of linear-quadratic optimal control problem for parabolic PDEs via Galerkin approach, spatial FEM discretization $\rightsquigarrow$

LQR Problem (finite-dimensional)

$$\text{Min } \mathcal{J}(u) = \frac{1}{2}\int_0^\infty \left(y^T Q y + u^T R u\right) dt \quad \text{for} \quad u \in \mathcal{L}_2(0,\infty;\mathbb{R}^m),$$

subject to $M\dot{x} = -Sx + Bu, \quad x(0) = x_0, \quad y = Cx,$
with stiffness $S \in \mathbb{R}^{n\times n}$, mass $M \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$.

Solution of finite-dimensional LQR problem: feedback control

$$u_*(t) = -B^T X_* x(t) =: -K_* x(t),$$

where $X_* = X_*^T \geq 0$ is the unique stabilizing[1] solution of the ARE

$$0 = \mathcal{R}(X) := C^T C + A^T X + XA - XBB^T X,$$

with $A := -M^{-1}S$, $B := M^{-1}BR^{-\frac{1}{2}}$, $C := CQ^{-\frac{1}{2}}$.

[1] X is stabilizing $\Leftrightarrow \Lambda(A - BB^T X) \subset \mathbb{C}^-$.

Numerical solution of linear-quadratic optimal control problem for parabolic PDEs via Galerkin approach, spatial FEM discretization $\rightsquigarrow$

LQR Problem (finite-dimensional)

$$\text{Min } \mathcal{J}(u) = \frac{1}{2}\int_0^\infty \left(y^T Q y + u^T R u\right) dt \quad \text{for} \quad u \in \mathcal{L}_2(0,\infty;\mathbb{R}^m),$$

subject to $M\dot{x} = -Sx + Bu, \quad x(0) = x_0, \quad y = Cx,$
with stiffness $S \in \mathbb{R}^{n\times n}$, mass $M \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$.

Solution of finite-dimensional LQR problem: feedback control

$$u_*(t) = -B^T X_* x(t) =: -K_* x(t),$$

where $X_* = X_*^T \geq 0$ is the unique stabilizing[1] solution of the ARE

$$0 = \mathcal{R}(X) := C^T C + A^T X + XA - XBB^T X,$$

with $A := -M^{-1}S$, $B := M^{-1}BR^{-\frac{1}{2}}$, $C := CQ^{-\frac{1}{2}}$.

[1] X is stabilizing $\Leftrightarrow \Lambda(A - BB^T X) \subset \mathbb{C}^-$.

Linear, Time-Invariant (LTI) Systems

$$\Sigma : \quad \left\{ \begin{array}{rcl} \dot{x}(t) & = & Ax + Bu, \\ y(t) & = & Cx + Du, \end{array} \right. \qquad \begin{array}{ll} A \in \mathbb{R}^{n \times n}, & B \in \mathbb{R}^{n \times m}, \\ C \in \mathbb{R}^{p \times n}, & D \in \mathbb{R}^{p \times m}. \end{array}$$

(A, B, C, D) is a realization of Σ (nonunique).

Linear, Time-Invariant (LTI) Systems

$$\Sigma : \begin{cases} \dot{x}(t) = Ax + Bu, & A \in \mathbb{R}^{n\times n}, \quad B \in \mathbb{R}^{n\times m}, \\ y(t) = Cx + Du, & C \in \mathbb{R}^{p\times n}, \quad D \in \mathbb{R}^{p\times m}. \end{cases}$$

(A, B, C, D) is a realization of Σ (nonunique).

Model Reduction Based on Balancing

Given $P, Q \in \mathbb{R}^{n\times n}$ symmetric positive definite (spd), and a contragredient transformation $T : \mathbb{R}^n \to \mathbb{R}^n$,

$$TPT^T = T^{-T}QT^{-1} = \mathrm{diag}(\sigma_1, \ldots, \sigma_n), \quad \sigma_1 \geq \ldots \geq \sigma_n \geq 0.$$

Balancing Σ w.r.t. P, Q:

$$\Sigma \equiv (A, B, C, D) \mapsto (TAT^{-1}, TB, CT^{-1}, D) \equiv \Sigma.$$

Model Reduction Based on Balancing

Given $P, Q \in \mathbb{R}^{n \times n}$ symmetric positive definite (spd), and a contragredient transformation $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$,

$$TPT^T = T^{-T}QT^{-1} = \mathrm{diag}(\sigma_1, \ldots, \sigma_n), \quad \sigma_1 \geq \ldots \geq \sigma_n \geq 0.$$

Balancing Σ w.r.t. P, Q:

$$\Sigma \equiv (A, B, C, D) \mapsto (TAT^{-1}, TB, CT^{-1}, D) \equiv \Sigma.$$

For Balanced Truncation: $P/Q =$ controllability/observability Gramian of Σ, i.e., for asymptotically stable systems, P, Q solve dual Lyapunov equations

$$AP + PA^T + BB^T \ = \ 0, \qquad A^TQ + QA + C^TC \ = \ 0.$$

Basic Model Reduction Procedure

1. Given $\Sigma \equiv (A, B, C, D)$ and balancing (w.r.t. given P, Q spd) transformation $T \in \mathbb{R}^{n\times n}$ nonsingular, compute

$$\begin{aligned}(A, B, C, D) &\mapsto (TAT^{-1}, TB, CT^{-1}, D)\\ &= \left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22}\end{bmatrix}, \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \begin{bmatrix} C_1 & C_2 \end{bmatrix}, D\right)\end{aligned}$$

2. Truncation $\rightsquigarrow$ reduced-order model:

$$(\hat{A}, \hat{B}, \hat{C}, \hat{D}) = (A_{11}, B_1, C_1, D).$$

Basic Model Reduction Procedure

1. Given $\Sigma \equiv (A, B, C, D)$ and balancing (w.r.t. given P, Q spd) transformation $T \in \mathbb{R}^{n \times n}$ nonsingular, compute

$$\begin{aligned}(A, B, C, D) &\mapsto (TAT^{-1}, TB, CT^{-1}, D)\\ &= \left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \begin{bmatrix} C_1 & C_2 \end{bmatrix}, D \right)\end{aligned}$$

2. Truncation $\rightsquigarrow$ reduced-order model:

$$(\hat{A}, \hat{B}, \hat{C}, \hat{D}) = (A_{11}, B_1, C_1, D).$$

Implementation: SR Method

1. Given Cholesky (square) or (low-rank approximation to) full-rank (maybe rectangular, "thin") factors of P, Q

$$P = S^T S, \quad Q = R^T R.$$

2. Compute SVD

$$SR^T = [\, U_1,\ U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$$

3. Set

$$W = R^T V_1 \Sigma_1^{-1/2}, \qquad V = S^T U_1 \Sigma_1^{-1/2}.$$

4. Reduced-order model is

$$(\hat{A}, \hat{B}, \hat{C}, \hat{D}) := (W^T AV, W^T B, CV, D) \ \ (\equiv (A_{11}, B_1, C_1, D).)$$

ADI Method for Lyapunov Equations

Background

Recall Peaceman Rachford ADI:
Consider $Au = s$ where $A \in \mathbb{R}^{n\times n}$ spd, $s \in \mathbb{R}^n$. ADI Iteration Idea:
Decompose $A = H + V$ with $H, V \in \mathbb{R}^{n\times n}$ such that

$$
\begin{aligned}
(H + pI)v &= r \\
(V + pI)w &= t
\end{aligned}
$$

can be solved easily/efficiently.

ADI Iteration

If H, V spd $\Rightarrow \exists p_k,\ k = 1, 2, \ldots$ such that

$$
\begin{aligned}
u_0 &= 0 \\
(H + p_k I)u_{k-\frac{1}{2}} &= (p_k I - V)u_{k-1} + s \\
(V + p_k I)u_k &= (p_k I - H)u_{k-\frac{1}{2}} + s
\end{aligned}
$$

converges to $u \in \mathbb{R}^n$ solving $Au = s$.

ADI Method for Lyapunov Equations

Background

Recall Peaceman Rachford ADI:
Consider $Au = s$ where $A \in \mathbb{R}^{n\times n}$ spd, $s \in \mathbb{R}^n$. ADI Iteration Idea:
Decompose $A = H + V$ with $H, V \in \mathbb{R}^{n\times n}$ such that

$$(H + pI)v = r$$
$$(V + pI)w = t$$

can be solved easily/efficiently.

ADI Iteration

If H, V spd $\Rightarrow \exists p_k,\ k = 1, 2, \ldots$ such that

$$\begin{aligned} u_0 &= 0 \\ (H + p_k I)u_{k-\frac{1}{2}} &= (p_k I - V)u_{k-1} + s \\ (V + p_k I)u_k &= (p_k I - H)u_{k-\frac{1}{2}} + s \end{aligned}$$

converges to $u \in \mathbb{R}^n$ solving $Au = s$.

The Lyapunov operator

$$\mathcal{L}: \quad P \quad \mapsto \quad AX + XA^T$$

can be decomposed into the linear operators

$$\mathcal{L}_H : X \mapsto AX \qquad \mathcal{L}_V : X \mapsto XA^T.$$

In analogy to the standard ADI method we find the

ADI iteration for the Lyapunov equation [Wachspress 1988

$$\begin{aligned} P_0 &= 0 \\ (A + p_k I)X_{k-\frac{1}{2}} &= -W - P_{k-1}(A^T - p_k I) \\ (A + p_k I)X_k^T &= -W - X_{k-\frac{1}{2}}^T (A^T - p_k I) \end{aligned}$$

Low-Rank ADI for Lyapunov equations

- For $A \in \mathbb{R}^{n\times n}$ stable, $B \in \mathbb{R}^{n\times m}$ ($w \ll n$), consider Lyapunov equation

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

- ADI Iteration: [Wachspress 1988]

$$\begin{aligned}(A + p_k I)X_{k-\frac{1}{2}} &= -BB^T - X_{k-1}(A^T - p_k I)\\ (A + \overline{p_k} I){X_k}^T &= -BB^T - X_{k-\frac{1}{2}}(A^T - \overline{p_k} I)\end{aligned}$$

with parameters $p_k \in \mathbb{C}^-$ and $p_{k+1} = \overline{p_k}$ if $p_k \notin \mathbb{R}$.

- For $X_0 = 0$ and proper choice of p_k: $\lim_{k\to\infty} X_k = X$ superlinear.
- Re-formulation using $X_k = Y_k Y_k^T$ yields iteration for Y_k...

Low-Rank ADI for Lyapunov equations

- For $A \in \mathbb{R}^{n \times n}$ stable, $B \in \mathbb{R}^{n \times m}$ ($w \ll n$), consider Lyapunov equation

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

- ADI Iteration: [Wachspress 1988]

$$\begin{aligned}(A + p_k I)X_{k-\frac{1}{2}} &= -BB^T - X_{k-1}(A^T - p_k I)\\ (A + \overline{p_k} I){X_k}^T &= -BB^T - X_{k-\frac{1}{2}}(A^T - \overline{p_k} I)\end{aligned}$$

with parameters $p_k \in \mathbb{C}^-$ and $p_{k+1} = \overline{p_k}$ if $p_k \notin \mathbb{R}$.

- For $X_0 = 0$ and proper choice of p_k: $\lim\limits_{k \to \infty} X_k = X$ superlinear.

- Re-formulation using $X_k = Y_k Y_k^T$ yields iteration for Y_k...

Setting $X_k = Y_k Y_k^T$, some algebraic manipulations $\Longrightarrow$

Algorithm [Penzl '97/'00, Li/White '99/'02, B. 04, B./Li/Penzl '99/'08]

$$V_1 \leftarrow \sqrt{-2\mathrm{Re}\,(p_1)}(A + p_1 I)^{-1}B, \qquad Y_1 \leftarrow V_1$$

FOR $k = 2, 3, \ldots$

$$V_k \leftarrow \sqrt{\frac{\mathrm{Re}\,(p_k)}{\mathrm{Re}\,(p_{k-1})}} \left(V_{k-1} - (p_k + \overline{p_{k-1}})(A + p_k I)^{-1} V_{k-1}\right)$$

$$Y_k \leftarrow \left[\begin{array}{cc} Y_{k-1} & V_k \end{array}\right]$$

$$Y_k \leftarrow \mathrm{rrlq}(Y_k, \tau) \qquad \text{\% column compression}$$

At convergence, $Y_{k_{\max}} Y_{k_{\max}}^T \approx X$, where (without column compression)

$$Y_{k_{\max}} = \left[\begin{array}{ccc} V_1 & \ldots & V_{k_{\max}} \end{array}\right], \quad V_k = \left[\,\right] \in \mathbb{C}^{n \times m}.$$

Note: Implementation in real arithmetic possible by combining two steps.

Setting $X_k = Y_k Y_k^T$, some algebraic manipulations $\Longrightarrow$

Algorithm [Penzl '97/'00, Li/White '99/'02, B. 04, B./Li/Penzl '99/'08]

$$V_1 \leftarrow \sqrt{-2\mathrm{Re}\,(p_1)}(A + p_1 I)^{-1}B, \qquad Y_1 \leftarrow V_1$$

FOR $k = 2, 3, \ldots$

$$V_k \leftarrow \sqrt{\frac{\mathrm{Re}\,(p_k)}{\mathrm{Re}\,(p_{k-1})}}\left(V_{k-1} - (p_k + \overline{p_{k-1}})(A + p_k I)^{-1}V_{k-1}\right)$$

$$Y_k \leftarrow \begin{bmatrix} Y_{k-1} & V_k \end{bmatrix}$$

$$Y_k \leftarrow \mathrm{rrlq}(Y_k, \tau) \qquad \text{\% column compression}$$

At convergence, $Y_{k_{\max}} Y_{k_{\max}}^T \approx X$, where (without column compression)

$$Y_{k_{\max}} = \begin{bmatrix} V_1 & \ldots & V_{k_{\max}} \end{bmatrix}, \qquad V_k = \big[\,\big] \in \mathbb{C}^{n \times m}.$$

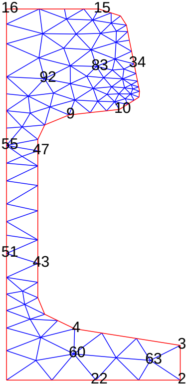
Note: Implementation in real arithmetic possible by combining two steps.

- Mathematical model: boundary control for linearized 2D heat equation.

$$
\begin{aligned}
c \cdot \rho \frac{\partial}{\partial t} x &= \lambda \Delta x, & \xi \in \Omega \\
\lambda \frac{\partial}{\partial n} x &= \kappa (u_k - x), & \xi \in \Gamma_k,\ 1 \le k \le 7, \\
\frac{\partial}{\partial n} x &= 0, & \xi \in \Gamma_7.
\end{aligned}
$$

$\Longrightarrow m = 7, p = 6.$

- FEM Discretization, different models for initial mesh ($n = 371$), 1, 2, 3, 4 steps of mesh refinement $\Rightarrow$ $n = 1357, 5177, 20209, 79841$.



Source: Physical model: courtesy of Mannesmann/Demag.

Math. model: Tröltzsch/Unger 1999/2001, Penzl 1999, Saak 2003.

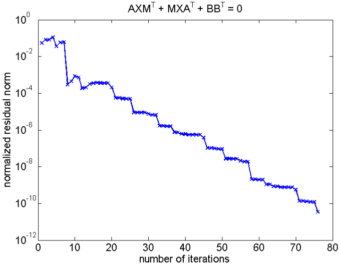
Numerical Results

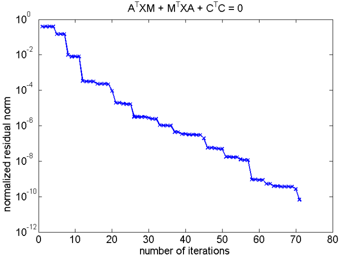
Computations by Jens Saak

- Solve dual Lyapunov equations needed for balanced truncation, i.e.,

$$APM^T + MPA^T + BB^T = 0, \qquad A^TQM + M^TQA + C^TC = 0,$$

  for 79,841. Note: $m = 7, p = 6$.
- 25 shifts chosen by Penzl's heuristic from 50/25 Ritz values of A of largest/smallest magnitude, no column compression performed.
- New version in MESS (Matrix Equations Sparse Solvers) requires no factorization of mass matrix!
- Computations done on Core2Duo at 2.8GHz with 3GB RAM and 32Bit-MATLAB.





Recent Numerical Results

Computations by Martin Köhler

- $A \in \mathbb{R}^{n \times n} \equiv$ FDM matrix for 2D heat equation on $[0,1]^2$ (LYAPACK benchmark `demo_l1`, $m = 1$).
- 16 shifts chosen by Penzl's heuristic from 50/25 Ritz values of A of largest/smallest magnitude.
- Computations using 2 dual core Intel Xeon 5160 with 16 GB RAM.

Recent Numerical Results

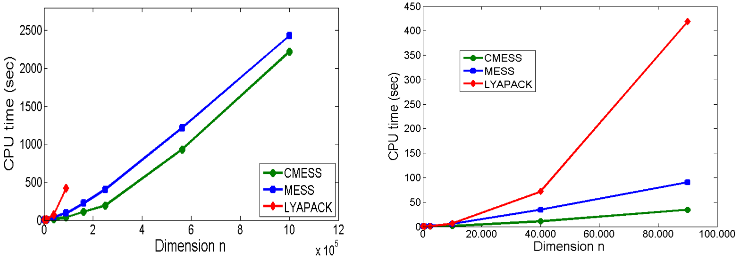
Computations by Martin Köhler

- $A \in \mathbb{R}^{n \times n} \equiv$ FDM matrix for 2D heat equation on $[0, 1]^2$ (LYAPACK benchmark `demo_l1`, $m = 1$).
- 16 shifts chosen by Penzl's heuristic from 50/25 Ritz values of A of largest/smallest magnitude.
- Computations using 2 dual core Intel Xeon 5160 with 16 GB RAM.

CPU Times

n	CMESS	LYAPACK	MESS
100	0.023	0.124	0.158
625	0.042	0.104	0.227
2,500	0.159	0.702	0.989
10,000	0.965	6.22	5.644
40,000	11.09	71.48	34.55
90,000	34.67	418.5	90.49
160,000	109.3	out of memory	219.9
250,000	193.7	out of memory	403.8
562,500	930.1	out of memory	1216.7
1,000,000	2220.0	out of memory	2428.6

- $A \in \mathbb{R}^{n \times n} \equiv$ FDM matrix for 2D heat equation on $[0,1]^2$ (LYAPACK benchmark `demo_l1`, $m = 1$).
- 16 shifts chosen by Penzl's heuristic from 50/25 Ritz values of A of largest/smallest magnitude.
- Computations using 2 dual core Intel Xeon 5160 with 16 GB RAM.



Note: for n=1,000,000, first sparse LU needs $\sim 1,100$ sec., using UMFPACK this reduces to 30 sec. (result of June 15, 2009).

Projection-based methods for Lyapunov equations with $A + A^T < 0$:

1. Compute orthonormal basis $\mathrm{range}(Z)$, $Z \in \mathbb{R}^{n \times r}$, for subspace $\mathcal{Z} \subset \mathbb{R}^n$, $\dim \mathcal{Z} = r$.
2. Set $\hat{A} := Z^T A Z$, $\hat{B} := Z^T B$.
3. Solve small-size Lyapunov equation $\hat{A}\hat{X} + \hat{X}\hat{A}^T + \hat{B}\hat{B}^T = 0$.
4. Use $X \approx Z\hat{X}Z^T$.

Examples:

- Krylov subspace methods, i.e., for $m = 1$:

$$\mathcal{Z} = \mathcal{K}(A, B, r) = \mathrm{span}\{B, AB, A^2B, \ldots, A^{r-1}B\}$$

[Saad '90, Jaimoukha/Kasenally '94, Jbilou '02–'08].

- K-PIK [Simoncini '07],

$$\mathcal{Z} = \mathcal{K}(A, B, r) \cup \mathcal{K}(A^{-1}, B, r).$$

Projection-based methods for Lyapunov equations with $A + A^T < 0$:

1. Compute orthonormal basis $\mathrm{range}(Z)$, $Z \in \mathbb{R}^{n \times r}$, for subspace $\mathcal{Z} \subset \mathbb{R}^n$, $\dim \mathcal{Z} = r$.
2. Set $\hat{A} := Z^T AZ$, $\hat{B} := Z^T B$.
3. Solve small-size Lyapunov equation $\hat{A}\hat{X} + \hat{X}\hat{A}^T + \hat{B}\hat{B}^T = 0$.
4. Use $X \approx Z\hat{X}Z^T$.

Examples:

- Krylov subspace methods, i.e., for $m = 1$:

$$\mathcal{Z} = \mathcal{K}(A, B, r) = \mathrm{span}\{B, AB, A^2B, \ldots, A^{r-1}B\}$$

  [Saad '90, Jaimoukha/Kasenally '94, Jbilou '02–'08].

- K-PIK [Simoncini '07],

$$\mathcal{Z} = \mathcal{K}(A, B, r) \cup \mathcal{K}(A^{-1}, B, r).$$

Projection-based methods for Lyapunov equations with $A + A^T < 0$:

1. Compute orthonormal basis $\mathrm{range}(Z)$, $Z \in \mathbb{R}^{n \times r}$, for subspace $\mathcal{Z} \subset \mathbb{R}^n$, $\dim \mathcal{Z} = r$.
2. Set $\hat{A} := Z^T A Z$, $\hat{B} := Z^T B$.
3. Solve small-size Lyapunov equation $\hat{A}\hat{X} + \hat{X}\hat{A}^T + \hat{B}\hat{B}^T = 0$.
4. Use $X \approx Z\hat{X}Z^T$.

Examples:

- ADI subspace [B./R.-C. Li/Truhar '08]:

$$\mathcal{Z} = \mathrm{colspan}\left[\begin{array}{ccc} V_1, & \ldots, & V_r \end{array}\right].$$

Note:

1. ADI subspace is rational Krylov subspace [J.-R. Li/White '02].
2. Similar approach: ADI-preconditioned global Arnoldi method [Jbilou '08].

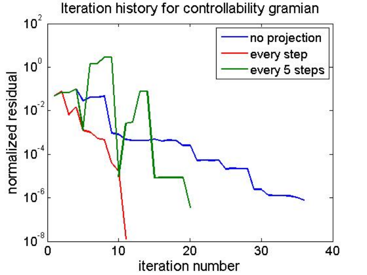
Factored Galerkin-ADI Iteration

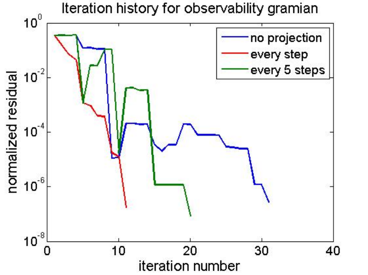
Numerical examples

FEM semi-discretized control problem for parabolic PDE:

- optimal cooling of rail profiles,
- $n = 20,209,\ m = 7,\ p = 6.$

Good ADI shifts





CPU times: 80s (projection every 5th ADI step) vs. 94s (no projection).

Computations by Jens Saak.

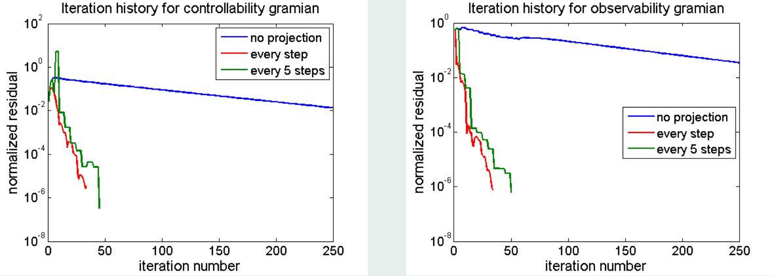
Factored Galerkin-ADI Iteration

Numerical examples

FEM semi-discretized control problem for parabolic PDE:

- optimal cooling of rail profiles,
- $n = 20,209,\ m = 7,\ p = 6.$

Bad ADI shifts



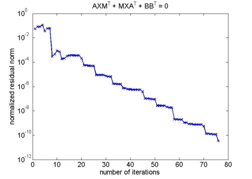
CPU times: 368s (projection every 5th ADI step) vs. 1207s (no projection).

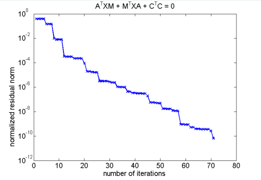
Computations by Jens Saak.

Factored Galerkin-ADI Iteration

Numerical examples: optimal cooling of rail profiles, $n = 79,841$, $m = 7$, $p = 6$.

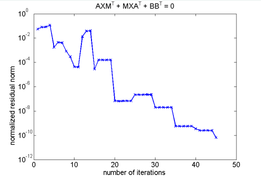
MESS w/ Galerkin projection and column compression

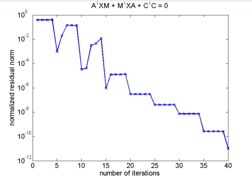




Rank of solution factors: 532 / 426

MESS with Galerkin projection and column compression





Rank of solution factors: 269 / 205

- Consider $0 = \mathcal{R}(X) = C^T C + A^T X + XA - XBB^T X.$
- Frechét derivative of $\mathcal{R}(X)$ at X:

$$\mathcal{R}'_X : Z \to (A - BB^T X)^T Z + Z(A - BB^T X).$$

- Newton-Kantorovich method:

$$X_{j+1} = X_j - \left(\mathcal{R}'_{X_j}\right)^{-1} \mathcal{R}(X_j), \quad j = 0, 1, 2, \dots$$

Newton's method (with line search) for AREs

FOR $j = 0, 1, \dots$

1. $A_j \leftarrow A - BB^T X_j =: A - BK_j$.
2. Solve the Lyapunov equation $A_j^T N_j + N_j A_j = -\mathcal{R}(X_j)$.
3. $X_{j+1} \leftarrow X_j + t_j N_j$.

END FOR j

Newton-ADI for AREs

Newton's Method for AREs [Kleinman '68, Mehrmann '91, Lancaster/Rodman '95, B./Byers '94/'98, B. '97, Guo/Laub '99]

- Consider $0 = \mathcal{R}(X) = C^TC + A^TX + XA - XBB^TX.$
- Frechét derivative of $\mathcal{R}(X)$ at X:

$$\mathcal{R}'_X : Z \to (A - BB^TX)^TZ + Z(A - BB^TX).$$

- Newton-Kantorovich method:

$$X_{j+1} = X_j - \left(\mathcal{R}'_{X_j}\right)^{-1} \mathcal{R}(X_j), \quad j = 0, 1, 2, \ldots$$

Newton's method (with line search) for AREs

FOR $j = 0, 1, \ldots$

1. $A_j \leftarrow A - BB^TX_j =: A - BK_j.$
2. Solve the Lyapunov equation $A_j^TN_j + N_jA_j = -\mathcal{R}(X_j).$
3. $X_{j+1} \leftarrow X_j + t_jN_j.$

END FOR j

Newton's Method for AREs [Kleinman '68, Mehrmann '91, Lancaster/Rodman '95, B./Byers '94/'98, B. '97, Guo/Laub '99]

- Consider $0 = \mathcal{R}(X) = C^T C + A^T X + XA - XBB^T X.$
- Frechét derivative of $\mathcal{R}(X)$ at X:

$$\mathcal{R}'_X : Z \rightarrow (A - BB^T X)^T Z + Z(A - BB^T X).$$

- Newton-Kantorovich method:

$$X_{j+1} = X_j - \left(\mathcal{R}'_{X_j}\right)^{-1} \mathcal{R}(X_j), \quad j = 0, 1, 2, \ldots$$

Newton's method (with line search) for AREs

FOR $j = 0, 1, \ldots$

1. $A_j \leftarrow A - BB^T X_j =: A - BK_j.$
2. Solve the Lyapunov equation $A_j^T N_j + N_j A_j = -\mathcal{R}(X_j).$
3. $X_{j+1} \leftarrow X_j + t_j N_j.$

END FOR j

Newton-ADI for AREs

Newton's Method for AREs [Kleinman '68, Mehrmann '91, Lancaster/Rodman '95, B./Byers '94/'98, B. '97, Guo/Laub '99]

- Consider $0 = \mathcal{R}(X) = C^T C + A^T X + XA - XBB^T X.$
- Frechét derivative of $\mathcal{R}(X)$ at X:

$$\mathcal{R}'_X : Z \to (A - BB^T X)^T Z + Z(A - BB^T X).$$

- Newton-Kantorovich method:

$$X_{j+1} = X_j - \left(\mathcal{R}'_{X_j}\right)^{-1} \mathcal{R}(X_j), \quad j = 0, 1, 2, \ldots$$

Newton's method (with line search) for AREs

FOR $j = 0, 1, \ldots$

1. $A_j \leftarrow A - BB^T X_j =: A - BK_j.$
2. Solve the Lyapunov equation $A_j^T N_j + N_j A_j = -\mathcal{R}(X_j).$
3. $X_{j+1} \leftarrow X_j + t_j N_j.$

END FOR j

Factored Galerkin-ADI Iteration

Properties and Implementation

- Convergence for K_0 stabilizing:
  - $A_j = A - BK_j = A - BB^T X_j$ is stable $\forall\ j \geq 0$.
  - $\lim_{j\to\infty} \|\mathcal{R}(X_j)\|_F = 0$ (monotonically).
  - $\lim_{j\to\infty} X_j = X_* \geq 0$ (locally quadratic).
- Need large-scale Lyapunov solver; here, ADI iteration: linear systems with dense, but "sparse+low rank" coefficient matrix A_j:

$$A_j \quad = \quad A \quad - \quad B \quad \cdot \quad K_j$$

$$= \quad \boxed{\text{sparse}} \quad - \quad \boxed{m} \quad \cdot \quad \boxed{}$$

- $m \ll n \implies$ efficient "inversion" using Sherman-Morrison-Woodbury formula:

$$(A - BK_j + p_k^{(j)} I)^{-1} = (I_n + (A + p_k^{(j)} I)^{-1} B (I_m - K_j (A + p_k^{(j)} I)^{-1} B)^{-1} K_j)(A + p_k^{(j)} I)^{-1}.$$

- BUT: $X = X^T \in \mathbb{R}^{n\times n} \implies n(n+1)/2$ unknowns!

Factored Galerkin-ADI Iteration

Properties and Implementation

- Convergence for K_0 stabilizing:
  - $A_j = A - BK_j = A - BB^T X_j$ is stable $\forall\ j \geq 0$.
  - $\lim_{j\to\infty} \|\mathcal{R}(X_j)\|_F = 0$ (monotonically).
  - $\lim_{j\to\infty} X_j = X_* \geq 0$ (locally quadratic).
- Need large-scale Lyapunov solver; here, ADI iteration:
  linear systems with dense, but "sparse+low rank" coefficient matrix A_j:

  $A_j \;=\; A \;-\; B \;\cdot\; K_j$

  $\;\;\;\;= \;$ sparse $\;-\;$ m $\;\cdot\;$ [ ]
- $m \ll n \implies$ efficient "inversion" using Sherman-Morrison-Woodbury formula:

$$(A - BK_j + p_k^{(j)} I)^{-1} = (I_n + (A + p_k^{(j)} I)^{-1} B (I_m - K_j (A + p_k^{(j)} I)^{-1} B)^{-1} K_j)(A + p_k^{(j)} I)^{-1}.$$

- BUT: $X = X^T \in \mathbb{R}^{n\times n} \implies n(n+1)/2$ unknowns!

Factored Galerkin-ADI Iteration

Properties and Implementation

- Convergence for K_0 stabilizing:
  - $A_j = A - BK_j = A - BB^T X_j$ is stable $\forall\ j \geq 0$.
  - $\lim_{j\to\infty} \|\mathcal{R}(X_j)\|_F = 0$ (monotonically).
  - $\lim_{j\to\infty} X_j = X_* \geq 0$ (locally quadratic).
- Need large-scale Lyapunov solver; here, ADI iteration: linear systems with dense, but "sparse+low rank" coefficient matrix A_j:

$$A_j \quad = \quad A \quad - \quad B \quad \cdot \quad K_j$$

$$= \quad \boxed{\text{sparse}} \quad - \quad \boxed{m} \quad \cdot \quad \boxed{}$$

- $m \ll n \implies$ efficient "inversion" using Sherman-Morrison-Woodbury formula:

$$(A - BK_j + p_k^{(j)} I)^{-1} = (I_n + (A + p_k^{(j)} I)^{-1} B (I_m - K_j (A + p_k^{(j)} I)^{-1} B)^{-1} K_j)(A + p_k^{(j)} I)^{-1}.$$

- BUT: $X = X^T \in \mathbb{R}^{n\times n} \implies n(n+1)/2$ unknowns!

- Convergence for K_0 stabilizing:
  - $A_j = A - BK_j = A - BB^T X_j$ is stable $\forall\ j \geq 0$.
  - $\lim_{j\to\infty} \|\mathcal{R}(X_j)\|_F = 0$ (monotonically).
  - $\lim_{j\to\infty} X_j = X_* \geq 0$ (locally quadratic).
- Need large-scale Lyapunov solver; here, ADI iteration: linear systems with dense, but "sparse+low rank" coefficient matrix A_j:

$$A_j \quad = \quad A \quad - \quad B \quad \cdot \quad K_j$$

= 

- $m \ll n \implies$ efficient "inversion" using Sherman-Morrison-Woodbury formula:

$$(A - BK_j + p_k^{(j)} I)^{-1} = (I_n + (A + p_k^{(j)} I)^{-1} B (I_m - K_j (A + p_k^{(j)} I)^{-1} B)^{-1} K_j)(A + p_k^{(j)} I)^{-1}.$$

- BUT: $X = X^T \in \mathbb{R}^{n\times n} \implies n(n+1)/2$ unknowns!

Low-Rank Newton-ADI for AREs

Re-write Newton's method for AREs

$$A_j^T N_j + N_j A_j = -\mathcal{R}(X_j)$$
$$\Longleftrightarrow$$
$$A_j^T \underbrace{(X_j + N_j)}_{=X_{j+1}} + \underbrace{(X_j + N_j)}_{=X_{j+1}} A_j = \underbrace{-C^T C - X_j B B^T X_j}_{=:-W_j W_j^T}$$

Set $X_j = Z_j Z_j^T$ for $\operatorname{rank}(Z_j) \ll n \Longrightarrow$

$$A_j^T \left(Z_{j+1} Z_{j+1}^T\right) + \left(Z_{j+1} Z_{j+1}^T\right) A_j = -W_j W_j^T$$

Factored Newton Iteration [B./Li/Penzl 1999/2008]

Solve Lyapunov equations for Z_{j+1} directly by factored ADI iteration and use 'sparse + low-rank' structure of A_j.

Low-Rank Newton-ADI for AREs

Re-write Newton's method for AREs

$$A_j^T N_j + N_j A_j = -\mathcal{R}(X_j)$$

$$\Longleftrightarrow$$

$$A_j^T \underbrace{(X_j + N_j)}_{=X_{j+1}} + \underbrace{(X_j + N_j)}_{=X_{j+1}} A_j = \underbrace{-C^T C - X_j B B^T X_j}_{=:-W_j W_j^T}$$

Set $X_j = Z_j Z_j^T$ for $\operatorname{rank}(Z_j) \ll n \Longrightarrow$

$$A_j^T \left(Z_{j+1} Z_{j+1}^T\right) + \left(Z_{j+1} Z_{j+1}^T\right) A_j = -W_j W_j^T$$

Factored Newton Iteration [B./Li/Penzl 1999/2008]

Solve Lyapunov equations for Z_{j+1} directly by factored ADI iteration and use 'sparse + low-rank' structure of A_j.

Application to LQR Problem
Feedback Iteration

Optimal feedback

$$K_* = B^T X_* = B^T Z_* Z_*^T$$

can be computed by direct feedback iteration:

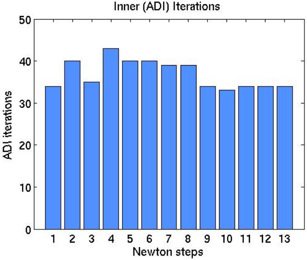
- jth Newton iteration:

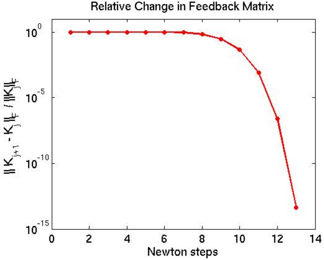
$$K_j = B^T Z_j Z_j^T = \sum_{k=1}^{k_{\max}} (B^T V_{j,k}) V_{j,k}^T \xrightarrow{j\to\infty} K_* = B^T Z_* Z_*^T$$

- K_j can be updated in ADI iteration, no need to even form Z_j, need only fixed workspace for $K_j \in \mathbb{R}^{m \times n}$!

Related to earlier work by [Banks/Ito 1991].

- Linear 2D heat equation with homogeneous Dirichlet boundary and point control/observation.
- FD discretization on uniform 150×150 grid.
- $n = 22.500$, $m = p = 1$, 10 shifts for ADI iterations.
- Convergence of large-scale matrix equation solvers:





Newton-ADI for AREs

Performance of matrix equation solvers

Performance of Newton's method for accuracy $\sim 1/n$

grid	unknowns	$\frac{\Vert\mathcal{R}(X)\Vert_F}{\Vert X\Vert_F}$	it. (ADI it.)	CPU (sec.)
8×8	2,080	4.7e-7	2 (8)	0.47
16×16	32,896	1.6e-6	2 (10)	0.49
32×32	524,800	1.8e-5	2 (11)	0.91
64×64	8,390,656	1.8e-5	3 (14)	7.98
128×128	134,225,920	3.7e-6	3 (19)	79.46

Here,

- Convection-diffusion equation,
- $m = 1$ input and $p = 2$ outputs,
- $X = X^T \in \mathbb{R}^{n\times n} \Rightarrow \frac{n(n+1)}{2}$ unknowns.

Confirms mesh independence principle for Newton-Kleinman [Burns/Sachs/Zietsman '08].

Performance of Newton's method for accuracy $\sim 1/n$

grid	unknowns	$\frac{\lVert\mathcal{R}(X)\rVert_F}{\lVert X\rVert_F}$	it. (ADI it.)	CPU (sec.)
8×8	2,080	4.7e-7	2 (8)	0.47
16×16	32,896	1.6e-6	2 (10)	0.49
32×32	524,800	1.8e-5	2 (11)	0.91
64×64	8,390,656	1.8e-5	3 (14)	7.98
128×128	134,225,920	3.7e-6	3 (19)	79.46

Here,

- Convection-diffusion equation,
- $m = 1$ input and $p = 2$ outputs,
- $X = X^T \in \mathbb{R}^{n \times n} \Rightarrow \frac{n(n+1)}{2}$ unknowns.

Confirms mesh independence principle for Newton-Kleinman [Burns/Sachs/Zietsman '08].

Recent Numerical Results

Computations by Jens Saak

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.

Recent Numerical Results

Computations by Jens Saak

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n\times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.

Newton-ADI

NWT	rel. change	rel. residual	ADI
1	1	9.99e–01	200
2	9.99e–01	3.41e+01	23
3	5.25e–01	6.37e+00	20
4	5.37e–01	1.52e+00	20
5	7.03e–01	2.64e–01	23
6	5.57e–01	1.56e–02	23
7	6.59e–02	6.30e–05	23
8	4.02e–04	9.68e–10	23
9	8.45e–09	1.09e–11	23
10	1.52e–14	1.09e–11	23

CPU time: 76.9 sec.

Recent Numerical Results

Computations by Jens Saak

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.

Newton-ADI

NWT	rel. change	rel. residual	ADI
1	1	9.99e–01	200
2	9.99e–01	3.41e+01	23
3	5.25e–01	6.37e+00	20
4	5.37e–01	1.52e+00	20
5	7.03e–01	2.64e–01	23
6	5.57e–01	1.56e–02	23
7	6.59e–02	6.30e–05	23
8	4.02e–04	9.68e–10	23
9	8.45e–09	1.09e–11	23
10	1.52e–14	1.09e–11	23

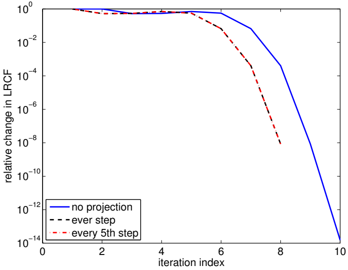
CPU time: 76.9 sec.

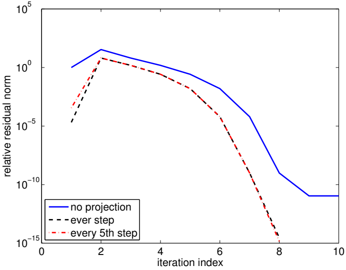
Newton-Galerkin-ADI

NWT	rel. change	rel. residual	ADI
1	1	3.56e–04	20
2	5.25e–01	6.37e+00	10
3	5.37e–01	1.52e+00	6
4	7.03e–01	2.64e–01	10
5	5.57e–01	1.57e–02	10
6	6.59e–02	6.30e–05	10
7	4.03e–04	9.79e–10	10
8	8.45e–09	1.43e–15	10

CPU time: 38.0 sec.

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.





Recent Numerical Results

Computations by Jens Saak

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.

Newton-ADI

NWT	rel. change	rel. residual	ADI
1	1	9.99e–01	200
2	9.99e–01	3.56e+01	60
3	3.11e–01	3.72e+00	39
4	2.88e–01	9.62e–01	40
5	3.41e–01	1.68e–01	45
6	1.22e–01	5.25e–03	42
7	3.88e–03	2.96e–06	47
8	2.30e–06	6.09e–13	47

CPU time: 185.9 sec.

Recent Numerical Results

Computations by Jens Saak

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.

Newton-ADI

NWT	rel. change	rel. residual	ADI
1	1	9.99e–01	200
2	9.99e–01	3.56e+01	60
3	3.11e–01	3.72e+00	39
4	2.88e–01	9.62e–01	40
5	3.41e–01	1.68e–01	45
6	1.22e–01	5.25e–03	42
7	3.88e–03	2.96e–06	47
8	2.30e–06	6.09e–13	47

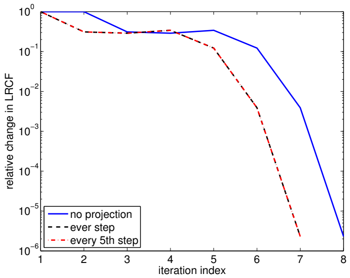
CPU time: 185.9 sec.

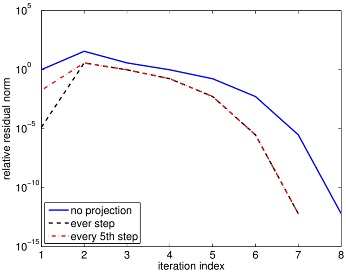
Newton-Galerkin-ADI

step	rel. change	rel. residual	ADI it.
1	1	1.78e–02	35
2	3.11e–01	3.72e+00	15
3	2.88e–01	9.62e–01	20
4	3.41e–01	1.68e–01	15
5	1.22e–01	5.25e–03	20
6	3.89e–03	2.96e–06	15
7	2.30e–06	6.14e–13	20

CPU time: 75.7 sec.

- FDM for 2D heat/convection-diffusion equations on $[0,1]^2$ (LYAPACK benchmarks, $m = p = 1$) $\rightsquigarrow$ symmetric/nonsymmetric $A \in \mathbb{R}^{n \times n}$, $n = 10,000$.
- 15 shifts chosen by Penzl's heuristic from 50/25 Ritz/harmonic Ritz values of A.
- Computations using Intel Core 2 Quad CPU of type Q9400 at 2.66GHz with 4 GB RAM and 64Bit-MATLAB.





Quadratic ADI for AREs

$0 = \mathcal{R}(X) = A^T X + XA - XBB^T X + W$

Basic QADI iteration [Wong/Balakrishnan et al. '05–'08]

$$\begin{aligned} \left((A - BB^T X_k)^T + p_k I\right) X_{k+\frac{1}{2}} &= -W - X_k((A - p_k I) \\ \left((A - BB^T X_{k+\frac{1}{2}}^T)^T + p_k I\right) X_{k+1} &= -W - X_{k+\frac{1}{2}}^T (A - p_k I) \end{aligned}$$

Derivation of complicated Cholesky factor version, but requires square and invertible Cholesky factors.

Idea of low-rank Galerkin-QADI [B./Saak '09]

$$V_1 \leftarrow \sqrt{-2\mathrm{Re}\,(p_1)}(A - B(B^T Y_0)Y_0^T + p_1 I)^{-T} B, \qquad Y_1 \leftarrow V_1$$

FOR $k = 2, 3, \ldots$

$$V_k \leftarrow V_{k-1} - (p_k + \overline{p_{k-1}})(A - B(B^T Y_{k-1})Y_{k-1}^T + p_k I)^{-T} V_{k-1}$$

$$Y_k \leftarrow \left[\begin{array}{cc} Y_{k-1} & \sqrt{\frac{\mathrm{Re}\,(p_k)}{\mathrm{Re}\,(p_{k-1})}} V_k \end{array}\right]$$

$Y_k \leftarrow \mathrm{rrlq}(Y_k, \tau)$ % column compression

If desired, project ARE onto $\mathrm{range}(Y_k)$, solve and prolongate.

Quadratic ADI for AREs

$0 = \mathcal{R}(X) = A^T X + XA - XBB^T X + W$

Basic QADI iteration [Wong/Balakrishnan et al. '05–'08]

$$\begin{aligned} \left((A - BB^T X_k)^T + p_k I\right) X_{k+\frac{1}{2}} &= -W - X_k((A - p_k I) \\ \left((A - BB^T X_{k+\frac{1}{2}}^T)^T + p_k I\right) X_{k+1} &= -W - X_{k+\frac{1}{2}}^T (A - p_k I) \end{aligned}$$

Derivation of complicated Cholesky factor version, but requires square and invertible Cholesky factors.

Idea of low-rank Galerkin-QADI [B./Saak '09]

$V_1 \leftarrow \sqrt{-2\mathrm{Re}\,(p_1)}(A - B(B^T Y_0) Y_0^T + p_1 I)^{-T} B, \qquad Y_1 \leftarrow V_1$

FOR $k = 2, 3, \ldots$

$V_k \leftarrow V_{k-1} - (p_k + \overline{p_{k-1}})(A - B(B^T Y_{k-1}) Y_{k-1}^T + p_k I)^{-T} V_{k-1}$

$Y_k \leftarrow \left[\; Y_{k-1} \quad \sqrt{\frac{\mathrm{Re}\,(p_k)}{\mathrm{Re}\,(p_{k-1})}} V_k \;\right]$

$Y_k \leftarrow \mathrm{rrlq}(Y_k, \tau)$ % column compression

If desired, project ARE onto $\mathrm{range}(Y_k)$, solve and prolongate.

AREs with High-Rank Constant Term

Consider ARE

$$0 = \mathcal{R}(X) = W + A^T X + XA - XBB^T X$$

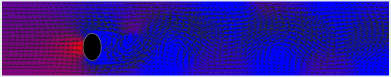
with $\operatorname{rank}(W) \not\ll n$, e.g., stabilization of flow problems described by Navier-Stokes eqns. requires solution of

$$0 = \mathcal{R}(X) = M_h - S_h^T X M_h - M_h X S_h - M_h X B_h B_h^T X M_h,$$

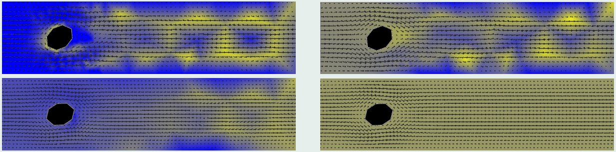
where M_h = mass matrix of FE velocity test functions.

Example: von Kármán vortex street, Re= 500

uncontrolled:



controlled using ARE:



$t = 1$ $t = 5$

$t = 8$ $t = 10$

AREs with High-Rank Constant Term

Solution: remove W from r.h.s. of Lyapunov eqns. in Newton-ADI

One step of Newton-Kleinman iteration for ARE:

$$A_j^T \underbrace{(X_j + N_j)}_{=X_{j+1}} + X_{j+1} A_j = -W - \underbrace{(X_j B)}_{=K_j^T} \underbrace{B^T X_j}_{=K_j} \qquad \text{for } j = 1, 2, \ldots$$

Subtract two consecutive equations $\Longrightarrow$

$$A_j^T N_j + N_j A_j = -N_{j-1}^T B B^T N_{j-1} \qquad \text{for } j = 1, 2, \ldots$$

See [Banks/Ito '91, B./Hernández/Pastor '03, Morris/Navasca '05] for details and applications of this variant.

But: need $B^T N_0 = K_1 - K_0$!

Assuming K_0 is known, need to compute K_1.

Solution idea:

$$
\begin{aligned}
K_1 &= B^T X_1 \\
&= B^T \int_0^\infty e^{(A-BK_0)^T t}(W + K_0^T K_0)e^{(A-BK_0)t}\,dt \\
&= \int_0^\infty g(t)\,dt \approx \sum_{\ell=0}^{N} \gamma_\ell g(t_\ell),
\end{aligned}
$$

where $g(t) = \left(\left(e^{(A-BK_0)t}B\right)^T (W + K_0^T K_0)\right) e^{(A-BK_0)t}$.

[Borggaard/Stoyanov '08]:
evaluate $g(t_\ell)$ using ODE solver applied to $\dot{x} = (A-BK_0)x$ + adjoint eqn.

Better solution idea:

(related to frequency domain POD [Willcox/Peraire '02])

$$
\begin{aligned}
K_1 &= B^T X_1 \qquad \text{(Notation: } A_0 := A - BK_0\text{)} \\
&= B^T \cdot \frac{1}{2\pi} \int_{-\infty}^{\infty} (\jmath\omega I_n - A_0)^{-H} (W + K_0^T K_0)(\jmath\omega I_n - A_0)^{-1}\, d\omega \\
&= \int_{-\infty}^{\infty} f(\omega)\, d\omega \approx \sum_{\ell=0}^{N} \gamma_\ell f(\omega_\ell),
\end{aligned}
$$

where $f(\omega) = \left(-\left((\jmath\omega I_n + A_0)^{-1} B\right)^T (W + K_0^T K_0)\right) (\jmath\omega I_n - A_0)^{-1}$.

Evaluation of $f(\omega_\ell)$ requires

- 1 sparse LU decmposition (complex!),
- $2m$ forward/backward solves,
- m sparse and $2m$ low-rank matrix-vector products.

Use adaptive quadrature with high accuracy, e.g. Gauß-Kronrod (Matlab's `quadgk`).

AREs with Indefinite Hessian

Now:

$$\mathcal{R}(X) := C^T C + A^T X + XA + X(B_1 B_1^T - B_2 B_2^T)X = 0.$$

AREs with Indefinite Hessian

Now:

$$\mathcal{R}(X) := C^T C + A^T X + XA + X(B_1 B_1^T - B_2 B_2^T)X = 0.$$

Problems

- For large-scale problems, resulting, e.g., from H_∞ control, standard methods based on Hamiltonian/even eigenvalue problem can not be used due to $\mathcal{O}(n^3)$ complexity/dense matrix algebra.
- Krylov subspace methods might be employed, but so far no convergence results, and in case of convergence, no guarantee that stabilizing solution is computed.
- Newton/Newton-ADI method will in general diverge/converge to a non-stabilizing solution.

AREs with Indefinite Hessian

Now:

$$\mathcal{R}(X) := C^T C + A^T X + XA + X(B_1 B_1^T - B_2 B_2^T)X = 0.$$

Problems

- For large-scale problems, resulting, e.g., from H_∞ control, standard methods based on Hamiltonian/even eigenvalue problem can not be used due to $\mathcal{O}(n^3)$ complexity/dense matrix algebra.
- Krylov subspace methods might be employed, but so far no convergence results, and in case of convergence, no guarantee that stabilizing solution is computed.
- Newton/Newton-ADI method will in general diverge/converge to a non-stabilizing solution.

AREs with Indefinite Hessian

Now:

$$\mathcal{R}(X) := C^T C + A^T X + XA + X(B_1 B_1^T - B_2 B_2^T)X = 0.$$

Problems

- For large-scale problems, resulting, e.g., from H_∞ control, standard methods based on Hamiltonian/even eigenvalue problem can not be used due to $\mathcal{O}(n^3)$ complexity/dense matrix algebra.
- Krylov subspace methods might be employed, but so far no convergence results, and in case of convergence, no guarantee that stabilizing solution is computed.
- Newton/Newton-ADI method will in general diverge/converge to a non-stabilizing solution.

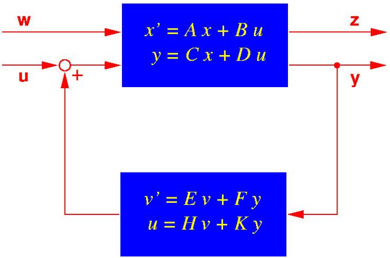
Motivation: H_∞-Control

Linear time-invariant systems

$$\Sigma : \begin{cases} \dot{x} &= Ax + B_1 w + B_2 u, \\ z &= C_1 x + D_{11} w + D_{12} u, \\ y &= C_2 x + D_{21} w + D_{22} u, \end{cases}$$

where $A \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times m_k}$, $C_j \in \mathbb{C}^{p_j\times n}$, $D_{jk} \in \mathbb{R}^{p_j\times m_k}$.

- x – states of the system,
- w – exogenous inputs
- u – control inputs,
- z – performance outputs
- y – measured outputs



Laplace transform $\Longrightarrow$ transfer function (in frequency domain)

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \equiv \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right].$$

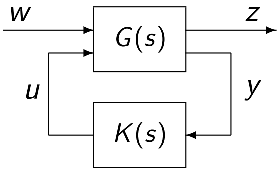
where for $x(0) = 0$, G_{ij} are the rational matrix functions

- $G_{11}(s) = C_1(sI - A)^{-1}B_1 + D_{11}$,
- $G_{12}(s) = C_1(sI - A)^{-1}B_2 + D_{12}$,
- $G_{21}(s) = C_2(sI - A)^{-1}B_1 + D_{21}$,
- $G_{22}(s) = C_2(sI - A)^{-1}B_2 + D_{22}$,

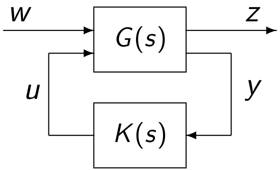
describing the transfer from inputs to outputs of Σ via

$$\begin{aligned} z(s) &= G_{11}(s)w(s) + G_{12}(s)u(s), \\ y(s) &= G_{21}(s)w(s) + G_{22}(s)u(s). \end{aligned}$$

Consider closed-loop system, where $K(s)$ is an internally stabilizing controller, i.e., K stabilizes G for $w \equiv 0$.



Consider closed-loop system, where $K(s)$ is an internally stabilizing controller, i.e., K stabilizes G for $w \equiv 0$.



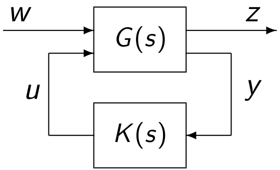
Goal:

find K that minimize error outputs

$$z = \left(G_{11} + G_{12}K(I - G_{22}K)^{-1}G_{21}\right) w =: \mathcal{F}(G, K)w,$$

where $\mathcal{F}(G, K)$ is the linear fractional transformation of G, K.

Consider closed-loop system, where $K(s)$ is an internally stabilizing controller, i.e., K stabilizes G for $w \equiv 0$.



Goal:

find K that minimize error outputs

$$z = \left(G_{11} + G_{12}K(I - G_{22}K)^{-1}G_{21}\right) w =: \mathcal{F}(G, K)w,$$

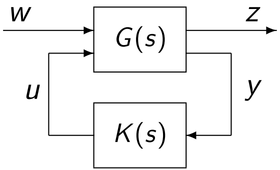
where $\mathcal{F}(G, K)$ is the linear fractional transformation of G, K.

H_∞-optimal control problem:

$$\min_{K \text{ stabilizing}} \|\mathcal{F}(G, K)\|_{\mathcal{H}_\infty}.$$

H_∞-Control
The H_∞-Optimization Problem

Consider closed-loop system, where $K(s)$ is an internally stabilizing controller, i.e., K stabilizes G for $w \equiv 0$.



Goal:

find K that minimize error outputs

$$z = \left(G_{11} + G_{12}K(I - G_{22}K)^{-1}G_{21}\right) w =: \mathcal{F}(G, K)w,$$

where $\mathcal{F}(G, K)$ is the linear fractional transformation of G, K.

H_∞-suboptimal control problem:

For given constant $\gamma > 0$, find all internally stabilizing controllers satisfying

$$\|\mathcal{F}(G, K)\|_{\mathcal{H}_\infty} < \gamma.$$

Simplifying assumptions

1. $D_{11} = 0$;
2. $D_{22} = 0$;
3. (A, B_1) stabilizable, (C_1, A) detectable;
4. (A, B_2) stabilizable, (C_2, A) detectable ($\Longrightarrow \Sigma$ internally stabilizable);
5. $D_{12}^T [\, C_1 \;\; D_{12} \,] = [\, 0 \;\; I_{m_2} \,]$;
6. $\begin{bmatrix} B_1 \\ D_{21} \end{bmatrix} D_{21}^T = \begin{bmatrix} 0 \\ I_{p_2} \end{bmatrix}$.

Remark. 1.,2.,5.,6. only for notational convenience, 3. can be relaxed, but derivations get even more complicated.

Theorem [Doyle/Glover/Khargonekar/Francis '89]

Given the Assumptions 1.–6., there exists an admissible controller $K(s)$ solving the H_∞-suboptimal control problem $\iff$

(i) There exists a solution $X_\infty = X_\infty^T \geq 0$ to the ARE

$$C_1C_1^T + A^TX + XA + X(\gamma^{-2}B_1B_1^T - B_2B_2^T)X = 0, \tag{1}$$

such that A_X is Hurwitz, where $A_X := A + (\gamma^{-2}B_1B_1^T - B_2B_2^T)X_\infty$.

(ii) There exists a solution $Y_\infty = Y_\infty^T \geq 0$ to the ARE

$$B_1B_1^T + AY + YA^T + Y(\gamma^{-2}C_1C_1^T - C_2C_2^T)Y = 0, \tag{2}$$

such that A_Y is Hurwitz where $A_Y := A + Y_\infty(\gamma^{-2}C_1C_1^T - C_2C_2^T)$.

(iii) $\gamma^2 > \rho(X_\infty Y_\infty)$.

H_∞-optimal control

Find minimal γ for which (i)–(iii) are satisfied $\rightsquigarrow$ γ-iteration based on solving AREs (1)–(2) repeatedly for different γ.

Theorem [Doyle/Glover/Khargonekar/Francis '89]

Given the Assumptions 1.–6., there exists an admissible controller $K(s)$ solving the H_∞-suboptimal control problem $\iff$

(i) There exists a solution $X_\infty = X_\infty^T \geq 0$ to the ARE

$$C_1C_1^T + A^TX + XA + X(\gamma^{-2}B_1B_1^T - B_2B_2^T)X = 0, \qquad (1)$$

such that A_X is Hurwitz, where $A_X := A + (\gamma^{-2}B_1B_1^T - B_2B_2^T)X_\infty$.

(ii) There exists a solution $Y_\infty = Y_\infty^T \geq 0$ to the ARE

$$B_1B_1^T + AY + YA^T + Y(\gamma^{-2}C_1C_1^T - C_2C_2^T)Y = 0, \qquad (2)$$

such that A_Y is Hurwitz where $A_Y := A + Y_\infty(\gamma^{-2}C_1C_1^T - C_2C_2^T)$.

(iii) $\gamma^2 > \rho(X_\infty Y_\infty)$.

H_∞-optimal control

Find minimal γ for which (i)–(iii) are satisfied $\rightsquigarrow$ γ-iteration based on solving AREs (1)–(2) repeatedly for different γ.

H_∞-(sub-)optimal controller

If (i)–(iii) hold, a suboptimal controller is given by

$$\hat{K}(s) = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & 0 \end{array}\right] = \hat{C}(sI_n - \hat{A})^{-1}\hat{B},$$

where for

$$Z_\infty := (I - \gamma^{-2} Y_\infty X_\infty)^{-1},$$

$$\begin{aligned} \hat{A} &:= A + (\gamma^{-2} B_1 B_1^T - B_2 B_2^T) X_\infty - Z_\infty Y_\infty C_2^T C_2, \\ \hat{B} &:= Z_\infty Y_\infty C_2^T, \\ \hat{C} &:= -B_2^T X_\infty. \end{aligned}$$

$\hat{K}(s)$ is the central or minimum entropy controller.

ARE with indefinite Hessian

$$0 = \mathcal{R}(X) := C^T C + A^T X + XA + X(B_1 B_1^T - B_2 B_2^T)$$

Consider $X^{-1}\mathcal{R}(X)X^{-1} = 0$

$\rightsquigarrow$ standard ARE for $\tilde{X} \equiv X^{-1}$

$$\tilde{\mathcal{R}}(\tilde{X}) := (B_1 B_1^T - B_2 B_2^T) + \tilde{X}A^T + A\tilde{X} + \tilde{X}C^T C\tilde{X} = 0.$$

Newton's method will converge to stabilizing solution, Newton-ADI can be employed (with modification for indefinite constant term).

But: low-rank approximation of $\tilde{X}$ will not yield good approximation of X $\Rightarrow$ not feasible for large-scale problems!

Lyapunov Iterations/Perturbed Hessian Approach

[Cherfi/Abou-Kandil/Bourles 2005 (Proc. ACSE 2005)]

Idea

Perturb Hessian to enforce semi-definiteness: write

$$0 = A^T X + XA + Q - XGX = A^T X + XA + Q - XDX + X(D - G)X,$$

where $D = G + \alpha I \geq 0$ with $\alpha \geq \min\{0, -\lambda_{\max}(G)\}$.

Lyapunov Iterations/Perturbed Hessian Approach

[Cherfi/Abou-Kandil/Bourles 2005 (Proc. ACSE 2005)]

Idea

Perturb Hessian to enforce semi-definiteness: write

$$0 = A^T X + XA + Q - XGX = A^T X + XA + Q - XDX + X(D-G)X,$$

where $D = G + \alpha I \geq 0$ with $\alpha \geq \min\{0, -\lambda_{\max}(G)\}$.

Here: $G = B_2 B_2^T - B_1 B_1^T$

$\Rightarrow$ use $\alpha = \|B_1\|^2$ for spectral/Frobenius norm or

$$\alpha = \|B_1\|_1 \cdot \|B_1\|_\infty.$$

Remark

$W \geq -G$ can be used instead of αI, e.g., $W = \beta B_1 B_1^T$ with $\beta \geq 1$.

Lyapunov Iterations/Perturbed Hessian Approach

[CHERFI/ABOU-KANDIL/BOURLES 2005 (Proc. ACSE 2005)]

Idea

Perturb Hessian to enforce semi-definiteness: write

$$0 = A^T X + XA + Q - XGX = A^T X + XA + Q - XDX + X(D-G)X,$$

where $D = G + \alpha I \geq 0$ with $\alpha \geq \min\{0, -\lambda_{\max}(G)\}$.

Lyapunov iteration

Based on

$$(A - DX)^T X + X(A - DX) = -Q - XDX - \alpha X^2,$$

iterate

FOR $k = 0, 1, \ldots$, solve Lyapunov equation

$$(A - DX_k)^T X_{k+1} + X_{k+1}(A - DX_k) = -Q - X_k D X_k - \alpha X_k^2.$$

Lyapunov Iterations/Perturbed Hessian Approach

[Cherfi/Abou-Kandil/Bourles 2005 (Proc. ACSE 2005)]

Lyapunov iteration

FOR $k = 0, 1, \ldots,$ solve Lyapunov equation

$$(A - DX_k)^T X_{k+1} + X_{k+1}(A - DX_k) = -Q - X_k D X_k - \alpha X_k^2.$$

Easy to convert to low-rank iteration employing low-rank ADI for Lyapunov equations, e.g. with $W = B_1 B_1^T$ instead of αI: the Lyapunov equation becomes

$$\begin{aligned}(A - B_2 B_2^T Y_k Y_k)^T Y_{k+1} Y_{k+1}^T + Y_{k+1} Y_{k+1}^T (A - B_2 B_2^T Y_k Y_k) \\ = -CC^T - Y_k Y_k^T B_1 B_1^T Y_k Y_k^T - Y_k Y_k^T B_2 B_2^T Y_k Y_k^T \\ = -[\, C,\; Y_k Y_k^T B_1,\; Y_k Y_k^T B_2 \,] \begin{bmatrix} C^T \\ B_1^T Y_k Y_k^T \\ B_2^T Y_k Y_k^T \end{bmatrix}.\end{aligned}$$

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Lyapunov Iterations/Perturbed Hessian Approach

Convergence

Theorem [CHERFI/ABOU-KANDIL/BOURLES 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Lyapunov Iterations/Perturbed Hessian Approach
Convergence

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Lyapunov Iterations/Perturbed Hessian Approach
Convergence

Theorem [Cherfi/Abou-Kandil/Bourles 2005]

If

- $\exists\ \hat{X}$ such that $\mathcal{R}(\hat{X}) \geq 0$,
- $\exists\ X_0 = X_0^T \geq \hat{X}$ such that $\mathcal{R}(X_0) \leq 0$ and $A - DX_0$ is Hurwitz,

then

a) $X_0 \geq \ldots \geq X_k \geq X_{k+1} \geq \ldots \geq \hat{X}$,

b) $\mathcal{R}(X_k) \leq 0$ for all $k = 0, 1, \ldots$,

c) $A - DX_k$ is Hurwitz for all $k = 0, 1, \ldots$,

d) $\exists \quad \lim_{k\to\infty} X_k =: \underline{X} \geq \hat{X}$,

e) $\underline{X}$ is semi-stabilizing.

Main problems

- Conditions for initial guess make its computation difficult.
- Observed convergence is linear.

Idea

Consider

$$A^T X + XA + C^T C + X(B_1 B_1^T - B_2 B_2^T)X =: \mathcal{R}(X).$$

Then

$$\begin{aligned}\mathcal{R}(X+Z) &= \mathcal{R}(X) + (\underbrace{A + (B_1 B_1^T - B_2 B_2^T)X}_{=:\widehat{A}})^T Z + Z\widehat{A} \\ &\quad + Z(B_1 B_1^T - B_2 B_2^T)Z.\end{aligned}$$

Furthermore, if $X = X^T$, $Z = Z^T$ solve the standard ARE

$$0 = \mathcal{R}(X) + \widehat{A}^T Z + Z\widehat{A} - ZB_2 B_2^T Z,$$

then

$$\begin{aligned}\mathcal{R}(X+Z) &= ZB_1 B_1^T Z, \\ \|\mathcal{R}(X)\|_2 &= \|B_1^T Z\|_2.\end{aligned}$$

Riccati Iterations

[Lanzon/Feng/B.D.O. Anderson 2007 (Proc. ECC 2007)]

Idea

Consider

$$A^T X + XA + C^T C + X(B_1 B_1^T - B_2 B_2^T)X =: \mathcal{R}(X).$$

Then

$$\begin{aligned}\mathcal{R}(X+Z) &= \mathcal{R}(X) + (\underbrace{A + (B_1 B_1^T - B_2 B_2^T)X}_{=:\widehat{A}})^T Z + Z\widehat{A} \\ &\quad + Z(B_1 B_1^T - B_2 B_2^T)Z.\end{aligned}$$

Furthermore, if $X = X^T$, $Z = Z^T$ solve the standard ARE

$$0 = \mathcal{R}(X) + \widehat{A}^T Z + Z\widehat{A} - ZB_2 B_2^T Z,$$

then

$$\begin{aligned}\mathcal{R}(X+Z) &= ZB_1 B_1^T Z, \\ \|\mathcal{R}(X)\|_2 &= \|B_1^T Z\|_2.\end{aligned}$$

Riccati Iterations
[LANZON/FENG/B.D.O. ANDERSON 2007 (Proc. ECC 2007)]

Idea

Consider

$$A^T X + XA + C^T C + X(B_1 B_1^T - B_2 B_2^T)X =: \mathcal{R}(X).$$

Then

$$\begin{aligned} \mathcal{R}(X+Z) &= \mathcal{R}(X) + (\underbrace{A + (B_1 B_1^T - B_2 B_2^T)X}_{=:\widehat{A}})^T Z + Z\widehat{A} \\ &\quad + Z(B_1 B_1^T - B_2 B_2^T)Z. \end{aligned}$$

Furthermore, if $X = X^T$, $Z = Z^T$ solve the standard ARE

$$0 = \mathcal{R}(X) + \widehat{A}^T Z + Z\widehat{A} - ZB_2 B_2^T Z,$$

then

$$\begin{aligned} \mathcal{R}(X+Z) &= ZB_1 B_1^T Z, \\ \|\mathcal{R}(X)\|_2 &= \|B_1^T Z\|_2. \end{aligned}$$

Riccati Iterations

[Lanzon/Feng/B.D.O. Anderson 2007 (Proc. ECC 2007)]

Riccati iteration

1. Set $X_0 = 0$.
2. FOR $k = 1, 2, \ldots,$
   (i) Set $A_k := A + B_1(B_1^T X_k) - B_2(B_2^T X_k)$.
   (ii) Solve the ARE
   $$\mathcal{R}(X_k) + A_k^T Z_k + Z_k A_k - Z_k B_2 B_2^T Z_k = 0.$$
   (iii) Set $X_{k+1} := X_k + Z_k$.
   (iv) IF $\|B_1^T Z_k\|_2 <$ tol THEN Stop.

Remark. ARE for $k = 0$ is the standard LQR/H_2 ARE.

Riccati Iterations
[Lanzon/Feng/B.D.O. Anderson 2007 (Proc. ECC 2007)]

Theorem [Lanzon/Feng/B.D.O. Anderson 2007]

If

- (A, B_2) stabilizable,
- (A, C) has no unobservable purely imaginary modes, and
- $\exists$ stabilizing solution X_-,

then

a) $(A + B_1 B_1^T X_k, B_2)$ stabilizable for all $k = 0, 1, \ldots,$

b) $Z_k \geq 0$ for all $k = 0, 1, \ldots,$

c) $A + B_1 B_1^T X_k - B_2 B_2^T X_{k+1}$ is Hurwitz for all $k = 0, 1, \ldots,$

d) $\mathcal{R}(X_{k+1}) = Z_k B_1 B_1^T Z_k$ for all $k = 0, 1, \ldots,$

e) $X_- \geq \ldots \geq X_{k+1} \geq X_k \geq \ldots \geq 0.$

f) If $\exists \lim_{k\to\infty} X_k =: \underline{X}$, then $\underline{X} = X_-$, and

g) convergence is locally quadratic.

Riccati Iterations

[Lanzon/Feng/B.D.O. Anderson 2007 (Proc. ECC 2007)]

Riccati iteration – low-rank version [B. 2008]

1. Solve the ARE

$$C^TC + A^TZ_0 + Z_0A - Z_0B_2B_2^TZ_0 = 0$$

using Newton-ADI, yielding Y_0 with $Z_0 \approx Y_0Y_0^T$.

2. Set $R_1 := Y_0$. $\{\%\ R_1R_1^T \approx X_1.\}$

3. FOR $k = 1, 2, \ldots,$
   - (i) Set $A_k := A + B_1(B_1^TR_k)R_k^T - B_2(B_2^TR_k)R_k^T$.
   - (ii) Solve the ARE

$$Y_{k-1}(Y_{k-1}^TB_1)(B_1^TY_{k-1})Y_{k-1}^T + A_k^TZ_k + Z_kA_k - Z_kB_2B_2^TZ_k = 0$$

     using Newton-ADI, yielding Y_k with $Z_k \approx Y_kY_k^T$.
   - (iii) Set $R_{k+1} := \texttt{rrqr}([R_k,\ Y_k], \tau)$. $\{\%\ R_{k+1}R_{k+1}^T \approx X_{k+1}\}$
   - (iv) IF $\|(B_1^TY_k)Y_k^T\|_2 <$ tol THEN Stop.

AREs with Indefinite Hessian

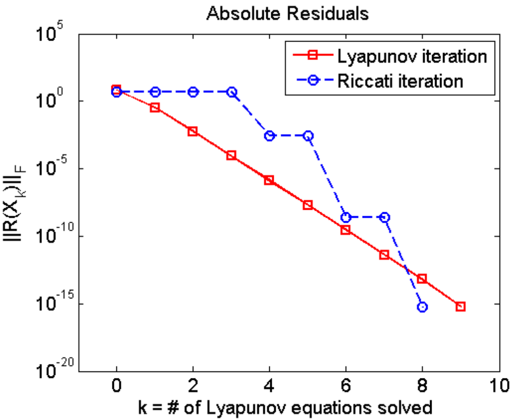
Numerical example

- Trivial example ($n = 2$) from [Cherfi/Abou-Kandil/Bourles 2005].
- Compare convergence of Lyapunov and Riccati iterations.
- Solution of standard AREs with Newton's method.

AREs with Indefinite Hessian

Numerical example

- Trivial example ($n = 2$) from [Cherfi/Abou-Kandil/Bourles 2005].
- Compare convergence of Lyapunov and Riccati iterations.
- Solution of standard AREs with Newton's method.



Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

MESS – Matrix Equations Sparse Solvers [Saak/Mena/B. 2008]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix.
- C version CMESS under development (Martin Köhler).

Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

MESS – Matrix Equations Sparse Solvers [Saak/Mena/B. 2008]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix.
- C version CMESS under development (Martin Köhler).

Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

MESS – Matrix Equations Sparse Solvers [Saak/Mena/B. 2008]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix.
- C version CMESS under development (Martin Köhler).

Conclusions and Open Problems

- Galerkin projection can significantly accelerate ADI iteration for Lyapunov equations.
- Low-rank Newton-ADI is a powerful and reliable method for solving large-scale AREs with semidefinite Hessian.
- Low-rank Galerkin-QADI may become a viable alternative to Newton-ADI.
- High-rank constant terms in ARE can be handled using quadrature rules.
- Software is available in MATLAB toolbox LYAPACK and its successor MESS.
- Low-rank Riccati iteration yields a reliable and efficient method for large-scale AREs with indefinite Hessian, useful, e.g., for H_∞ optimization of PDE control problems.
- Low-rank Lyapunov iteration is an extremely simple variant for large-scale problems, but exhibits slower convergence and requires difficult-to-compute initial value.

Conclusions and Open Problems

- Galerkin projection can significantly accelerate ADI iteration for Lyapunov equations.
- Low-rank Newton-ADI is a powerful and reliable method for solving large-scale AREs with semidefinite Hessian.
- Low-rank Galerkin-QADI may become a viable alternative to Newton-ADI.
- High-rank constant terms in ARE can be handled using quadrature rules.
- Software is available in MATLAB toolbox LYAPACK and its successor MESS.
- Low-rank Riccati iteration yields a reliable and efficient method for large-scale AREs with indefinite Hessian, useful, e.g., for H_∞ optimization of PDE control problems.
- Low-rank Lyapunov iteration is an extremely simple variant for large-scale problems, but exhibits slower convergence and requires difficult-to-compute initial value.

Conclusions and Open Problems

- To-Do list:

  ...for AREs with semidefinite Hessian:
  - computation of stabilizing initial guess.
    (If hierarchical grid structure is available, a multigrid approach is possible, other approaches based on "cheaper" matrix equations under development.)
  - Implementation of coupled Riccati solvers for LQG controller design and balancing-related model reduction.

Conclusions and Open Problems

- To-Do list:

  ...for AREs with semidefinite Hessian:

  - computation of stabilizing initial guess.
    (If hierarchical grid structure is available, a multigrid approach is possible, other approaches based on "cheaper" matrix equations under development.)
  - Implementation of coupled Riccati solvers for LQG controller design and balancing-related model reduction.

  ...for AREs with indefinite Hessian:

  - Implement Riccati iteration in LYAPACK/MESS style.
  - More numerical tests.
  - Re-write Riccati iteration as feedback iteration.
  - Efficient computation of initial value for Lyapunov iterations?
  - $\exists$ perturbed Hessian so that Lyapunov iteration quadratically convergent?

References

1. H.T. Banks and K. Ito.
A numerical algorithm for optimal feedback gains in high dimensional linear quadratic regulator problems.
SIAM J. Cont. Optim., 29(3):499–515, 1991.

2. U. Baur and P. Benner.
Factorized solution of Lyapunov equations based on hierarchical matrix arithmetic.
Computing, 78(3):211–234, 2006.

3. P. Benner.
Solving large-scale control problems.
IEEE Control Systems Magazine, 14(1):44–59, 2004.

4. P. Benner.
Contributions to the Numerical Solution of Algebraic Riccati Equations and Related Eigenvalue Problems.
Logos–Verlag, Berlin, Germany, 1997.
Also: Dissertation, Fakultät für Mathematik, TU Chemnitz–Zwickau, 1997.

5. P. Benner.
Editorial of special issue on "Large-Scale Matrix Equations of Special Type".
Numer. Lin. Alg. Appl., 15(9):747–754, 2008.

6. P. Benner and R. Byers.
Step size control for Newton's method applied to algebraic Riccati equations.
In J.G. Lewis, editor, *Proc. Fifth SIAM Conf. Appl. Lin. Alg., Snowbird, UT*, pages 177–181. SIAM, Philadelphia, PA, 1994.

7. P. Benner and R. Byers.
An exact line search method for solving generalized continuous-time algebraic Riccati equations.
IEEE Trans. Automat. Control, 43(1):101–107, 1998.

8. P. Benner, J.-R. Li, and T. Penzl.
Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic control problems.
Numer. Lin. Alg. Appl., 15(9):755–777, 2008.
Reprint of unpublished manuscript, December 1999.

References

9 J. Borggaard and M. Stoyanov.
A reduced order solver for Lyapunov equations with high rank matrices
Proc. 18th Intl. Symp. Mathematical Theory of Network and Systems, MTNS 2008, 11 pages, 2008.

10 J. Burns, E. Sachs, and L. Zietsman.
Mesh independence of Kleinman-Newton iterations for Riccati equations on Hilbert space.
SIAM J. Control Optim., 47(5):2663–2692, 2008.

11 L. Cherfi, H. Abou-Kandil, and H. Bourles.
Iterative method for general algebraic Riccati equation.
Proc. ICGST Intl. Conf. Automatic Control and System Engineering, ACSE 2005, 19–21 December 2005, Cairo, Egypt.

12 T. Damm.
Rational Matrix Equations in Stochastic Control.
No. 297 in Lecture Notes in Control and Information Sciences.
Springer-Verlag, Berlin/Heidelberg, FRG, 2004.

13 J. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis.
State-space solutions to standard H_2 and H_∞ control problems.
IEEE Trans. Automat. Control, 34:831–847, 1989.

14 L. Grasedyck, W. Hackbusch, and B.N. Khoromskij.
Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices.
Computing, 70:121–165, 2003.

15 C.-H. Guo and A.J. Laub.
On a Newton-like method for solving algebraic Riccati equations.
SIAM J. Matrix Anal. Appl., 21(2):694–698, 2000.

16 I.M. Jaimoukha and E.M. Kasenally.
Krylov subspace methods for solving large Lyapunov equations.
SIAM J. Numer. Anal., 31:227–251, 1994.

17 K. Jbilou.
Block Krylov subspace methods for large continuous-time algebraic Riccati equations.
Numer. Algorithms, 34:339–353, 2003.

References

18 K. Jbilou.
An Arnoldi based algorithm for large algebraic Riccati equations.
Appl. Math. Lett., 19(5):437–444, 2006.

19 K. Jbilou.
ADI preconditioned Krylov methods for large Lyapunov matrix equations.
Preprint, 2008.

20 D.L. Kleinman.
On an iterative technique for Riccati equation computations.
IEEE Trans. Automat. Control, AC-13:114–115, 1968.

21 P. Lancaster and L. Rodman.
The Algebraic Riccati Equation.
Oxford University Press, Oxford, 1995.

22 A. Lanzon, Y. Feng, and B.D.O. Anderson.
An iterative algorithm to solve algebraic Riccati equations with an indefinite quadratic term.
Proc. European Control Conf., ECC 2007, July 2–5, 2007, Kos, Greece, Paper WeA13.4.

23 J.-R. Li and J. White.
Low rank solution of Lyapunov equations.
SIAM J. Matrix Anal. Appl., 24(1):260–280, 2002.

24 V. Mehrmann.
The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution.
Number 163 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Heidelberg, July 1991.

25 T. Penzl.
A cyclic low rank Smith method for large sparse Lyapunov equations.
SIAM J. Sci. Comput., 21(4):1401–1418, 2000.

References

26 T. Penzl.
LYAPACK Users Guide.
Technical Report SFB393/00-33, Sonderforschungsbereich 393 *Numerische Simulation auf massiv parallelen Rechnern*, TU Chemnitz, 09107 Chemnitz, FRG, 2000.
Available from http://www.tu-chemnitz.de/sfb393/sfb00pr.html.

27 Y. Saad.
Numerical Solution of Large Lyapunov Equation.
In M.A. Kaashoek, J.H. van Schuppen, and A.C.M. Ran, editors, *Signal Processing, Scattering, Operator Theory and Numerical Methods*, pages 503–511. Birkhäuser, Basel, 1990.

28 J. Saak, H. Mena, and P. Benner.
Matrix Equation Sparse Solvers (MESS): a MATLAB Toolbox for the Solution of Sparse Large-Scale Matrix Equations.
In preparation.

29 V. Simoncini.
A new iterative method for solving large-scale Lyapunov matrix equations.
SIAM J. Sci. Comput., 29:1268–1288, 2007.

30 E.L. Wachspress.
Iterative solution of the Lyapunov matrix equation.
Appl. Math. Letters, 107:87–90, 1988.

31 K. Willcox and J. Peraire.
Balanced model reduction via the proper orthogonal decomposition.
AIAA J., 40(11):2323, 2002.

32 N. Wong, V. Balakrishnan, C.-K. Koh, and T.S. Ng.
Two algorithms for fast and accurate passivity-preserving model order reduction.
IEEE Trans. CAD Integr. Circuits Syst., 25(10):2062–2075, 2006.

33 N. Wong and V. Balakrishnan.
Fast positive-real balanced truncation via quadratic alternating direction implicit iteration.
IEEE Trans. CAD Integr. Circuits Syst., 26(9):1725–1731, 2007.